One Identity Manager 8.2.1

Konfigurationshandbuch

Copyright 2022 One Identity LLC.

ALLE RECHTE VORBEHALTEN.

Diese Anleitung enthält urheberrechtlich geschützte Informationen. Die in dieser Anleitung beschriebene Software wird unter einer Softwarelizenz oder einer Geheimhaltungsvereinbarung bereitgestellt. Diese Software darf nur in Übereinstimmung mit den Bestimmungen der geltenden Vereinbarung verwendet oder kopiert werden. Kein Teil dieser Anleitung darf ohne die schriftliche Erlaubnis von One Identity LLC in irgendeiner Form oder mit irgendwelchen Mitteln, elektronisch oder mechanisch reproduziert oder übertragen werden, einschließlich Fotokopien und Aufzeichnungen für irgendeinen anderen Zweck als den persönlichen Gebrauch des Erwerbers.

Die Informationen in diesem Dokument werden in Verbindung mit One Identity Produkten bereitgestellt. Durch dieses Dokument oder im Zusammenhang mit dem Verkauf von One Identity LLC Produkten wird keine Lizenz, weder ausdrücklich oder stillschweigend, noch durch Duldung oder anderweitig, an jeglichem geistigen Eigentumsrecht eingeräumt. MIT AUSNAHME DER IN DER LIZENZVEREINBARUNG FÜR DIESES PRODUKT GENANNTEN BEDINGUNGEN ÜBERNIMMT ONE IDENTITY KEINERLEI HAFTUNG UND SCHLIESST JEGLICHE AUSDRÜCKLICHE, IMPLIZIERTE ODER GESETZLICHE GEWÄHRLEISTUNG ODER GARANTIE IN BEZUG AUF IHRE PRODUKTE AUS, EINSCHLIESSLICH, ABER NICHT BESCHRÄNKT AUF DIE IMPLIZITE GEWÄHRLEISTUNG DER ALLGEMEINEN GEBRAUCHSTAUGLICHKEIT, EIGNUNG FÜR EINEN BESTIMMTEN ZWECK ODER NICHTVERLETZUNG VON RECHTEN. IN KEINEM FALL HAFTET ONE IDENTITY FÜR JEGLICHE DIREKTE, INDIREKTE, FOLGE-, STÖRUNGS-, SPEZIELLE ODER ZUFÄLLIGE SCHÄDEN (EINSCHLIESSLICH, OHNE EINSCHRÄNKUNG, SCHÄDEN FÜR VERLUST VON GEWINNEN, GESCHÄFTSUNTERBRECHUNGEN ODER VERLUST VON INFORMATIONEN), DIE AUS DER NUTZUNG ODER UNMÖGLICHKEIT DER NUTZUNG DIESES DOKUMENTS RESULTIEREN, SELBST WENN ONE IDENTITY AUF DIE MÖGLICHKEIT SOLCHER SCHÄDEN HINGEWIESEN HAT. One Identity übernimmt keinerlei Zusicherungen oder Garantien hinsichtlich der Richtigkeit und Vollständigkeit des Inhalts dieses Dokuments und behält sich das Recht vor, Änderungen an Spezifikationen und Produktbeschreibungen jederzeit ohne vorherige Ankündigung vorzunehmen. One Identity verpflichtet sich nicht, die in diesem Dokument enthaltenen Informationen zu aktualisieren.

Wenn Sie Fragen zu Ihrer potenziellen Nutzung dieses Materials haben, wenden Sie sich bitte an:

One Identity LLC.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Besuchen Sie unsere Website (http://www.OneIdentity.com) für regionale und internationale Büro-Adressen.

Patente

One Identity ist stolz auf seine fortschrittliche Technologie. Für dieses Produkt können Patente und anhängige Patente gelten. Für die aktuellsten Informationen über die geltenden Patente für dieses Produkt besuchen Sie bitte unsere Website unter http://www.OneIdentity.com/legal/patents.aspx.

Marken

One Identity und das One Identity Logo sind Marken und eingetragene Marken von One Identity LLC. in den USA und anderen Ländern. Für eine vollständige Liste der One Identity Marken besuchen Sie bitte unsere Website unter www.OneIdentity.com/legal. Alle anderen Marken sind Eigentum der jeweiligen Besitzer.

Legende

WARNUNG: Das Symbol WARNUNG weist auf ein potenzielles Risiko von Körperverletzungen oder Sachschäden hin, für das Sicherheitsvorkehrungen nach Industriestandard empfohlen werden. Dieses Symbol ist oft verbunden mit elektrischen Gefahren bezüglich Hardware.

VORSICHT: Das Symbol VORSICHT weist auf eine mögliche Beschädigung von Hardware oder den möglichen Verlust von Daten hin, wenn die Anweisungen nicht befolgt werden.

One Identity Manager Konfigurationshandbuch
Aktualisiert - 12. Mai 2022, 09:12 Uhr
Version - 8.2.1

Inhalt

Über dieses Handbuch **17**

Softwarearchitektur des One Identity Manager **18**

Arbeiten mit Objekten im One Identity Manager 22

Einfügen, Ändern und Löschen eines Objektes im One Identity Manager 26

Kundenspezifische Änderungen an der One Identity Manager Standard-konfiguration **28**

Dynamisches Nachladen von Änderungen 29

Einzelne Eigenschaften für die Bearbeitung sperren und entsperren 30

Berichte zur Systemkonfiguration 31

Anpassen der One Identity Manager Basiskonfiguration **34**

Übersicht über die Einstellungen einer Datenbank 35

Verbindungsdaten zur Datenbank ändern 38

Konfiguration der Datenbank für eine Test-, Entwicklungs- oder Produktivumgebung 39

Staging-Ebene der Datenbank ändern 40

Spracheinstellungen für die Anzeige und Pflege der Daten 40

Anmeldesprachen aktivieren 41

Standardland für Arbeitszeitermittlung und Feiertagen festlegen 41

Konfigurationsparameter zur Systemkonfiguration 42

Konfigurationsparameter bearbeiten 42

Kundenspezifische Konfigurationsparameter erstellen 43

Eigenschaften der Konfigurationsparameter 44

Optionen für Konfigurationsparameter 46

Grundlagen zum One Identity Manager Schema **47**

Übersicht über das One Identity Manager Schema 48

Datenmodell im Designer anzeigen 50

Spaltenabhängigkeiten aufgrund von Bildungsregeln anzeigen 51

Tabellenarten und Standardspalten im One Identity Manager Datenmodell 52

Hinweise zur Bearbeitung von Tabellendefinitionen und Spaltendefinitionen 56

Tabellendefinitionen 57

Tabellentypen im One Identity Manager 58

Datenbanksichten vom Typ View 59
Datenbanksichten vom Typ Proxy 61
Datenbanksichten vom Typ Union 63
Datenbanksichten vom Typ Read only 66
Tabellenskripte 67
Arbeiten mit Modul Globally Unique Identifier 68
Eindeutige Spalten für Tabellen definieren 70
Löschverzögerung für Objekte festlegen 71
Tabellendefinitionen bearbeiten 73
Eigenschaften von Tabellendefinitionen 73
Customizer für Tabellendefinitionen anzeigen 80
Spaltendefinitionen 81
Bildungsregeln zur Wertebildung 82
Bildungsregeln bearbeiten 83
Ändern einer Spalte verhindern 84
Ausführung von Bildungsregeln limitieren 85
Beispiel für lokale Bildungsregeln innerhalb eines Objektes 86
Beispiel für objektübergreifende Bildungsregeln 86
Spaltenlängen begrenzen 87
Kommastellen für die Anzeige von Werten festlegen 87
Vordefinierte Formatierungstypen verwenden 88
Formatierungsskripte erstellen 90
Spaltenabhängigkeiten zur Wertebestückung 91
Zulässige Werte von Spalten 92
Anforderungen an MVP-Spalten festlegen 93
Bitmasken definieren 94
Konfigurieren von Spalten für die Volltextsuche 95
Skripte zum bedingten Anzeigen und Bearbeiten von Spalten 97
Spaltendefinitionen bearbeiten 99
Eigenschaften von Spaltendefinitionen 99
Tabellenbeziehungen 109
Dynamische Fremdschlüssel 113
Unterstützung von Dateigruppen 115

Bearbeiten der Benutzeroberfläche 118
Objektdefinitionen für die Benutzeroberfläche 119

Auswahlkriterien für Objektdefinitionen ... 120
Verwenden der Anzeigetexte für Objektdefinitionen ... 121
Objektdefinitionen erstellen und bearbeiten ... 122
Eigenschaften von Objektdefinitionen ... 122
Menüführung der Benutzeroberfläche ... 124
Bestandteile einer Menüführung ... 125
Empfehlungen zur Bearbeitung der Menüführung ... 127
Hinweise zum Arbeiten mit dem Oberflächeneditor ... 128
Auswählen der Menüführung zur Bearbeitung ... 128
Komplette Menüführung laden ... 129
Menüführung über eine Anwendung laden ... 129
Menüeinträge direkt laden ... 130
Menüeinträge über Berechtigungsgruppen laden ... 130
Menüführung über eine Where-Klausel laden ... 131
Menüführung während der Bearbeitung simulieren ... 132
Bestehende Menüführung für eine neue Berechtigungsgruppe kopieren ... 133
Neue Menüführung erstellen ... 134
Menüeinträge kopieren ... 135
Neue Menüeinträge erstellen ... 136
Neue Menükategorien erstellen ... 137
Menüeinträge an Anwendungen zuweisen ... 137
Menüeinträge an Berechtigungsgruppen zuweisen ... 138
Allgemeine Eigenschaften der Menüeinträge ... 139
Erstellen der Datenbankabfrage für datenabhängige Menüeinträge ... 142
Rekursive datenabhängige Menüeinträge ... 144
Bearbeiten von Listen ... 144
Anzeigemuster für die Listendarstellung ... 146
Definieren von Einfügewerten ... 147
Verwenden von Links in der Menüführung ... 148
Verwenden von Variablen in der Menüführung ... 149
Variablen erstellen und anzeigen ... 151
Formulare für die Benutzeroberfläche ... 153
Empfehlungen für die Bearbeitung von Formularen ... 153
Bearbeiten von Oberflächenformularen ... 155
Hinweise zum Arbeiten mit dem Formulareditor ... 155

Oberflächenformulare deaktivieren 156
Oberflächenformulare kopieren 157
Oberflächenformulare erstellen 157
Oberflächenformulare an Anwendungen zuweisen 158
Oberflächenformulare an Berechtigungsgruppen zuweisen 159
Oberflächenformulare an Objektdefinitionen zuweisen 160
Auswirkung der Objektdefinitionen auf die Anzeige der Oberflächenformulare 160
Oberflächenformulare an Menüeinträge zuweisen 161
Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen 162
Formulare für kundenspezifische Erweiterungen 163
Hierarchische Darstellung von Daten auf Zuordnungsformularen 165
Standardformulare mit kundenspezifischen Formularen ersetzen 166
Eigenschaften von Oberflächenformularen 167
Formulardefinitionen und Formularvorlagen 169
Bearbeiten von Überblickformularen 179
Erstellen von Überblickformularen 181
Weitere Formularelemente auf Überblickformularen einfügen 183
Besonderheiten bei der Bearbeitung von Überblickformularen 185
Layout der Formularelemente anpassen 186
Vorschau eines Überblickformulars während der Bearbeitung anzeigen 188
Überblickformulare und Formularelemente deaktivieren 188
Formularelemente löschen 189
Überblickformulare löschen 189
Statistiken im One Identity Manager 190
Statistikdefinitionen erstellen und bearbeiten 191
Allgemeine Eigenschaften einer Statistikdefinition 192
Abfrage der Statistikmesswerte 194
Statistikdefinitionen deaktivieren 196
Einbinden von Statistiken in die Benutzeroberfläche 196
Berichte in Statistiken verwenden 197
Einfache Berichte in Statistiken verwenden 199
Diagrammtypen für die Darstellung von Statistiken 200
Beispiele für Statistikdefinitionen 204
Erweitern des Launchpad 208
Empfehlungen für die Erweiterung des Launchpad 209

Aktionen für das Launchpad ... 210
Menüeinträge und Aktionen für das Launchpad erstellen ... 211
Methodendefinitionen für die Benutzeroberfläche ... 212
Methodendefinitionen erstellen und bearbeiten ... 213
Methodendefinitionen deaktivieren ... 214
Skript zum bedingten Anzeigen von Methoden ... 214
Eigenschaften von Methodendefinitionen ... 215
Anwendungen für die Gestaltung der Benutzeroberfläche ... 218
Eigenschaften von Anwendungen ... 219
Symbole und Bilder für die Gestaltung der Benutzeroberfläche ... 221
Verwenden vordefinierter Datenbankabfragen ... 222

Lokalisierung im One Identity Manager ... 224
Arbeiten in verschiedenen Zeitzonen ... 225
Ermitteln von Arbeitszeiten ... 225
Bearbeiten von Länderinformationen ... 226
Länder und Bundesländer aktivieren ... 226
Standardarbeitszeiten für Länder und Bundesländer festlegen ... 227
Feiertage für Länder und Bundesländer anzeigen ... 227
Länder bearbeiten ... 228
Bundesländer bearbeiten ... 229
Eigenschaften von Ländern ... 230
Eigenschaften von Bundesländern ... 231
Eigenschaften von Feiertagen ... 232
Sprachabhängige Abbildung von Informationen ... 233
Grundlagen zur Verwendung sprachabhängiger Informationen ... 234
Spalten zur Übersetzung kennzeichnen ... 235
Verwenden des Textspeichers zur Übersetzung ... 237
Übersetzungen im Wörterbucheditor anzeigen ... 238
Verwendung einer Übersetzung anzeigen ... 239
Übersetzungen einer einzelnen Tabelle bearbeiten ... 240
Alle Übersetzungen bearbeiten ... 241
Schlüsseltext einer Übersetzung ändern ... 242
Übersetzungen aus dem Language Pack importieren ... 243

Prozess-Orchestrierung im One Identity Manager ... 244

Abbildung von Prozessen im One Identity Manager ... 244
Bearbeiten von Prozessen mit dem Prozesseditor ... 246
Definieren von Prozessen ... 248
Prozesse erstellen und bearbeiten ... 249
Prozesse kopieren ... 249
Prozessschritte erstellen und bearbeiten ... 250
Einzelne Prozessschritte kopieren ... 251
Prozessschritte in einen Prozess kopieren ... 252
Eintrag innerhalb eines Prozesses suchen ... 253
Prozesse vergleichen ... 253
Prozesse exportieren und importieren ... 254
Eigenschaften für Prozesse ... 254
Eigenschaften für Prozessschritte ... 256
Parameter für Prozessschritte ... 260
Ereignisse für Prozesse ... 264
Berechtigungen zum Auslösen von Prozessen ... 266
Prozessgenerierung simulieren ... 267
Gültigkeitsprüfung eines Prozesses ... 269
Kompilieren von Prozessen ... 270
Verwenden von prozesslokalen Variablen und globalen Variablen für die Prozessdefinition ... 272
Schwellwerte für die Verarbeitung von Prozessen ... 273
Festlegen des ausführenden Servers ... 274
Server über eine Serverfunktion auswählen ... 275
Server über ein Auswahlskript auswählen ... 275
Benachrichtigung zur Verarbeitung von Prozessschritten ... 276
Automatisierte Ausführung von Prozessen ... 278
Status von Prozessaufträgen anzeigen ... 278
Prozessauftrag sofort starten ... 279
Prozessaufträge erstellen und bearbeiten ... 279
Eigenschaften eines Prozessauftrags ... 280
Übersicht über die Prozesskomponenten ... 281
Ausführungstyp von Prozessfunktionen anzeigen und ändern ... 283
Anzahl der maximalen Instanzen für Prozessfunktionen und Prozesskomponenten ändern ... 284

Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen ... 286
Einrichten von Jobservern ... 289
Jobserver bearbeiten ... 290
Eigenschaften von Jobservern ... 291
Maschinenrollen und Serverfunktionen ... 294
Überblick über die Serverfunktionen ... 295
Überblick über die Maschinenrollen ... 296
Statistikinformationen eines Jobservers ... 297
Verbindungsinformationen für die Prozessgenerierung ... 299
Verbindungsinformationen zum Anwendungsserver erfassen ... 299
Verbindungsinformationen am Jobserver eintragen ... 301
One Identity Manager Service remote auf einem Jobserver installieren ... 301
Jobserver für die Verbindung zum Anwendungsserver konfigurieren ... 304

Arbeitsweise des One Identity Manager Service ... 305
Verarbeitung von Prozessen durch den One Identity Manager Service ... 306
Parallele Verarbeitung von Prozessen durch den One Identity Manager Service ... 307
Ausführung externer Prozesse durch den StdioProcessor ... 308
Konfiguration des One Identity Manager Service ... 309
Konfigurationsdateien des One Identity Manager Service ... 312
One Identity Manager Service Konfiguration für einen Jobserver anpassen ... 313
Vorlage für die Konfigurationsdatei ... 314
Modultypen auswählen und Parameter bearbeiten ... 315
Gültigkeitskontrolle der Konfigurationsdatei ... 316
Modul Prozessabholung ... 317
MSSQLJobProvider ... 317
FileJobProvider ... 318
FTPJobProvider ... 320
HTTPJobProvider ... 323
AppServerJobProvider ... 323
Modul Jobdestination ... 324
JobServiceDestination ... 324
FileJobDestination ... 327
FTPJobDestination ... 329
HTTPJobDestination ... 332

Modul Konfiguration ... 333
Modul Logwriter ... 335
EventLogLogWriter ... 336
FileLogWriter ... 337
Modul Dispatcher ... 338
Modul Connection ... 340
Modul HTTP-Authentifizierung ... 341
Modul Plugins ... 342
HTTPLogPlugin ... 342
ScheduleCommandPlugin ... 342
RequestWatchDogPlugin ... 343
PerformanceCounterPlugin ... 344
DebugMailPlugin ... 344
ShareInfoPlugin ... 345
RemoteConnectPlugin ... 345
DatabaseAgentPlugin ... 345
Modul Datei mit privatem Schlüssel ... 346

Prozessüberwachung zur Nachverfolgung von Änderungen ... 348
Grundlagen zur Prozessüberwachung ... 349
Aufzeichnen von Datenänderungen ... 350
Spalten für die Aufzeichnung von Datenänderungen kennzeichnen ... 351
Aufzeichnen von Prozessinformationen in der Prozessverarbeitung ... 352
Prozessinformationen für Prozesse bearbeiten ... 354
Prozessinformationen für Prozessschritte bearbeiten ... 355
Prozessinformationen für Ereignisse bearbeiten ... 356
Aufzeichnung von Meldungen in der Prozesshistorie ... 357
Prozessverfolgung für Operationen des DBQueue Prozessor ... 358
Beispiel für die Ersetzung der GenProcID ... 359
Archivieren und Löschen von Aufzeichnungen ... 364
Löschen der Aufzeichnungen in der One Identity Manager-Datenbank ohne Archivierung ... 365
Aufbewahrungszeiten für Aufzeichnungen festlegen ... 365
Performance-Optimierung zum Löschen von Aufzeichnungen ... 366

Bedingte Kompilierung mittels Präprozessorbedingungen ... 369
Präprozessorrelevante Konfigurationsparameter ... 370

Präprozessorbedingungen in Objekten ... 371
Präprozessorbedingungen in VB.Net-Ausdrücken ... 372
Auswertung der Präprozessorbedingungen bei der Kompilierung ... 373

Skripte im One Identity Manager ... 375
Verwendung von Visual Basic .NET-Skripten ... 375
Hinweise zur Ausgabe von Meldungen ... 376
Hinweise zur Verwendung von Datumswerten ... 376
Hinweise zur Verwendung von Windows PowerShell Skripten ... 378
Verwendung der $-Notation ... 378
Zugriff auf Spalten des lokalen Objektes ... 379
Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes ... 380
Zugriff auf den alten Wert einer Spalte ... 381
Zugriff auf den Anzeigewert einer Spalte ... 382
Zugriff auf Referenzen in Kommentaren ... 383
Zugriff auf Metawerte des lokalen Objektes ... 383
Zugriff auf die Anzeigewerte eines Objektes ... 384
Verwendung von base Objekt ... 385
Aufruf von Funktionen ... 385
Prä-Skripte zur Verwendung in Prozessen und Prozessschritten ... 385
Verwendung von Session Services ... 386
Abfrage von Konfigurationsparametern ... 386
Testen auf die Existenz eines bestimmten Datenbank-Eintrages ... 387
Abfrage von globalen Variablen des Session-Objektes ... 388
Verwendung der #LD-Notation ... 389
Verwendung der #LD-Notation in der Prozessverfolgung ... 391
Beispiel für die Angabe der Sprache oder der Sprachkultur ... 392
Skriptbibliothek ... 393
Unterstützung bei der Bearbeitung von Skripten im Skripteditor ... 394
Skripte mit dem Skripteditor erstellen und bearbeiten ... 401
Skripte im Skripteditor kopieren ... 402
Skripte im Skripteditor testen ... 403
Kompilieren eines Skriptes im Skripteditor testen ... 404
Überschreiben von Skripten ... 404
Berechtigungen zu Ausführen von Skripten ... 405
Bearbeiten und Testen von Skriptcode mit dem System Debugger ... 406

Systembibliothek laden ... 406
Hinweise zum Bearbeiten von Skriptcode im System Debugger ... 408
Aufzeichnen der Datenbankabfragen und Objektaktionen ... 409
Testen von Skriptcode im System Debugger ... 409
Skripte im System Debugger testen ... 410
Bildungsregeln und Formatierungsskripte im System Debugger testen ... 411
Methoden im System Debugger testen ... 412
Tabellenskripte im System Debugger testen ... 412
Änderungen in die Datenbank speichern ... 413
Erweitertes Debugging im Object Browser ... 413
Lokale Debug-Assemblies erstellen ... 414
Im Object Browser debuggen ... 414
Fehlerbehebung für das Debugging im Object Browser ... 416

One Identity Manager Abfragesprache ... 418
Sprachelemente der One Identity Manager Abfragesprache ... 418
Kommentare ... 419
Bezeichner ... 419
Literale für Werte ... 420
Zeichenkettenwerte ... 420
Ganzzahlige Werte ... 421
Dezimalwerte ... 421
Datumswerte und Zeitwerte ... 421
Referenzen auf Parameter ... 422
Vorformatierte Where-Klauseln ... 422
Formulieren von Abfragen in der One Identity Manager Abfragesprache ... 423
Abfrage-Header ... 424
Where-Klauseln ... 424
Search-Klauseln ... 425
Select-Klauseln ... 425
Order by-Klauseln ... 426
Klauseln für Paging ... 427
Klauseln für Anzeigewerte ... 427
Klauseln für Abfrageparameter ... 428
Abfragehinweise ... 429
Bedingungen ... 430

Besondere Bedingungen ... 430
Vergleichen von Spalten ... 431
Vergleichen mittels IN-Klauseln und NOT IN-Klauseln ... 432
Vergleichen von Datumsdifferenzen ... 433
Vergleichen von Datumsbereichen ... 434
Vergleichen von festen Werten ... 435
Vergleichen von Parametern ... 435
Verwenden vorformatierter Where-Klauseln ... 436

Berichte im One Identity Manager ... 437
Arbeiten mit dem Report Editor ... 437
Menüeinträge im Report Editor ... 438
Ansichten im Report Editor ... 440
Programmeinstellungen des Report Editors ... 440
Aufzeichnen der Datenbankabfragen ... 441
Berichte im Report Editor erstellen und bearbeiten ... 443
Allgemeinen Berichteigenschaften bearbeiten ... 444
Datenquellen erstellen und bearbeiten ... 446
Datenabfrage über eine SQL-Abfrage ... 447
Datenabfrage über eine Datenbanksicht ... 448
Datenabfrage über ein Objekt ... 449
Datenabfrage über die Historie eines Einzelobjektes ... 451
Datenabfrage über die Historie mehrerer Objekte ... 452
Datenabfrage über historische Zuordnungen ... 454
Datenabfrage für Simulationsdaten ... 456
Berichtsparameter ... 458
Berichtparameter bearbeiten ... 459
Allgemeine Parametereinstellungen bearbeiten ... 460
Definition der Parameterwerte bearbeiten ... 461
Einstellungen für Wertberechnung ... 464
Virtuellen Datenquellen verwenden ... 465
Bearbeiten des Berichtsformulars ... 465
Datenfelder in das Berichtsformular einfügen ... 466
Hinweise zur Verwendung von Datumsangaben in Berichten ... 467
Beispiel für einen einfachen Bericht mit Gruppierung der Daten ... 468
Übersetzen von Berichten ... 471

Berichte in die Benutzeroberfläche einbinden ... 472
Berichte zyklische erzeugen und exportieren ... 474

Erweiterung des One Identity Manager Schemas um kundenspezifische Tabellen oder Spalten ... 475
Neue Tabelle erstellen ... 477
Tabelle erweitern ... 478
Definieren von Spalten ... 478
Einfache Spalte erstellen ... 479
Fremdschlüsselspalte erstellen ... 480
Dynamische Fremdschlüsselspalte erstellen ... 482
Neue Spalte für Datenbanksichten mit dem Typ View erstellen ... 483
Erweiterte Konfiguration der Spalten ... 483
Datenbanksichten mit dem Typ Read only erstellen ... 486
Common Table Expressions in Read Only-Datenbanksichten verwenden ... 487
Datenbanksichten mit dem Typ Union erstellen ... 488
Neue Zuordnungstabelle erstellen ... 489
Indexe erstellen ... 490
Kundenspezifische Schemaerweiterungen entfernen ... 491
Mögliche Fehlermeldungen beim Entfernen kundenspezifischer Schemaerweiterungen ... 492
Berechtigungen für die Schemaerweiterungen ... 493
Änderungskennzeichen für die Schemaerweiterungen ... 494
Schemaerweiterungen in die Datenbank übernehmen ... 494
Empfehlungen für die weitere Konfiguration kundenspezifischer Schemaerweiterungen ... 495
Verwaltung von kundenspezifischen Datenbankobjekten innerhalb der Datenbank ... 498

Webservice Integration ... 500
Anbinden eines Webservices ... 501
Generischer Webserviceaufruf ... 502
Direkter Webserviceaufruf ... 503
Selbstdefinierter Webserviceaufruf ... 504
Webservice-Lösung mit dem Webservice-Integrationsassistenten erstellen ... 504
Webservice-Lösung ändern ... 508
Webservice-Lösung löschen ... 509

One Identity Manager als SCIM 2.0 Serviceprovider ... 510

Endpunkte und Basis-URL ... 511
SCIM Plugin Features ... 511
Authentifizierung von SCIM Clients ... 512
Authentifizierung des SCIM Plugins am One Identity Manager ... 512
Besonderheiten bei der Generierung des SCIM Schemas ... 513
Anfragen an das SCIM Plugin ... 514

SOAP Web Service ... 517
Installation und Konfiguration des SOAP Web Service ... 518
SOAP Web Service installieren ... 519
Konfigurieren des SOAP Web Service ... 523
Status eines SOAP Web Service anzeigen ... 525
SOAP Web Service deinstallieren ... 525
Beispiele für Aufrufe ... 526

One Identity Manager als SPML Provisioning Service Provider ... 533
SPML Webservice ... 533
Installation und Konfiguration des SPML Webservice ... 534
SPML Webservice installieren ... 536
Konfigurieren des SPML Webservice ... 539
SPML Webservice deinstallieren ... 541
Konfiguration des One Identity Manager Schemas ... 542
Vorbereiten des One Identity Manager Schemas für den Export in das SPML-Schema ... 542
Schemadateien erstellen ... 543
Testen der Funktionalität des SPML Webservice ... 544
Konfiguration des SPML Testfrontends ... 545
SPML Testfrontend ... 545

Verarbeitung von DBQueue Aufträgen ... 547
Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen ... 548
Benachrichtigungsverhalten für die DBQueue Prozessor Initialisierung konfigurieren 549
Erneutes Initialisieren des DBQueue Prozessor ... 550
Mengenverarbeitung im DBQueue Prozessor ... 550
Verarbeitung der DBQueue Prozessor Aufträge durch den SQL Server Agent ... 551
Verarbeitung der DBQueue Prozessor Aufträge durch den Database Agent Service ... 553
Steuerung der Verarbeitung von DBQueue Prozessor Aufträgen ... 554

Verarbeiten der DBQueue Prozessor Aufträge 555
Kommunikation des zentralen Dispatchers mit den einzelnen Slots 557
Beispiel für die Kommunikation während der Verarbeitung 558

Anhang: Konfigurationsdateien des One Identity Manager Service 560
Jobservice.cfg 560
viNetworkService.exe.config 562

Über uns 564
Kontaktieren Sie uns 564
Technische Supportressourcen 564

Index 565

1

Über dieses Handbuch

Im *One Identity Manager Konfigurationshandbuch* erhalten Sie einen Überblick über die Architektur des One Identity Manager und die Grundlagen zum Arbeiten mit Objekten im One Identity Manager. Es wird der Aufbau des One Identity Manager Schemas erläutert und Sie erfahren, wie Sie das One Identity Manager Schema kundenspezifisch anpassen und erweitern.

Es wird der erklärt, wie Sie die Benutzeroberfläche der Administrationswerkzeuge, insbesondere Manager und Launchpad, kundenspezifisch anpassen können. Sie erfahren, wie Sie die Menüführung erweitern, Formulare anpassen, Berichte erstellen oder kundenspezifische Anzeigetexte lokalisieren.

Es werden die Grundlagen der Prozessorchestrierung im One Identity Manager beschrieben. Sie erfahre, wie Sie Prozesse an Ihre Anforderungen anpassen und eigene Prozesse erstellen. Es wird erläutert, wie Sie die Aufzeichnung von Datenänderungen und Informationen aus der Prozessverarbeitung konfigurieren. Die erweiterten Konfigurationseinstellungen der Jobserver und des One Identity Manager Service werden beschrieben. Zusätzlich erhalten Sie Informationen zur Integration von Webservices, zur Anbindung eines SOAP Web Service sowie zum Datenaustausch über SPML.

Dieses Handbuch wurde als Nachschlagewerk für End-Anwender, Systemadministratoren, Berater, Analysten und andere IT-Fachleute entwickelt.

> HINWEIS: Dieses Handbuch beschreibt die Funktionen des One Identity Manager, die für den Standardbenutzer verfügbar sind. Abhängig von der Systemkonfiguration und den Berechtigungen stehen Ihnen eventuell nicht alle Funktionen zur Verfügung.

Verfügbare Dokumentation

Die One Identity Manager Dokumentation erreichen Sie im Manager und im Designer über das Menü **Hilfe** > **Suchen**. Die Online Version der One Identity Manager Dokumentation finden Sie im Support-Portal unter Online-Dokumentation. Videos mit zusätzlichen Informationen finden Sie unter www.YouTube.com/OneIdentity.

2

Softwarearchitektur des One Identity Manager

Grundlage für den Aufbau des One Identity Manager ist eine klassische 3-Schichten-Architektur. Im One Identity Manager ist jedoch die Objektschicht (die Geschäftslogik) geteilt. Dies ermöglicht einen hohen Performance-Gewinn durch eine räumlich und zeitlich getrennte Verarbeitung.

Datenbankschicht

Die Datenbank stellt den Kern des One Identity Manager dar. Sie erfüllt die Hauptaufgaben der Datenhaltung und der Berechnung von Vererbungen. Vererbt werden können Eigenschaften von Objekten entlang von hierarchischen Strukturen, wie Abteilungen, Kostenstellen, Standorten oder Geschäftsrollen. Bei der Datenhaltung bildet die Datenbank die zu verwaltenden Zielsysteme, die ERP-Strukturen sowie Regeln zur Compliance und Zugriffsberechtigungen ab.

Logisch ist die Datenbank in die zwei Bereiche der Nutzdaten und der Metadaten geteilt. Die Nutzdaten enthalten alle für die Datenpflege nötigen Informationen wie beispielsweise Informationen über Personen, Benutzerkonten, Gruppen, Mitgliedschaften und Betriebsdaten, Genehmigungsworkflows, Attestierungen, Rezertifizierungen und Complianceregeln.

Die Metadaten enthalten die Beschreibung des Anwendungsdatenmodell sowie Skripte für Format- und Bildungsvorschriften oder bedingte Wechselwirkungen. Die komplette Systemkonfiguration des One Identity Managers, die gesamten Einstellungen zur Steuerung der Frontends und die Queues für asynchrone Verarbeitung der Daten und Prozesse sind ebenfalls Teil der Metadaten.

Die Neuberechnung von Vererbungen wird durch die Triggerlogik der Datenbank ausgelöst. Die Trigger stellen dazu Verarbeitungsaufträge in eine als DBQueue bezeichnete Auftragsliste ein. Der DBQueue Prozessor verarbeitet diese Aufträge und berechnet die Vererbungen der jeweiligen Datenbankobjekte neu. Eine als Jobqueue bezeichnete Tabelle dient der Ablage von Verarbeitungsaufträgen, die von der Objektschicht auszuführen sind.

Als Datenbanksystem kommt SQL Server oder eine verwaltete Instanz in Azure SQL-Datenbank zum Einsatz.

Objektschicht

Die Objektschicht ermöglicht den objektorientierten Zugriff auf die Daten der Datenbank. Die `VI.DB.DLL` erzeugt Entitäten für Objekte und Collections. Entitäten nutzen externe Services der Session zum Laden (EntitySource) und Speichern (UnitOfWork) von Datenobjekten. Speicheroperationen werden gruppiert, so dass mehrere Datenobjekte im Bulkverfahren gespeichert werden. Für jedes Objekt stehen die Standardereignisse Insert, Update und Delete zur Verfügung, die nach dem Speichern generiert werden können.

Jeder Entität sind eine oder mehrere Verarbeitungslogiken zugeordnet (EntityLogic). Diese fassen Operationen zusammen, die für eine Entität ausgeführt werden können. Für die unterschiedlichen Entitäten wurden eigene Customizer entwickelt. Ein Customizer ist eine EntityLogic, die ein definiertes Verhalten für eine Entität bereitstellt. Customizer führen Verarbeitungslogiken aus, die üblicherweise im Objektcode selbst implementiert würden, wie beispielsweise den gegenseitigen Ausschluss von Eigenschaften.

Jeder Eigenschaft des generierten Objekts kann eine Bildungsregel zugeordnet werden. Bildungsregeln werden zur Generierung von Nutzdaten oder zur Transformation von Werten eingesetzt. Über Bildungsregeln ist es möglich Eigenschaften von Objekten mit Standardwerten zu belegen oder den Wert einer Eigenschaft aus anderen Eigenschaften desselben oder anderer Objekte zu bilden.

Der One Identity Manager verwendet zur Abbildung von Geschäftsprozessen sogenannte Prozesse. Ein Prozess besteht aus Prozessschritten, die Verarbeitungsaufgaben darstellen und über Vorgänger-Nachfolger-Beziehungen miteinander verbunden sind. Dieses Funktionsprinzip erlaubt es, flexibel Aktionen und Abläufe an die Ereignisse von Objekten zu koppeln. Die Modellierung der Prozesse erfolgt über Prozessvorlagen. Die Umwandlung der als Skript definierten Vorlagen in Prozessen und Prozessschritten in einen konkreten Prozess in der Jobqueue übernimmt der Jobgenerator.

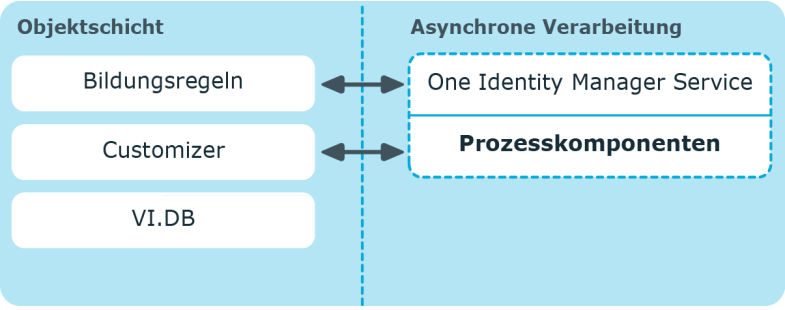
Der One Identity Manager Service sorgt für die Verbreitung der in der One Identity Manager-Datenbank verwalteten Informationen im Netzwerk. Der One Identity Manager Service übernimmt die Datensynchronisation zwischen Datenbank und den angebundenen Zielsystemen sowie die Durchführung von Aktionen auf Datenbank- und Dateiebene.

Der One Identity Manager Service holt die Prozessschritte aus der Jobqueue ab. Die Prozessschritte werden von Prozesskomponenten ausgeführt. Der One Identity Manager Service erzeugt dazu eine Instanz der benötigten Prozesskomponente und übergibt die Parameter des Prozessschrittes. Eine Entscheidungslogik überwacht die Ausführung der Prozessschritte und veranlasst abhängig vom gemeldeten Ausführungsergebnis die weitere Verarbeitung des Prozesses. Der One Identity Manager Service ermöglicht die parallele Verarbeitung von Prozessschritten, da er mehrere Instanzen von Prozesskomponenten erzeugen kann.

Der One Identity Manager Service ist die einzige Komponente des One Identity Manager, die berechtigt ist, Änderungen in den Zielsystemen auszuführen.

Streng betrachtet ist der One Identity Manager Service kein Bestandteil der Objektschicht, da er keine Geschäftslogik enthält. Der One Identity Manager Service stellt ein Hilfsmittel zur Realisierung der asynchronen Verarbeitung dar.

Abbildung 1: Objektschicht des One Identity Manager



Präsentationsschicht

Die Präsentationsschicht besteht aus Frontends die zur Eingabe und Ausgaben von Daten dienen. Für unterschiedliche Aufgabenstellungen gibt es verschiedene Frontends. Beispielsweise wird zur Konfiguration des One Identity Manager ein anderes Frontend verwendet als zur Verwaltung von Personendaten. Die darzustellenden Inhalte und ihre Änderbarkeit werden in Abhängigkeit der Zugriffsberechtigungen des jeweiligen Benutzers durch die Objektschicht bestimmt. Als Frontends stehen sowohl Clients als auch eine browserbasierte Lösung zur Verfügung.

Die Clients verbinden sich zu einem Anwendungsserver, der die Geschäftslogik hält. Der Anwendungsserver stellt einen Verbindungspool für den Zugriff auf die Datenbank zur Verfügung und sorgt für eine sichere Verbindung zur Datenbank. Die Clients senden ihre Anfragen an den Anwendungsserver, dieser führt die Verarbeitung der Objekte wie beispielsweise die Bildung von Werten nach definierten Bildungsregeln aus und sendet die Ergebnisse an die Clients zurück. Mit dem Speichern eines Objektes werden die Daten vom Anwendungsserver an die Datenbank übergeben.

Die Clients können alternativ ohne externen Anwendungsserver arbeiten und selbst die Objektschicht halten und direkt auf die Datenbankschicht zugreifen. In den Clients kommt in diesem Fall nur der Teil der Objektschicht zum Einsatz, der die Erfassungsprozesse abbildet.

Für den Einsatz browserbasierter Frontends wird auf einem Webserver eine Applikation betrieben, die aus einer Renderengine für Webseiten besteht. Der Benutzer greift mittels eines Webbrowsers auf die für ihn dynamisch erstellte und angepasste Website zu. Der Austausch zwischen Datenbank und Webserver kann über den Anwendungsserver oder direkt erfolgen.

Abbildung 2: Verteilung der Schichten mit Anwendungsserver

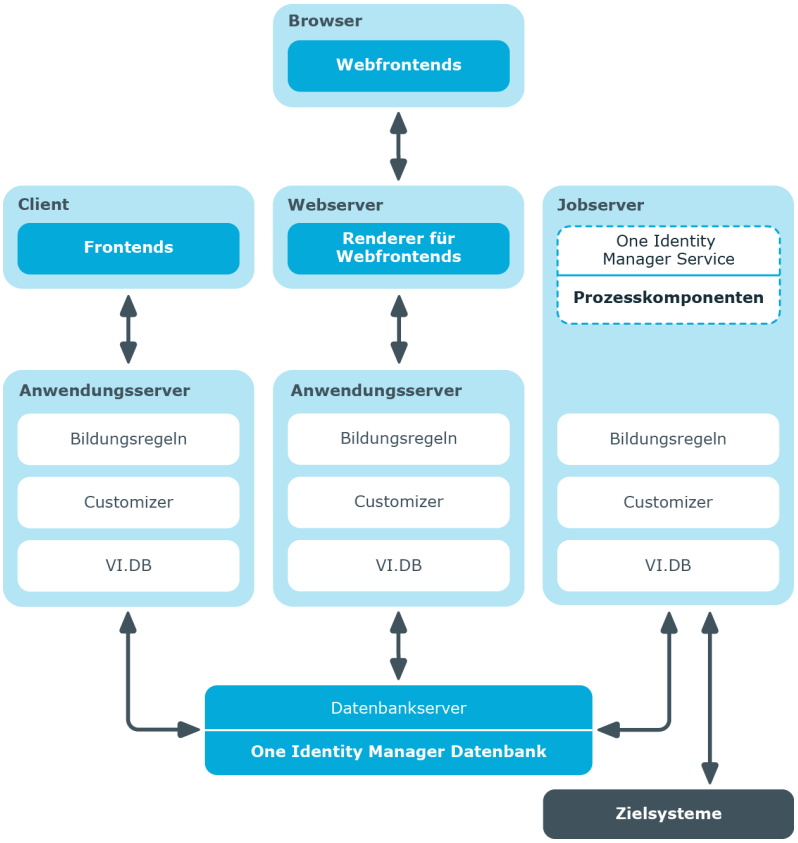
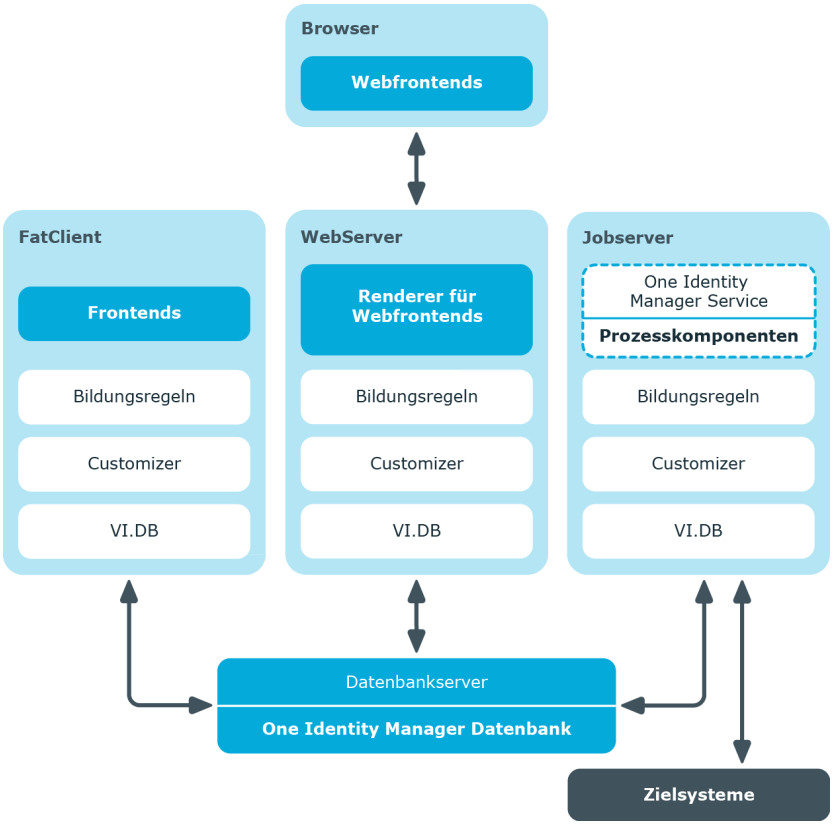


Abbildung 3: Verteilung der Schichten ohne Anwendungsserver



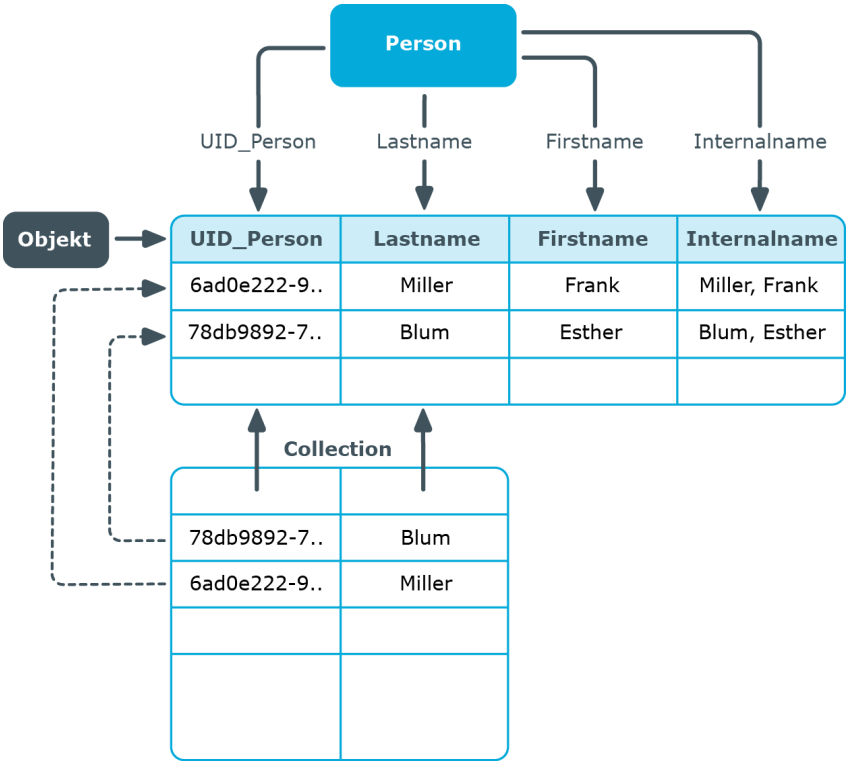
Verwandte Themen

- Arbeiten mit Objekten im One Identity Manager auf Seite 22
- Einfügen, Ändern und Löschen eines Objektes im One Identity Manager auf Seite 26

Arbeiten mit Objekten im One Identity Manager

Durch die Objektschicht des One Identity Manager erfolgt der objektorientierte Zugriff auf die Tabellen und Datensätze.

Abbildung 4: Zugriff auf Tabellen und Datensätze



Dabei gelten folgende Abbildungen:

- Objektklasse - Tabelle
- Eigenschaften - Spalten
- Objekt - Zeile
- Collection - Anzahl (1-n) Spalten einer Tabelle über mehrere Zeilen

Objekte und Collections werden über Entitäten abgebildet. Entitäten sind die Dateneinheiten, die aus der Datenbank abgerufen und in der Datenbank gespeichert werden können. Eine Entität entspricht einer Zeile einer Tabelle in der Datenbank und enthält Datenspalten und einige Metawerte wie Anzeigewerte und Berechtigungen.

Entitäten können nur einige Spalten oder alle Spalten einer Tabelle enthalten. Im ersten Fall sind sie durch die `IsPartial`-Eigenschaft gekennzeichnet und können nicht geändert werden.

Es gibt drei Arten von Entitäten:

- Nur Lesen (Read only)

  Datenwerte können nur gelesen werden. Die Entitäten können nicht gespeichert werden.

- Verzögerte Logik (Delayed logic)

  Die Entitäten können geändert und gespeichert werden. Der verzögerte Logikmodus führt alle Regeln und Methoden der Geschäftslogik beim Speichern der Entität aus.

Wenn die Entität gegen einen Anwendungsserver läuft, lebt die Entität auf der Client-Seite und verwendet keine Server-Ressourcen.

- Interaktiv (Interactive)

  Die Entitäten können geändert und gespeichert werden. Die zugrundeliegende Logik wird direkt nach der Änderung eines Wertes angewendet. Ihre primäre Anwendung sind Benutzeroberflächen, bei denen die Anwender das Ergebnis der Geschäftslogik direkt sehen wollen. Um die Logik ohne Einschränkung mit den Berechtigungen des Benutzers ausführen zu können, muss die Entität auf dem Anwendungsserver leben, wenn sie nicht direkt gegen eine Datenbank läuft.

Die Entitäten haben die folgenden Standardmethoden zum Ausführen der Datenbankoperationen.

Tabelle 1: Standardmethoden

Methode	Beschreibung
EntitySource	Erzeugen neuer Objekte und Collections oder Laden von Objekten und Collections.
UnitOfWork	Gruppieren von Speicheroperationen mehrerer Objekte und Collections.
Discard	Verwerfen von Objekten.
MarkForDeletion	Markieren der Objekte werden zum Löschen. Gelöscht wird erst beim Speichern.

Beim Laden eines Objektes werden alle Spalten geladen. Beim Laden einer Collection werden aus Performancegründen nicht alle Spalten geladen. Es werden die Primärschlüsselspalten, alle Spalten, die im Anzeigemuster enthalten sind, und Spalten mit der Angabe, ob ein Objekt zum Löschen markiert ist, geladen. Definierte Anzeigemuster geben an, wie die einzelnen Objekte einer Collection im Frontend angezeigt werden. Die Standards für die Anzeigemuster jeder Tabelle sind im One Identity Manager Schema hinterlegt und können angepasst werden.

Objekte kennen die folgenden Standardereignisse, die als Folge des Speicherns generiert werden können.

Tabelle 2: Standardereignisse der Objekte

Ereignis	Beschreibung
Insert	Einfügen eines Objektes.
Update	Ändern eines Objektes.
Delete	Löschen eines Objektes.
Assign	Hinzufügen von M:N Zuweisungen.
Remove	Entfernen von M:N Zuweisungen.

An diese Ereignisse können Prozesse gekoppelt werden, welche Aktionen in den verschiedenen Zielsystemen ausführen, wie beispielsweise Benutzerkonten anlegen, ein Homeverzeichnis auf einem Server anlegen oder Werte in die One Identity Manager-Datenbank schreiben.

Tabelle 3: Lebenszyklus eines Objektes

Aktion im Frontend	Zustand des Objektes	Ereignis beim Speichern	Aktion in der Datenbank
Einfügen eines Objektes.	Objekt existiert nicht.	Insert	UID wird erzeugt, Objekt wird in die Datenbank eingefügt.
Ändern von Eigenschaften.	Objekt existiert in der Datenbank und ist geladen.	Update	Eigenschaften des Objektes werden geändert.
Löschen des Objektes.	Objekt existiert in der Datenbank und ist geladen.	Delete	Für Objekte, die die Eigenschaft **Zum Löschen markiert** (`XMarkedForDeletion`) besitzen: • Es wird die Methode **MarkForDeletion** ausgeführt. Die Objekte werden zur weiteren Bearbeitung gesperrt. • Ist eine Löschverzögerung **> 0** Tage konfiguriert, wird eine verzögerte Operation für das Löschen erzeugt. Die Objekte werden zunächst deaktiviert. Während dieser Zeit besteht die Möglichkeit, die Objekte wiederherzustellen. Beim Wiederherstellen eines gelöschten Objektes werden die Eigenschaften des Objektes wieder in den Zustand vor dem Löschen zurückgesetzt. Nach Ablauf der Löschverzögerung werden die Objekte endgültig gelöscht. • Objekte mit einer Löschverzögerung von **0** Tagen, werden sofort gelöscht. Objekte, die keine Eigenschaft **Zum Löschen markiert** besitzen, werden sofort gelöscht.

Verwandte Themen

- Einfügen, Ändern und Löschen eines Objektes im One Identity Manager auf Seite 26

Einfügen, Ändern und Löschen eines Objektes im One Identity Manager

Alle Aktionen im One Identity Manager werden über die Objektschicht ausgeführt und in der One Identity Manager-Datenbank gespeichert. Jede Veränderung eines Objektes (Einfügen, Ändern und Löschen) wird innerhalb einer Transaktionsklammer ausgeführt. Ebenfalls fester Bestandteil einer solchen Transaktion ist das Erstellen der Verarbeitungsaufträge selbst. Nur wenn sowohl das Speichern der Änderung als auch das Erstellen der Verarbeitungsaufträge erfolgreich waren, wird die Transaktion erfolgreich abgeschlossen. Sollte innerhalb der Transaktion ein Fehler auftreten, erfolgt ein Rollback der gesamten Transaktion.

Nachfolgend wird der Ablauf beispielhaft für das Einfügen eines Objektes im One Identity Manager erläutert.

Folgende Aktionen werden im Frontend ausgeführt:

- Einfügen eines neuen Objektes
- Erfassen der Eigenschaften des Objektes

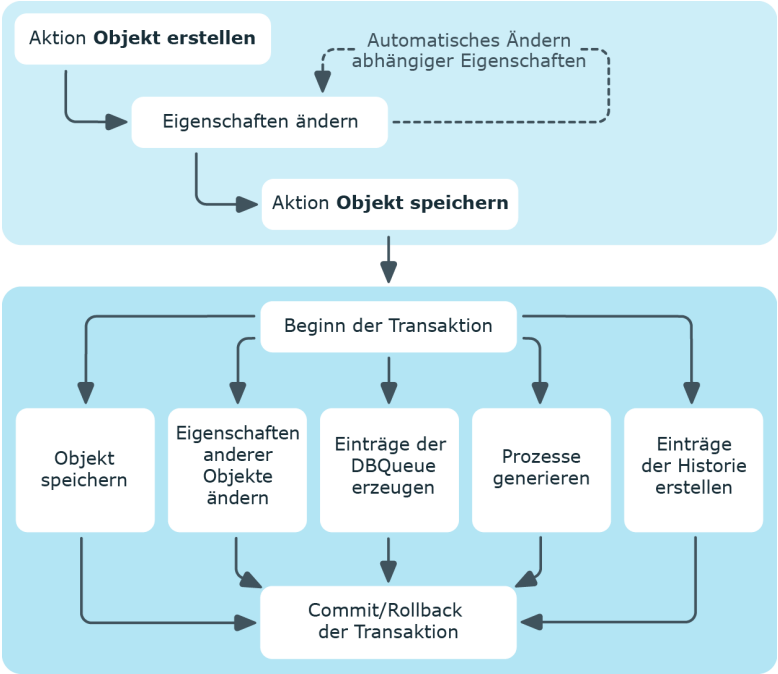
  Abhängige Eigenschaften innerhalb des Objektes werden durch Bildungsregeln erzeugt. Im Customizer implementierte Nebenwirkungen, wie beispielsweise das gegenseitige Ausschließen bestimmter Eigenschaften, werden angewendet.
- Speichern des Objektes

Nach dem Speichern des Objektes im Frontend, werden durch die Objektschicht die folgenden Schritte ausgeführt:

- Starten einer Transaktion (Begin Transaction)
- Parallele Verarbeitung der folgenden Schritte:
  - Speichern des Objektes in der Datenbank
  - Anwenden der Bildungsregeln und Formatierungsskripte für abhängige Objekte
  - Erzeugen der Verarbeitungsaufträge für den One Identity Manager Service in der Jobqueue
  - Erzeugen der Verarbeitungsaufträge für den DBQueue Prozessor in der DBQueue
  - Erzeugen der Einträge zur Aufzeichnung von Änderungen in einer Historie
- Beenden der Transaktion im Erfolgsfall (Commit Transaction) oder Zurückrollen der Änderungen im Fehlerfall (Rollback Transaction)

Nachfolgende Abbildung verdeutlicht den Datenfluss beim Einfügen eines Objektes.

Abbildung 5: Datenfluss für das Einfügen eines Objektes



Kundenspezifische Änderungen an der One Identity Manager Standardkonfiguration

Große Teile der One Identity Manager Standardkonfiguration können Sie kundenspezifisch anpassen. So können Sie beispielsweise eigene Anzeigenamen für Spalten oder Menüeinträge festlegen oder eigene Bildungsregeln und Formatierungsregeln von Spaltenwerten definieren.

Wird eine Standardkonfiguration kundenspezifisch geändert, dann wird die Änderung über Trigger abgefangen und die Standardkonfiguration in einen Konfigurationsspeicher verschoben. Sie können Änderungen aus dem Konfigurationsspeicher zurückholen und somit die Standardkonfiguration wiederherstellen.

- Änderungen an Daten werden durch das Symbol vor dem geänderten Wert gekennzeichnet. Solange die Änderungen noch nicht gespeichert sind, können Sie diese mit einem Mausklick auf das Symbol rückgängig machen.
- Änderungen zur Standardkonfiguration werden im Designer mit dem Symbol gekennzeichnet. Um die Standardkonfiguration wieder herzustellen, klicken Sie auf das Symbol.

Sollte während einer Versionsänderung des One Identity Manager durch ein Service Pack oder eine vollständige Versionsänderung oder beim Einspielen eines Hotfixpaketes eine Änderung an einer Standardkonfiguration vorgenommen werden, so zunächst geprüft, ob es bereits eine kundenspezifische Anpassung gibt. Ist dies der Fall, dann wird die geänderte Standardkonfiguration in den Konfigurationsspeicher installiert. Somit ist sichergestellt, dass kundenspezifische Änderungen erhalten bleiben.

Verwandte Themen

- Dynamisches Nachladen von Änderungen auf Seite 29
- Einzelne Eigenschaften für die Bearbeitung sperren und entsperren auf Seite 30
- Berichte zur Systemkonfiguration auf Seite 31

Dynamisches Nachladen von Änderungen

Gecachte Systemdaten können, wenn sich diese geändert haben, dynamisch nachgeladen werden. Das Nachladen der Änderungen erfolgt automatisch im Hintergrund.

Eine Ausnahme bilden die Änderungen, welche die Gestaltung der Benutzeroberfläche betreffen. Diese Anpassungen werden nur nach Rückfrage beim Benutzer nachgeladen. Der Benutzer kann entscheiden, wann er diese Änderungen übernimmt. In der Statuszeile des Managers wird über das Symbol 🔄 angezeigt, dass sich die Benutzeroberfläche geändert hat.

Bei Änderungen werden die Semaphore inkrementiert. Die Berechnung der Semaphore wird durch den DBQueue Prozessor ausgeführt.

Um das Nachladen von Änderungen zu konfigurieren

1. Prüfen Sie im Designer, ob der Konfigurationsparameter **Common | CacheReload** aktiviert ist. Anderenfalls aktivieren Sie den Konfigurationsparameter und kompilieren Sie die Datenbank.
2. Legen Sie über den Konfigurationsparameter **Common | CacheReload | Type** das Verfahren fest, nach welchem die Gültigkeit der gecachten Informationen überprüft werden soll. Zulässige Werte sind:
   - **ALWAYS**: Die Gültigkeit der gecachten Informationen wird bei jedem Zugriff geprüft.
   - **NEVER**: Die Gültigkeit der gecachten Informationen wird nie geprüft.
   - **TIMER**: Die Gültigkeit der gecachten Informationen wird nach Ablauf des Intervalls geprüft.
3. Wenn Sie das Verfahren **TIMER** verwenden, legen Sie im Konfigurationsparameter **Common | CacheReload | Interval** die Zeit in Sekunden an, nach der bei einem Zugriff die Werte zu prüfen sind.

Im Datenmodell ist definiert, welche Spalten ein Nachladen erfordern. Einen Überblick über die Semaphore erhalten Sie im Designer in der Kategorie **Basisdaten > Erweitert > Semaphore**.

- Um bei Änderungen einer Spalte die Daten nachzuladen, muss die Spalte dem Semaphor zugewiesen sein.
- Um beim Einfügen oder Löschen in eine Tabelle die Daten nachzuladen, muss die Primärschlüsselspalte dem Semaphor zugewiesen sein.

Tabelle 4: Nachzuladene Änderungen

Änderungen	Semaphor
Skriptassembly und Customizer	Assembly

Änderungen	Semaphor
Berechnung der Spaltenabhängigkeiten	BulkdDependencies
Bezeichnungen, zum Beispiel Spaltenbezeichnungen oder Anzeigetexte	Caption
Konfigurationsparameter	Config
Länder und Zeitzonen	Country
Bestandteile der Benutzeroberfläche	Dialog
Nutzung spezieller Programmfunktionen	Feature
Symbole	Image
Tabellen, Spalten, Tabellenbeziehungen, Spaltenbeziehungen, Objekte, Methoden	Model
Notification	Notification
Berechtigungen und Berechtigungsgruppen	Right
Revisionsstand der Software (für Softwareaktualisierung)	SoftwareRevision
Statistikdefinitionen	DashBoardDef
Statistikinhalte	DashBoardContent
Modulabhängigkeiten	ModuleDepend
Im Speicher vorgehaltene Benutzerdaten	UserDataResident
Änderungen der Synchronisationskonfiguration	DPRConfiguration
Änderungen an Modulabhängigkeiten	ModuleDepend
Änderungen an der Web Portal Konfiguration	AEDS
Änderungen an vordefinierten SQL Abfragen	LimitedSQL
Änderungen an Berechtigungen für Web API Methoden	AEDSGROUP
Änderungen an Kennwortrichtlinien	PasswordPolicy

Einzelne Eigenschaften für die Bearbeitung sperren und entsperren

Einzelne Eigenschaften können gegen Überschreiben durch einen Transport als auch normale Bearbeitung gesperrt werden.

Beispiele, für die eine Bearbeitungssperre erforderlich sein kann:

- Konfigurationsparameter mit ihren Werten sollen während eines Transportes von einer Testumgebung in der Produktivumgebung nicht überschrieben werden.
- Konfigurationen einzelner Server, die sowohl in der Testumgebung als auch in der Produktivumgebung vorhanden sind, sollen bei einem Transport nicht überschrieben werden.

HINWEIS: Um Eigenschaften für die Bearbeitung zu sperren, benötigen die Benutzer die Programmfunktion **Ermöglicht das Setzen einer Änderungssperre für bestimmte Eigenschaften einzelner Objekte** (`Common_AllowPropertyLocks`).

Soll es bestimmten Benutzern möglich sein, Eigenschaften für die Bearbeitung zu sperren, können Sie die Berechtigungen über Berechtigungsgruppen an die Benutzer vergeben.

- Für die nicht-rollenbasierte Anmeldung wird die Berechtigungsgruppe **QBM_ PropertyLock** bereitgestellt. Diese Gruppe besitzt die Programmfunktion. Nehmen Sie die Systembenutzer in die Berechtigungsgruppe auf. Administrative Systembenutzer erhalten diese Berechtigungsgruppe automatisch.
- Für die rollenbasierte Anmeldung wird die Berechtigungsgruppe **QER_4_ PropertyLock** bereitgestellt. Diese Gruppe besitzt die Programmfunktion. Die Berechtigungsgruppe ist mit der Anwendungsrolle **Basisrollen | Sperren einzelner Eigenschaften** verbunden. Nehmen Sie die Personen in die Anwendungsrolle auf.

Um eine einzelne Eigenschaft zu sperren und zu entsperren

1. Öffnen Sie im Designer oder Manager das Objekt.
2. Klicken Sie auf die Bezeichnung der Eigenschaft und wählen Sie über das Kontextmenü einen der folgenden Einträge:
   - **Bearbeitung sperren**: Die Eigenschaft wird für die Bearbeitung gesperrt. Das Eingabefeld wird gesperrt und grau angezeigt.
   - **Bearbeitung zulassen:** Die Eigenschaft wird für die Bearbeitung freigeschaltet.

Berichte zur Systemkonfiguration

Im Designer werden verschiedene Berichte zur Systemkonfiguration und zu den Kundenanpassungen bereitgestellt. Bei Auswahl eines Eintrages in dieser Kategorie wird der entsprechende Bericht generiert. Je nach Umfang des Berichtes kann die Generierung einige Zeit in Anspruch nehmen.

Um Berichte zur Systemkonfiguration anzuzeigen

- Wählen Sie im Designer die Kategorie **Dokumentation**.

Tabelle 5: Verfügbare Berichte zur Systemkonfiguration

Bericht	Inhalt
Systemkonfiguration	Der Bericht enthält die Beschreibungen und Einstellungen der aktivierten Konfigurationsparameter.
Prozesse	Der Bericht enthält die Beschreibungen aller aktiven Standardprozesse. Zu einem Prozess werden die Prozessschritte und deren Parameter sowie die verwendeten Skripte und Konfigurationsparameter aufgeführt.
Prozesskomponenten	Der Bericht enthält die Beschreibungen aller Prozesskomponenten mit ihren Funktionen und Parametern.
Bildungsregeln	Der Bericht enthält die Beschreibungen aller Standardbildungsregeln inklusive der betroffenen Spalten und der verwendeten Skripte und Konfigurationsparameter.
Formatvorschriften	Der Bericht enthält die Beschreibungen aller Standardformatvorschriften inklusive der verwendeten Skripte und Konfigurationsparameter.
Skripte	Der Bericht enthält die Beschreibungen aller Standardskripte inklusive der verwendeten Konfigurationsparameter. Zu jedem Skript wird die Verwendung in Prozessen, Prozessschritten, Bildungsregeln, Formatvorschriften und Skripten angezeigt.
TimeTrace	Der Bericht zeigt die Konfiguration des TimeTrace.
Gesamtbericht	Gesamtbericht über die Systemkonfiguration. Der Bericht fasst die Informationen der Teilberichte zusammen.

Tabelle 6: Verfügbare Berichte zu Kundenanpassungen

Bericht	Inhalt
Systemkonfiguration	Der Bericht enthält die Beschreibungen und Einstellungen der aktivierten Konfigurationsparameter.
Prozesse	Der Bericht enthält die Beschreibungen aller aktiven kundenspezifischen Prozesse. Zu einem Prozess werden die Prozessschritte und deren Parameter sowie die verwendeten Skripte und Konfigurationsparameter aufgeführt.
Bildungsvorschriften	Der Bericht enthält die Beschreibungen aller kundenspezifischen Bildungsregeln inklusive der betroffenen Spalten und der verwendeten Skripte und Konfigurationsparameter.
Formaktvorschriften	Der Bericht enthält die Beschreibungen aller kundenspezifischen Formatvorschriften inklusive der verwendeten Skripte und Konfigurationsparameter.
Skripte	Der Bericht enthält die Beschreibungen aller kundenspezifischen

Bericht	Inhalt
	Skripte inklusive der verwendeten Konfigurationsparameter. Zu jedem Skript wird die Verwendung in Prozessen, Prozess-schritten, Bildungsregeln, Formatvorschriften und Skripten angezeigt
One Identity Manager Schema	Der Bericht enthält die Beschreibungen kundenspezifischen Erweiterungen des One Identity Manager Schemas (Tabellen und Spalten). Zusätzlich werden die Informationen zu kunden-spezifischen Datenbankobjekten, wie Datenbankprozeduren, Funktionen, Trigger oder Sichtdefinitionen aufgeführt.
Gesamtbericht	Gesamtbericht über die Systemkonfiguration. Der Bericht fasst die Informationen der Teilberichte zusammen.

Anpassen der One Identity Manager Basiskonfiguration

Die Basisdaten umfassen Grundeinstellungen zur Konfiguration des One Identity Manager, die in der Regel einmalig vor Inbetriebnahme des Systems überprüft und angepasst werden. Dies beinhaltet beispielsweise die Datenbankverbindungsdaten, die Nutzung der Authentifizierungsmodule, die verwendeten Sprachen oder die Einstellung der Konfigurationsparameter.

Verwandte Themen

- Übersicht über die Einstellungen einer Datenbank auf Seite 35
- Verbindungsdaten zur Datenbank ändern auf Seite 38
- Konfiguration der Datenbank für eine Test-, Entwicklungs- oder Produktivumgebung auf Seite 39
- Staging-Ebene der Datenbank ändern auf Seite 40
- Spracheinstellungen für die Anzeige und Pflege der Daten auf Seite 40
- Anmeldesprachen aktivieren auf Seite 41
- Standardland für Arbeitszeitermittlung und Feiertagen festlegen auf Seite 41
- Konfigurationsparameter zur Systemkonfiguration auf Seite 42
- Konfigurationsparameter bearbeiten auf Seite 42
- Kundenspezifische Konfigurationsparameter erstellen auf Seite 43

- Ausführliche Informationen den Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Übersicht über die Einstellungen einer Datenbank

HINWEIS: Änderungen an diesen Daten sind in der Regel nicht erforderlich und sollten nur von erfahrenen Benutzern vorgenommen werden.

Um die Datenbankinformationen anzuzeigen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Datenbanken**.
2. Wählen Sie im Listeneditor die Datenbank.
3. Folgende Informationen werden angezeigt.

Tabelle 7: Datenbankinformationen

Eigenschaft	Bedeutung
Hauptdatenbank	Kennzeichnung der Hauptdatenbank. Bei der initialen Schemainstallation wird die One Identity Manager-Datenbank mit dieser Option gekennzeichnet.
Kunde	Name des Kunden.
Beschreibung	Beschreibung der Datenbank.
Kundenpräfix	Kennzeichen des Kunden. Das Kundenpräfix wird für die Erstellung und den Transport kundenspezifischer Skripte, Prozesse und Erweiterungen am One Identity Manager Schema verwendet.
Modul Eigentümer	Kennzeichen des Moduleigentümers. Das Präfix wird für die Erstellung und den Transport kundenspezifischer Skripte, Prozesse und Erweiterungen am One Identity Manager Schema verwendet.
Staging Ebene	Gibt an, ob die Datenbank eine Testdatenbank, eine Entwicklungsdatenbank oder eine produktive Datenbank ist. Die zulässigen Werte sind **Entwicklungssystem**, **Testumgebung** und **Produktivsystem**.
Kundenspezifische Staging Ebene	Detaillierte Informationen zur Staging Ebene. Die Informationen werden in der Statuszeile der Programme im Tooltip zur Datenbankverbindung sowie in der Installationsübersicht im Launchpad angezeigt.
Farbe Statuszeile	Abhängig von der Staging-Ebene kann die Statuszeile in einer vom Layout abweichenden Farbe dargestellt werden. Die Farbdefinition ist per Bildungsregel angegeben und kann kundenspezifisch angepasst werden. Im Standard sind folgende Farben definiert:

Eigenschaft	Bedeutung
	• keine - Die Datenbank des Entwicklungssystems ist verbunden. • Grün - Die Datenbank der Testumgebung ist verbunden. • Gelb - Die Datenbank des Produktivsystems ist verbunden.
Letztes Kompiler-relevantes Konfigurationsdatum	Zeitpunkt der letzten Kompiler-relevanten Änderung. Bei Änderung des Wertes ist eine erneute Kompilierung der Datenbank ist erforderlich.
Start der Simulation	Zeitpunkt, an dem die letzte Simulation in einem Frontend gestartet wurde.
Stopp DBQueue Prozessor	Wenn diese Option für die Hauptdatenbank gesetzt ist, wird der DBQueue Prozessor keine Aufträge mehr verarbeiten. Den DBQueue Prozessor können Sie mit entsprechenden administrativen Berechtigungen im Job Queue Info anhalten und starten. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.
Stopp One Identity Manager Service	Wenn diese Option für die Hauptdatenbank gesetzt ist, wird der One Identity Manager Service keine Aufträge mehr verarbeiten. Den Dienst können Sie mit entsprechenden administrativen Berechtigungen im Job Queue Info anhalten und starten. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.
Provider	Für die Verbindung zum SQL Server ist **VI.DB.ViSqlFactory,VI.DB** eingetragen.
Verbindungsparameter	Anmeldedaten des Datenbanknutzer, der Datenbankserver und die Datenbank. Die Daten werden bei der Schemainstallation in die Datenbank eingetragen.
Authentifizierungsmodul	Standardauthentifizierungsmodul zur Anmeldung an der Datenbank. Ausführliche Informationen zu den One Identity Manager Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.
Sprache (Standard)	Standardsprache. Fallback-Variante für die Anzeige sprachabhängiger Texte.

Eigenschaft	Bedeutung
Land (Standard)	Standardland. Das Land wird bei der Ermittlung von Arbeitszeiten und Feiertagen berücksichtigt.
Edition	Bezeichnung der installierten Edition.
Version der Edition	Version der Edition.
Beschreibung der Edition	Nähere Beschreibung der Edition.
Datenbank-ID	ID der Datenbank. Die Datenbank-ID wird aus den Originaldaten von Datenbankserver und Datenbank. Die Neuberechnung der Datenbank-ID ist beim Erstellen einer Datenbank aus einem Datenbankbackup von einem anderen Server erforderlich. Bei Kompilierung einer Datenbank wird die Datenbank-ID geprüft und gegebenenfalls geändert
Einzelbenutzermodus Prozess	Prozess, der Einzelbenutzermodus erfordert. Ist der Wert **0**, ist kein Einzelbenutzermodus erforderlich. HINWEIS: Der DBQueue Prozessor prüft in regelmäßigen Abständen, ob der Einzelbenutzermodus noch erforderlich ist und setzt gegebenenfalls die Angabe zurück.
Einzelbenutzermodus Startzeit	Zeitpunkt, zu dem der Einzelbenutzermodus gestartet wurde. HINWEIS: Der DBQueue Prozessor prüft in regelmäßigen Abständen, ob der Einzelbenutzermodus noch erforderlich ist und setzt gegebenenfalls die Angabe zurück.
Öffentlicher Schlüssel zur Verschlüsselung	Der öffentlicher Schlüssel wird durch das Programm Crypto Configuration eingetragen und wird bei der Verschlüsselung von Datenbankinformationen benötigt. Ausführliche Informationen zur Verschlüsselung der Datenbank finden Sie im *One Identity Manager Installationshandbuch*.
Vorbereitungsphase für Migration	Anzeige zur stufenweise Vorbereitung einer Migration. Mögliche Phasen sind: • grün: Die Datenbank befindet sich im normalen Betriebsmodus. • gelb: Alle Datenbanknutzer werden über die bevorstehende Aktualisierung informiert. Das System nimmt keine Prozesse mehr an. Die Vorbereitungsphase wird in der Statuszeile der Programme angezeigt. • orange: Neue Benutzer können sich an der Datenbank nicht mehr anmelden. Alle offenen Prozesse werden noch verarbeitet. Die Vorbereitungsphase wird in der Statuszeile der Programme angezeigt.

Eigenschaft	Bedeutung
	• rot: Die Datenbank ist bereit für die Aktualisierung. Die Vorbereitungsphase wird in der Statuszeile der Programme angezeigt. • rot/grün: Die durch die Migration eingestellten Prozesse werden final verarbeitet.

Verwandte Themen

- Verbindungsdaten zur Datenbank ändern auf Seite 38
- Konfiguration der Datenbank für eine Test-, Entwicklungs- oder Produktivumgebung auf Seite 39
- Spracheinstellungen für die Anzeige und Pflege der Daten auf Seite 40
- Standardland für Arbeitszeitermittlung und Feiertagen festlegen auf Seite 41

Verbindungsdaten zur Datenbank ändern

Die Verbindungsdaten zur One Identity Manager-Datenbank werden bei der initialen Schemainstallation übernommen. Auf diese Informationen wird auch beim Generieren von Aufträgen für den One Identity Manager Service zugegriffen.

> HINWEIS: Änderungen an diesen Daten sind in der Regel nicht erforderlich und sollten nur von erfahrenen Benutzern vorgenommen werden.

Um die Verbindungsparameter zu ändern

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Datenbanken**.
2. Wählen Sie im Listeneditor die Datenbank.
3. Wählen Sie die Aufgabe **Verbindungsparameter für Datenbank definieren**.
4. Geben Sie die Verbindungsdaten zur Datenbank an.
   - **Server**: Datenbankserver.
   - (Optional) **Windows Authentifizierung**: Gibt an, ob integrierte Windows-Authentifizierung verwendet wird. Die Verwendung dieser Authentifizierung wird nicht empfohlen. Sollten Sie dieses Verfahren dennoch einsetzen, stellen Sie sicher, dass Ihre Umgebung Windows-Authentifizierung unterstützt.
   - **Nutzer**: SQL Server Anmeldename des Benutzer.
   - **Kennwort**: Kennwort für die SQL Server Anmeldung des Benutzer.
   - **Datenbank**: Wählen Sie die Datenbank.
5. Klicken Sie **OK**.

6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Konfiguration der Datenbank für eine Test-, Entwicklungs- oder Produktivumgebung

Über die Staging-Ebene der One Identity Manager-Datenbank legen Sie fest, ob es sich um eine Testdatenbank, Entwicklungsdatenbank oder produktive Datenbank handelt. Über die Staging-Ebene werden einige Datenbankeinstellungen gesteuert.

Wenn Sie die Staging-Ebene der Datenbank ändern, werden die folgenden Einstellungen konfiguriert.

Tabelle 8: Standardeinstellungen für Entwicklungsumgebung, Testumgebung und Produktivumgebung

Einstellung	Entwicklungsumgebung	Testumgebung	Produktivumgebung
Farbe der Statuszeile der One Identity Manager-Werkzeuge	keine	Grün	Gelb
Maximale Laufzeit DBQueue Prozessor	20 Minuten	40 Minuten	120 Minuten
Maximale Anzahl der Slots für DBQueue Prozessor	5	7	Maximale Anzahl der Slots laut Hardwarekonfiguration

Verwandte Themen

- Staging-Ebene der Datenbank ändern auf Seite 40
- Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen auf Seite 548

Staging-Ebene der Datenbank ändern

Um die Staging-Ebene einer Datenbank anzupassen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Datenbanken**.
2. Wählen Sie im Listeneditor die Datenbank.
3. Wählen Sie in der Bearbeitungsansicht den Tabreiter **Allgemein**.
4. Ändern Sie den Wert der Eigenschaft **Staging Ebene** auf **Testumgebung**, **Entwicklungssystem** oder **Produktivsystem**.
5. Bestätigen Sie die Sicherheitsabfrage mit **Ja**.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Konfiguration der Datenbank für eine Test-, Entwicklungs- oder Produktivumgebung auf Seite 39
- Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen auf Seite 548

Spracheinstellungen für die Anzeige und Pflege der Daten

Die Standardinstallation des One Identity Manager wird in den Sprachen **English - United States [en-US]** und **German - Germany [de-DE]** ausgeliefert. Bei Bedarf können Sie weitere Sprachen zur Gestaltung der Benutzeroberfläche und der Ausgabetexte einpflegen. Für diesen Fall müssen Sie vor Einsatz des One Identity Manager die verwendeten Texte in die neue Sprache übersetzen. Für die Übersetzung steht Ihnen im Designer einen Wörterbucheditor zur Verfügung. In den One Identity Manager-Werkzeugen wird die Eingabe mehrsprachiger Inhalte durch ein spezielles Steuerelement unterstützt.

Standardsprache des One Identity Manager

Die Stammdatenpflege erfolgt immer in der Standardsprache. Die Standardsprache einer One Identity Manager-Installation ist **English - United States [en-US]**. Die Standardsprache ist systemweit gültig. Eine Änderung der Standardsprache im laufenden Betrieb wird nicht empfohlen.

Im Idealfall stimmen Standardsprache des One Identity Manager und die Anmeldesprache der Benutzer an den Administrationswerkzeugen überein. Unterscheiden sich die Anmeldesprache des Benutzers und die Standardsprache, so wird die Standardsprache dann verwendet, wenn bei einer sprachabhängigen Datenauflösung für die angeforderte Anmeldesprache des Benutzers keine übersetzten Bezeichnungen gefunden werden.

Anmeldesprache des Benutzers

Die Darstellung der Anzeigetexte in der Benutzeroberfläche erfolgt in der Sprache, mit der sich ein Benutzer an den Administrationswerkzeugen anmeldet. Bei der erstmaligen Anmeldung an den Werkzeugen wird die Systemsprache zur Anzeige der Benutzeroberfläche verwendet. Der Benutzer kann seine Anmeldesprache in jedem Administrationswerkzeug in den Programmeinstellungen ändern. Dabei wird die Anmeldesprache global für alle Werkzeuge, mit denen der Benutzer arbeitet, festgelegt. Somit muss die Einstellung der Anmeldesprache nicht in jedem Werkzeug erneut erfolgen. Die Änderung der Anmeldesprache wird erst mit dem Neustart der Werkzeuge wirksam.

Als Anmeldesprachen werden alle Sprachen angeboten, für die die Option **Wählbar im Frontend** aktiviert ist.

Verwandte Themen

- Anmeldesprachen aktivieren auf Seite 41
- Sprachabhängige Abbildung von Informationen auf Seite 233

Anmeldesprachen aktivieren

Als Anmeldesprachen werden alle Sprachen angeboten, für die die Option **Wählbar im Frontend** aktiviert ist.

Um weitere Anmeldesprachen zu aktivieren

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Sprachkulturen**.
2. Wählen Sie im Listeneditor die Sprache.
3. Setzen Sie in der Ansicht **Eigenschaften** die Eigenschaft **Wählbar im Frontend** auf den Wert **True**.
4. Speichern Sie die Änderungen.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Standardland für Arbeitszeitermittlung und Feiertagen festlegen

Um innerhalb eines Bestellprozesses im IT Shop oder bei der Attestierung Reaktionszeiten von Entscheidern oder Attestierern zu ermitteln, muss die Arbeitszeit der Personen bekannt sein. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für das Identity Management Basismodul*.

Kann für eine Person kein Land ermittelt werden, wird auf das Land zurückgegriffen, das in der Datenbank als Standard eingetragen ist.

Um ein Standardland festzulegen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Datenbanken**.
2. Wählen Sie im Listeneditor die Hauptdatenbank.
3. Wählen Sie in der Bearbeitungsansicht den Tabreiter **Einstellungen**.
4. Wählen Sie in der Eigenschaft **Land (Standard)** das Land.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Konfigurationsparameter zur Systemkonfiguration

Über Konfigurationsparameter konfigurieren Sie die Grundeinstellungen zum Systemverhalten. Der One Identity Manager stellt für verschiedene Konfigurationsparameter Standardeinstellungen zur Verfügung. Prüfen Sie die Konfigurationsparameter und passen Sie die Konfigurationsparameter gegebenenfalls an das gewünschte Verhalten an.

Die Konfigurationsparameter sind in den One Identity Manager Modulen definiert. Jedes One Identity Manager Modul kann zusätzliche Konfigurationsparameter installieren. Einen Überblick über alle Konfigurationsparameter finden Sie im Designer in der Kategorie **Basisdaten > Allgemein > Konfigurationsparameter**.

Detaillierte Informationen zum Thema

- Konfigurationsparameter bearbeiten auf Seite 42

Konfigurationsparameter bearbeiten

Die mitgelieferten Konfigurationsparameter und zulässigen Werte werden durch die Schemainstallation gepflegt. Die Eigenschaften dieser Konfigurationsparameter können Sie nicht bearbeiten. Sie können diese Konfigurationsparameter aktivieren oder deaktivieren und den aktuellen Wert des Konfigurationsparameters festlegen. Weitere Eigenschaften vordefinierter Konfigurationsparameter sind nicht bearbeitbar. Aus der Änderung eines Konfigurationsparameters können Berechnungsaufträge für den DBQueue Prozessor resultieren.

TIPP: Geänderte Konfigurationsparameter werden im Designer in der hierarchischen Ansicht des Konfigurationsparametereditor mit einem gelben Rand gekennzeichnet.

Um Konfigurationsparameter zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Konfigurationsparameter**.
2. Wählen Sie im Konfigurationsparametereditor den Konfigurationsparameter.
3. Wählen Sie in der Ansicht **Konfigurationsparameter** den Tabreiter **Eigenschaften**.
4. Passen Sie die folgenden Einstellungen an.
   - **Aktiviert**: Gibt an, ob der Konfigurationsparameter wirksam ist. Um einen Konfigurationsparameter zu aktivieren, setzen Sie die Option. Um einen Konfigurationsparameter zu deaktivieren, entfernen Sie die Option.
   - **Wert**: Wert des Konfigurationsparameters.

     TIPP: Für einige Konfigurationsparameter können Sie den Wert im Konfigurationsparametereditor mit einem separaten Assistenten bearbeiten. Klicken Sie **...** hinter dem Eingabefeld, um den Assistenten zu starten.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

WICHTIG: Handelt es sich um einen präprozessorrelevanten Konfigurationsparameter, kompilieren Sie die Datenbank.

Verwandte Themen

- Kundenspezifische Konfigurationsparameter erstellen auf Seite 43
- Präprozessorrelevante Konfigurationsparameter auf Seite 370

Kundenspezifische Konfigurationsparameter erstellen

Sollte die Definition weiterer kundenspezifischer Konfigurationsparameter erforderlich sein, können Sie diese unterhalb des Konfigurationsparameters **Custom** einfügen.

Um einen neuen Konfigurationsparameter zu erstellen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Konfigurationsparameter**.
2. Wählen Sie den Konfigurationsparameter **Custom** und fügen Sie über das Kontextmenü **Einfügen** einen neuen Konfigurationsparameter ein.
3. Bearbeiten Sie in der Ansicht **Konfigurationsparameter** auf dem Tabreiter **Eigenschaften** die Stammdaten des Konfigurationsparameters.
4. (Optional) Soll ein Konfigurationsparameter nur bestimmte Werte zulassen, legen Sie auf dem Tabreiter **Optionen** die zulässigen Werte fest.

- Um eine neue Option zu erstellen, klicken Sie **Einfügen**.
- Um eine Option zu entfernen, klicken Sie **Löschen**.

5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Konfigurationsparameter bearbeiten auf Seite 42
- Eigenschaften der Konfigurationsparameter auf Seite 44
- Optionen für Konfigurationsparameter auf Seite 46
- Präprozessorrelevante Konfigurationsparameter auf Seite 370

Eigenschaften der Konfigurationsparameter

Tabelle 9: Eigenschaften der Konfigurationsparameter

Eigenschaft	Beschreibung
Vollständiger Name	Vollständige Name des Konfigurationsparameters. Dieser setzt sich zusammen aus dem Namen des Parameters und den Namen seiner übergeordneten Parameter
Parameter	Technische Name des Konfigurationsparameters.
Anzeigename	Der Anzeigename dient zur Anzeige der Konfigurationsparameter. Der Anzeigename kann sprachabhängig hinterlegt werden. HINWEIS: Um die Anzeigenamen im Konfigurationsparametereditor einzublenden, wählen Sie den Menüeintrag **Konfigurationsparameter > Beschriftungen anzeigen**. Konfigurationsparameter, für die kein Anzeigename hinterlegt ist, werden in diesem Modus in Klammern (<<>>) angezeigt. Zusätzlich wird ein Tooltip mit dem technischen Namen angezeigt.
Reihenfolge	Die Reihenfolge beeinflusst die Sortierung der Konfigurationsparameter im Konfigurationsparametereditor. HINWEIS: Die Reihenfolge wirkt nur wenn die Anzeigenamen im Konfigurationsparametereditor angezeigt werden.
Wert	Wert des Konfigurationsparameters. Für jeden Konfigurationsparameter müssen Sie einen Wert angeben. Auch übergeordnete Konfigurationsparameter, die nur zur Gliede-

Eigenschaft	Beschreibung
	rung dienen, dürfen nicht leer sein, da sonst auf die untergeordneten Konfigurationsparameter nicht zugegriffen werden kann. Einige Konfigurationsparameter besitzen mehrere zulässige Werte. Diese werden über die Konfigurationsparameteroptionen vorgegeben und sind hier auswählbar. Zusätzlich wird die Beschreibung der gewählten Option angezeigt. TIPP: Für einige Konfigurationsparameter können Sie den Wert im Konfigurationsparametereditor mit einem separaten Assistenten bearbeiten. Klicken Sie **...** hinter dem Eingabefeld, um den Assistenten zu starten.
Beschreibung	Beschreibung des Konfigurationsparameters. Im Konfigurationsparametereditor klicken Sie die Schaltfläche **Bearbeiten**, um die Beschreibung zu bearbeiten.
Aktiviert	Gibt an, ob der Konfigurationsparameter wirksam ist. Ist die Option gesetzt, ist der Konfigurationsparameter aktiv. Ist diese Option nicht gesetzt, gilt der gesamte Teilbaum ab dieser Stelle als deaktiviert und der Konfigurationsparameter sowie seine untergeordneten Konfigurationsparameter als nicht vorhanden.
Verschlüsselt	Konfigurationsparameter, die mit dieser Option gekennzeichnet sind, enthalten verschlüsselte Daten, beispielsweise Kennworte. Bei Eingabe eines neuen Wertes wird dieser sofort verschlüsselt.
Präprozessorrelevanter Parameter	Gibt an, ob es sich um einen präprozessorrelevanten Konfigurationsparameter handelt. Für die zugehörige Konfigurationsparameteroption ist ein Präprozessorausdruck eingetragen. HINWEIS: Durch die Aktivierung eines präprozessorrelevanten Konfigurationsparameters gilt die Präprozessorbedingung systemweit als vorhanden. Erst durch die Kompilierung der Datenbank wird die Präprozessorbedingung wirksam. Jede Änderung an einem präprozessorrelevanten Konfigurationsparameter und den Optionen erfordert eine Kompilierung der Datenbank.

Verwandte Themen

- Konfigurationsparameter bearbeiten auf Seite 42
- Kundenspezifische Konfigurationsparameter erstellen auf Seite 43
- Optionen für Konfigurationsparameter auf Seite 46
- Präprozessorrelevante Konfigurationsparameter auf Seite 370

Optionen für Konfigurationsparameter

Soll ein Konfigurationsparameter nur bestimmte Werte zulassen, sind diese zulässigen Werte über die Konfigurationsparameteroptionen definiert.

Tabelle 10: Eigenschaften von Optionen

Eigenschaft	Beschreibung
Wert	Zulässiger Wert des Konfigurationsparameters.
Beschreibung	Beschreibung der Konfigurationsparameteroption.
Präprozessorausdruck	Für präprozessorrelevante Konfigurationsparameter wird der zulässige Präprozessorausdruck in den Parameteroptionen angegeben. Dieser Präprozessorausdruck kann als Präprozessorbedingung für die bedingte Kompilierung verwendet werden.

Verwandte Themen

- Konfigurationsparameter bearbeiten auf Seite 42
- Kundenspezifische Konfigurationsparameter erstellen auf Seite 43
- Eigenschaften der Konfigurationsparameter auf Seite 44
- Präprozessorrelevante Konfigurationsparameter auf Seite 370

5

Grundlagen zum One Identity Manager Schema

Das Datenmodell des One Identity Manager ist logisch in Module gruppiert. Die Module sind über Vorgänger-Beziehungen verknüpft. Ein Modul kann einen oder mehrere Vorgänger besitzen. Jedes Modul erweitert Schema um eigene Tabellen und Spalten und installiert eigene Standardobjekte wie beispielsweise eigene Bildungsregeln, Skripte oder Prozesse.

Die Funktionen eines Moduls sind erst verfügbar, wenn das Modul in der Datenbank vorhanden ist. Somit sind beispielsweise die Attestierungsfunktionen erst verfügbar, wenn das Modul Attestierung vorhanden ist. Sollen die Berichtsfunktionen des One Identity Manager verwendet werden, muss das Modul Berichtsabonnement in der Datenbank vorhanden sein.

Immer vorhanden ist das Konfigurationsmodul. Das Konfigurationsmodul enthält die Metadaten zur Beschreibung des Anwendungsdatenmodells sowie Skripte für Format- und Bildungsvorschriften oder bedingte Wechselwirkungen. Die Systemkonfiguration des One Identity Manager, die Einstellungen zur Steuerung der Frontends und die Queues für asynchrone Verarbeitung der Daten und Prozesse sind ebenfalls Teil der Metadaten. Die Metadaten werden durch das Systemdatenmodell beschrieben.

Die weiteren Module enthalten alle für die Datenpflege nötigen Informationen (Nutzdaten) wie beispielsweise Informationen über Personen, Benutzerkonten, Gruppen, Mitgliedschaften und Betriebsdaten, Genehmigungsworkflows, Attestierungen, Rezertifizierungen und Complianceregeln. Die Nutzdaten werden durch das Anwendungsdatenmodell beschrieben.

Die Tabellendefinitionen sind in der Tabelle `DialogTable` hinterlegt. Die Spaltendefinitionen aller Tabellen sind in der Tabelle `DialogColumn` hinterlegt. Die Tabellenbeziehungen und Spaltenbeziehungen sind in den Tabellen `QBMRelation` und `DialogValidDynamicRef` hinterlegt.

Verwandte Themen

- Übersicht über das One Identity Manager Schema auf Seite 48
- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52

- Hinweise zur Bearbeitung von Tabellendefinitionen und Spaltendefinitionen auf Seite 56
- Tabellendefinitionen auf Seite 57
- Spaltendefinitionen auf Seite 81
- Tabellenbeziehungen auf Seite 109
- Dynamische Fremdschlüssel auf Seite 113
- Erweiterung des One Identity Manager Schemas um kundenspezifische Tabellen oder Spalten auf Seite 475
- Softwarearchitektur des One Identity Manager auf Seite 18

Übersicht über das One Identity Manager Schema

Die Abbildung und Bearbeitung des Datenmodells erfolgt im Designer in der Kategorie **One Identity Manager Schema**. In dieser Kategorie werden die Standardtabellen des One Identity Manager sowie die kundenspezifischen Tabellen mit Ihren Eigenschaften dargestellt. Sie erhalten einen Überblick über Anpassungen an der Standardkonfiguration, die Bildungsregeln und Formatierungsvorschriften der Datenbankspalten.

Um die Schemaübersicht anzuzeigen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Öffnen Sie die Schemaübersicht über die Aufgabe **One Identity Manager Schema anzeigen**.

TIPP: Bei Auswahl einer Tabelle oder einer Spalte im Designer können Sie die Schemaübersicht über die Aufgaben **Tabelle <Tabellenname> im Schema anzeigen** und **Spalte <Spaltenname> im Schema anzeigen** öffnen.

Die Schemaübersicht verfügt über zwei Modi zur Darstellung des One Identity Manager Schemas.

- Darstellung des Datenmodells

  In diesem Modus erhalten Sie einen Überblick über alle Tabellen mit ihren Spalten sowie die Tabellenbeziehungen.

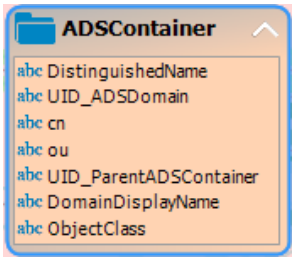
- Darstellung der Abhängigkeiten

  In diesem Modus werden nur die Tabellen mit den Spalten dargestellt, für die aufgrund von Bildungsregeln Abhängigkeiten bestehen. Tabellen und Spalten ohne Abhängigkeiten werden nicht angezeigt.

Die Darstellung der Tabellen und ihrer Spalten erfolgt über ein spezielles Steuerelement. In der Kopfzeile des Steuerelementes wird die Bezeichnung der Datenbanktabelle angezeigt, alle weiteren Einträge repräsentieren die Spalten dieser Tabelle. Jeder Eintrag des Steuerelementes verfügt über einen Tooltip. Der Inhalt des Tooltip ist abhängig vom

gewählten Darstellungsmodus. Die Spalteneinträge sind mit Symbolen gekennzeichnet, die abhängig vom gewählten Darstellungsmodus bestimmte Spalteneigenschaften repräsentieren.

Abbildung 6: Steuerelement zur Darstellung der Datenbanktabellen und ihrer Spalten



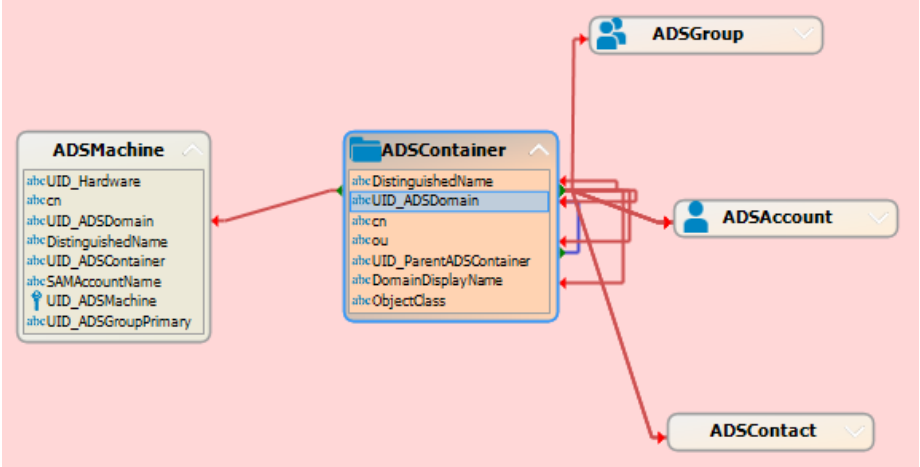
Über die Menüeinträge **Optionen > Alle Spalten einblenden** und **Optionen > Alle Spalten ausblenden** oder das Symbole in der Kopfzeile des Steuerelementes steuern Sie die Anzeige der Spalteneinträge. Über den Menüeintrag **Optionen > Kleine Tabellen ausblenden** wird für Tabellen mit maximal 20 Spalteneinträgen nur die Bezeichnung der Tabelle eingeblendet.

Um Tabellen und Spalten anzuzeigen, die über Präprozessorbedingungen deaktiviert sind, verwenden Sie das Menü **Optionen > Deaktivierte Spalten anzeigen**.

Die Layoutpositionen der Steuerelemente können Sie innerhalb der Schemaübersicht mausgesteuert verändern. Über den Menüeintrag **Optionen > Tabellenanordnung speichern** werden die geänderten Layoutpositionen zunächst in die interne Datenbank gespeichert und in das Änderungsprotokoll des Designers übernommen.

Beziehungen zwischen Tabellen oder Spalten werden über Verbinder dargestellt. Über den Menüeintrag **Optionen > Tabellenbeziehungen ausblenden** regeln Sie die Anzeige der Verbinder. Ist dieser Menüeintrag eingeschaltet werden keine Verbinder dargestellt. Ist der Menüeintrag ausgeschaltet, werden alle Verbinder angezeigt. Unabhängig von dieser Option werden bei Aktivierung eines Steuerelementes die Verbinder zu diesem Element farbig hervorgehoben.

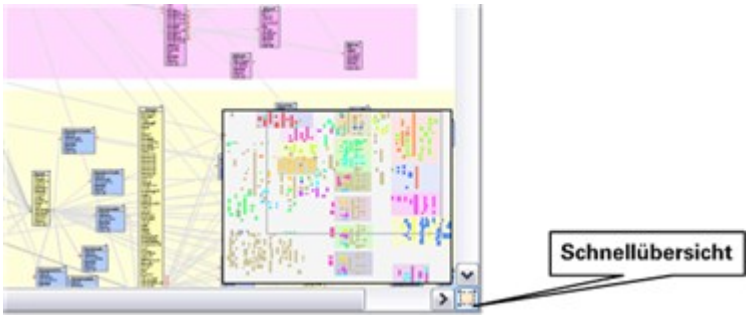
Abbildung 7: Darstellung von Beziehungen über Verbinder



Ein Verbinder zeigt auf die Spalteneinträge, die miteinander in Bezug stehen. Über die Verbinder können Sie zwischen den Bezugspunkten navigieren. Bei Auswahl eines Verbinders wechselt der Mauszeiger zum Pfeilsymbol, bei Maus-Doppelklick auf einen Verbinder wird von einem Bezugspunkt zum anderen Bezugspunkt gewechselt. Die Navigationsrichtung wird durch die Pfeilausrichtung angezeigt. Die Bewegung wird über den Menüeintrag **Optionen > Bewegung animieren**gesteuert. Wenn der Mauszeiger einen Verbinder berührt, wird ein Tooltip angezeigt, dessen Inhalt vom gewählten Darstellungsmodus abhängig ist.

Um innerhalb der Schemaübersicht schnell zu navigieren, können Sie die Schnellübersicht nutzen. Die Schnellübersicht öffnen Sie über die Schaltfläche am rechten unteren Rand der Schemaübersicht. Der aktuelle Bildschirmausschnitt wird in der Schnellübersicht mit einem Rahmen gekennzeichnet. Diesen Rahmen können Sie innerhalb der Schnellübersicht mausgesteuert verschieben. Die Schemaübersicht wird an den gewählten Bildschirmausschnitt angepasst.

Abbildung 8: Schnellübersicht öffnen



Verwandte Themen

- Datenmodell im Designer anzeigen auf Seite 50
- Spaltenabhängigkeiten aufgrund von Bildungsregeln anzeigen auf Seite 51

Datenmodell im Designer anzeigen

In diesem Modus erhalten Sie einen Überblick über alle Tabellen mit ihren Spalten sowie die Tabellenbeziehungen.

Um das Datenmodell anzuzeigen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Öffnen Sie die Schemaübersicht über die Aufgabe **One Identity Manager Schema anzeigen**.
3. Wählen Sie den Menüeintrag **Optionen > Datenmodell**.

Der Tooltip eines Tabelleneintrags enthält den Namen der Tabelle sowie die Präprozessorbedingungen der Tabelle. Der Tooltip eines Spalteneintrags enthält den

Namen der Spalte, die Beschreibung, den Datentyp sowie die minimale Länge und die maximale Länge der Spalte.

Der Tooltip eines Verbinders zeigt die Tabellenbeziehungen an. Der Inhalt des Tooltip enthält die Namen der Tabellen, die miteinander in Bezug stehen, sowie die Eigenschaften der Tabellenbeziehung. Bei einem Mausklick auf einen Verbinder werden in der Bearbeitungsansicht die Eigenschaften der Tabellenbeziehungen dargestellt.

Im Steuerelement sind die Spalteneinträge mit Symbolen gekennzeichnet, welche spezielle Eigenschaften wie beispielsweise den .Net-Datentyp der Spalte repräsentieren.

Tabelle 11: Bedeutung der Symbole

Symbol	Bedeutung
	Die Spalte ist eine Fremdschlüsselspalte (FK).
	Die Spalte ist eine Primärschlüsselspalte (PK).
abc	Die Spalte ist vom Datentyp `String` oder `Text`.
	Die Spalte ist vom Datentyp `Binary`.
	Die Spalte ist vom Datentyp `Bool`.
123	Die Spalte ist vom Datentyp `Int`, `Byte` oder `Short`.
1.3	Die Spalte ist vom Datentyp `Double` oder `Decimal`.
	Die Spalte ist vom Datentyp `Date`.

Verwandte Themen

- Spaltenabhängigkeiten aufgrund von Bildungsregeln anzeigen auf Seite 51

Spaltenabhängigkeiten aufgrund von Bildungsregeln anzeigen

In diesem Modus werden nur die Tabellen mit den Spalten dargestellt, für die aufgrund von Bildungsregeln Abhängigkeiten bestehen. Tabellen und Spalten ohne Abhängigkeiten werden nicht angezeigt.

Um die Spaltenabhängigkeiten anzuzeigen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Öffnen Sie die Schemaübersicht über die Aufgabe **One Identity Manager Schema anzeigen**.
3. Wählen Sie den Menüeintrag **Optionen > Abhängigkeiten**.

Der Tooltip eines Tabelleneintrags enthält den Namen der Tabelle. Der Tooltip eines Spalteneintrags enthält den Namen der Spalte. Enthält eine Spalte eine Bildungsregel, so

wird im Tooltip diese Bildungsregel angezeigt. Hat eine Spalte selbst keine Bildungsregel, wird aber in Bildungsregeln anderer Spalten referenziert, so werden im Tooltip diese betroffenen Spalten angezeigt.

Bei Auswahl einer Spalte werden die Verbindungen zu anderen Spalten farblich gekennzeichnet. Der Tooltip eines Verbinders zeigt die Sender-Empfänger-Beziehung der Spaltenabhängigkeiten an. Der Inhalt des Tooltip enthält die Namen der Tabellen, die miteinander in Bezug stehen. Zusätzlich werden der Sender, der Empfänger und der Teil der Bildungsregel, auf den sich die Abhängigkeit begründet, angezeigt.

Tabelle 12: Bedeutung der Farben für Sender-Empfänger-Beziehungen

Farbe	Bedeutung
Grün	Die Spalte ist der Sender.
Rot	Die Spalte ist der Empfänger.

Verwandte Themen

- Datenmodell im Designer anzeigen auf Seite 50

Tabellenarten und Standardspalten im One Identity Manager Datenmodell

Auf Datenbankebene werden im One Identity Manager verschiedene Arten von Tabellen verwendet.

Tabelle 13: Tabellenarten

Art der Tabelle	Beschreibung
Einfache Tabelle	Einfache Tabellen sind die gebräuchlichste Form zur Ablage von Daten. Für einfache Tabellen werden folgende Spalten definiert • ein Primärschlüssel, der aus einer Spalte besteht • ein Objektschlüssel (`XObjectKey`)
M:N-Tabelle	M:N-Tabellen sind Tabellen, in der Beziehungen zwischen zwei anderen Tabellen hergestellt werden. Für M:N-Tabellen werden folgende Spalten definiert • ein zweispaltiger Primärschlüssel Beide Spalten werden als Fremdschlüsselspalten auf die referenzierten Tabellen definiert. • ein Objektschlüssel (`XObjectKey`)

Art der Tabelle	Beschreibung
	M:N-Tabellen werden in dieser Dokumentation auch als Zuordnungstabellen bezeichnet.
M:all-Tabelle	M:all-Tabellen sind eine besondere Art von Zuordnungstabellen, die für den One Identity Manager entwickelt wurden. M:all-Tabellen werden eingesetzt, wenn ein Anteil einer Zuweisung (all) auf unterschiedliche Tabellen verweisen kann, also dynamisch ermittelt wird. Die zulässigen Tabellen können dabei eingeschränkt werden. Beispielsweise kann der Eigentümer einer Gruppe ein Benutzerkonto oder eine Gruppe sein. Des Weiteren werden M:all-Tabellen verwendet, wenn zu einer Zuweisung zusätzliche Informationen abgebildet werden sollen, beispielsweise Gültigkeitszeiträume einer Zuweisung. Für M:all-Tabellen werden folgende Spalten definiert • ein Primärschlüssel • ein, als `NOT NULL` definierter, Fremdschlüssel, der den Primärschlüssel einer andere Tabelle referenziert • ein, als `NOT NULL` definierter, dynamischer Fremdschlüssel, der auf den Objektschlüssel (`XObjectKey`) der zulässigen Tabellen verweist • ein Objektschlüssel (`XObjectKey`) Es können weitere Fremdschlüssel und dynamische Fremdschlüssel definiert sein. Diese Spalten müssen als `NULL` definiert sein.
Arbeitstabellen	Arbeitstabellen dienen zur Ablage von Daten, für die keine Objekt über die Objektschicht erzeugt werden. Für Arbeitstabellen ist kein Primärschlüssel erforderlich. Es können jedoch maximal zwei Primärschlüssel definiert werden.

Tabelle 14: Standardspalten

Spalte	Beschreibung
Primärschlüssel	• Wenn aus der Tabelle Objekte über die Objektschicht erzeugt werden, benötigt die Tabelle einen Primärschlüssel. • Repräsentiert eine Tabelle eine M:N-Abbildung ist ein zweispaltiger Primärschlüssel definiert. Beide Primärschlüsselspalten werden als Fremdschlüsselspalten auf die referenzierten Tabellen definiert. • Für Arbeitstabellen ist kein Primärschlüssel erforderlich. • Primärschlüsselspalten müssen im Globally Unique Identifier (GUID) Format definiert werden.

Spalte	Beschreibung
	Standard GUID's werden im Format `[0-9,a-f](8-4-4-4-12)` erzeugt. Vordefinierte Modul GUID's werden im Format `<MMM>-[0-9,a-f](32)` abgebildet, wobei `<MMM>` dem Modulkürzel entspricht. Kundenspezifsche Modul GUID's werden im Format `<CCC>-[0-9,a-f](32)` erzeugt. Weitere Informationen finden Sie unter Arbeiten mit Modul Globally Unique Identifier auf Seite 68.

XObjectKey	Wenn aus der Tabelle Objekte über die Objektschicht erzeugt werden, benötigt die Tabelle zwingend eine Objektschlüsselspalte. Der Objektschlüssel (`XObjectKey`) ist ein eineindeutiger Schlüssel, der in der Lage ist jedes Objekt in der Datenbank zu referenzieren.
	
	Syntax eines `XObjectKey`:
	
	`<Key><T>TableName</T><P>PrimaryKeyOfRow</P></Key>`
	
	mit:
	
	• `TableName`: Bezeichnung der Tabelle
	• `PrimaryKeyOfRow`: GUID der Primärschlüsselspalte
	
	Für zweispaltige Primärschlüssel wird ein zusätzlicher `<P>SecondPrimaryKeyOfRow</P>` verwendet. Die Sortierung der im `XObjectKey` verwendeten Spalten ist abhängig von der Bezeichnung der Fremdschlüsselspalten (alphabetische Sortierung).
	
	Beispiel:
	
	Tabelle `PersonInProfitcenter`
	
	`<Key><T>PersonInProfitCenter</T><P><UID_Person></P><P><>UID_Profitcenter</P></Key>`
	
	Tabelle `PersonInDepartment`
	
	`<Key><T>PersonInDepartment</T><P><UID_Department></P><P><>UID_Person</P></Key>`

Fremdschlüssel	• Die Bezeichnung der Fremdschlüsselspalte entspricht, soweit möglich, der Bezeichnung der Primärschlüssel der referenzierten Tabelle. • Fremdschlüsselspalten sind im GUID Format definiert. • Die Referenzierung einer Tabelle erfolgt über den Primärschlüssel der referenzierten Tabelle. • Ist die Fremdschlüsselspalte Teil einer M:all-Tabelle, dann wird die Spalte im One Identity Manager Schema mit der

Spalte	Beschreibung
	Option **Teil des Schlüssels einer M:all-Tabelle** (`DialogColumn.IsMAllKeyMember`) gekennzeichnet.
Dynamische Fremd-schlüssel	• Dynamische Fremdschlüssel werden verwendet, wenn eine Referenz auf unterschiedliche Tabellen verweisen kann. Beispielsweise kann der Manager eines Benutzerkontos (Tabelle `ADSAccount.ObjectKeyManager`) ein anderes Benutzerkonto (Tabelle `ADSAccount`) oder ein Kontakt (Tabelle `ADSContact`) sein. • Dynamische Fremdschlüssel verweisen auf den Objektschlüssel (`XObjectKey`) der zulässigen Tabellen. • Die zulässigen Tabellen können eingeschränkt werden. Ist keine Einschränkung festgelegt, sind alle Tabellen zulässig. • Ein dynamischer Fremdschlüssel wird im One Identity Manager Schema mit der Option **Dynamische Fremdschlüsselspalte** (`DialogColumn.IsDynamicFK`) gekennzeichnet. • Ist die dynamische Fremdschlüsselspalte Teil einer M:all-Tabelle, dann wird die Spalte im One Identity Manager Schema mit der Option **Teil des Schlüssels einer M:all-Tabelle** (`DialogColumn.IsMAllKeyMember`) gekennzeichnet.
XDateInserted	Die Spalten enthalten die Informationen darüber, welcher Benutzer zu welchem Zeitpunkt Änderungen vorgenommen hat. Die Spalten sind immer gemeinsam vorhanden.
XDateUpdated	
XUserInserted	
XUserUpdated	
XTouched	Die Spalte enthält den Bearbeitungsstatus eines Elementes. Der Bearbeitungsstatus wird bei der Erstellung von Kundenkonfigurationspaketen genutzt.
XMarkedForDeletion	Die Spalte enthält Informationen, ob das Objekt zum Löschen markiert ist. Die Spalte ist vorhanden, wenn: • Die Funktion des verzögerten Löschens an der Tabelle genutzt werden kann. • Die Tabelle mittels Synchronization Engine gegen eine Zielsystemumgebung abgeglichen wird und die Behandlung ausstehender Objekte erfolgen kann.
XOrigin	Zur Ermittlung der Herkunft einer Zuweisung ist an einer M:N-Tabelle oder M:all-Tabelle eine Spalte `XOrigin` definiert. Die einzelnen Bit-Positionen geben die Herkunft einer Mitgliedschaft an.

Spalte	Beschreibung
	Ausführliche Informationen zur Berechnung der Zuweisungen finden Sie im *One Identity Manager Administrationshandbuch für das Identity Management Basismodul*.
XIsInEffect	• Zur Ermittlung, ob eine Zuweisung wirksam ist, ist an einer Zuordnungstabelle eine Spalte `XIsInEffect` definiert. • Die Spalte ist nur vorhanden, wenn sich die Anzahl der Zuweisungen von der Anzahl der wirksamen Zuweisungen unterscheiden kann. Ist beispielsweise eine Person deaktiviert, zum Löschen markiert oder als sicherheitsgefährdend eingestuft, so kann für diese Person die Vererbung der Unternehmensressourcen unterbunden werden. Die Zuweisung der Gruppen bleibt erhalten, diese Zuweisung wird jedoch nicht wirksam. • Wenn Spalte `XIsInEffect` genutzt wird, muss eine Spalte `XOrigin` vorhanden sein. Ausführliche Informationen zur Berechnung der Zuweisungen finden Sie im *One Identity Manager Administrationshandbuch für das Identity Management Basismodul*.
XDateSubItem	Die Spalte enthält das Änderungsdatum für Abhängigkeiten und wird benötigt, um Änderungen von Mitgliedschaften in einem Zielsystem bei der Synchronisation und Provisionierung zu berücksichtigen. Ausführliche Informationen zur Synchronisation und Provisionierung von Mitgliedschaften finden Sie im *One Identity Manager Referenzhandbuch für die Zielsystemsynchronisation*.

Hinweise zur Bearbeitung von Tabellendefinitionen und Spaltendefinitionen

- Die Tabellen und Spalten des One Identity Manager Schemas können Sie in großem Umfang an Ihre Anforderungen anpassen. Die Tabellen und Spalten bearbeiten Sie im Designer im Schemaeditor.
- Die Standardkonfiguration wird bei Bearbeitung in einen Konfigurationsspeicher verschoben. Sie können Änderungen aus dem Konfigurationsspeicher zurückholen und somit die Standardkonfiguration wiederherstellen.

    - Änderungen an Daten werden durch das Symbol vor dem geänderten Wert gekennzeichnet. Solange die Änderungen noch nicht gespeichert sind, können Sie diese mit einem Mausklick auf das Symbol rückgängig machen.
    - Änderungen zur Standardkonfiguration werden im Designer mit dem Symbol gekennzeichnet. Um die Standardkonfiguration wieder herzustellen, klicken Sie auf das Symbol.
- Kundenspezifisch angepasste Standardtabellen und -spalten werden im Designer in der Kategorie **One Identity Manager Schema > Angepasste Tabellen** angezeigt. Im Schema Schemaeditor werden die Tabellendefinitionen und Spaltendefinitionen mit einem Sternchen (*) gekennzeichnet. Ein Tooltip zeigt weitere Informationen zu den Anpassungen.
- Einige Änderungen an Tabellen und Spalten erfordern die Kompilierung der Datenbank.
- Setzen Sie zur Erweiterung des One Identity Manager Schemas um kundenspezifische Tabellen oder Spalten das Programm Schema Extension ein. Durch das Programm Schema Extension werden die Schemaerweiterungen in der Datenbank angelegt und die notwendigen Erweiterungen im One Identity Manager Schema sichergestellt.

  Anschließend passen Sie im Designer die Tabellendefinitionen und Spaltendefinitionen weiter an.
- Kundenspezifische Tabellen werden im Designer **One Identity Manager Schema > Kundenspezifische Tabellen** angezeigt.
- Einen Überblick über die im System vorhandenen Spalten mit Bildungsregeln erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Bildungsregeln**. Die Abhängigkeiten von Spalten aufgrund von Bildungsregeln werden in der Schemaübersicht im Schemaeditor dargestellt.
- Einen Überblick über die im System vorhandenen Spalten mit vordefinierten Formatierungstypen oder Formatierungsskripten erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Formatvorschriften**.
- Berichte zur Systemkonfiguration und kundenspezifischen Anpassungen von Tabellen und Spalten finden Sie im Designer in der Kategorie **Dokumentation**.

Verwandte Themen

- Kundenspezifische Änderungen an der One Identity Manager Standardkonfiguration auf Seite 28

Tabellendefinitionen

Die Tabellendefinitionen der One Identity Manager-Module sind in der Tabelle `DialogTable` hinterlegt. Die vordefinierten Tabellendefinitionen des One Identity Manager Schemas werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar.

Die Tabellendefinitionen des One Identity Manager Schemas bearbeiten Sie im Designer im Schemaeditor.

Detaillierte Informationen zum Thema

- Hinweise zur Bearbeitung von Tabellendefinitionen und Spaltendefinitionen auf Seite 56
- Tabellentypen im One Identity Manager auf Seite 58
- Tabellenskripte auf Seite 67
- Arbeiten mit Modul Globally Unique Identifier auf Seite 68
- Löschverzögerung für Objekte festlegen auf Seite 71
- Tabellendefinitionen bearbeiten auf Seite 73
- Eigenschaften von Tabellendefinitionen auf Seite 73
- Unterstützung von Dateigruppen auf Seite 115

Tabellentypen im One Identity Manager

Für den Zugriff über die Objektschicht sind die Tabellen im One Identity Manager Schema mit einem Tabellentyp gekennzeichnet. Abhängig vom Tabellentyp werden bei der Tabellendefinition weitere Eigenschaften benötigt.

Tabelle 15: Tabellentypen im One Identity Manager Schema

Tabellentyp	Bedeutung
Tabelle	Der Tabellentyp **Tabelle** wird für einfache Tabellen, M:N-Tabellen, M:all-Tabellen und Arbeitstabellen verwendet.
Basistabelle	Der Tabellentyp **Basistabelle** wird für einfache Tabellen, M:N-Tabellen und M:all-Tabellen verwendet, um daraus Datenbanksichten mit dem Typ **View** zu definieren. Beispiele für Basistabellen sind die Tabelle `BaseTree` zur Abbildung von Rollen und Organisationen und die Zuordnungstabellen `BasetreeHas*` für die Zuordnung von Unternehmensressourcen zu Organisationen und Rollen.
View	Der Tabellentyp **View** wird für Datenbanksichten auf Tabellen mit dem Typ **Basistabelle** verwendet. Datenbanksichten mit dem Typ **View** repräsentieren Teilausschnitte der zugrunde liegenden Tabellen. Datenbanksichten mit dem Typ **View** werden hauptsächlich für die Abbildung von Rollen verwendet. So sind beispielsweise die Datenbanksichten `Department`, `Locality` und `Profitcenter` Teilausschnitte der Basistabelle `Basetree`.
Proxy	Der Tabellentyp **Proxy** wird für Datenbanksichten auf Tabellen mit dem Typ **Tabelle** oder auf Datenbanksichten mit dem Typ **View** verwendet.

Tabellentyp	Bedeutung
	Datenbanksichten mit dem Typ **Proxy** sind Vereinigungssichten verschiedener Tabellen. Die Abbildung der Spalten zwischen einer Datenbanksicht mit dem Typ **Proxy** und den zugrunde liegenden Tabellen erfolgt über die Spaltendefinitionen und über die Erweiterungen zur Proxyview. Datenbanksichten mit dem Typ **Proxy** werden vorrangig für die Abbildung im Unified Namespace eingesetzt.
Union	Der Tabellentyp **Union** wird für Datenbanksichten auf Tabellen mit dem Typ **Tabelle** oder auf Datenbanksichten mit den Typen **View** oder **Proxy** verwendet. Datenbanksichten mit dem Typ **Union** sind Vereinigungssichten verschiedener Tabellen und dienen zur Zusammenfassung verschiedener Objekttypen gleichen Kontextes. So wird beispielsweise in der Datebanksicht `QERAccProductUsage` ermittelt, welche Leistungspositionen in welchen IT Shop Produkten verwendet werden. Datenbanksichten mit dem Typ **Union** werden hauptsächlich bei der Bearbeitung der Benutzeroberfläche und für die Erstellung von Berichten verwendet.
Read only	Der Tabellentyp **Read only** wird für Datenbanksichten auf Tabellen mit dem Typ **Tabelle** oder auf Datenbanksichten mit den Typen **View**, **Proxy** oder **Union** verwendet. Datenbanksichten mit dem Typ **Read only** können sowohl Teilausschnitte als auch Vereinigungsmengen der zugrunde liegenden Tabellen sein. Datenbanksichten mit dem Typ **Read only** dienen nur zur Anzeige und werden hauptsächlich bei der Bearbeitung der Benutzeroberfläche und für die Erstellung von Berichten verwendet.

Verwandte Themen

- Datenbanksichten vom Typ View auf Seite 59
- Datenbanksichten vom Typ Proxy auf Seite 61
- Datenbanksichten vom Typ Union auf Seite 63
- Datenbanksichten vom Typ Read only auf Seite 66

Datenbanksichten vom Typ View

Datenbanksichten mit dem Typ **View** repräsentieren Teilausschnitte der zugrunde liegenden Tabellen. Datenbanksichten mit dem Typ **View** werden hauptsächlich für die Abbildung von Rollen verwendet. So sind beispielsweise die Datenbanksichten `Department`, `Locality` und `Profitcenter` Teilausschnitte der Basistabelle `Basetree`.

Datenbanksichten mit dem Typ **View** sind vordefinierte Datenbanksichten. Für die Spalten dieser Datenbanksichten können Bildungsregeln und Formatierungsregeln definiert werden.

Zur Definition einer Datenbanksichten vom Typ **View** werden folgende Informationen verwendet.

Tabelle 16: Eigenschaften für die Definition einer Datenbanksicht vom Typ View

Eigenschaft	Bedeutung
Tabelle	Bezeichnung der Tabelle im Datenmodell.
Typ	Typ der Tabelle **View**.
Basistabelle	Basistabelle, auf der die Datenbanksicht basiert.
Bedingung für Sichtdefinition	Einschränkende Bedingung für die Erstellung der Datenbanksicht als Where-Klausel für Datenbankabfragen. Die Bedingung bezieht sich auf die zugrunde liegende Basistabelle.
Spalten	Für jede Spalte der Datenbanksicht wird ein Verweis auf die Spalte der zugrunde liegenden Basisspalte benötigt. Die Zuordnung treffen Sie in der Spaltendefinition.
Einzufügende Werte	Standardeinstellungen für einzelne Spalten, die beim Einfügen eines neuen Datensatzes zugewiesen werden. Die Angaben erfolgt in VB.Net-Syntax.
Auswahlskript	Auswahlskript als VB.Net-Ausdruck, um zur Laufzeit zu bestimmen, ob das übergebene Objekt zu dieser Datenbanksicht gehört.

Beispiel:

Die Tabelle `Department` ist als Datenbanksicht vom Typ **View** definiert. Bei Einfügen von Daten in die Tabelle `Department` soll die Spalte `UID_OrgRoot` mit dem Wert **QER-V-Department** belegt werden.

Tabelle 17: Beispiel für Definition einer Datenbanksicht vom Typ "View"

Eigenschaft	Wert
Tabelle	Department
Typ	View
Basistabelle	BaseTree
Bedingung für Sichtdefinition	UID_OrgRoot = 'QER-V-Department'
Einzufügende Werte	base.putvalue("UID_OrgRoot", "QER-V-Department")
Auswahlskript	Value = (String.Equals($UID_OrgRoot$, "QER-V-Department", StringComparison.OrdinalIgnoreCase))

Eigenschaft	Wert
Spalten-->Basisspalten (Auszug aus der Spaltendefinition)	Department.DepartmentName-->BaseTree.Ident_Org Department.Description-->BaseTree.Description
Resultierende Sichtdefinition	create view dbo.Department as select Ident_Org as DepartmentName, Description as Description, ... from BaseTree where UID_OrgRoot = 'QER-V-Department'

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73
- Eigenschaften von Spaltendefinitionen auf Seite 99
- Definieren von Einfügewerten auf Seite 147
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483
- Datenbanksichten vom Typ Proxy auf Seite 61
- Datenbanksichten vom Typ Union auf Seite 63
- Datenbanksichten vom Typ Read only auf Seite 66

Datenbanksichten vom Typ Proxy

Datenbanksichten mit dem Tabellentyp **Proxy** sind Vereinigungssichten verschiedener Tabellen. Die Abbildung der Spalten zwischen einer Datenbanksicht vom Typ **Proxy** und den zugrunde liegenden Tabellen erfolgt über die Spaltendefinitionen und über die Erweiterungen zur Proxyview. Der DBQueue Prozessor berechnet abhängig von der Spaltenabbildung die aktuelle Sichtdefinition. Dabei werden nur Tabellen berücksichtigt, die nicht per Präprozessorbedingung deaktiviert sind. Für die Spalten dieser Datenbanksichten können keine Bildungsregeln und Formatierungsregeln definiert werden.

Datenbanksichten vom Typ **Proxy** werden vorrangig für die Abbildung des Unified Namespace eingesetzt. So wird beispielsweise die Abbildung der Tabellen `ADSDomain` oder `LDAPDomain` im Unified Namespace die Datenbanksicht `UNSRoot` verwendet.

Für Definition einer Datenbanksichten vom Typ **Proxy** werden folgende Informationen verwendet.

Tabelle 18: Eigenschaften für die Definition einer Datenbanksicht vom Typ Proxy

Eigenschaft	Bedeutung
Tabelle	Bezeichnung der Tabelle im Datenmodell.
Typ	Typ der Tabelle **Proxy**.
Zusätzlich Sicht-definition	Generierte Datenbankabfragen für die Erstellung der Datenbanksicht als Select-Anweisung. Die Erweiterungen zur Sichtdefinition werden durch den DBQueue Prozessor generiert. Bei der Generierung werden berücksichtigt: • Tabellen, in denen die Datenbanksicht als Proxyview eingetragen • Spalten, die einen Verweis auf ein Spalte der Proxyview besitzen • Spalten, die als Erweiterungen zur Proxyview definiert sind. Die Erweiterungen werden intern über den Union-Operator miteinander verknüpft.
Bedingung für Sichtdefinition	Einschränkende Bedingung für die Erstellung der Datenbanksicht als Where-Klausel für Datenbankabfragen.
Spalten	Spalten der Datenbanksicht.

Beispiel:

Damit die Tabelle `ADSDomain` im Unified Namespace auf die Datenbanksicht `UNSRoot` abgebildet werden kann, sind folgende Abbildungen notwendig.

- In der Tabelle `ADSDomain` ist die Datenbanksicht `UNSRoot` als Proxview eingetragen.
- Die Spalten der Tabelle `ADSDomain`, die im Unified Namespace abgebildet werden sollen, erhalten einen Verweis auf die korrespondierende Spalten der Proxyview.

  Beispielsweise wird die Spalte `Ident_Domain` der Tabelle `ADSDomain` auf die Spalte `Ident_UNSRoot` der Proxyview `UNSRoot` abgebildet.
- Spalten die in der Datenbanksicht `UNSRoot` erwartet werden, jedoch nicht in der Tabelle `ADSDomain` vorhanden sind, sind an der Tabelle `ADSDomain` als Erweiterungen zur Proxyview eingetragen.

  Beispielsweise erwartet die Sicht `UNSRoot` in der Spalte `UID_DPRNameSpace` die Angabe des Zielsystemtyps. Diese Spalte ist an den Tabellen `ADSDomain` nicht vorhanden. Daher wird an der Tabelle `ADSDomain` als Erweiterung zur Proxyview `'ADS-DPRNameSpace-ADS' as UID_DPRNameSpace` eingetragen.

Aus den Informationen generiert der DBQueue Prozessor die Erweiterung zur Sichtdefinition. Das nachfolgende Statement zeigt einen Auszug aus der generierten Erweiterung.

```
select ... Ident_Domain as Ident_UNSRoot..., 'ADS-DPRNameSpace-ADS' as UID_
DPRNameSpace from ADSDomain
```

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73
- Eigenschaften von Spaltendefinitionen auf Seite 99
- Datenbanksichten vom Typ View auf Seite 59
- Datenbanksichten vom Typ Union auf Seite 63
- Datenbanksichten vom Typ Read only auf Seite 66

Datenbanksichten vom Typ Union

Datenbanksichten mit dem Tabellentyp **Union** sind Vereinigungssichten verschiedener Tabellen und dienen zur Zusammenfassung verschiedener Objekttypen gleichen Kontextes. So wird beispielsweise in der Vereinigungssicht `QERAccProductUsage` ermittelt, welche Leistungspositionen in welchen IT Shop Produkten verwendet werden.

Datenbanksichten mit dem Typ **Union** sind vordefinierte Sichten. Für die Spalten dieser Datenbanksichten können keine Bildungsregeln und Formatierungsregeln definiert werden. In der Sichtdefinition muss die Objektschlüsselspalte (`XObjectkey`) referenziert werden. Somit ist die Erzeugung eines Einzelobjektes mit den dafür geltenden Berechtigungen möglich.

Datenbanksichten vom Typ **Union** werden hauptsächlich bei der Bearbeitung der Benutzeroberfläche und für die Erstellung von Berichten verwendet.

Für Definition einer Datenbanksichten vom Typ **Union** werden folgende Informationen verwendet.

Tabelle 19: Eigenschaften für die Definition einer Datenbanksicht vom Typ Union

Eigenschaft	Bedeutung
Tabelle	Bezeichnung der Tabelle im Datenmodell.
Typ	Typ der Tabelle **Union**.
Zusätzliche Sichtdefinition	Datenbankabfragen für die Erstellung der Datenbanksicht als Select-Anweisung. HINWEIS: Selektieren Sie niemals `NULL as <Spalte>`. Konvertieren Sie stattdessen explizit in die gewünschte Wertart.

Eigenschaft	Bedeutung

	Beispiele:
	`convert(nvarchar(max), NULL) as <Spalte>`
	`convert(varchar(38), NULL) as UID_<Spalte>`
	`convert(varchar(138), NULL) as ObjectKey<Spalte>`
	Es können mehrere Erweiterungen zur Sichtdefinition definiert werden. Die Erweiterungen werden intern über den Union-Operator miteinander verknüpft.
	Wenn Sie für eine kundenspezifischeTabelle eine Spalte hinzufügen, wird ein Eintrag in der Tabelle `DialogColumn` erzeugt. Wenn Sie eine Spalte löschen, wird der Eintrag in der Tabelle `DialogColumn` entfernt. Schemaänderungen an Standard-Datenbanksichten sind nicht zulässig.

Bedingung für Sichtdefinition	Einschränkende Bedingung für die Erstellung der Datenbanksicht als Where-Klausel für Datenbankabfragen.
Spalten	Spalten der Datenbanksicht.

Beispiel:

Die Tabelle `QERAccProductUsage` ist als Datenbanksicht vom Typ **Union** definiert. In der Vereinigungssicht wird ermittelt, welche Leistungspositionen in welchen Produkten verwendet werden. Nachfolgendes Beispiel zeigt einen Auszug aus der Definition am Beispiel Systemberechtigungen (Tabelle `ESet`) und Berichtabonnements (Tabelle `RPSReport`).

Tabelle 20: Beispiel für Definition einer Datenbanksicht vom Typ Union

Eigenschaft	Wert
Tabelle	QERAccProductUsage
Typ	Union
Spalten	TableName, UID_AccProduct, XObjectKey
Erweiterung 1: Zusätzliche Sichtdefinition	ESet
Erweiterung 1: Abfrage	select 'ESet' as TableName, g.XObjectKey, g.UID_AccProduct from ESet g
Erweiterung 2:	RPSReport

Eigenschaft	Wert
Zusätzliche Sichtdefinition	
Erweiterung 2: Abfrage	select 'RPSReport' as TableName, g.XObjectKey, g.UID_AccProduct from RPSReport g
Resultierende Sichtdefinition	create view dbo.QERAccProductUsage as select * from (select convert(varchar(11), null) as TableName, convert (varchar(38), null) as UID_AccProduct, convert(varchar (138), null) as XObjectKey where 1=0 union all select xxTab.TableName, xxTab.UID_AccProduct, xxTab.XObjectKey from (select 'ESet' as TableName, g.XObjectKey, g.UID_AccProduct from ESet g) as xxTab union all select xxTab.TableName, xxTab.UID_AccProduct, xxTab.XObjectKey from (select 'RPSReport' as TableName, g.XObjectKey, g.UID_AccProduct from RPSReport g) as xxTab) as x

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73
- Eigenschaften von Spaltendefinitionen auf Seite 99
- Datenbanksichten mit dem Typ Union erstellen auf Seite 488
- Datenbanksichten vom Typ View auf Seite 59

- Datenbanksichten vom Typ Proxy auf Seite 61
- Datenbanksichten vom Typ Read only auf Seite 66

Datenbanksichten vom Typ Read only

Datenbanksichten mit dem Tabellentyp **Read only** können sowohl Teilausschnitte als auch Vereinigungsmengen der zugrunde liegenden Tabellen sein. Datenbanksichten mit dem Typ **Read only** sind vordefinierte Datenbanksichten. Für die Spalten dieser Datenbanksichten können keine Bildungsregeln und Formatierungsregeln definiert werden.

Datenbanksichten vom Typ **Read only** dienen nur zur Anzeige und werden hauptsächlich bei der Bearbeitung der Benutzeroberfläche und für die Erstellung von Berichten verwendet.

Für Definition einer Datenbanksichten vom Typ **Read only** werden folgende Informationen verwendet.

Tabelle 21: Eigenschaften für die Definition einer Datenbanksicht vom Typ Read only

Eigenschaft	Bedeutung
Tabelle	Bezeichnung der Tabelle im Datenmodell.
Typ	Typ der Tabelle **Read only**.

Zusätzliche Sichtdefinition	Datenbankabfragen für die Erstellung der Datenbanksicht als Select-Anweisung.
	HINWEIS: Selektieren Sie niemals `NULL as <Spalte>`. Konvertieren Sie stattdessen explizit in die gewünschte Wertart.
	Beispiele:
	`convert(nvarchar(max), NULL) as <Spalte>`
	`convert(varchar(38), NULL) as UID_<Spalte>`
	`convert(varchar(138), NULL) as ObjectKey<Spalte>`
	Es können mehrere Erweiterungen zur Sichtdefinition definiert werden. Die Erweiterungen werden intern über den Union-Operator miteinander verknüpft.
	Wenn Sie für eine kundenspezifischeTabelle eine Spalte hinzufügen, wird ein Eintrag in der Tabelle `DialogColumn` erzeugt. Wenn Sie eine Spalte löschen, wird der Eintrag in der Tabelle `DialogColumn` entfernt. Schemaänderungen an Standard-Datenbanksichten sind nicht zulässig.

Bedingung für Sichtdefinition	Einschränkende Bedingung für die Erstellung der Datenbanksicht als Where-Klausel für Datenbankabfragen. Die Bedingung wird an die aus der Erweiterung generierte Sichtdefinition angehängt.
Spalten	Spalten der Datenbanksicht.

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73
- Eigenschaften von Spaltendefinitionen auf Seite 99
- Datenbanksichten mit dem Typ Read only erstellen auf Seite 486
- Common Table Expressions in Read Only-Datenbanksichten verwenden auf Seite 487
- Datenbanksichten vom Typ View auf Seite 59
- Datenbanksichten vom Typ Proxy auf Seite 61
- Datenbanksichten vom Typ Union auf Seite 63

Tabellenskripte

Mit Hilfe der Tabellenskripte definieren Sie Aktionen, die vor oder nach dem Speichern, Laden oder Verwerfen eines Objektes ausgeführt werden. So können beispielsweise vor dem Speichern eines Objektes umfangreichere Änderungen oder Überprüfungen von Werten, die über Bildungsregeln oder Formatierungsskripte nur schwer realisierbar sind, durch ein Tabellenskript ausgeführt werden. Nach dem Speichern eines Objektes können beispielsweise Änderungen weiterer Objekte oder die Generierung von Methoden und Prozessen über Tabellenskripte ausgeführt werden. Die in den Customizern definierten Nebenwirkungen und Methoden werden nach der Ausführung der Tabellenskripte angewendet.

Sie können die vordefinierten Standard-Tabellenskripte an Ihre Anforderungen anpassen und zusätzliche eigene Tabellenskripte erstellen. Tabellenskripte werden in VB.Net Syntax abgelegt, was die Nutzung aller VB.Net Skriptfunktionen zulässt.

Um Tabellenskripte zu erfassen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie in der Ansicht **Tabelleneigenschaften** den Tabreiter **Tabellenskript** und erstellen Sie die benötigten Skripte.

Tabelle 22: Tabellenskripte

Skript	Beschreibung
Skript (OnDiscarded)	Das Skript wird nach Verwerfen des Objektes ausgeführt.
Skript (OnDiscarding)	Das Skript wird vor dem Verwerfen des Objektes ausgeführt.
Skript (OnLoaded)	Das Skript wird nach Laden des Objektes ausgeführt.

Skript	Beschreibung
Skript (OnSaved)	Das Skript wird nach Speichern des Objektes ausgeführt.
Skript (OnSaving)	Das Skript wird vor dem Speichern des Objektes ausgeführt.

4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

WICHTIG: Damit die Tabellenskripte wirken, kompilieren Sie die Datenbank.

Verwandte Themen

- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Bildungsregeln zur Wertebildung auf Seite 82
- Formatierungsskripte erstellen auf Seite 90

Arbeiten mit Modul Globally Unique Identifier

Um beispielsweise vordefinierte Berichte, Prozesse, Workflows oder Maildefinitionen mit einem kompletten Systemkonfigurationstransport zu transportieren, benötigen diese Objekte einen Primärschlüssel mit einem Modul Globally Unique Identifier (Modul GUID). Durch eine Modul GUID werden diese Objekte als Systemkonfigurationsbestandteil erkannt.

Syntax

Der Primärschlüssel der Tabelle hat das Format `CCC-[0-9,a-f](32)`.

HINWEIS: Einträge mit einer Modul GUID werden beim Erstellen eines Transportes der kompletten Systemkonfiguration automatisch in das Transportpaket übernommen.

Für die Generierung einer Modul GUID sind an der Tabellendefinition folgende Einstellungen möglich:

- Sind die Optionen **Modul GUID zulässig** und **Modul GUID erforderlich** aktiviert, müssen die Objekte eine Modul GUID erhalten. Die Objekte in derart gekennzeichneten Tabellen, erhalten GUIDs mit dem Modulkürzel **CCC** .
- Ist nur die Option **Modul GUID zulässig** aktiviert, können die Objekte eine entsprechend formatierte Modul GUID enthalten. Standardmäßig erhalten die Objekte eine Standard GUID im Format `[0-9,a-f](8-4-4-4-12)`. Sollen die Objekte eine Modul GUID erhalten, erstellen Sie die Objekte beginnend mit dem Modulkürzel **CCC**. Verwenden Sie dazu den Object Browser.

Beispiel:

- Die Tabelle `DialogGroup` ist mit den Optionen **Modul GUID erforderlich** und **Modul GUID zulässig** gekennzeichnet. Beim Erstellen einer neuen Berechtigungsgruppe wird der Primärschlüssel automatisch im Format einer Modul GUID erzeugt.
- Die Tabelle `AERole` ist nur mit der Option **Modul GUID zulässig** gekennzeichnet. Um sicherzustellen, dass Ihre eigenen Anwendungsrollen in das Transportpaket aufgenommen werden, erstellen Sie die Anwendungsrolle im Object Browser mit einer Modul GUID.

HINWEIS:

- Im Standardfall wird der Primärschlüssel der Tabelle mit einer Standard GUID erzeugt. Um nachträglich eine Standard GUID in eine Modul GUID zu ändern, verwenden Sie den Object Browser.
- GUIDs in Tabellen, die in der View `QBM_VHeavyLoadTables` mit `IsNoReload = 1` gekennzeichnet sind, können nicht geändert werden.

WICHTIG: Führen Sie die nachfolgenden Schritte nicht auf produktiven Datenbanken aus. Führen Sie die Schritte nur innerhalb des Wartungsfenster aus. Anderenfalls könnten inkonsistente Daten entstehen.

Um eine Standard GUID in eine Modul GUID zu ändern

1. Wählen Sie im Object Browser das Objekt, dessen Standard GUID Sie ändern möchten.
2. Öffnen Sie das Kontextmenü **Eigenschaften**.
3. Wählen Sie auf dem Tabreiter **Methoden** die Methode **SwitchToModuleGuid()** und klicken Sie **Ausführen**.

Um eine Modul GUID in eine Standard GUID zu ändern

1. Wählen Sie im Object Browser das Objekt, dessen Modul GUID Sie ändern möchten.
2. Öffnen Sie das Kontextmenü **Eigenschaften**.
3. Wählen Sie auf dem Tabreiter **Methoden** die Methode **SwitchToNormalGuid()** und klicken Sie **Ausführen**.

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73

Eindeutige Spalten für Tabellen definieren

Gibt es für eine Tabelle eine Spalte oder eine Kombination von Spalten, die eindeutig sein muss, definieren Sie im Designer mehrspaltige Eindeutigkeiten. Die Spalten werden in einer eindeutigen Gruppe zusammengefasst.

Beispiel:

- Für die Tabelle `Hardware` muss sicher gestellt sein, das die Bezeichnung der Hardware eindeutig ist. Dazu wird für die Tabelle `Hardware` eine eindeutige Gruppe **Hardware** mit der Spalte `Ident_Hardewarelist` erstellt.
- Für die Tabelle `ADSDomain` muss die Kombination aus der Bezeichnung der Domäne und definiertem Namen der Domäne eindeutige Werte liefern. Dazu wird für die Tabelle `ADSDomain` eine eindeutige Gruppe **ADSDomain** mit den Spalten `Ident_Domain` und `DistinguishedName` erstellt.

Um Spalten zu einer eindeutigen Gruppe zusammenzufassen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie in der Ansicht **Tabelleneigenschaften** den Tabreiter **Mehrspaltige Eindeutigkeiten** und klicken Sie .
4. Erfassen Sie die folgenden Informationen.

Tabelle 23: Eigenschaften von Tabellen für eindeutige Gruppen

Eigenschaft	Beschreibung
Eindeutige Gruppe	Bezeichnung der eindeutigen Gruppe von Spalten.
Spalten	Aktivieren Sie die Spalten, die zusammen eindeutig sein müssen.
Fehlermeldung	Meldungstext für Fehlermeldungen, wenn nicht die Standard-fehlermeldung verwendet werden soll. Angepasste Fehler-meldungen werden in der Form ausgegeben: `<Anzeigename (Plural) der Tabelle>: <Fehlermeldung der eindeutigen Gruppe>` In der Fehlermeldung sind optional folgende Platzhalter möglich: • **`{0}`**: Anzeigewert des anderen Objektes, das bereits den selben Wert oder die Wertekombination besitzt.

Eigenschaft	Beschreibung
	• `{1}`: Der kollidierende Wert. Die Syntax der Platzhalter entspricht einem Format-Platzhalter in Visual Basic .NET. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Leerwerte ignorieren	Gibt an, ob Leerwerte in einer eindeutigen Gruppe zulässig sind. Die Option kann nur aktiviert werden, wenn alle Spalten der Gruppe leer sein dürfen. • Ist die Option aktiviert, sind Leerwerte in den beteiligten Spalten zulässig. Ist mindestens eine der beteiligten Spalten nicht leer, wird die Eindeutigkeit geprüft. Sind alle Spalten der Gruppe leer, wird die Eindeutigkeit nicht geprüft. Damit ist es zulässig mehrere Datensätze einzufügen, bei denen alle Spalten der Gruppe leer sind. • Ist die Option nicht gesetzt, sind Leerwerte zwar zulässig, aber pro Spaltenkombination nur einmal erlaubt. Mehrere Datensätze, bei denen alle Spalten der Gruppe leer sind, können nicht eingefügt werden.

5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

TIPP: Um Leerwerte in einer Spalte zu verhindern, definieren Sie die minimale Länge der Spalte an der Spaltendefinition.

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73
- Eigenschaften von Spaltendefinitionen auf Seite 99

Löschverzögerung für Objekte festlegen

Über eine Löschverzögerung legen Sie fest, wie lange die Objekte nach dem Auslösen des Löschens in der Datenbank verbleiben, bevor sie endgültig entfernt werden.

- Ist eine Löschverzögerung **> 0** Tage konfiguriert, wird eine verzögerte Operation für das Löschen erzeugt. Die Objekte werden zunächst deaktiviert. Während dieser Zeit besteht die Möglichkeit, die Objekte wiederherzustellen. Beim Wiederherstellen eines gelöschten Objektes werden die Eigenschaften des Objektes wieder in den Zustand vor dem Löschen zurückgesetzt. Nach Ablauf der Löschverzögerung werden die Objekte endgültig gelöscht.
- Objekte mit einer Löschverzögerung von **0** Tagen, werden sofort gelöscht.

Beispiel:

Die Löschverzögerung wird insbesondere für Zielsysteme angewendet. Standardmäßig werden Benutzerkonten mit einer Löschverzögerung von 30 Tagen endgültig aus der Datenbank entfernt. Die Benutzerkonten werden zunächst deaktiviert oder gesperrt. Bis zum Ablauf der Löschverzögerung besteht die Möglichkeit die Benutzerkonten wieder zu aktivieren. Nach Ablauf der Löschverzögerung werden die Benutzerkonten aus der Datenbank gelöscht und ein Wiederherstellen ist nicht mehr möglich.

Die Löschverzögerung legen Sie an den einzelnen Tabellen fest. Nutzen Sie folgende Tabelleneigenschaften:

- **Löschverzögerung [Tage]**(Standard): Verzögerung der Ausführung von Löschoperationen in Tagen. Ist der Wert **0**, wird sofort gelöscht. Nutzen Sie diese Angabe, wenn alle Objekte einer Tabelle mit einer definierten Löschverzögerung behandelt werden soll.
- **Skript (Löschverzögerung)**: Skript in VB.Net-Syntax zur Ermittlung einer objekt-spezifischen Löschverzögerung. Das Skript überschreibt den Wert aus der Eigenschaft **Löschverzögerung [Tage]**. Nutzen Sie das Skript, um beispielsweise, abhängig von bestimmten Eigenschaften, für einzelne Objekte einer Tabelle unterschiedliche Zeiträume zu definieren.

Beispiel:

Die Löschverzögerung für privilegierte Benutzerkonten soll 10 Tage betragen. An der Tabelle `UNSAccountB` wird folgendes **Skript (Löschverzögerung)** eingetragen.

```
If $IsPrivilegedAccount:Bool$ Then
     Value = 10
End If
```

Verwandte Themen

- Tabellendefinitionen bearbeiten auf Seite 73
- Eigenschaften von Tabellendefinitionen auf Seite 73
- Arbeiten mit Objekten im One Identity Manager auf Seite 22

Tabellendefinitionen bearbeiten

Um Tabelleneigenschaften zu bearbeiten

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Bearbeiten Sie in der Ansicht **Tabelleneigenschaften** die Tabelleneigenschaften.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73

Eigenschaften von Tabellendefinitionen

Tabelle 24: Eigenschaften von Tabellendefinitionen

Eigenschaft	Beschreibung
Tabelle	Bezeichnung der Tabelle im Datenmodell.
Nutzungstyp	Der Nutzungstyp der Tabelle dient als Grundlage für Berichte und die Auswahl von Aufträgen für tägliche Wartungsarbeiten. Zulässige Werte sind: • **Arbeitstabellen**: Die Tabelle ist eine Arbeitstabelle und enthält Bewegungsdaten. • **Historische Bewegungsdaten**: Die Tabelle enthält Bewegungsdaten zur Erstellung von Historien. • **Konfiguration**: Die Tabelle enthält Daten zur Systemkonfiguration. • **Materialisierte Daten**: Die Tabelle enthält materialisierte Daten. Diese werden durch Berechnungen des DBQueue Prozessors wieder erstellt. • **Nur lesbare Daten**: Die Tabelle enthält Daten, die nur zum Lesen vorhanden sind. • **Nutzdaten**: Die Tabelle enthält Nutzdaten.
Anzeigename (Singular)	Anzeigename für einen einzelnen Datensatz der Tabelle. Übersetzen Sie den eingegebenen Text über die Schalt-

Eigenschaft	Beschreibung
	fläche.
Anzeigename (Plural)	Anzeigename der Tabelle. Der Anzeigename wird beispielsweise in der Datenbanksuche oder der Fehlerausgabe zur Anzeige der Tabelle verwendet. Übersetzen Sie den eingegebenen Text über die Schaltfläche.
Anzeigemuster	Mit einem Anzeigemuster legen Sie fest, in welcher Form die Objekte beispielsweise in der Ergebnisliste der Administrationswerkzeuge oder in Berichten dargestellt werden. Übersetzen Sie den eingegebenen Text über die Schaltfläche.Weitere Informationen finden Sie unter Anzeigemuster für die Listendarstellung auf Seite 146. HINWEIS: Für M:N-Tabellen müssen Sie kein Anzeigemuster eintragen. Das Anzeigemuster wird in diesem Fall von der `viDB.DLL` aus den Fremdschlüsseln gebildet.
Anzeigemuster (lang)	Zusätzliches Anzeigemuster für einzelne Tabellen, das den vollständigen Namen des Objekts enthält.
Pfad in der Hierarchie	Für die hierarchische Darstellung von Tabellen, beispielsweise auf den Zuordnungsformularen, tragen Sie in diesem Eingabefeld die Fremdschlüsselspalten ein, über welche die Hierarchie aufgebaut werden soll. Weitere Informationen finden Sie unter Hierarchische Darstellung von Daten auf Zuordnungsformularen auf Seite 165. Beispiel: Ein Active Directory Benutzerkonto (Tabelle `ADSAccount`) wird auf den Zuordnungsformularen typischerweise unterhalb seines Active Directory Containers (Spalte `UID_ADSContainer`) angezeigt. Der Active Directory Container (Tabelle `ADSContainer`) wird wiederum unterhalb seiner Active Directory Domäne (Spalte `UID_ADSDomain`) dargestellt. Für den Aufbau der Hierarchie ist der Pfad in der Hierarchie folgendermaßen eingetragen: Tabelle: `ADSAccount` — Pfad in der Hierarchie: `UID_ADSContainer,UID_ADSDomain` Tabelle: `ADSContainer` — Pfad in der Hierarchie: `UID_ADSDomain` Für Objekte, die nicht in allen Fremdschlüsselspalten Werte enthalten, kann nach Pipe (|) eine alternative Auflistung angegeben werden. Beispiel:

Eigenschaft	Beschreibung
	`(UID_ADSContainer,UID_ADSDomain|UID_ADSDomain)`
Bemerkungen	Freitextfeld für zusätzliche Erläuterungen.
Cache Informationen	Ladeverhalten für die Tabelle im Designer. Diese Angaben sind nur für Systemtabellen erforderlich. Die Cache Informationen für eine Tabelle setzen sich zusammen aus einer Sortierreihenfolge und dem Ladeverhalten. Zulässige Werte sind: • **Basistabelle**: Die Tabelle wird vor dem Laden der Benutzeroberfläche geladen. • **Benutzertabelle**: Die Tabelle wird nur für den angemeldeten Benutzer gefüllt. • **Datentabelle**: Die Tabelle wird im Hintergrund nach dem Laden der Benutzeroberfläche geladen. • **Proxy**: Die Tabelle wird im Designer als Sicht auf die originale Tabelle abgebildet. Die Daten werden geladen, können aber nicht bearbeitet werden. • **BLOBS mit laden**: Spalten mit großen Datenmengen (BLOB-Spalten) werden geladen. • **Nicht cachen**: Die Tabelle wird nicht im Designer geladen.
Deaktiviert durch Präprozessor	Ist eine Tabelle durch eine Präprozessorbedingung deaktiviert, dann wird durch den Database Compiler die Option gesetzt. Weitere Informationen finden Sie unter Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369.
Präprozessorbedingung	Tabellen können Sie mit Präprozessorbedingungen versehen. Damit ist die Tabelle mit ihren Spalten nur verfügbar, wenn die Präprozessorbedingung erfüllt ist. Weitere Informationen finden Sie unter Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369.
Löschverzögerung [Tage]	Verzögerung der Ausführung von Löschoperationen in Tagen. Ist der Wert **0**, wird sofort gelöscht. Weitere Informationen finden Sie unter Löschverzögerung für Objekte festlegen auf Seite 71.
Skript (Löschverzögerung)	Skript in VB.Net-Syntax zur Ermittlung einer objektspezifischen Löschverzögerung. Weitere Informationen finden Sie unter Löschverzögerung für Objekte festlegen auf Seite 71.

Eigenschaft	Beschreibung
Symbol	Symbol zur Darstellung der Tabelle in der Oberfläche der Administrationswerkzeuge.
Hintergrundfarbe	Farbe, mit der das Steuerelement für diese Tabelle in der Schemaübersicht angezeigt wird.
Proxyview	Verweis auf die Datenbanksicht, vom Typ **Proxy**, die Inhalte der Tabelle verwenden soll. Beispiel: Für die Abbildung der Tabellen `ADSDomain` oder `LDAPDomain` im Unified Namespace wird die Datenbanksicht `UNSRoot` verwendet. Weitere Informationen finden Sie unter Datenbanksichten vom Typ Proxy auf Seite 61.
Erweiterungen zur Proxyview	Aufzählung von Spalten als SQL-Text. Diese werden in der Select-Anweisung der Datenbanksicht verwendet, die unter **Proxyview** ausgewählt ist. Die Erweiterungen zur Proxyview verwenden Sie beispielsweise wenn Spalten doppelt abgebildet werden oder wenn zusätzliche Spalten der Proxyview befüllt werden sollen. Beispiel: Die Sicht `UNSRoot` erwartet in der Spalte `UID_DPRNameSpace` die Angabe des Zielsystemtyps. Diese Spalte ist an den Tabellen `ADSDomain` und `LDAPDomain` nicht vorhanden. Die Erweiterung zur Proxyview wird folgendermaßen definiert: Tabelle: `ADSDomain` Erweiterung zur Proxyview: `'ADS-DPRNameSpace-ADS' as UID_DPRNameSpace` Tabelle: `LDPDomain` Erweiterung zur Proxyview: `'LDP-DPRNameSpace-LDAP' as UID_DPRNameSpace` Weitere Informationen finden Sie unter Datenbanksichten vom Typ Proxy auf Seite 61.
Logischer Speicherort	Logischer Speicherort der Tabelle. In logischen Speicherorten werden zusammengehörende Tabellen zusammengefasst. In der Standardinstallation sind für die Tabellen jedes Moduls des One Identity Manager sowie die Systemtabellen logische Speicherorte vordefiniert. Die Zuordnungen können Sie nicht ändern.

Eigenschaft	Beschreibung
	Sie können zur Gruppierung kundenspezifischer Tabellen eigene logische Speicherorte erstellen. Unterstützung von Dateigruppen auf Seite 115
Scope Hierarchie	Komma-getrennte Liste aller Fremdschlüsselspalten, die zur Darstellung des Objekts in der Scope-Hierarchie im Synchronization Editor benötigt werden. Liste aller Spalten, die zu den Tabellen führen, die das übergeordnete Objekt bereitstellen können.
Pfad zum Person-Objekt für tabellenübergreifende Suche	Pfad zum `Person`-Objekt, um die Person innerhalb der tabellenübergreifenden Suche für Benutzerkonten oder E-Mail-Adressen zu ermitteln. Die ermittelten Daten werden in `QBMSplittedLookup.SplittedElement` abgebildet. Ist der eingetragene Wert **no**, ist keine Person ermittelbar, beispielsweise für Gruppen oder `BaseTree`-Ableitungen. Beispiel: Für Exchange Online Postfächer (Tabelle `O3EMailbox`) wird die Person über die Azure Active Directory Benutzerkonten ermittelt. Der Pfad zum Person-Objekt wird folgendermaßen eingetragen: `FK(UID_AADUser).UID_Person`
Export für SPML-Schema	Diese Angabe bestimmt, ob die Tabelle für das SPML-Schema exportiert werden soll. Weitere Informationen finden Sie unter Vorbereiten des One Identity Manager Schemas für den Export in das SPML-Schema auf Seite 542.
M:N-Tabelle	Kennzeichnung von Zuordnungstabellen (M:N-Tabellen). Zuordnungstabellen sind Tabellen, in der Beziehungen zwischen zwei anderen Tabellen hergestellt werden. Weitere Informationen finden Sie unter Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52.
M:all-Tabelle	Kennzeichnung von Zuordnungstabellen, bei denen einer der Partner ein dynamischer Fremdschlüssel ist. Weitere Informationen finden Sie unter Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52.
Kein DB Transport	Tabellen, die mit dieser Option gekennzeichnet sind, können nicht in ein Kundenkonfigurationspaket ausgenommen werden. Diese Tabellen werden von Datentransport ausgeschlossen.

Eigenschaft	Beschreibung
Zuweisung per Ereignis	Angabe, wie Zuweisungen und Löschungen in Tabellen behandelt werden. Diese Option ist nur für Zuordnungstabellen (M:N-Tabellen) im Anwendungsdatenmodell wirksam. • Ist die Option nicht gesetzt, so werden Zuweisungen und Löschungen direkt durch den DBQueue Prozessor vorgenommen. • Ist die Option gesetzt, werden Aufträge für die Prozesskomponente `HandleObjectComponent` in die Jobqueue eingestellt, die dann die entsprechenden Operationen vornehmen. Damit ist es möglich, spezifische Prozesse direkt an die Ereignisse `Assign` und `Remove` zu knüpfen. Dieses Verhalten müssen Sie kundenspezifisch implementieren.
Im Speicher vorhalten	Gibt an, ob der Tabelleninhalt an der Datenbankverbindung gepuffert werden darf. Der Schwellwert wird im Konfigurationsparameter **Common | ResidentTableLimit** definiert.
Modul GUID zulässig	Gibt an, ob für Objekte ein Primärschlüssel mit einem Modul Globally Unique Identifier (Modul GUID) zulässig ist. Weitere Informationen finden Sie unter Arbeiten mit Modul Globally Unique Identifier auf Seite 68.
Modul GUID erforderlich	Gibt an, ob für Objekte ein Primärschlüssel mit einem Modul Globally Unique Identifier (Modul GUID) erforderlich ist. Weitere Informationen finden Sie unter Arbeiten mit Modul Globally Unique Identifier auf Seite 68.
Typ	Typ der Tabelle. Weitere Informationen finden Sie unter Tabellentypen im One Identity Manager auf Seite 58.
Basistabelle	Nur für Datenbanksichten: Verweis auf die Basistabelle, auf der eine Datenbanksicht basiert.
Bedingung für Sichtdefinition	Nur für Datenbanksichten: Einschränkende Bedingung für die Erstellung der Datenbanksicht als Where-Klausel für Datenbankabfragen.
Einzufügende Werte	Standardeinstellungen für eine Spalte fest, die beim Einfügen eines neuen Datensatzes zugewiesen werden. Die Angaben erfolgt in VB.Net-Syntax.
Auswahlskript	Nur für Datenbanksichten: Auswahlskript als VB.Net-Ausdruck, um zur Laufzeit zu bestimmen, ob das übergebene Objekt zu dieser Datenbanksicht gehört.

Eigenschaft	Beschreibung
Skript (OnLoaded)	Skript in VB.Net-Syntax, das nach Laden des Objektes ausgeführt wird. Weitere Informationen finden Sie unter Tabellenskripte auf Seite 67.
Skript (OnSaving)	Skript in VB.Net-Syntax, vor dem Speichern des Objektes ausgeführt wird. Weitere Informationen finden Sie unter Tabellenskripte auf Seite 67.
Skript (OnSaved)	Skript in VB.Net-Syntax, nach Speichern des Objektes ausgeführt wird. Weitere Informationen finden Sie unter Tabellenskripte auf Seite 67.
Skript (OnDiscarding)	Skript in VB.Net-Syntax, das vor dem Verwerfen des Objektes ausgeführt wird. Weitere Informationen finden Sie unter Tabellenskripte auf Seite 67.
Skript (OnDiscarded)	Skript in VB.Net-Syntax, das nach dem Verwerfen des Objektes ausgeführt wird. Weitere Informationen finden Sie unter Tabellenskripte auf Seite 67.
Anzahl der Zeilen	Anzahl der Zeilen in der Tabelle. Die Anzahl der Zeilen in der Tabelle wird einmal täglich in den Wartungsaufträgen ermittelt. Das Datenmaterial kann bei der Planung von Kapazitäten und Wartungsarbeiten an der Datenbank helfen.
Basissatzlänge	Maximale Länge des Datensatzes innerhalb des (clustered) Haupindexes. Für LOBs ist darin nur der Verweis gespeichert, die Lob-Inhalte selber liegen im HEAP-Speicher. Die Basissatzlänge wird einmal täglich in den Wartungsaufträgen ermittelt. Das Datenmaterial kann bei der Planung von Kapazitäten und Wartungsarbeiten an der Datenbank helfen.
Größe der Tabelle	Größe der Tabelle in der Datenbank in MB. Die Größe der Tabelle in der Datenbank wird einmal täglich in den Wartungsaufträgen ermittelt. Das Datenmaterial kann bei der Planung von Kapazitäten und Wartungsarbeiten an der Datenbank helfen.
Bedingung für Transport	Bedingung zur Auswahl der transportierbaren Objekte. Eine leere Bedingung bedeutet, alle Objekte werden übertragen.
Darstellungsinformationen	(Nur für interne Verwendung) Informationen zur Darstellung im Designer.
Primärschlüssel 1	(Nur für interne Verwendung) Name der ersten Primärschlüsselspalte der Tabelle. Die Reihenfolge von **Primärschlüssel 1** und **Primärschlüssel 2** entspricht

Eigenschaft	Beschreibung
	der physischen Reihenfolge im Schema.
Primärschlüssel 2	(Nur für interne Verwendung) Name der zweiten Primärschlüsselspalte der Tabelle. Die Reihenfolge von **Primärschlüssel 1** und **Primärschlüssel 2** entspricht der physischen Reihenfolge im Schema.
Synchronisationsmodus	Zulässige Synchronisationsrichtungen und Verarbeitungsmethoden für diese Tabelle, wenn die Synchronisation von zwei One Identity Manager Datenbanken automatisch eingerichtet wird.
Spalten für alternative Regeln	Komma-getrennte Liste der Spalten, die für die Erstellung alternativer Object-Matching-Regeln in einem automatisch erstellten Synchronisationsprojekt verwendet werden sollen. Wenn der One Identity Manager Konnektor kein Systemobjekt durch die primäre Object-Matching-Regel identifizieren kann, wendet er die alternativen Regeln an, um ein passendes Systemobjekt zu ermitteln. Ausführliche Informationen dazu finden Sie im *One Identity Manager Anwenderhandbuch für den One Identity Manager Konnektor*.
CLR Typ für Projektgenerator	.NET-Klasse, die zur Berücksichtigung von Sonderfällen bei der Generierung eines Synchronisationsprojekts zwischen zwei One Identity Manager Datenbanken dient.

Verwandte Themen

- Tabellendefinitionen bearbeiten auf Seite 73

Customizer für Tabellendefinitionen anzeigen

Customizer führen Verarbeitungslogiken aus, die üblicherweise im Objektcode selbst implementiert würden, wie beispielsweise den gegenseitigen Ausschluss von Eigenschaften. Customizer enthalten spezielle Methoden und Nebenwirkungen an den Spalten der Tabelle. Für eine Tabelle können mehrere Customizer definiert sein.

In der One Identity Manager Standardinstallation sind verschiedene Customizer enthalten, die ein definiertes Verhalten bereitstellen.

Um die Customizer für eine Tabellendefinition anzuzeigen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie in der Ansicht **Tabelleneigenschaften** den Tabreiter **Customizer**.

Verwandte Themen

- Spaltenabhängigkeiten zur Wertebestückung auf Seite 91

Spaltendefinitionen

Die Spaltendefinitionen aller Tabellen des Anwendungsmodells und des Systemdatenmodells sind in der Tabelle `DialogColumn` hinterlegt. Die vordefinierten Spalteneigenschaften des One Identity Manager Schemas werden durch die Schemainstallation gepflegt und sind bis auf einige Ausnahmen nicht bearbeitbar.

Die Spaltendefinitionen des One Identity Manager Schemas bearbeiten Sie im Designer im Schemaeditor.

Detaillierte Informationen zum Thema

- Hinweise zur Bearbeitung von Tabellendefinitionen und Spaltendefinitionen auf Seite 56
- Bildungsregeln zur Wertebildung auf Seite 82
- Eindeutige Spalten für Tabellen definieren auf Seite 70
- Dynamische Fremdschlüssel
- Spaltenlängen begrenzen auf Seite 87
- Kommastellen für die Anzeige von Werten festlegen auf Seite 87
- Vordefinierte Formatierungstypen verwenden auf Seite 88
- Formatierungsskripte erstellen auf Seite 90
- Spaltenabhängigkeiten zur Wertebestückung auf Seite 91
- Zulässige Werte von Spalten auf Seite 92
- Anforderungen an MVP-Spalten festlegen auf Seite 93
- Bitmasken definieren auf Seite 94
- Konfigurieren von Spalten für die Volltextsuche auf Seite 95
- Skripte zum bedingten Anzeigen und Bearbeiten von Spalten auf Seite 97
- Spalten zur Übersetzung kennzeichnen auf Seite 235

- Spaltendefinitionen bearbeiten auf Seite 99
- Eigenschaften von Spaltendefinitionen auf Seite 99

Bildungsregeln zur Wertebildung

Bildungsregeln werden im One Identity Manager zur Generierung von Nutzdaten oder zur Transformation von Werten eingesetzt. Über Bildungsregeln ist es möglich, Eigenschaften von Objekten mit Standardwerten zu belegen oder den Wert einer Eigenschaft aus anderen Eigenschaften zu bilden. Bildungsregeln können sowohl innerhalb eines Objektes als auch objektübergreifend wirken. Bildungsregeln ohne Abhängigkeiten wirken, wenn der Wert der Spalte abgefragt wird und in der Spalte noch kein Wert eingetragen ist. Bildungsregeln, die sich auf andere Spalten beziehen, wirken, wenn sich diese Spalten ändern.

Bildungsregeln wirken ohne Berücksichtigung der aktuellen Berechtigungen, es ist keine explizite Berechtigungsvergabe auf die abhängigen Spalten notwendig. Bei der Anwendung von Bildungsregeln werden angesprochene Spalten eines Objektes also auch gefüllt, wenn sie nicht auf dem aktuellen Formular im Programm Manager sichtbar sind.

Die Abhängigkeiten von Spalten aufgrund von Bildungsregeln werden in der Tabelle `DialogNotification` abgebildet. In dieser Tabelle werden die verbundenen Eigenschaften als Sender-Empfänger-Paar abgebildet. Die Spalte, die eine Änderung auslöst, ist dabei der Sender, die Spalte, die daraufhin geändert wird, der Empfänger. Die Verknüpfung der betroffenen Objekte erfolgt unter Einbeziehung der Spaltenbeziehungen. Die Einträge werden bei der Kompilierung der Bildungsregeln erzeugt und aktualisiert.

> HINWEIS: Einen Überblick über die im System vorhandenen Spalten mit Bildungsregeln erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Bildungsregeln**. Die Abhängigkeiten von Spalten aufgrund von Bildungsregeln werden in der Schemaübersicht im Schemaeditor dargestellt.

Detaillierte Informationen zum Thema

- Bildungsregeln bearbeiten auf Seite 83
- Ändern einer Spalte verhindern auf Seite 84
- Ausführung von Bildungsregeln limitieren auf Seite 85
- Beispiel für lokale Bildungsregeln innerhalb eines Objektes auf Seite 86
- Beispiel für objektübergreifende Bildungsregeln auf Seite 86
- Spaltenabhängigkeiten aufgrund von Bildungsregeln anzeigen auf Seite 51

Verwandte Themen

- Eindeutige Spalten für Tabellen definieren auf Seite 70
- Zulässige Werte von Spalten auf Seite 92
- Spaltenabhängigkeiten zur Wertebestückung auf Seite 91

Bildungsregeln bearbeiten

Sie können die vordefinierten Standardbildungsregeln an Ihre Anforderungen anpassen und zusätzliche eigene Bildungsregeln erstellen.

WICHTIG: Bei der Definition von Bildungsregeln sollten Sie Performance-Überlegungen mit in Betracht ziehen. Unter Umständen kann es bei einer überschreibenden Bildungsregel durch Änderung einer Eigenschaft dazu kommen, dass eine große Anzahl von abhängigen Objekten geändert, gespeichert und eventuell Prozesse generiert werden.

Um die Anzahl der direkt durch eine Bildungsregel geänderten Objekte zu limitieren, können Sie Schwellwerte für die Ausführung der Bildungsregel festlegen. Weitere Informationen finden Sie unter Ausführung von Bildungsregeln limitieren auf Seite 85.

Um eine Bildungsregel zu erstellen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Wertbildung** und bearbeiten Sie die folgenden Eigenschaften.

Tabelle 25: Eigenschaften zur Wertebildung einer Spalte

Eigenschaft	Beschreibung
Überschreibend	Gibt an, ob die Bildungsregel überschreibend oder nicht überschreibend wirken soll. Ist die Option aktiviert, wird die Bildungsregel immer angewendet. Ist die Option deaktiviert, wird die Bildungsregel nur angewendet, wenn in der Spalte noch kein Wert eingetragen ist. HINWEIS: Das One Identity Manager-Schema kennt für Spalten von Datentyp `Bool` nur die Werte **0** und **1**. Der Wert **0** ist gleichzusetzen mit **leer**. Das heißt, wenn die Option **Überschreibend** deaktiviert ist, wird beim Ändern des Spaltenwertes von **0** auf **1** die Bildungsregel ausgeführt.
Bildungsregel	Bildungsskript der Bildungsregel. Das Bildungsskript wird in VB.Net-Syntax abgelegt, was die Nutzung aller VB.Net-Skript-funktionen zulässt. TIPP: Um die Spalten anzuzeigen, die eine Bildungsregel auslösen, klicken Sie auf den Link **Auslöser der Bildungs-regel**.
Kein automa-tische Einkürzen	Gibt an, ob der Wert einer Spalte eingekürzt werden soll, wenn bei der Verarbeitung der Bildungsregel die maximale Länge der

Eigenschaft	Beschreibung
durch Bildungsregel	Spalte überschritten wird. Ist die Option aktiviert, erfolgt keine automatische Einkürzung des Wertes auf die maximale Spaltenlänge.

5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

WICHTIG: Damit die Bildungsregel wirkt, kompilieren Sie die Datenbank.

TIPP: Testen Sie die Kompilierung im Designer über das Menü **Schema > Kompiliertest**.

Verwandte Themen

- Ändern einer Spalte verhindern auf Seite 84
- Beispiel für lokale Bildungsregeln innerhalb eines Objektes auf Seite 86
- Beispiel für objektübergreifende Bildungsregeln auf Seite 86
- Präprozessorbedingungen in VB.Net-Ausdrücken auf Seite 372
- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Eigenschaften von Spaltendefinitionen auf Seite 99

Ändern einer Spalte verhindern

Sie können durch eine Bildungsregel verhindern, dass der Anwender eine Spalte ändert, die durch eine Bildungsregel gefüllt wird. Ergänzen Sie dazu in der Bildungsregel einen Kommentar mit dem Spaltennamen dieser Spalte in $-Notation. Die Bildungsregel verweist damit auf sich selbst. Eine Änderung in der Spalte wird sofort durch die Bildungsregel überschrieben. Sich selbst überschreibende Bildungsregeln wirken nur, wenn diese Bildungsregeln als überschreibend gekennzeichnet sind.

Beispiel:

Der Anwender soll das zentrale Benutzerkonto einer Person nicht ändern können. Das soll durch die Bildungsregel verhindert werden.

- Definieren Sie eine kundenspezifische Bildungsregel an der Spalte `Person.CentralAccount`.
- Aktivieren Sie für die Bildungsregel die Option **Überschreibend**.

- Ergänzen Sie die Standardbildungsregel um folgenden Eintrag: ’$CentralAccount$.

```
’$CentralAccount$
If Not CBool(Session.Variables.Get("FULLSYNC")) Then
     Value=VI_AE_BuildCentralAccount(GetValue("UID_
     Person").String,$Lastname$, $Firstname$)
End If
```

Ausführung von Bildungsregeln limitieren

Um die Anzahl der direkt durch eine Bildungsregel geänderten Objekte zu limitieren, können Sie Schwellwerte definieren.

Um Schwellwerte für eine Bildungsregel festzulegen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Wertbildung** und bearbeiten Sie die folgenden Eigenschaften.
   - **Schwellwert (asynchron)**: Erfassen Sie die maximale Anzahl von Objekten, die direkt durch die Bildungsregel geändert werden dürfen. Wird die angegebene Anzahl erreicht, erfolgt die Verarbeitung asynchron über den One Identity Manager Service.
   - **Schwellwert (Abbruch)**: Erfassen Sie die Anzahl an Objekten, bei denen ein Abbruch erfolgen soll. Bei Erreichen der angegebenen Anzahl wird die Verarbeitung mit einer Fehlermeldung abgebrochen.

   HINWEIS: Ist ein Schwellwert für den Abbruch angegeben, muss er größer als der Schwellwert für die asynchrone Verarbeitung sein.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Bildungsregeln bearbeiten auf Seite 83
- Eigenschaften von Spaltendefinitionen auf Seite 99

Beispiel für lokale Bildungsregeln innerhalb eines Objektes

Der vollständige Name (`Person.Internalname`) einer Person soll aus dem Nachnamen (`Person.Lastname`) und dem Vornamen (`Person.Firstname`) der Person gebildet werden. Als Bildungsregel für die Spalte `Person.Internalname` wird angegeben:

```
Value = $Lastname$ & ", " & $Firstname$
```

Ist die Bildungsregel als überschreibend gekennzeichnet, wird bei jeder Änderung des `Lastname` einer Person geprüft, ob in abhängigen Spalten Bezug auf diesen Wert in Form einer Bildungsregel genommen wird. Trifft dieses zu, wird das Bildungsskript abgearbeitet und der konkrete Wert in die Spalte `Internalname` eingetragen. Ist die Bildungsregel nicht als überschreibend gekennzeichnet, wird diese nur wirksam, wenn in der Spalte `Internalname` noch kein Wert eingetragen ist.

Die Spalten `Person.Lastname` und `Person.Firstname` sind die Sender und die Spalte `Person.Internalname` ist der Empfänger. Die Abbildung zum Einfügen eines Datenbankobjektes in der Tabelle `DialogNotification` lautet:

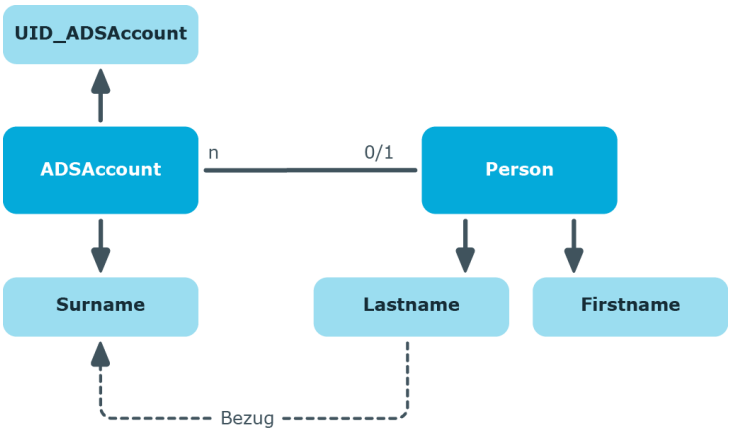
```
Person.Lastname --> Person.Internalname
```

```
Person.Firstname --> Person.Internalname
```

Beispiel für objektübergreifende Bildungsregeln

Bezieht sich eine Bildungsregel auf einen Wert eines anderen Objektes, so kann über die Fremdschlüssel (FK)-Beziehung zugegriffen werden.

Abbildung 9: Wirkungsweise von objektübergreifenden Bildungsregeln



Soll beispielsweise der Nachname eines Active Directory Benutzerkontos (`ADSAccount.Surname`) aus dem Nachnamen einer Person (`Person.Lastname`) gebildet werden, so ist die Bildungsregel für die Spalte `ADSAccount.Surname` folgendermaßen anzugeben:

```
Value = $FK(UID_Person),Person.Lastname$
```

Erfolgt eine Änderung des Nachnamens der Person, wird der Nachname des Active Directory Benutzers ebenfalls geändert. Die Spalte `Person.Lastname` ist somit der Sender und die Spalte `ADSAccount.Surname` ist der Empfänger. Die Abbildung der Beziehung in der Tabelle `DialogNotification` lautet:

```
Person.Lastname --> ADSAccount.Surname
```

Spaltenlängen begrenzen

Über die Spaltendefinition können Sie die Länge der einzugebenden Werte beeinflussen. So ist beispielsweise der Anmeldename eines Active Directory Benutzerkontos auf maximal 20 Zeichen begrenzt. Welche Spalten Pflichtfelder sind, legen Sie ebenfalls über die Spaltendefinition fest.

Um die Länge einer Spalte festzulegen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Wertbildung** und bearbeiten Sie die folgenden Eigenschaften.
   - **Max. Länge**: Erfassen Sie die maximale Länge der Spalte. Ist der Wert gleich **0**, wird die Länge aus dem Datenbankschema benutzt.
   - **Min. Länge**: Erfassen Sie die minimale Länge der Spalte. Spalten mit einer minimalen Länge von **1** oder größer werden in den Frontends als Pflichtfelder markiert.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Kommastellen für die Anzeige von Werten festlegen

Für Spalten mit den .Net-Datentypen `Double`, `Decimal`, `Int`, `Long` oder `Short` können Sie die Anzahl der Kommastellen für die Anzeige der Werte in den Frontends festlegen.

Bei Spalten mit den .Net-Datentypen `Int`, `Long` oder `Short` bewirkt die Angabe der Kommastellen eine Verschiebung des Kommas für die Anzeige der Werte. Bei Spalten mit den .Net-Datentypen `Double` oder `Decimal` wird der Wert mit der entsprechenden Anzahl an

Nachkommastellen angezeigt. Beachten Sie dieses Verhalten insbesondere bei Berechnungen mit unterschiedlichen Datentypen.

Beispiel:

- Preisangaben mit .Net-Datentyp `Int` werden inklusive Cent-Angabe nach dem Komma erfasst. Im Frontend wird eine Preisangabe von 3,50 € mit dem Wert **3,50** angezeigt, in der Datenbank ist der Wert **350** gespeichert.
- Eine Mengenangabe mit .Net-Datentyp `Double` erfolgt beispielsweise mit drei Stellen nach dem Komma. Im Frontend wird eine Mengenangabe von 100 Stück mit dem Wert **100,000** angezeigt, in der Datenbank ist der Wert **100** gespeichert.

Um die Anzahl der Kommastellen festzulegen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Spalte** und erfassen Sie im Eingabefeld **Anzahl Kommastellen** die Anzahl der zu verwendenden Kommastellen für die Anzeige.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Vordefinierte Formatierungstypen verwenden

Legen Sie das Format einer Spalte anhand vordefinierter Formatierungstypen fest. Die Formatierungstypen können mit einander kombiniert werden um eine gewünschte Formatierung zu erhalten.

HINWEIS: Gibt es für eine Tabelle eine Spalte oder eine Kombination von Spalten, die eindeutig sein muss, definieren Sie im Designer mehrspaltige Eindeutigkeiten. Weitere Informationen finden Sie unter Eindeutige Spalten für Tabellen definieren auf Seite 70.

Um Formatierungstypen festzulegen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Wertbildung** und legen Sie im Eingabefeld **Spaltenformat** die Formatierungstypen fest.

Tabelle 26: Zulässige Formatierungstypen

Wert	Formatierungstyp	Zulässige Werte
0	Kein	Keine spezielle Formatierung = Standard
1	IP Adresse	IP-Adresse [0-9]³.[0-9]³. [0-9]³.[0-9]³
2	MAC-ID	MACID [0-9,A-F]12
4	Laufwerksbuchstabe	Laufwerksbuchstabe [A-Z]1:
8	Zahl	[0-9]+
16	Großbuchstaben	Großbuchstaben
32	Großbuchstaben serverabhängig	(nur aus Kompatibilitätsgründen mitgeführt)
64	NT Name	Alle Zeichen zulässig außer „!@/\:,"[];|=+*?<>"
128	Telefonnummer	Telefonnummer [0123456789#/-+*]n
256	Exchangename	Alle Zeichen zulässig außer „ÄÖÜäöüß"!§$%&|/<>#*{}[] ²³~^,"
512	ASCII-Buchstaben und Zahlen	Alle Zeichen des ASCII-Zeichensatzes
2048	Uri	Uniform Ressource Identifier
4096	E-Mail-Adresse	Gültige E-Mail-Adresse
8192	XSS-Zeichen verhindern	Bestimmte Zeichen, die für Webseitenübergreifendes Skripting (XSS) verwendet werden könnten, sind unzulässig. Geprüft werden die Zeichen **<** und **>**. Die Eingabeprüfung erfolgt nur, wenn der Konfigurationsparameter **QBM | XssCheck** aktiviert ist. Wenn der Konfigurationsparameter **QBM | XssCheck | Sync** aktiviert ist, erfolgt die Prüfung auch während der Synchronisation.

5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

WICHTIG: Damit der Formatierungstyp wirkt, kompilieren Sie die Datenbank.

Verwandte Themen

- Formatierungsskripte erstellen auf Seite 90
- Eigenschaften von Spaltendefinitionen auf Seite 99

Formatierungsskripte erstellen

Sie können zur Wertüberprüfungen einer Spalte ein Formatierungsskript verwenden. Im Gegensatz zu Bildungsregeln werden Formatierungsskripte nur bei Wertzuweisung an die jeweilige Spalte ausgeführt.

Um ein Formatierungsskript zu erstellen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Wertbildung** und erfassen Sie im Eingabefeld **Formatierungsskript** das Formatierungsskript für die Spalte.

   Ein Formatierungsskript zur Wertüberprüfung wird in VB.Net Syntax abgelegt, was die Nutzung aller VB.Net Skript-Funktionen zulässt.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

WICHTIG: Damit das Formatierungsskript wirkt, kompilieren Sie die Datenbank.

TIPP: Testen Sie die Kompilierung über das Menü **Schema > Kompiliertest**.

Beispiel:

Der Wert der Spalte `Mail` in der Tabelle `ADSAccount` soll dem SMPT-Format entsprechen. Ist dies nicht der Fall, soll eine Fehlermeldung ausgegeben werden. Das Formatierungsskript für die Spalte `ADSAccount.Mail` könnte folgendermaßen formuliert werden:

```
Dim str as String = Convert.ToString(Value)

If str.Length > 0 Then

     If Not VID_IsSMTPAddress(str) Then
```

```
            Throw New Exception("""" & str & """ ist keine gültige SMTP-
           Adresse.")
      End If
End If
```

Verwandte Themen

- Vordefinierte Formatierungstypen verwenden auf Seite 88
- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Eigenschaften von Spaltendefinitionen auf Seite 99

Spaltenabhängigkeiten zur Wertebestückung

Zwischen den einzelnen Werten eines Objektes bestehen, beispielsweise durch Bildungsregeln oder Customizer, Abhängigkeiten, die eine bestimmte Reihenfolge in der Wertebestückung voraussetzen. In den One Identity Manager-Werkzeugen wird die richtige Bestückungsreihenfolge durch die Sperrung oder die Freischaltung der Eingabefelder erzwungen. Für Datenimporte sowie bei Nutzung der SPML- und Webservice-Schnittstellen muss die richtige Bestückungsreihenfolge sichergestellt werden.

Folgende Datenquellen gehen in genannter Reihenfolge in die Bestimmung der Bestückungsreihenfolge ein:

1. Customizer

   In den Customizern sind die Abhängigkeiten zwischen Spalten eines Objektes hinterlegt.

2. Kundenspezifisch definierte Abhängigkeiten

   Um kundenspezifische Abhängigkeiten zwischen Spalten zu definieren

   a. Wählen Sie im Schemaeditor eine Tabellenspalte aus.

   b. Legen Sie in der Ansicht **Spalteneigenschaften** auf dem Tabreiter **Abhängigkeiten** die Vorgänger dieser Spalte fest.

3. Abhängigkeiten von Spalten aufgrund von Bildungsregeln

   Hierbei werden Werte, die eine Bildungsregel auslösen (beispielsweise `Person.Firstname`, `Person.Lastname`), vor den Werten bestückt, die durch eine Bildungsregel gebildet werden (beispielsweise `Person.CentralAccount`).

Treten bei der Ermittlung der Bestückungsreihenfolge zirkuläre Abhängigkeiten auf, werden diese an der Stelle mit der niedrigsten Priorität aufgebrochen.

Verwandte Themen

- Customizer für Tabellendefinitionen anzeigen auf Seite 80
- Bildungsregeln zur Wertebildung auf Seite 82

Zulässige Werte von Spalten

Um für eine Spalte nur bestimmte Werte zu erlauben, definieren Sie eine Liste mit den zulässigen Werten. Bei der Bildung des Anzeigewertes für eine Spalte wird die Liste der zulässigen Werte aufgelöst. Für einige Spalten des One Identity Manager Schemas werden bei der Schemainstallation bereits zulässige Werte mitgeliefert.

HINWEIS: Eine Liste mit zulässigen Werten für eine Spalte können Sie nur erfassen oder erweitern, wenn für die Spalte die Option **Anpassung der Liste zulässiger Werte nicht erlaubt** nicht gesetzt ist.

Um eine Liste zulässiger Werte zu erstellen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Werteigenschaften** und aktivieren Sie die Option **Werteliste definiert**.
5. Klicken Sie und erfassen Sie die folgenden Eigenschaften.
   - **Wert**: Wert für die Werteliste (Technische Bezeichnung).
   - **Anzeigename**: Bezeichnung des Wertes zur Anzeige. Übersetzen Sie den eingegebenen Text über die Schaltfläche **...**.
   - **Reihenfolge**: Reihenfolge für die Anzeige in der Werteliste.
   - **Deaktiviert**: Gibt an, ob der Wert deaktiviert ist.

   HINWEIS: Um einen Wert zu bearbeiten, wählen Sie den Wert und klicken Sie . Um einen Wert zu löschen, wählen Sie den Wert und klicken Sie . Um alle Werte zu löschen, klicken Sie .
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

WICHTIG: Damit die Liste zulässiger Werte wirkt, kompilieren Sie die Datenbank.

Beispiel:

Im Eingabefeld **Freies Feld Nr. 01** einer Person sollen die Werte **intern** und **extern** zulässig sein. Die Liste der zulässigen Werte wird wie folgt definiert:

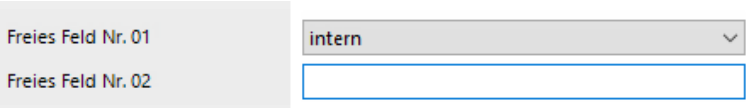
`1=intern 2=Extern`

Für eine Person mit dem Wert **1** wird auf den Formularen im Manager der Anzeigewert **intern** dargestellt.

Darstellung von Spalten mit zulässigen Werten im Manager

Für die Darstellung von Spalten, für die eine Liste an zulässigen Werten definiert ist, wird im Manager ein spezielles Steuerelement verwendet. Ist keine Liste definiert, wird das Steuerelement als einfaches Eingabefeld dargestellt. Ist eine Liste definiert, wird das Steuerelement als Auswahlliste angezeigt.

Abbildung 10: Eingabefeld für Liste definierte Werte (mit und ohne definierte Liste)



Das Steuerelement wird auf den Standardformularen nur für die vordefinierten Spalten sowie die kundenspezifischen Spalten (in der Regel `CustomProperty01-CustomProperty10`) angeboten.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Anforderungen an MVP-Spalten festlegen

Die Werte in Multi-Valued-Property (MVP)-Spalten werden durch **char(7)** oder **chr(7)** getrennt. Spezifizieren Sie weitere Anforderungen an die Einzelwerte der MVP-Spalte.

Um die Anforderungen an eine MVP-Spalte festzulegen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Werteeigenschaften** und aktivieren Sie die Option **MVP-Spalte**.

5. In der Liste **Multi-Value Spezifizierung** legen Sie folgende Einstellungen fest.
    - **Eindeutig**: Wenn die Werte eindeutig sein müssen, aktivieren Sie die Option.
    - **Groß- und Kleinschreibung beachten**: Wenn bei der Prüfung der Werte die Groß-und Kleinschreibung beachtet werden soll, aktivieren Sie die Option.
    - **Akzent nicht berücksichtigen**: Wenn Akzentzeichen bei der Prüfung der Werte nicht berücksichtigt werden sollen, aktivieren Sie die Option.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Bitmasken definieren

Bitmasken können Sie nur für Spalten mit dem Datentyp **Int** definieren.

HINWEIS: Eine Bitmaske für eine Spalte können Sie nur erfassen oder erweitern, wenn für die Spalte die Option **Anpassung der Bitmaske nicht erlaubt** nicht gesetzt ist.

Um eine Bitmaske zu erstellen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Werteigenschaften** und aktivieren Sie die Option **Definierte Bitmaske**.
5. Wählen Sie unter **Sortierkriterium der Bitmaske** das Sortierungskriterium für die Anzeige der Werte.

   Zur Auswahl stehen:
    - **Bit-Position**: Die Sortierung erfolgt nach der Position.
    - **Anzeigewert:** Die Sortierung erfolgt nach dem Anzeigewert.
6. Klicken Sie und erfassen Sie die folgenden Eigenschaften.
    - **Bit-Position**: Die einzelne Bit-Position. Das erste Bit in der Definition startet mit dem Index **0**.
    - **Anzeigename**: Bezeichnung des Wertes zur Anzeige. Übersetzen Sie den eingegebenen Text über die Schaltfläche **...**.
    - **Deaktiviert**: Gibt an, ob der Wert deaktiviert ist.

    HINWEIS: Um einen Wert zu bearbeiten, wählen Sie den Wert und klicken Sie . Um einen Wert zu löschen, wählen Sie den Wert und klicken Sie . Um alle Werte

zu löschen, klicken Sie .

7. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Konfigurieren von Spalten für die Volltextsuche

Die Volltextsuche verwendet einen externen Suchindex, der als Ergebnisse Objektschlüssel zurück gibt. Mit den Objektschlüsseln wird eine Suchanfrage in der Datenbank ausgeführt. Die Suchanfrage in der Datenbank erfolgt unter Berücksichtigung der Berechtigungen das angemeldeten Benutzers. Durch den Suchindex werden maximal 1000 Objekte zurück geliefert.

Die Volltextsuche des One Identity Manager kann im Web Portal und im Manager genutzt werden. Ausführliche Informationen finden Sie im *One Identity Manager Web Designer Web Portal Anwenderhandbuch* und im *One Identity Manager Anwenderhandbuch für die Benutzeroberfläche der One Identity Manager-Werkzeuge*.

- Voraussetzung für die Nutzung der Volltextsuche ist ein Anwendungsserver, auf dem der Suchdienst installiert ist.
- Wenn Sie das Web Portal direkt über einen Anwendungsserver mit installiertem Suchdienst betreiben, können Sie die Volltextsuche sofort nutzen.
- Wenn Sie das Web Portal mit einem Anwendungsserver ohne installierten Suchdienst oder mit einer direkten Datenbankverbindung betreiben, tragen Sie in der Konfigurationsdatei des Web Portal einen Anwendungsserver mit installiertem Suchdienst ein. Erst dann ist die Volltextsuche im Web Portal verfügbar.
- Um die Volltextsuche im Manager zu nutzen, müssen Sie den Manager über einen Anwendungsserver mit installiertem Suchdienst betreiben.

Ausführliche Informationen zur Installation eines Anwendungsservers und zur Konfiguration des Web Portal für die Volltextsuche finden Sie im *One Identity Manager Installationshandbuch*.

Für die Konfiguration für die Volltextsuche gilt:

- Soll eine Tabelle oder eine Datenbanksicht für die Volltextsuche indiziert werden, müssen die Spalten `XDateInserted`, `XDateUpdated` und `XObjectKey` vorhanden sein.
- Spalten für die Volltextsuche benötigen eine Wichtung. Eine höhere Wichtung führt zu einer höheren Position im Suchergebnis. In der Standardinstallation werden Spalten für die Volltextsuche mit einer Wichtung von **1** bereitgestellt.

Beispiel:

Die Spalte `Person.CentralAccount` ist mit dem Wert **1** gewichtet. Die Spalte `ADSAccount.SAMAccountName` ist mit dem Wert **0,5** gewichtet. Damit wird im Ergebnis der Volltextsuche die Person vor dem Benutzerkonto aufgelistet.

- Es können nur Spalten mit den .Net-Datentypen `String` oder `Text` in die Volltextsuche aufgenommen werden.

  Ausnahme: Spalten, die eine Liste zulässiger Werte enthalten, können immer in die Volltextsuche aufgenommen werden.
- Spalten von Tabellen mit dem Nutzungstyp **Arbeitstabellen** oder **Historische Bewegungsdaten** können nicht in die Volltextsuche aufgenommen werden.
- Spalten von Zuordnungstabellen (`M:N`-Tabellen, `M:all`-Tabellen) können nicht in die Volltextsuche aufgenommen werden.

Durch den Suchdienst indiziert werden:

- der Inhalt einer Spalte,
- die Anzeigewerte von Fremdschlüsselspalten,
- die Anzeigewerte von Listen zulässiger Werte,
- die Übersetzungen für jede aktivierte Sprachkultur,
- der Anzeigewert eines Objektes, wenn die Primärschlüsselspalte der Tabelle für die Volltextsuche konfiguriert ist.

  Der Anzeigewert eines Objektes ergibt sich aus dem Anzeigemuster, das für die Tabelle definiert ist. Die Wichtung des Anzeigewertes ergibt sich aus der Wichtung der Primärschlüsselspalte der Tabelle.

Beispiel:

Die Spalte `Person.UID_Person` ist für die Volltextsuche konfiguriert. Das Anzeigemuster der Tabelle `Person` ist definiert als `%InternalName% (%CentralAccount%)`.

Für die Person Clara Harris wird somit der Anzeigewert **Clara Harris (CLARAH)** indiziert.

Bei Änderungen an einer Tabelle mit indizierten Spalten, den referenzierten Tabellen oder den Übersetzungen wird der Suchindex aktualisiert.

In der Standardinstallation sind bereits einige wichtige Spalten für die Volltextsuche konfiguriert. Bei Bedarf können Sie weitere Spalten für die Volltextsuche konfigurieren.

Um eine Spalte für die Volltextsuche zu konfigurieren

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Spalte** und bearbeiten Sie die Eigenschaft **Index Wichtung**.
   - Ist der Wert kleiner oder gleich **0** erfolgt keine Indizierung.
   - Ist der Wert größer als **0** wird der Datenwert indiziert.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Skripte zum bedingten Anzeigen und Bearbeiten von Spalten

Grundsätzlich werden die Berechtigungen eines Benutzers zum Anzeigen und Bearbeiten von Spalten über die Berechtigungen der Berechtigungsgruppen gesteuert.

Zusätzlich können Sie Skripte zum bedingten Anzeigen oder Bearbeiten einer Spalte verwenden. So kann beispielsweise gesteuert werden, dass eine Spalte auf einem Stammdatenformular im Manager nur angezeigt wird oder bearbeitbar ist, wenn eine andere Spalte einen bestimmten Wert besitzt.

Beispiel:

Eine Systemrolle ist bis zum Erreichen des Freigabedatums deaktiviert. Für diesem Zeitraum soll verhindert werden, dass die Benutzer im Manager die Option **deaktiviert** bearbeiten können.

Für diesen Fall ist ein Skript für die Bearbeitbarkeit an der Spalte `ESet.IsInActive` hinterlegt.

```
If $ReleaseDate:Date$>Connection.LocalNow Then
     Value = False
Else
     Value = True
```

```
End If
```

HINWEIS:

- Die Skripte verändern nicht die Berechtigungen eines Benutzers, sondern lediglich das Verhalten beim Laden eines Objektes in den One Identity Manager-Werkzeugen. Wenn Sie die Sichtbarkeit und die Bearbeitbarkeit einer Spalte grundsätzlich einschränken wollen, passen Sie die Spaltenberechtigungen für die Berechtigungsgruppen an. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.
- Die Skripte wirken nur für interaktiv geladene Objekte.
- In Listen, beispielsweise im Manager oder im Web Portal, werden die Objekte nicht-interaktiv geladen. Damit wirken die Skripte hier nicht.
- Im Web Portal werden viele Objekte aus Performancegründen nicht-interaktiv geladen. Wenn Sie das Verhalten im Web Portal nutzen möchten, passen Sie die Komponenten im Web Designer kundenspezifisch an. In diesem Fall kann es zur Beeinträchtigung der Performance beim Laden von Objekten kommen. Ausführliche Informationen zum Bearbeiten der Komponenten Web Designer finden Sie im *One Identity Manager Referenzhandbuch für den Web Designer*.

Um ein Skript zum bedingten Anzeigen oder Bearbeiten für eine Spalte festzulegen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Zugangsbildung** und erfassen Sie folgende Skripte in VB.Net Syntax.
   - **Skript zur Sichtbarkeit**: Skript zum bedingten Anzeigen der Spalte. Ist der Rückgabewert **False**, wird die Spalte in den One Identity Manager-Werkzeugen nicht angezeigt.
   - **Skript zur Bearbeitbarkeit**: Skript zum bedingten Bearbeiten der Spalte. Ist der Rückgabewert **False**, kann die Spalte in den One Identity Manager-Werkzeugen nicht bearbeitet werden.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99
- Arbeiten mit Objekten im One Identity Manager auf Seite 22

Spaltendefinitionen bearbeiten

Um Spalteneigenschaften zu bearbeiten

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie im Schemaeditor die Spalte und bearbeiten Sie die Spalteneigenschaften.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Spaltendefinitionen auf Seite 99

Eigenschaften von Spaltendefinitionen

Tabelle 27: Eigenschaften von Spalten

Eigenschaft	Beschreibung
Tabelle	Bezeichnung der Tabelle, zu der die Spalte gehört.
Spalte	Bezeichnung der Spalte im Datenmodell.
Anzeigename	Sprachabhängige Bezeichnung der Spalte zur Darstellung in der Oberfläche der Administrationswerkzeuge. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Bemerkung	Hinweise zur Nutzung der Spalte. Die Bemerkung wird in der Hilfefunktion zu einer Spalte in den einzelnen Administrationswerkzeugen angezeigt. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Deaktiviert durch Präprozessor	Ist eine Spalte durch eine Präprozessorbedingung deaktiviert, dann wird durch den Database Compiler die Option gesetzt. Weitere Informationen finden Sie unter Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369.
Präprozessorbedingung	Spalten können Sie mit Präprozessorbedingungen versehen. Damit ist die Spalte nur verfügbar, wenn die Präprozessorbedingung erfüllt ist. Weitere Informationen finden Sie unter Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369. HINWEIS: Einen Überblick über die vorhandenen Präprozessorabhängigkeiten erhalten Sie im Designer in

Eigenschaft	Beschreibung
	der Kategorie **One Identity Manager Schema > Präprozessorabhängigkeiten**.
Reihenfolge	Positionierung der Spalte zur Darstellung auf einem generischen Formular sowie auf dem kundenspezifischen Tabreiter der Standardformulare. Spalten, deren Wert kleiner **1** ist, werden nicht auf den Formularen angezeigt. Weitere Informationen finden Sie unter Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen auf Seite 162.
Gruppe	Die Gruppierung wird zur Darstellung der Spalte auf allgemeinen Stammdatenformularen verwendet. Pro Gruppe wird ein eigener Tabreiter auf dem generischen Formular oder dem Stammdatenformular erzeugt. Weitere Informationen finden Sie unter Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen auf Seite 162.
Basisspalte	Handelt es sich um eine Datenbanksicht mit dem Tabellentyp **View**, so ist hier der Verweis auf die Spalte der Basistabelle eingetragen. Weitere Informationen finden Sie unter Datenbanksichten vom Typ View auf Seite 59. Beispiel: Die Datenbanksicht `Department` ist ein Teilausschnitt der Basistabelle `Basetree`. Als Basisspalten sind die Spalten der Tabelle `Basetree` eingetragen. Spalte: `Department.DepartmentName` — Basisspalte: `BaseTree.Ident_Org` Spalte: `Department.Description` — Basisspalte: `BaseTree.Description`
Anpassung der Liste zulässiger Werte nicht erlaubt	Gibt an, ob eine Spalte für die Anpassung der zulässigen Werten freigeschaltet ist. Weitere Informationen finden Sie unter Zulässige Werte von Spalten auf Seite 92.
Werteliste definiert	Kennung, dass die Werte in dieser Spalte einem der Werte aus der **Liste zulässiger Werte** entsprechen müssen, oder aber leer sind. Weitere Informationen finden Sie unter Zulässige Werte von Spalten auf Seite 92.
Liste zulässiger Werte	Ist eine Spalte für die Anpassung der zulässigen Werten freigeschaltet, das heißt die Option **Anpassung der Liste zulässiger Werte nicht erlaubt** ist nicht gesetzt

Eigenschaft	Beschreibung
	und ist die Option **Werteliste definiert** gesetzt, dann können Sie eine Werteliste erfassen oder erweitern. Weitere Informationen finden Sie unter Zulässige Werte von Spalten auf Seite 92.
Anpassung der Bitmaske nicht erlaubt	Gibt an, ob eine Spalte für die Anpassung der Bitmaske freigeschaltet ist. Weitere Informationen finden Sie unter Bitmasken definieren auf Seite 94.
Definierte Bitmaske	Bedeutung der einzelnen Bit-Positionen, wenn die Spalte eine Bitmaske enthält. Das erste Bit in der Definition startet mit dem Index **0**. Weitere Informationen finden Sie unter Bitmasken definieren auf Seite 94.
Sortierkriterium der Bitmaske	Sortierungskriterium für die Anzeige der Werte. Zur Auswahl stehen: • **Bit-Position**: Die Sortierung erfolgt nach der Position. • **Anzeigewert:** Die Sortierung erfolgt nach dem Anzeigewert. Weitere Informationen finden Sie unter Bitmasken definieren auf Seite 94.
Mehrsprachig	Gibt an, ob der Wert in dieser Spalte mehrsprachig angegeben werden kann. Zulässige Werte sind: • **Übersetzungsziel**: Der Inhalt der Spalte soll übersetzt angezeigt werden. • **Übersetzungsquelle**: Die Spalte liefert Übersetzungen. • **#LD Inhalt**: Die Spalte besitzt Inhalte in #LD-Notation. Die Inhalte werden für die Übersetzung extrahiert. • **Ohne Fallback aus Textspeicher**: Der Textspeicher soll für die Spalte nicht als Fallback verwendet werden. Aus der Kombination der Werte ergibt sich, wie die Übersetzung ermittelt wird. Weitere Informationen finden Sie unter Spalten zur Übersetzung kennzeichnen auf Seite 235.
Syntax	Syntaxtyp von Daten dieser Spalte. Der Syntaxtyp einer Spalte wird genutzt, um in den One Identity Manager-

Eigenschaft	Beschreibung
	Werkzeugen die entsprechende Syntaxhervorhebung oder Eingabehilfen zur Verfügung zu stellen. Zulässige Syntaxtypen sind: • **HTML**: Eingaben im HTML Format. • **Picture**: Bilder. • **SQL.Query**: Komplette Datenbankabfragen. • **SQL.Special**: Spezielle Syntax für Datenbankabfragen. • **SQL.WhereClause**: Where-Klausel für Datenbankabfragen. • **Text.Dollar**: Eingabe von $-Notation. • **UNC**: UNC Pfade. • **URL**: URL Angaben. • **VB.Class**: Vollständige VB.Net-Klassendefinitionen. • **VB.Instruction**: VB.Net-Anweisungen in der Form `Value =`. • **VB.Method**: Einzelne Methoden oder Funktionen in VB.Net. • **XML**: Eingaben im XML Format. • **ConnectionString**: Eingabe einer Verbindungszeichenkette. • **JSON**: Eingaben im JSON Format. • **Color**: Eingabe von Farbcodes.
Anzahl Kommastellen	Anzahl der Kommastellen, die bei der Abbildung von Werten.Weitere Informationen finden Sie unter Kommastellen für die Anzeige von Werten festlegen auf Seite 87.
Datumszusatz	Zusätzliche Informationen zur Darstellung von Datumsangaben und Zeitangaben in den One Identity Manager-Werkzeugen.
Index Wichtung	Wichtung der Spalte in der Indizierung. Wird verwendet für die Suchindizierung für die Volltextsuche. Eine höhere Wichtung führt zu einer höheren Position im Suchergebnis. Ist der Wert keiner oder gleich **0** erfolgt keine Indizierung. Ist der Wert größer als **0** wird der Datenwert indiziert. In der Standardinstallation zu indizierende Spalten werden

Eigenschaft	Beschreibung
	mit einer Wichtung von **1** bereitgestellt. Weitere Informationen finden Sie unter Konfigurieren von Spalten für die Volltextsuche auf Seite 95.
Datentyp in Datenbank	Angezeigt wird der .Net-Datentyp der Spalte. Dieser dient zur internen Verwendung und kann nicht bearbeitet werden. Die .Net-Datentypen werden intern in SQL-Datentypen abgebildet. Ist kein Wert eingetragen, so wird der Datentyp aus dem Datenbankschema benutzt. Zulässige Datentypen sind: **.Net-Datentyp** / **Abbildung als SQL Datentyp** Binary / varbinary, timestamp Bool / bit Date / datetime Double / float Int / int Long / bigint String / nvarchar/varchar/nchar Text / nvarchar/varchar
maximale Größe in DB	Maximale Länge der Spalte im Datenbankschema.
minimale Größe in DB	Minimale Länge der Spalte im Datenbankschema.
Primärschlüssel	Der Primärschlüssel wird bei der Erstellung der Datenbanktabelle angegeben.
UID-Spalte	Gibt an, ob es sich bei der Spalte um eine UID handelt. Die Angabe ist nur zulässig für Spalten mit dem .Net-Datentyp `String` und einer Länge von 38 Zeichen.
Standardwert	Gibt an, ob für diese Spalte über eine Bildungsregel ein Standardwert definiert ist.
BLOB-Feld	Mit dieser Option sind Textspalten gekennzeichnet, deren Dateninhalt so groß ist, dass er intern im SQL Server nicht innerhalb der Zeile gehalten werden kann, sondern nur als Verweis gespeichert wird. Damit wird ein schnellerer Zugriff auf die Daten erreicht.
Aufzeichnen von Änderungen	Gibt an, ob Änderungen an dieser Spalte aufzuzeichnen sind. Weitere Informationen finden Sie unter Aufzeichnen

Eigenschaft	Beschreibung
	von Datenänderungen auf Seite 350.
Aufzeichnen beim Löschen	Gibt an, ob die Spalte beim Löschen eines Objektes aufzuzeichnen ist. Weitere Informationen finden Sie unter Aufzeichnen von Datenänderungen auf Seite 350.
Export für SPML-Schema	Gibt an, ob die Tabelle für das SPML-Schema exportiert werden soll. Weitere Informationen finden Sie unter Vorbereiten des One Identity Manager Schemas für den Export in das SPML-Schema auf Seite 542.
Nicht exportieren (XML-Export)	Die Spalte wird bei Datentransport nicht exportiert. Die Eigenschaft wird beim Transport von Daten zwischen Datenbanken berücksichtigt.
Nicht importieren (XML-Import)	Die Spalte wird bei Datentransport nicht importiert. Die Eigenschaft wird beim Transport von Daten zwischen Datenbanken berücksichtigt.
MVP-Spalte	Diese Spalte ist ein Multi-Valued-Property (MVP), dessen einzelne Werteinträge durch **char(7)** oder **chr(7)** getrennt werden.
Multi-Value Spezifizierung	Sie können zusätzliche Anforderungen an die Werte einer MVP-Spalte festlegen. Zur Auswahl stehen: • **Eindeutig**: Wenn die Werte eindeutig sein müssen, aktivieren Sie die Option. • **Groß- und Kleinschreibung beachten:** Wenn bei der Prüfung der Werte die Groß-und Kleinschreibung beachtet werden soll, aktivieren Sie die Option. • **Akzent nicht berücksichtigen**: Wenn Akzentzeichen bei der Prüfung der Werte nicht berücksichtigt werden sollen, aktivieren Sie die Option. Weitere Informationen finden Sie unter Anforderungen an MVP-Spalten festlegen auf Seite 93.
Mehrzeilig	Gibt an, ob der Inhalt einer Spalte mehrzeilig sein darf. Spalten, die mit dieser Option gekennzeichnet sind, werden auf einem generischen Formular mit mehrzeiligen Eingabefeldern dargestellt.
Keine automatische Berechtigungsvergabe	Für kundenspezifische Spalte an einer vordefinierten Tabelle erfolgt keine automatische Berechtigung von vordefinierten Berechtigungsgruppen, obwohl der Konfigurationsparameter **Common | AutoEx-**

Eigenschaft	Beschreibung
	tendPermissions gesetzt ist.
Spalte enthält Beschreibung	Pro Tabelle kann eine Spalte, die eine Beschreibung enthält mit dieser Option gekennzeichnet werden. Die Beschreibung wird zur Anzeige auf den Zuordnungssteuerelementen der Benutzeroberfläche verwendet.
Enthält Namensbestandteile für die Kennwortprüfung	Gibt an, ob die Spalte Namensbestandteile enthält. Abhängig von der Konfiguration der Kennwortrichtlinien können Spalten mit Namensbestandteilen bei der Kennwortprüfung eingeschlossen werden. Ausführliche Informationen zu Kennwortrichtlinien finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.
Spalte enthält Hierarchieinformation	Pro Tabelle kann eine Spalte, die Hierarchieinformationen in lesbarer Form abbildet, mit dieser Option gekennzeichnet werden. Diese Spalte wird dann zur Abbildung der Hierarchie auf den Zuordnungssteuerelementen der Benutzeroberfläche verwendet.
Teil des Primärschlüssels	Die Spalte ist Teil des Primärschlüssels.
Teil des alternativen Primärschlüssels	Die alternativen Primärschlüssel sind in der Standardauslieferung bereits festgelegt, die Definition ist jedoch kundenspezifisch änderbar. Genutzt werden die alternativen Primärschlüssel unter anderem beim Transport von Daten.
Teil des Schlüssels einer M:all-Tabelle	Kennzeichnung der Fremdschlüssel einer M:all-Tabelle. Der Fremdschlüssel und der dynamische Fremdschlüssel einer M:all-Tabelle werden mit dieser Option gekennzeichnet.
Anzeige im Assistenten	Gibt an, ob die Spalte im Regeleditor für Complianceregeln zum Erstellen von Abfragen und in tabellarischen Übersichten im Web Portal zur Anzeige angeboten wird.
Rekursiver Schlüssel	Gibt an, ob diese Spalte den Verweis auf ein übergeordnetes Objekt enthält. Diese Angabe wird für die Darstellung hierarchischer Tabellen benötigt. Beispiel: In der Tabelle `ADSContainer` enthält die Spalte `UID_ParentADSContainer` den Verweis auf den übergeordneten Active Directory Container. Zur Abbildung dieser hierarchischen Verknüpfung auf den Formularen wird die Spalte `UID_ParentADSContainer` mit dieser Option gekenn-

Eigenschaft	Beschreibung
	zeichnet.
Verschlüsselt	Gibt an, ob der Wert in dieser Spalte verschlüsselt ist. Bei der Verschlüsselung der Datenbank wird der Wert dieser Spalte verschlüsselt. HINWEIS: Wenn Sie Datenbankspalten mit dieser Option versehen, müssen Sie die Datenbank erneut verschlüsseln. Ausführliche Informationen zur Verschlüsselung der Datenbank finden Sie im *One Identity Manager Installationshandbuch*.
Dynamische Fremd-schlüsselspalte	Dynamische Fremdschlüssel verweisen auf den Objekt-schlüssel anderer Tabellen. Der Objektschlüssel setzt sich aus dem Tabellennamen und den Primärschlüsselwerten des konkreten Objektes zusammen. Die zulässigen Tabellen können eingeschränkt werden. Ist keine Einschränkung festgelegt, sind alle Tabellen zulässig. Weitere Informationen finden Sie unter Dynamische Fremdschlüssel auf Seite 113.
Keine Protokollierung	Gibt an, ob der Inhalt der Spalte in Protokollen aufge-zeichnet werden soll, beispielsweise im Protokoll des One Identity Manager Service.
Spalte der Proxyview	Wird die Spalte in einer Datenbanksicht vom Typ **Proxy** verwendet, ist hier die korrespondierende Spalte der Sicht eingetragen. Beispielsweise wird die Spalte `ADSDomain.DisplayName` in der Sicht `UNSRoot` auf die Spalte `RootObjectDisplay` abgebildet. Weitere Informationen finden Sie unter Datenbanksichten vom Typ Proxy auf Seite 61.
Tabellenübergreifende Suche	Die einzelnen Werte dieser Spalte werden für eine schnellere tabellenübergreifende Suche vorbereitet. Zusätzlich wird die Suche von Einzelwerten in MVP-Spalten unterstützt. Die interne Abbildung der vorbereiteten Daten erfolgt in der Tabelle `QBMSplittedLookup`. Zulässig sind die Werte: • **Zentrales Benutzerkonto** (CentralAccount) • **E-Mail Adresse** (EMail) Bei Bedarf können Sie die Liste der zulässigen Werte erweitern und kundenspezifisch auswerten. • Spalten, die einen Benutzerkontennamen enthalten, kennzeichnen Sie im Designer in der

Eigenschaft	Beschreibung
	mit dem Wert **Zentrales Benutzerkonto**. • Spalten, die eine E-Mail-Adresse enthalten, kennzeichnen Sie im Designer in der Eigenschaft **Tabellenübergreifende Suche** mit dem Wert **E-Mail-Adresse**. Die Funktionalität wird beispielsweise verwendet bei der Ermittlung eines eindeutigen zentralen Benutzerkontos oder einer eindeutigen Standard-E-Mail-Adresse für eine Person. In der Standardinstallation sind Spalten, die bei der Bildung des zentralen Benutzerkonto oder einer E-Mail-Adresse berücksichtigt werden, mit dieser Eigenschaft gekennzeichnet. Die Auswertung erfolgt in den Datenbanksichten `QERCentralAccount` und `QERMailAddress`.
Bemerkungen (Kunde)	Freitextfeld für zusätzliche Erläuterungen.
Kundenspezifische Bildungsregel/Formatierung nicht erlaubt	Gibt an, ob die Standardkonfiguration der Spalte durch den Anwender geändert werden kann, beispielsweise Bildungsregel, Formatierungsskript, minimale Länge, maximale Länge, Spaltenformat.
Max. Länge	Maximale Länge der Spalte. Ist der Wert gleich **0**, so wird die Länge aus dem Datenbankschema benutzt.
Min. Länge	Minimale Länge der Spalte. Für Spalten, welche in der Oberfläche der Administrationswerkzeuge als Pflichteingabefeld dargestellt werden, setzen Sie die Mindestlänge auf **1** oder größer.
Spaltenformat	Legen Sie die zulässige Formatierung von Werten einer Spalte fest. Weitere Informationen finden Sie unter Vordefinierte Formatierungstypen verwenden auf Seite 88. Die zulässige Formatierung einer Spalte können Sie über Formatierungstypen oder Formatierungsskripte steuern.
Überschreibend	Gibt an, ob die Bildungsregel überschreibend oder nicht überschreibend wirken soll. Ist die Option aktiviert, wird die Bildungsregel immer angewendet. Ist die Option deaktiviert, wird die Bildungsregel nur angewendet, wenn in der Spalte noch kein Wert eingetragen ist. Weitere Informationen finden Sie unter Bildungsregeln bearbeiten auf Seite 83.
Bildungsregel	Bildungsskript der Bildungsregel. Das Bildungsskript wird in VB.Net-Syntax abgelegt, was die Nutzung aller VB.Net-Skriptfunktionen zulässt. Weitere Informationen finden

Eigenschaft	Beschreibung
	Sie unter Bildungsregeln bearbeiten auf Seite 83.
Schwellwert (Abbruch)	Limitierung der Anzahl der direkt durch eine Bildungsregel geänderten Objekte. Bei Erreichen der angegebenen Anzahl, wird die Verarbeitung mit einer Fehlermeldung abgebrochen. Weitere Informationen finden Sie unter Ausführung von Bildungsregeln limitieren auf Seite 85. HINWEIS: Ist ein Schwellwert für den Abbruch angegeben, muss er größer als der Schwellwert für die asynchrone Verarbeitung sein.
Schwellwert (asynchron)	Limitierung der Anzahl der direkt durch eine Bildungsregel geänderten Objekte. Wird die angegebene Anzahl erreicht, wird die Verarbeitung asynchron über den One Identity Manager Service erfolgen. Weitere Informationen finden Sie unter Ausführung von Bildungsregeln limitieren auf Seite 85.
Kein automatisches Einkürzen durch Bildungsregel	Gibt an, ob der Wert einer Spalte eingekürzt werden soll, wenn bei der Verarbeitung der Bildungsregel die maximale Länge der Spalte überschritten wird. Ist die Option aktiviert, erfolgt keine automatische Einkürzung des Wertes auf die maximale Spaltenlänge. Weitere Informationen finden Sie unter Bildungsregeln bearbeiten auf Seite 83.
Formatierungsskript	Formatierungsskript für die Spalte. Ein Formatierungsskript zur Wertüberprüfung wird in VB.Net-Syntax abgelegt, was die Nutzung aller VB.Net-Skriptfunktionen zulässt.
Skript zur Sichtbarkeit	Skript zum bedingten Anzeigen der Spalte in den One Identity Manager-Werkzeugen. Weitere Informationen finden Sie unter Skripte zum bedingten Anzeigen und Bearbeiten von Spalten auf Seite 97.
Skript zur Bearbeitung	Skript zum bedingten Bearbeiten der Spalte in den One Identity Manager-Werkzeugen. Weitere Informationen finden Sie unter Skripte zum bedingten Anzeigen und Bearbeiten von Spalten auf Seite 97.
Fremdschlüssel	Die Spalte referenziert ein Objekt einer anderen Tabelle.
Mittlere Spaltenlänge	Die Informationen werden einmal täglich in den Wartungsaufträgen ermittelt. Das Datenmaterial kann bei der Planung von Kapazitäten und Wartungsarbeiten an der Datenbank helfen.
Bildungsregel geändert	(Nur für interne Verwendung) Kennzeichen, dass die

Eigenschaft	Beschreibung
	Bildungsregel geändert wurde.
Kein DB Transport	Spalten, die mit dieser Option gekennzeichnet sind, können nicht in ein Kundenkonfigurationspaket ausgenommen werden. Diese Spalten werden von Datentransport ausgeschlossen.
Mappingrichtung	Gibt die Mappingrichtung an, welche für diese Spalte bei der Synchronisation zwischen zwei One Identity Manager-Datenbanken angewendet wird. Ausführliche Informationen dazu finden Sie im *One Identity Manager Anwenderhandbuch für den One Identity Manager Konnektor*.

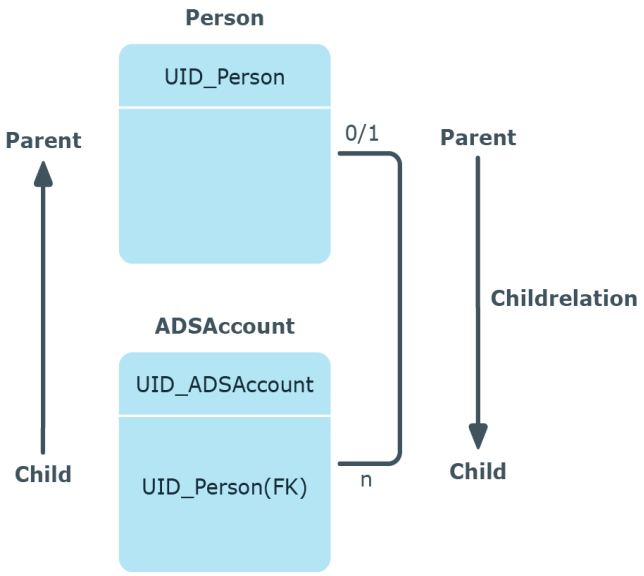
Verwandte Themen

- Spaltendefinitionen bearbeiten auf Seite 99
- Verwendung von Visual Basic .NET-Skripten auf Seite 375

Tabellenbeziehungen

Wie aus dem Datenmodell des One Identity Manager ersichtlich, bestehen zwischen den Objekten Beziehungen (Parent/Childrelations). Bei Bearbeitung eines Objektes über die Objektschicht des One Identity Manager besteht Zugriff auf alle mit diesem Objekt in Beziehung stehenden Fremdschlüssel-Objekte (FK). Der Zugriff auf Objekte über die Beziehungen erfolgt in VB.Net-Notation.

Abbildung 11: Parent/Childrelation am Beispiel Person-ADSAccount



HINWEIS: Die Tabellenbeziehungen kundenspezifischer Tabellen sind immer bearbeitbar. Die Bearbeitung der Tabellenbeziehung der mitgelieferten Standardtabellen ist nur möglich, wenn die Überprüfung der referentiellen Integrität durch die Objektschicht erfolgt.

Um Tabellenbeziehungen zu bearbeiten

1. Wählen Sie im Designer die Kategorie**One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie unter **Tabellenrelationen** die Tabellenbeziehung und bearbeiten Sie in der Ansicht **Relationseigenschaften** die Eigenschaften.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Tabelle 28: Eigenschaften von Tabellenbeziehungen

Eigenschaft	Beschreibung
Anzeigename	Sprachabhängige Bezeichnung der Tabellenbeziehung zur Darstellung in der Oberfläche der Administrationswerkzeuge.
Transport nur im Verbund	Gibt an, ob bei Datentransporten der Inhalt der Tabelle zusammen mit den Inhalten der referenzierten Tabelle übertragen werden soll. Die Werte können Sie miteinander kombinieren. Zulässige Werte sind: • **Kein Wert**: Abhängigkeiten werden nicht beachtet. • **CR Richtung**: Alle Objekte, die auf dieses Objekt verweisen, werden mit exportiert. Es erfolgt eine Übermengenbehandlung. • **FK Richtung**: Alle per Fremdschlüssel referenzierten Objekte werden mit exportiert. Es erfolgt eine Übermengenbehandlung. • **Bei Übermengenbehandlung ignorieren**: Referenzierte Objekte, die im Ziel vorhanden sind, jedoch im Transportpaket nicht enthalten sind, werden nicht gelöscht. Beispiel: Beim Transport eines Prozesses (Tabelle `JobChain`) sollen die Prozessschritte (Tabelle `Job`), die Ereignisse (Tabellen `JobEventGen` und `QBMEvent`) sowie die Prozessschrittparameter (Tabelle `JobRunParameter`) immer mit transportiert werden. Das soll unabhängig davon erfolgen, ob der Prozess, ein einzelner Prozessschritt oder ein Prozessschrittparameter in ein Transportpaket übernommen wird. Die Tabellenrelationen sind mit den Werten **CR Richtung** und **FK Richtung** gekennzeichnet. Die Parametervorlagen (Tabelle `JobParameter`), die in den

Eigenschaft	Beschreibung
	Prozessschrittparameter (Tabelle `JobRunParameter`) verwendet werden, sind beim Transport nicht zu übernehmen. Die Tabellenrelationen sind mit keinem Wert gekennzeichnet.
Änderungsdatum für Abhängigkeiten aktualisieren	Beim Einfügen, Ändern und Löschen von M:N-Einträgen wird der Wert der Spalte `XDateSubItem` im zugehörigen Parent-Eintrag aktualisiert. Wird benötigt für die Provisionierung von Mitgliedschaften in die Zielsysteme.
Export für SPML-Schema	Gibt an, ob die Tabellenbeziehung für das SPML-Schema exportiert werden soll.
Übergeordnetes Objekt in der Jobqueue	Gibt an, ob das übergeordnete Objekt in die Liste der von einem Prozess betroffenen Objekte aufgenommen werden soll. Damit kann verhindert werden, dass das übergeordnete Objekt mehrfach gleichzeitig verarbeitet wird.
Parent-Spalte	Eindeutige Kennung der Parent-Spalte.
Parent-Beziehung konfigurierbar	Gibt an, ob diese referentielle Integrität konfiguriert werden darf.
Prüfinstanz der Parent-Beziehung	Gibt an, wer die Überprüfung dieser referentiellen Integrität ausführen soll. Zulässige Werte sind: • **DLL**: Die Überprüfung erfolgt durch die Objektschicht. • **Trigger**: Zur Überwachung durch die Datenbank werden Trigger und Constraints eingesetzt. Die Trigger und Constraints werden unter Einbeziehung der vorgegebenen Beschränkungen durch den DBQueue Prozessor automatisch erzeugt und gegebenenfalls angepasst. Für kundenspezifische Tabellen legen Sie die Prüfinstanz und die Beschränkungen bei der Erweiterung des One Identity Manager Schemas fest. • **Nothing**: keine Überprüfung
Restriktion der Parent-Beziehung	Restriktion der Beziehung. Zulässige Werte sind: • **Delete**: Beim Löschen des Objektes werden Abhängigkeiten nicht beachtet. • **Delete Cascade**: Beim Löschen des Objektes werden alle davon abhängigen Objekte ebenfalls gelöscht. • **Delete Restrict**: Das Objekt wird erst gelöscht, nachdem keine Beziehungen zu anderen Objekten bestehen. • **Delete Set NULL**: Beim Löschen des Objektes werden in allen abhängigen Objekten die Verweise auf das zu löschende Objekt entfernt (`SetNULL`).

Eigenschaft	Beschreibung
Generierte Restriktionsprüfung der Parent-Beziehung	Kennung für Trigger und Constraints, die durch den DBQueue Prozessor automatisch generiert werden.
Verbundene Spalte	Eindeutige Kennung der verbundenen Spalte.
Child-Beziehung konfigurierbar	Gibt an, ob diese referentielle Integrität konfiguriert werden darf.
Prüfinstanz der Child-Beziehung	Gibt an, wer die Überprüfung dieser referentiellen Integrität ausführen soll. Zulässige Werte sind: • **DLL**: Die Überprüfung erfolgt durch die Objektschicht. • **Trigger**: Zur Überwachung durch die Datenbank werden Trigger und Constraints eingesetzt. Die Trigger und Constraints werden unter Einbeziehung der vorgegebenen Beschränkungen durch den DBQueue Prozessor automatisch erzeugt und gegebenenfalls angepasst. Für kundenspezifische Tabellen legen Sie die Prüfinstanz und die Beschränkungen bei der Erweiterung des One Identity Manager Schemas fest. • **Nothing**: keine Überprüfung
Restriktion der Child-Beziehung	Restriktion der Beziehung. Zulässige Werte sind: • **Insert**: Beim Einfügen des Objektes werden Abhängigkeiten nicht beachtet. • **Insert Restrict**: Beim Einfügen des Objektes wird geprüft, ob das referenzierte Objekt existiert.
Generierte Restriktionsprüfung der Child-Beziehung	Kennung für Trigger und Constraints, die durch den DBQueue Prozessor automatisch generiert werden.
Beziehungs-ID	Identifikator der Beziehung. Dieser wird für beide Richtungen verwendet.
M:N Beziehung	Gibt an, ob die Child-Beziehung über eine M:N-Beziehung erreichbar ist.
Tabellenbeziehung	Eindeutige Kennung der Tabellenbeziehung.
Beziehung (Basis)	Verweis auf zugrunde liegende Basisbeziehung, sofern an der Beziehung eine Sicht (View) beteiligt ist.
Beziehung (M:N)	Eindeutige Kennung der M:N Beziehung.

Verwandte Themen

- Dynamische Fremdschlüssel auf Seite 113
- Datenmodell im Designer anzeigen auf Seite 50

- Vorbereiten des One Identity Manager Schemas für den Export in das SPML-Schema auf Seite 542

Dynamische Fremdschlüssel

Dynamische Fremdschlüssel werden verwendet, wenn eine Referenz auf unterschiedliche Tabellen verweisen kann. Beispielsweise kann der Manager eines Benutzerkontos (Tabelle `ADSAccount.ObjectKeyManager`) ein anderes Benutzerkonto (Tabelle `ADSAccount`) oder ein Kontakt (Tabelle `ADSContact`) sein.

Dynamische Fremdschlüssel verweisen auf den Objektschlüssel (`XObjectKey`) der zulässigen Tabellen. Die zulässigen Tabellen können eingeschränkt werden. Ist keine Einschränkung festgelegt, sind alle Tabellen zulässig. Einschränkungen werden in der Tabelle `DialogValidDynamicRef` abgelegt.

Wenn Sie kundenspezifische dynamische Fremdschlüssel definieren, muss mindestens einer der beteiligten Partner - dynamische Fremdschlüsselspalte oder referenzierte Tabelle - ein kundenspezifisches Objekt sein. Die Erweiterung von vordefinierten dynamischen Fremdschlüsseln um Referenzen auf vordefinierte Tabellen ist nicht möglich.

Um dynamische Fremdschlüssel anzuzeigen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.

   Die dynamischen Fremdschlüssel werden unter **Dynamische Tabellenrelationen** angezeigt.

Um einen dynamischen Fremdschlüssel zu definieren

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Auf dem Tabreiter **Sonstiges** bearbeiten Sie folgende Informationen.
   a. Aktivieren Sie die Option **Dynamische Fremdsschlüsselspalte**.
   b. Ist der dynamische Fremdschlüssel Bestandteil einer M:all-Tabelle, aktivieren Sie die Option **Teil des Schlüssels einer M:all-Tabelle**.
5. Auf dem Tabreiter **Zulässige Zieltabellen** klicken Sie neben der Auswahlliste **Dynamische referenzierte Tabellen** die Schaltfläche und erfassen Sie die folgenden Informationen.

Tabelle 29: Eigenschaften für dynamischer Fremdschlüssel

Eigenschaft	Beschreibung
Tabelle	Tabelle, aus der der Objektschlüssel ermittelt werden soll.
Restriktion der Parent-Beziehung	Restriktion der Beziehung. Zulässige Werte sind: • **Delete**: Beim Löschen des Objektes werden Abhängigkeiten nicht beachtet. • **Delete Cascade**: Beim Löschen des Objektes werden alle davon abhängigen Objekte ebenfalls gelöscht. • **Delete Restrict**: Das Objekt wird erst gelöscht, nachdem keine Beziehungen zu anderen Objekten bestehen. • **Delete Set NULL**: Beim Löschen des Objektes werden in allen abhängigen Objekten die Verweise auf das zu löschende Objekt entfernt (`SetNULL`).
Prüfinstanz der Parent-Beziehung	Gibt an, wer die Überprüfung dieser referentiellen Integrität ausführen soll. Zulässige Werte sind: • **DLL**: Die Überprüfung erfolgt durch die Objektschicht. • **Trigger**: Zur Überwachung durch die Datenbank werden Trigger und Constraints eingesetzt.
Restriktion der Child-Beziehung	Restriktion der Beziehung. Zulässige Werte sind: • **Insert**: Beim Einfügen des Objektes werden Abhängigkeiten nicht beachtet. • **Insert Restrict**: Beim Einfügen des Objektes wird geprüft, ob das referenzierte Objekt existiert.
Prüfinstanz der Child-Beziehung	Gibt an, wer die Überprüfung dieser referentiellen Integrität ausführen soll. Zulässige Werte sind: • **DLL**: Die Überprüfung erfolgt durch die Objektschicht. • **Trigger**: Zur Überwachung durch die Datenbank werden Trigger und Constraints eingesetzt.
Transport nur im Verbund	Der Inhalt der Spalte wird bei Datentransporten immer zusammen mit den Inhalten der referenzierten Spalte übertragen.
Übergeordnetes Objekt in der Jobqueue	Gibt an, ob das übergeordnete Objekt in die Liste der von einem Prozess betroffenen Objekte aufgenommen werden soll. Damit kann verhindert werden, dass das übergeordnete Objekt mehrfach gleichzeitig verarbeitet wird.

6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Tabellenbeziehungen auf Seite 109
- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52

Unterstützung von Dateigruppen

One Identity Manager unterstützt Dateigruppen, um Tabellen zum Zweck der Verwaltung, Datenzuordnung und Datenverteilung zu Gruppen zusammenzufassen. Es werden dazu logische Speicherorte und physische Speicherorte unterschieden.

In der Standardinstallation sind für die Tabellen jedes Moduls des One Identity Manager sowie die Systemtabellen logische Speicherorte vordefiniert. Die Zuordnungen können Sie nicht ändern. Sie können zur Gruppierung kundenspezifischer Tabellen eigene logische Speicherorte erstellen.

Um logische Speicherorte für kundenspezifische Tabellen zu definieren

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema > Logische Speicherorte**.
2. Wählen Sie den Menüeintrag **Objekt > Neu**.
3. Erfassen Sie die Bezeichnung des logische Speicherortes und eine Beschreibung.
4. Weisen Sie dem logischen Speicherort die kundenspezifischen Tabellen zu.
5. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogTable`. In der Bearbeitungsansicht wird der Tabreiter **Tabellen** angezeigt, über welchen Sie die Tabellen zuweisen.

Logische Speicherorte können im One Identity Manager Schema mit physischen Speicherorten - den Dateigruppen - verbunden werden.

Werden beispielsweise für Tabellen mit Personendaten und für Tabellen mit Active Directory Inhalten Dateigruppen auf verschiedenen Datenträger erzeugt, kann durch parallelisierten Zugriff über eigene E/A-Controller die Performance erhöht werden. Werden beispielsweise die Tabellen zur Verarbeitung der DBQueue Prozessor Aufträge oder die Tabellen zur Prozessverarbeitung in Dateigruppen zusammengefasst, die auf einer SSD liegen, kann ebenfalls die Performance erhöht werden.

HINWEIS: Um die einwandfreie Funktion der One Identity Manager Datenbank zu gewährleisten, können die folgenden Tabellen nicht in andere Dateigruppen verschoben werden.

- DialogColumn
- DialogTable
- DialogValidDynamicRef
- QBMDBQueueTask

- QBMDBQueueTaskDepend
- QBMModuleDef
- QBMModuleDepend
- QBMRelation
- QBMViewAddOn
- QBMDiskStoreLogical
- QBMDiskStorePhysical

Der One Identity Manager unterstützt die Verteilung der Tabellen auf Dateigruppen mit einer Reihe von Datenbankprozeduren, die Sie in einem geeigneten Programm zur Ausführung von SQL Abfragen in der Datenbank ausführen.

VORSICHT: **Die nachfolgenden Schritte zum Einsatz von Dateigruppen sollten nur in Zusammenarbeit mit einem erfahrenen Datenbankadministrator ausgeführt werden.**

Stellen Sie sicher, dass während der Bereitstellung der Dateigruppen keine Zugriffe auf die Datenbank erfolgen, beispielsweise durch Jobserver, Anwendungsserver, Webserver, Frontends, Web Portal. Warten Sie nach dem Reaktivieren des DBQueue Prozessor bis alle Aufträge der DBQueue verarbeitet wurden, bevor Sie Verbindungen zur Datenbank wieder zulassen.

WICHTIG: Verwenden Sie für die Ausführung der SQL Abfragen den Benutzer, den Sie auch für die Migration der Datenbank verwenden.

Um Tabellen unter SQL Server auf Dateigruppen zu verteilen

1. Erstellen Sie Ihre Dateigruppen. Ausführliche Informationen dazu entnehmen Sie der Dokumentation Ihrer eingesetzten SQL Server Version.
2. Synchronisieren Sie die Dateigruppen in die One Identity Manager-Datenbank. Führen Sie dazu die folgende Abfrage in einem geeigneten Programm zur Ausführung von SQL Abfragen in der Datenbank aus.

   ```
   exec QBM_PDiskStorePhysicalSync
   ```

3. Weisen Sie im Designer die physischen Speicherorte an die logischen Speicherorte zu.
   a. Wählen Sie im Designer die Kategorie **One Identity Manager Schema > Logische Speicherorte**.
   b. Wählen Sie den logischen Speicherort und wählen Sie in der Ansicht **Eigenschaften** unter **Physischer Speicherort** die Dateigruppe.
   c. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.
4. Deaktivieren Sie die Verarbeitung der DBQueue Prozessor Aufträge und die Prozessverarbeitung. Führen Sie dazu die folgende Abfragen in einem geeigneten Programm zur Ausführung von SQL Abfragen in der Datenbank aus.

   ```
   exec QBM_PWatchDogPrepare 1
   ```

```
exec QBM_PDBQueuePrepare 1
```

5. Verschieben Sie die Tabellen in die konfigurierten Dateigruppen. Führen Sie dazu die folgende Abfrage in einem geeigneten Programm zur Ausführung von SQL Abfragen in der Datenbank aus.

   ```
   exec QBM_PTableMove
   ```

6. Reaktivieren Sie den DBQueue Prozessor. Führen Sie dazu die folgende Abfragen in einem geeigneten Programm zur Ausführung von SQL Abfragen in der Datenbank aus.

   ```
   exec QBM_PDBQueuePrepare 0,1
   ```

   ```
   exec QBM_PWatchDogPrepare
   ```

Bearbeiten der Benutzeroberfläche

Einige der Bestandteile der Benutzeroberfläche der One Identity Manager-Werkzeuge werden im One Identity Manager Schema abgelegt und können an die benutzerdefinierten Anforderungen angepasst werden. So können die Menüeinträge der Navigationsstruktur, die Oberflächenformulare und die Methodendefinitionen (Aufgaben) konfiguriert werden.

Menüeinträge, Oberflächenformulare und Methodendefinitionen werden den Berechtigungsgruppen zugewiesen. Die gültigen Bestandteile der Benutzeroberfläche eines Benutzers sind abhängig vom Authentifizierungsmodul, das für die Anmeldung an den One Identity Manager-Werkzeugen verwendet wird. Meldet sich ein Benutzer an einem One Identity Manager-Werkzeug an, so wird ein Systembenutzer ermittelt und abhängig von den Berechtigungsgruppen des Systembenutzers die verfügbaren Menüeinträge, Oberflächenformulare, Methodendefinitionen sowie einzelne Programmfunktionen ermittelt und die angepasste Benutzeroberfläche geladen.

Daten werden in der Benutzeroberfläche als Objekte dargestellt. Die Objekte der Benutzeroberfläche sind Meta-Objekte. Sie stellen eine Auswahl von konfigurierbaren Elementen zur Verfügung, die die Sichtweise der in der Datenbank hinterlegten Daten beschreiben. Diese Objekte erlauben es, die Daten nach bestimmten Eigenschaften zu unterscheiden. Die Objekte bieten somit eine zusätzliche Steuerfunktion zur Gestaltung der Benutzeroberfläche. So werden Oberflächenformulare und Methoden an Objektdefinitionen gebunden, um abhängig vom ausgewählten Objekt unterschiedliche Formulare und Methoden in der Benutzeroberfläche anzubieten.

Die mitgelieferten Oberflächenbestandteile sind begrenzt bearbeitbar und werden bei Schemainstallationen überschrieben. Sie können die Bestandteile der Standardbenutzeroberflächen in Ihre benutzerdefinierten Benutzeroberflächen integrieren. Bei Bedarf können Sie einzelne Bestandteile der Standardbenutzeroberflächen deaktivieren, um deren Anzeige zu verhindern. Die mitgelieferten Systembenutzer sind von dieser Einschränkung nicht betroffen. Die Deaktivierung der Bestandteile bleibt auch bei Schemainstallationen erhalten.

Um benutzerfreundliche Bezeichnungen für die verschiedenen Bestandteilen der Benutzeroberfläche, wie beispielsweise Menüeinträge, Methodendefinitionen oder Spaltennamen anzuzeigen, werden in der Oberflächengestaltung Anzeigetexte verwendet. Um die Bezeichnungen in unterschiedlichen Sprachen anzuzeigen, können Sie die Anzeigetexte im One Identity Manager sprachabhängig pflegen.

Die Standardinstallation des One Identity Manager wird in den Sprachen **English - United States [en-US]** und **German - Germany [de-DE]** ausgeliefert. Bei Bedarf können Sie weitere Sprachen zur Gestaltung der Benutzeroberfläche und der Ausgabetexte einpflegen.

Für diesen Fall müssen Sie vor Einsatz des One Identity Manager die verwendeten Texte in die neue Sprache übersetzen. Für die Übersetzung steht Ihnen im Designer einen Wörterbucheditor zur Verfügung. In den One Identity Manager-Werkzeugen wird die Eingabe mehrsprachiger Inhalte durch ein spezielles Steuerelement unterstützt.

Die Erstellung einer Benutzeroberfläche erfolgt immer für eine Anwendung. In der Standardauslieferung des One Identity Manager werden die Anwendungen und die vordefinierte Menüführung für die Werkzeuge Manager, Designer und Launchpad mitgeliefert.

Detaillierte Informationen zum Thema

- Objektdefinitionen für die Benutzeroberfläche auf Seite 119
- Menüführung der Benutzeroberfläche auf Seite 124
- Formulare für die Benutzeroberfläche auf Seite 153
- Statistiken im One Identity Manager auf Seite 190
- Erweitern des Launchpad auf Seite 208
- Methodendefinitionen für die Benutzeroberfläche auf Seite 212
- Anwendungen für die Gestaltung der Benutzeroberfläche auf Seite 218
- Symbole und Bilder für die Gestaltung der Benutzeroberfläche auf Seite 221
- Sprachabhängige Abbildung von Informationen auf Seite 233

Objektdefinitionen für die Benutzeroberfläche

Die Darstellung der Daten in der Benutzeroberfläche erfolgt über Objekte. Die Objekte der Benutzeroberfläche stellen eine Sicht auf die in der Datenbank hinterlegten Daten dar. Die Objekte sind konfigurierbar und erlauben es, die Daten nach bestimmten Eigenschaften zu unterscheiden.

Um abhängig vom ausgewählten Objekt unterschiedliche Formulare und Aufgaben in der Benutzeroberfläche anzubieten, werden Oberflächenformulare und Methodendefinitionen an Objektdefinitionen gebunden. Objektdefinitionen bieten somit eine zusätzliche Steuerfunktion zur Gestaltung der Benutzeroberfläche.

Zu jeder Tabelle des One Identity Manager Schemas können Sie mehrere Objektdefinitionen zuordnen. Grundsätzlich sollten Sie jedoch zu jeder Tabelle mindestens eine Objektdefinition mit allgemeiner Gültigkeit, das bedeutet, ohne einschränkende Auswahlkriterien, definieren. Weitere Objektdefinitionen beziehen sich dann auf den jeweiligen, den allgemeinen Fall einschränkenden, Sonderfall.

> TIPP: Um Objektdefinitionen für neue Tabellen zu erstellen, führen Sie im Designer die Konsistenzprüfung **Missing DialogObject** aus und verwenden Sie die Reparaturmethode. Die so erstellten Objektdefinitionen müssen Sie nachträglich weiter bearbeiten.

Tabelle 30: Beispiel für die Beziehung von Tabelle und Objektdefinitionen der Benutzeroberfläche

Tabelle	Objektdefinition	Einschränkung laut Objektdefinition
ESet	Systemrollen (`ESet`)	keine Einschränkung
ESet	Systemrollen für IT Shop (`ESet_ITShop`)	Systemrollen, die in den IT Shop ausgenommen werden können

Detaillierte Informationen zum Thema

- Auswahlkriterien für Objektdefinitionen auf Seite 120
- Verwenden der Anzeigetexte für Objektdefinitionen auf Seite 121
- Objektdefinitionen erstellen und bearbeiten auf Seite 122
- Eigenschaften von Objektdefinitionen auf Seite 122
- Auswirkung der Objektdefinitionen auf die Anzeige der Oberflächenformulare auf Seite 160

Auswahlkriterien für Objektdefinitionen

Die darzustellenden Einträge einer Tabelle werden über ein Auswahlskript und eine Bedingung der Objektdefinition ermittelt.

- Formulieren Sie ein Auswahlskript als VB.Net-Ausdruck, das entweder **Wahr** (`True`) oder **Falsch** (`False`) zurück gibt, je nachdem, ob der übergebene Datensatz zu dieser Objektdefinition gehört oder nicht.
- Formulieren Sie eine Bedingung als Where-Klausel als Datenbankabfrage, damit eine Objektdefinition auch für die Darstellung in Ergebnislisten genutzt werden kann.

WICHTIG: Damit die Auswahlkriterien wirksam werden, müssen Sie die Datenbank kompilieren.

Beispiel: Anzeige von Systemrollen für den IT Shop

Auswahlskript, um zur Laufzeit zu bestimmen, ob es sich bei diesem Datensatz um eine Systemrolle für den IT Shop handelt:

```
Value = $IsForITShop:Bool$
```

Bedingung (Where-Klausel), um zu ermitteln, ob es sich um eine Systemrolle für den IT Shop handelt:

```
IsForITShop=1
```

Verwandte Themen

- Objektdefinitionen erstellen und bearbeiten auf Seite 122
- Eigenschaften von Objektdefinitionen auf Seite 122

Verwenden der Anzeigetexte für Objektdefinitionen

Für jede Objektdefinition können Sie folgende Anzeigetexte zur Darstellung in der Benutzeroberfläche der Administrationswerkzeuge definieren.

- Anzeigetext der Liste

  Der Anzeigetext der Liste wird in den One Identity Manager-Werkzeugen für den Listentitel einer Ergebnisliste verwendet. Als Listentitel wird dabei der Anzeigetext derjenigen Objektdefinition verwendet, die Sie über die Listeneigenschaften des gewählten Menüeintrags festgelegt haben.
- Anzeigetext des Formulars

  Der Anzeigetext des Formulars wird zur Anzeige der wirksamen Objektdefinition selbst, beispielsweise in der Statuszeile des Manager, verwendet.

Bei Auswahl eines Eintrags in einer Ergebnisliste der Administrationswerkzeuge wird die wirksame Objektdefinition ermittelt. Anhand des Auswahlskriptes werden die gültigen Objektdefinitionen und somit die möglichen Anzeigetexte bestimmt. Aus den möglichen Anzeigetexten wird der Anzeigetext der Objektdefinition mit der kleinsten Sortierreihenfolge angezeigt.

Beispiel:

Tabelle 31: Anzeigetexte anhand der Sortierreihenfolge der Objektdefinitionen

Objektdefinition	Einschränkungen	Reihenfolge	Anzeigetext
Systemrollen (`ESet`)	keine Einschränkung	99	Systemrollen
Systemrollen für IT Shop (`ESet_ITShop`)	Systemrollen, die in den IT Shop ausgenommen werden können	8	Systemrollen für IT Shop

Bei der Auswahl einer Systemrolle in der Ergebnisliste lautet der zugehörige Anzeigetext **Systemrollen**. Ist die Systemrollen zusätzlich für den IT Shop gekennzeichnet (`IsForITShop=1`), dann wird dieses Objekt durch den VB.Net-

Ausdruck einer anderen Objektdefinition zugeordnet und der Anzeigetext **Systemrollen für IT Shop** verwendet.

Verwandte Themen

- Eigenschaften von Objektdefinitionen auf Seite 122
- Bearbeiten von Listen auf Seite 144

Objektdefinitionen erstellen und bearbeiten

Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar.

Um eine Objektdefinition zu erstellen oder zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Objektdefinitionen**.
2. Wählen Sie eine vorhandene Objektdefinition aus.

   - ODER -

   Fügen Sie eine neue Objektdefinition über den Menüeintrag **Objekt > Neu** ein.
3. Bearbeiten Sie die Stammdaten der Objektdefinition.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Kundenspezifische Änderungen an der One Identity Manager Standardkonfiguration auf Seite 28
- Eigenschaften von Objektdefinitionen auf Seite 122

Eigenschaften von Objektdefinitionen

Tabelle 32: Eigenschaften einer Objektdefinition

Eigenschaft	Beschreibung
Alleingültig	Objektdefinitionen, die mit dieser Option gekennzeichnet sind, gelten als exklusiv. Das bedeutet alle anderen eventuell zutreffenden Objektdefinitionen werden als nicht zutreffend angenommen. Sind mehrere Objektdefinitionen einer Tabelle als alleingültig gekennzeichnet, gilt die Objektdefinition mit

Eigenschaft	Beschreibung
	der niedrigsten Reihenfolge.
Anzeigemuster	Das Anzeigemuster legt fest, in welcher Form die Datensätze in der Ergebnisliste der Administrationswerkzeuge dargestellt werden.
Anzeigename	Der Anzeigename des Objektes wird beispielsweise in der Datenbanksuche oder der Fehlerausgabe zur Anzeige des Objektes verwendet. Den Anzeigenamen können Sie mehrsprachig angeben.
Anzeigetext der Liste	Anzeigetext für den Listentitel in der Benutzeroberfläche.
Anzeigetext des Formulars	Anzeigetext für den Formulartitel in der Benutzeroberfläche.
Auswahlskript	Auswahlskript als VB.Net-Ausdruck an, um zur Laufzeit zu bestimmen, ob das übergebene Datenbankobjekt zu dieser Objektdefinition gehört. HINWEIS: Die Anpassung des Auswahlskriptes erfordert die Kompilierung der Datenbank.
Bearbeitungsstatus	Bearbeitungsstatus des Objektes. Der Bearbeitungsstatus wird bei der Erstellung von Kundenkonfigurationspaketen genutzt.
Bedingung	Bedingung, damit die Objektdefinition auch für die Anzeige in Listen verwendet werden kann. Die Bedingung definieren Sie als gültige Where-Klausel für Datenbankabfragen. HINWEIS: Das Auswahlskript und die Bedingung müssen einander entsprechen. Wird eine der Eigenschaften gefüllt, ist die jeweils andere Eigenschaft ebenfalls zu füllen!
Bemerkungen	Freitextfeld für zusätzliche Erläuterungen.
Deaktiviert durch Präprozessor	Ist eine Objektdefinition durch eine Präprozessorbedingung ausgeschaltet, dann wird durch den Database Compiler die Option gesetzt.
Einzufügende Werte	Standardeinstellungen für Felder, die beim Einfügen eines neuen Datensatzes zugewiesen werden. Die Angabe erfolgt in VB.Net-Syntax. HINWEIS: Die Anpassung der Werte erfordert die Kompilierung der Datenbank.
Hintergrundfarbe	Farbe, mit der das Steuerelement für dieses Objekt in der Schemaübersicht angezeigt wird.
Objektname	Name des Objektes.

Eigenschaft	Beschreibung
Präprozessorbedingung	Objektdefinitionen können mit Präprozessorbedingungen versehen werden. Damit ist eine Objektdefinition nur verfügbar, wenn die Präprozessorbedingung erfüllt ist.
Reihenfolge	Die Reihenfolge dient zur Anzeige des Formulartitels bei der Auswahl eines Objektes. Je kleiner die Sortierreihenfolge desto stärker sind die Restriktionen für das Objekt definiert.
Symbol	Symbol für die Anzeige der Objektdefinition.
Tabelle	Tabelle, für welche die Objektdefinition erstellt wird.

Verwandte Themen

- Objektdefinitionen erstellen und bearbeiten auf Seite 122
- Auswahlkriterien für Objektdefinitionen auf Seite 120
- Verwenden der Anzeigetexte für Objektdefinitionen auf Seite 121
- Anzeigemuster für die Listendarstellung auf Seite 146
- Sprachabhängige Abbildung von Informationen auf Seite 233
- Definieren von Einfügewerten auf Seite 147
- Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369
- Symbole und Bilder für die Gestaltung der Benutzeroberfläche auf Seite 221

Menüführung der Benutzeroberfläche

Für die One Identity Manager-Werkzeuge mit eigener Benutzeroberfläche wird eine eigene Menüführung zur Verfügung gestellt. Durch die Menüführung werden gezielte Einstiegspunkte in die Oberflächennavigation der One Identity Manager-Werkzeuge vorgegeben und die anwenderorientierte Navigation bis hin zur Auswahl eines Objektes in der Ergebnisliste gesteuert. Der Aufbau der Menüführung der Benutzeroberflächen wird über Menüeinträge realisiert. Es gibt unterschiedliche Arten von Menüeinträgen mit definiertem Einsatzzweck. Durch die Kombination der verschiedenen Arten von Menüeinträgen können vielfältige Menüführungen entwickelt werden.

Die Menüführung wird im Designer in der Kategorie **Benutzeroberfläche > Menüführung** abgebildet und bearbeitet. Die Verfügbarkeit und Bearbeitbarkeit der Eigenschaften richten sich nach der Eintragsart der Menüeinträge.

Detaillierte Informationen zum Thema

- Bestandteile einer Menüführung auf Seite 125
- Empfehlungen zur Bearbeitung der Menüführung auf Seite 127

- Hinweise zum Arbeiten mit dem Oberflächeneditor auf Seite 128
- Auswählen der Menüführung zur Bearbeitung auf Seite 128
- Menüführung während der Bearbeitung simulieren auf Seite 132
- Bestehende Menüführung für eine neue Berechtigungsgruppe kopieren
- Menüeinträge kopieren auf Seite 135
- Neue Menüführung erstellen auf Seite 134
- Neue Menüeinträge erstellen auf Seite 136
- Neue Menükategorien erstellen auf Seite 137
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139
- Erstellen der Datenbankabfrage für datenabhängige Menüeinträge auf Seite 142
- Bearbeiten von Listen auf Seite 144
- Verwenden von Links in der Menüführung auf Seite 148
- Verwenden von Variablen in der Menüführung auf Seite 149

Bestandteile einer Menüführung

Tabelle 33: Arten von Menüeinträgen

Art	Beschreibung
Menükategorie	Menükategorien werden in der oberste Ebene der Menüführung dargestellt und dienen zur Gruppierung der verwalteten Daten nach definierten Gesichtspunkten. Menükategorien bilden die Einstiegspunkte in die Oberflächennavigation. In der Benutzeroberfläche werden die Menükategorien als Kategorie dargestellt.
Fixer Menüeintrag	Fixe Menüeinträge werden zur übersichtlicheren Gliederung innerhalb der Menükategorien verwendet. Diese Menüeinträge werden in der Menüführung immer angezeigt. Für fixe Menüeinträge können Listeneigenschaften definiert werden. Diese bestimmen die Darstellung der Tabelleneinträge in der Ergebnisliste der Benutzeroberfläche.
Datenabhängiger Menüeintrag	Datenabhängige Menüeinträge werden aufgrund einer Datenbankabfrage generiert, welche als Ergebnismenge mehrere Datensätze zurückliefert. Diese Menüeinträge sind somit keine einzelnen Menüeinträge, sondern je nach Ergebnismenge der Datenbankabfrage, eine Menge von Menüeinträgen. Für datenabhängige Menüeinträge können Listeneigenschaften definiert werden. Diese bestimmen die Darstellung der Tabelleneinträge in der Ergebnisliste der Benutzeroberfläche.

Art	Beschreibung
Freier Menüeintrag	Freie Menüeinträge werden zur Gruppierung anderer Menüeinträge oder zur Definition eines Startmenüeintrags einer Anwendung eingesetzt. Somit kann beispielsweise die Startseite ein Webfrontends festgelegt werden. Freie Menüeinträge sollten immer in der obersten Ebene einer Menüführung erstellt werden. Sie werden jedoch selbst nicht in der Navigationsansicht der Administrationswerkzeuge angezeigt.
Link	Links unterstützen den Aufbau einer Menüführung. Sie werden eingesetzt, um häufig verwendete Menüeinträge zu referenzieren. Teile der Menüführung, die in der gesamten Menüführung einer Anwendung mehrfach benötigt werden, müssen somit nur einmal erstellt werden. Links werden in der Menüführung selbst nicht angezeigt, sondern es werden immer die referenzierten Menüeinträge dargestellt.
Zentrales Formularelement	Zentrale Formularelemente stellen keinen Menüeintrag in der Navigationsstruktur dar, sondern werden verwendet, um das zentrale Element auf Überblickformularen zu Objekten zu erstellen. Alle untergeordneten Menüeinträge werden um dieses zentrale Formularelement angeordnet.
Aufgabenkategorie	Aufgabenkategorien werden in der obersten Ebene der Menüführung dargestellt und dienen zur Zusammenfassung handlungsorientierter Abläufe. Die Abbildung der Aufgabenkategorien erfolgt in den Administrationswerkzeugen nicht in der Navigationsansicht sondern über ein spezielles Formular.
Aufgabe	Aufgaben dienen zur Abbildung von Einzelaufgaben innerhalb einer Aufgabenkategorie. Sie werden beispielsweise als Startpunkte für Assistenten in der Oberfläche der Administrationswerkzeuge eingesetzt. Aufgaben werden immer unter einem Menüeintrag einer Aufgabenkategorie angeordnet. Die Abbildung der Aufgabenkategorien und ihrer Aufgaben erfolgt in den Administrationswerkzeugen nicht in der Navigationsansicht sondern über ein spezielles Formular.
Statistik	Diese Menüeinträge werden zur Darstellung von Statistiken verwendet. Typischerweise werden Statistiken im Infosystem angezeigt. Alle Statistiken, die in einer Menüebene definiert sind, können auf einem Formular oder als einzelne Menüeinträge angezeigt. Statistiken können zusätzlich als Formularelemente eingebunden werden.

Verwandte Themen

- Empfehlungen zur Bearbeitung der Menüführung auf Seite 127
- Hinweise zum Arbeiten mit dem Oberflächeneditor auf Seite 128
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

- Erstellen der Datenbankabfrage für datenabhängige Menüeinträge auf Seite 142
- Bearbeiten von Listen auf Seite 144
- Verwenden von Links in der Menüführung auf Seite 148
- Bearbeiten von Überblickformularen auf Seite 179
- Einbinden von Statistiken in die Benutzeroberfläche auf Seite 196

Empfehlungen zur Bearbeitung der Menüführung

- Für fixe Menüeinträge und datenbankhängige Menüeinträge können Sie Listeneigenschaften, wie beispielsweise Anzeigemuster oder zu verwendende Objektdefinition festlegen. Diese Eigenschaften bestimmen die Darstellung der Tabelleneinträge in der Ergebnisliste der Benutzeroberfläche.

  > TIPP: Anzeigemuster können Sie für die Listen der Menüeinträge, Objektdefinitionen und Tabellen definieren.
  >
  > Die Ermittlung des zu verwendenden Anzeigemusters erfolgt in der Reihenfolge:
  >
  > 1. Anzeigemuster der Liste des Menüeintrags
  > 2. Anzeigemuster der Objektdefinition
  > 3. Anzeigemuster der Tabelle

- Erstellen Sie Menüeinträge, die Sie als Verweise (Links) verwenden können. Teile der Menüführung, die in der gesamten Menüführung einer Anwendung mehrfach benötigt werden, müssen somit Sie nur einmal erstellen. Links werden in der Menüführung selbst nicht angezeigt, sondern es werden immer die referenzierten Menüeinträge dargestellt.
- Verwenden Sie Variablen, die Sie bei der Gestaltung der Bezeichnungen und der Anzeigemuster von Menüeinträgen, in Einfügewerten und in Datenbankabfragen einsetzen.

  > TIPP: Definieren Sie benötigte Variablen bereits am Menüeintrag für die Menükategorie. Innerhalb der hierarchischen Menüführung werden Variablen vererbt. Somit sind Variablen in tieferen Ebenen einer Hierarchie wiederverwendbar oder überschreibbar. Zur Laufzeit wird der aktuelle Wert an die Variablen übergeben.

- Zur Darstellung der Menüeinträge in der Benutzeroberfläche weisen Sie die Menüeinträge an die Anwendung, beispielsweise **Manager**, zu.
- Weisen Sie die Menüeinträge an die Berechtigungsgruppen für die nicht-rollenbasierte Anmeldung und die rollenbasierte Anmeldung zu.

Verwandte Themen

- Bestandteile einer Menüführung auf Seite 125
- Menüeinträge an Anwendungen zuweisen auf Seite 137

- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Bearbeiten von Listen auf Seite 144
- Verwenden von Links in der Menüführung auf Seite 148
- Verwenden von Variablen in der Menüführung auf Seite 149

Hinweise zum Arbeiten mit dem Oberflächeneditor

Mit dem Oberflächeneditor bearbeiten Sie die Menüführung der One Identity Manager-Werkzeuge. In der Übersicht der Menüführung werden die alle Menüeinträge hierarchisch dargestellt.

- Um eine Vorauswahl der Menüeinträge zur weiteren Bearbeitung zu treffen, verwenden Sie den Auswahlassistenten des Oberflächeneditors.
- Per Drag and Drop können Sie im Menüeinträge innerhalb der Hierarchie verschieben.
- Über das Menü **Optionen > Beschriftungen anzeigen** wechseln Sie zwischen den technischen Bezeichnungen der Menüeinträge und den benutzerfreundlichen Anzeigetexten.
- Über das Menü **Optionen > Menümarkierungen erstellen** können Sie Menüeinträge markieren. Über einen Where-Klausel-Assistenten legen Sie die Menüeinträge fest. Diese werden in der Übersicht der Menüführung rot markiert. Über das Menü **Optionen > Menümarkierungen löschen** entfernen Sie die Markierung.
- Über das Menü **Optionen > Spalten wählen** können Sie zusätzliche Spalten in der Übersicht der Menüführung anzeigen.
- Nutzen Sie den Simulationsmodus, um während der Bearbeitung die Menüführung zu simulieren.

Verwandte Themen

- Auswählen der Menüführung zur Bearbeitung auf Seite 128
- Menüführung während der Bearbeitung simulieren auf Seite 132

Auswählen der Menüführung zur Bearbeitung

Um die Menüführung zur Bearbeitung auszuwählen haben Sie verschiedene Möglichkeiten. Sie können die komplette Menüführung laden, die Menüführung einer einzelnen

Anwendung auswählen oder über den Auswahlassistenten des Oberflächeneditors eine Vorauswahl der Menüeinträge treffen.

Detaillierte Informationen zum Thema

- Komplette Menüführung laden auf Seite 129
- Menüführung über eine Anwendung laden auf Seite 129
- Menüeinträge direkt laden auf Seite 130
- Menüeinträge über Berechtigungsgruppen laden auf Seite 130
- Menüführung über eine Where-Klausel laden auf Seite 131

Komplette Menüführung laden

Über diese Aufgabe laden Sie die komplette Menüführung zur Bearbeitung.

Um die komplette Menüführung zu laden

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Aufgabe **Menüführung bearbeiten.**

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.

Verwandte Themen

- Menüführung über eine Anwendung laden auf Seite 129
- Menüeinträge direkt laden auf Seite 130
- Menüeinträge über Berechtigungsgruppen laden auf Seite 130
- Menüführung über eine Where-Klausel laden auf Seite 131

Menüführung über eine Anwendung laden

Mit dieser Aufgabe laden Sie die Menüführung einer Anwendung. In der Standardauslieferung des One Identity Manager werden die Anwendungen und die vordefinierte Menüführung für die Werkzeuge Manager, Designer und Launchpad mitgeliefert.

Um die Menüführung einer Anwendung zu laden

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Anwendung **Manager**, **Designer** oder **Launchpad**.
3. Wählen Sie die Aufgabe **Menüführung der Anwendung bearbeiten**.

Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.

Verwandte Themen

- Komplette Menüführung laden auf Seite 129
- Menüeinträge direkt laden auf Seite 130
- Menüeinträge über Berechtigungsgruppen laden auf Seite 130
- Menüführung über eine Where-Klausel laden auf Seite 131

Menüeinträge direkt laden

Über diese Aufgabe wählen Sie die Menüeinträge, die Sie bearbeiten möchten, direkt im Auswahlassistenten des Oberflächeneditors aus.

Um Menüeinträge direkt auszuwählen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Aufgabe **Auswahlassistent zum Bearbeiten der Menüführung starten**.
3. Auf der Startseite des Assistenten klicken Sie **Weiter**.
4. Auf der Seite **Lademethode wählen** klicken Sie 🗁.
5. Auf der Seite **Auswahl der Menüführung** werden die Menüführungen alle Anwendungen aus der Datenbank angezeigt. Aktivieren Sie die Menüeinträge, die Sie laden möchten.
6. Um den Assistenten zu beenden, klicken Sie **Fertig**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.

Verwandte Themen

- Komplette Menüführung laden auf Seite 129
- Menüführung über eine Anwendung laden auf Seite 129
- Menüeinträge über Berechtigungsgruppen laden auf Seite 130
- Menüführung über eine Where-Klausel laden auf Seite 131

Menüeinträge über Berechtigungsgruppen laden

Über diese Aufgabe wählen Sie die Menüeinträge, die Sie bearbeiten möchten, im Auswahlassistenten des Oberflächeneditors über Berechtigungsgruppen aus.

Um Menüeinträge über Berechtigungsgruppen zu laden

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Aufgabe **Auswahlassistent zum Bearbeiten der Menüführung starten**.
3. Auf der Startseite des Assistenten klicken Sie **Weiter**.
4. Auf der Seite **Lademethode wählen** klicken Sie .
5. Auf der Seite **Auswahl über die Berechtigungsgruppe** wählen Sie die Berechtigungsgruppen deren Menüeinträge geladen werden.

   Sie können die Einschränkung der Berechtigungsgruppen über einen bestimmten Systembenutzer treffen oder die zu Berechtigungsgruppen direkt wählen.
6. Um den Assistenten zu beenden, klicken Sie **Fertig**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.

Verwandte Themen

- Komplette Menüführung laden auf Seite 129
- Menüführung über eine Anwendung laden auf Seite 129
- Menüeinträge direkt laden auf Seite 130
- Menüführung über eine Where-Klausel laden auf Seite 131

Menüführung über eine Where-Klausel laden

Über diese Aufgaben wählen Sie die Menüeinträge, die Sie bearbeiten möchten, im Auswahlassistenten des Oberflächeneditors über eine Where-Klausel aus.

Um die Menüführung über eine Where-Klausel zu laden

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Aufgabe **Auswahlassistent zum Bearbeiten der Menüführung starten**.
3. Auf der Startseite des Assistenten klicken Sie **Weiter**.
4. Auf der Seite **Lademethode wählen** klicken Sie .
5. Auf der Seite **Benutzerdefinierte Auswahl** erfassen Sie die Where-Klausel zum Laden der Menüführung. Geben Sie die Where-Klausel manuell ein oder nutzen Sie den Where-Klausel-Assistenten.
6. Um den Assistenten zu beenden, klicken Sie **Fertig**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.

Verwandte Themen

- Komplette Menüführung laden auf Seite 129
- Menüführung über eine Anwendung laden auf Seite 129
- Menüeinträge direkt laden auf Seite 130
- Menüeinträge über Berechtigungsgruppen laden auf Seite 130

Menüführung während der Bearbeitung simulieren

Über die Simulation der Menüführung im Oberflächeneditor sehen Sie für einen Systembenutzer, welche Menüeinträge er aufgrund seiner Berechtigungsgruppe in einer Anwendung sieht.

Um die Menüführung einer Anwendung zu simulieren

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
2. Legen Sie die Simulationsdaten fest.
   a. Wählen Sie im Oberflächeneditor das Menü **Simulation > Simulationsdaten festlegen**.
   b. Auf der Startseite des Assistenten klicken Sie **Weiter**.
   c. Auf der Seite **Simulationsdaten festlegen** legen Sie folgende Einstellungen fest.
      - **Benutzer für die Simulation**: Wählen Sie den Systembenutzer, für den die Menüführung simuliert werden soll.
      - **Anwendung für die Simulation**: Wählen Sie die Anwendung, für die die Menüführung simuliert werden soll.
   d. Auf der Seite **Basisobjekt wählen** klicken Sie **Weiter**.
   e. Um den Assistenten zu beenden, klicken Sie auf der letzten Seite **Fertig**.
3. Starten Sie die Simulation über das Menü **Simulation > Simulation starten**.

   Die Anwendung wird im Simulationsfenster geöffnet.

HINWEIS: Sie können die Simulation jederzeit beenden, indem Sie das Simulationsfenster schließen. Über **F9** können Sie die Simulation wieder starten. Die Simulationsdaten (Systembenutzer und Anwendung) bleiben erhalten.

Bestehende Menüführung für eine neue Berechtigungsgruppe kopieren

Mit dem Auswahlassistenten des Oberflächeneditors können Sie die Menüeinträge einer Berechtigungsgruppe für eine andere Berechtigungsgruppe kopieren. Optional können Sie mit dem Assistenten auch die benötigten Berechtigungen auf Tabellen und Spalten sowie die Objektdefinitionen und Methodendefinitionen für die Berechtigungsgruppe übernehmen.

Um eine bestehende Menüführung zu kopieren

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Aufgabe **Auswahlassistent zum Bearbeiten der Menüführung starten**.
3. Auf der Startseite des Assistenten klicken Sie **Weiter**.
4. Auf der Seite **Lademethode wählen** klicken Sie .
5. Auf der Seite **Auswahl über die Berechtigungsgruppe** wählen Sie die Berechtigungsgruppen deren Menüeinträge kopiert werden.

   Sie können die Einschränkung der Berechtigungsgruppen über einen bestimmten Systembenutzer treffen oder die zu Berechtigungsgruppen direkt wählen.
6. Auf der Seite **Zielberechtigungsgruppe definieren** erfassen Sie folgende Informationen.
   - **Kopieren auf (neue) Gruppe**: Berechtigungsgruppe, auf die die einzelnen Bestandteile der Menüführung kopiert werden.
     - Geben Sie den Namen der neuen Berechtigungsgruppe an. Beachten Sie, dass eigene Berechtigungsgruppen mit dem Kundenpräfix beginnen.
     - Wählen Sie eine vorhandene Berechtigungsgruppe aus.
   - **Namenspräfix/-suffix**: Erweiterte Kennzeichnung der Menüeinträge. Zur Bildung der Bezeichnungen für die neuen Menüeinträge ist mindestens ein Namenspräfix erforderlich. Das Namenspräfix ist **CCC**. Erfassen Sie optional ein Namenssuffix.
7. (Optional) Wählen Sie die Kopieroptionen.
   - **Spaltenberechtigungen kopieren**: Die Spaltenberechtigungen der Berechtigungsgruppe werden kopiert.
   - **Tabellenberechtigungen kopieren**: Die Tabellenberechtigungen der Berechtigungsgruppe werden kopiert.
   - **Zuordnungen der Oberflächenformulare kopieren**: Die Oberflächenformulare der Berechtigungsgruppe werden kopiert.
   - **Methodenzuordnungen kopieren**: Die Methodendefinitionen der Berechtigungsgruppe werden kopiert.
8. Um die Kopie zu starten, klicken Sie **Weiter**.

9. Auf der Seite **Menüdaten kopieren** werden die kopierten Bestandteile werden angezeigt. Je nach Umfang der ausgewählten Bestandteile kann die anschließende Kopie einige Zeit in Anspruch nehmen. Die kopierten Bestandteile werden angezeigt. Nach Abschluss des Kopiervorgangs, klicken Sie **Weiter**.
10. Um den Assistenten zu beenden, klicken Sie **Fertig**.

    Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung bereitgestellt.

HINWEIS: Nach dem Einfügen, Bearbeiten oder Löschen eines Menüeintrages müssen Sie die Datenbank kompilieren.

Verwandte Themen

- Neue Menüführung erstellen auf Seite 134
- Menüeinträge an Anwendungen zuweisen auf Seite 137
- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

Neue Menüführung erstellen

Mit dieser Aufgabe erstellen Sie mit dem Auswahlassistenten des Oberflächeneditors eine neue Menüführung mit einer initialen Menükategorie.

Um eine neue Menüführung zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Aufgabe **Auswahlassistent zum Bearbeiten der Menüführung starten**.
3. Auf der Startseite des Assistenten klicken Sie **Weiter**.
4. Auf der Seite **Lademethode wählen** klicken Sie .
5. Um den Assistenten zu beenden, klicken Sie **Fertig**.
6. Bearbeiten Sie die Stammdaten der Menükategorie. Erfassen Sie mindestens die Bezeichnung des Menüeintrags.
7. Weisen Sie eine Anwendung und die Berechtigungsgruppen zu.
8. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

HINWEIS: Nach dem Einfügen, Bearbeiten oder Löschen eines Menüeintrages müssen Sie die Datenbank kompilieren.

Verwandte Themen

- Bestehende Menüführung für eine neue Berechtigungsgruppe kopieren auf Seite 133
- Neue Menükategorien erstellen auf Seite 137
- Menüeinträge an Anwendungen zuweisen auf Seite 137
- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

Menüeinträge kopieren

Mit dieser Aufgabe kopieren Sie einen Menüeintrag einer Menüführung und fügen ihn an einer anderen Stelle der Menüführung ein.

Um einen Menüeintrag zu kopieren und einzufügen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
2. Wählen Sie die Anwendung **Manager**, **Designer** oder **Launchpad**.
3. Wählen Sie die Aufgabe **Menüführung der Anwendung bearbeiten**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
4. Wählen Sie in der Übersicht der Menüführung den Menüeintrag, den Sie kopieren möchten.
5. Wählen Sie im Kontextmenü eine der Kopieroptionen.
   - **Kopieren**: Wählen Sie diese Kopieroption, um den gewählten Menüeintrag zu kopieren.
   - **Kopieren mit Untereinträgen**: Wählen Sie diese Kopieroption, um den gewählten Menüeintrag und die ihm untergeordneten Menüeinträge zu kopieren.
6. Wählen Sie in der Übersicht der Menüführung den Menüeintrag, unter dem Sie den neuen Menüeintrag einfügen möchten.
7. Wählen Sie das Kontextmenü **Einfügen**.
8. Bearbeiten Sie die Stammdaten des Menüeintrags.
9. Weisen Sie eine Anwendung und die Berechtigungsgruppen zu.
10. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

HINWEIS: Nach dem Einfügen, Bearbeiten oder Löschen eines Menüeintrages müssen Sie die Datenbank kompilieren.

Verwandte Themen

- Auswählen der Menüführung zur Bearbeitung auf Seite 128
- Neue Menüeinträge erstellen auf Seite 136
- Neue Menükategorien erstellen auf Seite 137
- Menüeinträge an Anwendungen zuweisen auf Seite 137
- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

Neue Menüeinträge erstellen

Mit dieser Aufgabe erstellen Sie einen neuen Menüeintrag in einer bestehenden Menüführung.

Um einen neuen Menüeintrag zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
2. Wählen Sie in der Übersicht der Menüführung den Menüeintrag, unter dem Sie den neuen Menüeintrag erstellen möchten.
3. Wählen Sie das Kontextmenü **Neu**.
4. Bearbeiten Sie die Stammdaten des Menüeintrag.
5. Weisen Sie eine Anwendung und die Berechtigungsgruppen zu.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

HINWEIS: Nach dem Einfügen, Bearbeiten oder Löschen eines Menüeintrages müssen Sie die Datenbank kompilieren.

Verwandte Themen

- Auswählen der Menüführung zur Bearbeitung auf Seite 128
- Neue Menüführung erstellen auf Seite 134
- Neue Menükategorien erstellen auf Seite 137
- Menüeinträge kopieren auf Seite 135
- Menüeinträge an Anwendungen zuweisen auf Seite 137
- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

Neue Menükategorien erstellen

Mit dieser Aufgabe erstellen Sie eine neue Menükategorie in einer bestehenden Menüführung.

Um eine neue Kategorie zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
2. Wählen Sie das Menü **Menüeintrag > Menükategorie erstellen**.
3. Bearbeiten Sie die Stammdaten der Menükategorie. Erfassen Sie mindestens die Bezeichnung des Menüeintrags.

   HINWEIS: Soll der Eintrag eine Aufgabenkategorie repräsentieren, ändern Sie die Eintragsart des Menüeintrags auf **Aufgabenkategorie**.
4. Weisen Sie eine Anwendung und die Berechtigungsgruppen zu.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

HINWEIS: Nach dem Einfügen, Bearbeiten oder Löschen eines Menüeintrages müssen Sie die Datenbank kompilieren.

Verwandte Themen

- Auswählen der Menüführung zur Bearbeitung auf Seite 128
- Neue Menüeinträge erstellen auf Seite 136
- Neue Menüführung erstellen auf Seite 134
- Menüeinträge an Anwendungen zuweisen auf Seite 137
- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

Menüeinträge an Anwendungen zuweisen

Alle Menüeinträge, die in der Benutzeroberfläche einer Anwendung dargestellt werden sollen, müssen Sie der Anwendung zuweisen.

Um einen Menüeintrag an eine Anwendung zuzuweisen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
2. Wählen Sie in der Übersicht der Menüführung den Menüeintrag.

3. Wählen Sie in der Bearbeitungsansicht die Ansicht **Anwendung** und wählen Sie die Anwendung.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

TIPP: Über das Kontextmenü **Anwendung rekursiv zuweisen** weisen Sie die Anwendung dem gewählten Menüeintrag und seinen untergeordneten Menüeinträgen zu. Über das Kontextmenü **Anwendung rekursiv entfernen** entfernen Sie die Zuweisung der Anwendung vom gewählten Menüeintrag und seinen untergeordneten Menüeinträgen.

Verwandte Themen

- Menüeinträge an Berechtigungsgruppen zuweisen auf Seite 138
- Anwendungen für die Gestaltung der Benutzeroberfläche auf Seite 218

Menüeinträge an Berechtigungsgruppen zuweisen

Alle Menüeinträge, die in der Benutzeroberfläche einer Anwendung dargestellt werden sollen, müssen Sie einer Berechtigungsgruppe zuweisen. Weisen Sie die Menüeinträge an Berechtigungsgruppen für die nicht-rollenbasierte Anmeldung und die rollenbasierte Anmeldung zu. Die Menüeinträge werden den Systembenutzern abhängig von ihren Mitgliedschaften in Berechtigungsgruppen zur Verfügung gestellt. Ausführliche Informationen zu Berechtigungsgruppen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Um einen Menüeintrag an eine Berechtigungsgruppe zuzuweisen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
2. Wählen Sie in der Übersicht der Menüführung den Menüeintrag.
3. Wählen Sie in der Bearbeitungsansicht die Ansicht **Berechtigungsgruppe** und wählen Sie die Berechtigungsgruppen.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

TIPP: Über das Kontextmenü **Berechtigungsgruppe rekursiv zuweisen** weisen Sie die Berechtigungsgruppe dem gewählten Menüeintrag und seinen untergeordneten Menüeinträgen zu. Über das Kontextmenü **Berechtigungsgruppe rekursiv entfernen** entfernen Sie die Zuweisung der Berechtigungsgruppe vom gewählten Menüeintrag und seinen untergeordneten Menüeinträgen.

Verwandte Themen

- Menüeinträge an Anwendungen zuweisen auf Seite 137

Allgemeine Eigenschaften der Menüeinträge

Die nachfolgend beschriebenen Eigenschaften sind für alle Menüeinträge gültig. Für die unterschiedlichen Menüeintragsarten können weitere Eigenschaften erforderlich sein.

Tabelle 34: Allgemeine Eigenschaften von Menüeinträgen

Eigenschaft	Beschreibung
Menüeintrag	Unikale Bezeichnung des Menüeintrags. Sie sollten hier möglichst sprechende Namen vergeben, die sich in den untergeordneten Strukturen fortsetzen. Damit verbessert sich die Nachvollziehbarkeit der Position der untergeordneten Menüeinträge. Mit der Einfügeposition in die Menüführung wird der übergeordnete Menüeintrag bestimmt und die Hierarchie aufgebaut. Zur Abbildung der Menüeinträge kann der Name des Menüeintrags Variablen enthalten.
Eintragsart	Eintragsart des Menüeintrags.
Anzeigetext	Sprachabhängiger Anzeigetext, mit dem dieser Menüeintrag in der Benutzeroberfläche angezeigt wird. Der Anzeigetext datenabhängiger Menüeinträge kann feste Zeichenketten und Variablen enthalten. Der Anzeigetext rekursiver datenabhängiger Menüeinträge wird vom übergeordneten Menüeintrag geerbt. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Reihenfolge	Besitzen mehrere Menüeinträge denselben übergeordneten Menüeintrag, so legt die Reihenfolge der einzelnen Menüeinträge deren Position in der Darstellungsreihenfolge fest. Ist der Konfigurationsschalter **Nachsortierung der untergeordneten Menüeinträge nach Anzeigetext** beim übergeordneten Menüeintrag gesetzt, dann ist die hier angegebene Reihenfolge unwirksam.
Symbol	Symbol für die Anzeige des Menüeintrags in der Menüführung. Ist für rekursive datenabhängige Menüeinträge kein Symbol angegeben, wird das Symbol vom übergeordneten Menüeintrag geerbt.
Overlay-Symbol Definition	VB.Net-Ausdruck zur Definition von Overlays für das Symbol. Wird verwendet für die Statusanzeigen im Launchpad.
Bedingung	Gibt an, unter welchen Bedingungen der Menüeintrag in der

Eigenschaft	Beschreibung
	Menüführung angezeigt wird. Die Angabe muss der Where-Klausel-Syntax von Datenbankabfragen genügen. Bei der Formulierung einer Bedingung können Sie Variablen verwenden.
Konfigurationsschalter	Mit dem Konfigurationsschalter werden spezielle Funktionen für einen Menüeintrag eingestellt. Weitere Informationen finden Sie unter Tabelle 35 auf Seite 140.
Präprozessorbedingung	Menüeinträge können Sie mit Präprozessorbedingungen versehen. Damit ist ein Menüeintrag nur verfügbar, wenn die Präprozessorbedingung erfüllt ist. HINWEIS: Einen Überblick über die vorhandenen Präprozessorabhängigkeiten erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Präprozessorabhängigkeiten**.
Deaktiviert durch Präprozessor	Ist ein Menüeintrag durch eine Präprozessorbedingung ausgeschaltet, dann wird durch den Database Compiler die Option gesetzt.
Beschreibung	Freitextfeld für zusätzliche Erläuterungen.
Deaktiviert	Gibt an, ob ein Menüeintrag in der Benutzeroberfläche angezeigt wird. Deaktivierte Menüeinträge werden nicht in der Benutzeroberfläche angezeigt. HINWEIS: Diese Änderung ist auch bei Menüeinträgen der Standardoberflächen zulässig und bleibt bei Schemainstallationen bestehen.
Anzeigen unter "Mein One Identity Manager"	Mit dieser Option kennzeichnen Sie Menüeinträge, die im Manager in der Kategorie **Mein One Identity Manager** angezeigt werden sollen.

Tabelle 35: Konfigurationsschalter für spezielle Funktionen

Konfigurationsschalter	Beschreibung
Automatische Aktualisierung nach Einfügen	Ist dieser Konfigurationsschalter gesetzt, wird nach dem Einfügen eines neuen Datensatzes der Menüeintrag erneut geladen und aktualisiert.
Bei leerer Menge nicht anzeigen	Werden zur Laufzeit keine untergeordneten Menüeinträge zu einem derart gekennzeichneten Menüeintrag generiert, so wird der Menüeintrag in der Benutzeroberfläche ebenfalls nicht dargestellt.
Benutzer darf Menüeintrag nicht öffnen	Menüeinträge, die mit dieser Option gekennzeichnet sind, können nicht geöffnet werden, auch wenn es untergeordnete Menüeinträge gibt. Dieser Konfigurationsschalter

Konfigurationsschalter	Beschreibung
	wird hauptsächlich im Infosystem zur Anzeige der Statistiken verwendet.
Formulare der Ergebnisliste ignorieren	Für Menüeinträge, die mit dieser Option gekennzeichnet sind, werden keine Formulare in einer Ergebnisliste angeboten. Dies kann eingesetzt werden, um auf einem Überblickformular die Navigation zu den Objekten einer Listendarstellung zu verhindern. Dies bietet sich beispielsweise an, wenn für die einzelnen Objekte der Ergebnisliste keine Formulare definiert sind. Anderenfalls wird ein leeres Formular angezeigt.
Keine Oberflächenformulare laden	Der Konfigurationsschalter kann für datenabhängige Menüeinträge eingesetzt werden. Ist dieser Konfigurationsschalter aktiviert, so werden bei Auswahl des Menüeintrags in der Benutzeroberfläche keine objektabhängige Oberflächenformulare angezeigt. Dieser Konfigurationsschalter wird hauptsächlich beim Aufbau der Benutzeroberfläche für Webfrontends Verwendung finden.
Menüeintrag immer aktivieren	Ist dieser Konfigurationsschalter gesetzt, wird der Menüeintrag immer geöffnet. Es erfolgt keine Prüfung, ob dem Menüeintrag etwas zugewiesen ist, beispielsweise Oberflächenformulare.
Nachsortierung des datenabhängigen Menüeintrags nach Anzeigetext	Der Konfigurationsschalter kann für datenabhängige Menüeinträge eingesetzt werden. Der Konfigurationsschalter sollte gesetzt werden, wenn sprachabhängige Daten angezeigt werden. Ist der Konfigurationsschalter aktiviert, werden nach dem Laden die anzuzeigenden Daten für die Menüführung sprachabhängig sortiert.
Nachsortierung der Ergebnismenge nach Anzeigetext	Der Konfigurationsschalter kann für Listen eingesetzt werden. Der Konfigurationsschalter sollte gesetzt werden, wenn sprachabhängige Daten angezeigt werden. Ist der Konfigurationsschalter aktiviert, werden nach dem Laden die anzuzeigenden Daten in der Ergebnisliste sprachabhängig sortiert.
Nachsortierung der untergeordneten Menüeinträge nach Anzeigetext	Der Konfigurationsschalter sollte gesetzt werden, wenn sprachabhängige Daten angezeigt werden. Ist der Konfigurationsschalter aktiviert, werden nach dem Laden die anzuzeigenden Daten aller untergeordneten Menüeinträge sprachabhängig sortiert. Damit können beispielsweise innerhalb einer Containerstruktur alle Benutzerkonten, Gruppen und Container alphabetisch sortiert angezeigt werden. Die Sortierung

Konfigurationsschalter	Beschreibung
	wirkt nicht nur auf datenabhängige Menüeinträge sondern auf alle untergeordneten Menüeinträge.
Navigationskontext bei der Aktivierung berücksichtigen	Ist dieser Konfigurationsschalter gesetzt, wird beim Laden der Objekte berücksichtigt, aus welcher Stelle der Navigation der Menüeintrag geöffnet wird. Ist ein Objekt mehrfach innerhalb einer Navigationsstruktur vorhanden, wird der Inhalt abhängig von der Stelle der Navigationsstruktur geladen und angezeigt. Ist der Konfigurationsschalter nicht gesetzt, bleiben die Daten bestehen, auch wenn das Objekt aus einer anderen Stelle der Navigation erneut geöffnet wird.

Verwandte Themen

- Bestandteile einer Menüführung auf Seite 125
- Erstellen der Datenbankabfrage für datenabhängige Menüeinträge auf Seite 142
- Bearbeiten von Listen auf Seite 144
- Verwenden von Links in der Menüführung auf Seite 148
- Bearbeiten von Überblickformularen auf Seite 179
- Einbinden von Statistiken in die Benutzeroberfläche auf Seite 196
- Erweitern des Launchpad auf Seite 208
- Verwenden von Variablen in der Menüführung auf Seite 149
- Symbole und Bilder für die Gestaltung der Benutzeroberfläche auf Seite 221
- Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369

Erstellen der Datenbankabfrage für datenabhängige Menüeinträge

Datenabhängige Menüeinträge werden aufgrund einer Datenbankabfrage generiert, welche als Ergebnismenge mehrere Datensätze zurückliefert. Diese Menüeinträge sind somit keine einzelnen Menüeinträge, sondern je nach Ergebnismenge der Datenbankabfrage, eine Menge von Menüeinträgen.

Die Beschreibung der allgemeinen Eigenschaften für Menüeinträge finden Sie unter Allgemeine Eigenschaften der Menüeinträge auf Seite 139. Für die Zusammenstellung der Datenbankabfrage sind die folgenden Eigenschaften erforderlich.

Tabelle 36: Eigenschaften für die Datenabfrage

Eigenschaft	Beschreibung
Tabelle	Tabelle, aus der die Werte ausgelesen werden.
Sortierung	Spalten der Tabelle, nach denen die Anzeigeelemente sortiert werden. Die Angabe muss der Order By-Syntax von Datenbankabfragen genügen. Ist keine Sortierung angegeben, wird nach den Spalten des Anzeigemusters sortiert. Eine Sortierung sollten Sie verwenden, wenn die Daten ein Datum oder sprachunabhängige Daten repräsentieren. HINWEIS: Für eine sprachabhängige Sortierung verwenden Sie den Konfigurationsschalter **Nachsortierung des datenabhängigen Menüeintrags nach Anzeigetext**.
Bedingung	Bedingung zur Einschränkung der darzustellenden Ergebnismenge. Die Angabe muss der Where-Klausel-Syntax von Datenbankabfragen genügen. Bei der Formulierung einer Bedingung können Sie Variablen einsetzen. Für rekursive datenabhängige Menüeinträge ist die Verwendung von Variablen zwingend erforderlich. HINWEIS: Die Bedingung darf kein `JOIN` enthalten, gegebenenfalls muss die Abfrage als Unterabfrage formuliert werden.
Eindeutig	Die Ergebnismenge der Abfrage darf keine doppelten Einträge enthalten. Im Zweifelsfall wird dies durch Setzen dieser Option erreicht. Menüeinträge, die mit der Option **Eindeutig** gekennzeichnet sind, müssen in ihrem Namen Variablen enthalten, über welche die Eindeutigkeit erreicht wird. Sollen beispielsweise alle Softwareanwendungen (Tabelle `Application`) nach Sprachen gruppiert werden, so muss der Name des entsprechenden Menüeintrags eine Variable enthalten, die auf die Spalte `UID_DialogCulture` in der Tabelle `Application` verweist. HINWEIS: Für die über die Datenbankabfrage ermittelten Objekte werden keine objektabhängigen Oberflächenformulare angezeigt. HINWEIS: Die Option wird unwirksam, wenn der Konfigurationsschalter **Menüeintrag immer aktivieren** gesetzt ist.
Rekursiver Aufruf	Dieser Menüeintrag ist ein rekursiver Nachfolger des vorhergehenden Menüeintrag. Ist diese Option nicht gesetzt werden die Ergebnisse als flache Struktur dargestellt. Setzen Sie diese Option, wenn die Menüeinträge zur Darstellung hierarchischer Strukturen dienen. Rekursive datenabhängige Menüeinträge definieren Sie immer unterhalb eines datenabhängigen Menüeintrags ohne Rekursion.

Verwandte Themen

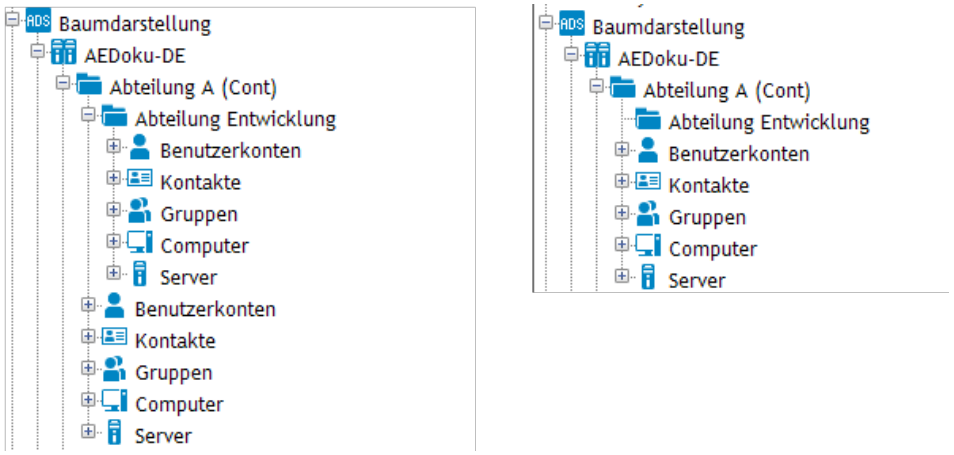
- Neue Menüeinträge erstellen auf Seite 136
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139

- Rekursive datenabhängige Menüeinträge auf Seite 144
- Verwenden von Variablen in der Menüführung auf Seite 149

Rekursive datenabhängige Menüeinträge

Kernstück des Hierarchieaufbaus ist die Variablenersetzung. Variablen werden in einer hierarchischen Menüführung weitergereicht und können somit in tieferen Ebenen weiterverwendet oder überschrieben werden. Für rekursive datenabhängige Menüeinträge wird eine in der Datenbankabfrage enthaltene Variable zunächst mit dem vorhandenen Wert dieser Variablen aus der übergeordneten Ebene ersetzt und dann die Abfrage gestartet. Der Wert des Ergebnisses ergibt sofort den neuen Variablenwert, mit dem im nächsten Schritt der übergeordnete Elternknoten noch einmal verarbeitet wird. Der ursprüngliche Wert der alten Variablen ist nach der Ausführung der Datenbankabfrage nicht mehr verfügbar. Liefert die Datenbankabfrage eine leere Ergebnismenge, so erfolgt der Abbruch der Rekursion.

Abbildung 12: Darstellung datenabhängiger Menüeinträge mit rekursivem Aufruf (links) und ohne rekursiven Aufruf (rechts)



Verwandte Themen

- Erstellen der Datenbankabfrage für datenabhängige Menüeinträge auf Seite 142
- Verwenden von Variablen in der Menüführung auf Seite 149

Bearbeiten von Listen

Für fixe Menüeinträge und datenbankhängige Menüeinträge können Sie Listeneigenschaften festlegen. Diese Eigenschaften bestimmen die Darstellung der Tabelleneinträge in der Ergebnisliste der Benutzeroberfläche.

Die Beschreibung der allgemeinen Eigenschaften für Menüeinträge finden Sie unter Allgemeine Eigenschaften der Menüeinträge auf Seite 139. Für die Definition der Listen verwenden Sie die folgende Eigenschaften.

Tabelle 37: Eigenschaften für Listen

Eigenschaft	Beschreibung
Anzeigemuster	Anzeigemuster, mit dem die Tabelleneinträge in der Ergebnisliste der Administrationswerkzeuge dargestellt werden. Ist ein kunden-spezifisches Anzeigemuster vorhanden, so wird dieses anstelle des Standardanzeigemusters verwendet. Syntax: `%Spaltenname%`
Objekt	Objektdefinition, nach der die Einträge der Liste ermittelt werden.
Bedingung	Bedingung zur Einschränkung der Ergebnismenge. Die Angabe muss der `Where`-Klausel-Syntax von Datenbankabfragen genügen. Die Bedingung bezieht sich auf die angegebene Objektdefinition. Die hier eingegebene Bedingung wird mit einer bereits hinterlegten Bedingung für die Objektdefinition zusammengeführt. Es können die Variablen verwendet werden, welche in der Navigation bis zu diesem Punkt verfügbar sind.
Symbol	Symbol für die Anzeige der Einträge in der Listendarstellung.
Sortierung	Spalten, nach denen die Elemente der Liste sortiert werden. Die Angabe muss der `Order By`-Syntax von Datenbankabfragen genügen. Eine Sortierung sollten Sie verwenden, wenn die Daten ein Datum oder sprachunabhängige Daten repräsentieren. Für eine sprachabhängige Sortierung verwenden Sie den Konfigurationsschalter **Nachsor-tierung der Ergebnismenge nach Anzeigetext**.
Einzufügende Werte	Einfügewerte, um beim Einfügen eines neuen Datensatzes über die Ergebnisliste einzelne Werte bereits vorzubelegen. Einfügewerte erfassen Sie in VB.Net-Syntax. Bei der Definition der Einfügewerte können Sie die Variablen verwenden, die in der Navigation bis zu diesem Punkt verfügbar sind.
Einfügen in Liste erlaubt	Gibt an, ob in der entsprechenden Ergebnisliste das Einfügen von Einträgen generell erlaubt ist. Ob ein Benutzer Einträge einfügen darf, ist abhängig von seinen Berechtigungen. Ausführliche Informationen zum vergeben von Berechtigungen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.
Löschen in Liste erlaubt	Gibt an, ob in der entsprechenden Ergebnisliste das Löschen von Einträgen generell erlaubt ist. Ob ein Benutzer Einträge löschen darf, ist abhängig von seinen Berechtigungen. Ausführliche Informationen zum vergeben von Berechtigungen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Verwandte Themen

- Neue Menüeinträge erstellen auf Seite 136
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139
- Anzeigemuster für die Listendarstellung auf Seite 146
- Definieren von Einfügewerten auf Seite 147
- Verwenden von Variablen in der Menüführung auf Seite 149
- Sprachabhängige Abbildung von Informationen auf Seite 233
- Objektdefinitionen für die Benutzeroberfläche auf Seite 119

Anzeigemuster für die Listendarstellung

Mit einem Anzeigemuster für die Listendarstellung legen Sie fest, in welcher Form die Tabelleneinträge in der Ergebnisliste der Administrationswerkzeuge angezeigt werden. Anzeigemuster können Sie für die Listen der Menüeinträge, Objektdefinitionen und Tabellen definieren.

Die Ermittlung des zu verwendenden Anzeigemusters erfolgt in der Reihenfolge:

1. Anzeigemuster der Liste des Menüeintrags
2. Anzeigemuster der Objektdefinition
3. Anzeigemuster der Tabelle

Das Anzeigemuster für die Listendarstellung wird nach folgender Syntax beschrieben.

```
%Spaltenname%
```

Es sind alle Spalten der anzuzeigenden Tabelle zur Verwendung in einem Anzeigemuster zugelassen. Die Verwendung von Variablen in Anzeigemustern für die Listendarstellung ist nicht zulässig.

Die Ersetzung des Anzeigemusters unterstützt den `??` Operator. Damit können Sie bedingte Anzeigemuster nach folgender Syntax formulieren.

```
%Spaltenname1??Spaltenname2??Spaltenname3%
```

```
%Spaltenname1 ?? Spaltenname2%
```

Aus der Auflistung von Spaltennamen, wird die erste Spalte verwendet, die einen Wert zurückliefert. Vor und nach dem `??` Operator sind Leerzeichen zulässig. Am Anfang und Ende des bedingten Anzeigemusters sind Leerzeichen aus Performancegründen nicht zulässig.

Beispiel:

Die Active Directory Benutzerkonten (Tabelle `ADSAccount`) sollen derart angezeigt werden:

`Common Name (Vollqualifizierter Domänenname)`

Das Anzeigemuster für die Tabelle `ADSAccount` ist dazu festzulegen mit:

`%cn% (%CanonicalName%)`

Verwandte Themen

- Bearbeiten von Listen auf Seite 144
- Definition der Parameterwerte bearbeiten auf Seite 461

Definieren von Einfügewerten

Einfügewerte verwenden Sie um beim Einfügen eines neuen Datensatzes über die Ergebnisliste einzelne Werte bereits vorzubelegen. Einfügewerte können Sie für Oberflächenformulare, Objektdefinitionen, Listen für die Menüeinträge und Tabellen einsetzen.

Einfügewerte erfassen Sie in VB.Net-Syntax. Mit `base.` wird immer das aktuell geladene Objekt angesprochen. Einfügewerte werden nach folgender Syntax beschrieben:

- Einfache Wertzuweisung

  ```
  Base.PutValue("<Spalte>", <Wert>)
  ```

- Wertzuweisung mit Variablenersetzung (Wert muss eine Zeichenkette sein)

  ```
  Base.PutValue("<Spalte>", context.Replace(<Wert>))
  ```

Es sind alle Spalten der anzuzeigenden Tabelle zur Verwendung zugelassen. Sie können bei der Definition von Einfügewerten Variablen verwenden.

Beispiel:

```
Base.PutValue("IsITShopOnly", 1)
Base.PutValue("UID_ADSContainer", context.Replace("%cont%"))
```

HINWEIS: Haben Sie Einfügewerte bearbeitet, müssen Sie anschließend die Datenbank kompilieren!

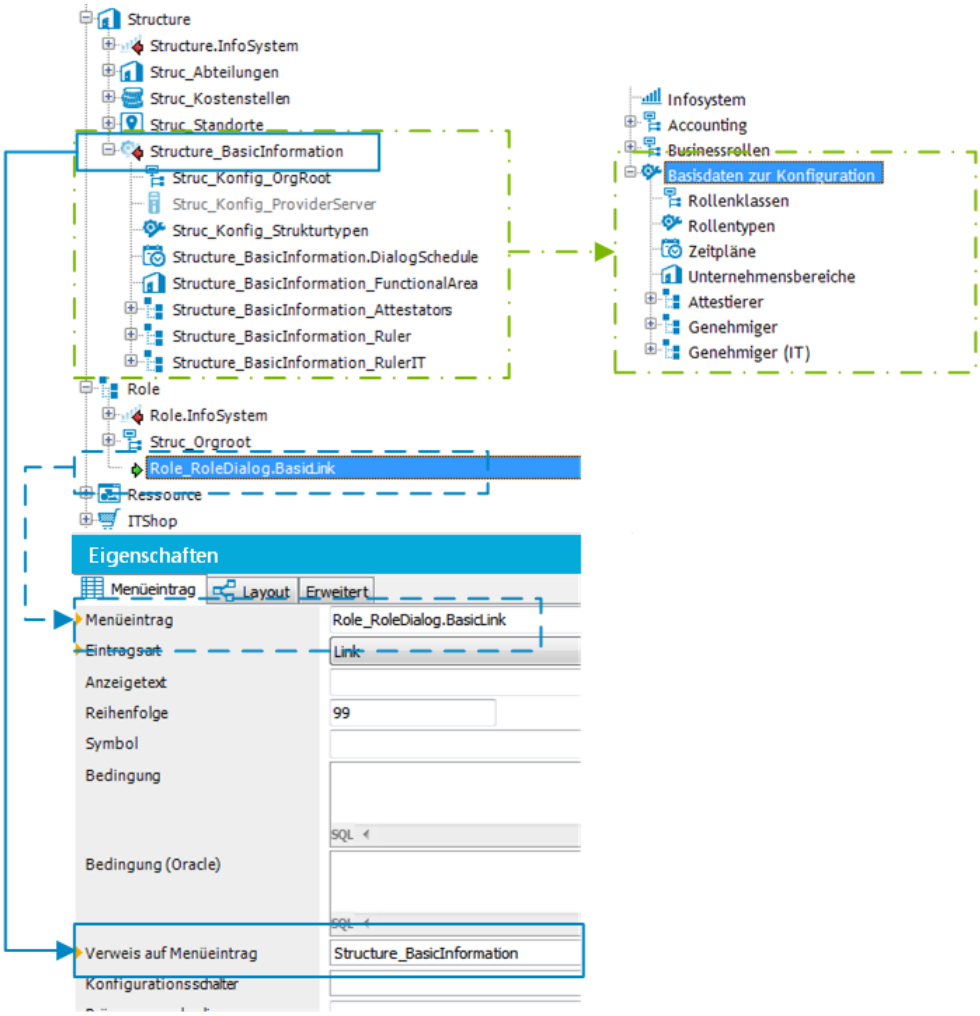
Verwandte Themen

- Verwenden von Variablen in der Menüführung auf Seite 149

Verwenden von Links in der Menüführung

Verweise (Links) unterstützen den Aufbau einer Menüführung. Links werden eingesetzt, um häufig verwendete Menüeinträge zu referenzieren. Teile der Menüführung, die in der gesamten Menüführung einer Anwendung mehrfach benötigt werden, müssen somit nur einmal erstellt werden. Links werden in der Menüführung selbst nicht angezeigt, sondern es werden immer die referenzierten Menüeinträge und ihre untergeordneten Menüeinträge dargestellt.

Abbildung 13: Aufbau der Menüführung unter Verwendung von Links im Oberflächeneditor (links) und Darstellung in der Benutzeroberfläche des Manager (rechts)



Besonderheiten bei der Verwendung von Links

- Einige Eigenschaften des Referenzeintrags werden an die Links vererbt.
- Im Referenzeintrag können Sie Variablen verwenden, beispielsweise in Bedingungen für Listen oder datenabhängige Menüeinträge. Die Wertzuweisung zu den Variablen erfolgt erst im Link. Die Variablen müssen Sie im Link definieren.
- Der Anzeigetext und das Symbol des Referenzeintrags werden mit den entsprechenden Werten des Links überschrieben.

Um Links zu verwenden

1. Erstellen Sie den Menüeintrag, der als Referenzeintrag verwendet werden soll.
2. Erstellen Sie bei Bedarf unterhalb des Referenzeintrags weitere Menüeinträge.
3. Erstellen Sie die Menüeinträge, die als Link auf den Referenzeintrag verweisen. Erfassen Sie für den Link mindestens die folgenden Informationen.
    - **Menüeintrag**: Erfassen Sie die Bezeichnung des Menüeintrags.
    - **Eintragsart**: Wählen Sie die Eintragsart **Link**.
    - **Verweis auf Menüeintrag**: Wählen Sie den Referenzeintrag, der bei Aufruf des Links zur Laufzeit angezeigt wird.
4. Weisen Sie eine Anwendung und die Berechtigungsgruppen zu.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

TIPP:

- Handelt es sich um einen Menüeintrag vom Typ **Link**, können Sie über das Kontextmenü **Link folgen** zum Referenzeintrag navigieren.
- Für einen Referenzeintrag können Sie über das Kontextmenü **Referenziert von** alle Links anzeigen, die auf diesen Referenzeintrag verweisen und zu diesen Einträgen navigieren.

Verwandte Themen

- Neue Menüeinträge erstellen auf Seite 136
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139
- Verwenden von Variablen in der Menüführung auf Seite 149

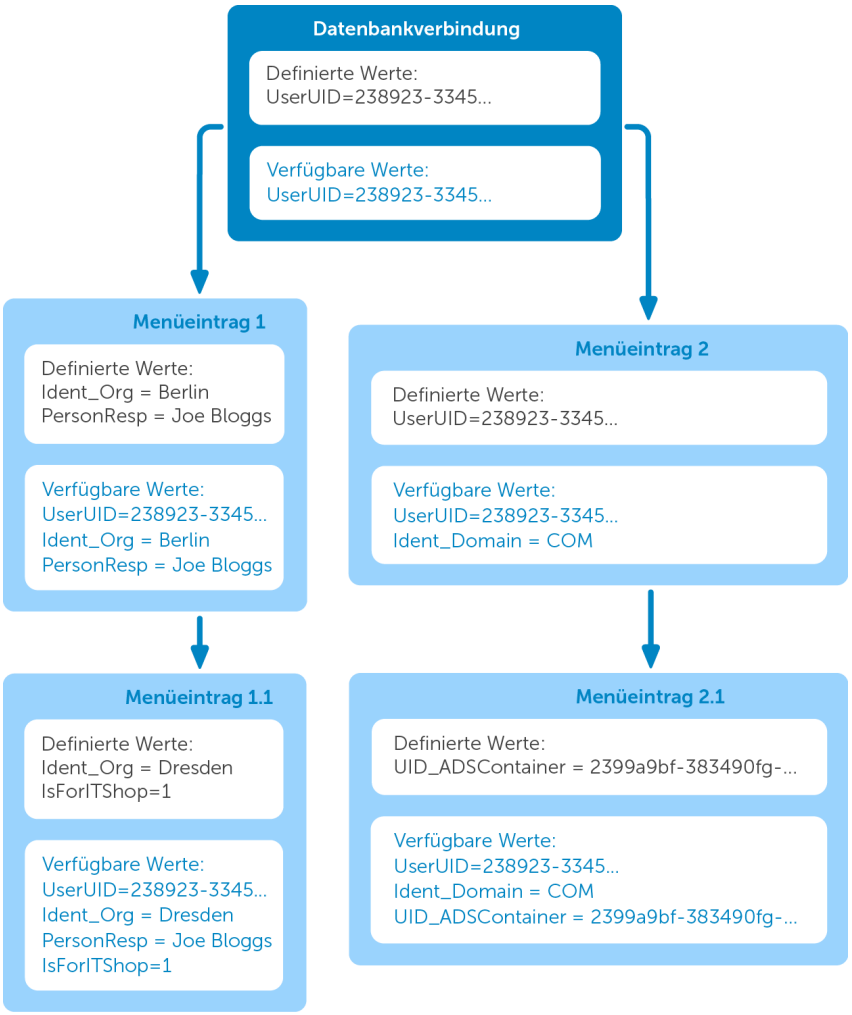
Verwenden von Variablen in der Menüführung

Bei der Gestaltung der Bezeichnungen und der Anzeigemuster von Menüeinträgen, in Einfügewerten und in Datenbankabfragen können Sie Variablen einsetzen. In einigen Bestandteile der Menüführung ist die Verwendung von Variablen sogar zwingend

erforderlich, so zum Beispiel bei der Formulierung der Datenbankabfrage für rekursive datenabhängige Menüeinträge.

Innerhalb der hierarchischen Menüführung werden Variablen vererbt. Somit sind Variablen in tieferen Ebenen einer Hierarchie wiederverwendbar oder überschreibbar. Zur Laufzeit wird der aktuelle Wert an die Variable übergeben.

Abbildung 14: Vererbung von Variablen in der hierarchischen Menüführung



Die nachfolgend aufgeführten Variablen des `Session`-Objektes stehen bei der Einrichtung der Menüeinträge immer zur Verfügung.

Tabelle 38: Globale Variablen des Session-Objektes

Variable	Bedeutung
EnvUserName	Name des Benutzers in der Umgebung, gegen die authentifiziert wurde, beispielsweise **Domain\User** bei Active Directory.
LogonUser	`DialogUser.Username` des angemeldeten Benutzers.

Variable	Bedeutung
DialogUserUID	`DialogUser.UID_DialogUser` des angemeldeten Benutzers.
UserName	Der Name, der in `XUserInserted` bzw. `XUserUpdated` abgebildet wird.
UserUID	Die `UID_Person` des angemeldeten Benutzers, falls ein personenbezogenes Authentifizierungsmodul benutzt wurde.
ShowCommonData	Gibt an, ob Systemdaten angezeigt (**1**) oder nicht angezeigt (**0**) werden. Die Variable wird im Designer über die Programmeinstellungen ausgewertet.
SessionType	Gibt an, ob eine direkte Datenbankverbindung oder eine Verbindung über Anwendungsserver unterstützt wird. Nur direkte Datenbankverbindung: `'%SessionType%' = 'Direct'` Nur Verbindung über Anwendungsserver: `'%SessionType%' = 'AppServer'`

Der Zugriff auf die Variablen erfolgt über folgende Syntax:

```
%<Variable>%
```

Verwandte Themen

- Variablen erstellen und anzeigen auf Seite 151
- Allgemeine Eigenschaften der Menüeinträge auf Seite 139
- Erstellen der Datenbankabfrage für datenabhängige Menüeinträge auf Seite 142
- Bearbeiten von Listen auf Seite 144
- Verwenden von Links in der Menüführung auf Seite 148
- Abfrage von globalen Variablen des Session-Objektes auf Seite 388

Variablen erstellen und anzeigen

Zusätzlich zu den Variablen des `Session`-Objektes können Sie weitere Variablen definieren. Die Variablendefinition setzt sich zusammen aus dem Variablentyp, dem Namen der Variablen und dem Wert. Grundsätzlich ist zur Definition eigener Variablen jede mögliche Zeichenkette zulässig. Es hat sich jedoch bewährt, ein Muster zu verwenden, das nur mit geringer Wahrscheinlichkeit in Daten auftreten wird, jedoch als Zeichenkette vom verwendeten Datenbankserver akzeptiert wird.

Der Zugriff auf die Variablen erfolgt über folgende Syntax:

```
%<Variable>%
```

Tabelle 39: Variablendefinitionen

Variablentyp	Variablenname	Wert	Verwendung
Spalte	beliebige Zeichenkette	Name der Spalte des aktuellen Objektes	Nur in datenabhängigen Menüeinträgen verwendbar.
Anzeigewert	beliebige Zeichenkette	Name der Spalte des aktuellen Objektes	Nur in datenabhängigen Menüeinträgen verwendbar. Bei der Bildung des Anzeigewertes für eine Spalte werden die Spalteneigenschaften **Mehrsprachig** und **Liste zulässiger Werte** aufgelöst.
Text	beliebige Zeichenkette	Frei definierter Wert	In allen Menüeinträgen verwendbar.

Um Variablen zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
2. Wählen Sie in der Übersicht der Menüführung den Menüeintrag.
3. Wählen Sie in der Bearbeitungsansicht die Ansicht **Variablendefinition**.

   In dieser Ansicht erfolgt die tabellarische Darstellung aller zum gewählten Menüeintrag verfügbaren Variablendefinitionen mit ihrem Typ, dem Namen und der Wertzuweisung.

   > TIPP: Um Variablen, die aus übergeordneten Menüeinträgen geerbt werden, anzuzeigen, klicken Sie .
4. Um eine Variable zu erstellen, klicken Sie und erfassen Sie die folgenden Informationen.
   - **Variablentyp**: Wählen Sie **Spalte**, **Anzeigewert** oder **Text**.
   - **Variable**: Erfassen Sie die Bezeichnung der Variable.
   - **Wert**: Erfassen Sie den Wert der Variable. Der einzugebende Wert ist abhängig vom Variablentyp.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Die aktuelle Wertbelegung der Variablen kann in den Administrationswerkzeugen als zusätzliche Menüführungsinformation angezeigt werden.

Um die Werte der Variablen eines Menüeintrages im Manager anzuzeigen

- Aktivieren Sie im Manager die Programmeinstellung **Zusätzliche Informationen zu Objekten der Menüführung anzeigen**.

- Wählen Sie im Manager den Menüeintrag in der Menüführung aus und wählen das Kontextmenü **Definition > Definierte Variablen**.

Verwandte Themen

- Verwenden von Variablen in der Menüführung auf Seite 149

Formulare für die Benutzeroberfläche

Zur Anzeige und Bearbeitung von Daten in der Benutzeroberfläche werden Oberflächenformulare verwendet. Grundlegende Informationen zur Darstellung der Daten auf den Oberflächenformularen werden in Formulardefinitionen und Formularvorlagen beschrieben. Für die Anzeige eines Oberflächenformulars wird die referenzierte Formulardefinition ermittelt. Es wird überprüft, ob die in der Formulardefinition angegebene Formularvorlage in den Formulararchiven vorhanden ist und ob diese Formularvorlage für den gewünschten Anzeigezweck gekennzeichnet ist.

Detaillierte Informationen zum Thema

- Empfehlungen für die Bearbeitung von Formularen auf Seite 153
- Bearbeiten von Oberflächenformularen auf Seite 155
- Formulare für kundenspezifische Erweiterungen auf Seite 163
- Bearbeiten von Überblickformularen auf Seite 179

Empfehlungen für die Bearbeitung von Formularen

- Bei Bedarf können Sie einzelne vordefinierte Formulare deaktivieren, um deren Anzeige in der Benutzeroberfläche zu verhindern. Die Deaktivierung bleibt auch bei Schemainstallationen erhalten.
- In der Standardinstallation des One Identity Manager steht bereits eine Reihe von Formularvorlagen und Formulardefinitionen zur Verfügung, beispielsweise für die Bearbeitung von Stammdaten sowie M:N-Beziehungen und Objektbeziehungen (Parent/ChildRelation). Diese können Sie für die einfache Erstellung eigener Formulare verwenden.
- Zur Anzeige der Informationen zu einem Basisobjekt erstellen Sie ein Überblickformular.

- Verwenden Sie dazu im Designer den Überblicksformulareditor.
- Erstellen Sie Menüeinträge für häufig dazustellende Objektbeziehungen und verwenden Sie diese Menüeinträge als Verweise in den Formularelementen des Überblickformulars.

  TIPP: Sie können die Menüeinträge für Objektbeziehungen vom Überblicksformulareditor erzeugen lassen.

  - Wählen Sie die Objektbeziehung, die Sie darstellen möchten, und ziehen Sie diese per Drag and Drop auf ein Element in der Elementregion des Überblicksformulareditor.
  - Verwenden Sie die Kontextmenüs **Listenelement referenzieren** oder **Datenelement referenzieren**.

  Die Menüeinträge werden unterhalb des Menüeintrags `InfoSheets.QIM.Links` mit den Bezeichnungen `InfoSheet.List.<Tabelle>` beziehungsweise `InfoSheet.Node.<Tabelle>` angelegt.

  Die Bedingung der Menüeinträge wird als Variable `%<Tabelle>WhereClause%` definiert. Der Variablen weisen Sie am Formularelement eine als Where-Klausel formulierte Bedingung zu.

- Für kundenspezifische Spaltenerweiterungen an Standardtabellen können unter bestimmten Voraussetzungen die Standardformulare verwendet werden.
- Für die Bearbeitung der Stammdaten kundenspezifischer Tabellen erstellen Sie mit dem Formulareditor des Designer ein Oberflächenformular mit der Formulardefinition **VI_Generic_MasterData**.
- Um Zuordnungen vorzunehmen, erstellen Sie weitere Oberflächenformulare mit dem Formulartyp **MemberRelation**.
- Weisen Sie die Formulare und Menüeinträge an die Anwendung, beispielsweise **Manager**, zu.
- Weisen Sie die Formulare und Menüeinträge an die Berechtigungsgruppen für die nicht-rollenbasierte Anmeldung und die rollenbasierte Anmeldung zu.
- Sollte es erforderlich sein, können Sie eigen-entwickelte Formularvorlagen in einem Formulararchiv zur Verfügung stellen (`*.Forms.vif`).

Verwandte Themen

- Bearbeiten von Oberflächenformularen auf Seite 155
- Oberflächenformulare deaktivieren auf Seite 156
- Oberflächenformulare erstellen auf Seite 157
- Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen auf Seite 162
- Formulare für kundenspezifische Erweiterungen auf Seite 163
- Standardformulare mit kundenspezifischen Formularen ersetzen auf Seite 166
- Bearbeiten von Überblickformularen auf Seite 179

Bearbeiten von Oberflächenformularen

Oberflächenformulare werden an Objektdefinitionen gebunden, so dass abhängig vom ausgewählten Objekt unterschiedliche Formulare in der Benutzeroberfläche angeboten werden. Durch die zusätzliche Zuweisung der Oberflächenformulare an Berechtigungsgruppen, werden diese Oberflächenformulare den Systembenutzern abhängig von ihren Mitgliedschaften in Berechtigungsgruppen zur Verfügung gestellt. Des Weiteren können Oberflächenformulare für einzelne Menüeinträge definiert werden. Diese Oberflächenformulare werden für alle Systembenutzer unabhängig von Mitgliedschaften in Berechtigungsgruppen angezeigt, wenn der zugehörige Menüeintrag in der Menüführung beziehungsweise ein Eintrag in der Ergebnisliste ausgewählt wird.

Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar.

HINWEIS: Bei Bedarf können Sie einzelne vordefinierte Oberflächenformulare deaktivieren, um deren Anzeige in der Benutzeroberfläche zu verhindern. Die Deaktivierung bleibt auch bei Schemainstallationen erhalten.

Verwandte Themen

- Hinweise zum Arbeiten mit dem Formulareditor auf Seite 155
- Oberflächenformulare deaktivieren auf Seite 156
- Oberflächenformulare kopieren auf Seite 157
- Oberflächenformulare erstellen auf Seite 157
- Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen auf Seite 162
- Formulare für kundenspezifische Erweiterungen auf Seite 163

Hinweise zum Arbeiten mit dem Formulareditor

Mit dem Formulareditor erstellen und bearbeiten Sie die Oberflächenformulare, wie beispielsweise Stammdatenformulare oder Zuordnungsformulare. In der Formularübersicht des Editors werden alle Oberflächenformulare dargestellt.

- Formulare, die über Präprozessorbedingungen deaktiviert sind, werden in der Formularübersicht grau dargestellt.
- Die Formularübersicht im Formulareditor aktualisieren Sie mit **F5**.
- Die Anzeige der Formulare kann als Liste oder als hierarchisch erfolgen. In der hierarchischen Darstellung erfolgt eine Gruppierung der Oberflächenformulare nach Formularvorlagen und Formulardefinitionen. Verwenden Sie den Menüeintrag **Optionen > Listen/Baum-Darstellung** um die Darstellung zu ändern.
- Über das Menü **Optionen > Beschriftungen anzeigen** wechseln Sie zwischen den technischen Bezeichnungen der Formulare und den benutzerfreundlichen

Anzeigetexten.

- Über das Menü **Optionen > Spalten wählen** können Sie zusätzliche Spalten in der Formularübersicht anzeigen.
- Um die Menge der angezeigten Formulare in der Formularliste einzuschränken definieren Sie Filter. Verwenden Sie die Menüeinträge Filter definieren oder Filter verwalten. Ausführliche Informationen zum Arbeiten mit benutzerdefinierten Filtern im Designerfinden Sie im *One Identity Manager Anwenderhandbuch für die Benutzeroberfläche der One Identity Manager-Werkzeuge*.
- Nutzen Sie während der Bearbeitung von Stammdatenformularen die Formularvorschau. Verwenden Sie den Menüeintrag **Ansicht > Formularvorschau** einen zusätzlichen Tabreiter **Formularvorschau** im Formulareditor einzublenden.

  In der Formularvorschau wird der Inhalt des Oberflächenformulars anzeigt. Sie sehen, welche Basistabelle zur Darstellung der Daten verwendet wird. Beim Laden und Anzeigen eines Oberflächenformulars werden die Berechtigungen des angemeldeten Designer-Benutzers berücksichtigt.

  Kann ein Formular nicht geladen und angezeigt werden, wird eine entsprechende Fehlermeldung ausgegeben.

Oberflächenformulare deaktivieren

Bei Bedarf können Sie einzelne Oberflächenformulare deaktivieren, um deren Anzeige in der Benutzeroberfläche zu verhindern. Die Deaktivierung für vordefinierte Oberflächenformulare bleibt auch bei der Schemaaktualisierung erhalten.

Um ein Oberflächenformular zu deaktivieren

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie im Formulareditor das Oberflächenformular.
4. Wählen Sie in der Bearbeitungsansicht die Ansicht **Eigenschaften.**
5. Wählen Sie den Tabreiter **Oberflächenformular** und setzen Sie die Option **Deaktiviert**.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Zusätzlich können Oberflächenformulare über Präprozessorbedingungen deaktiviert sein.

Verwandte Themen

- Eigenschaften von Oberflächenformularen auf Seite 167

Oberflächenformulare kopieren

Nutzen Sie diese Aufgabe, wenn Sie nur geringfügige Anpassungen vornehmen möchten, wie beispielsweise einen anderen Anzeigetext oder eine andere Sortierreihenfolge.

Um ein Oberflächenformular zu kopieren

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie im Formulareditor das Oberflächenformular, das Sie kopieren möchten.
4. Wählen Sie den Menüeintrag **Formular > Einfügen**.

   Es wird eine Kopie des gewählten Oberflächenformulars erstellt.
5. Bearbeiten Sie die weiteren Stammdaten des Oberflächenformulars.
6. Weisen Sie das Oberflächenformular an die Anwendungen und Berechtigungsgruppen zu.
7. (Optional) Weisen Sie das Oberflächenformular an die Objektdefinitionen zu.
8. (Optional) Weisen Sie das Oberflächenformular an die Menüeinträge zu.
9. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

HINWEIS: Deaktivieren Sie das originale Oberflächenformular. Anderenfalls werden beide Oberflächenformulare in der Benutzeroberfläche angezeigt.

Verwandte Themen

- Oberflächenformulare deaktivieren auf Seite 156
- Eigenschaften von Oberflächenformularen auf Seite 167
- Oberflächenformulare an Anwendungen zuweisen auf Seite 158
- Oberflächenformulare an Berechtigungsgruppen zuweisen auf Seite 159
- Oberflächenformulare an Objektdefinitionen zuweisen auf Seite 160
- Oberflächenformulare an Menüeinträge zuweisen auf Seite 161
- Oberflächenformulare erstellen auf Seite 157

Oberflächenformulare erstellen

Erstellen Sie ein neues Oberflächenformular beispielsweise wenn Sie kundenspezifische Schemaerweiterungen in der Benutzeroberfläche anzeigen möchten. Der One Identity Manager stellt in der Standardinstallation eine Reihe von Formularvorlagen und Formulardefinitionen zur Verfügung, die Sie für die einfache Erstellung eigener Formulare verwenden können.

Um ein neues Oberflächenformular zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie den Menüeintrag **Formular > Einfügen**.

   Der Formulareditor öffnet in der Bearbeitungsansicht ein neues Formular mit der Bezeichnung **new sheet**.
4. Auf dem Tabreiter **Formulardefinition** wählen Sie eine Formularvorlage und erfassen den Namen der Formulardefinition.
5. Auf dem Tabreiter **Oberflächenformular** erfassen Sie einen Formularnamen und den Anzeigetext. Bearbeiten Sie die weiteren Stammdaten des Oberflächenformulars.
6. Weisen Sie das Oberflächenformular an die Anwendungen und Berechtigungsgruppen zu.
7. (Optional) Weisen Sie das Oberflächenformular an die Objektdefinitionen zu.
8. (Optional) Weisen Sie das Oberflächenformular an die Menüeinträge zu.
9. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Eigenschaften von Oberflächenformularen auf Seite 167
- Oberflächenformulare an Anwendungen zuweisen
- Oberflächenformulare an Berechtigungsgruppen zuweisen
- Oberflächenformulare an Objektdefinitionen zuweisen
- Oberflächenformulare an Menüeinträge zuweisen auf Seite 161
- Oberflächenformulare kopieren auf Seite 157
- Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen auf Seite 162
- Formulare für kundenspezifische Erweiterungen auf Seite 163

Oberflächenformulare an Anwendungen zuweisen

Um ein Oberflächenformular in der Benutzeroberfläche einer Anwendung anzuzeigen, weisen Sie das Formular der Anwendung zu

Um ein Oberflächenformular an eine Anwendung zuzuweisen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.

2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie im Formulareditor das Oberflächenformular.
4. Wählen Sie in der Bearbeitungsansicht die Ansicht **Anwendung** und wählen Sie die Anwendung.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Oberflächenformulare erstellen auf Seite 157
- Oberflächenformulare an Berechtigungsgruppen zuweisen auf Seite 159
- Anwendungen für die Gestaltung der Benutzeroberfläche auf Seite 218

Oberflächenformulare an Berechtigungsgruppen zuweisen

Alle Oberflächenformulare, die in der Benutzeroberfläche einer Anwendung dargestellt werden sollen, müssen Sie einer Berechtigungsgruppe zuweisen. Weisen Sie die Oberflächenformulare an Berechtigungsgruppen für die nicht-rollenbasierte Anmeldung und die rollenbasierte Anmeldung zu. Die Oberflächenformulare werden den Systembenutzern abhängig von ihren Mitgliedschaften in Berechtigungsgruppen zur Verfügung gestellt. Ausführliche Informationen zu Berechtigungsgruppen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Um ein Oberflächenformular an Berechtigungsgruppen zuzuweisen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie im Formulareditor das Oberflächenformular.
4. Wählen Sie in der Bearbeitungsansicht die Ansicht **Berechtigungsgruppe** und wählen Sie die Berechtigungsgruppen.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Oberflächenformulare erstellen auf Seite 157
- Oberflächenformulare an Anwendungen zuweisen auf Seite 158

Oberflächenformulare an Objektdefinitionen zuweisen

Soll ein Oberflächenformular abhängig vom gewählten Objekt in der Benutzeroberfläche dargestellt werden, müssen Sie das Formular an die gültige Objektdefinition zuweisen.

Um ein Oberflächenformular an eine Objektdefinition zuzuweisen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie im Formulareditor das Oberflächenformular.
4. Wählen Sie in der Bearbeitungsansicht die Ansicht **Objektzuweisung** und wählen Sie die Objektdefinition.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Oberflächenformulare erstellen auf Seite 157
- Objektdefinitionen für die Benutzeroberfläche auf Seite 119
- Auswirkung der Objektdefinitionen auf die Anzeige der Oberflächenformulare auf Seite 160

Auswirkung der Objektdefinitionen auf die Anzeige der Oberflächenformulare

Oberflächenformulare, die für alle Einträge einer Datenbanktabelle gültig sein sollen, werden der allgemeinen Objektdefinition zugeordnet. Weiter eingeschränkte Objektdefinitionen können zusätzliche Oberflächenformulare erhalten. Wenn ein Eintrag in der Benutzeroberfläche angewählt wird, dann wird über die gültigen Objektdefinitionen die Gesamtmenge der Oberflächenformulare gebildet und die Oberflächenformulare werden in ihrer Sortierreihenfolge in der Aufgabenansicht und im Formularkontextmenü der Benutzeroberfläche angezeigt.

Beispiel:

Für die Tabelle `ESet` wurden die folgenden Objektdefinitionen mit Oberflächenformularen angelegt.

Tabelle 40: Beispiel: Oberflächenformulare für Objektdefinitionen

Objektdefinition	zugewiesene Oberflächenformulare
Systemrollen (`ESet`)	Überblick über die Systemrolle
Systemrollen für IT Shop (`ESet_ITShop`)	In den IT Shop aufnehmen

Zu einer Systemrolle, die die Objektdefinition **Systemrollen** erfüllt, wird das folgende Oberflächenformular angezeigt:

- Überblick über die Systemrolle

Zu einer Systemrolle, die die Objektdefinition **Systemrollen für IT Shop** erfüllt, werden die folgenden Oberflächenformulare angezeigt:

- Überblick über die Systemrolle
- In den IT Shop aufnehmen

Verwandte Themen

- Oberflächenformulare an Objektdefinitionen zuweisen auf Seite 160
- Objektdefinitionen für die Benutzeroberfläche auf Seite 119

Oberflächenformulare an Menüeinträge zuweisen

Sie können Oberflächenformulare für einzelne Menüeinträge zuweisen. Wählt ein Benutzer den Menüeintrag in der Navigationsansicht oder einen Eintrag in der Ergebnisliste wird das Oberflächenformular angezeigt. Die Anzeige erfolgt unabhängig von den Berechtigungsgruppen des Benutzers.

Um ein Oberflächenformular an einen Menüeintrag zuzuweisen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie im Formulareditor das Oberflächenformular.
4. Wählen Sie in der Bearbeitungsansicht die Ansicht **Menüzuweisung** und wählen Sie den Menüeintrag.
5. (Optional) Aktivieren Sie die Option **Anzeige aus der Navigationsstruktur**, um das Formular aus der Navigationsansicht zu öffnen.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Oberflächenformulare erstellen auf Seite 157

Anzeigen kundenspezifischer Spalten und Tabellen auf Stammdatenformularen

Anzeige von Spalten kundenspezifischer Tabellen

Für die Darstellung kundenspezifischer Datenbanktabellen in der Oberfläche der Administrationswerkzeuge und die Bearbeitung der Stammdaten:

- Erstellen Sie ein Oberflächenformular mit der Formulardefinition **VI_Generic_ MasterData**. Diese Formulardefinition stellt die Steuerelemente zur Bearbeitung der Stammdaten in der Benutzeroberfläche zur Verfügung.
- Legen Sie im Designer in der Eigenschaft **Reihenfolge** (`DialogColumn.SortOrder`) die Anzeigereihenfolge der Eingabefelder fest. Spalten, deren Reihenfolge kleiner eins ist, werden nicht angezeigt.
- Eine übersichtlichere Darstellung der Eingabefelder erreichen Sie über die Gruppierung der Datenbankspalten. Passen Sie im Designer die Eigenschaft **Gruppe** (`DialogColumn.ColumnGroup`) in der Spaltendefinition an. Pro Gruppe wird ein eigener Tabreiter angezeigt. Die Bezeichnung des Tabreiters entspricht dem Namen der Gruppe.
- Spalten, deren Dateninhalt mehrzeilig sein kann, werden auf dem generischen Formular mit einem mehrzeiligen Eingabefeld dargestellt. Die Spalten kennzeichnen Sie mit der Option **Mehrzeilig**.

Anzeige von kundenspezifischen Spalten an vordefinierten Tabellen

Für die Darstellung kundenspezifischer Spaltenerweiterungen an den Standardtabellen können auf vordefinierten Stammdatenformularen separate Tabreiter eingeblendet werden.

Wenn das vordefinierte Stammdatenformular die Formulardefinition **VI_Generic_ MasterData** verwendet, gelten die oben beschriebenen Besonderheiten. Anderenfalls sind folgende Voraussetzungen für die Nutzung dieser Funktionalität notwendig:

- Das Stammdatenformular besitzt bereits Tabreiter. Einfache Stammdatenformulare ohne Tabreiter werden nicht erweitert.
- Die Anzeigereihenfolge der Eingabefelder auf dem Formular ändern Sie über die Eigenschaft **Reihenfolge** (`DialogColumn.SortOrder`) der Datenbankspalten. Spalten, deren Reihenfolge kleiner eins ist, werden nicht angezeigt.
- Die Datenbankspalten sind gruppiert. Passen Sie im Designer die Eigenschaft **Gruppe** (`DialogColumn.ColumnGroup`) in der Spaltendefinition an. Pro Gruppe wird ein eigener Tabreiter angezeigt. Die Bezeichnung des Tabreiters entspricht dem Namen

der Gruppe. Wird keine Gruppe angegeben, dann wird ein Tabreiter mit der Bezeichnung **kundenspezifisch** angezeigt.

HINWEIS: Für die Anzeige kundenspezifischer Schemaerweiterungen an den Tabellen `UNSAccountB`, `UNSContainerB`, `UNSGroupB`, `UNSItemB` und `UNSRootB` gelten weitere Besonderheiten. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für die Anbindung kundendefinierter Zielsysteme*.

Verwandte Themen

- Formulare für kundenspezifische Erweiterungen auf Seite 163
- Bearbeiten von Oberflächenformularen auf Seite 155
- Eigenschaften von Spaltendefinitionen auf Seite 99

Formulare für kundenspezifische Erweiterungen

Der One Identity Manager stellt in der Standardinstallation eine Reihe von Formularvorlagen und Formulardefinitionen zur Verfügung, die Sie für die einfache Erstellung eigener Formulare verwenden können.

Eine zusätzliche Möglichkeit kundenspezifische Formulare zu erstellen, ist die Bereitstellung eines kundenspezifischen Formulararchivs. Hierbei werden in der Regel im One Identity Manager enthaltene Standardformulare durch eigen-entwickelte Formulare ersetzt.

Tabelle 41: Formularvorlagen und Formulardefinitionen für kundenspezifische Erweiterungen

Formularvorlage	Formulardefinition	Verwendung
FrmCommonChildRelationGrid	VI_Common_ ChildRelation_Grid	Zur Bearbeitung von M:N-Beziehungen mit erweiterten Eigenschaften in Form einer Tabelle.
FrmCommonOneChild AndMemberRelation FrmCommonOneMember AndChildRelation	Es muss eine eigene Formulardefinition erstellt werden, auf der die Konfiguration der darzustellenden Daten erfolgt.	Zuweisen von M:N-Beziehungen und Objektbeziehungen (Parent/ChildRelation) auf einem Formular. Für die Abbildung werden auf einem Formular zwei Tabreiter angezeigt.
FrmCommonOneChildRelation	Es muss eine eigene Formulardefinition	Abbildung von Objektbeziehungen (Parent/ChildRelation). Wenn auf einem Formular

Formularvorlage	Formulardefinition	Verwendung
	erstellt werden, auf der die Konfiguration der darzustellenden Daten erfolgt.	mehrere weitere Objektbeziehungen abgebildet werden, können alternativ die Formularvorlagen **FrmCommonTwoChildRelation**, **FrmCommonThreeChildRelation** verwendet werden. Pro Objektbeziehung wird ein Tabreiter angezeigt.
FrmCommonOneDynamicRelation	Es muss eine eigene Formulardefinition erstellt werden, auf der die Konfiguration der darzustellenden Daten erfolgt.	Darstellung von dynamischen M:N-Beziehungen, deren zugewiesenes Objekt über einen dynamischen Schlüssel referenziert wird. Die zulässigen dynamischen Objekte werden aus der Tabelle `DialogValidDynamicRef` ermittelt. Zur Auswahl des Objekttyps wird eine Auswahlliste angeboten.
FrmCommonOneGenericRelation	Es muss eine eigene Formulardefinition erstellt werden, auf der die Konfiguration der darzustellenden Daten erfolgt.	Darstellung von dynamischen M:N-Beziehungen: • Basisobjekt kann über einen dynamischen Schlüssel referenziert werden -ODER- • zugewiesenes Objekt wird über einen dynamischen Schlüssel referenziert. Hierbei muss die Eigenschaft **MembersTableName** in der Formularkonfiguration festgelegt werden.
FrmCommonOneMemberRelation	Es muss eine eigene Formulardefinition erstellt werden, auf der die Konfiguration der darzustellenden Daten erfolgt.	Zuweisen von M:N-Beziehungen. Wenn auf einem Formular mehrere weitere M:N-Beziehungen abgebildet werden, können alternativ die Formularvorlagen **FrmCommonTwoMemberRelat**

Formularvorlage	Formulardefinition	Verwendung
		ion, **FrmCommonFourMemberRelation**, **FrmCommonFiveMemberRelation** verwendet werden. Pro M:N-Tabelle wird ein Tabreiter angezeigt.
FrmElementNavigation	VI_ ElementNavigation	Zur Darstellung des Überblickformulars.
frmGeneric	VI_Generic_ MasterData	Zur Bearbeitung der Stammdaten eines Objektes.
ReportForm	VI_Report	Zur Anzeige von Berichten.
WizardForm	VI_Wizard	Zum Einbinden von Assistenten. Die Anzeige der Formulare erfolgt in einem modalen Dialogfenster.

Verwandte Themen

- Hierarchische Darstellung von Daten auf Zuordnungsformularen auf Seite 165
- Konfigurationsdaten für die Darstellung von M:N-Beziehungen und Objektbeziehungen auf Formularen auf Seite 173
- Standardformulare mit kundenspezifischen Formularen ersetzen auf Seite 166

Hierarchische Darstellung von Daten auf Zuordnungsformularen

Zur Darstellung der Daten in einer Zuordnungsliste (M:N-Beziehungen) werden Formulare vom Typ **MemberRelation** eingesetzt. Für die hierarchische Darstellung von Tabellen tragen Sie in der Tabellendefinition den Pfad in der Hierarchie ein. Geben Sie die Fremdschlüsselspalten, über die die Hierarchie aufgebaut werden soll.

Beispiel:

Ein Active Directory Benutzerkonto (Tabelle `ADSAccount`) wird auf den Zuordnungsformularen typischerweise unterhalb seines Active Directory Containers (Spalte `UID_ADSContainer`) angezeigt. Der Active Directory Container (Tabelle `ADSContainer`) wird wiederum unterhalb seiner Active Directory Domäne (Spalte `UID_`

`ADSDomain`) dargestellt. Für den Aufbau der Hierarchie ist der Pfad in der Hierarchie folgendermaßen eingetragen.

Tabelle 42: Beispiel für den Pfad zur Hierarchie

Tabelle	Pfad in der Hierarchie
ADSAccount	UID_ADSContainer, UID_ADSDomain
ADSContainer	UID_ADSDomain

Für Objekte, die nicht in allen Fremdschlüsselspalten Werte enthalten, können Sie nach Pipe (`|`) eine alternative Auflistung angeben.

Beispiel:

`(UID_ADSContainer,UID_ADSDomain|UID_ADSDomain)`

Verwandte Themen

- Eigenschaften von Tabellendefinitionen auf Seite 73

Standardformulare mit kundenspezifischen Formularen ersetzen

Für kundenspezifische Formulare können Ihnen Eigen-entwickelte Formularvorlagen in einem Formulararchiv zur Verfügung gestellt werden (`*.CustomForms.*.vif`). Um kundenspezifische Formulare in der Benutzeroberfläche anzuzeigen, müssen Sie mit Hilfe des Formulareditors die Formularvorlage, die Formulardefinition und das Oberflächenformular einfügen.

Um ein Standardformular mit allen Abhängigkeiten gegen ein kundenspezifisches Formular auszutauschen, wird ein Assistent angeboten. Der Assistent erstellt das Oberflächenformular mit der Formulardefinition und der Formularvorlage. Die Eigenschaften des neuen Formulars werden dabei vom ersetzten Formular übernommen. Für das neue Formular werden die benötigten Zuweisungen (Objektdefinition, Menüeintrag, Berechtigungsgruppe und Anwendung) hergestellt und das ersetzte Formular wird deaktiviert.

Um ein Standardformular mit allen Abhängigkeiten zu ersetzen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die hierarchische Darstellung der Formularübersicht. Aktivieren Sie dazu den Menüeintrag **Optionen > Listen/Baum-Darstellung**.

3. Wählen Sie in der Formularübersicht in der obersten Hierarchieebene die Formularvorlage des zu ersetzenden Formulars und starten Sie den Assistenten über das Kontextmenü **Ersetzen durch**.
4. Auf der Startseite des Assistenten klicken Sie **Weiter**.
5. Auf der Seite **Datei und Formular wählen** erfassen Sie folgende Informationen.
   - **Formulararchivdatei**: Wählen Sie die Formulararchivdatei (`*.CustomForms.*.vif`).
   - **Formularvorlage**: Wählen Sie die Formularvorlage für das neue Oberflächenformular.
6. Auf der Seite **Formularstruktur definieren** prüfen Sie die Bezeichnungen der Formulardefinitionen und der Oberflächenformulare. Die Bezeichnungen der Formulardefinitionen und der Oberflächenformulare sollten mit **CCC** beginnen. Ändern Sie mit **F2** die Bezeichnungen und übernehmen Sie die Änderung mit **Enter**.
7. Auf der Seite **Berechtigungsgruppe wählen** wählen Sie die Berechtigungsgruppe, der das neue Oberflächenformular zugeordnet werden soll. Über die Schaltfläche + können Sie eine neue Berechtigungsgruppe erstellen.
8. Auf der letzten Seite des Assistenten werden die Einstellungen für die Formularersetzung zusammengefasst. Um die Formularersetzung zu starten, klicken Sie **Fertig**.

   Der Assistent wird nach der Ersetzung geschlossen. Nach Beenden des Assistenten wird das neue Formular in der Formularübersicht des Formulareditor angezeigt und kann dort weiter bearbeitet werden. Das ersetzte Formular wird deaktiviert und ist somit in der Benutzeroberfläche nicht mehr verfügbar.

Verwandte Themen

- Formulare für kundenspezifische Erweiterungen auf Seite 163

Eigenschaften von Oberflächenformularen

Tabelle 43: Eigenschaften von Oberflächenformularen

Eigenschaft	Bedeutung
Formularname	Der Formularname wird zur schnelleren Auswahl des Oberflächenformulars beispielsweise im Designer verwendet. TIPP: Der Formularname wird in den Administrationswerkzeugen als zusätzliche Menüführungsinformation angezeigt.
Formulardefinition	Formulardefinition, mit der das Oberflächenformular verbunden ist. HINWEIS: Um eine neue Formulardefinition für das Oberflächenformular einzubinden, verwenden Sie die

Eigenschaft	Bedeutung
	Schaltfläche neben dem Eingabefeld.
Anzeigetext	Anzeigetext des Oberflächenformulars. Der Anzeigetext wird zur Darstellung des Oberflächenformulars in der Aufgabenansicht und im Formularkontextmenü der Benutzeroberfläche verwendet. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Verweis auf Online-Hilfe	Hilfeschlüssel des Formulars zur Navigation auf das Kapitel der Online-Hilfe.
Beschreibung	Nähere Beschreibung zum Oberflächenformular. TIPP: Die Beschreibung wird in der Aufgabenansicht als Tooltipp angezeigt.
Symbol	Symbol das Oberflächenformular in der Benutzeroberfläche kennzeichnet.
Reihenfolge	Die Sortierreihenfolge bestimmt die Position des Oberflächenformulars in der Aufgabenansicht und im Formularkontextmenü der Administrationswerkzeuge. HINWEIS: Beim Einfügen von Objekten im Manager wird immer das Oberflächenformular vom Formulartyp **Edit** mit der niedrigsten Sortierreihenfolge angezeigt.
Präprozessorbedingung	Oberflächenformulare können mit einer Präprozessorbedingung versehen werden. Damit ist ein Oberflächenformular in der Benutzeroberfläche nur verfügbar, wenn die Präprozessorbedingung erfüllt ist. HINWEIS: Einen Überblick über die vorhandenen Präprozessorabhängigkeiten erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Präprozessorabhängigkeiten**.
Schaltflächen deaktivieren	Gibt an, welche Schaltflächen der Symbolleiste in den Frontends zu deaktivieren sind. Zulässige Werte sind: • **Objekt einfügen**: Die Schaltfläche ist deaktiviert. • **Objekt aktualisieren**: Die Schaltfläche ist deaktiviert. • **Objekt löschen**: Die Schaltfläche ist deaktiviert. • **Änderungen speichern**: Die Schaltfläche ist deaktiviert.
Deaktiviert	Mit dieser Option kennzeichnen Sie Oberflächenformulare, die in der Benutzeroberfläche nicht angezeigt werden sollen.

Eigenschaft	Bedeutung
	HINWEIS: Diese Änderung ist auch bei Oberflächenformularen der Standardoberflächen zulässig und bleibt bei Schemainstallationen bestehen.
Deaktiviert durch Präprozessor	Ist ein Oberflächenformular durch eine Präprozessorbedingung ausgeschaltet, dann wird durch den Database Compiler die Option gesetzt.
Modal anzeigen	Gibt an, ob das Formular in einem separaten Dialogfenster angezeigt wird. Zur Verwendung in Assistenten zum Einfügen von Daten.
Neuen Tabreiter verwenden	Das Formular wird in einem neuen Tabreiter angezeigt.
Konfiguration	Über die Konfiguration werden weitere Einschränkungen für die darzustellenden Tabellen und Spalten getroffen. Über die Auswahlliste **XML-Vorlage** stehen Vorlagen für die Definition der Konfigurationsdaten zur Verfügung. In der Sektion `Properties` werden spezielle Eigenschaften des Formulars übergeben, die bei der Formularentwicklung implementiert wurden. So werden beispielsweise für die Anzeige von Berichten über Oberflächenformulare der Name des auszuführenden Berichtes sowie die speziellen Berichtparameter in dieser Sektion übergeben.
Einzufügende Werte	Einzufügende Werte sind nur bei Oberflächenformularen vom Formulartyp **Edit** von Bedeutung. Damit werden Standardeinstellungen für Spalten definiert, die beim Einfügen eines Objektes zugewiesen werden. Die Angabe erfolgt in VB.Net-Syntax.

Verwandte Themen

- Formulardefinitionen und Formularvorlagen auf Seite 169
- Hierarchische Darstellung von Daten auf Zuordnungsformularen auf Seite 165
- Definieren von Einfügewerten auf Seite 147
- Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369

Formulardefinitionen und Formularvorlagen

Grundlage für die Darstellung der Oberflächenformulare bilden die Formulardefinitionen und Formularvorlagen. Formulardefinitionen enthalten Informationen über die auf den Formularen darzustellenden Daten, beispielsweise die Tabellen und Spalten sowie Beschriftungen von Tabreitern und Wurzelknoten in hierarchischen Zuordnungselementen

(ChildRelationControl, Mitgliederbaum) für die in den Formulararchiven (`*.Forms.*.vif`) definierten Formularvorlagen.

Detaillierte Informationen zum Thema

- Formularvorlagen auf Seite 170
- Formulardefinitionen auf Seite 172
- Konfigurationsdaten für die Darstellung von M:N-Beziehungen und Objektbeziehungen auf Formularen auf Seite 173

Formularvorlagen

Alle Formularvorlagen werden im Designer in der Kategorie **Benutzeroberfläche > Formulare > Formularvorlagen** abgebildet. Die Definition eigener Formularvorlagen ist in der Regel nicht erforderlich.

Um die Formularvorlage eines Oberflächenformulars anzuzeigen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie im Formulareditor das Oberflächenformular.
3. Wählen Sie in der Bearbeitungsansicht in der Ansicht **Eigenschaften** den Tabreiter **Formularvorlage**.

Tabelle 44: Eigenschaften der Formularvorlage

Eigenschaft	Bedeutung
Formularquelle	Quelle der Formatvorlage. Zulässig sind die Werte: • **Form**: Zur Anzeige eines Formulars aus einem Formulararchiv. • **Assembly**: Zur Anzeige von Steuerelementen. Es ist nicht notwendig ein Formular zu bauen, sondern das Steuerelement wird direkt als Formular angezeigt.
Name des Assemblies	Name des Assemblies, welches des Steuerelement enthält.
Klasse	Vollständiger Typ des Steuerelements.
Name der Formularvorlage	Der Name des Formularvorlage wird benötigt, um die Formularvorlage aus den Formulararchiven zu laden. TIPP: Der Name der Formularvorlage wird in den Administrationswerkzeugen als zusätzliche Menüführungsinformation angezeigt.
Formulararchiv	Name des Formulararchivs (`Forms.*.vif`) , in dem die Formu-

Eigenschaft	Bedeutung
	larvorlage enthalten ist.
Beschreibung	Nähere Beschreibung zur Formularvorlage.
Alternative Formularvorlage	Für die Darstellung eines Oberflächenformulars kann es notwendig sein, verschiedene Formularvorlagen einzusetzen, beispielsweise für die Anzeige im One Identity Manager-Webfrontend oder in den Administrationswerkzeugen. Um nicht für jede Formularvorlage eine eigene Formulardefinition und ein eigenes Oberflächenformular anzulegen, kann eine Verkettung der Formularvorlagen erfolgen. Dazu wird einer Formularvorlage eine alternative Formularvorlage zugeordnet. Diese alternative Formularvorlage wird genutzt, wenn die Bedingungen für die Anzeige der ursprünglichen Formularvorlage nicht erfüllt sind. Für die Darstellung eines Oberflächenformulars wird die referenzierte Formulardefinition ermittelt. Es wird überprüft, ob das in der Formulardefinition angegebene Formularvorlage in den Formulararchiven vorhanden ist und ob diese Formularvorlage für den gewünschten Anzeigezweck gekennzeichnet ist. Sind diese Bedingungen nicht erfüllt, wird die Prüfung für die angegebene alternative Formularvorlage ausgeführt. Zur Darstellung des Oberflächenformulars wird dann die erste alternative Formularvorlage verwendet, welche die Bedingungen erfüllt.
Formulartyp	Typ des Formulars.
Freigeschaltet für	Gibt an , für welchen Einsatzzweck eine Formularvorlage geeignet ist. Zulässige Werte sind: • Anzeigbar in der grafischen Benutzeroberfläche • Anzeigbar in Webapplikationen • TimeTrace wird unterstützt • Mehrfachobjektbearbeitung möglich • Verzögerter Operationen einstellbar • Anwendungsserver wird nicht unterstützt

Tabelle 45: Formulartypen und ihr Einsatzzweck

Formulartyp	Einsatzzweck
Info (I)	Formulare vom Typ **Info** dienen nur zur Anzeige von Informationen. Datenänderungen auf diesen Formularen können nicht gespeichert werden. Im Direktbearbeitungsmodus werden diese Formulare bei der automatischen Formularauswahl übersprungen.
Edit (E)	Formulare vom Typ **Edit** werden zur Bearbeitung von Daten eingesetzt. Im Direktbearbeitungsmodus wird bei der automatischen

Formulartyp	Einsatzzweck
	Formularauswahl das erste Formular mit diesem Formulartyp geladen.
Grid (G)	Formulare vom Typ **Grid** werden zur Anzeige der Daten in Tabellenform eingesetzt.
MemberRelation (M))	Formulare vom Typ **MemberRelation** werden zur Darstellung der Daten in einer Zuordnungsliste (M:N-Beziehungen) eingesetzt.
Report (R)	Formulare vom Typ **Report** werden zur Anzeige der Daten in Berichtsformular eingesetzt.
Virtual (V)	Formulare vom Typ **Virtual** werden nicht im Formularmenü angeboten. Der Formulartyp wird genutzt, um die Editoren im Designer anzuzeigen.
Wizard (W)	Formulare vom Typ **Wizard** werden zum Einfügen von Daten mittels Assistenten eingesetzt. Die Anzeige der Formulare erfolgt in einem modalen Dialogfenster.

Verwandte Themen

- Formulardefinitionen auf Seite 172
- Hierarchische Darstellung von Daten auf Zuordnungsformularen auf Seite 165

Formulardefinitionen

Die Formulardefinitionen werden im Designer in der Kategorie **Benutzeroberfläche > Formulare > Formulardefinitionen** abgebildet. Die Definition eigener Formulardefinitionen ist in der Regel nicht erforderlich.

Um die Formularvorlage eines Oberflächenformulars anzuzeigen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie im Formulareditor das Oberflächenformular.
3. Wählen Sie in der Bearbeitungsansicht in der Ansicht **Eigenschaften** den Tabreiter **Formulardefinition**.

Tabelle 46: Eigenschaften einer Formulardefinition

Eigenschaft	Bedeutung
Name der Formulardefinition	Name der Formulardefinition. Der Name dient zur Anzeige der Formulardefinition im Designer.
Formularvorlage	Name der Formularvorlage, die aus den Formararchiven zu laden ist. Eine Formularvorlage kann von mehreren Formulardefinitionen

Eigenschaft	Bedeutung
	genutzt werden, wie beispielsweise die Formularvorlage zur Darstellung von Mitgliederbäumen oder die Formularvorlage zur Anzeige von Berichten. Um eine neue Formularvorlage für die Formulardefinition einzubinden, verwenden Sie die Schaltfläche neben dem Eingabefeld.
Basisformular für Formularfolgen	Über die Angabe einer Formulardefinition als Basis für Formularfolgen kann eine Gruppierung der Formulardefinitionen einer Objektdefinition erreicht werden. Alle Formulardefinitionen einer Formularfolge erhalten dasselbe Basisformular. Die Definition des Oberflächenformulars erfolgt dann nur für dieses Basisformular. Wird das Oberflächenformular zu Anzeige geladen, dann werden über die referenzierte Formulardefinition alle weiteren Formulardefinitionen der Formularfolge geladen. Innerhalb der Formularfolge kann beliebig navigiert werden, ohne den Gültigkeitsbereich des Oberflächenformulars zu verlassen.
Beschreibung	Nähere Beschreibung zur Formulardefinition.
Konfiguration	Über die Konfigurationsdaten werden die Formulareigenschaften beschrieben. Die Definition der Formulareigenschaften erfolgt in XML-Notation.
Benötigte Tabellen	Einer Formulardefinition können zusätzlich Tabellen zugewiesen sein, die zur Darstellung der Daten verwendet werden. HINWEIS: Ist eine der aufgeführten Tabellen durch Präprozessorbedingungen deaktiviert, dann gilt die Formulardefinition ebenfalls als deaktiviert und das damit verbundene Oberflächenformular wird in der Benutzeroberfläche nicht angezeigt.

Verwandte Themen

- Formularvorlagen auf Seite 170
- Konfigurationsdaten für die Darstellung von M:N-Beziehungen und Objektbeziehungen auf Formularen auf Seite 173

Konfigurationsdaten für die Darstellung von M:N-Beziehungen und Objektbeziehungen auf Formularen

Über die Konfigurationsdaten der Formulardefinition werden die Formulareigenschaften beschrieben. Die Definition der Formulareigenschaften erfolgt in XML-Notation.

Beispiel: Aufbau der Konfigurationsdaten

```
<DialogFormDefinition FormatVersion="1.0">
     <ComponentDefinitions>
          <ComponentDefinition Name="TabPage1" Type="VI.Components.TabPage">
               <Properties>
                    <Property Name="Caption" Value="Department"/>
                    <Property Name="CaptionTranslationSource"
                    Value="DatabaseSchema" />
               </Properties>
          </ComponentDefinition>
          ...
          <ComponentDefinition Name="MemberRelation1"
          Type="VI.Components.MemberRelation">
               <Properties>
                    <Property Name="MNBaseColumnName" Value="UID_ADSGroup"
                    IsMandatory="True" />
                    <Property Name="MNTableName"
                    Value="DepartmentHasADSGroup" IsMandatory="True" />
                    ...
               </Properties>
          </ComponentDefinition>
          ...
     </ComponentDefinitions>
</DialogFormDefinition>
```

Darstellung von Beziehungen

Tabelle 47: Eigenschaften für die Definition von Beziehungen

Komponente	Eigenschaft	Bedeutung
Alle		Gültig für alle Abbildungen.
	WhereClause	Einschränkende Bedingung, die auf die dargestellten Objekte (Member, Child) angewendet wird. In der `WhereClause` kann der Ausdruck `%Spalte%` verwendet

Komponente	Eigenschaft	Bedeutung
		werden, um Werte des Basisobjekts zu referenzieren. Es sind $-Ausdrücke zulässig, um vom Basisobjekte aus andere Werte zu erreichen, beispielsweise `$FK(UID_ADSContainer).UID_ADSDomain$` Beispiel: `<Property Name="WhereClause" Value="IsITShopOnly=0" />`
	EditWhereClause	Bedingung zur Bearbeitung. Die Elemente, die der Bedingungen entsprechen, können bearbeitet werden. Alle anderen Elemente werden ebenfalls angezeigt, sind jedoch nicht bearbeitbar. Beispiel: `<Property Name="EditWhereClause" Value="XMarkedForDeletion=0" />`
MemberRelation 1-MemberRelationN		Darstellung von M:N-Beziehungen. Beispiel: `<ComponentDefinition Name="MemberRelation1" Type="VI.Components.MemberRelation">`
	MNTableName	M:N-Tabelle. Beispiel: `<Property Name="MNTableName" Value="OrgHasADSGroup" />`
	MNBaseColumnName	Spalte der M:N-Tabelle, die auf das Basisobjekt zeigt. Beispiel: `<Property Name="MNBaseColumnName" Value="UID_ADSGroup" />`
	RootFilterTableName	Tabelle, mit der die zuweisbaren Elemente vom Benutzer gefiltert werden können. Falls definiert zeigt das Steuerelement eine Auswahlliste mit den Objekten dieser Tabelle. Wird

Komponente	Eigenschaft	Bedeutung
		beispielsweise bei der Zuweisung von Geschäftsrollen verwendet. Beispiel: `<Property Name="RootFilterTableName" Type="String" Value="OrgRoot" />`
	RootFilterWhereClause	Bedingung, um die Elemente der `RootFilterTableName` in der Auswahlliste weiter zu filtern. Beispiel: `<Property Name="RootFilterWhereClause" Type="String" Value="UID_OrgRoot in (select UID_OrgRoot from Org) and exists (select 1 from OrgRootAssign where IsDirectAssignmentAllowed = 1 and UID_OrgRoot=OrgRoot.UID_OrgRoot and UID_BaseTreeAssign='ADS-AsgnBT-ADSGroup')" />`
	RootFil-terMemberWhereClause	Bedingung, die nach der Auswahl eines Basisobjektes gebildet wird und der `WhereClause` angehängt wird. Die Bedingung muss immer einen Spaltenbezug auf das Basisobjekt enthalten. Beispiel: `<Property Name="RootFilterMemberWhereClause" Type="String" Value="UID_OrgRoot=N'%UID_OrgRoot%'" />`
	ShowExtendedProperties	Gibt an, ob für M:N-Tabellen mit zusätzlichen Spalten auf dem Zuweisungsformular ein zusätzlicher Kontextmenüeintrag **Erweiterte Eigenschaften** angeboten wird. Über den Kontextmenüeintrag wird zum Detailformular navigiert, um die zusätzlichen Eigenschaften zu bearbeiten. Beispiel:

Komponente	Eigenschaft	Bedeutung
		`<Property Name="ShowExtendedProperties" Value="True" />`
ChildRelation1-ChildRelationN		Darstellung von Parent-Child-Beziehungen. Beispiel: `<ComponentDefinition Name="ChildRelation1" Type="VI.Components.MemberRelation">`
	CRTableName	Tabelle, in der die Child-Objekte abgebildet werden. Beispiel: `<Property Name="CRTableName" Value="ADSAccount" />`
	CRColumnName	Fremdschlüssel der Child-Tabelle, der auf das Basisobjekt zeigt. Beispiel: `<Property Name="CRColumnName" Value="UID_Person" />`
	ShowForeign	Gibt an, ob die Fremdzuordnungen (Objekte, die einem anderen Objekt zugewiesen sind) angezeigt werden sollen. Beispiel: `<Property Name="ShowForeign" Value="True" />`
GenericRelation1-GenericN		Darstellung von dynamischen M:N-Beziehungen. Beispiel: `<ComponentDefinition Name="GenericRelation1" Type="VI.Components.MemberRelation">`
	MNTableName	M:N-Tabelle. Beispiel `<Property Name="MNTableName"`

Komponente	Eigenschaft	Bedeutung
		`Value="ADSPolicyAppliesTo"/>`
	MNBaseColumnName	Spalte der M:N-Tabelle, die auf das Basisobjekt zeigt. Beispiel: `<Property Name="MNBaseColumnName" Value="ObjectKeyAppliesTo" />`
	MNMembersColumnName	Spalte der M:N-Tabelle, die auf die Mitglieder zeigt. Beispiel: `<Property Name="MNMembersColumnName" Value="UID_ADSPolicy" />`
	MembersTableName	Tabelle, deren Objekte zugewiesen werden sollen. Beispiel: `<Property Name="MembersTableName" Value="ADSPolicy"/>`

Verwenden von Tabreitern

Sollen für die abgebildeten Beziehungen Tabreiter angezeigt werden, verwenden Sie die Komponente `TabPage`. In der Regel werden Tabreiter für Formulare verwendet, die mehrere Beziehungen abbilden, beispielsweise **FrmCommonTwoMemberRelation** oder **FrmCommonTwoChildRelation**. `TabPage1` bildet den Tabreiter für `Relation1`, `TabePage2` bildet den Tabreiter für `Relation2`.

Tabelle 48: Eigenschaften für die Definition von Tabreitern

Komponente	Eigenschaft	Bedeutung
TabPage1-TabPageN		Anzeige von 1- n Tabreitern, je anzuzeigender Beziehung. Beispiel: `<ComponentDefinition Name="TabPage1" Type="VI.Components.TabPage">`
	Caption	Beschriftung des Tabreiters. Als Wert können Tabellenbezeichnungen oder beliebige Zeichenketten verwendet werden. Beispiel: `<Property Name="Caption"`

Komponente	Eigenschaft	Bedeutung
		`Value="Department"/>`

	CaptionTranslationSource	Quelle für die Übersetzung der Tabreiterbeschriftung.
		`Value="DatabaseSchema"` ermittelt die Übersetzung der Tabellenbeschriftung aus unter `Caption` eingetragenen Tabelle des One Identity Manager Schemas.
		`Value="TranslationAddOnSource"` ermittelt die Übersetzung aus dem Textspeicher.
		Beispiel:
		`<Properties>`
		`<Property Name="Caption" Value="Department"/>`
		`<Property Name="CaptionTranslationSource" Value="DatabaseSchema" />`
		`</Properties>`
		`<Properties>`
		`<Property Name="Caption" Value="is member of"/>`
		`<Property Name="CaptionTranslationSource" Value="TranslationAddOnSource" />`
		`</Properties>`

Verwandte Themen

- Formulare für kundenspezifische Erweiterungen auf Seite 163
- Formulardefinitionen auf Seite 172

Bearbeiten von Überblickformularen

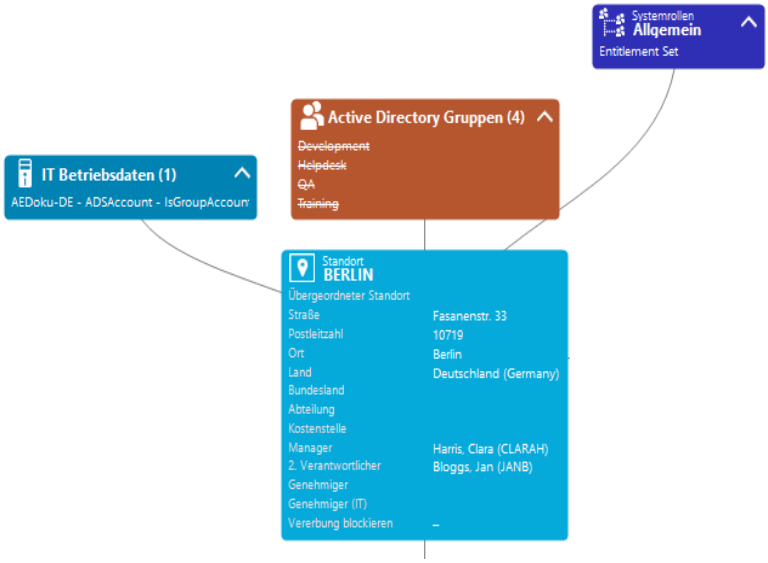
Zur Anzeige der Überblickformulare in der Benutzeroberfläche wird ein spezielles Steuerelement verwendet. Die Konfiguration der darzustellenden Informationen auf einem Überblickformular erfolgt über Menüeinträge. Die Menüeinträge werden auf dem Überblickformular als Formularelemente dargestellt, die miteinander verbunden sind. In der Oberflächenkonfiguration wird dazu eine hierarchische Struktur von Menüeinträgen erstellt.

Die Basis bildet ein Menüeintrag mit der Eintragsart **zentrales Formularelement**. Dieser Menüeintrag bestimmt das zentrale Element auf einem Überblickformular. Zur Anzeige in einer Anwendung muss ein Oberflächenformular eingerichtet werden, welches auf diesen Menüeintrag verweist. Das zentrale Formularelement wird immer im Zentrum des Überblickformulars angezeigt.

Unterhalb des Menüeintrags für das zentrales Formularelement werden die weiteren Menüeinträge als fixe Menüeinträge, datenabhängige Menüeinträge, Links oder Statistiken eingerichtet. Diese Menüeinträge werden als zusätzliche Formularelemente auf dem Überblickformular um das zentrale Formularelement gruppiert.

Farbe und Positionierung der Formularelemente auf dem Überblickformular sowie die dargestellten Eigenschaften werden über Layoutinformationen der Menüeinträge bestimmt und können auch für die mitgelieferten Standardüberblickformulare angepasst werden.

Abbildung 15: Beispiel für Formularelemente auf einem Überblickformular



In der Kopfzeile eines Formularelementes werden der Anzeigetext des Menüeintrags, der Anzeigetext des darzustellenden Objektes sowie das Symbol des Objektes dargestellt. Die weiteren Einträge repräsentieren die Eigenschaften des Objektes und deren Werte. Zu jeder Eigenschaft wird ein Tooltip mit der Beschreibung zur Verwendung angezeigt. Einige Einträge des Formularelementes werden farbig hervorgehoben, wenn Sie diese mit dem Mauszeiger berühren. Per Mausklick auf den Eintrag können Sie dann zum referenzierten Objekt wechseln.

Wird das Formularelement zur Abbildung von Listen verwendet, werden die Einträge mit ihrem Anzeigenamen angezeigt. Die Anzahl der Einträge wird in der Kopfzeile des Formularelementes eingeblendet. Zusätzlich wird in der Kopfzeile des Formularelementes ein Symbol zum Einblenden und Ausblenden der Einträge angezeigt. Für die Listeneinträge wird kein Tooltip angezeigt.

Tabelle 49: Symbol des Formularelementes

Symbol	Bedeutung
⌄	Listeneinträge einblenden.
⌃	Listeneinträge ausblenden.

HINWEIS: Auf den Überblickformularen werden die zum Löschen markierten Objekte ~~durchgestrichen~~ angezeigt.

Detaillierte Informationen zum Thema

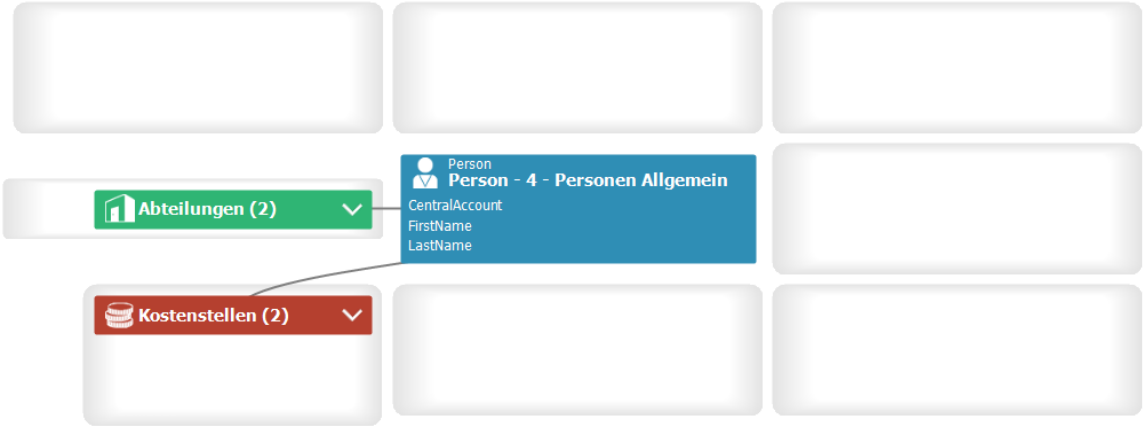
- Erstellen von Überblickformularen auf Seite 181
- Weitere Formularelemente auf Überblickformularen einfügen auf Seite 183
- Besonderheiten bei der Bearbeitung von Überblickformularen auf Seite 185
- Vorschau eines Überblickformulars während der Bearbeitung anzeigen auf Seite 188
- Layout der Formularelemente anpassen auf Seite 186
- Überblickformulare und Formularelemente deaktivieren auf Seite 188
- Formularelemente löschen auf Seite 189
- Überblickformulare löschen auf Seite 189

Erstellen von Überblickformularen

Der Überblicksformulareditor unterstützt Sie beim Erstellen von Überblickformularen. Der Überblicksformulareditor führt die folgenden Schritte zur Erstellung von Überblickformularen aus.

- Erstellen eines Menüeintrags mit der Eintragsart **zentrales Formularelement**.
- Erstellen der weiteren Menüeinträge unterhalb des zentralen Formularelementes.
- Erstellen eines Oberflächenformulars für das zentrale Formularelement.

Abbildung 16: Entwurfsansicht im Überblicksformulareditor



Um ein neues Überblickformular zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie die Aufgabe **Neue Übersichtsseite erstellen**.
3. Erfassen Sie die Basisdaten für das Überblickformular.

Tabelle 50: Basisdaten für ein Überblickformular

Eigenschaft	Bedeutung
Menüeintrag	Bezeichnung des Menüeintrags. Sie sollten hier möglichst sprechende Namen vergeben, die sich in den untergeordneten Strukturen fortsetzen.
Anzeigetext	Anzeigetext des Oberflächenformulars. Der Anzeigetext wird zur Darstellung des Oberflächenformulars in der Aufgabenansicht und im Formularkontextmenü der Benutzeroberfläche verwendet. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Objekt	Objektdefinition, für welche das Formular angezeigt werden soll.
Übergeordneter Menüeintrag	Übergeordneter Menüeintrag zur Gruppierung der Überblickformulare; in der Regel eine Menükategorie.
Produktzuordnung	Anwendung, in welcher das Formular angezeigt wird.
Gruppenzuordnung	Berechtigungsgruppe, für welche das Formular angezeigt wird.
Darzustellende Spalten	Spalten, die auf dem zentralen Formularelement angezeigt werden. TIPP: Über den Link **Beschriftung anzeigen** wechseln

Eigenschaft	Bedeutung
	Sie zwischen Anzeigetexten der Spalten und technischen Namen.

4. Um das Überblickformular zu erstellen, klicken Sie **OK**.

   Der Entwurf des Überblickformulars wird im Überblicksformulareditor angezeigt. Sie können das Überblickformular weiter bearbeiten.

Verwandte Themen

- Weitere Formularelemente auf Überblickformularen einfügen auf Seite 183
- Besonderheiten bei der Bearbeitung von Überblickformularen auf Seite 185
- Vorschau eines Überblickformulars während der Bearbeitung anzeigen auf Seite 188
- Layout der Formularelemente anpassen auf Seite 186
- Überblickformulare und Formularelemente deaktivieren auf Seite 188
- Menüführung der Benutzeroberfläche auf Seite 124

Weitere Formularelemente auf Überblickformularen einfügen

Um weitere Formularelemente auf einem Überblickformular einzufügen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie das Überblickformular und wählen Sie die Aufgabe **Übersichtseite bearbeiten**.
3. Wählen Sie Ansicht **Objektbeziehungen**.

   Angezeigt werden alle Fremdschlüsselbeziehungen vom Objekt (FK), Fremdzuordnungen des Objektes (CR) und Zuordnungsbeziehungen vom Objekt (M:N).
4. Wählen Sie die Objektbeziehung, die Sie darstellen möchten, und ziehen Sie diese per Drag and Drop auf ein Element in der Elementregion der Entwurfsansicht.
5. Wählen Sie beim Einfügen die Art des Menüeintrags, der erzeugt werden soll. Zur Auswahl stehen:
   - **Listenelement erstellen**: Es wird ein fixer Menüeintrag mit vordefinierten Listeneigenschaften erstellt.
   - **Datenelement erstellen**: Es wird ein datenabhängiger Menüeintrag erstellt.
   - **Listenelement referenzieren**: Es wird Menüeintrag mit dem Eintragsart **Link** mit einem Verweis auf einen Menüeintrag zur Darstellung als Liste

erstellt.

- **Datenelement referenzieren**: Es wird Menüeintrag mit dem Eintragsart **Link** mit einem Verweis auf einen datenabhängigen Menüeintrag erstellt.

Die Stammdaten des Menüeintrags werden automatisch durch den Überblicksformulareditor erzeugt. Das Formularelement wird in der Entwurfsansicht des Überblicksformulareditor angezeigt.

Beim Erstellen von Formularelemente über die Kontextmenüs **Listenelement referenzieren** und **Datenelement referenzieren** gelten folgende Besonderheiten.

- Es werden die Referenzeinträge unterhalb des Menüeintrags `InfoSheets.QIM.Links` verwendet.
- Sind die benötigten Referenzeinträge noch nicht vorhanden, werden neue Referenzeinträge mit den Bezeichnungen `InfoSheet.List.<Tabelle>` beziehungsweise `InfoSheet.Node.<Tabelle>` erzeugt.
- In der Bedingung des Referenzeintrags wird eine Variable `%<Tabelle>WhereClause%` verwendet.
- Am Formularelement wird eine Variable mit dem Variablentyp **Text** erzeugt. Dieser Variablen wird am Formularelement eine als Where-Klausel formulierte Bedingung zugewiesen. Die Bedingung können Sie bei Bedarf weiter bearbeiten. Bearbeiten Sie die Variable in der Bearbeitungsansicht des Überblicksformulareditor in der Ansicht **Variablendefinition**.

TIPP: Über das Kontextmenü **Element erstellen** können Sie weitere Menüeinträge, Links oder Statistiken als Formularelemente in der Entwurfsansicht des Überblicks-formulareditor erzeugen. In diesem Fall erfassen Sie manuell die Stammdaten für die Menüeinträge, Links oder Statistiken.

Verwandte Themen

- Erstellen von Überblickformularen auf Seite 181
- Besonderheiten bei der Bearbeitung von Überblickformularen auf Seite 185
- Vorschau eines Überblickformulars während der Bearbeitung anzeigen auf Seite 188
- Menüführung der Benutzeroberfläche auf Seite 124
- Verwenden von Links in der Menüführung auf Seite 148
- Verwenden von Variablen in der Menüführung auf Seite 149
- Einbinden von Statistiken in die Benutzeroberfläche auf Seite 196

Besonderheiten bei der Bearbeitung von Überblickformularen

Besonderheiten eines Oberflächenformulars für das zentrale Formularelement eines Überblickformulars

- Das Oberflächenformular wird mit der Formulardefinition **VI_ElementNavigation** erzeugt. Diese Formulardefinition stellt das Steuerelement zur Darstellung des Überblickformulars in der Benutzeroberfläche zur Verfügung.
- In den Konfigurationsdaten des Oberflächenformulars wird in der Sektion `Properties` der Name des zentralen Formularelementes eingetragen.

Beispiel:

```
<DialogSheetDefinition FormatVersion="1.0">
     <Properties>
          <Property Name="OverviewNode">VI_Person_Person_
          Overview</Property>
     </Properties>
</DialogSheetDefinition>
```

Besonderheiten zur Abbildung von Listen auf einem Überblickformular

Wird ein Formularelement zur Abbildung von Listen verwendet, werden die Einträge mit ihrem Anzeigenamen angezeigt. Per Mausklick auf den Eintrag können Sie dann zum referenzierten Objekt wechseln.

Um zu verhindern, dass zum referenzierten Objekt navigiert werden kann, setzen Sie am Menüeintrag den Konfigurationsschalter auf den Wert **Formulare der Ergebnisliste ignorieren**. Dies bietet sich beispielsweise an, wenn für die einzelnen Objekte der Ergebnisliste keine Formulare definiert sind. Anderenfalls wird ein leeres Formular angezeigt.

Verwandte Themen

- Erstellen von Überblickformularen auf Seite 181
- Weitere Formularelemente auf Überblickformularen einfügen auf Seite 183
- Layout der Formularelemente anpassen auf Seite 186
- Eigenschaften von Oberflächenformularen auf Seite 167
- Formulare für kundenspezifische Erweiterungen auf Seite 163

Layout der Formularelemente anpassen

Farbe und Positionierung der Formularelemente auf dem Überblickformular sowie die dargestellten Eigenschaften werden über Layout-Informationen der Menüeinträge bestimmt. Sie können diese Eigenschaften auch für die vordefinierten Überblickformulare anpassen.

Um die Layout-Informationen eines Formularelementes anzupassen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie das Überblickformular und öffnen Sie dieses im Überblicksformulareditor.
3. Wählen Sie das Formularelement in der Entwurfsansicht aus.
4. Wählen Sie in der Ansicht **Eigenschaften** den Tabreiter **Layout** und passen Sie die Eigenschaften an.

Tabelle 51: Layout-Informationen für Formularelemente

Eigenschaft	Bedeutung
Ausrichtung	Positionierung des Formularelementes auf dem Überblick-formular. Die Ausrichtung des zentralen Formularelementes ist nicht konfigurierbar. Das zentrale Formularelement wird immer im Zentrum des Überblickformulars angezeigt. Alle untergeordneten Menüeinträge werden entsprechende der Ausrichtung um dieses zentrale Formularelement angeordnet.
Hintergrundfarbe	Farbe des Formularelementes für die Darstellung auf dem Überblickformular fest. Die Hintergrundfarbe des zentralen Formularelementes ist nicht konfigurierbar. Bei der Einrich-tung eines Links wird die Hintergrundfarbe des referenzierten Menüeintrags zum Link übernommen.
max. Anzahl gleicher Elemente	Definiert ein Menüeintrag eine Liste von Einträgen, so wird jeder Eintrag der Ergebnisliste eines Menüeintrags mit einem separaten Formularelement dargestellt. Legen Sie die maximale Anzahl gleicher Einträge fest, bis zu welcher Anzahl von Einträgen diese in separaten Formu-larelementen dargestellt werden. Wird die Anzahl überschrit-ten, werden die Einträge zu einer Liste zusammengefasst und in einem Formularelement dargestellt werden. Eventuell angegebene Spalten werden in diesem Fall nicht dargestellt. Die Einträge werden mit ihrem Anzeigemuster angezeigt. Die Anzahl der Einträge wird in der Kopfzeile des Formu-larelementes eingeblendet. Zusätzlich wird in der Kopfzeile des Formularelementes ein Symbol zum Ein- und Ausblenden der Einträge angezeigt.

Eigenschaft	Bedeutung
	TIPP: Wenn Sie im Anzeigemuster einer Liste maximal zwei Spaltennamen verwenden, dann können Sie im Anzeigemuster einen Tabulator einsetzen, um eine zweispaltige Ausrichtung zu erreichen.
Darzustellende Spalten	Spalten der gültigen Objektdefinition im Formularelement angezeigt werden. Für das zentrale Formularelement beziehen sich die Spalten auf die Objektdefinition des zugehörigen Oberflächenformulars. Die weiteren Formularelemente erhalten ihre gültige Objektdefinition über die Menüeinträge. Bei der Einrichtung eines Links werden die ausgewählten Spalten des referenzierten Menüeintrags initial zum Link übernommen. Die Darstellungsreihenfolge der Eigenschaften in einem Formularelement entspricht der festgelegten Spaltenreihenfolge im Menüeintrag. TIPP: Um in einem Formularelement eine Trennlinie zur optischen Trennung der Informationen anzuzeigen, fügen Sie als darzustellende Spalte ein Minus-Zeichen (**-**) ein. In der Spaltendefinition können Sie Skripte verwenden, um den Anzeigewert der Spalte zu beeinflussen. Mit einem langen Mausklick auf den Spaltennamen wird die Spaltendefinition zur Bearbeitung freigeschaltet. Erweitern Sie die Spaltendefinition folgendermaßen: `Spalte[S(Name des Skriptes)]` Enthalten Spalten eine Webadresse und soll zu dieser Webadresse navigiert werden, dann verwenden Sie den Formatierungstyp **Uri** für die Spaltendefinition. Damit wird auf dem Überblicksformular ein Link angezeigt.

Gestaltung der Kopfzeile eines Formularelementes

In der Kopfzeile eines Formularelementes werden der Anzeigetext des Menüeintrags, der Anzeigetext des darzustellenden Objektes sowie das Symbol des Menüeintrags dargestellt.

TIPP: Bei Klick auf den Anzeigetext in der Kopfzeile eines Formularelementes kann ein Oberflächenformular geöffnet werden.

- Weisen Sie dazu einem fixen Menüeintrag, der unterhalb des zentralen Formularelementes angeordnet ist, das Oberflächenformular zu. Das Oberflächenformular muss sich auf das Objekt des zentralen Formularelementes beziehen, wie beispielsweise ein Formular für Zuweisungen zu diesem Objekt.
- Zur Steuerung der Anzeige setzen Sie in der Ansicht der Formularzuweisungen die Option **Anzeige aus der Navigationsstruktur**.

Verwandte Themen

- Vordefinierte Formatierungstypen verwenden auf Seite 88
- Oberflächenformulare an Menüeinträge zuweisen auf Seite 161

Vorschau eines Überblickformulars während der Bearbeitung anzeigen

Um eine Vorschau eines Überblickformulars zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie das Überblickformular und öffnen Sie dieses im Überblicksformulareditor.
3. Wählen Sie in der Symbolleiste des Überblicksformulareditor in der Auswahlliste **Tabelle** die Tabelle des zentralen Formularelementes und wählen Sie in der Auswahlliste **Objekt** ein konkretes Objekt, mit dem Sie die Vorschau erzeugen möchten.

   HINWEIS: Wählen Sie in der Auswahlliste **Objekt** den Eintrag **Ohne Objekt** um die Vorschau zu beenden.

Überblickformulare und Formularelemente deaktivieren

Bei Bedarf können Sie einzelne vordefinierte Überblickformulare oder einzelne Formularelemente auf einem Überblickformular deaktivieren, um deren Anzeige in der Benutzeroberfläche zu verhindern. Die Deaktivierung bleibt auch bei Schemainstallationen erhalten.

Um ein Überblickformular zu deaktivieren

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie das Überblickformular und starten Sie den Formulareditor über die Aufgabe **Oberflächenformular bearbeiten**.
3. Setzen Sie die Option **Deaktiviert**.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Um ein Formularelement auf einem Überblickformular zu deaktivieren

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.

2. Wählen Sie das Überblickformular und starten Sie den Überblicksformulareditor über die Aufgabe **Übersichtseite bearbeiten**.
3. Wählen Sie in der Entwurfsansicht das Formularelement.
4. Setzen Sie die Option **Deaktiviert**.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Zusätzlich können Überblickformulare oder einzelne Formularelemente über Präprozessorbedingungen deaktiviert sein.

HINWEIS: Einen Überblick über die vorhandenen Präprozessorabhängigkeiten erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Präprozessorabhängigkeiten**.

Verwandte Themen

- Formularelemente löschen auf Seite 189
- Überblickformulare löschen auf Seite 189

Formularelemente löschen

Um ein Formularelement auf einem Überblickformular zu löschen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie das Überblickformular und wählen Sie die Aufgabe **Übersichtseite bearbeiten**.
3. Wählen Sie in der Entwurfsansicht des Überblicksformulareditors das Formularelement und wählen Sie das Kontextmenü **Element löschen**.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Überblickformulare löschen auf Seite 189
- Überblickformulare und Formularelemente deaktivieren auf Seite 188

Überblickformulare löschen

Um ein Überblickformular zu löschen, löschen Sie das Oberflächenformular, den Menüeintrag für das zentrale Formularelement und die untergeordneten Menüeinträge für die weiteren Formularelemente.

Um ein Überblickformular zu löschen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Übersichtsformulare**.
2. Wählen Sie das Überblickformular und wählen Sie die Aufgabe **Oberflächenformular bearbeiten**.
3. Wählen Sie das Kontextmenü **Löschen**.
4. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   Die Menüeinträge werden geladen und im Oberflächeneditor zur Bearbeitung angezeigt.
5. Wählen Sie in der Übersicht der Menüführung den Menüeintrag, der mit dem Überblickformular verbunden war.
6. Um den Menüeintrag und seine untergeordneten Menüeintrag zu löschen, wählen Sie das Kontextmenü **Löschen**.
7. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Formularelemente löschen auf Seite 189
- Überblickformulare und Formularelemente deaktivieren auf Seite 188

Statistiken im One Identity Manager

Im Infosystem des One Identity Manager erhalten Sie einen schnellen Überblick über die Systemsituation. Die Statistiken werden in zyklischen Abständen neu berechnet und in der Benutzeroberfläche über verschiedene Diagramme visualisiert. Mit dem One Identity Manager werden bereits Statistikdefinitionen mitgeliefert. Weitere kundenspezifische Statistikinformationen können Sie bei Bedarf im Designer erstellen.

Die Bereitstellung von Statistiken erfordert die folgenden Schritte:

1. Erstellen der Statistikdefinition
2. Einbinden der Statistikdefinition in die Benutzeroberfläche

Detaillierte Informationen zum Thema

- Statistikdefinitionen erstellen und bearbeiten auf Seite 191
- Statistikdefinitionen deaktivieren auf Seite 196
- Einbinden von Statistiken in die Benutzeroberfläche auf Seite 196
- Diagrammtypen für die Darstellung von Statistiken auf Seite 200
- Beispiele für Statistikdefinitionen auf Seite 204

Statistikdefinitionen erstellen und bearbeiten

Grundlage für das Infosystem ist die Definition von Statistiken. Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar. Die Standardkonfiguration wird bei Bearbeitung in einen Konfigurationsspeicher verschoben. Sie können Änderungen aus dem Konfigurationsspeicher zurückholen und somit die Standardkonfiguration wiederherstellen.

Um eine Statistikdefinition zu erstellen oder zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Statistikdefinitionen**.
2. Wählen Sie die Statistikdefinition und wählen Sie die Aufgabe **Stammdaten bearbeiten**.

   - ODER -

   Um eine neue Statistikdefinition zu erstellen, wählen Sie **Objekt > Einfügen**.
3. Erfassen Sie auf dem Tabreiter **Allgemein** die allgemeinen Eigenschaften.
4. Erfassen Sie auf dem Tabreiter **Abfragen** die Messwertabfrage.
5. Prüfen Sie die Abfragen und die Statistikdefinition auf Fehler.
   - Über die Schaltfläche **Überprüfen** können Sie die einzelnen Abfragen prüfen.

     Die SQL-Abfrage wird und das Ergebnis wird auf Gültigkeit geprüft. Dabei werden Anzahl der Spalten, Bezeichnung der Spalten sowie die Datentypen geprüft.
   - Über die Schaltfläche **Vollständiger Test** prüfen Sie die gesamte Statistikdefinition.

     Für die Prüfung wird die Statistikdefinition in der Datenbank gespeichert und eine Statistikberechnung simuliert. Nach der Simulation wird diese testweise Statistikberechnung wieder aus der Datenbank entfernt.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Detaillierte Informationen zum Thema

- Allgemeine Eigenschaften einer Statistikdefinition auf Seite 192
- Abfrage der Statistikmesswerte auf Seite 194
- Beispiele für Statistikdefinitionen auf Seite 204

Allgemeine Eigenschaften einer Statistikdefinition

Tabelle 52: Eigenschaften von Statistikdefinitionen

Eigenschaft	Bedeutung
Statistik	Name der Statistik.
Anzeigename	Anzeigename, mit dem die Statistikdefinition in den Einstellungen des Infosystems in den Administrationswerkzeuge angezeigt wird. Der Anzeigename bildet den Titel einer Statistik. Übersetzen Sie den eingegebenen Text über die Schaltfläche . HINWEIS: Ist beim Menüeintrag ein Anzeigetext eingetragen, dann überschreibt dieser den Anzeigenamen der Statistikdefinition.
Beschreibung	Beschreibung zur Statistikdefinition. Die Beschreibung wird in den Einstellungen des Infosystems in den Administrationswerkzeugen angezeigt. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Zeitplan der Berechnung	Wählen Sie den Zeitplan zur Berechnung der Statistikinformationen aus. Mitgeliefert werden die Zeitpläne **Statistiken berechnen**, **Statistiken wöchentlich berechnen** und **Statistiken monatlich am 1. berechnen**. HINWEIS: Die Zeitpläne zur Berechnung der Statistikinformationen aktivieren Sie im Designer in der Kategorie **Basisdaten > Allgemein > Zeitpläne**. Ausführliche Informationen zum Bearbeiten von Zeitplänen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.
Aggregatfunktion	Verwenden Sie die Aggregatfunktion, wenn die Abfrage der Messwerte mehrere Messwerte liefert, in der Statistik jedoch ein eindeutiger Wert angezeigt werden soll. Beispiel: Es soll die Anzahl der Personen ermittelt werden für ein Abteilungsleiter verantwortlich ist. Um eine Statistik mit der Gesamtanzahl der Personen aller Abteilungen anzuzeigen, für die eine Person verantwortlich ist, verwenden Sie die Aggregatfunktion `SUM`. Um pro Abteilung eine Statistik anzuzeigen, geben Sie keine Aggregatfunktion an.
Basisaggregatfunktion	Verwenden Sie die Basisaggregatfunktion, wenn aus der Abfrage der Basismesswerte kein eindeutiger Basiswert für die Statistikanzeige gebildet werden kann. HINWEIS:

Eigenschaft	Bedeutung
	• Aggregatfunktion und Basisaggregatfunktion werden nur ausgewertet, wenn die formulierte Messwertabfrage durch eine Bedingung auf den angemeldeten Benutzer eingeschränkt wird. • Aggregatfunktion und Basisaggregatfunktion werden nur für Statistiken berücksichtigt, die im Web Portal angezeigt werden.
Schwellwert grün	Schwellwertfaktor im Wertebereich von **0** bis **1**. Mithilfe dieses Schwellwertfaktors wird der prozentuale Anteil des Basismesswertes ermittelt, der einen ordnungsgemäßen Zustand repräsentiert.
Schwellwert rot	Schwellwertfaktor im Wertebereich von **0** bis **1**. Mithilfe dieses Schwellwertfaktors wird der prozentuale Anteil des Basismesswertes ermittelt, der einen noch tolerierten Zustand repräsentiert.
Maßeinheit	Maßeinheit für die gemessenen Werte. Die Maßeinheit wird in den Statistiken im Infosystem angezeigt. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Zeitskalierung	Für Statistikdefinitionen, die eine Zeitabfrage beinhalten (zum Beispiel Anzahl der neuen Personen in der letzten Woche), geben Sie hier die Anzeigegenauigkeit der Daten auf der Zeitachse an. Zulässige Werte sind **Stunde**, **Tag**, **Woche**, **Monat**, **Quartal** und **Jahr**.
Anzahl Messläufe Historie	Anzahl der Messläufe (außer dem aktuellen Messwert), die für die Darstellung des historischen Verlaufs gespeichert werden sollen. Soll nur der jeweils aktuellste Messwert erhalten bleiben, tragen Sie den Wert **0** ein.
Deaktiviert	Gibt an, ob die Statistikdefinition deaktiviert ist. Deaktivierte Statistikdefinitionen werden nicht berechnet.
Präprozessorbedingung	Statistikdefinitionen können Sie mit Präprozessorbedingungen versehen. Damit ist eine Statistikdefinition nur verfügbar, wenn die Präprozessorbedingung erfüllt ist.
Deaktiviert durch Präprozessor	Ist eine Statistikdefinition durch eine Präprozessorbedingung ausgeschaltet, dann wird durch den Database Compiler die Option gesetzt.
Sofortberechnung	Aktivieren Sie diese Option für Statistikdefinitionen, die zum Zeitpunkt der Anzeige im Web Portal berechnet werden sollen. Ist die Option nicht gesetzt, wird die Statistik innerhalb der Wartungsaufträge berechnet.

Eigenschaft	Bedeutung
Importierte Statistikdaten	Gibt an, ob diese Statistik zum Zeitpunkt der Anzeige berechnet wird (zur Verwendung im Web Portal). Ist die Option nicht gesetzt, wird die Statistik durch den DBQueue Prozessor asynchron berechnet.

Verwandte Themen

- Abfrage der Statistikmesswerte auf Seite 194
- Beispiele für Statistikdefinitionen auf Seite 204
- Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369

Abfrage der Statistikmesswerte

Tabelle 53: Eigenschaften zur Messwertabfrage

Eigenschaft	Bedeutung
Abfrage Messwerte	Zur Ermittlung der Statistikmesswerte geben Sie die vollständige Datenbankabfrage in SQL-Syntax an. Die Abfrage muss die Spalten `ElementName` und `ElementValue` als Ergebnis liefern. Für die Darstellung der Statistikinformationen im Web Portal können Sie optional die Spalten `ElementObjectKey`, `ElementObjectKey2` und `ElementValueZ` als Ergebnis ausgeben. Die Anzeigereihenfolge der Statistikmesswerte können Sie optional durch die Spalte `ElementOrder` beeinflussen. Ist die Spalte `ElementOrder` nicht vorhanden, wird alphabetisch nach der Spalte `ElementName` sortiert.
Abfrage Basismesswerte	Zur Ermittlung der Statistikmesswerte geben Sie die vollständige Datenbankabfrage in SQL-Syntax an. Die Abfrage muss die Spalten `ElementName` und `ElementValue` als Ergebnis liefern. Für die Darstellung der Statistikinformationen im Web Portal können Sie optional die Spalten `ElementObjectKey`, `ElementObjectKey2` und `ElementValueZ` als Ergebnis ausgeben. Die Anzeigereihenfolge der Statistikmesswerte können Sie optional durch die Spalte `ElementOrder` beeinflussen. Ist die Spalte `ElementOrder` nicht vorhanden, wird alphabetisch nach der Spalte `ElementName` sortiert. In den Eingabefeldern **Schwellwert grün** und **Schwellwert rot** angegebene Schwellwertfaktoren beziehen sich auf das Ergebnis der Spalte `ElementValue`. Für die Ermittlung des prozentualen Anteils des Basismesswertes wird dabei Ergebnis der Spalte `ElementValue` mit 100% angesetzt.

Eigenschaft	Bedeutung
	HINWEIS: Die Bezeichnung der Spalte `ElementName` in der Abfrage der Basismesswerte muss der Bezeichnung der Spalte `ElementName` in der Abfrage der Messwerte entsprechen.
Bedingung	Formulieren Sie eine Bedingung mit der die Darstellung der Statistikmesswerte auf den angemeldeten Benutzer eingeschränkt wird. Die Bedingung ist als gültige Where-Klausel für Datenbankabfragen zu formulieren und schränkt das Abfrageergebnis anhand der Spalte `ElementObjectKey` unter Verwendung der Variablen "`%UserUID%`" weiter ein. HINWEIS: Die Bedingung wird nur für Statistiken berücksichtigt, die im Web Portal angezeigt werden.

Beispiel: Schwellwertberechnung

Mithilfe der Schwellwertfaktoren wird der prozentuale Anteil des Basismesswertes ermittelt, der einen ordnungsgemäßen Zustand oder einen noch tolerierten Zustand repräsentiert.

Tabelle 54: Beispiele für die Ermittlung des Zustandes

Basis-messwert	Schwellwert Grün	Schwellwert Rot	prozentualer Anteil	Zustand
100	0,25	0,75	< = 25	ordnungs-gemäß
			>25 bis >75	toleriert
			>= 75	unerlaubt
	0,75	0,25	> = 75	ordnungs-gemäß
			<75 bis <25	toleriert
			<= 25	unerlaubt

Verwandte Themen

- Allgemeine Eigenschaften einer Statistikdefinition auf Seite 192
- Beispiele für Statistikdefinitionen auf Seite 204

Statistikdefinitionen deaktivieren

Bei Bedarf können Sie einzelne Statistikdefinitionen deaktivieren. Deaktivierte Statistikdefinitionen werden nicht berechnet. Die Deaktivierung für vordefinierte Statistikdefinitionen bleibt auch bei der Schemaaktualisierung erhalten.

Um eine Statistikdefinition zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Benutzeroberfläche > Statistikdefinitionen** die Statistikdefinition.
2. Wählen Sie in der Bearbeitungsansicht die Ansicht **Eigenschaften.**
3. Wählen Sie den Tabreiter **Eigenschaften** und setzen Sie die Option **Deaktiviert**.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Zusätzlich können Statistikdefinitionen über Präprozessorbedingungen deaktiviert sein.

Verwandte Themen

- Allgemeine Eigenschaften einer Statistikdefinition auf Seite 192

Einbinden von Statistiken in die Benutzeroberfläche

Um Statistiken in den Administrationswerkzeugen des One Identity Manager, wie beispielsweise dem Manager, darzustellen, müssen Sie die Statistik als kundenspezifischen Menüeintrag in die Benutzeroberfläche einbinden.

Typischerweise werden Statistiken im Manager unterhalb einer Kategorie im Menüeintrag **Infosystem** angezeigt. Kundenspezifische Menüeinträge für Statistiken, sollten Sie unterhalb eines solchen Infosystems einrichten. Alle Statistiken, die in einer Menüebene definiert sind, werden auf einem Formular angezeigt.

Berichte, die Sie im Report Editor oder im Manager erstellen, können Sie in den Statistiken anzeigen. Im Infosystem des Managers wird der Bericht bei Doppelklick auf die Kopfzeile der Statistik geöffnet.

Zusätzlich können Statistiken als Formularelemente in Überblickformulare eingebunden werden. Dazu verwenden Sie den Überblicksformulareditor.

> HINWEIS: Wenn Sie ein kundenspezifisches Infosystem einrichten, achten Sie darauf, dass der Menüeintrag, unter dem Sie die Statistiken definieren, mit den Konfigurationsschaltern **Benutzer darf Menüeintrag nicht öffnen** und **Menüeintrag immer aktivieren** gekennzeichnet ist.

Die Beschreibung der allgemeinen Eigenschaften für Menüeinträge finden Sie unter Allgemeine Eigenschaften der Menüeinträge auf Seite 139. Beachten Sie bei der Erstellung von Menüeinträgen die folgenden Eigenschaften.

Tabelle 55: Eigenschaften für Statistiken

Eigenschaft	Bedeutung
Eintragsart	Wählen Sie die Eintragsart **Statistik**.
Anzeigetext	Der hier eingegebene Anzeigetext überschreibt den Anzeigenamen der Statistikdefinition. Lassen Sie das Eingabefeld leer, wenn Sie den Anzeigenamen der Statistikdefinition verwenden möchten.
Statistik	Geben Sie die Statistikdefinition an, die dargestellt werden soll.
Diagrammtyp	Wählen Sie den Diagrammtyp mit dem die Statistik dargestellt werden soll.
Ausrichtung	Positionierung der Statistik auf dem Überblickformular. Diese Layoutinformation wird verwendet, wenn die Statistik als Formularelement auf einem Überblickformular verwendet wird.
Hintergrund	Hintergrundfarbe des Formularelementes auf dem Überblickformular. Diese Layoutinformation wird verwendet, wenn die Statistik als Formularelement auf einem Überblickformular verwendet wird.

Alle Menüeinträge, die in der Benutzeroberfläche einer Anwendung dargestellt werden sollen, müssen Sie der Anwendung und einer Berechtigungsgruppe zuweisen.

Verwandte Themen

- Diagrammtypen für die Darstellung von Statistiken auf Seite 200
- Beispiele für Statistikdefinitionen auf Seite 204
- Neue Menüeinträge erstellen auf Seite 136
- Berichte in Statistiken verwenden auf Seite 197
- Einfache Berichte in Statistiken verwenden auf Seite 199
- Erstellen von Überblickformularen auf Seite 181

Berichte in Statistiken verwenden

Berichte, die Sie im Report Editor erzeugen, können Sie für Statistiken im Infosystem des Managers anzeigen. Dazu müssen Sie die Benutzeroberfläche für die Anwendung **Manager** anpassen. Der Bericht wird bei Doppelklick auf die Kopfzeile der Statistik geöffnet.

Um einen Bericht für eine Statistik anzuzeigen

1. Erstellen Sie im Designer ein Oberflächenformular.
    a. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.

b. Wählen Sie die Aufgabe **Formulare bearbeiten**.

c. Wählen Sie den Menüeintrag **Formular > Einfügen**.

d. Bearbeiten Sie die Stammdaten des Oberflächenformulars.

Beachten Sie die folgenden Besonderheiten:

- Verwenden Sie die Formulardefinition **VI_Report**.

  Diese Formulardefinition ist zur Anzeige in der grafischen Benutzeroberfläche und zur Anzeige in Webapplikationen konfiguriert. Damit müssen Sie nur ein Oberflächenformular einrichten. Anhand des Einsatzzweckes wird dynamisch entschieden, welche der Formularvorlagen zur Darstellung des Oberflächenformulars genutzt wird.

- In den Konfigurationsdaten des Formulars übergeben Sie in der Sektion `Properties` die Bezeichnung des auszuführenden Berichtes (`DialogReport.ReportName`).

  Syntax:

```
<DialogSheetDefinition FormatVersion="1.0">
    <Properties>
        <Property Name="ReportName">ReportName aus Tabelle
        DialogReport</Property>
    </Properties>
</DialogSheetDefinition>
```

e. Weisen Sie das Oberflächenformular an die Anwendungen und an die Berechtigungsgruppen zu.

2. Erstellen Sie im Designer einen Menüeintrag.

   a. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.

   b. Wählen Sie im Oberflächeneditor den Menüeintrag für die Statistik, für die der Bericht angezeigt werden soll.

   c. Wählen Sie das Kontextmenü **Neu**.

   d. Bearbeiten Sie die Stammdaten des Menüeintrag.

   e. Weisen Sie den Menüeintrag an die Anwendung **Manager** und an die Berechtigungsgruppen zu.

3. Weisen Sie das Oberflächenformular an den Menüeintrag zu.

4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Berichte im Report Editor erstellen und bearbeiten auf Seite 443
- Oberflächenformulare erstellen auf Seite 157
- Oberflächenformulare an Menüeinträge zuweisen auf Seite 161

- Neue Menüeinträge erstellen auf Seite 136
- Einfache Berichte in Statistiken verwenden auf Seite 199

Einfache Berichte in Statistiken verwenden

Einfache Berichte, die Sie im Manager erzeugen, können Sie für Statistiken im Infosystem des Managers anzeigen. Dazu müssen Sie im Designer die Benutzeroberfläche für die Anwendung **Manager** anpassen. Im Infosystem des Managers wird der einfache Bericht bei Doppelklick auf die Kopfzeile der Statistik geöffnet.

Ausführliche Informationen wie Sie einfache Berichte im Manager erstellen, zum finden Sie im *One Identity Manager Administrationshandbuch für Berichtsabonnements*.

Um einen einfachen Bericht für eine Statistik anzuzeigen

1. Erstellen Sie im Designer ein Oberflächenformular.
   a. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
   b. Wählen Sie die Aufgabe **Formulare bearbeiten**.
   c. Wählen Sie den Menüeintrag **Formular > Einfügen**.
   d. Bearbeiten Sie die Stammdaten des Oberflächenformulars.

      Beachten Sie die folgenden Besonderheiten:

      - Verwenden Sie die Formulardefinition **VI_Report**.

        Diese Formulardefinition ist zur Anzeige in der grafischen Benutzeroberfläche und zur Anzeige in Webapplikationen konfiguriert. Damit müssen Sie nur ein Oberflächenformular einrichten. Anhand des Einsatzzweckes wird dynamisch entschieden, welche der Formularvorlagen zur Darstellung des Oberflächenformulars genutzt wird.
      - In den Konfigurationsdaten des Formulars übergeben Sie in der Sektion `Properties` die UID des einfachen Berichtes (`RPSReport.UID_RPSReport`).

        Syntax:

        ```
        <DialogSheetDefinition FormatVersion="1.0">
            <Properties>
                <Property Name="UIDRPSReport">UID_RPSReport aus der
                Tabelle RPSReport</Property>
            </Properties>
        </DialogSheetDefinition>
        ```
   e. Weisen Sie das Oberflächenformular an die Anwendung **Manager** und an die Berechtigungsgruppen zu.
2. Erstellen Sie im Designer einen Menüeintrag.

a. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung**.
b. Wählen Sie im Oberflächeneditor den Menüeintrag für die Statistik, für die der Bericht angezeigt werden soll.
c. Wählen Sie das Kontextmenü **Neu**.
d. Bearbeiten Sie die Stammdaten des Menüeintrag.
e. Weisen Sie den Menüeintrag an die Anwendung **Manager** und an die Berechtigungsgruppen zu.

3. Weisen Sie das Oberflächenformular an den Menüeintrag zu.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Oberflächenformulare erstellen auf Seite 157
- Oberflächenformulare an Menüeinträge zuweisen auf Seite 161
- Neue Menüeinträge erstellen auf Seite 136
- Berichte in Statistiken verwenden auf Seite 197

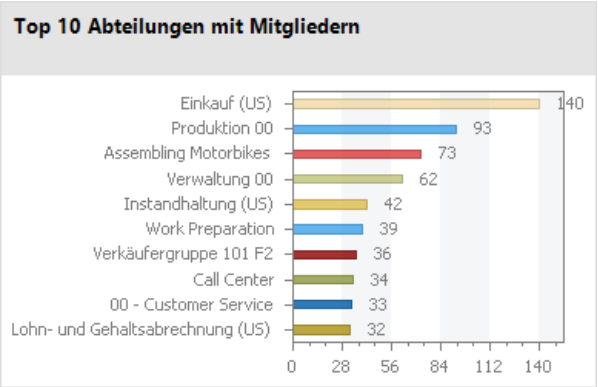
Diagrammtypen für die Darstellung von Statistiken

Für die Darstellung der Statistiken werden verschiedene Diagrammtypen zur Verfügung gestellt.

Balkendiagramm

Mit einem Balkendiagramm können Vergleiche zwischen Messwerten dargestellt werden. Der aktuelle Messwert der Spalte `ElementValue` und die Bezeichnung der Spalte `ElementName` werden als Diagrammbeschriftung angezeigt.

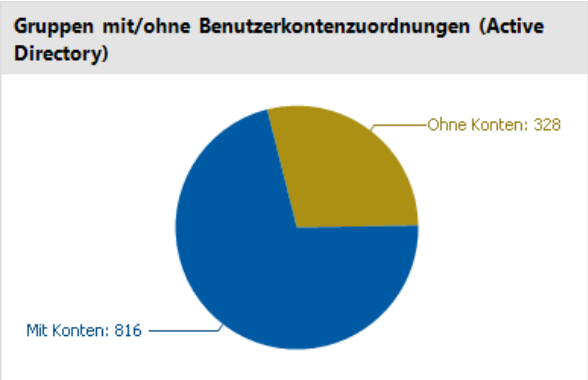
Abbildung 17: Beispiel für Balkendiagramm



Kreisdiagramm

Mit einem Kreisdiagramm wird der prozentuale Anteil der Messwerte am Basismesswert dargestellt. Der aktuelle Messwert der Spalte `ElementValue` und die Bezeichnung der Spalte `ElementName` werden als Diagrammbeschriftung angezeigt.

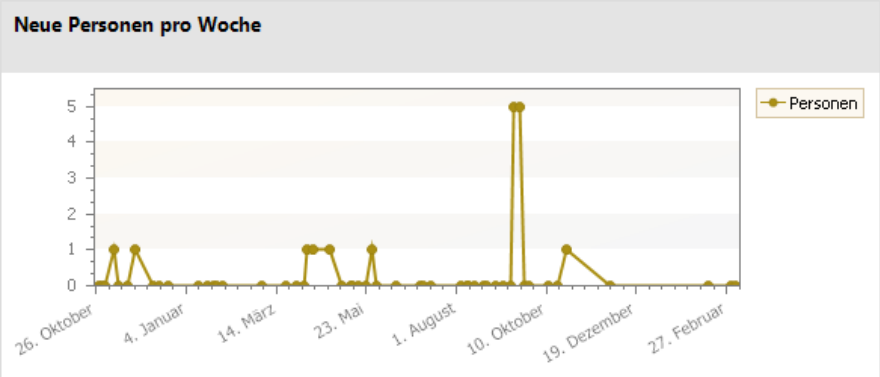
Abbildung 18: Beispiel für Kreisdiagramm



Liniendiagramm

Mit einem Liniendiagramm wird der Verlauf der Daten über einen bestimmten Zeitraum dargestellt. Die Skalierung der Zeitachse richtet sich nach der Zeitskalierung, die in der Statistikdefinition angegeben ist. Die Anzahl der Messwerte im Liniendiagramm ergibt sich aus der Anzahl der Messläufe für die Historie, die in der Statistikdefinition angegeben ist. Mit Mausklick auf einen Messpunkt wird ein Tooltip mit dem Messwert angezeigt.

Abbildung 19: Beispiel für Liniendiagramm



Ampel

Mit einem Ampeldiagramm wird der Systemzustand dargestellt. Der Zustand wird über Farben angezeigt. Wann welcher Zustand erreicht ist, ergibt sich aus den Schwellwertfaktoren, die in der Statistikdefinition angegeben sind.

Tabelle 56: Bedeutung der Farben

Farbe	Zustand
grün	ordnungsgemäß
gelb	toleriert
rot	unerlaubt

Der aktuelle Messwert der Spalte `ElementValue` und die Bezeichnung der Spalte `ElementName` werden als Diagrammunterschrift angezeigt.

Abbildung 20: Beispiel für Ampel

Tachometer

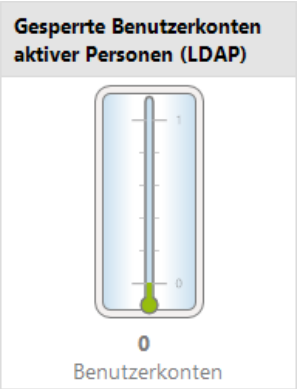
Mit einem Tachometer-Diagramm wird der Systemzustand detaillierter als mit einem Ampeldiagramm dargestellt. Zusätzlich wird der Basismesswert abgebildet. Der Zustand wird über Farben angezeigt. Wann welcher Zustand erreicht ist, ergibt sich aus den Schwellwertfaktoren, die in der Statistikdefinition angegeben sind. Der aktuelle Messwert der Spalte `ElementValue` und die Bezeichnung der Spalte `ElementName` werden als Diagrammunterschrift angezeigt.

Abbildung 21: Beispiel für Tachometer

Thermometer

Mit einem Thermometer-Diagramm wird der Systemzustand detaillierter als mit einem Ampeldiagramm dargestellt. Der Zustand wird über eine Farbskala neben dem Diagramm angezeigt. Wann welcher Zustand erreicht ist, ergibt sich aus den Schwellwertfaktoren, die in der Statistikdefinition angegeben sind. Der aktuelle Messwert der Spalte `ElementValue` und die Bezeichnung der Spalte `ElementName` werden als Diagrammunterschrift angezeigt.

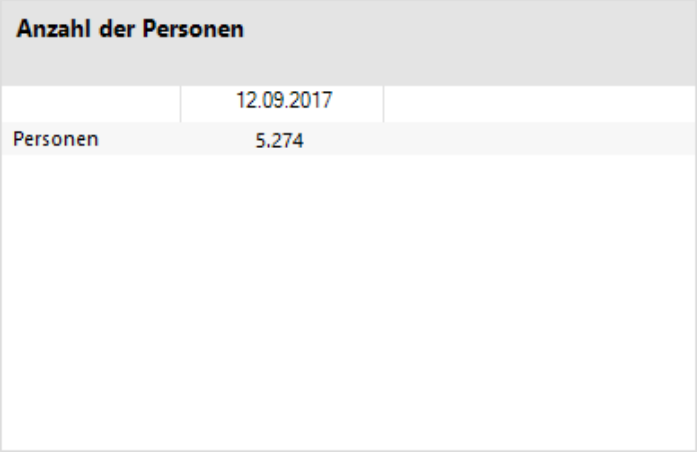
Abbildung 22: Beispiel für Thermometer



Tabelle

Mit diesem Diagrammtyp werden die Messwerte tabellarisch aufbereitet. Um den Verlauf der Daten über einen bestimmten Zeitraum darzustellen, ist in der Statistikdefinition die Anzahl der Messläufe für die Historie anzugeben.

Abbildung 23: Beispiel für Tabelle



Beispiele für Statistikdefinitionen

Beispiel 1:

In einer Statistik soll die Anzahl der Personen im Unternehmen dargestellt werden. Die Berechnung der Statistik soll täglich erfolgen. Die Definition der Statistik könnte folgendermaßen aussehen:

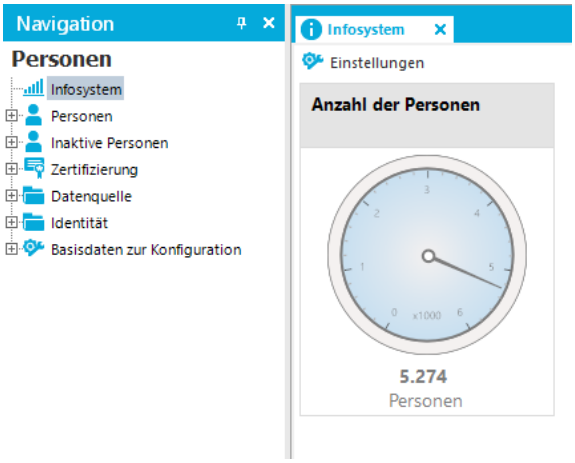
Statistik:	CountEmployees
Anzeigename:	Anzahl der Personen
Beschreibung:	Ermittelt täglich die Anzahl der Personen im Unternehmen.
Zeitplan der Berechnung:	Statistiken berechnen
Abfrage Messwerte:	`select 'Personen' as ElementName, count (*) as ElementValue from Person`

Um die Statistik im Manager in der Kategorie **Personen > Infosystem** anzuzeigen, wird ein Menüeintrag mit folgenden Werten erstellt:

Menüeintrag:	Person.InfoSystem.CountEmployees
Eintragsart:	Statistik
Reihenfolge:	1
Statistik:	Anzahl der Personen
Diagrammtyp:	Thermometer

Der Menüeintrag wird der Anwendung **Manager** und einer Anwendungsrolle zugewiesen und kann dann im Manager angezeigt werden.

Abbildung 24: Anzeige der Statistik im Manager



Beispiel 2:

In einer Statistik soll die Anzahl der externen Personen im Unternehmen dargestellt werden. Die Berechnung der Statistik soll wöchentlich erfolgen. Sind mehr als 20 % aller Personen des Unternehmens externe Personen, soll im Infosystem statt eines ordnungsgemäßen Zustandes ein tolerierten Zustand angezeigt werden, ab 80% externer Personen soll ein unerlaubter Zustand angezeigt werden.

Statistik:	CountExternalEmployees
Anzeigename:	Anzahl der externen Personen
Beschreibung:	Ermittelt wöchentlich die Anzahl der externen Personen im Unternehmen.
Zeitplan der Berechnung:	Statistiken wöchentlich berechnen
Abfrage Messwerte:	`Select 'Personen' as ElementName, Count (*) as ElementValue` `from Person where IsExternal = 1`
Abfrage Basismesswerte:	`Select 'Personen' as ElementName, Count (*) as ElementValue` `from Person`
Schwellwert grün:	0,2
Schwellwert rot:	0,8

Um die Statistik im Manager in der Kategorie **Personen > Infosystem** anzuzeigen, wird ein Menüeintrag mit folgenden Werten erstellt:

Menüeintrag:	Person.InfoSystem.CountExternalEmployees
Eintragsart:	Statistik
Reihenfolge:	2
Statistik:	Anzahl der externen Personen
Diagrammtyp:	Ampel

Der Menüeintrag wird der Anwendung **Manager** und einer Anwendungsrolle zugewiesen und kann dann im Manager angezeigt werden.

Beispiel 3:

In einer Statistik soll die Anzahl der Personen dargestellt werden, für welche der angemeldete Benutzer direkt als Manager eingetragen ist. Die Einschränkung der für den angemeldeten Benutzer anzuzeigenden Werte erfolgt über die Bedingung.

Statistik:	CountEmployeesPersonHead
Anzeigename:	Anzahl der direkt verantworteten Personen
Beschreibung:	Ermittelt täglich die Anzahl der Personen, für die ein Manager direkt verantwortlich ist.
Zeitplan der Berechnung:	Statistiken berechnen

Abfrage Messwerte:	`select XObjectKey as ElementObjectKey, 'Personen' as ElementName, Count (*) as ElementValue`
	`from Person where IsExternal = 1`
	`Group by XObjectKey`

Bedingung:	`ElementObjectKey in` `(select XObjectKey from Person where uid_PersonHead = '%uid%')`

Um die Statistik im Infosystem des Web Portals anzuzeigen, konfigurieren Sie das Webprojekt im Web Designer.

Beispiel 4:

In einer Statistik soll die Anzahl der internen und externen Personen angezeigt werden, für welche der angemeldete Benutzer als Abteilungsleiter verantwortlich ist. Da ein Abteilungsleiter für mehrere Abteilungen verantwortlich sein kann, werden hier zur Ermittlung eines eindeutigen Ergebnisses zur Anzeige die Messwerte der einzelnen Abteilungen addiert.

Statistik:	PersonCountInternalExternal_By_Department
Anzeigename:	Anzahl der internen und externen Personen
Beschreibung:	Ermittelt täglich die Anzahl der internen und externen Personen pro Abteilung.
Zeitplan der Berechnung:	Statistiken berechnen

Abfrage Messwerte:	`select d.XObjectKey as ElementObjectKey, 'Internal' as ElementName, count(p.uid_person) as ElementValue`
	`from Department d Left Outer Join Person p on p.UID_Department = d.UID_Department and p.IsExternal = 0`
	`Group By d.XObjectKey`
	`UNION ALL`
	`select d.XObjectKey as ElementObjectKey, 'External' as ElementName, count(p.uid_person) as ElementValue`
	`from Department d Left Outer Join Person p on p.UID_Department = d.UID_Department and p.IsExternal = 1`
	`Group By d.XObjectKey`

Bedingung:	`ElementObjectKey in` `(select d.XObjectKey from Department d join helperheadorg hpo on d.UID_Department = hpo.UID_Org where hpo.UID_PersonHead = '%useruid%')`
Aggregatfunktion	SUM

Um die Statistik im Infosystem des Web Portals anzuzeigen, konfigurieren Sie das Webprojekt im Web Designer.

Beispiel 5:

In einer Statistik sollen täglich die 10 aktiven Personen mit dem höchsten Risikoindex ermittelt werden. Die Darstellung soll sortiert nach der Messwertgröße erfolgen.

Statistik:	Top10ActivePersonByRiskIndex
Anzeigename:	Top 10 aktive Personen nach Risikoindex
Beschreibung:	Ermittelt täglich die 10 aktiven Personen mit den höchsten Risiko-indizes.
Zeitplan der Berechnung:	Statistiken berechnen
Abfrage Messwerte:	`select top 10 p.InternalName as ElementName,`

```
Round(100 * IsNull(p.RiskIndexCalculated, 0), 0) as
ElementValue,
p.XObjectKey as ElementObjectKey,
ROW_NUMBER() over (order by IsNull(p.RiskIndexCalculated, 0)
desc, p.InternalName) as ElementOrder
from Person p
where p.IsInActive = 0
order by ElementOrder
```

Um die Statistik im Infosystem des Web Portals anzuzeigen, konfigurieren Sie das Webprojekt im Web Designer.

Erweitern des Launchpad

Das Launchpad ist das zentrale Werkzeug zum Starten der Administrationswerkzeuge und Konfigurationswerkzeuge des One Identity Manager. Mit dem Launchpad können Sie die vorhandene One Identity Manager Installation prüfen und die Werkzeuge des One Identity Manager zur Ausführung einzelner Aufgaben starten.

Das Launchpad ist kundenspezifisch erweiterbar. Sie können im Designer eigene Menüeinträge und Aktionen für das Launchpad definieren.

Über die Hierarchie von Menüeinträgen und die verschiedenen Eintragsarten für Menüeinträge steuern Sie, an welcher Stelle und wie die Menüeinträge im Launchpad angezeigt werden. Ausführliche Informationen über den Aufbau einer Menüstruktur und die einzelnen Eintragsarten mit ihren Eigenschaften finden Sie unter Menüführung der Benutzeroberfläche auf Seite 124

One Identity Manager liefert eine Reihen von Launchpad Aktionen, die Sie zum Starten von Anwendungen über das Launchpad verwenden können. Bei Bedarf können Sie auch eigene Anwendungen über Launchpad Aktionen starten.

Detaillierte Informationen zum Thema

- Empfehlungen für die Erweiterung des Launchpad auf Seite 209
- Aktionen für das Launchpad auf Seite 210
- Menüeinträge und Aktionen für das Launchpad erstellen auf Seite 211

Empfehlungen für die Erweiterung des Launchpad

- Um eine neue Kategorie im linken Navigationsbereich des Launchpad zu erzeugen, verwenden Sie Menüeinträge mit der Eintragsart **Menükategorie**. Die Einträge werden mit ihrem Anzeigetext dargestellt.
- Um Aufgaben im Hauptbereich des Launchpad zusammenzufassen, verwenden Sie Menüeinträge mit der Eintragsart **Aufgabenkategorie**. Die Einträge werden mit ihrem Anzeigetext dargestellt.
- Für die einzelnen Aufgaben des Launchpad verwenden Sie Menüeinträge mit den Eintragsarten **Aufgabe**, **Fixer Menüeintrag** oder **Datenabhängiger Menüeintrag**. Die Einträge werden mit ihrem Anzeigetext und ihrer Beschreibung dargestellt.
- Legen Sie eine Reihenfolge für die Anzeige der Menüeinträge fest.
- Für eine Statusanzeige der Aufgaben verwenden Sie am Menüeintrag eine Overlay-Symbol Definition als VB.Net-Ausdruck. Verwenden Sie die `NavigationNodeState`-Klasse.

  Syntax:

  ```
  public NavigationNodeState(string state, string imageUidOrName, string description)
  ```

  ```
  public NavigationNodeState(string state, string imageUidOrName, string description, bool enabled, bool visible, int count)
  ```

Tabelle 57: Parameter des Skriptes NavigationNodeState

Parameter	Beschreibung
State	Status, der zurückgegeben wird, beispielsweise **Info**, **Ok**, **Error**, **Warning**.
ImageUidOrName	UID oder Name des anzuzeigenden Symbols.
description	Text, der als Tooltip angezeigt wird.
enabled	Gibt an, ob die Start-Schaltfläche für die Aktion aktiviert/deaktiviert werden soll.
visible	Gibt an, ob die Aufgabe angezeigt werden soll.
count	Anzahl der betroffenen Objekte.

Aufrufbeispiel:

```
Value = New NavigationNodeState("Ok", "QBM-33228392E9863141A9306B38ADF3D502", #LD("Project is completed.")#)
```

```
Value = New NavigationNodeState("Error", "QBM-a486f0eabf674392bbbdf8572453258c", #LD("Project is not completed.")#)
```

- Über die Bedingung können Sie einschränken, ob eine Aufgabe nur für eine direkte Datenbankverbindung oder für eine Verbindung über Anwendungsserver angeboten wird. Verwenden Sie dazu die Variable `SessionType`.

  Beispiele für Bedingungen:

  Nur direkte Datenbankverbindung: `'%SessionType%' = 'Direct'`

  Nur Verbindung über Anwendungsserver: `'%SessionType%' = 'AppServer'`

  Ist keine Bedingung angegeben, wird die Aufgabe immer angeboten.
- Soll über eine Aufgabe eine Aktion ausgeführt werden, verknüpfen Sie eine Launchpad Aktion mit dem Menüeintrag.

  Damit wird die Schaltfläche **Starten** für die Aufgabe angezeigt. Die Beschreibung einer Launchpad Aktion wird als Tooltip der Schaltfläche verwendet.
- Sollen Aktionen im Launchpad nicht für alle Benutzer verfügbar sein, steuern Sie die Berechtigungen über die Zuweisung von Launchpad Aktionen an Programmfunktionen (Tabelle `QBMLaunchActionHasFeature`). Es werden nur die Aufgaben im Launchpad angezeigt, deren Aktionen ein Benutzer über seine Programmfunktion ausführen darf.

  Ausführliche Informationen zum Steuern von Berechtigungen zum Ausführen von Launchpad Aktionen über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Verwandte Themen

- Menüeinträge und Aktionen für das Launchpad erstellen auf Seite 211
- Menüführung der Benutzeroberfläche auf Seite 124
- Aktionen für das Launchpad auf Seite 210

Aktionen für das Launchpad

One Identity Manager liefert eine Reihen von Launchpad Aktionen, die Sie zum Starten von Anwendungen verwenden können. Bei Bedarf können Sie auch eigene Anwendungen über Launchpad Aktionen starten.

Zum Start einer Anwendung können Sie Aufrufparameter, Methoden und Methodenparameter übergeben, die der Anwendung bekannt sind. Dabei ist die Verwendung von Variablen zulässig. Unterstützt werden

- Umgebungsvariablen in der Syntax `%Variable%`
- Variablen aus der Menüführung in der Syntax `%Variable%`
- Spalten des übergebenen Objektes in der `$`-Notation.

Um die Launchpad Aktionen anzuzeigen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Launchpad Aktionen**.
2. Wählen Sie die Launchpad Aktion. Folgende Stammdaten werden für eine Launchpad Aktion abgebildet.

Tabelle 58: Eigenschaften einer Aktion

Eigenschaft	Beschreibung
Beschreibung	Nähere Beschreibung der Aktion. Die Beschreibung wird als Tooltip angezeigt.
Ausführbare Datei	Kompletter Name der auszuführenden Datei.
Aufrufparameter	Zusätzliche Aufrufparameter zum Starten der Anwendung.
Aktion	Bezeichnung der Aktion.
Administrativer Kontext	Gibt an, ob die Anwendung im administrativen Kontext gestartet werden muss. Die Anwendung erwartet bei aktivierter Benutzerkontensteuerung die Angabe einer administrativen Anmeldung.
Methode	Methode, die zusätzlich als Startparameter übergeben werden muss.
Methodenparameter	Zusätzliche Parameter für die Methode.
Editor	Spezieller Editor, um Aufgaben direkt im Launchpad auszuführen.

Menüeinträge und Aktionen für das Launchpad erstellen

Sie können im Designer eigene Menüeinträge und Aktionen für das Launchpad definieren.

Um das Launchpad zu erweitern

1. Erstellen Sie neue Menüeinträge für das Launchpad.
   a. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Menüführung > Launchpad**.
   b. Starten Sie den Oberflächeneditor über die Aufgabe **Menüführung der Anwendung 'Launchpad' bearbeiten**.
   c. Erstellen Sie die Menüeinträge.

   d. Weisen Sie die Menüeinträge der Anwendung **Launchpad** zu.
   e. Weisen Sie die Menüeinträge der Berechtigungsgruppe zu **QBM-LaunchPad**.
2. Weisen Sie Launchpad Aktionen an die Menüeinträge zu.
   a. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Launchpad Aktionen**.
   b. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogTree`.
   c. Wählen Sie die Launchpad Aktion und weisen Sie der Aktion über den Tabreiter **Menüeinträge** den Menüeintrag zu.
3. (Optional) Weisen Sie an die Launchpad Aktion eine Programmfunktion zu.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Empfehlungen für die Erweiterung des Launchpad auf Seite 209
- Aktionen für das Launchpad auf Seite 210

Methodendefinitionen für die Benutzeroberfläche

Methoden setzen Sie ein um in den One Identity Manager-Werkzeugen bestimmte Aktionen mit Objekten auszuführen. Methodendefinitionen werden für Objektdefinitionen erstellt. Abhängig vom ausgewählten Objekt können unterschiedliche Methoden in der Benutzeroberfläche angeboten werden. Durch die zusätzliche Zuweisung der Methodendefinitionen an Berechtigungsgruppen, werden diese Methoden den Systembenutzern abhängig von ihren Mitgliedschaften in Berechtigungsgruppen zur Verfügung gestellt.

Wird eine Methodendefinition mit einer Programmfunktion (Tabelle `QBMMethodHasFeature`) versehen, so kann ein Benutzer diese Methode nur noch ausführen, wenn er auch die nötige Programmfunktion über seine Berechtigungsgruppen besitzt. Besitzt der Benutzer die Programmfunktion nicht, so wird beim Ausführungsversuch eine Fehlermeldung geworfen.

Ausführliche Informationen zum Steuern von Berechtigungen zum Ausführen von Methoden über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar. Bei Bedarf können Sie einzelne vordefinierte Methoden deaktivieren, um deren Anzeige in der Benutzeroberfläche zu verhindern. Die Deaktivierung bleibt auch bei Schemainstallationen erhalten.

HINWEIS: Die Methoden werden in der Aufgabenansicht der One Identity Manager-Werkzeuge in alphabetischer Sortierung angezeigt.

Neben objektabhängigen Methodendefinitionen werden Formularmethoden und Customizermethoden bereitgestellt, die nicht bearbeitbar sind.

Detaillierte Informationen zum Thema

- Methodendefinitionen deaktivieren auf Seite 214
- Methodendefinitionen erstellen und bearbeiten auf Seite 213
- Skript zum bedingten Anzeigen von Methoden auf Seite 214
- Eigenschaften von Methodendefinitionen auf Seite 215

Methodendefinitionen erstellen und bearbeiten

Um eine Methode zu erstellen oder zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Methodendefinitionen**.
2. Wählen Sie die Methode.

   - ODER -

   Um eine neue Methode zu erstellen, wählen Sie den Menüeintrag **Objekt > Neu**.
3. Wählen Sie die Aufgabe **Stammdaten bearbeiten** und erfassen Sie die Stammdaten der Methode.
4. Weisen Sie die Methodendefinition einer Berechtigungsgruppe zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogGroupHasMethod`.
   b. Wählen Sie in der Bearbeitungsansicht die Ansicht **Berechtigungsgruppe** und wählen Sie die Berechtigungsgruppen.
5. Weisen Sie die Methodendefinition den Objektdefinition zu, für welche die Methode in der Benutzeroberfläche angeboten wird.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogObjectHasMethod`.
   b. Wählen Sie in der Bearbeitungsansicht die Ansicht **Objekt** und wählen Sie die Objektdefinitionen.
6. (Optional) Weisen Sie an die Methodendefinition eine Programmfunktion zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `QBMMethodHasFeature`.

b. Wählen Sie in der Bearbeitungsansicht die Ansicht **Programmfunktion** und wählen Sie die Programmfunktionen.

7. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Methodendefinitionen deaktivieren
- Skript zum bedingten Anzeigen von Methoden auf Seite 214
- Eigenschaften von Methodendefinitionen

Methodendefinitionen deaktivieren

Bei Bedarf können Sie einzelne vordefinierte Methoden deaktivieren, um deren Anzeige in der Benutzeroberfläche zu verhindern. Die Deaktivierung bleibt auch bei Schemainstallationen erhalten.

Um eine Methode zu deaktivieren

1. Wählen Sie im Designer in der Kategorie **Benutzeroberfläche > Methodendefinitionen** die Methode.
2. Wählen Sie die Aufgabe **Stammdaten bearbeiten**.
3. Setzen Sie die Option **Deaktiviert**.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Methodendefinitionen erstellen und bearbeiten auf Seite 213
- Skript zum bedingten Anzeigen von Methoden auf Seite 214
- Eigenschaften von Methodendefinitionen auf Seite 215

Skript zum bedingten Anzeigen von Methoden

Grundsätzlich werden die Berechtigungen eines Benutzers zum Anzeigen und Bearbeiten von Methoden über die Berechtigungen der Berechtigungsgruppen gesteuert.

Zusätzlich können Sie in kundendefinierten Methodendefinitionen ein Skript zum bedingten Anzeigen einer Methode verwenden. So kann beispielsweise gesteuert werden, dass eine Methode im Manager nur angezeigt wird, wenn eine bestimmte Bedingung erfüllt ist.

HINWEIS: Das Skript verändert nicht die Berechtigungen eines Benutzers, sondern lediglich das Verhalten beim Laden eines Objektes in den One Identity Manager-Werkzeugen. Wenn Sie die Sichtbarkeit einer Methode grundsätzlich einschränken wollen, passen Sie die Berechtigungen für die Berechtigungsgruppen an. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Um ein Skript zum bedingten Anzeigen für eine Methode festzulegen

1. Wählen Sie im Designer in der Kategorie **Benutzeroberfläche > Methodendefinitionen** die Methode.
2. Wählen Sie die Aufgabe **Stammdaten bearbeiten**.
3. Erfassen Sie ein **Skript zur Sichtbarkeit** in VB.Net Syntax. Ist der Rückgabewert **False**, wird die Methode in den One Identity Manager-Werkzeugen nicht angezeigt.
4. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Verwandte Themen

- Methodendefinitionen erstellen und bearbeiten auf Seite 213
- Methodendefinitionen deaktivieren auf Seite 214
- Eigenschaften von Methodendefinitionen auf Seite 215

Eigenschaften von Methodendefinitionen

Tabelle 59: Eigenschaften einer Methode

Eigenschaft	Bedeutung
Methodenname	Namen für die Methode.
Anzeigetext	Anzeigename, mit dem die Methode in der Aufgabenansicht der Administrationswerkzeuge angezeigt wird. Den Anzeigenamen können Sie mehrsprachig angeben. HINWEIS: Die Methoden werden in der Aufgabenansicht der One Identity Manager-Werkzeuge in alphabetischer Sortierung angezeigt.
Beschreibung	Beschreibung der Methode. Die Beschreibung wird in der Aufgabenansicht der Oberflächen als Tooltip angezeigt.
Freigeschaltet für	Diese Eigenschaft legt fest, für welchen Einsatzzweck eine Methode geeignet ist. Zulässige Werte sind: • **Fat Client**: Die Methode kann in der grafischen

Eigenschaft	Bedeutung
	Benutzeroberfläche angezeigt werden. • **Web Client**: Die Methode kann in Webanwendungen angezeigt werden. • **Fat Client + Web Client**: Die Methode kann in der grafischen Benutzeroberfläche und in Webanwendungen angezeigt werden.
Methodenverhalten	Kennzeichen zum Verhalten der Methode. Folgende Angaben sind zulässig: • **Keine Angaben**: Standard. Die Methode wird bei Einzelobjektbearbeitung und Mehrfachbearbeitung von Objekten angeboten. Die Änderungen werden für jedes Objekt einzeln ausgeführt, auch wenn die Mehrfachbearbeitung verwendet wird. • **Methode speichert**: Die Methode speichert Daten. Es wird eine entsprechende Hinweismeldung angezeigt. • **Nur für Einzelobjekte**: Die Methode ist nur für Einzelobjektbearbeitung zulässig. • **Methode speichert + Nur für Einzelobjekte**: Die Methode speichert Daten. Es wird eine entsprechende Hinweismeldung angezeigt. Die Methode ist nur für Einzelobjektbearbeitung zulässig. • **Auf Multi-Objekten ausführen**: Die Methode wird bei Mehrfachbearbeitung von Objekten angeboten. Die Änderungen werden für alle Objekte gemeinsam über ein Multi-Objekt ausgeführt. • **Methode speichert + Auf Multi-Objekten ausführen**: Die Methode speichert Daten. Es wird eine entsprechende Hinweismeldung angezeigt. Die Methode wird bei Mehrfachbearbeitung von Objekten angeboten. Änderungen werden für alle Objekte gemeinsam über ein Multi-Objekt ausgeführt.
Symbol	Symbol, mit welchem die Methode in der Benutzeroberfläche angezeigt wird.
Skript	Methodenskript. Für das auszuführende Methodenskript können Sie Funktionsaufrufe oder Kommandoeingaben als VB.Net-Ausdrücke verwenden. Mit `base.` wird immer das aktuell geladene Objekt angesprochen. HINWEIS: Die Anpassung eines Methodenskriptes erfordert die Kompilierung der Datenbank.

Eigenschaft	Bedeutung
Skript zur Sichtbarkeit	Skript zum bedingten Anzeigen der Methode in den One Identity Manager-Werkzeugen. Weitere Informationen finden Sie unter Skript zum bedingten Anzeigen von Methoden auf Seite 214.
Deaktiviert	Gibt an, ob eine Methode in der Benutzeroberfläche angezeigt wird. Deaktivierte Methoden werden nicht in der Benutzeroberfläche angezeigt. Die vordefinierten Systembenutzer sind von dieser Einschränkung nicht betroffen. Diese Änderung ist auch bei vordefinierten Methoden de Standard-benutzeroberfläche zulässig und bleibt bei Schemainstallationen bestehen.
Bearbeitungsstatus	Der Bearbeitungsstatus wird bei der Erstellung von Kunden-konfigurationspaketen genutzt.
Objekt	Zuweisung der Objektdefinitionen (Tabelle `DialogObjectHasMethod`), für welche die Methode in der Benutzeroberfläche angeboten wird.
Berechtigungsgruppe	Zuweisung der Berechtigungsgruppen (Tabelle `DialogGroupHasMethod`), deren Benutzer diese Methode benutzen können.
Programmfunktion	Programmfunktion, die mit der Methodendefinition verbunden ist. Wird eine Methodendefinition mit einer Programmfunktion (Tabelle `QBMMethodHasFeature`) versehen, so kann ein Benutzer diese Methode nur noch ausführen, wenn er auch die nötige Programmfunktion über seine Berechtigungsgruppen erhält. Ausführliche Informationen zum Steuern von Berechtigungen zum Ausführen von Methoden über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Verwandte Themen

- Methodendefinitionen erstellen und bearbeiten auf Seite 213
- Methodendefinitionen für die Benutzeroberfläche auf Seite 212
- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Verwendung der #LD-Notation auf Seite 389
- Sprachabhängige Abbildung von Informationen auf Seite 233

Anwendungen für die Gestaltung der Benutzeroberfläche

In der Standardauslieferung des One Identity Manager werden die Anwendungen und die vordefinierte Menüführung für die One Identity Manager-Werkzeuge Manager, Designer und Launchpad mitgeliefert. Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar. Die Definition eigener Anwendungen ist in der Regel nicht erforderlich. Lediglich für kundenspezifische Webfrontends könnten eigene Anwendungen erforderlich werden.

Die verfügbaren Anwendungen werden im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Anwendungen** abgebildet.

Tabelle 60: Vordefinierte Anwendungen

Produkt	Bedeutung
Default	Standard für Frontends ohne spezielle Anwendung, beispielsweise Job Queue Info oder Report Editor. Wird benötigt für die Bestimmung der Authentifizierungsmodule.
Designer	Anwendung für den Designer.
Manager	Anwendung für den Manager.
Launchpad	Anwendung für das Launchpad.
WebDesigner	Anwendung zu Installation des Web Portals.
WebDesignerEditor	Anwendung für den Web Designer, um das Web Portal zu konfigurieren und zu erweitern.
Application Server	Anwendung zur Installation des Anwendungsservers.
SOAP Service	Anwendung zur Installation des SOAP Web Service.
SPML Service	Anwendung zur Installation des SPML Webservice.
API Designer	Anwendung zu Konfiguration des Web Portals.
OperationsSupportWebPortal	Anwendung zur Installation des Web Portal für Betriebsunterstützung.
PasswordReset	Anwendung zur Installation des Kennwortrücksetzungsportals.

Eigenschaften von Anwendungen

Tabelle 61: Eigenschaften von Anwendungen

Eigenschaft	Bedeutung
Anwendung	Name der Anwendung.
Bemerkungen	Bemerkungen zur Anwendung.
Startmenüeintrag	Wird einem Systembenutzer der angegebene Startmenüeintrag in der Menüführung einer Anwendung zur Verfügung gestellt, so wird bei Start der Anwendung auf diesen Menüeintrag navigiert. Somit können Sie beispielsweise die Startseite für den Systembenutzer vorgeben. Diese Funktion wird hauptsächlich in Webfrontends Verwendung finden.
Konfigurationsdaten	Die Konfigurationsdaten werden zur Ermittlung eines Systembenutzers über die dynamischen Authentifizierungsmodule genutzt. Die Konfigurationsdaten können Sie auch für die mitgelieferten Standardanwendungen anpassen. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.
Min. Version	Minimale Anwendungsversion, die mit der eingesetzten Datenbankversion lauffähig ist. Die Angabe dient lediglich als Information, es erfolgt keine Überprüfung der Versionsnummer.
Engine basiert	Gibt an, ob der Anwendung Menüführung und Formulare zugewiesen werden können.
Bearbeitungsstatus	Der Bearbeitungsstatus wird bei der Erstellung von Kundenkonfigurationspaketen genutzt.
Authentifizierungsmodul	Authentifizierungsmodule, welche die Anwendung verwenden. Ausführliche Informationen zu den One Identity Manager Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*. ***Um die Authentifizierungsmodule für eine Anwendung anzuzeigen*** 1. Wählen Sie im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Anwendungen** die Anwendung. 2. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogProductHasAuthentifier`.

Eigenschaft	Bedeutung
	3. Wählen Sie den Tabreiter **Authentifizierungsmodul**.
Formular	Formulare, die in der Anwendung angezeigt werden. ***Um die Formulare für eine Anwendung anzuzeigen*** 1. Wählen Sie im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Anwendungen** die Anwendung. 2. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogProductHasSheet`. 3. Wählen Sie den Tabreiter **Formularzuweisung**.
Menü	Menüeinträge, die in der Anwendung angezeigt werden. ***Um die Menüeinträge für eine Anwendung anzuzeigen*** 1. Wählen Sie im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Anwendungen** die Anwendung. 2. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogTreeInDialogProduct`. 3. Wählen Sie den Tabreiter **Menüzuweisung**.
Systembenutzer	Systembenutzer, die die Anwendung nutzen. ***Um die Systembenutzer für eine Anwendung anzuzeigen*** 1. Wählen Sie im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Anwendungen** die Anwendung. 2. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogUserConfiguration`. 3. Wählen Sie den Tabreiter **Systembenutzer**.
Berechtigungsgruppen	Berechtigungsgruppen, deren Berechtigungen zusätzlich für diese Anwendung gültig sind. Ausführliche Informationen zu Berechtigungsgruppen im One Identity Manager finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*. ***Um die Berechtigungsgruppen für eine Anwendung anzuzeigen***

Eigenschaft	Bedeutung
	1. Wählen Sie im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Anwendungen** die Anwendung. 2. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogGroupInProductLimited`. 3. Wählen Sie den Tabreiter **Berechtigungsgruppe**.

Verwandte Themen

- Anwendungen für die Gestaltung der Benutzeroberfläche auf Seite 218

Symbole und Bilder für die Gestaltung der Benutzeroberfläche

Bei der Gestaltung der Benutzeroberfläche können Sie für verschiedene Bestandteile Symbole und Bilder für die Anzeige in den One Identity Manager-Werkzeugen hinterlegen. In der Standardauslieferung des One Identity Manager werden Symbole und Bilder mitgeliefert, die Sie bei der Bearbeitung der Benutzeroberfläche und bei der Erstellung von Berichten verwenden können. Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar.

Für die graphische Oberfläche werden die Symbole im PNG-Format mit den Abmessungen 16x16 Pixel und 24x24 Pixel benötigt.

Symbole werden für die folgenden Verwendungen benötigt.

Tabelle 62: Verwendung der Symbole

Zustand	Verwendung
Normal	Symbole, die einen aktiven Zustand anzeigen. Diese Symbole müssen in der One Identity Manager-Datenbank vorhanden sein.
Invertiert	Symbole, die einen aktiven Zustand anzeigen auf schwarzem Hintergrund. Diese Symbole können automatisch konvertiert werden.
Inaktiv	Symbole, die einen inaktiven Zustand anzeigen. Diese Symbole müssen in der One Identity Manager-Datenbank vorhanden sein.

Um Symbole einzufügen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Symbole**.
2. Wählen Sie den Menüeintrag **Objekt > Neu**.

3. Geben Sie den Symbolnamen an.
4. Laden Sie das Symbol über die Schaltfläche .
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

Um Bilder für Berichte einzufügen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Große Bilder**.
2. Wählen Sie den Menüeintrag **Objekt > Neu**.
3. Geben Sie den Namen des Bildes an.
4. Laden Sie das Bild über die Schaltfläche .
5. Speichern Sie das Bild über die Schaltfläche .

   Es wird das Bild mit einigen wichtigen Bildeigenschaften angezeigt.
6. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

HINWEIS: Wenn Sie Symbole oder Bilder bearbeiten, die in einer Webanwendung verwendet werden, müssen Sie anschließend die API sowie die Webanwendung mit dem Database Compiler kompilieren.

Verwenden vordefinierter Datenbankabfragen

Mit dem Einsatz eines Anwendungsservers können aus Sicherheitsgründen von den Frontends und Webanwendungen keine direkten Datenbankanfragen ausgeführt werden. Datenbankabfragen, die beispielsweise auf Formularen benötigt werden, müssen im One Identity Manager als vordefinierte Datenbankabfragen formuliert werden. Die Ausführung der Datenbankabfragen erfolgt immer mit den Berechtigungen des angemeldeten Benutzers. Die vordefinierten Datenbankabfragen müssen einer Berechtigungsgruppe zugewiesen sein.

Im Web Designer unterstützt Sie ein Assistent dabei Datenbankabfragen für das Web Portal zu erstellen und mit mindestens einer Berechtigungsgruppe zu verknüpfen. Weiterer vordefinierte Datenbankabfragen können Sie im Designer erfassen.

Um vordefinierte Datenbankabfragen zu erstellen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Erweitert > Vordefiniertes SQL**.
2. Wählen Sie den Menüeintrag **Objekt > Neu**.
3. Bearbeiten Sie die Stammdaten.

Tabelle 63: Eigenschaften von vordefinierten Datenbankabfragen

Eigenschaft	Beschreibung
Bearbeitungsstatus	Bearbeitungsstatus des Objektes. Der Bearbeitungsstatus wird bei der Erstellung von Kundenkonfigurationspaketen genutzt.
Beschreibung	Freitextfeld für zusätzliche Erläuterungen.
Bezeichner des SQL Codes	Eindeutiger Bezeichner, der verwendet wird, um die Abfrage zu identifizieren
SQL-Typ	Typ der SQL-Abfrage. Zulässige Werte sind: • **Statement**: Es handelt sich um eine komplette SQL-Abfrage. • **WHERE clause**: Es handelt sich um den Where-Klausel Anteil einer SQL-Abfrage.
Code	Vollständige Datenbankabfrage in SQL-Syntax. In der Abfrage können Sie SQL Parameter verwenden.

4. Weisen Sie Berechtigungsgruppen zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `QBMGroupHasLimitedSQL`.
   b. Wählen Sie in der Bearbeitungsansicht die Ansicht **Berechtigungsgruppe** und wählen Sie die Berechtigungsgruppen.
5. Wählen Sie den Menüeintrag **Datenbank > Übertragung in Datenbank** und klicken Sie **Speichern**.

7

Lokalisierung im One Identity Manager

Der One Identity Manager benötigt an verschiedenen Stellen Länderinformationen, so wird beispielsweise bei der Erstellung von E-Mail Benachrichtigungen oder bei der Ermittlung von IT Shop Abläufen auf die Zuordnung der Personen zu Ländern und Bundesländern zurückgegriffen. Neben den Ländern und Bundesländern werden Sprachkultur, Zeitzonen, Feiertage und Arbeitszeiten abgebildet. Die Basisdaten werden durch die Schemainstallation in die Datenbank eingespielt.

Der One Identity Manager unterstützt die sprachabhängige Abbildung von Informationen. So können Sie Anzeigetexte für die Benutzeroberflächen der One Identity Manager-Werkzeuge für unterschiedliche Sprachen anpassen. Des Weiteren können Sie Ausgabetexte der Prozessinformationen und der Skriptverarbeitung sowie Benachrichtigungen aus der Prozessverarbeitung mehrsprachig gestalten.

Die Standardinstallation des One Identity Manager wird in den Sprachen **English - United States [en-US]** und **German - Germany [de-DE]** ausgeliefert. Bei Bedarf können Sie weitere Sprachen zur Gestaltung der Benutzeroberfläche und der Ausgabetexte einpflegen. Für diesen Fall müssen Sie vor Einsatz des One Identity Manager die verwendeten Texte in die neue Sprache übersetzen. Für die Übersetzung steht Ihnen im Designer einen Wörterbucheditor zur Verfügung. In den One Identity Manager-Werkzeugen wird die Eingabe mehrsprachiger Inhalte durch ein spezielles Steuerelement unterstützt.

Zur Übersetzung des Web Portal werden mit dem One Identity Manager Language Pack weitere Sprachen bereitgestellt.

> HINWEIS: Das One Identity Manager Language Pack finden Sie im Support Portal unter https://support.oneidentity.com/.

Detaillierte Informationen zum Thema

- Spracheinstellungen für die Anzeige und Pflege der Daten auf Seite 40
- Arbeiten in verschiedenen Zeitzonen auf Seite 225
- Ermitteln von Arbeitszeiten auf Seite 225
- Bearbeiten von Länderinformationen auf Seite 226
- Sprachabhängige Abbildung von Informationen auf Seite 233
- Übersetzungen aus dem Language Pack importieren auf Seite 243

Arbeiten in verschiedenen Zeitzonen

Zeitstempel, wie beispielsweise Einfügedatum oder Änderungsdatum, werden im One Identity Manager mit der jeweils aktuellen UTC Zeit vermerkt. Die Objektschicht wandelt diese Zeitinformationen beim Laden eines Objektes in die jeweils gültige Zeitzoneninformationen um. Der Anwender sieht also alle Werte in seiner lokalen Zeit. Beim Speichern eines Objektes werden die aktuellen Zeitzoneninformationen in UTC Zeitinformationen umgewandelt.

Länder und Zeitzonen sind im One Identity Manager Schema miteinander verknüpft. Damit können die Zeitzonen bei Einsatz von Webfrontends wie beispielsweise dem Web Portal einfach ermittelt werden.

Verwandte Themen

- Länder und Bundesländer aktivieren auf Seite 226
- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229

Ermitteln von Arbeitszeiten

Für verschiedene Funktionen des One Identity Manager werden die Arbeitszeiten berechnet, beispielsweise bei der Ermittlung von Bearbeitungszeiten im IT Shop oder für die Ermittlung von Reaktionszeiten und Lösungszeiten für Calls im Helpdeskmodul. Die Ermittlung der Arbeitszeiten erfolgt unter Berücksichtigung der Wochenenden und der Feiertage.

Um die Arbeitszeiten zu ermitteln

- Stellen Sie sicher, dass in den Stammdaten der Personen ein Bundesland und/oder ein Bundesstaat eingetragen ist.
- Die Feiertage werden im One Identity Manager nach Bundesländern erfasst. Für die Bundesländer können Sie die einzelnen Feiertage hinterlegen.
- Um Feiertage bei der Berechnung der Arbeitszeiten nicht zu berücksichtigen, aktivieren Sie im Designer den Konfigurationsparameter **QBM | WorkingHours | IgnoreHoliday**.
- Um Wochenenden bei der Berechnung der Arbeitszeiten nicht zu berücksichtigen, aktivieren Sie im Designer den Konfigurationsparameter **QBM | WorkingHours | IgnoreWeekend**.

Verwandte Themen

- Standardarbeitszeiten für Länder und Bundesländer festlegen auf Seite 227
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227

- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229

Bearbeiten von Länderinformationen

Der One Identity Manager benötigt an verschiedenen Stellen Länderinformationen, so wird beispielsweise bei der Erstellung von E-Mail Benachrichtigungen oder bei der Ermittlung von IT Shop Abläufen auf die Zuordnung der Personen zu Ländern und Bundesländern zurückgegriffen.

Neben den Ländern und Bundesländern werden Sprachkultur, Zeitzonen, Feiertage und Arbeitszeiten abgebildet. Die Basisdaten werden durch die Schemainstallation in die Datenbank eingespielt.

Detaillierte Informationen zum Thema

- Länder und Bundesländer aktivieren auf Seite 226
- Standardarbeitszeiten für Länder und Bundesländer festlegen auf Seite 227
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227
- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229
- Eigenschaften von Ländern auf Seite 230
- Eigenschaften von Bundesländern auf Seite 231
- Eigenschaften von Feiertagen auf Seite 232

Länder und Bundesländer aktivieren

Um ein Land zu aktivieren

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land > Deaktiviert**.
2. Wählen Sie das Land.
3. Setzen Sie die Option **Aktiviert**.

Um ein Bundesland zu aktivieren

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land > <Länderbezeichnung> > Bundesländer**.
2. Wählen Sie das Bundesland.
3. Setzen Sie die Option **Aktiviert**.

Verwandte Themen

- Arbeiten in verschiedenen Zeitzonen auf Seite 225
- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229

Standardarbeitszeiten für Länder und Bundesländer festlegen

Legen Sie gültigen Arbeitszeiten Länder und Bundesländer fest. Die Arbeitszeiten werden bei der Berechnung von Zeitabläufen, beispielsweise innerhalb des IT Shops, berücksichtigt.

Um die Arbeitszeiten eines Landes zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land**.
2. Wählen Sie das Land.
3. Legen Sie unter **Arbeitsstunden (Standard)** die Standardarbeitszeiten fest.

Um die Standardarbeitszeiten für ein Bundesland zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land > <Länderbezeichnung> > Bundesländer**.
2. Wählen Sie das Bundesland.
3. Legen Sie unter **Arbeitsstunden (Standard)** die Standardarbeitszeiten fest.

Verwandte Themen

- Ermitteln von Arbeitszeiten auf Seite 225

Feiertage für Länder und Bundesländer anzeigen

Die Feiertage werden durch die Schemainstallation in die Datenbank eingespielt und müssen in der Regel nicht angepasst werden.

Um die Feiertage eines Landes zu anzuzeigen

- Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land > <Länderbezeichnung> > Feiertage**.

Um die Feiertage eines Bundeslandes anzuzeigen

- Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land > <Länderbezeichnung> > Bundesländer > <Bundesland> > Feiertage**.

Verwandte Themen

- Eigenschaften von Feiertagen auf Seite 232
- Ermitteln von Arbeitszeiten auf Seite 225

Länder bearbeiten

Die Länder werden durch die Schemainstallation in die Datenbank eingespielt und müssen in der Regel nicht angepasst werden.

HINWEIS: Für aktivierte Länder wird im Designer in der Kategorie **Basisdaten > Lokalisierung > Land** ein Eintrag mit der Länderbezeichnung eingeblendet. Nicht aktivierte Länder werden in der Kategorie **Basisdaten > Lokalisierung > Land > Deaktiviert** angezeigt.

Um ein Land zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land**.
2. Wählen Sie das Land.
3. Bearbeiten Sie die Stammdaten.
4. (Optional) Weisen Sie Sprachkulturen an das Land zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogCountryHasCulture`.
   b. Wählen Sie auf dem Tabreiter **Sprachkulturen** die Sprachkulturen.
5. (Optional) Weisen Sie Zeitzonen an das Land zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogCountryHasTimeZone`.
   b. Wählen Sie auf dem Tabreiter **Zeitzonen** die Zeitzonen.
6. (Optional) Weisen Sie Feiertage an das Land zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogCountryHoliday`.
   b. Wählen Sie auf dem Tabreiter **Feiertage** die Feiertage.
7. (Optional) Weisen Sie Bundesländer an das Land zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogState`.
   b. Wählen Sie auf dem Tabreiter **Bundesländer** die Bundesländer.

Verwandte Themen

- Arbeiten in verschiedenen Zeitzonen auf Seite 225
- Länder und Bundesländer aktivieren auf Seite 226
- Standardarbeitszeiten für Länder und Bundesländer festlegen auf Seite 227
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227
- Bundesländer bearbeiten auf Seite 229
- Eigenschaften von Ländern auf Seite 230

Bundesländer bearbeiten

Die Bundesländer werden durch die Schemainstallation in die Datenbank eingespielt und müssen in der Regel nicht angepasst werden.

Um ein Bundesland zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Land > <Länderbezeichnung> > Bundesländer**.
2. Wählen Sie das Bundesland.
3. Bearbeiten Sie die Stammdaten.
4. (Optional) Weisen Sie Sprachkulturen an das Bundesland zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogStateHasCulture`.
   b. Wählen Sie auf dem Tabreiter **Sprachkulturen** die Sprachkulturen.
1. (Optional) Weisen Sie Zeitzonen an das Bundesland zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogStateHasTimeZone`.
   b. Wählen Sie auf dem Tabreiter **Zeitzonen** die Zeitzonen.
2. (Optional) Weisen Sie Feiertage an das Bundesland zu.
   a. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `DialogStateHoliday`.
   b. Wählen Sie auf dem Tabreiter **Feiertage** die Feiertage.

Verwandte Themen

- Arbeiten in verschiedenen Zeitzonen auf Seite 225
- Länder und Bundesländer aktivieren auf Seite 226
- Standardarbeitszeiten für Länder und Bundesländer festlegen auf Seite 227
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227

- Länder bearbeiten auf Seite 228
- Eigenschaften von Bundesländern auf Seite 231

Eigenschaften von Ländern

Tabelle 64: Eigenschaften von Ländern

Eigenschaft	Beschreibung
Länderbezeichnung	Bezeichnung des Landes.
Beschreibung	Nähere Beschreibung zum Land.
Aktiviert	Ist die Option aktiviert, dann wird dieses Land in den Administrationswerkzeugen zur Auswahl angeboten. Damit kann die Auswahl von Zeitzonen und Sprachkulturen eingeschränkt werden.
Sommerzeit	Angabe , ob bei der Berechnung der Differenz zur UTC Zeit eine Sommerzeit zu berücksichtigen ist.
Arbeitsstunden (Standard)	Legen Sie die landesweit gültigen Arbeitszeiten fest. Die Arbeitszeiten werden bei der Berechnung von Zeitabläufen, beispielsweise innerhalb des IT Shops, berücksichtigt.
Länderbezeichnung (Landessprache)	Name des Landes in der nationalen Sprache mit nationalen Schriftzeichen.
Hauptstadt (Landessprache)	Name der Hauptstadt in der nationalen Sprache mit nationalen Schriftzeichen.
Länderkürzel	Internationale Telefonvorwahlnummer des Landes.
Kfz-Nationalitätskennzeichen	Nationalitätskennzeichen für Kraftfahrzeuge.
ISO-Kennung (2-stellig)	Zweibuchstabiges Länderkürzel eines Landes. Die Angaben sind entsprechend der ISO 3166 einzutragen, einem Standard für die Kodierung von geographischen Einheiten.
ISO-Kennung (3-stellig)	Dreibuchstabiges Länderkürzel eines Landes. Die Angaben sind entsprechend der ISO 3166 einzutragen, einem Standard für die Kodierung von geographischen Einheiten.
ISO-Kennung (numerisch)	Numerischen Code für das Länderkürzel ein. Die Angaben sind entsprechend der ISO 3166 einzutragen, einem Standard für die Kodierung von geographischen Einheiten.
Objektklasse	Objektklasse für die Abbildung der Länderinformationen in einem LDAP Schema.
Suchmaske	Suchmaske für die Abbildung der Länderinformationen in

Eigenschaft	Beschreibung
	einem LDAP Schema.
UTC Offset (Mittelwert)	Mittlere Zeitdifferenz eines Landes zur UTC Zeit. Der Wert wird anhand der Zeitzonen eines Landes durch den DBQueue Prozessor berechnet.
Sprachkultur	Sprachkultur des Landes. Die Sprachkultur entscheidet beispielsweise, in welcher Sprache E-Mail Benachrichtigungen an einen Benutzer zugestellt werden.
Zeitzone	Zeitzone des Landes. Die Angabe einer Zeitzone wird beispielsweise bei der Berechnung zeitabhängige Abläufe, beispielsweise innerhalb des IT Shops, berücksichtigt.
Feiertage	Landesweit gültige Feiertage.
Bundesländer	Bundesländer des Landes.

Verwandte Themen

- Arbeiten in verschiedenen Zeitzonen auf Seite 225
- Ermitteln von Arbeitszeiten auf Seite 225
- Länder und Bundesländer aktivieren auf Seite 226
- Standardarbeitszeiten für Länder und Bundesländer festlegen auf Seite 227
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227
- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229

Eigenschaften von Bundesländern

Tabelle 65: Eigenschaften von Bundesländern

Eigenschaft	Beschreibung
Bundesland	Namen des Bundeslandes.
Name des Bundeslandes (Landessprache)	Name des Bundeslandes in der nationalen Sprache mit nationalen Schriftzeichen.
Land	Land, zu dem das Bundesland gehört.
Aktiviert	Aktivieren Sie die Bundesländer, die in Ihrem System verwendet werden.
Sommerzeit	Gibt an, ob bei der Berechnung der Differenz zur UTC Zeit eine

Eigenschaft	Beschreibung
	Sommerzeit zu berücksichtigen ist.
Arbeitsstunden (Standard)	Legen Sie gültigen Arbeitszeiten für dieses Bundesland fest. Die Arbeitszeiten werden bei der Berechnung von Zeitabläufen, beispielsweise innerhalb des IT Shops, berücksichtigt.
Hauptstadt	Name der Hauptstadt des Bundeslandes.
Hauptstadt (Landessprache)	Name der Hauptstadt in der nationalen Sprache mit nationalen Schriftzeichen.
Kurzname	Code entsprechend ISO 3166-2 für das Bundesland, beispielsweise **CA** für Kalifornien oder **SN** für Sachsen.
UTC Offset (Mittelwert)	Mittlere Zeitdifferenz eines Landes zur UTC Zeit. Der Wert wird anhand der Zeitzonen eines Bundeslandes durch den DBQueue Prozessor berechnet.
Sprachkultur	Sprachkultur des Landes. Die Sprachkultur entscheidet beispielsweise, in welcher Sprache E-Mail Benachrichtigungen an einen Benutzer zugestellt werden.
Zeitzone	Zeitzone des Landes. Die Angabe einer Zeitzone wird beispielsweise bei der Berechnung zeitabhängige Abläufe, beispielsweise innerhalb des IT Shops, berücksichtigt.
Feiertage	Feiertage des Bundeslandes.

Verwandte Themen

- Arbeiten in verschiedenen Zeitzonen auf Seite 225
- Ermitteln von Arbeitszeiten auf Seite 225
- Länder und Bundesländer aktivieren auf Seite 226
- Standardarbeitszeiten für Länder und Bundesländer festlegen auf Seite 227
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227
- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229

Eigenschaften von Feiertagen

Tabelle 66: Eigenschaften von Feiertagen

Eigenschaft	Beschreibung
Datum (ISO Format)	Eingetragen ist das Datum des Feiertages im ISO Format, beispielsweise `yyyy-mm-dd` wobei:

Eigenschaft	Beschreibung
	yyyy - Jahr, vierstellig mm - Monat, zweistellig dd - Tag, zweistellig
Name des Feiertages	Bezeichnung des Feiertages.
Name des Feiertages (Landesprache)	Name des Feiertages in der nationalen Sprache mit nationalen Schriftzeichen.
Land /Bundesland	Name des Landes/Bundeslandes für den Feiertag.
Bearbeitungsstatus	Der Bearbeitungsstatus wird bei der Erstellung von Kunden-konfigurationspaketen genutzt.
Deaktiviert	Gibt an, ob der Feiertag deaktiviert ist.

Verwandte Themen

- Ermitteln von Arbeitszeiten auf Seite 225
- Feiertage für Länder und Bundesländer anzeigen auf Seite 227
- Länder bearbeiten auf Seite 228
- Bundesländer bearbeiten auf Seite 229

Sprachabhängige Abbildung von Informationen

Der One Identity Manager unterstützt die sprachabhängige Abbildung von Informationen. So können Sie Anzeigetexte für die Benutzeroberflächen der One Identity Manager-Werkzeuge für unterschiedliche Sprachen anpassen. Des Weiteren können Sie Ausgabetexte der Prozessinformationen und der Skriptverarbeitung sowie Benachrichtigungen aus der Prozessverarbeitung mehrsprachig gestalten.

Detaillierte Informationen zum Thema

- Grundlagen zur Verwendung sprachabhängiger Informationen auf Seite 234
- Spalten zur Übersetzung kennzeichnen auf Seite 235
- Verwenden des Textspeichers zur Übersetzung auf Seite 237
- Übersetzungen im Wörterbucheditor anzeigen auf Seite 238
- Verwendung einer Übersetzung anzeigen auf Seite 239
- Übersetzungen einer einzelnen Tabelle bearbeiten auf Seite 240

- Alle Übersetzungen bearbeiten auf Seite 241
- Schlüsseltext einer Übersetzung ändern auf Seite 242
- Übersetzungen aus dem Language Pack importieren auf Seite 243

Verwandte Themen

- Spracheinstellungen für die Anzeige und Pflege der Daten auf Seite 40

Grundlagen zur Verwendung sprachabhängiger Informationen

Um die sprachabhängige Abbildung von Informationen innerhalb des One Identity Manager zu verwenden, müssen die folgenden Voraussetzungen erfüllt werden:

- Die Sprache ist in der Datenbank eingerichtet und mit der Option **Wählbar im Frontend** gekennzeichnet.
- Es ist eine Sprache als Fallback-Variante für die Datenbank bekannt. In der Standardinstallation des One Identity Manager wird die Sprache **English - United States [en-US]** verwendet. Diese Sprache wird verwendet, wenn bei einer sprachabhängigen Datenauflösung für die angeforderte Anmeldesprache des Benutzers keine übersetzen Bezeichnungen gefunden werden.
- Zur Verwendung sprachabhängiger Anzeigetexte in der Benutzeroberfläche sind die Spaltendefinitionen mit der Option **Mehrsprachig** gekennzeichnet.
- Es sind die Quelle der Übersetzung und das Ziel der Übersetzung bekannt.
- Zur Ausgabe sprachabhängiger Informationen innerhalb von Visual Basic .NET-Ausdrücken wird die #LD-Notation genutzt. Die #LD Texte werden für die Übersetzung automatisch extrahiert. Dazu muss eine Spalte als #LD Inhalt gekennzeichnet sein.

Die Übersetzungen werden in der Tabelle `DialogMultiLanguage` abgelegt. In der Tabelle wird ein Schlüssel, die Sprache und die sprachabhängige Ersetzung eingetragen.

Beispiel:

Als Spaltenbezeichnung (`DialogColumn.Caption`) für die Spalte `QERResource.Ident_QERResource` soll in der Anmeldesprache **English - United States [en-US]** der Text **Resource** angezeigt werden. Für die Anmeldesprache **German - Germany [de-DE]** soll die Spaltenbezeichnung **Ressource** verwendet werden.

In der Spalte `QERResource.Ident_QERResource` ist der Wert **Car** eingetragen. Einem Benutzer mit der Anmeldesprache **English - United States [en-US]** soll der Wert

Car angezeigt werden. Einem Benutzer mit der Anmeldesprache **German - Germany [de-DE]** soll der Wert **Auto** angezeigt werden.

Tabelle 67: Beispiel für sprachabhängige Einträge in der Tabelle DialogMultiLanguage

Anzeige für Spalte	Schlüssel	Sprache	Wert
DialogColumn.Caption	Resource	English - United States [en-US]	Resource
	Resource	German - Germany [de-DE]	Ressource
QERResource.Ident_ QERResource	Car	English - United States [en-US]	Car
	Car	German - Germany [de-DE]	Auto

Verwandte Themen

- Spalten zur Übersetzung kennzeichnen auf Seite 235
- Verwenden des Textspeichers zur Übersetzung auf Seite 237
- Übersetzungen im Wörterbucheditor anzeigen auf Seite 238
- Spracheinstellungen für die Anzeige und Pflege der Daten auf Seite 40
- Verwendung der #LD-Notation auf Seite 389

Spalten zur Übersetzung kennzeichnen

Damit der Inhalt einer Spalte übersetzt werden kann, müssen Sie die Spalte für die Übersetzung kennzeichnen.

Um eine Spalte für die Übersetzung zu kennzeichnen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Spalte** und bearbeiten Sie die Eigenschaft **Mehrsprachig**. Legen Sie die folgenden Einstellungen fest:

- **Übersetzungsziel**: Der Inhalt der Spalte soll übersetzt angezeigt werden.
- **Übersetzungsquelle**: Die Spalte liefert Übersetzungen.
- **#LD Inhalt**: Die Spalte besitzt Inhalte in #LD-Notation. Die Inhalte werden für die Übersetzung extrahiert.
- **Ohne Fallback aus Textspeicher**: Der Textspeicher soll für die Spalte nicht als Fallback verwendet werden.

Sie können die Werte kombinieren. Aus der Kombination der Werte ergibt sich, wie die Übersetzung ermittelt wird.

5. Ein Übersetzungsziel wird in der Regel selbst auch die Übersetzungsquelle sein. Sollen die Übersetzungen jedoch aus einer anderen Übersetzungsquelle ermittelt werden, geben Sie diese zusätzlich als Sprachabhängigkeit an.
   - Wechseln Sie auf den Tabreiter **Sprachabhängigkeiten** und wählen Sie unter **Quelle für Übersetzungen** die Spalte, die als Übersetzungsquelle dienen soll.

     HINWEIS: Stellen Sie sicher, dass die Spalte, die als Übersetzungsquelle verwendet wird, mit dem Wert **Übersetzungsquelle** gekennzeichnet ist.

Beispiel: Eine Spalte ist Übersetzungsziel und Übersetzungsquelle.

Der Inhalt der Spalte `QERResource.Ident_QERResource` soll übersetzt werden. In der Spalte `Ident_QERResource` ist der Wert **Car** eingetragen. Einem Benutzer mit der Anmeldesprache **English - United States [en-US]** soll der Wert **Car** angezeigt werden. Einem Benutzer mit der Anmeldesprache **German - Germany [de-DE]** soll der Wert **Auto** angezeigt werden. Die Übersetzung selbst soll an der Spalte `QERResource.Ident_QERResource` gepflegt werden.

- Kennzeichnen Sie die Spalte `QERResource.Ident_QERResource` in der Eigenschaft **Mehrsprachig** mit dem Wert **Übersetzungsziel**.
- Kennzeichnen Sie die Spalte `QERResource.Ident_QERResource` in der Eigenschaft **Mehrsprachig** mit dem Wert **Übersetzungsquelle**.
- Übersetzen Sie im Wörterbucheditor die Einträge für die Spalte `Ident_QERResource` der Tabelle `QERResource`.

Bei Laden der Spalte wird ermittelt, dass die Spalte `QERRessource.Ident_QERResource` übersetzt werden soll. Zur Übersetzung wird aus der Tabelle `DialogMultiLanguage` der entsprechende Schlüssel für die Spalte `QERRessource.Ident_QERResource` ermittelt und der Wert angezeigt, der für die Anmeldesprache des Benutzers hinterlegt ist.

Beispiel: Eine Spalte ist Übersetzungsziel und bezieht ihre Übersetzungen aus einer anderen Übersetzungsquelle.

In der Prozessansicht des Manager wird die ausgeführte Aktion in der Anmeldesprache des Benutzers angezeigt. In der Prozessverfolgung wird der Inhalt der Spalte `DialogProcess.DisplayName` aus der Spalte `JobEventGen.ProcessDisplay` ermittelt. Zur Bildung der Anzeigetexte kann die Spalte `JobEventGen.ProcessDisplay` die #LD-Notation verwenden.

- Kennzeichnen Sie die Spalte `JobEventGen.ProcessDisplay` in der Eigenschaft **Mehrsprachig** mit dem Wert **Übersetzungsquelle** und dem Wert **#LD Inhalt**.
- Kennzeichnen Sie die Spalte `DialogProcess.DisplayName` in der Eigenschaft **Mehrsprachig** mit dem Wert **Übersetzungsziel** und tragen Sie als **Sprachabhängigkeit** die Spalte `JobEventGen.ProcessDisplay` ein.
- Übersetzen Sie im Wörterbucheditor die Einträge für die Spalte `ProcessDisplay` der Tabelle `JobEventGen` .

Bei Laden der Prozessansicht wird ermittelt, dass die Spalte `DialogProcess.DisplayName` übersetzt werden soll. Zur Übersetzung wird aus der Tabelle `DialogMultiLanguage` der entsprechende Schlüssel für die Spalte `JobEventGen.ProcessDisplay` ermittelt und der Wert angezeigt, der für die Anmeldesprache des Benutzers hinterlegt ist.

Verwandte Themen

- Verwenden des Textspeichers zur Übersetzung auf Seite 237
- Übersetzungen im Wörterbucheditor anzeigen auf Seite 238
- Eigenschaften von Spaltendefinitionen auf Seite 99
- Verwendung der #LD-Notation auf Seite 389

Verwenden des Textspeichers zur Übersetzung

Übersetzungen, die häufig vorkommen oder nicht einer bestimmten Datenbankspalte zugeordnet werden können, können in einem Textspeicher (Tabelle `QBMTranslationAddOnSource`) abgelegt werden. So bezieht beispielsweise das Web Portal Übersetzungen von Anzeigetexten aus diesem Textspeicher. Ebenso werden Ausgabetexte aus Datenbanktriggern aus dem Textspeicher ermittelt. Den Textspeicher können Sie als Übersetzungsquelle referenzieren.

Zusätzlich wird der Textspeicher als Fallback eingesetzt, wenn über andere Übersetzungsquellen keine passende Übersetzung gefunden wird.

TIPP: Um die Verwendung des Textspeichers als Fallback abzuschalten, kennzeichnen Sie die Spalte in der Eigenschaft **Mehrsprachig** mit dem Wert **Ohne Fallback aus Textspeicher**.

Um Einträge für den Textspeicher zu erfassen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung > Übersetzbare Texte**.
2. Wählen Sie **Objekt > Neu** und erfassen Sie den zu übersetzenden Schlüssel.
3. Übersetzen Sie im Wörterbucheditor die Einträge für `QBMTranslationAddOnSource.Entrykey`.

Verwandte Themen

- Spalten zur Übersetzung kennzeichnen auf Seite 235

Übersetzungen im Wörterbucheditor anzeigen

Mit dem Wörterbucheditor können Sie Übersetzungen vornehmen für:

- Inhalte von Spalten, die für die mehrsprachige Eingabe gekennzeichnet sind
- #LD - Ausdrücke aus Spalten, die VB.Net-Code enthalten
- Texte, die im Textspeicher (Tabelle `QBMTranslationAddOnSource`) abgelegt sind

Alle Einträge, für die Sie Übersetzungen vornehmen können, werden in der Übersetzungstabelle des Wörterbucheditors dargestellt.

Um die Übersetzungen anzuzeigen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung**.
2. Starten Sie den Wörterbucheditor über die Aufgabe **Übersetzungen bearbeiten**.

   Angezeigt werden die folgenden Informationen.

   Tabelle 68: Informationen in der Übersetzungstabelle

Eigenschaften	Bedeutung
Status	Status des Eintrags.
Tabelle	Tabelle der Übersetzungsquelle.
Spalte	Spalte der Übersetzungsquelle.
Quelle	Gibt an, aus welcher Datenquelle der Schlüsselwert kommt. Zulässige Werte sind **Daten**, **Bitmaske**, **Liste zulässiger**

Eigenschaften	Bedeutung
	Werte, **Teil einer MVP-Spalte**, **DBQueue Prozessor**, **LD-Notation**, **Web** und **Extern**.
Verwendung	Anzahl der Verwendung.
Schlüssel	Schlüsselwert, der übersetzt wird.
Geprüft	Gibt an, ob die Übersetzung geprüft ist.
Sprache	Übersetzung in der gewählten Sprache.

TIPP: Über einen Mausklick auf eine Spalte im Tabellenkopf wird nach der gewählten Spalte sortiert.

Verwandte Themen

- Verwendung einer Übersetzung anzeigen auf Seite 239
- Übersetzungen einer einzelnen Tabelle bearbeiten auf Seite 240
- Alle Übersetzungen bearbeiten auf Seite 241
- Schlüsseltext einer Übersetzung ändern auf Seite 242

Verwendung einer Übersetzung anzeigen

Eine Übersetzung kann an mehreren Stellen verwendet werden. Wenn Sie eine Übersetzung ändern, wird diese an allen Stellen angezeigt. Bevor Sie eine Übersetzung ändern, prüfen Sie im Designer die Verwendung der Übersetzung.

Um die Verwendung eines Schlüsseltextes anzuzeigen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung**.
2. Starten Sie den Wörterbucheditor über die Aufgabe **Übersetzungen bearbeiten**.
3. Wählen Sie für den gewünschten Eintrag in der Spalte **Verwendung** das Kontextmenü **Verwendung anzeigen**.

   Es wird ein Dialogfenster geöffnet, in dem alle Fundstellen angezeigt werden. Mit Maus-Doppelklick auf einen Eintrag sehen Sie die erweiterten Eigenschaften zum Objekt.

Verwandte Themen

- Übersetzungen einer einzelnen Tabelle bearbeiten auf Seite 240
- Alle Übersetzungen bearbeiten auf Seite 241
- Schlüsseltext einer Übersetzung ändern auf Seite 242

Übersetzungen einer einzelnen Tabelle bearbeiten

HINWEIS:

- Eine Übersetzung kann an mehreren Stellen verwendet werden. Wenn Sie eine Übersetzung ändern, wird diese an allen Stellen angezeigt. Bevor Sie eine Übersetzung ändern, prüfen Sie im Designer die Verwendung der Übersetzung.
- Benutzer können nur Übersetzungen für die Objekte und Spalten bearbeiten, für die sie die Bearbeitungsberechtigungen besitzen.
- Um alle Übersetzungen zu bearbeiten, benötigen die Benutzer die Programmfunktion **Erlaubt die Übersetzung von Textdaten unabhängig von den Bearbeitungsberechtigungen für das Basisobjekt** (`Common_Translation`).

Um die Inhalte einer einzelnen Tabelle zu übersetzen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Wörterbucheditor über die Aufgabe **Übersetzungen in der Tabelle bearbeiten**.

   Der Wörterbucheditor zeigt alle vorhanden Übersetzungen für die Spalten der gewählten Tabelle, die für die Übersetzung gekennzeichnet sind.
3. Wählen Sie über den Eintrag **Sprachen wählen** in der Symbolleiste, die Sprachen, deren Übersetzungen Sie bearbeiten möchten.
4. Um Einträge zu ermitteln, für welche noch keine Übersetzung vorhanden ist, klicken Sie .
5. Bearbeiten Sie die Übersetzungen.
   a. Schalten Sie über Maus-Doppelklick das Eingabefeld frei und tragen Sie die Übersetzung ein.
   b. Wenn Sie eine Übersetzung überprüft haben, setzen Sie die Übersetzung auf **Geprüft**.

   TIPP:

   - Über **Strg + K** können Sie den Schlüsselwert in die Übersetzung übernehmen.
   - Über die Schaltfläche können Sie eine Rechtschreibprüfung ausführen lassen.
   - Verwenden Sie das Kontextmenü **Bearbeiten** um:
     - die Datenquelle eines Schlüsselwertes einzutragen
     - einen zusätzlichen Kommentar
     - eine Übersetzung für eine Sprache zu löschen

Um die sprachabhängigen Informationen allen Systembenutzern zur Verfügung zu stellen, muss nach der Übernahme der Änderungen in die Hauptdatenbank der DBQueue Prozessor die Systemdaten neu berechnen.

Ausführliche Informationen zur Übersetzung der zulässigen Werte einer Spalte finden Sie unter Zulässige Werte von Spalten auf Seite 92.

Verwandte Themen

- Verwendung einer Übersetzung anzeigen auf Seite 239
- Alle Übersetzungen bearbeiten auf Seite 241
- Schlüsseltext einer Übersetzung ändern auf Seite 242

Alle Übersetzungen bearbeiten

HINWEIS:

- Eine Übersetzung kann an mehreren Stellen verwendet werden. Wenn Sie eine Übersetzung ändern, wird diese an allen Stellen angezeigt. Bevor Sie eine Übersetzung ändern, prüfen Sie im Designer die Verwendung der Übersetzung.
- Benutzer können nur die Objekte und Spalten bearbeiten, für die sie die Bearbeitungsberechtigungen besitzen.
- Um alle Übersetzungen zu bearbeiten, benötigen die Benutzer die Programmfunktion **Erlaubt die Übersetzung von Textdaten unabhängig von den Bearbeitungsberechtigungen für das Basisobjekt** (`Common_Translation`).

Um alle Übersetzungen im Wörterbucheditor anzuzeigen und zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung**.
2. Starten Sie den Wörterbucheditor über die Aufgabe **Übersetzungen bearbeiten**.
3. Wählen Sie über den Eintrag **Sprachen wählen** in der Symbolleiste, die Sprachen, deren Übersetzungen Sie bearbeiten möchten.
4. Um Einträge zu ermitteln, für welche noch keine Übersetzung vorhanden ist, klicken Sie .
5. Schränken Sie bei Bedarf die angezeigten Daten über Filter ein.
   - Über einen Mausklick auf eine Spalte im Tabellenkopf wird nach der gewählten Spalte sortiert.
   - Um die Einträge einer Spalte einzuschränken, klicken Sie auf den Pfeil im Tabellenkopf einer Spalte. Es wird ein Eingabefeld geöffnet, im welches Sie den Text eingeben, nach dem gefiltert werden soll. Ist ein Filter definiert, wird im Tabellenkopf ein Symbol angezeigt. Im den Filter zu entfernen, klicken Sie auf den Pfeil im Tabellenkopf und wählen Sie den Eintrag **Filter entfernen**.
   - Zusätzlich können die Sie über den Eintrag **Darstellung filtern** in der Symbolleiste einen Filter eingeben. Über das Symbol wird der Filter auf die Schlüssel und die Übersetzungen angewendet. Um den Filter zurückzusetzen, verwenden Sie das Symbol . Über das Symbol können Sie zusätzliche

Filterabfragen über Platzhalter, Volltextsuche oder SQL Abfragen erstellen und speichern.

Wenn Sie in einem Übersetzungsfeld einen Eintrag per Maus-Doppelklick zur Bearbeitung freischalten und dann markieren, wird dieser automatisch in das Eingabefeld **Darstellung filtern** der Darstellung übernommen.

6. Bearbeiten Sie die Übersetzungen.
   a. Schalten Sie über Maus-Doppelklick das Eingabefeld frei und tragen Sie die Übersetzung ein.
   b. Wenn Sie eine Übersetzung überprüft haben, setzen Sie die Übersetzung auf **Geprüft**.

TIPP:

- Über **Strg + K** können Sie den Schlüsselwert in die Übersetzung übernehmen.
- Über die Schaltfläche können Sie eine Rechtschreibprüfung ausführen lassen.
- Verwenden Sie das Kontextmenü **Bearbeiten** um:
  - die Datenquelle eines Schlüsselwertes einzutragen
  - einen zusätzlichen Kommentar
  - eine Übersetzung für eine Sprache zu löschen

Um die sprachabhängigen Informationen allen Systembenutzern zur Verfügung zu stellen, muss nach der Übernahme der Änderungen in die Hauptdatenbank der DBQueue Prozessor die Systemdaten neu berechnen.

Verwandte Themen

- Verwendung einer Übersetzung anzeigen auf Seite 239
- Übersetzungen einer einzelnen Tabelle bearbeiten auf Seite 240
- Schlüsseltext einer Übersetzung ändern auf Seite 242

Schlüsseltext einer Übersetzung ändern

Mit dieser Aufgabe ändern Sie den Schlüssel einer Übersetzung (`Entrykey`) in der Tabelle `DialogMultiLanguage` und den Quelltext an allen Objekten, die diesen Schlüssel verwenden, beispielsweise eine Spaltenbezeichnung oder einen Beschreibungstext.

WICHTIG: Bevor Sie einen Schlüsseltext ändern, prüfen Sie im Designer die Verwendung der Übersetzung.

Um einen Schlüsseltext zu ändern

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung**.
2. Starten Sie den Wörterbucheditor über die Aufgabe **Übersetzungen bearbeiten**.

3. Wählen Sie für den gewünschten Eintrag in der Spalte **Schlüssel** das Kontextmenü **Schlüsseltext ändern**.

Verwandte Themen

- Verwendung einer Übersetzung anzeigen auf Seite 239

Übersetzungen aus dem Language Pack importieren

Die Standardinstallation des One Identity Manager wird in den Sprachen **English - United States [en-US]** und **German - Germany [de-DE]** ausgeliefert. Zur Übersetzung des Web Portal werden mit dem One Identity Manager Language Pack `CSV`-Dateien mit weiteren Sprachen bereitgestellt.

HINWEIS: Das One Identity Manager Language Pack finden Sie im Support Portal unter https://support.oneidentity.com/.

Der Import:

- Erstellt die Übersetzungen in der Tabelle `DialogMultiLanguage`.
- Aktualisiert bestehende Einträge, anhand des Schlüssels, der Tabelle und der Spalte.
- Löscht keine Einträge.

Um die Sprachdateien zu importieren

1. Wählen Sie im Designer die Kategorie **Basisdaten > Lokalisierung**.
2. Starten Sie den Wörterbucheditor über die Aufgabe **Übersetzungen bearbeiten**.
3. Klicken Sie in der Symbolleiste des Editors .
4. Wählen Sie `*.CSV`-Datei mit der benötigten Sprache und klicken Sie **Öffnen**.

   Der Import wird gestartet. Der Vorgang kann einige Zeit in Anspruch nehmen.
5. Übertragen Sie die Änderungen in die Hauptdatenbank. Verwenden Sie den Menüeintrag **Datenbank > Übertragung in die Datenbank**.

   Um die sprachabhängigen Informationen allen Systembenutzern zur Verfügung zu stellen, muss nach der Übernahme der Änderungen in die Hauptdatenbank der DBQueue Prozessor die Systemdaten neu berechnen.

8

Prozess-Orchestrierung im One Identity Manager

Der One Identity Manager verwendet zur Abbildung von Geschäftsprozessen sogenannte Prozesse. Ein Prozess besteht aus Prozessschritten, die Verarbeitungsaufgaben darstellen und über Vorgänger-Nachfolger-Beziehungen miteinander verbunden sind. Dieses Funktionsprinzip erlaubt es, flexibel Aktionen und Abläufe an die Ereignisse von Objekten zu koppeln. Die Modellierung der Prozesse erfolgt über Prozessvorlagen. Die Umwandlung der als Skript definierten Vorlagen in Prozessen und Prozessschritten in einen konkreten Prozess in der Jobqueue übernimmt der Jobgenerator.

Die Verarbeitung der definierten Prozesse erfolgt über den One Identity Manager Service. Zur Prozessverarbeitung muss der Dienst auf den Servern des One Identity Manager-Netzwerkes installiert sein. Die Server müssen in der One Identity Manager-Datenbank als Jobserver bekannt gegeben werden.

Der One Identity Manager Service ist die einzige Komponente des One Identity Manager, die berechtigt ist, Änderungen in den Zielsystemen auszuführen.

Zur Überwachung der Prozessverarbeitung nutzen Sie das Programm Job Queue Info. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.

Detaillierte Informationen zum Thema

- Einrichten von Jobservern auf Seite 289
- Abbildung von Prozessen im One Identity Manager auf Seite 244
- Arbeitsweise des One Identity Manager Service auf Seite 305

Abbildung von Prozessen im One Identity Manager

Der One Identity Manager verwendet zur Abbildung von Geschäftsprozessen sogenannte Prozesse. Ein Prozess besteht aus Prozessschritten, die Verarbeitungsaufgaben darstellen und über Vorgänger-Nachfolger-Beziehungen miteinander verbunden sind. Dieses

Funktionsprinzip erlaubt es, flexibel Aktionen und Abläufe an die Ereignisse von Objekten zu koppeln.

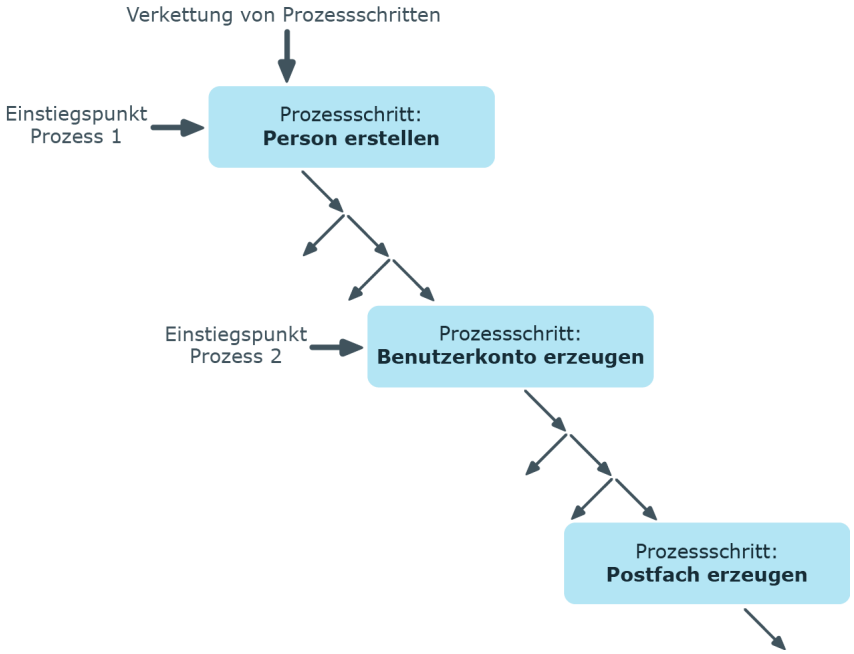
Um die einzelnen elementaren Arbeiten auf Systemebene zu verrichten, wie beispielsweise das Anlegen von Verzeichnissen, werden sogenannte Prozessfunktionen verwendet. Ein oder mehrere Prozessfunktionen und deren Parameter sind in Prozesskomponenten zusammengefasst. Die Prozesskomponenten sowie deren Prozessfunktionen und Parameter sind in den Tabellen `Jobcomponent`, `Jobtask` und `Jobparameter` definiert. Die vordefinierten Konfigurationen werden durch die Schemainstallation gepflegt und sind bis auf einige Eigenschaften nicht bearbeitbar.

Die Modellierung der Prozesse erfolgt über Prozessvorlagen. Die Umwandlung der als Skript definierten Vorlagen in Prozessen und Prozessschritten in einen konkreten Prozess in der Jobqueue übernimmt der Jobgenerator.

In der nachfolgenden Abbildung ist als Beispiel eine Verkettung von Prozessschritten dargestellt, mit der Sie eine Person anlegen, anschließend zu dieser Person ein Active Directory Benutzerkonto einrichten und im letzten Schritt ein Postfach anlegen.

Diesen Ablauf können Sie durch einen Prozess abbilden. Innerhalb dieser Verkettung können Sie jedoch auch Einstiegspunkte für verschiedene Prozesse definieren. Der Einstiegspunkt des Prozesses 1 hat das Anlegen einer Person mit Active Directory Benutzerkonto und Postfach als Ergebnis. Der Einstiegspunkt des Prozesses 2 führt nur zum Anlegen eines Active Directory Benutzerkontos mit einem Postfach.

Abbildung 25: Erstellen einzelner Prozesse aus einer Prozessschrittverkettung



Verwandte Themen

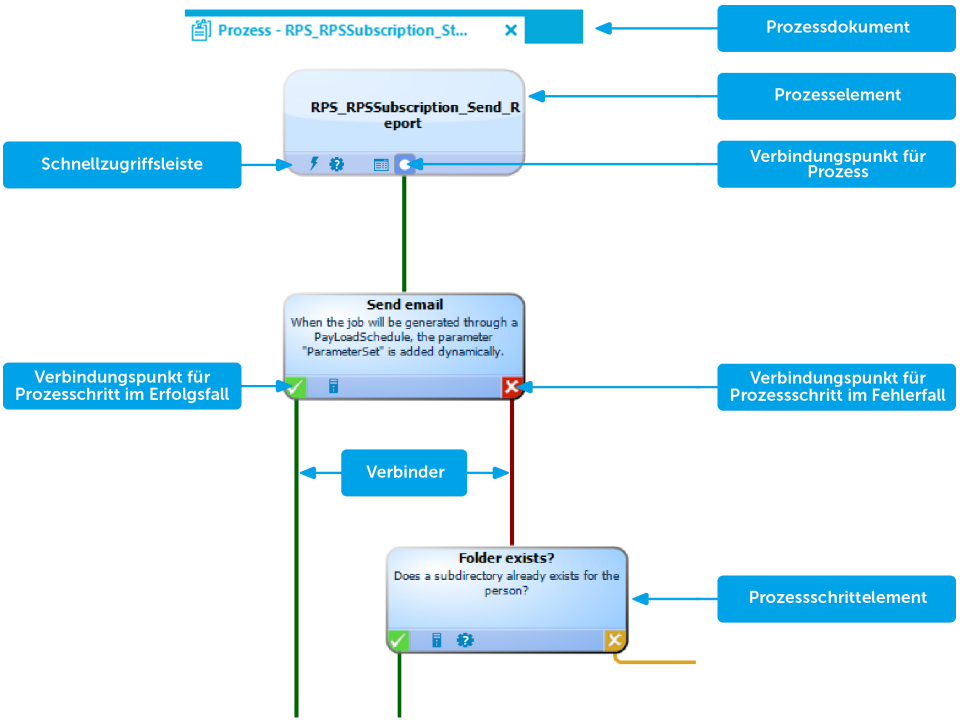
- Bearbeiten von Prozessen mit dem Prozesseditor auf Seite 246
- Definieren von Prozessen auf Seite 248

- Automatisierte Ausführung von Prozessen auf Seite 278
- Übersicht über die Prozesskomponenten auf Seite 281

Bearbeiten von Prozessen mit dem Prozesseditor

Prozesse bearbeiten Sie im Designer mit dem Prozesseditor. Im Prozesseditor wird ein Prozess mit seinen Prozessschritten in einem Prozessdokument zusammengefasst und über spezielle Steuerelemente dargestellt und bearbeitet.

Abbildung 26: Darstellung eines Prozesses im Prozesseditor



Beim Einfügen eines neuen Prozesses wird zunächst ein neues Prozessdokument mit einem Prozesselement erzeugt. Beim Einfügen eines Prozessschrittes wird ein zugehöriges Prozessschrittelement erzeugt.

Die einzelnen Elemente verketten Sie über Verbinder miteinander. Die Verbindungspunkte aktivieren Sie mausgesteuert.

- Um eine Verbindung herzustellen, klicken Sie auf einen Verbindungspunkt, halten Sie die linke Maustaste gedrückt und ziehen Sie einen Verbinder zum zweiten Verbindungspunkt.
- Um eine Verbindung zu entfernen, aktivieren Sie den Ausgangspunkt der Verbindung erneut per Mausklick. Bestätigen Sie die Sicherheitsabfrage mit **OK**.

Mit Maus-Doppelklick auf ein Prozesselement oder ein Prozessschrittelement wird die jeweilige Bearbeitungsansicht geöffnet, in welcher Sie den Prozess oder den Prozessschritt bearbeiten.

Die Elemente verfügen über einen Tooltip. Der Tooltip eines Prozesselementes zeigt den Namen und die Beschreibung des Prozesses. Der Tooltip eines Prozessschrittelementes zeigt den Namen und die Beschreibung des Prozessschrittes sowie die Beschreibung der verwendeten Prozessfunktion.

Die Elemente besitzen eine Schnellzugriffsleiste. Die Symbole repräsentieren spezielle Eigenschaften des Prozesses oder Prozessschrittes. Der Tooltip eines Symbols zeigt genauere Informationen zur Eigenschaft. Bei Maus-Doppelklick auf ein Symbol wird der Bearbeitungsansicht des Prozesses oder des Prozessschrittes geöffnet und zur entsprechenden Eigenschaft gewechselt.

Tabelle 69: Symbole der Schnellzugriffsleisten

Symbol	Bedeutung
	Es sind Ereignisse definiert.
	Der Prozess wird nicht generiert.
	Es ist eine Wartezeit bei Fehlern festgelegt.
	Es erfolgt eine Aufzweigung der Verarbeitung. Der Verbindungspunkt im Fehlerfall und der Verbinder zur nachfolgenden Prozessschritt werden gelb dargestellt.
	Fehler bei der Ausführung werden ignoriert. Der Verbindungspunkt im Fehlerfall wird grau dargestellt. Kein Prozessschritt im Fehlerfall möglich.
	Tritt ein Fehler auf, dann werden keine weiteren Prozessschritte dieses Prozesses verarbeitet.
	Es ist eine Generierungsbedingung eingetragen.
	Die Prozessinformationen sind aktiviert.
	Es ist eine Skript zur Serverauswahl oder Servermaske eingetragen.
	Die Benachrichtigung im Fehlerfall oder die Benachrichtigung im Erfolgsfall ist aktiviert.
	Der Prozess oder der Prozessschritte wurde kundenspezifisch angepasst. Ein Tooltip zeigt weitere Informationen zu den Anpassungen.

Einige wichtige Eigenschaften werden durch die Farbe der Elemente angezeigt.

Tabelle 70: Farben der Elemente

Farbe	Bedeutung
Blau	Standard.

Farbe	Bedeutung
Gelb	Die Gültigkeitskontrolle ergibt eine Warnung oder eine Information.
Rot	Die Gültigkeitskontrolle ergibt einen Fehler.
Grau	Der Prozess ist deaktiviert.

Sie können die Elemente im Prozessdokument mausgesteuert verschieben. Über das Kontextmenü **Anordnen** setzen Sie die Elemente wieder auf ihre Standardpositionen zurück. Mit dem Speichern des gesamten Prozesses werden die Positionen der einzelnen Elemente in die One Identity Manager-Datenbank übertragen und stehen damit bei Neustart des Designer für alle Benutzer zur Verfügung.

Definieren von Prozessen

WICHTIG: Der Prozess und die Prozessschrittes werden in der One Identity Manager-Datenbank erst durch das Speichern des gesamten Prozesses erzeugt. Der Prozess ist ab diesem Zeitpunkt für weitere Benutzer im Prozesseditor bearbeitbar, kann jedoch noch nicht generiert werden. Die Generierung wird erst durch Kompilieren des Prozesses möglich.

Bei Bedarf können Sie die Standardprozesse an Ihre kundenspezifischen Anforderungen anpassen. Um zu einem Prozess weitere Prozessschritte hinzuzufügen, erstellen Sie einen kundenspezifischen Prozess.

Um einen Prozess zu erstellen sind folgende Schritte erforderlich

1. Erstellen eines Prozesses.
2. Festlegen des auszulösenden Ereignisses.
3. Erstellen der Prozessschritte.
4. Bearbeiten der Parameter.
5. Testen des Prozesses.
6. Kompilieren des Prozesses.

Verwandte Themen

- Bearbeiten von Prozessen mit dem Prozesseditor auf Seite 246
- Prozesse erstellen und bearbeiten auf Seite 249
- Prozesse kopieren auf Seite 249
- Prozessschritte erstellen und bearbeiten auf Seite 250
- Einzelne Prozessschritte kopieren auf Seite 251
- Prozessschritte in einen Prozess kopieren auf Seite 252
- Eintrag innerhalb eines Prozesses suchen auf Seite 253

- Parameter für Prozessschritte auf Seite 260
- Ereignisse für Prozesse auf Seite 264
- Berechtigungen zum Auslösen von Prozessen auf Seite 266
- Prozessgenerierung simulieren auf Seite 267
- Gültigkeitsprüfung eines Prozesses auf Seite 269
- Kompilieren von Prozessen auf Seite 270

Prozesse erstellen und bearbeiten

Um einen vorhandenen Prozess zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess aus.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
   Der Prozess wird im Prozesseditor geöffnet.

Um einen neuen Prozess zu erstellen

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung**.
2. Starten Sie den Prozesseditor über die Aufgabe **Einen neuen Prozess** erstellen.
   Ein neues Element für den Prozess wird erzeugt und im Prozesseditor geöffnet.

Verwandte Themen

- Prozesse kopieren auf Seite 249
- Eigenschaften für Prozesse auf Seite 254
- Prozesse vergleichen auf Seite 253
- Prozesse exportieren und importieren auf Seite 254

Prozesse kopieren

Um einen Prozess zu kopieren, wird eine Assistent bereitgestellt.

Um einen Prozess zu kopieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozess > Kopieren**.
4. Auf der Startseite des Assistenten legen Sie die Kopieroptionen fest.

Tabelle 71: Kopieroptionen

Option	Bedeutung
Prozessschritte umbenennen	Setzen Sie diese Option, so haben Sie im Assistenten die Möglichkeit die einzelnen Prozessschritte umzubenennen.
Ereignisse kopieren	Aktivieren Sie diese Option damit die Ereignisse, die diesem Prozess zugeordnet sind, ebenfalls kopiert werden.
Ausgangsprozess deaktivieren	Mit dieser Option legen Sie fest, ob der Ausgangsprozess nach der Kopie deaktiviert werden soll. Wenn Sie diese Option aktivieren, dann wird für den Ausgangsprozess die Option **nicht generieren** aktiviert.
Kopierten Prozess deaktivieren	Mit dieser Option legen Sie fest, ob der Prozess nach der Kopie deaktiviert werden soll. Wenn Sie diese Option aktivieren, dann wird für den kopierten Prozess die Option **nicht generieren** aktiviert.

5. Auf der Seite **Kopieroptionen** legen Sie den Namen des neuen Prozesses fest.
6. (Optional) Auf der Seite **Namen der Prozessschritte festlegen** benennen Sie die einzelnen Prozessschritte.

   Durch Mausklick auf den neuen Namen des Prozessschrittes können Sie diesen ändern.

   HINWEIS: Dieser Schritt wird nur angeboten, wenn Sie die Kopieroption **Prozessschritte umbenennen** gesetzt haben.
7. Um den Kopiervorgang zu starten, klicken Sie auf der letzten Seite des Assistenten **Fertig**.

   Der Prozess wird im Prozesseditor geöffnet und kann weiter bearbeitet werden.

Verwandte Themen

- Prozesse erstellen und bearbeiten auf Seite 249
- Prozesse vergleichen auf Seite 253
- Prozesse exportieren und importieren auf Seite 254

Prozessschritte erstellen und bearbeiten

Um einen neuen Prozessschritt zu erstellen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozessschritt > Neu**.

Ein neues Element für den Prozessschritt wird erzeugt und im Prozesseditor angezeigt.

4. Bearbeiten Sie in der Ansicht **Prozessschritteigenschaften** die Stammdaten des Prozessschrittes.
5. Verknüpfen Sie den Prozessschritt innerhalb des Prozesses.

Um einen vorhandenen Prozessschritt zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozessschritt.

   HINWEIS: Um mehrere Prozessschritte zu bearbeiten, halten Sie die **Strg**-Taste gedrückt und klicken Sie die Prozessschritte an.

   In der Bearbeitungsansicht für Prozessschritte werden Eingabefelder, deren Werte unterschiedlich belegt sind, mit dem Symbol hervorgehoben. Bei Bearbeitung eines Eingabefeldes und Speichern der Änderung wird der Wert in alle gewählten Prozessschritte übernommen.

Verwandte Themen

- Einzelne Prozessschritte kopieren auf Seite 251
- Prozessschritte in einen Prozess kopieren auf Seite 252
- Eigenschaften für Prozessschritte auf Seite 256

Einzelne Prozessschritte kopieren

Um einen Prozessschritt zu kopieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den zu kopierenden Prozessschritt und kopieren Sie den Prozessschritt über das Kontextmenü **Kopieren** oder **Strg + C** in die Zwischenablage.

   HINWEIS: Um mehrere Prozessschritte zu kopieren, halten Sie die **Strg**-Taste gedrückt und klicken Sie die Prozessschritte an.
4. Fügen Sie den Prozessschritt über das Kontextmenü **Einfügen** oder **Strg + V** ein.

   Der Prozessschritt erhält einen neue UID, alle Eigenschaften des Prozessschrittes werden übernommen.
5. Bearbeiten Sie die Stammdaten des Prozesses.
6. Verknüpfen Sie den Prozessschritt innerhalb des Prozesses.

Verwandte Themen

- Prozessschritte erstellen und bearbeiten auf Seite 250
- Prozessschritte in einen Prozess kopieren auf Seite 252
- Eigenschaften für Prozessschritte auf Seite 256

Prozessschritte in einen Prozess kopieren

Um einen Prozessschritt zu importieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie dem Menüeintrag **Prozessschritt > Importieren**.
4. Wählen Sie in der Bearbeitungsansicht die Ansicht **Prozessschritte suchen und importieren**.
5. Geben Sie in das Eingabefeld den Suchtext ein.
6. Legen Sie über die Suchoptionen fest, in welchen Objekten gesucht werden soll.

   In den angegebenen Objekten wird intern über eine Where-Klausel gesucht. Sind mehrere Objekte angegeben, wird diese intern über Join-Bedingungen erweitert.

   Tabelle 72: Durchsuchbare Objekte und Eigenschaften

Suchoptionen	Durchsucht werden die Eigenschaften
Prozess	Name
Prozessschritt	Name, Beschreibung, Generierungsbedingung, Skript zur Serverauswahl
Parameter	Name, Wert
Prozesskomponente	Komponentenklasse, Komponentenassembly
Prozessfunktion	Name
Parametervorlage	Name, Wertvorlage

7. Um die Suche zu starten, klicken Sie .

   Die gefunden Prozessschritte werden in der Ergebnisliste aufgeführt.
8. Wählen Sie in der Ergebnisliste der Suche den gewünschten Prozessschritt aus und klicken Sie .

   Der Prozessschritt wird in das Prozessdokument importiert.
9. Bearbeiten Sie die Stammdaten des Prozesses.
10. Verknüpfen Sie den Prozessschritt innerhalb des Prozesses.

Eintrag innerhalb eines Prozesses suchen

Um einen Eintrag innerhalb eines Prozesses zu suchen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Öffnen Sie den Suchendialog über **Strg + F**.
4. Geben Sie unter **Text** den Suchtext ein.
5. Starten Sie die Suche über die Schaltfläche **Suchen**.
6. Verwenden Sie **F3** zum weiteren suchen.

Der eingetragene Text wird im Prozess und in den Prozessschritten gesucht.

Tabelle 73: Durchsuchte Objekte und Eigenschaften

Suchen in Objekt	Durchsucht werden die Eigenschaften
Prozess	Name
Prozessschritt	Name, Beschreibung, Generierungsbedingung, Skript zur Serverauswahl

Prozesse vergleichen

Um Unterschiede zwischen zwei Prozessen zu ermitteln

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozess > Prozesse vergleichen**.

   In der Auswahlliste **Prozess A** ist der aktuelle Prozess vorausgewählt.
4. Wählen Sie in der Auswahlliste **Prozess B** den Prozess mit dem verglichen werden soll.
5. (Optional) Über die Schaltfläche legen Sie fest, welche Eigenschaften der Prozesse in den Vergleich einbezogen werden. Im Standard werden alle Eigenschaften des Prozesses, der Prozessschritte, der Parameter und der Ereignisse verglichen.
6. Starten Sie den Vergleich über die Schaltfläche .

   In der Auswertung werden die Unterschiede zwischen den Prozessen werden farblich hervorgehoben.

   TIPP: Um Text aus der Auswertung in die Zwischenablage zu kopieren, markieren Sie den Text und klicken Sie die Schaltfläche .

Prozesse exportieren und importieren

Der Export und der Import von Prozessen erfolgt über XML-Dateien.

Um einen Prozess in eine XML-Datei zu exportieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozess > Exportieren**.
4. Geben Sie den Dateinamen an und klicken Sie **Speichern**.

Um einen Prozess aus einer XML-Datei zu importieren

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung**.
2. Starten Sie den Import über die Aufgabe **Prozesse importieren**.
3. Wählen Sie die XML-Datei und klicken Sie **Öffnen**.

   Der Prozess wird im Prozesseditor geöffnet.

Verwandte Themen

- Prozesse erstellen und bearbeiten auf Seite 249

Eigenschaften für Prozesse

Tabelle 74: Eigenschaften eines Prozesses

Eigenschaft	Bedeutung
Bezeichnung	Name des Prozesses. Der Name eines Prozesses muss eindeutig sein. Kennzeichnen Sie kundenspezifische Prozesse mit dem Präfix **CCC_**.
Tabelle	Basisobjekt (Tabelle), auf dessen Ereignisse der Prozess generiert wird.
Beschreibung	Zusätzliche Beschreibung zum Prozess.
Bemerkungen	Zusätzliche Bemerkungen zum Prozess.
Prozess	UID des Prozesses. Diese kann nicht bearbeitet werden.
Prozessinformation	Gibt an, ob die Ausführung dieses Prozesses aufgezeichnet werden sollen. Die Aufzeichnung erfolgt abhängig vom Konfigurationsparameter **Common | ProcessState | ProgressView**. Zulässige Werte sind:

Eigenschaft	Bedeutung
	• **Keine**: Die Prozessinformationen werden nicht aufgezeichnet. • **Komplette Prozessverfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager angezeigt. • **Web Portal Verfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager und im Web Portal angezeigt.
Prozessinformation Bezeichnung	VB.Net-Ausdruck zur Bildung des Anzeigenamens in der Prozessanzeige.
Prä-Skript zur Generierung	Das Prä-Skript wird vor der Ausführung von anderen Skripten ausgeführt. Über das Prä-Skript können Sie beispielsweise globale Variablen ermitteln oder prozesslokale Variablen definieren, die dann die innerhalb der Prozesse und der Prozessschritte weiterverwendet werden können, beispielsweise in Generierungsbedingungen, Serverauswahlskripten oder in den Parametern. HINWEIS: Bei der Verarbeitung eines Prozesses wird zuerst das Prä-Skript zur Generierung ausgeführt und anschließend die Generierungsbedingung ausgewertet.
Generierungsbedingung	Für einen Prozess definieren Sie eine Bedingung in VB.Net-Syntax, anhand der wird entschieden, ob es notwendig ist, den Prozess zu generieren. Ist eine Generierungsbedingung angegeben, wird der Prozess nur generiert, wenn die Bedingung erfüllt ist. Beispielskripte finden Sie auf dem Installationsmedium im Verzeichnis `QBM\dvd\AddOn\SDK\ScriptSamples`.
Nicht generieren	Mit der Option entscheiden Sie, ob ein Prozess generiert wird. Ist die Option aktiviert, so wird der Prozess nicht generiert und kann nicht kompiliert werden. HINWEIS: Ist die Option für Prozesse aktiviert, so bleibt diese Option auch während einer Schemaaktualisierung aktiviert und wird nicht zurückgesetzt.
Präprozessorbedingung	Für den Prozess können Sie eine Präprozessorbedingung zur bedingten Kompilierung definieren. Damit ist ein Prozess nur verfügbar, wenn die Präprozessorbedingung erfüllt ist.
Deaktiviert durch Präprozessor	Ist ein Prozess durch eine Präprozessorbedingung deaktiviert, dann wird durch den Database Compiler die Option gesetzt.
Schwellwert (Warnung)	Maximale Anzahl dieser Prozesse für eine Queue, die gleich-

Eigenschaft	Bedeutung
	zeitig vorhanden sein dürfen. Wird diese Anzahl überschritten, wird eine Warnung versendet. Der One Identity Manager Service verarbeitet die Prozesse jedoch weiter.
Schwellwert (Sperrung)	Maximale Anzahl dieser Prozesse für eine Queue, die gleichzeitig vorhanden sein dürfen. Wird diese Anzahl überschritten werden weitere Prozesse auf den Status **Overlimit** gesetzt und durch den One Identity Manager Service nicht verarbeitet.

Verwandte Themen

- Verwenden von prozesslokalen Variablen und globalen Variablen für die Prozessdefinition auf Seite 272
- Schwellwerte für die Verarbeitung von Prozessen auf Seite 273
- Aufzeichnen von Prozessinformationen in der Prozessverarbeitung auf Seite 352
- Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369
- Verwendung von Visual Basic .NET-Skripten auf Seite 375

Eigenschaften für Prozessschritte

Tabelle 75: Allgemeine Eigenschaften eines Prozessschrittes

Eigenschaft	Bedeutung
Bezeichnung	Name des Prozessschrittes.
Prozessfunktion	Auszuführende Prozessfunktion einer Prozesskomponente. Mit der Auswahl einer Prozessfunktion legen Sie fest, welche Aktion durch den Prozessschritt ausgeführt wird. Die Parametervorlagen der Prozessfunktion werden kopiert und als Parameter in den Prozessschritt übernommen. Dadurch können Sie jedem Prozessschritt, der diese Prozessfunktion verwendet, andere Parameterwerte übergeben. Das Original wird nicht verändert.
Beschreibung	Zusätzliche Beschreibung zum Prozessschritt.
Priorität	Die Priorität legt fest, mit welcher Rangordnung der Prozessschritt in die Jobqueue eingestellt und verarbeitet wird. Es sind die Werte **1** bis **15** erlaubt. Je höher der Wert ist, desto eher wird der Prozessschritt verarbeitet.
Definition der Priorität	VB.Net- Ausdruck zur Bestimmung der Priorität in Abhängigkeit vom Inhalt des Prozesses.

Eigenschaft	Bedeutung

	Wenn ein Prozessschritt ein Skript für die dynamische Ermittlung der Priorität enthält, wird dieses Skript verwendet. Ansonsten wird die fest definierte Priorität verwendet.
	Beispiel:
	Kennwortänderungen eines Benutzerkontos sollen mit einer höheren Priorität (im Beispiel **7**) ausgeführt werden, Änderungen anderer Stammdaten mit Priorität **3**.
	`If $UserPassword[o]$ <> $UserPassword$ Then`
	`  Value = 7`
	`Else`
	`  Value = 3`
	HINWEIS: Das Eingabefeld wird beim Öffnen eines Prozessschrittes im Prozesseditor nicht angezeigt. Klicken Sie die Schaltfläche neben dem Feld **Priorität**, um das Eingabefeld einzublenden.

Prozessinformation	Gibt an, ob die Ausführung dieses Prozessschrittes aufgezeichnet werden sollen. Die Aufzeichnung erfolgt abhängig vom Konfigurationsparameter **Common | ProcessState | ProgressView**. Zulässige Werte sind: • **Keine**: Die Prozessinformationen werden nicht aufgezeichnet. • **Komplette Prozessverfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager angezeigt. • **Web Portal Verfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager und im Web Portal angezeigt.
Prozessinformation Bezeichnung	VB.Net-Ausdruck zur Bildung des Anzeigenamens in der Prozessanzeige.
Detailtiefe	Informationsgrad zur Abbildung der Prozessinformationen.
Benachrichtigung (Erfolg)	Gibt an, ob bei Erfolg eine Benachrichtigung versendet werden soll.
Benachrichtigung (Fehler)	Gibt an, ob bei Fehlern eine Benachrichtigung versendet werden soll.
Prä-Skript zur Generierung	Das Prä-Skript wird vor der Ausführung von anderen Skripten ausgeführt. Über das Prä-Skript können Sie beispielsweise

Eigenschaft	Bedeutung
	globale Variablen ermitteln oder prozesslokale Variablen definieren, die dann die innerhalb der Prozessschritte weiterverwendet werden können, beispielsweise in Generierungsbedingungen, Serverauswahlskripten oder in den Parametern.
Generierungsbedingung	Für einen Prozessschritt definieren Sie eine Bedingung in VB.Net-Syntax, anhand der wird entschieden, ob es notwendig ist, den Prozessschritt zu generieren. Ist eine Generierungsbedingung angegeben, wird der Prozessschritt nur generiert, wenn die Bedingung erfüllt ist.
Präprozessorbedingung	Für den Prozessschritt können Sie eine Präprozessorbedingung zur bedingten Kompilierung definieren. Damit ist ein Prozessschritt nur verfügbar, wenn die Präprozessorbedingung erfüllt ist.
Deaktiviert durch Präprozessor	Ist ein Prozessschritt durch eine Präprozessorbedingung deaktiviert, dann wird durch den Database Compiler die Option gesetzt.
Serverfunktion	Angabe der für diesen Prozessschritt zulässigen Servertypen. Über diese Angabe wird festgelegt, für welchen Server der Prozessschritt eingestellt werden soll. Die Auswahl muss zu einem eindeutigen Ergebnis führen, wie zum Beispiel SQL Ausführungsserver.
Skript zur Serverauswahl	Ist es für den Jobgenerator anhand der Serverfunktion nicht möglich zu entscheiden, welcher Server genutzt werden soll, kann hier ein Auswahlskript in VB.Net-Syntax zur näheren Bestimmung des Servers hinterlegt werden.
Wartemodus bei Fehlern	Ist zum Ausführungszeitpunkt des Prozessschrittes eine bestimmte Bedingung noch nicht erfüllt, so kann der One Identity Manager Service den Prozessschritt wiederholen. Die Aktivierung der Option bewirkt, dass der Prozessschritt abhängig von der Wartezeit (min) und den Wiederholversuchen nochmals ausgeführt wird.
Wartezeit (min)	Wartezeit in Minuten. Anzahl der Minuten, um die ein Prozessschritt, der als im Fehlerfall zurückzustellen markiert ist, in der wiederholten Ausführung verzögert wird.
Wiederholversuche	Anzahl der Wiederholversuche.
Aufzweigung der Verarbeitung	Prozessschritte, die nur zur Verzweigung der Prozessverarbeitung dienen, kennzeichnen Sie mit der Option. Ein Beispiel wäre ein Prozessschritt, welcher die Existenz eines Verzeichnisses überprüft. Abhängig vom zurückgelieferten

Eigenschaft	Bedeutung
	Ergebnis wird anschließend der Nachfolgeschritt im Erfolgsfall oder der Nachfolgeschritt im Fehlerfall, ohne Generierung einer Fehlermeldung, abgearbeitet.
Fehler ignorieren	Gibt an, ob Fehler bei der Ausführung ignoriert werden sollen. In diesem Fall wird der nachfolgende Prozessschritt auch ausgeführt, obwohl der vorgehende Prozessschritt nicht korrekt abgearbeitet wurde.
Stopp bei Fehlern	Tritt bei der Verarbeitung des Prozessschrittes ein Fehler auf, so bleibt der Prozessschritt in der Jobqueue und erhält den Status **Frozen**. In diesem Fall werden keine weiteren Prozessschritte des betroffenen Prozesses verarbeitet, sondern verbleiben in der Jobqueue. Die Prozessschritte mit dem Status **Frozen** können Sie im Programm Job Queue Info reaktivieren. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*. Ist der Konfigurationsparameter **Common | MailNotification | NotifyAboutWaitingJobs** aktiviert, wird bei Auftreten von Prozessen mit dem Status **Frozen** zusätzlich eine E-Mail Benachrichtigung versendet und ein entsprechender Eintrag im Ereignisprotokoll des Aktualisierungsservers erzeugt. Voraussetzung für die Nutzung des Benachrichtigungssystems ist die Einrichtung eines SMTP-Host für den Mailversand sowie die Aktivierung der Konfigurationsparameter für die Mailbenachrichtigung. Prozessschritte, die per SQL aus der Datenbank erzeugt werden, können immer mit der Option **Stopp bei Fehler** gekennzeichnet werden. Das Verhaltens konfigurieren Sie über den Konfigurationsparameter **Common | DBJobCreateWithFreeze**.
Fehler im Journal protokollieren	Ist die Option aktiviert, werden die Fehlermeldung aus der Prozessverarbeitung im Systemprotokoll aufgezeichnet. Fehlermeldung aus der Prozessverarbeitung können in der Prozesshistorie aufgezeichnet werden.
Protokollierungsmodus	Für Prozessschritte können Sie im Job Queue Info eine erweiterte Protokollierung der Meldungen aktivieren. Verwenden Sie den Protokollierungsmodus, um für einzelne Prozessschritte diese erweiterte Protokollierung immer zu aktivieren. Mit dem Wert **Immer** werden die Meldungen des Prozessschrittes im Erfolgsfall und im Fehlerfall protokolliert. Mit dem Wert **Fehler** werden die Meldungen des Prozessschrittes nur im Fehlerfall protokolliert.

Eigenschaft	Bedeutung
Prozesshistorie	Gibt an, ob die Meldung des Prozessschrittes wird in die Prozesshistorie geschrieben werden.
DBQueue wartet nicht	Gibt an, ob mit der Verarbeitung von DBQueue Prozessor Aufträgen gewartet werden muss, bis der Prozessschritt verarbeitet wurde. Ein Warten auf die Verarbeitung von Prozessschritten ist nur erforderlich, wenn der Prozessschritt potentiell Daten ändert, die für DBQueue Prozessor Aufträge relevant sind.

Verwandte Themen

- Gültigkeitsprüfung eines Prozesses auf Seite 269
- Festlegen des ausführenden Servers auf Seite 274
- Benachrichtigung zur Verarbeitung von Prozessschritten auf Seite 276
- Verwenden von prozesslokalen Variablen und globalen Variablen für die Prozessdefinition auf Seite 272
- Übersicht über die Prozesskomponenten auf Seite 281
- Aufzeichnen von Prozessinformationen in der Prozessverarbeitung auf Seite 352
- Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369
- Verwendung von Visual Basic .NET-Skripten auf Seite 375

Parameter für Prozessschritte

Mit der Auswahl einer Prozessfunktion im Prozessschritt legen Sie fest, welche Aktion durch den Prozessschritt ausgeführt wird. Die Parametervorlagen der Prozessfunktion werden kopiert und als Parameter in den Prozessschritt übernommen. Dadurch können Sie jedem Prozessschritt, der diese Prozessfunktion verwendet, andere Parameterwerte übergeben. Das Original wird nicht verändert.

Pflichtparameter werden bei der Auswahl der Prozessfunktion sofort in den Prozessschritt eingefügt, optionale Parameter fügen Sie einzeln zum Prozessschritt hinzu. Beim Einfügen eines Parameters wird die Wertvorlage aus der Parametervorlage übernommen. Die Vorlagen für die Werte der Parameter sind zum größten Teil vordefiniert, weil beispielsweise Prozeduren die UID’s von Objekten auswerten und entsprechend vermerken.

Detaillierte Informationen zum Thema

- Parameter für Prozessschritte bearbeiten auf Seite 261
- Eigenschaften der Parameter für Prozessschritte auf Seite 261
- Wertbelegung von Parametern auf Seite 262

Parameter für Prozessschritte bearbeiten

Um Parameter eines Prozessschrittes zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozessschritt.
4. Wählen Sie die Ansicht **Parameter**.

   Es werden alle Parameter angezeigt, die für den Prozess definiert sind.
5. Prüfen Sie, ob die benötigten Parameter zugewiesen sind und bearbeiten Sie die Parameter.

   Über die Symbolleiste können Sie Parameter einfügen, löschen oder bearbeiten.

   > TIPP: Durch Mausklick auf einen Eintrag können Sie den Parameterwert direkt bearbeiten.

Tabelle 76: Bedeutung der verwendeten Symbole

Symbol	Bedeutung
	Pflichtparameter der Prozessfunktion.
	Optionaler Parameter der Prozessfunktion, welcher dem Prozessschritt zugewiesen ist.
	Optionaler Parameter der Prozessfunktion, welcher dem Prozessschritt nicht zugewiesen ist.

Verwandte Themen

- Parameter für Prozessschritte bearbeiten auf Seite 261
- Wertbelegung von Parametern auf Seite 262

Eigenschaften der Parameter für Prozessschritte

Tabelle 77: Eigenschaften für Parameter

Eigenschaft	Bedeutung
Bezeichnung	Name des Parameters. HINWEIS: Der Name eines Parameters sollte nicht geändert werden. Ausnahme bilden die speziellen Parameter der Prozesskomponente `HandleObjectComponent`.
Versteckt	Gibt an, ob der Parameter in der Protokolldatei des One Identity Manager Service und im Programm Job Queue Info angezeigt wird. Werte für versteckte Parameter werden als `<HIDDEN>` angezeigt.

Eigenschaft	Bedeutung
	HINWEIS: Benutzer mit der Programmfunktion **Möglichkeit zum Anzeigen versteckter Parameter im Job Queue Info** (`JobQueue_ShowHiddenParameters`) können die versteckten Parameter im Job Queue Info einsehen. Weisen Sie die Programmfunktion der entsprechenden Berechtigungsgruppe zu.
Verschlüsselt	Gibt an, ob der Parameter verschlüsselt übergeben wird, wenn die Datenbank verschlüsselt ist. Verschlüsselte Parameter werden in der Protokolldatei des One Identity Manager Service und im Programm Job Queue Info als `<Hidden>` angezeigt. HINWEIS: Ist bereits in der Parametervorlage die Option **verschlüsselt** gesetzt, so ist der Parameter ebenfalls verschlüsselt zu übergeben.
Enthält verschlüsselte Bestandteile	Gibt an, ob in diesem Wert verschlüsselte Sequenzen enthalten sind. Verwenden Sie diese Option, wenn in komplexen Parametern, zum Beispiel Windows PowerShell Skripten, teilweise verschlüsselte Sequenzen zu übergeben sind, zum Beispiel Kennwörter. Verschlüsselte Bestandteile eines Parameters werden in der Protokolldatei des One Identity Manager Service und im Programm Job Queue Info als `<Hidden>` angezeigt.
Wertvorlage	Die Wertvorlagen definieren Sie in VB.Net-Syntax. Beim Einfügen eines Parameters wird die Wertvorlage aus der Parametervorlage übernommen. TIPP: Um die Standardwertvorlage wiederherzustellen, wählen Sie in der **Ansicht > Parameter** die Schaltfläche und klicken Sie in der Ansicht **Parameter bearbeiten** die Schaltfläche **Vorlage**.
Typ	Typ des Parameters. Es sind die Werte **IN**, **OUT** und **INOUT** zulässig. Parameter vom Typ **OUT** oder **INOUT** sind Parameter, in der eine Prozesskomponente einen Wert nach außen liefern kann. Dieser Wert steht dann allen nachfolgenden Prozessschritten des Prozesses zur Verfügung und kann als Wertbelegung für Parameter vom Typ **IN** dienen.

Verwandte Themen

- Wertbelegung von Parametern auf Seite 262
- Verwendung von Visual Basic .NET-Skripten auf Seite 375

Wertbelegung von Parametern

Die Wertvorlagen definieren Sie in VB.Net-Syntax. Für die Belegung der Werte können folgende Ausdrücke verwendet werden:

- leer
- Spalten eines Objektes oder Spalten eines über eine Beziehung verbundenen Objektes

  Syntax:

  ```
  Value = $<Spaltenname>:<Datentyp>$
  Value = ${FK(<Fremdschlüsselspalte>).}<Spaltenname>:<Datentyp>$
  ```

  Beispiel:

  ```
  Value = $Lastname$
  Value = $PasswordNeverExpires:bool$
  Value = $FK(Ident_Domain).Description$
  ```

- Parameter aus der optionalen Parametercollection

  Syntax:

  ```
  Value = $PC(<Parametername>)$
  ```

  Beispiel:

  ```
  Value = $PC(SRCUID_Application)$
  ```

- Out-Parameter

  Parameter vom Typ **OUT** oder **INOUT** sind Parameter, in der eine Prozesskomponente einen Wert nach außen liefern kann. Dieser Wert steht dann allen nachfolgenden Prozessschritten des Prozesses zur Verfügung und kann als Wertbelegung für Parameter vom Typ **IN** dienen.

  Bei der Verwendung von **OUT**-Parametern ist darauf zu achten, dass diese zur Laufzeit auch belegt sind. Anderenfalls wird bei Abarbeitung der Text `"&OUT(<parametername>)&"` eingetragen, das bedeutet die Variable wird nicht ersetzt.

  Syntax:

  ```
  Value = "&OUT(<Parametername>)&"
  ```

  Beispiel:

  ```
  Value = "&Out(FileSize)&"
  ```

- Globale Variablen, die das einstellende Programm besetzt

  Syntax:

  ```
  Value = Variables("<Variablenname>")
  ```

  Beispiel:

  ```
  Value = Variables("GENPROCID")
  Value = Variables("FULLSYNC")
  ```

- Per Prä-Skript erzeugte lokale Variablen des Prozessschrittes oder des Prozesses

  Syntax:

  ```
  Value = values("Name")
  ```

Beispiel:

```
Value = Values("FirstHomeServer")
```

- Abfrage von Konfigurationsparametern

  Hierbei muss immer der gesamte Pfad zum Konfigurationsparameter angegeben werden.

  Syntax:

  ```
  Value = Session.Config().GetConfigParm("<vollständiger Pfad>")
  ```

  Beispiel:

  ```
  Value = Session.Config().GetConfigParm("TargetSystem\ADS\PersonAutoDefault")
  ```

- VB.Net

  Es können beliebige Ausdrücke in VB.Net-Syntax angegeben werden.

- Abfrage von Umgebungsvariablen

  Syntax:

  ```
  &ENV(Variablenname)&
  ```

  Beispiel:

  ```
  Value = "&ENV(COMPUTERNAME)&"
  ```

- Abfrage von Secrets

  In der Konfiguration des One Identity Manager Service müssen die Parameter `SecretAllowList` und `SecretsFolder` konfiguriert sein.

  Syntax:

  ```
  &SECRET(Name)&
  ```

  Beispiel:

  ```
  Value = "&SECRET(API_KEY)&"
  ```

Verwandte Themen

- Eigenschaften der Parameter für Prozessschritte auf Seite 261
- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Konfiguration des One Identity Manager Service auf Seite 309

Ereignisse für Prozesse

Um Prozesse zu Objekten zuzuordnen, werden Ereignisse definiert. Erst nachdem die Verbindung zwischen Objekt, Ereignis und Prozess hergestellt wurde, können die Prozesse generiert und verarbeitet werden. Es werden bereits einige vordefinierte Ereignisse zur Verfügung gestellt. Diese sind in der nachfolgenden Tabelle aufgeführt.

Tabelle 78: Vordefinierte Ereignisse

Ereignis	Anmerkung
Insert	Das Ereignis wird beim Einfügen eines Objektes ausgelöst. Das Ereignis ist an jedem Objekt verfügbar.
Update	Das Ereignis wird beim Ändern eines Objektes ausgelöst. Das Ereignis ist an jedem Objekt verfügbar.
Delete	Das Ereignis wird beim Löschen eines Objektes ausgelöst. Das Ereignis ist an jedem Objekt verfügbar.
Execute	Das Ereignis wird vom DBQueue Prozessor ausgelöst, wenn der Ausführungszeitpunkt einer verzögerte Operation erreicht ist.
Assign	Das Ereignis wird beim Hinzufügen von M:N Zuweisungen ausgelöst.
Remove	Das Ereignis wird beim Entfernen von M:N Zuweisungen ausgelöst.

Weitere Ereignisse werden über die Customizer zur Verfügung gestellt.Sie können weitere kundenspezifische Ereignisse definieren, um Prozesse auszulösen.

Detaillierte Informationen zum Thema

- Ereignisse für Prozesse erstellen auf Seite 265
- Berechtigungen zum Auslösen von Prozessen auf Seite 266

Ereignisse für Prozesse erstellen

Wird das Objektereignis mit einer Programmfunktion versehen, dann können Benutzer, die diese Programmfunktion über ihre Berechtigungsgruppen besitzen, das Objektereignis und damit auch den Prozess auslösen, unabhängig von ihren Berechtigungen. Ausführliche Informationen zum Steuern von Berechtigungen zum Ausführen von Prozessen über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Um ein Ereignis zu erstellen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozess.
4. Wählen Sie die Ansicht **Ereignisse** und klicken Sie .
5. Erfassen Sie folgende Informationen.

Tabelle 79: Eigenschaften von Ereignissen

Eigenschaft	Beschreibung
Objektereignis	Bezeichnung des Objektereignisses. In der Auswahlliste **Objektereignis** werden die Objektereignisse der im Prozess festgelegten Tabelle angezeigt. 1. Wählen Sie ein bestehendes Objektereignis. - ODER - 2. Klicken Sie und erfassen Sie die Bezeichnung des neuen Objektereignisses.
Reihenfolge	Angabe, in welcher Reihenfolge die Prozesse generiert werden, wenn mehrere Prozesse auf das gleiche Ereignis des Basisobjektes weisen. Prozesse mit niedrigerer Reihenfolge werden vor Prozessen mit höherer Reihenfolge generiert.
Prozessinformation für Ereignis	VB.Net-Ausdruck zur Bildung des Anzeigenamens in der Prozessanzeige.

6. (Optional) Weisen Sie an das Objektereignis eine Programmfunktion zu.
   a. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung > Objektereignisse** das Ereignis.
   b. Wählen Sie den Menüeintrag **Ansicht > Tabellenrelationen wählen** und aktivieren Sie die Tabelle `QBMEventHasFeature`.
   c. Wählen Sie in der Bearbeitungsansicht die Ansicht **Programmfunktion** und wählen Sie die Programmfunktionen.

Verwandte Themen

- Aufzeichnen von Prozessinformationen in der Prozessverarbeitung auf Seite 352
- Berechtigungen zum Auslösen von Prozessen auf Seite 266

Berechtigungen zum Auslösen von Prozessen

Die grundlegende Berechtigung zum Auslösen von Prozessen erhält der angemeldete Benutzer über die Programmfunktion **Erlaubt das Auslösen von Ereignissen im Frontend** (`Common_TriggerEvents`).

Im One Identity Manager ist das Auslösen von Ereignissen an den hinterlegten Prozessen mit dem Berechtigungskonzept verbunden. Benutzer dürfen nur an solchen Objekten Ereignisse auslösen, für die Sie auch Bearbeitungsberechtigungen besitzen. Dies kann dazu führen, dass Benutzer an Tabellen, für die nur Sichtbarkeitsberechtigungen definiert sind, keine zusätzlichen Ereignisse für Prozesse auslösen können.

Für diesen Fall gibt es die Möglichkeit die Objektereignisse (Tabelle `QBMEvent`) mit einer Programmfunktion (Tabelle `QBMFeature`) zu verbinden. Ein Ereignis (Tabelle `JobEventGen`), welches für einen Prozess definiert wird, wird mit einem Objektereignis (Spalte `JobEventGen.UID_QBMEvent`) verknüpft. Das Objektereignis wird mit einer Programmfunktion (Tabelle `QBMEventHasFeature`) verbunden. Benutzer mit dieser Programmfunktion können, unabhängig von ihren Berechtigungen, das Objektereignis und damit auch den Prozess auslösen.

Ausführliche Informationen zum Steuern von Berechtigungen zum Ausführen von Prozessen über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Verwandte Themen

- Ereignisse für Prozesse erstellen auf Seite 265

Prozessgenerierung simulieren

Mit der Simulation der Prozessgenerierung können Sie überprüfen, ob der gewählte Prozess erfolgreich generiert werden kann oder die Syntax der Parameterübergabe korrekt ist. Somit können die Prozesse bei Bedarf ohne weiteren Aufwand weiter angepasst werden.

HINWEIS:

- Bei der Simulation eines Prozesses wird die Option **Nicht generieren** beachtet. Deaktivieren Sie die Option für die Simulation des Prozesses.
- Die generierten Assemblies werden lokal auf der Arbeitsstation, auf der die Simulation durchgeführt wird, abgelegt. Eine Simulation hat daher keine Auswirkung auf andere Benutzer.

Um die Generierung eines Prozesses zu testen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Starten Sie über den Menüeintrag **Prozess > Ansicht > Simulationsansicht** den Simulationsassistenten.
4. Auf der Startseite des Assistenten klicken Sie **Weiter**.
5. Auf der Seite **Ereignis wählen** wählen Sie das Ereignis, für welches der Prozess generiert werden soll und legen Sie die Datenbankverbindung für die Simulation fest. Wählen Sie **Designer Datenbank** oder **Hauptdatenbank**.
6. Auf der Seite **Objekt wählen** wählen Sie das Objekt, für welches das Ereignis simuliert werden soll.
7. (Optional) Auf der Seite **Objekteigenschaften ändern** ändern Sie die Eigenschaften des Objektes.

8. (Optional) Auf der Seite **Parametercollection definieren** erfassen Sie die Parameter für die Parametercollection. Folgende Aktionen können Sie ausführen:
   - **Prozessschritt laden**: Für den gewählten Prozess werden alle Elemente der Parametercollections aller Prozessschritte geladen.
   - **Einfügen**: Fügen Sie einzelne Parameter für die Simulation ein. Erfassen Sie den Parameternamen und den Parameterwert .
   - **Löschen**: Löschen Sie einzelne Parameter für die Simulation.

   HINWEIS: Für Prozesse, die mit Parametercollections generiert werden, müssen Sie die Parameter und die zu übergebenden Werte festlegen (beispielsweise Parameter `SourceDir` bei Kopien von Profilen oder Parameter `ConfigName` für das Einlesen eines Zielsystems). In Prozessen, die für die Standardereignisse (Insert, Update, Delete) generiert werden, werden keine Parametercollections verwendet.
9. (Optional) Auf der Seite **Connectionvariablen definieren** legen Sie die globalen Variablen des Session-Objektes fest, die für die Simulation verwendet werden. Klicken Sie **Einfügen** und geben Sie den Variablennamen und den Wert an.
10. (Optional) Auf der Seite **Präprozessoranweisungen definieren** wählen Sie die Präprozessorbedingungen, die bei der Generierung des Prozesses beachtet werden.
11. Um die Simulation zu starten, klicken Sie auf der letzten Seite des Assistenten **Fertig**.

    Der Simulationsvorgang kann einige Zeit in Anspruch nehmen. Nach Abschluss der Simulation wird der generierte Prozess im Prozesseditor angezeigt. Je nach Generierung werden die Prozessschritte farbig gekennzeichnet.

    Tabelle 80: Farbcode in der Simulation

Farbe	Bedeutung
hellgrau	Der Prozessschritt wurde nicht generiert.
blau	Der Prozessschritt wurde erfolgreich generiert.

    HINWEIS:

    Mit Maus-Doppelklick auf einen erfolgreich generierten Prozessschritt werden im Bearbeitungsfenster die Eigenschaften und Parameter mit den konkreten Werten dargestellt.

    Mit **Strg + C** können Sie die Parameterwerte in die Zwischenablage kopieren.

TIPP:

- Zur Nachbearbeitung der Prozesse können Sie über das Menü **Prozesse > Ansicht** zwischen der Bearbeitungsansicht und der Simulationsansicht wechseln.
- Für jeden Simulationsvorgang wird in der Symbolleiste des Prozesseditor ein Eintrag erzeugt, über den Sie die Simulation erneut ausführen können, ohne die Simulationsdaten nochmals festzulegen.

- Um das Protokoll zur Prozessgenerierung anzuzeigen, wählen Sie den Menüeintrag **Prozess > Ansicht > Prozessgeneratorlog**.

Gültigkeitsprüfung eines Prozesses

HINWEIS: Bevor Sie einen Prozess kompilieren, sollten Sie eine Gültigkeitsprüfung des Prozesses und der Prozessschritte durchführen.

Um einen Prozess zu prüfen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozess > Fehlerkontrolle**.

   Das Ergebnis der Gültigkeitsprüfung wird in der Ansicht **Prozessfehlerkontrolle** ausgegeben und bleibt bis zur erneuten Gültigkeitsprüfung erhalten.

Tabelle 81: Verwendete Symbole in der Gültigkeitsprüfung

Symbol	Bedeutung
	Es wurde kein Fehler gefunden.
	Fehler.
	Warnung, Information.

TIPP:

- Bei Warnungen und Informationen werden die Steuerelemente des Prozesses und der Prozessschritte gelb dargestellt. Bei Fehlern werden die Steuerelemente des Prozesses und der Prozessschritte rot dargestellt.
- Durch Maus-Doppelklick auf eine Fehlermeldung in der Ansicht **Prozessfehlerkontrolle** wird zum entsprechenden Eintrag im Prozess gesprungen.

Tabelle 82: Mögliche Fehlerursachen in Prozessen

Fehlerkategorie	Mögliche Ursachen
Fehler	Der Prozess hat keinen Namen. Es wurde kein Basisobjekttyp angegeben. Die Generierungsbedingung entspricht nicht der geforderten Notation (`Value =`).
Warnung	Der Prozess hat keinen Basisprozessschritt. Der Prozess hat kein Ereignis.
Information	Für den Prozess ist die Option **Nicht generieren** gesetzt.

Tabelle 83: Mögliche Fehlerursachen in Prozessschritten

Fehlerkategorie	Mögliche Ursachen
Fehler	Der Prozessschritt hat keinen Namen. Es wurde kein Prozessfunktion zugeordnet. Die Generierungsbedingung entspricht nicht der geforderten Notation (`Value =`). Es wurde kein Server zur Ausführung festgelegt (Skript zur Serverauswahl oder Servermaske). Der Name des Prozessschrittes ist nicht eindeutig. Der Prozessschritt hat keine Parameter. Der angegebene Parameterwert entspricht nicht der geforderten Notation (`Value =`). Am Prozessschritt sind die beiden Eigenschaften **Fehler ignorieren** und **Wartemodus bei Fehler** gesetzt. Der Prozessschritt wird nie wiederholt, da alle Fehler ignoriert werden. Am Prozessschritt sind die beiden Eigenschaften **Fehler ignorieren** und **Stopp bei Fehlern** gesetzt. Der Prozessschritt geht nicht in den Zustand **Frozen** da alle Fehler ignoriert werden. Am Prozessschritt sind die beiden Eigenschaften **Fehler ignorieren** und **Aufzweigung der Verarbeitung** gesetzt. Der Fehlerzweig wird damit niemals ausgeführt.
Warnung	Der Prozessschritt ist nicht in den Prozess eingebunden. Am Prozessschritt sind die beiden Eigenschaften **Aufzweigung der Verarbeitung** und **Wartemodus bei Fehler** gesetzt und keine Wiederholversuche definiert. Der Prozessschritt wird bei Fehlern sofort angehalten und es erfolgt damit keine Aufzweigung der Verarbeitung.

Verwandte Themen

- Kompilieren von Prozessen auf Seite 270

Kompilieren von Prozessen

Haben Sie einen neuen Prozess erstellt, einen Prozess importiert oder Änderungen an einem vorhandenen Prozess vorgenommen, müssen Sie den Prozess kompilieren. Erst durch die Kompilierung wird der Prozess generierbar.

> HINWEIS: Bevor Sie einen Prozess kompilieren, sollten Sie eine Gültigkeitsprüfung des Prozesses und der Prozessschritte durchführen.

Die Kompilierung erfolgt pro Basisobjekt, das bedeutet alle Prozesse eines Basisobjektes werden übersetzt. Bei der Kompilierung werden die Assemblies erstellt und lokal auf der Arbeitsstation, auf der die Generierung durchgeführt wird, abgelegt. Während der Übersetzung wird eine Gültigkeitskontrolle des Quellcodes durchgeführt. Der Vorgang kann daher etwas Zeit in Anspruch nehmen.

Im Prozesseditor sind zur Kompilierung eines Prozesses zwei Verfahren verfügbar.

Lokale Kompilierung

Dieses Verfahren können Sie nutzen, um die Kompilierung eines Prozesses zu testen.

Um einen Prozess lokal zu kompilieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozess > Kompilieren**.

Kompilierung mit Übernahme der Assemblies in die Hauptdatenbank

Haben Sie die Kompilierung eines Prozesses getestet, setzen Sie dieses Verfahren ein, um nach der Kompilierung eines Prozesses, die generierten Assemblies in die Hauptdatenbank zu übernehmen. Mit Übernahme der Änderungen stehen die geänderten Prozesse sofort im System zur Verfügung.

Um einen Prozess mit Übernahme der Assemblies in die Hauptdatenbank zu kompilieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Wählen Sie den Menüeintrag **Prozess > Kompilieren und in DB speichern**.

Fehler anzeigen

Fehlermeldungen während der Kompilierung werden in der Ansicht **Compilerfehler** ausgegeben. Treten während der Übersetzung Fehler auf, dann ist die Quellcodeansicht aktiviert. Diese Ansicht dient lediglich zur Darstellung des Quellcodes, eine Bearbeitung des Eintrages ist nicht möglich.

HINWEIS:

- Bearbeiten mehrere Benutzer gleichzeitig Prozesse eines Basisobjektes, so werden eventuelle Fehlermeldungen anderer Benutzer ebenfalls ausgegeben. Diese können jedoch vom aktuellen Benutzer nicht bearbeitet werden.
- Durch Maus-Doppelklick auf eine Fehlermeldung in der Ansicht **Compilerfehler** wird im Prozess zum entsprechenden Eintrag gesprungen. Diesen können Sie hier bearbeiten.
- Durch Maus-Doppelklick auf die Fehlermeldung in der Ansicht **Compilerfehler** wird bei angeschalteter Quellcodeansicht zur entsprechenden Zeile gesprungen.

Verwandte Themen

- Gültigkeitsprüfung eines Prozesses auf Seite 269

Verwenden von prozesslokalen Variablen und globalen Variablen für die Prozessdefinition

Prozesslokale Variablen sind lokale Datenspeicher bei der Generierung eines Prozesses. Sie dienen zur einmaligen Bestimmung von Werten innerhalb eines Prä-Skriptes, die dann innerhalb der Prozesse und ihrer Prozessschritte weiterverwendet werden können, beispielsweise in Generierungsbedingungen, Serverauswahlskripten oder in den Parametern.

> HINWEIS: Es wird empfohlen, die prozesslokalen Variablen nur im Prä-Skript zu besetzen und bei der weiteren Verwendung lesend darauf zuzugreifen.

Syntax im Prä-Skript

```
values("Name") = "Wert"
```

Verwendung in den Codeteilen des Prozesses und seiner Prozessschritte

```
Value = values("Name")
```

Zur Steuerung der Prozessgenerierung können zusätzlich globale Variablen verwendet werden, die über das `Session`-Objekt zur Verfügung gestellt werden. Diese Variablen sind solange gültig wie die Session besteht. Zusätzlich zu den vordefinierten Variablen können alle am `Session`-Objekt definierten kundenspezifischen Variablen verwendet werden. Kundenspezifische globale Variablen können beispielsweise über Skripte, Methoden oder Customizer definiert und in den Prozessen verwendet werden.

> HINWEIS:
>
> - Globale Variablen sollten in den Prozessen nur lesend verwendet werden.
> - Bei der Verarbeitung eines Prozesses wird zuerst das Prä-Skript zur Generierung ausgeführt und anschließend die Generierungsbedingung ausgewertet. Es wird empfohlen, globale Variablen, die in der Generierungsbedingung verwendet werden sollen, ebenfalls im Prä-Skript auszuwerten. Dadurch können unnötige Datenzugriffe verhindert werden.
>
> Wird eine kundenspezifische `Session`-Variable definiert, dann sollte sie anschließend wieder entfernt werden. Andernfalls bleibt sie bei Nachnutzung der Session bestehen und es werden unter Umständen falsche Prozesse generiert.

Beispiel:

Ein Prozess soll nur im Fall einer Vollsynchronisation generiert werden. Dazu wird die `Session`-Variable **FullSync** verwendet. Diese Variable kann die Werte **True** und **False** annehmen. Die Variable ist für alle Prozesse verfügbar, die innerhalb der Vollsynchronisation generiert werden.

Die Variable wird im Prä-Skript zur Generierung und in der Generierungsbedingung abgefragt. Dadurch wird bereits bei der Ausführung des Prä-Skriptes verhindert, das weitere nicht benötigte Objekte geladen werden.

Prä-Skript zur Generierung:

```
If CBool(Session.Variables("FULLSYNC")) Then
     values("Name1") = "Wert1"
     values("Name2") = "Wert2"
     ...
End If
```

Generierungsbedingung:

```
Value = CBool(Session.Variables("FULLSYNC"))
```

Verwandte Themen

- Prä-Skripte zur Verwendung in Prozessen und Prozessschritten auf Seite 385
- Abfrage von globalen Variablen des Session-Objektes auf Seite 388

Schwellwerte für die Verarbeitung von Prozessen

Zur Verhinderung von Massenänderungen können Sie pro Prozess festlegen, wie oft dieser Prozess in der Jobqueue vorhanden sein darf.

Voraussetzungen

- Bei Überschreitung des Schwellwertes für die Warnung wird eine E-Mail Benachrichtung an einen bestimmten Empfänger versendet. Voraussetzung für die Nutzung des Benachrichtigungssystems ist die Einrichtung eines SMTP Host für den Mailversand sowie die Aktivierung der Konfigurationsparameter für die Mailbenachrichtigung. Ausführliche Informationen zur Konfiguration der E-Mail Benachrichtigung finden Sie im *One Identity Manager Installationshandbuch*.
- Prüfen Sie im Designer den Konfigurationsparameter **Common | MailNotification | NotifyAboutWaitingJobs** und aktivieren Sie diesen bei Bedarf. Ist der

Konfigurationsparameter aktiviert, wird bei Auftreten von Prozessen mit dem Status **Overlimit** eine E-Mail Benachrichtigung versendet und ein entsprechender Eintrag im Ereignisprotokoll des Aktualisierungsservers erzeugt.

Um Schwellwerte zu definieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess aus.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozess.
4. Bearbeiten Sie in der Ansicht **Prozesseigenschaften** auf dem Tabreiter **Allgemein** die folgenden Informationen.
   - **Schwellwert (Warnung)**: Erfassen Sie die maximale Anzahl dieser Prozesse für eine Queue, die gleichzeitig vorhanden sein dürfen. Wird diese Anzahl überschritten, wird eine Warnung versendet. Der One Identity Manager Service verarbeitet die Prozesse jedoch weiter.
   - **Schwellwert (Sperrung)**: Erfassen Sie die maximale Anzahl dieser Prozesse für eine Queue, die gleichzeitig vorhanden sein dürfen. Bei Überschreitung des Schwellwertes für die Sperrung werden die betroffenen Prozesse in der Jobqueue auf den Status **Overlimit** gesetzt. Diese Prozesse werden nicht mehr durch den One Identity Manager Service zur Verarbeitung abgeholt und verbleiben in der Jobqueue.

     Im Job Queue Info können Sie diese Prozesse reaktivieren. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.

     > TIPP: Für die initiale Befüllung des Schwellwertes für die Sperrung können Sie das Datenbankskript `SDK_SetLimitationCount_in_Jobchain` verwenden. Dieses Datenbankskript finden Sie auf dem Installationsmedium im Verzeichnis `QBM\dvd\AddOn\SDK\SQLSamples`.

Verwandte Themen

- Eigenschaften für Prozesse auf Seite 254

Festlegen des ausführenden Servers

Für jeden Prozessschritt legen Sie fest, welcher Server den Prozessschritt verarbeiten soll. Die Auswahl des ausführenden Servers können Sie über die Serverfunktion oder ein Auswahlskript vornehmen. Die Auswahl des Servers muss immer zu einem eindeutigen Ergebnis führen. Bei der Ermittlung des gültigen Servers wird zuerst das Auswahlskript ausgewertet. Kann über das Auswahlskript kein Server bestimmt werden, wird die Serverfunktion ausgewertet. Der Prozessschritt wird für den ersten gefundenen Server zur Verarbeitung eingestellt.

Detaillierte Informationen zum Thema

- Server über eine Serverfunktion auswählen auf Seite 275
- Server über ein Auswahlskript auswählen auf Seite 275

Server über eine Serverfunktion auswählen

Es sind die gängigen Serverfunktionen, wie beispielsweise Domänen-Controller oder SQL Ausführungsserver definiert. Nutzen Sie die direkte Angabe einer Serverfunktion, wenn Sie den Server eindeutig bestimmen können.

Um einen Server über eine Serverfunktion festzulegen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozessschritt.
4. Wählen Sie in der Ansicht **Prozessschritteigenschaften** auf dem Tabreiter **Generierung** in der Auswahlliste **Serverfunktion** die Serverfunktion.

Verwandte Themen

- Überblick über die Serverfunktionen auf Seite 295
- Eigenschaften für Prozessschritte auf Seite 256

Server über ein Auswahlskript auswählen

Ist es anhand der Serverfunktion nicht möglich zu entscheiden, welcher Server genutzt werden soll, beispielsweise bei Vorhandensein mehrerer SMTP Server, verwenden Sie ein Auswahlskript zur näheren Bestimmung des Servers.

Zur Ermittlung des Servers über ein Auswahlskript verwenden Sie VB.Net-Ausdrücke:

- die eine Zeichenkette mit der UID des Jobservers zurückgeben
- die eine Zeichenkette mit der Angabe einer Where-Klausel für Datenbankabfragen liefern. Die Auswahl muss eine Zeichenkette liefern, die mit `where` beginnt und eine logische Bedingung enthält. Die Where-Klausel wird auf die Tabelle `QBMServer` angewendet.

Alternativ können Sie im Auswahlskript die Queue, welche den Prozessschritt abarbeiten soll, direkt angeben. Jeder One Identity Manager Service innerhalb des gesamten Netzwerkes hat einen eindeutigen Queuenamen. Mit exakt diesem Queuenamen werden die Prozessschritte an der Jobqueue angefordert.

Syntax für die direkte Angabe der Queue

```
DIRECT:<Queue>
```

Beispiel:

```
Value = "DIRECT:\Server01"
```

Um einen Server über ein Auswahlskript festzulegen

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozessschritt.
4. Erfassen Sie in der Ansicht **Prozessschritteigenschaften** auf dem Tabreiter **Generierung** in der Eigenschaft **Skript zur Serverauswahl** das Auswahlskript.

Verwandte Themen

- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Server über eine Serverfunktion auswählen auf Seite 275
- Eigenschaften für Prozessschritte auf Seite 256

Benachrichtigung zur Verarbeitung von Prozessschritten

Sie haben die Möglichkeit für einen Prozessschritt E-Mail Benachrichtigungen für den Erfolgsfall und Fehlerfall der Prozessverarbeitung zu versenden. Voraussetzung für die Nutzung des Benachrichtigungssystems ist die Einrichtung eines SMTP-Host für den Mailversand sowie die Aktivierung der Konfigurationsparameter für die E-Mail Benachrichtigung. Nutzen Sie bei der Einrichtung der Benachrichtigungen die verschiedenen Konfigurationsparameter für die Mailbenachrichtigung. Ausführliche Informationen zur Konfiguration der E-Mail Benachrichtigung finden Sie im *One Identity Manager Installationshandbuch*.

Um E-Mail Benachrichtigungen für einen Prozessschritt zu konfigurieren

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozessschritt.
4. Wählen Sie die Ansicht **Prozessschritteigenschaften**.
5. Auf dem Tabreiter **Allgemein** aktivieren Sie die Optionen **Benachrichtigung (Erfolg)** und **Benachrichtigung (Fehler)**.
6. Auf den Tabreitern **Benachrichtigung bei Erfolg** und **Benachrichtigung bei Fehler** erfassen Sie die Daten zum Versenden von Benachrichtigungen.

   HINWEIS: Alle Angaben müssen Sie in VB.Net-Syntax eintragen. Zur sprachabhängigen Bildung der Informationen verwenden Sie die #LD-Notation.

Tabelle 84: Eigenschaften für Benachrichtigungen

Eigenschaft	Bedeutung
Absender E-Mail-Adresse	E-Mail-Adresse des Absenders der Benachrichtigung.
Empfänger E-Mail-Adresse	E-Mail-Adresse des Empfängers der Benachrichtigung.
Betreff	Betreffzeile.
Meldung	Die zu übermittelnde Nachricht.

HINWEIS: E-Mail Benachrichtigungen während der Verarbeitung werden nur versendet, wenn alle Angaben für einen Fall (Fehler oder Erfolg) hinterlegt sind!

Beispiel für die Konfiguration der E-Mail Benachrichtigung

Absender E-Mail-Adresse	`Value = Connection.GetConfigParm ("Common\MailNotification\DefaultSender")`
Empfänger E-Mail-Adresse	`Value = Connection.GetConfigParm ("Common\MailNotification\DefaultAddress")`
Betreff	`Value = #LD("Error updating the Active Directory user account {0}.", $CanonicalName$)#`
Meldung	`Value = #LD("The user account {0} could not be updated.)#`

Zum Versenden von E-Mail Benachrichtigungen aus der Prozessverarbeitung wird der Prozess `VID_SendMail` (Tabelle `DialogDatabase`) genutzt. Dieser Prozess nutzt die Parameter der Datenbankprozedur `vid_InsertForSendMail`. Um diesen Prozess kundenspezifisch anzupassen, erstellen Sie eine Kopie des Prozesses und passen den Prozess weiter an.

TIPP:

Um im Fehlerfall die vom One Identity Manager Service protokollierten Fehlermeldungen per E-Mail Benachrichtigung zu versenden, stellt die Datenbankprozedur `vid_InsertForSendMail` den Parameter `pcAdditionalMessage` zur Verfügung.

Verwenden Sie bei der Einrichtung der Benachrichtigungen im Fehlerfall in der Meldung die Variable `[AdditionalMessage]` um auf diese Funktionalität zuzugreifen.

Beispiel für eine Meldung:

```
Value = "Prozess fehlgeschlagen." & vbcrlf _
     & vbcrlf _
     & "-----------------------------------------------------------------------
     " & vbcrlf _
     & "[AdditionalMessage]"
```

Verwandte Themen

- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Verwendung der #LD-Notation auf Seite 389
- Eigenschaften für Prozessschritte auf Seite 256

Automatisierte Ausführung von Prozessen

Um die zyklische Ausführung von Prozessen, wie beispielsweise die regelmäßige Synchronisation mit einer Zielsystem-Umgebung zu realisieren, richten Sie Prozessaufträge ein. Prozessaufträge sind mit Zeitplänen verbunden und werden somit in regelmäßigen Abständen ausgeführt.

Für die automatisierte Ausführung von Prozessen sind die folgenden Schritte erforderlich:

1. Erstellen eines Prozessauftrags

   Ein Prozessauftrag umfasst die Grundkonfiguration für die automatisierte Ausführung eines Prozesses.

2. Einrichten und Konfiguration eines Zeitplans

   Ein Zeitplan umfasst die Konfiguration der Ausführungszeiten für einen regelmäßig auszuführenden Prozess. Ausführliche Informationen zu Zeitplänen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Detaillierte Informationen zum Thema

- Status von Prozessaufträgen anzeigen auf Seite 278
- Prozessauftrag sofort starten auf Seite 279
- Prozessaufträge erstellen und bearbeiten auf Seite 279
- Eigenschaften eines Prozessauftrags auf Seite 280

Status von Prozessaufträgen anzeigen

Um den Status der Prozessaufträge anzuzeigen

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozessautomatisierung**.
2. Starten Sie den Editor über die Aufgabe **Prozessaufträge bearbeiten**.

   In der Listenansicht des Editors für Prozessaufträge werden alle Prozessaufträge mit ihrem Status dargestellt.

Tabelle 85: Bedeutung der Symbole in der Listenansicht

Symbol	Bedeutung
	Der Zeitplan des Prozessauftrags ist nicht aktiviert.
	Der Prozessauftrag wurde planmäßig ausgeführt.
	Der Prozessauftrag wurde nicht ausgeführt. Dieser Zustand kann auftreten, wenn die planmäßige Ausführung nicht möglich war oder wenn der Zeitplan neu aktiviert wurde und der Zeitpunkt der erstmaligen Ausführung noch nicht erreicht wurde.
	Die aktuelle Zeit (Serverzeit) liegt nicht zwischen Startdatum und Enddatum des verwendeten Zeitplans.

TIPP: Um die Startzeiten aller Prozessaufträge neu zu laden, verwenden Sie den Menüeintrag **Prozessauftrag > Aktualisieren**.

Verwandte Themen

- Prozessauftrag sofort starten auf Seite 279
- Prozessaufträge erstellen und bearbeiten auf Seite 279

Prozessauftrag sofort starten

Prozessaufträge sind mit Zeitplänen verbunden und werden somit in regelmäßigen Abständen ausgeführt. Bei Bedarf können Sie einen Prozessauftrag sofort ausführen.

Um einen Prozessauftrag sofort zu starten

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozessautomatisierung**.
2. Starten Sie den Editor über die Aufgabe **Prozessaufträge bearbeiten**.
3. Wählen Sie den Prozessauftrag und wählen Sie das Kontextmenü **Ausführen**.

Verwandte Themen

- Status von Prozessaufträgen anzeigen auf Seite 278
- Prozessaufträge erstellen und bearbeiten auf Seite 279

Prozessaufträge erstellen und bearbeiten

Ein Prozessauftrag umfasst die Grundkonfiguration für die automatisierte Ausführung eines Prozesses.

Um einen Prozessauftrag zu erstellen oder bearbeiten

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozessautomatisierung**.
2. Starten Sie den Editor über die Aufgabe **Prozessaufträge bearbeiten**.
3. Erstellen Sie einen neuen Prozessauftrag über den Menüeintrag **Prozessauftrag > Neu**.

   - ODER-

   Wählen Sie einen vorhanden Prozessauftrag.
4. Bearbeiten Sie in der Ansicht **Prozessauftrag konfigurieren** die Stammdaten des Prozessauftrags.

TIPP: Welcher Prozess durch einen Prozessauftrag ausgelöst wird, sehen Sie über das Kontextmenü **Prozess bearbeiten**.

Verwandte Themen

- Status von Prozessaufträgen anzeigen auf Seite 278
- Prozessauftrag sofort starten auf Seite 279
- Eigenschaften eines Prozessauftrags auf Seite 280

Eigenschaften eines Prozessauftrags

Tabelle 86: Eigenschaften eines Prozessauftrags

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung des Prozessauftrages. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Tabelle	Basisobjekt (Tabelle), für welches der Prozessauftrag ausgeführt wird.
Ereignis	Ereignis, welches ausgeführt werden soll. Für neue Prozessaufträge werden alle Ereignisse des Basisobjektes angeboten.
Zeitplan der Aktivierung	Zeitplan, der die Ausführungszeiten für den Prozessauftrag enthält. HINWEIS: Über die Schaltfläche neben der Auswahlliste erstellen Sie einen neuen Zeitplan. Ausführliche Informationen zu Zeitplänen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.
max. Laufzeit	Anzahl der Stunden, nach denen ein Prozessauftrag automatisch beendet werden soll.
Beschreibung	Nähere Beschreibung des Prozessauftrags.

Eigenschaft	Bedeutung
Bedingung	Einschränkende Bedingung für die Elemente, auf die dieser zeitgesteuerte Auftrag anzuwenden ist. Die Angabe muss der Where-Klausel-Syntax von Datenbankabfragen genügen.
Parameter	Liste von Parametern einer Parametercollection, die bei Generierung des Prozesses, der über diesen Prozessauftrag ausgelöst wird, zu besetzen sind.

Verwandte Themen

- Status von Prozessaufträgen anzeigen auf Seite 278
- Prozessauftrag sofort starten auf Seite 279
- Prozessaufträge erstellen und bearbeiten auf Seite 279

Übersicht über die Prozesskomponenten

Die Prozesskomponenten und ihre Prozessfunktionen bilden das Vorgabegerüst, auf dem alle Prozessschritte basieren. Die Tabellen `Jobcomponent`, `JobTask` und `Jobparameter` definieren den Gesamtumfang der zum One Identity Manager gehörenden Prozesskomponenten und deren Prozessfunktionen mit ihren zugehörigen Parametern.

Um die einzelnen elementaren Arbeiten auf Systemebene zu verrichten, wie beispielsweise das Anlegen von Verzeichnissen, werden die Prozessfunktionen verwendet. Ein oder mehrere Prozessfunktionen und deren Parameter sind in Prozesskomponenten zusammengefasst.

Beim Erstellen eines Prozesses werden die Parametervorlagen der Prozessfunktion kopiert und als Parameter in den Prozessschritt übernommen. Dadurch können Sie jedem Prozessschritt, der diese Prozessfunktion verwendet, andere Parameterwerte übergeben. Das Original wird nicht verändert.

> HINWEIS: Die zur Verfügung stehenden Informationen zu den Prozesskomponenten und ihren Prozessfunktionen werden durch die Schemainstallation eingepflegt und sind bis auch wenige Eigenschaften nicht bearbeitbar.

Um eine vollständige Übersicht über die Prozesskomponenten mit ihren Prozessfunktionen und Parametern zu erhalten

- Wählen Sie im Designer die Kategorie **Dokumentation > Berichte zur Systemkonfiguration** den Bericht **Prozesskomponenten**.

Um einzelnen Prozesskomponenten mit ihren Prozessfunktionen und Parametern anzuzeigen

- Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozesskomponenten**.

Die nachfolgende Tabelle enthält die Kurzbeschreibungen der Prozesskomponenten.

HINWEIS: Abhängig von den installierten Modulen können weitere Prozesskomponenten verfügbar sein.

Tabelle 87: Kurzbeschreibung der Prozesskomponenten

Komponente	Beschreibung
AutoUpdateComponent	Die Prozesskomponente bildet die Built-In-Tasks des One Identity Manager Service ab.
CommandComponent	Die Prozesskomponente führt ein beliebiges Kommando aus.
DelayComponent	Die Prozesskomponente steuert die Startzeit des nachfolgenden Prozessschrittes.
FileComponent	Die Prozesskomponente erzeugt, löscht, kopiert und modifiziert Dateien und Verzeichnisse sowie deren Zugriffsberechtigungen. Als Voraussetzung für die Nutzung der Prozesskomponente unter Linux-Betriebssystemen wird das Programm RSync benötigt. Unter Windows benötigen einige der Prozessfunktionen der Prozesskomponente für das Bearbeiten von Berechtigungen das Programm XCacls. Dieses finden Sie im Ressourcekit Ihrer Serverinstallation.
FtpComponent	Die Prozesskomponente kann Dateien per FTP übertragen.
HandleObjectComponent	Die Prozesskomponente führt Standardereignisse und benutzerdefinierte Ereignisse für Datenbankobjekte aus. Dabei werden analog zu den Frontends die jeweils zugeordneten Standardprozessen generiert. Weiterhin stellt die Komponente die Möglichkeit zur Verfügung, benutzerdefinierte Ereignisse für das Anstoßen der Generierung eines speziellen Prozesses objektbezogen auszulösen.
LogComponent	Die Prozesskomponente dient zum Protokollieren von Nachrichten, beispielsweise ins Ereignisprotokoll.
MailComponent	Die Prozesskomponente kann E-Mails versenden.
PowerShellComponent	Die Prozesskomponente dient zum Aufruf der Windows PowerShell. Windows PowerShell muss in der Version 2.0 installiert sein.
PowershellComponentNet4	Die Prozesskomponente dient zum Aufruf einer .NET 4 Windows PowerShell. Windows PowerShell muss mit einer Version größer 2.0 installiert sein.
ProjectorComponent	Die Prozesskomponente enthält Methoden zur Synchro-

Komponente	Beschreibung
	nisation und Provisionierung von Daten mit der One Identity Manager-Datenbank.
ReportComponent	Die Prozesskomponente kann Berichte erstellen und dann in eine externe Datei unterschiedlichen Typs exportieren.
ScriptComponent	Die Prozesskomponente führt Skripte aus Assemblies aus.
SFtpComponent	Die Prozesskomponente kann Dateien per SFTP übertragen.
SQLComponent	Die Prozesskomponente führt SQL Abfragen aus und kann zum Ermitteln von Datensatzanzahl und Datensatzexistenz verwendet werden.
ZipComponent	Die Prozesskomponente erstellt oder entpackt ZIP-Dateien.

Detaillierte Informationen zum Thema

- Ausführungstyp von Prozessfunktionen anzeigen und ändern auf Seite 283
- Anzahl der maximalen Instanzen für Prozessfunktionen und Prozesskomponenten ändern auf Seite 284
- Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen auf Seite 286

Ausführungstyp von Prozessfunktionen anzeigen und ändern

Über den Ausführungstyp einer Prozessfunktion ist definiert, ob die Verarbeitung intern im One Identity Manager Service oder in einem externen Prozess erfolgen soll.

Um den Ausführungstyp der Prozessfunktionen einer Prozesskomponente anzuzeigen

- Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozesskomponenten > <Prozesskomponente>**.

  Auf dem Formular Überblick zu Prozesskomponente werden alle Prozessfunktionen mit ihren Ausführungstypen angezeigt.

Um den Ausführungstyp einer Prozessfunktion zu ändern

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozesskomponenten**.
2. Wählen Sie im Prozesskomponenteneditor die Prozesskomponente.
3. Wählen Sie die Prozessfunktion und passen Sie in der Bearbeitungsansicht auf dem Tabreiter **Eigenschaften** den Wert für den Ausführungstyp an.

   Zulässige Werte sind:

   - **Internal**: Interne Ausführung im One Identity Manager Service.
   - **External**: Externe Ausführung als eigener Prozess.
   - **External32**: Externe Ausführung als eigener 32-bit Prozess.
4. Speichern Sie die Änderungen.

Verwandte Themen

- Übersicht über die Prozesskomponenten auf Seite 281
- Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen auf Seite 286
- Arbeitsweise des One Identity Manager Service auf Seite 305
- Ausführung externer Prozesse durch den StdioProcessor auf Seite 308

Anzahl der maximalen Instanzen für Prozessfunktionen und Prozesskomponenten ändern

Über die Anzahl der maximalen Instanzen für Prozessfunktionen und Prozesskomponenten kann die Parallelisierung der Prozessverarbeitung durch den One Identity Manager Service erreicht werden.

Um die Anzahl der maximalen Instanzen einer Prozessfunktion zu ändern

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozesskomponenten**.
2. Wählen Sie im Prozesskomponenteneditor die Prozesskomponente.
3. Wählen Sie die Prozessfunktion und passen Sie in der Bearbeitungsansicht auf dem Tabreiter **Eigenschaften** im Eingabefeld **Max.Instanzen** den Wert für den die maximalen Instanzen an.

   Zulässige Werte sind:

   - **-1**: Alle Instanzen dieser Prozessfunktion werden nacheinander abgearbeitet. Weitere Instanzen von Prozessfunktionen derselben Prozesskomponente

werden zeitgleich nicht ausgeführt.

- **0**: Es wird die an der Prozesskomponente angegebene Anzahl der maximalen Instanzen genutzt.
- **1** oder größer: Genaue Anzahl der Instanzen einer Prozessfunktion, die gleichzeitig abgearbeitet werden.

4. Speichern Sie die Änderungen.

Um die Anzahl der maximalen Instanzen einer Prozesskomponente zu ändern

1. Wählen Sie im Designer die Kategorie **Prozess-Orchestrierung > Prozesskomponenten**.
2. Wählen Sie im Prozesskomponenteneditor die Prozesskomponente und passen Sie in der Bearbeitungsansicht auf dem Tabreiter **Eigenschaften** im Eingabefeld **Max.Instanzen** den Wert für den die maximalen Instanzen an.

   Zulässige Werte sind:

   - **-1**: Alle Instanzen dieser Prozesskomponente werden nacheinander abgearbeitet.

     Für diese Komponenten muss sichergestellt sein, dass sie exklusiv auf einem Jobserver ausgeführt werden, das heißt es dürfen nicht mehrere Queues vorhanden sein, die Prozesse dieser Komponente verarbeiten.
   - **0**: Alle Instanzen dieser Prozesskomponente können gleichzeitig abgearbeitet werden.
   - **1** oder größer: Genaue Anzahl der Instanzen einer Prozesskomponente, die gleichzeitig abgearbeitet werden.

   HINWEIS: Der Wert wird nur genutzt, wenn die maximale Anzahl der Instanzen einer Prozessfunktion auf **0** gesetzt ist. Andernfalls gilt der an der Prozessfunktion gesetzte Wert.
3. Speichern Sie die Änderungen.

Verwandte Themen

- Übersicht über die Prozesskomponenten auf Seite 281
- Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen auf Seite 286
- Arbeitsweise des One Identity Manager Service auf Seite 305
- Parallele Verarbeitung von Prozessen durch den One Identity Manager Service auf Seite 307

Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen

Tabelle 88: Eigenschaften einer Prozesskomponente

Eigenschaft	Bedeutung
Anzeigename	Bezeichnung der Komponente zur Anzeige.
Klasse der Komponente	Klasse der Komponente.
Name des Assemblies	Name der Komponente.
Beschreibung	Beschreibung zur Funktion der Prozesskomponente.
Bemerkungen	Zusätzliche Bemerkungen zur Prozesskomponente.
Max. Instanzen	Der Wert legt die maximale Anzahl der Instanzen fest, in der diese Prozesskomponente in einer Queue auf dem Jobserver laufen darf. Zulässige Werte sind: • **-1**: Alle Instanzen dieser Prozesskomponente werden nacheinander abgearbeitet. Für diese Komponenten muss sichergestellt sein, dass sie exklusiv auf einem Jobserver ausgeführt werden, das heißt es dürfen nicht mehrere Queues vorhanden sein, die Prozesse dieser Komponente verarbeiten. • **0**: Alle Instanzen dieser Prozesskomponente können gleichzeitig abgearbeitet werden. • **1** oder größer: Genaue Anzahl der Instanzen einer Prozesskomponente, die gleichzeitig abgearbeitet werden. HINWEIS: Der Wert wird nur genutzt, wenn die maximale Anzahl der Instanzen einer Prozessfunktion auf **0** gesetzt ist. Andernfalls gilt der an der Prozessfunktion gesetzte Wert.
Konfiguration	Definition möglicher zusätzlicher Optionen für die Komponente in XML-Syntax.

Tabelle 89: Eigenschaften einer Prozessfunktion

Eigenschaft	Bedeutung
Bezeichnung	Name der Prozessfunktion.
Betriebssystemklasse	Diese Angabe bestimmt, auf welchem Betriebssystem diese Prozessfunktion ausgeführt werden kann. Zulässig sind die Werte **Win32**, **Linux** und **ALL**, wobei der Wert **ALL** für die

Eigenschaft	Bedeutung
	Ausführbarkeit dieser Prozessfunktion auf jedem Betriebssystem verwendet wird.
Ausführungstyp	Ausführungstyp für die Prozessfunktion. Zulässige Werte sind: • **Internal**: Interne Ausführung im One Identity Manager Service. • **External**: Externe Ausführung als eigener Prozess. • **External32**: Externe Ausführung als eigener 32-bit Prozess.
Beschreibung	Beschreibung der Prozessfunktion.
Max. Instanzen	Der Wert legt die maximale Anzahl der Instanzen, die vom One Identity Manager Service parallel pro Prozessfunktion ausgeführt werden können. Zulässige Werte sind: • **-1**: Alle Instanzen dieser Prozessfunktion werden nacheinander abgearbeitet. Weitere Instanzen von Prozessfunktionen derselben Prozesskomponente werden zeitgleich nicht ausgeführt. • **0**: Es wird die an der Prozesskomponente angegebene Anzahl der maximalen Instanzen genutzt. • **1** oder größer: Genaue Anzahl der Instanzen einer Prozessfunktion, die gleichzeitig abgearbeitet werden.
Letzter Schritt eines Prozessteilbaumes	Gibt an, ob eine Prozessfunktion prinzipiell das Ende eines Prozessteilbaums markiert.
Komponente	Prozesskomponente, zu der die Prozessfunktion gehört.
Direkte Datenbankverbindung erforderlich	Gibt an, ob die Prozessfunktion eine direkte Datenbankverbindung benötigt.
Exklusiv pro Objekt	Gibt an, ob die Ausführung der Prozessfunktion exklusiv pro Objekt erfolgen soll. Ist die Option für einen Prozessfunktion aktiviert, darf für ein bestimmtes Objekt immer nur ein Prozessschritt mit dieser Prozessfunktion ausgeführt werden. Es erfolgt keine Parallelverarbeitung.
DBQueue wartet nicht	Gibt an, ob mit der Verarbeitung von DBQueue Prozessor Aufträgen gewartet werden muss, bis der Prozessschritt verarbeitet wurde. Ein Warten auf die Verarbeitung von Prozessschritten ist nur erforderlich, wenn der Prozessschritt potentiell Daten ändert, die für DBQueue Prozessor Aufträge relevant sind.

Tabelle 90: Eigenschaften eines Parametervorlagen

Eigenschaft	Bedeutung
Bezeichnung	Name des Parameters.
Wertvorlage	Standardvorlage zur Wertermittlung. Beim Einfügen eines Parameters in einen Prozessschritt wird die Wertvorlage aus der Parametervorlage übernommen. Die Wertvorlagen definieren Sie in VB.Net-Syntax.
Bildungsregel (Beispiel)	Beispiel für die Wertvorlage.
Beschreibung	Beschreibung des Parameters.
Typ	Hierbei sind die Belegungen **IN**, **OUT** und **INOUT** zulässig.
Optional	Kennzeichnung des Parameters als Pflichtparameter oder optionaler Parameter.
Versteckt	Gibt an, ob der Parameter in der Protokolldatei des One Identity Manager Service und im Programm Job Queue Info angezeigt wird. Werte für versteckte Parameter werden als `<HIDDEN>` angezeigt. HINWEIS: Folgende Benutzer können die versteckten Parameter im Job Queue Info einsehen • administrative Systembenutzer • Benutzer mit der Programmfunktion **Möglichkeit zum Anzeigen versteckter Parameter im Job Queue Info** (`JobQueue_ShowHiddenParameters`)
Verschlüsselt	Gibt an, ob der Parameter verschlüsselt übergeben wird.
Enthält verschlüsselte Bestandteile	Gibt an, ob in diesem Wert verschlüsselte Sequenzen enthalten sind.
Prozessfunktion	Prozessfunktion, zu der der Parameter gehört.

Verwandte Themen

- Anzahl der maximalen Instanzen für Prozessfunktionen und Prozesskomponenten ändern auf Seite 284
- Ausführungstyp von Prozessfunktionen anzeigen und ändern auf Seite 283
- Parallele Verarbeitung von Prozessen durch den One Identity Manager Service auf Seite 307
- Ausführung externer Prozesse durch den StdioProcessor auf Seite 308

Einrichten von Jobservern

Die Verarbeitung der definierten Prozesse erfolgt über den One Identity Manager Service. Zur Prozessverarbeitung muss der One Identity Manager Service auf den Servern des One Identity Manager-Netzwerkes installiert sein. Ausführliche Informationen finden Sie im *One Identity Manager Installationshandbuch*.

Jobserver können Sie über verschiedene Wege einrichten:

- Bei der initialen Schemainstallation mit dem Configuration Wizard richten Sie bereits einen Jobserver mit den Serverfunktionen **SQL Ausführungsserver** und **Aktualisierungsserver** ein. Mit dem Configuration Wizard konfigurieren Sie den Dienst und installieren den Dienst remote auf einem Server.
- Weitere Jobserver richten Sie mit dem Programm Server Installer ein.

  Mit dem Server Installer erstellen Sie den Jobserver mit seinen Maschinenrollen und Serverfunktionen in der Datenbank. Mit dem Server Installer konfigurieren Sie den Dienst und installieren den Dienst remote auf einem Server.
- Jobserver können Sie im Designer erstellen.

  Im Designer können Sie einen Jobserver mit die Maschinenrollen und Serverfunktionen erstellen, den Dienst auf dem Server konfigurieren und remote installieren.
- Falls eine Remote-Installation nicht möglich ist, können Sie den Dienst lokal auf einem Server installieren und konfigurieren.
  - Installieren Sie die Dienstkomponenten mit dem Installationsassistenten auf dem Server.
  - Konfigurieren Sie den Dienst mit dem Programm Job Service Configuration.
  - Wenn der Konfigurationsparameter **Common | Jobservice | AutoCreateServerFromQueues** aktiviert ist, werden bei Anfragen des One Identity Manager Service für unbekannte Queues neue Jobserver in der Datenbank erzeugt. Die Informationen zu Maschinenrollen und Serverfunktionen werden in die Datenbank übertragen.

Die Einrichtung eines Jobservers umfasst folgende Schritte:

- Erstellen Sie einen Eintrags für den Jobserver in der One Identity Manager-Datenbank.
- Legen Sie die Maschinenrollen und Serverfunktionen für den Jobserver fest.

  Abhängig von den gewählten Maschinenrollen werden die Installationspakete ermittelt, die auf dem Jobserver installiert werden. Die Serverfunktion definiert die Funktion eines Servers in der One Identity Manager-Umgebung. Abhängig von der Serverfunktion wird die Verarbeitung der One Identity Manager-Prozesse ausgeführt.
- Installieren Sie den One Identity Manager Service.
- Konfigurieren Sie den One Identity Manager Service.
- Starten Sie den One Identity Manager Service.

Jeder Jobserver innerhalb des gesamten Netzwerkes muss eine eindeutige Queue-Bezeichnung erhalten. Mit exakt diesem Queue-Bezeichnung werden die Prozessschritte an der Jobqueue angefordert.

- Für jede Queue muss ein Jobserver in der One Identity Manager-Datenbank bekannt sein.
- Die Queue-Bezeichnung tragen Sie in die Konfigurationsdatei des One Identity Manager Service ein.

Detaillierte Informationen zum Thema

- Jobserver bearbeiten auf Seite 290
- Maschinenrollen und Serverfunktionen auf Seite 294
- Statistikinformationen eines Jobservers auf Seite 297
- Verbindungsinformationen für die Prozessgenerierung auf Seite 299
- One Identity Manager Service remote auf einem Jobserver installieren auf Seite 301
- Jobserver für die Verbindung zum Anwendungsserver konfigurieren auf Seite 304
- One Identity Manager Service Konfiguration für einen Jobserver anpassen auf Seite 313
- Arbeitsweise des One Identity Manager Service auf Seite 305
- Konfiguration des One Identity Manager Service auf Seite 309

Jobserver bearbeiten

Um einen Jobserver zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Jobserver**.
2. Fügen Sie über den Menüeintrag **Jobserver > Neu** einen neuen Jobserver ein.

   - ODER -

   Wählen Sie in der Jobserverübersicht den Jobserver zur Bearbeitung aus.
3. Bearbeiten Sie die Stammdaten des Jobservers.
4. Wählen Sie den Menüeintrag **Ansicht > Serverfunktionen** und legen Sie die Serverfunktionen fest.
5. Wählen Sie den Menüeintrag **Ansicht > Maschinenrollen** und weisen Sie die Rollen des Servers zu.

   Die von einer Serverfunktion erwarteten Maschinenrollen sind bereits zugewiesen.

Detaillierte Informationen zum Thema

- Eigenschaften von Jobservern auf Seite 291
- Maschinenrollen und Serverfunktionen auf Seite 294

- Überblick über die Serverfunktionen auf Seite 295
- Überblick über die Maschinenrollen auf Seite 296
- Statistikinformationen eines Jobservers auf Seite 297
- Verbindungsinformationen für die Prozessgenerierung auf Seite 299

Eigenschaften von Jobservern

HINWEIS: Abhängig von den installierten Modulen können weitere Eigenschaften verfügbar sein.

Tabelle 91: Eigenschaften eines Jobservers

Eigenschaft	Bedeutung
Server	Bezeichnung des Jobservers.
Vollständiger Servername	Vollständiger Servername gemäß DNS Syntax. Syntax: `<Name des Servers>.<Vollqualifizierter Domänenname>`
Server ist Cluster	Gibt an, ob der Server einen Cluster abbildet.
Server gehört zu Cluster	Cluster, zu dem der Server gehört. HINWEIS: Die Eigenschaften **Server ist Cluster** und **Server gehört zu Cluster** schließen einander aus.
IP-Adresse (IPv6)	Internet Protokoll Version 6 (IPv6)-Adresse des Servers.
IP-Adresse (IPv4)	Internet Protokoll Version 4 (IPv4)-Adresse des Servers.
Codierung	Codierung des Zeichensatzes mit der Dateien auf dem Server geschrieben werden.
Übergeordneter Jobserver	Bezeichnung des übergeordneten Jobservers.
Ausführender Server	Bezeichnung des ausführenden Servers. Eingetragen wird der Name des physisch vorhandenen Servers, auf dem die Prozesse verarbeitet werden. Diese Angabe wird bei der automatischen Aktualisierung des One Identity Manager Service ausgewertet. Verarbeitet ein Server mehrere Queues, wird mit der Auslieferung von Prozessschritten solange gewartet, bis alle Queues, die auf demselben Server abgearbeitet werden, die automatische Aktualisierung abgeschlossen haben.
Queue	Bezeichnung der Queue, welche die Prozessschritte verarbeitet. Mit dieser Queue-Bezeichnung werden die Prozess-

Eigenschaft	Bedeutung
	schritte an der Jobqueue angefordert. Die Queue-Bezeichnung wird in die Konfigurationsdatei des One Identity Manager Service eingetragen.
Serverbetriebssystem	Betriebssystem des Servers. Diese Angabe wird für die Pfadauslösung bei der Replikation von Softwareprofilen benötigt. Zulässig sind die Werte **Win32**, **Windows**, **Linux** und **Unix**. Ist die Angabe leer, wird **Win32** angenommen.
Angaben zum Dienstkonto	Benutzerkonteninformationen des One Identity Manager Service. Für die Replikation zwischen nicht vertrauenden Systemen (beispielsweise non-trusted Domänen, Linux-Server) müssen für die Server die Benutzerkonteninformationen des One Identity Manager Service in der Datenbank bekanntgegeben werden. Dazu sind das Dienstkonto, die Domäne des Dienstkontos und das Kennwort des Dienstkontos für die Server entsprechend einzutragen.
One Identity Manager Service installiert	Gibt an, ob auf diesem Server ein One Identity Manager Service installiert und aktiv ist. Die Option wird durch die Prozedur `QBM_PJobQueueLoad` aktiviert, sobald die Queue das erste Mal angefragt wird. Die Option wird nicht automatisch entfernt. Für Server, deren Queue nicht mehr aktiv ist, können Sie diese Option im Bedarfsfall manuell zurücksetzen.
Stopp One Identity Manager Service	Gibt an, ob der One Identity Manager Service gestoppt ist. Wenn diese Option für den Jobserver gesetzt ist, wird der One Identity Manager Service keine Aufträge mehr verarbeiten. Den Dienst können Sie mit entsprechenden administrativen Berechtigungen im Programm Job Queue Info stoppen und starten. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.
Kein automatisches Softwareupdate	Gibt an, ob der Server von der automatischen Softwareaktualisierung auszuschließen ist. HINWEIS: Server, für welche die Option aktiviert ist, müssen Sie manuell aktualisieren.
Softwareupdate läuft	Gibt an, ob gerade eine Softwareaktualisierung ausgeführt wird.
Port	Port zur Anzeige der Protokolldatei des One Identity Manager Service über einen Browser.
Keine direkte Daten-	Gibt an, ob der Jobserver eine direkte Verbindung zur

Eigenschaft	Bedeutung
bankverbindung	Datenbank hat. Aktivieren Sie die Option, wenn der Jobserver seine Prozesse über einen Anwendungsserver erhält.
Keine Prozesszuteilung	Gibt an, ob der Jobserver an der Lastverteilung teilnimmt.
Verbindungsinformation	Wenn der Jobserver keine direkte Verbindung zur Datenbank hat, erfassen Sie die Verbindungsinformationen zum Anwendungsserver. Verbindungsinformationen erfassen Sie im Designer in der Kategorie **Basisdaten > Sicherheitseinstellungen > Verbindungsinformationen**.
Erweiterte Eigenschaften	Zusätzliche Informationen zum Jobserver. Es werden die UID des Jobservers und die Erstellungsdaten und Änderungsdaten (Benutzer, Datum) angezeigt. Diese können nicht bearbeitet werden.
Letzter Abrufzeitpunkt	Zeitpunkt der letzten Prozessabholung.
Letzte Timeout Prüfung	Zeitpunkt der letzten Prüfung für geladene Prozessschritte, deren Auslieferung den Wert im Konfigurationsparameter **Common | Jobservice | LoadedJobsTimeOut** überschreitet.
Port extern	(Für Docker-Container) Kundenspezifischer Port zur Anzeige der Protokolldatei des One Identity Manager Service über einen Browser.
Vollständiger Servername extern	(Für Docker-Container) Kundenspezifischer vollständiger Servername gemäß DNS Syntax. Syntax: `<Name des Servers>.<Vollqualifizierter Domänenname>`
Serverfunktion	Funktion des Servers in der One Identity Manager-Umgebung. Abhängig von der Serverfunktion wird die Verarbeitung der One Identity Manager-Prozesse ausgeführt.
Maschinenrolle	Rolle der Jobserver im One Identity Manager. Abhängig von der gewählten Maschinenrolle werden die Installationspakete ermittelt, die auf dem Jobserver installiert werden.

Verwandte Themen

- Überblick über die Serverfunktionen auf Seite 295
- Überblick über die Maschinenrollen auf Seite 296
- Statistikinformationen eines Jobservers auf Seite 297
- Verbindungsinformationen für die Prozessgenerierung auf Seite 299
- JobServiceDestination auf Seite 324

Maschinenrollen und Serverfunktionen

Eine Maschinenrolle beschreibt die Rolle, die ein Computer oder ein Server in einer One Identity Manager-Umgebung einnimmt. Jeder Computer oder Server kann eine oder mehrere Rollen übernehmen. Somit können eine oder mehrere Maschinenrollen zugeordnet werden. Die Maschinenrollen werden bei der Installation der One Identity Manager Komponenten gewählt.

Maschinenrollen sind hierarchisch aufgebaut. Wird eine Maschinenrolle bei der Installation gewählt so werden zusätzlich alle übergeordneten Maschinenrollen zugewiesen.

Beispiel: Aufbau von Maschinenrollen

```
Server
    Jobserver
        Active Directory
```

Wird bei der Installation die Maschinenrolle **Active Directory** gewählt, dann werden zusätzlich die Maschinenrollen **Jobserver** und **Server** zugewiesen.

Einige Maschinenrolle wie beispielsweise **Web** können bei der Installation nicht aktiv gewählt werden. Diese Maschinenrollen werden automatisch bei der Installation der verschiedenen Webanwendungen mit dem Web Installer zugewiesen.

Maschinenrollen für die Installation des One Identity Manager Service sind mit Serverfunktionen verbunden. Die Serverfunktion definiert die Funktion eines Servers in der One Identity Manager-Umgebung. Abhängig von der Serverfunktion wird die Verarbeitung der One Identity Manager-Prozesse ausgeführt. Bei der Installation eines Servers sind aufgrund der ausgewählten Maschinenrolle die möglichen Serverfunktionen vorgegeben.

Beispiel: Zusammenhang von Maschinenrollen und Serverfunktionen

Die Maschinenrolle **Active Directory** ist mit der Serverfunktion **Active Directory Konnektor** verbunden. Somit steht nach Installation dieser Maschinenrolle der Server im One Identity Manager als Synchronisationsserver bei der Einrichtung eines Active Directory Synchronisationsprojektes zur Verfügung.

In einer Maschinenrolle sind die Installationspakete und Dateien festgelegt, welche auf dem Computer oder dem Server installiert werden sollen. Die Informationen über die Maschinenrollen, die Installationspakete und die Dateien werden bei der Installation in einer Datei `InstallState.config` gespeichert und stehen damit für die automatische Softwareaktualisierung zur Verfügung.

HINWEIS: Wenn Sie neue Dateien mit dem Software Loader in die One Identity Manager-Datenbank importieren, sollten Sie die Dateien einer Maschinenrolle zuordnen. Damit wird sichergestellt, das die Dateien über die Mechanismen der automatischen Softwareaktualisierung verteilt werden. Ausführliche Informationen zur automatischen Softwareaktualisierung finden Sie im *One Identity Manager Installationshandbuch*.

Verwandte Themen

- Überblick über die Serverfunktionen auf Seite 295
- Überblick über die Maschinenrollen auf Seite 296
- Jobserver bearbeiten auf Seite 290

Überblick über die Serverfunktionen

Um Informationen zu Serverfunktionen anzuzeigen

- Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Serverfunktionen**.

Die Serverfunktion definiert die Funktion eines Servers in der One Identity Manager-Umgebung. Abhängig von der Serverfunktion wird die Verarbeitung der One Identity Manager-Prozesse ausgeführt.

HINWEIS: Abhängig von den installierten Modulen können weitere Serverfunktionen verfügbar sein.

Tabelle 92: Zulässige Serverfunktionen

Serverfunktion	Anmerkungen
Aktualisierungsserver	Der Server führt die automatische Softwareaktualisierung aller anderen Server aus. Der Server benötigt eine direkte Verbindung zum Datenbankserver, auf dem die One Identity Manager-Datenbank installiert ist. Der Server kann SQL Aufträge ausführen. Bei der initialen Schemainstallation wird der Server, auf dem die One Identity Manager-Datenbank installiert ist, mit dieser Serverfunktion gekennzeichnet.
SQL Ausführungsserver	Der Server kann SQL Aufträge ausführen. Der Server benötigt eine direkte Verbindung zum Datenbankserver, auf dem die One Identity Manager-Datenbank installiert ist. Für eine Lastverteilung der SQL Prozesse können mehrere SQL Ausführungsserver eingerichtet werden. Das System verteilt die erzeugten SQL Prozesse über alle Jobserver mit dieser Serverfunktion.
CSV Skriptserver	Der Server kann CSV-Dateien per Prozesskomponente

Serverfunktion	Anmerkungen
	`ScriptComponent` verarbeiten.
One Identity Manager Service installiert	Server, auf dem ein One Identity Manager Service installiert werden soll.
SMTP Host	Server, auf dem durch den One Identity Manager Service E-Mail Benachrichtigungen verschickt werden. Voraussetzung zum Versenden von Mails durch den One Identity Manager Service ist ein konfigurierter SMTP Host.
Standard Berichtserver	Server, auf dem die Berichte generiert werden.

Verwandte Themen

- Maschinenrollen und Serverfunktionen auf Seite 294
- Jobserver bearbeiten auf Seite 290
- Überblick über die Maschinenrollen auf Seite 296

Überblick über die Maschinenrollen

Um Informationen zu Maschinenrollen anzuzeigen

- Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Maschinenrollen**.

Abhängig von der gewählten Maschinenrolle werden die Installationspakete ermittelt, die auf dem Jobserver installiert werden.

Tabelle 93: Mögliche Maschinenrollen und zugehörige Installationspakete

Maschinenrolle		Beschreibung zum Installationspaket
Workstation		Enthält alle Basiskomponenten zur Installation der Werkzeuge auf einer administrativen Arbeitsstation.
	Administration	Enthält die One Identity Manager Administrationswerkzeuge, die ein Standardbenutzer zur Erfüllung seiner Aufgaben mit dem One Identity Manager benötigt. Neben den Werkzeugen, welche die Grundfunktionalität für die Arbeit mit One Identity Manager sicherstellen, zählt dazu auch der Manager als zentrales Administrationswerkzeug.
	Configuration	Enthält alle One Identity Manager Werkzeuge des Standardbenutzers und zusätzliche Programme,

Maschinenrolle		Beschreibung zum Installationspaket
		welche zur Konfiguration des Systems erforderlich sind. Dazu gehören beispielsweise Configuration Wizard, Database Compiler, Database Transporter, Crypto Configuration, Designer, Web Designer sowie Konfigurationswerkzeuge für den One Identity Manager Service.
	Development & Testing	Enthält die One Identity Manager Werkzeuge zur Entwicklung und zum Testen kundenspezifischer Skripte und Formulare, wie beispielsweise System Debugger.
	Monitoring	Enthält One Identity Manager Programme zur Überwachung des Systemstatus, wie beispielsweise Job Queue Info.
Documentation		Enthält die One Identity Manager Dokumentation in verschiedenen Sprachen.
Server		Enthält alle Basiskomponenten zur Einrichtung eines Servers.
	Jobserver	Enthält den One Identity Manager Service und die Basisprozesskomponenten. Zusätzliche Maschinenrollen enthalten die Konnektoren zur Synchronisation der einzelnen Zielsysteme.

HINWEIS: In der Kategorie **Basisdaten > Installationen > Maschinenrollen** werden zusätzlich die Maschinenrollen **API** und **Web** angezeigt. Diese sind für die interne Verwendung reserviert und dürfen nicht geändert oder zugewiesen werden.

Verwandte Themen

- Maschinenrollen und Serverfunktionen auf Seite 294
- Überblick über die Serverfunktionen auf Seite 295
- Jobserver bearbeiten auf Seite 290

Statistikinformationen eines Jobservers

Die Statistikinformationen eines Jobservers werden ausgewertet und bilden die Grundlage für die Konfigurationsvorschläge für die Ladeintervalle von Jobservern. Es werden die Statistikinformationen der letzten 100 Tage in die Berechnung der Konfigurationsvorschläge einbezogen. Diese Konfigurationsvorschläge sollten Sie bei der Konfiguration des One Identity Manager Service berücksichtigen.

Um Statistikinformationen zu ermitteln

- Aktivieren Sie im Designer den Konfigurationsparameter **Common | JobQueueStats**. Ist der Konfigurationsparameter aktiviert, werden Statistikinformationen des One Identity Manager Service in die Tabelle `JobQueueStats` geschrieben.
- Aktivieren Sie im Designer den Konfigurationsparameter **Common | JobQueueStats | MaxAge** und geben Sie die Aufbewahrungszeit der Statistikinformationen in Tagen an.

Es werden bei jeder Aktion in der Jobqueue, wie Einfügen, Ändern oder Löschen von Prozessen, neue Statistikeinträge für die Jobserver erstellt. Durch den DBQueue Prozessor Auftrag `QBMJobQueueStatsShrink` werden diese Einträge stundenweise komprimiert. Die Komprimierung erfolgt für alle Stunden, die nicht der aktuellen Stunde entsprechen.

Um Statistikinformationen der Jobserver anzuzeigen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Jobserver**.
2. Starten Sie den Jobservereditor über die Aufgabe **Jobserver bearbeiten**.
3. Wählen Sie in der Jobserverübersicht den Jobserver zur Bearbeitung aus.
4. Wählen Sie die Spalten mit Statistikinformationen über das Kontextmenü **Spalten wählen**.

   Die Spalten mit Statistikinformationen werden in der Jobserverübersicht farblich hervorgehoben.

Tabelle 94: Spalten für die Abbildung der Statistikinformationen

Spalte	Bezeichnung	Bedeutung
AverageLoad	Gemittelt Prozess/Stunde	Durchschnittliche Anzahl der Prozesse pro Stunde.
MaxLoad	Maximal Prozes-se/Stunde	Maximale Anzahl der Prozesse pro Stunde.
LoadDuration	Empfohlenes Ladeintervall (Sekunden)	Konfigurationsvorschlag für den Parameter **Prozessanfrageintervall** (`StartInterval`) in der One Identity Manager Service Konfiguration.
StatisticsDuration	Empfohlenes Statistikintervall (Sekunden)	Konfigurationsvorschlag für den Parameter **Zeitintervall für Statis-tikinformationen** (`StatisticInterval`) in der One Identity Manager Service Konfiguration.

Verwandte Themen

- Konfiguration des One Identity Manager Service auf Seite 309
- JobServiceDestination auf Seite 324

Verbindungsinformationen für die Prozessgenerierung

Für die Generierung von Prozessen für die Jobserver werden der Provider, Verbindungsparameter und die Authentifizierungsdaten benötigt. Diese Informationen werden im Standardfall aus den Verbindungsdaten der Datenbank ermittelt.

Wenn ein Jobserver keine direkte Verbindung zur Datenbank hat, sondern gegen einen Anwendungsserver arbeitet:

- Erfassen Sie die Verbindungsinformationen zum Anwendungsserver.
- Kennzeichnen Sie den Jobserver mit der Option **Keine direkte Datenbankverbindung** und weisen Sie die Verbindungsinformationen zum Anwendungsserver zu.

TIPP: Kennzeichnen Sie eine der Verbindungsinformationen zum Anwendungsserver als **Fallback Verbindung**. Diese Verbindungsinformation wird verwendet, wenn Sie am Jobserver keinen Verweis auf eine konkrete Verbindungsinformation eintragen.

Ermitteln der Verbindungsinformationen bei der Prozessgenerierung

- Für alle Jobserver mit direkter Datenbankverbindung werden die Verbindungsinformationen aus den Datenbankinformationen verwendet.
- Für Jobserver ohne direkte Datenbankverbindung werden die Verbindungsinformationen folgendermaßen ermittelt:
  1. Verbindungsinformation, die am Jobserver eingetragen ist.
  2. Verbindungsinformation, die als Fallback Verbindung gekennzeichnet ist.
  3. Verbindungsinformationen, die in den Datenbankinformationen eingetragen sind.

Detaillierte Informationen zum Thema

- Verbindungsdaten zur Datenbank ändern auf Seite 38
- Verbindungsinformationen zum Anwendungsserver erfassen auf Seite 299
- Verbindungsinformationen am Jobserver eintragen auf Seite 301

Verbindungsinformationen zum Anwendungsserver erfassen

Erfassen Sie die Verbindungsinformationen zum Anwendungsserver.

Um Verbindungsinformationen zum Anwendungsserver zu erfassen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Sicherheitseinstellungen> Verbindungsinformationen**.
2. Fügen Sie über den Menüeintrag **Objekt > Neu** eine neue Verbindungsinformation ein.
3. Erfassen Sie folgende Informationen.

Tabelle 95: Eigenschaften einer Verbindungsinformation

Eigenschaft	Beschreibung
Anzeigename	Anzeigename der Verbindungsinformation. Unter diesem Anzeigenamen kann die Verbindungsinformation am Jobserver-Eintrag ausgewählt werden.
Fallback Verbindung	Kennzeichnen Sie eine der Verbindungsinformationen zum Anwendungsserver als **Fallback Verbindung**. Diese Verbindungsinformation wird verwendet, wenn Sie am Jobserver keinen Verweis auf eine konkrete Verbindungsinformation eintragen.
Provider	Für Verbindungsinformationen zum Anwendungsserver, wählen Sie **Anwendungsserver**.
Verbindungsparameter	Webadresse (URL) zum Anwendungsserver. Über die Schaltfläche **...** wird der Standardverbindungsdialog geöffnet, über welchen Sie weitere Optionen festlegen können und die Verbindung testen können.

Authentifizierungsdaten	Erfassen Sie die Authentifizierungsdaten.
	Syntax:
	`Module=<Name>;<Property1>=<Value1>;<Property2>=<Value2>`
	Beispiel:
	`Module=DialogUserAccountBased`
	Über die Schaltfläche **...** wird ein Dialog geöffnet, über den Sie das Authentifizierungsmodul direkt auswählen können. Die Authentifizierungsdaten werden beim Schließen des Dialogs übernommen.
	Ausführliche Informationen zu den One Identity Manager Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Verwandte Themen

- Verbindungsinformationen für die Prozessgenerierung auf Seite 299
- Verbindungsinformationen am Jobserver eintragen auf Seite 301

Verbindungsinformationen am Jobserver eintragen

Kennzeichnen Sie den Jobserver mit der Option **Keine direkte Datenbankverbindung** und weisen Sie die Verbindungsinformationen zum Anwendungsserver zu.

Um die Verbindungsinformation am Jobserver bekanntzugeben

1. Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Jobserver**.
2. Wählen Sie in der Jobserverübersicht den Jobserver zur Bearbeitung aus.
3. Auf dem Tabreiter **Eigenschaften** bearbeiten Sie folgende Informationen.
    - Aktivieren Sie für den Jobserver die Option **Keine direkte Datenbankverbindung**.
    - Wählen Sie unter **Verbindungsinformation** die Verbindungsinformation zum Anwendungsserver.

Verwandte Themen

- Verbindungsinformationen für die Prozessgenerierung auf Seite 299
- Verbindungsinformationen zum Anwendungsserver erfassen auf Seite 299

One Identity Manager Service remote auf einem Jobserver installieren

Im Jobservereditor haben Sie die Möglichkeit einzelne Jobserver remote zu installieren. Der Assistent zur Remote-Installation führt die folgende Schritte aus:

- Installieren der One Identity Manager Service Komponenten.
- Konfigurieren des One Identity Manager Service.
- Starten des One Identity Manager Service.

HINWEIS: Für die Generierung von Prozessen für die Jobserver werden der Provider, Verbindungsparameter und die Authentifizierungsdaten benötigt. Diese Informationen werden im Standardfall aus den Verbindungsdaten der Datenbank ermittelt. Arbeitet der Jobserver über einen Anwendungsserver müssen Sie zusätzliche Verbindungsinformationen im Designer konfigurieren. Weitere Informationen finden Sie unter Jobserver für die Verbindung zum Anwendungsserver konfigurieren auf Seite 304.

Voraussetzung für die Remote-Installation

- Der Jobserver ist in der Datenbank eingetragen.
- Es steht ein Benutzerkonto mit ausreichenden Berechtigungen zur Installation des One Identity Manager Service zur Verfügung.
- Die Remote-Installation wird nur innerhalb einer Domäne oder in Domänen mit Vertrauensstellung unterstützt.

HINWEIS: Wenn Sie mit einer verschlüsselten One Identity Manager-Datenbank arbeiten, beachten Sie die im *One Identity Manager Installationshandbuch* aufgeführten Hinweise zum Arbeiten mit einer verschlüsselten Datenbank.

Um den One Identity Manager Service remote zu installieren

1. Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Jobserver**.
2. Starten Sie den Jobservereditor über die Aufgabe **Jobserver bearbeiten**.
3. Wählen Sie in der Jobserverübersicht den Jobserver zur Bearbeitung aus.
4. Wählen Sie den Menüeintrag **Jobserver > Dienst installieren**.

   Der Assistent zur Remote-Installation des One Identity Manager Service wird gestartet.
5. Auf der Startseite des Assistenten klicken Sie **Weiter.**
6. Auf der Seite **Dienst konfigurieren** erfassen Sie die Konfigurationseinstellungen des One Identity Manager Service.

   Die initiale Konfiguration des Dienstes für die Verbindung mit einer Datenbank ist bereits vordefiniert. Um diese Vorlage zu nutzen, erfassen Sie die Verbindungsdaten für die Prozessabholung. Für eine erweitere Konfiguration sind in der Modulliste die einzelnen Konfigurationssektionen des One Identity Manager Service aufgeführt.

   - Für eine direkte Verbindung zu Datenbank:
     1. Wählen Sie **Prozessabholung > sqlprovider**
     2. Klicken Sie auf den Eintrag **Verbindungsparameter** und klicken Sie die Schaltfläche **Bearbeiten**.
     3. Erfassen Sie die Verbindungsdaten zur One Identity Manager-Datenbank.
   - Für eine Verbindung zum Anwendungsserver:
     1. Wählen Sie **Prozessabholung**, klicken Sie die Schaltfläche **Einfügen** und wählen Sie **AppServerJobProvider**.
     2. Klicken Sie auf den Eintrag **Verbindungsparameter** und klicken Sie die Schaltfläche **Bearbeiten**.
     3. Erfassen Sie die Verbindungsdaten zum Anwendungsserver.
     4. Klicken Sie auf den Eintrag **Authentifizierungsdaten** und klicken Sie die Schaltfläche **Bearbeiten**.

5. Wählen Sie das Authentifizierungsmodul. Abhängig vom Authentifizierungsmodul können weitere Daten, wie beispielsweise Benutzer und Kennwort erforderlich sein. Ausführliche Informationen zu den One Identity Manager Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

7. Auf der Seite **Installationsquelle und -ziel** erfassen Sie folgende Informationen.
   a. Allgemeine Informationen:
      - **Installationsverzeichnis**: Wählen Sie das Verzeichnis mit den Installationsdateien.
      - **Privater Schlüssel**: Wenn die Datenbank verschlüsselt ist, wählen Sie die Datei mit dem privaten Schlüssel.
   b. Klicken Sie **Weiter**.
   c. Erfassen Sie die Installationsinformationen für den Dienst.
      - **Computer**: Erfassen Sie den Namen oder die IP-Adresse des Servers, auf dem der Dienst installiert und gestartet wird.
      - **Dienstkonto**: Erfassen Sie die Angaben zum Benutzerkonto unter dem der One Identity Manager Service läuft. Erfassen Sie das Benutzerkonto, das Kennwort zum Benutzerkonto und die Kennwortwiederholung.

      Die Installation des Dienstes erfolgt mit dem Benutzerkonto, mit dem Sie an der administrativen Arbeitsstation angemeldet sind. Möchten Sie ein anderes Benutzerkonto für die Installation des Dienstes nutzen, können Sie dieses in den erweiterten Optionen eintragen. Angaben zum One Identity Manager Service können Sie ebenfalls über die erweiterten Optionen ändern, beispielsweise das Installationsverzeichnis, den Namen, den Anzeigenamen und die Beschreibung für den One Identity Manager Service.
8. Um die Installation des Dienstes zu starten, klicken Sie **Weiter**.

   Die Installation des Dienstes wird automatisch ausgeführt und kann einige Zeit dauern.
9. Klicken Sie **Fertig**, um den Installationsassistenten zu beenden.

HINWEIS: In einer Standardinstallation wird der Dienst mit der Bezeichnung **One Identity Manager Service** in der Dienstverwaltung des Servers eingetragen.

TIPP: Über den Menüeintrag **Jobserver > HTTP Anfrage starten** wird für einen Jobserver der HTTP Server des One Identity Manager Service angesprochen und die verschiedenen Dienste des One Identity Manager Service werden angezeigt.

Verwandte Themen

- Einrichten von Jobservern auf Seite 289
- Arbeitsweise des One Identity Manager Service auf Seite 305
- Konfiguration des One Identity Manager Service auf Seite 309

Jobserver für die Verbindung zum Anwendungsserver konfigurieren

Um einen Jobserver für die Verbindung zum Anwendungsserver zu konfigurieren

- Geben Sie den Jobserver in der One Identity Manager-Datenbank bekannt.
- Installieren Sie den One Identity Manager Service und konfigurieren Sie den **AppServerJobProvider** für die Prozessabholung.
- Für die Generierung der Prozesse über einen Anwendungsserver tragen Sie die Verbindungsinformationen am Jobserver ein.
    - Erfassen Sie die Verbindungsinformationen zum Anwendungsserver.
    - Kennzeichnen Sie den Jobserver mit der Option **Keine direkte Datenbankverbindung** und weisen Sie die Verbindungsinformationen zum Anwendungsserver zu.

Verwandte Themen

- Jobserver bearbeiten auf Seite 290
- One Identity Manager Service remote auf einem Jobserver installieren auf Seite 301
- AppServerJobProvider auf Seite 323
- Verbindungsinformationen für die Prozessgenerierung auf Seite 299

9

Arbeitsweise des One Identity Manager Service

Der One Identity Manager Service sorgt für die Verbreitung der in der One Identity Manager-Datenbank verwalteten Informationen im Netzwerk. Der One Identity Manager Service übernimmt die Datensynchronisation zwischen Datenbank und den angebundenen Zielsystemen sowie die Durchführung von Aktionen auf Datenbank- und Dateiebene.

Die Prozessschritte werden von Prozesskomponenten ausgeführt. Der One Identity Manager Service erzeugt dazu eine Instanz der benötigten Prozesskomponente und übergibt die Parameter des Prozessschrittes. Eine Entscheidungslogik überwacht die Ausführung der Prozessschritte und veranlasst abhängig vom gemeldeten Ausführungsergebnis die weitere Verarbeitung des Prozesses.

Ausführliche Informationen zur Installation und zur Aktualisierung des One Identity Manager Service finden Sie im *One Identity Manager Installationshandbuch*.

Ausführliche Informationen zur Protokollierung und Überwachung der Prozessverarbeitung sowie zur Unterstützung bei der Fehlersuche finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.

Verwandte Themen

- Verarbeitung von Prozessen durch den One Identity Manager Service auf Seite 306
- Parallele Verarbeitung von Prozessen durch den One Identity Manager Service auf Seite 307
- Ausführung externer Prozesse durch den StdioProcessor auf Seite 308
- Konfiguration des One Identity Manager Service auf Seite 309
- Einrichten von Jobservern auf Seite 289
- Prozess-Orchestrierung im One Identity Manager auf Seite 244

Verarbeitung von Prozessen durch den One Identity Manager Service

Die Verarbeitungsaufträge für den One Identity Manager Service werden in der Jobqueue (Tabelle `JobQueue`) für eine definierte Queue abgelegt. Ein One Identity Manager Service kann mehrere Queues verarbeiten. Die Queues, die ein One Identity Manager Service verarbeiten kann, werden in der Konfiguration des One Identity Manager Service bekanntgegeben. Für jede Queue muss ein Jobserver in der One Identity Manager-Datenbank bekannt sein. Der One Identity Manager Service fragt in der Jobqueue, welche Prozesse für seine Queues bereitstehen. Die Initialisierung einer Queue erfolgt beim Start des One Identity Manager Service.

Die Prozessanfragen und die Verarbeitungsergebnisse werden intern in einer Anfrage-Queue (`RequestQueue`) und einer Rückmelde-Queue (`ResultQueue`) gechached. Die Anfrage-Queue wird parallel zur Rückmelde-Queue abgearbeitet. Über die zwei internen Queues gehen folgende Anfragen:

- Anfrage-Queue
  - Anfrage der anstehenden Prozessschritte
  - Anfrage von Statistikinformationen
- Rückmelde-Queue
  - Anfrage der Verarbeitungsergebnisse
  - Anfrage des Verarbeitungsstatus
  - Anfrage von Ereignisse
  - Initiale Anfrage der anstehenden Prozessschritte beim Start des One Identity Manager Service. Damit wird sichergestellt, dass noch fehlende Verarbeitungsergebnisse in der Datenbank angekommen sind, bevor Prozessschritte, die noch im Status **Loaded** sind, wieder aktiviert werden.

Beim Herunterfahren des One Identity Manager Service werden eventuell noch in der Rückmelde-Queue vorhandene Anfragen in eine Datei serialisiert, die beim nächsten Start des One Identity Manager Service verarbeitet wird. Damit soll sichergestellt werden, dass keine Verarbeitungsergebnisse verloren gehen. Die Sicherungsdateien werden im lokalen Anwendungsdatenverzeichnis abgelegt (`%APPDATA%\One Identity\One Identity Manager\JobService`).

Verwandte Themen

- Konfiguration des One Identity Manager Service auf Seite 309
- Einrichten von Jobservern auf Seite 289

Parallele Verarbeitung von Prozessen durch den One Identity Manager Service

Der One Identity Manager Service ermöglicht die parallele Verarbeitung von Prozessschritten, da er mehrere Instanzen von Prozesskomponenten erzeugen kann. Ob eine parallele Verarbeitung von Prozessschritten erfolgen kann, ist für jede Prozesskomponente und ihre Prozessfunktionen festgelegt.

Die Parallelisierung wird über folgende Eigenschaften der Prozessfunktionen und Prozesskomponten beeinflusst:

- Anzahl der maximale zulässigen Instanzen für eine Prozessfunktion (`JobTask.MaxInstance`)

  Zulässige Werte sind:

  - **-1**: Alle Instanzen dieser Prozessfunktion werden nacheinander abgearbeitet. Weitere Instanzen von Prozessfunktionen derselben Prozesskomponente werden zeitgleich nicht ausgeführt.
  - **0**: Es wird die an der Prozesskomponente angegebene Anzahl der maximalen Instanzen genutzt.
  - **1** oder größer: Genaue Anzahl der Instanzen einer Prozessfunktion, die gleichzeitig abgearbeitet werden.
- Anzahl der maximale zulässigen Instanzen für eine Prozesskomponte (`JobComponent.MaxInstance`)

  Der Wert wird nur genutzt, wenn die maximale Anzahl der Instanzen einer Prozessfunktion auf **0** gesetzt ist. Andernfalls gilt der an der Prozessfunktion gesetzte Wert.

  Zulässige Werte sind:

  - **-1**: Alle Instanzen dieser Prozesskomponente werden nacheinander abgearbeitet.

    Für diese Komponenten muss sichergestellt sein, dass sie exklusiv auf einem Jobserver ausgeführt werden, das heißt es dürfen nicht mehrere Queues vorhanden sein, die Prozesse dieser Komponente verarbeiten.
  - **0**: Alle Instanzen dieser Prozesskomponente können gleichzeitig abgearbeitet werden.
  - **1** oder größer: Genaue Anzahl der Instanzen einer Prozesskomponente, die gleichzeitig abgearbeitet werden.
- Angabe, ob die exklusive Ausführung einer Prozessfunktion pro Objekt erforderlich ist (`JobTask.IsExclusivePerObject`)

  Ist die Option für eine Prozessfunktion aktiviert, darf für ein bestimmtes Objekt immer nur ein Prozessschritt mit dieser Prozessfunktion ausgeführt werden. Es erfolgt keine Parallelverarbeitung.

Verwandte Themen

- Übersicht über die Prozesskomponenten auf Seite 281
- Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen auf Seite 286
- Anzahl der maximalen Instanzen für Prozessfunktionen und Prozesskomponenten ändern auf Seite 284

Ausführung externer Prozesse durch den StdioProcessor

Über den Ausführungstyp einer Prozessfunktion ist definiert, ob die Verarbeitung intern im One Identity Manager Service oder in einem externen Prozess erfolgen soll. Wenn eine Prozessfunktion in einem externen Prozess ausgeführt werden soll, wird automatisch der StdioProcessor (`StdioProcessor.exe`) in einem externen Slot gestartet. Nach der Ausführung bleibt der externe Prozess für weitere Ausführungen verfügbar, bis eine der folgenden Bedingungen eintritt:

- Es wurde für 30 Sekunden kein weiterer Prozessschritt in diesem externen Slot gestartet.
- Die Anzahl der maximalen Nachnutzungen laut Konfiguration des One Identity Manager Service wurde erreicht.

  Die Anzahl der maximalen Nachnutzungen wird in Konfiguration des One Identity Manager Service über den Parameter **Max. Nachnutzung externer Prozessoren** (`MaxExternalSlotReuse`) festgelegt. Der Standardwert ist **100**. Weitere Informationen finden Sie unter JobServiceDestination auf Seite 324.
- Der Prozessschritt hat den Rückgabe-Code `ErrorAndTerminate` zurückgegeben.

Verwandte Themen

- Übersicht über die Prozesskomponenten auf Seite 281
- Eigenschaften von Prozesskomponenten, Prozessfunktionen und Parametervorlagen auf Seite 286
- Ausführungstyp von Prozessfunktionen anzeigen und ändern auf Seite 283
- Konfiguration des One Identity Manager Service auf Seite 309

Konfiguration des One Identity Manager Service

Innerhalb des One Identity Manager Service stellt ein Jobprovider einer Jobdestination die Prozessschritte zur Verfügung. Die Jobdestination verarbeitet die Prozessschritte und liefert ein Ergebnis an den Jobprovider zurück. Der Jobprovider wertet das Ergebnis aus.

Die Kombination von einem Jobprovider auf dem einen Server und einer Jobdestination auf einem anderen Server wird auch als Jobgate bezeichnet. In einem Jobgate sind Jobprovider und Jobdestination so konfiguriert, dass diese miteinander kommunizieren können.

Abbildung 27: Beispiel für die Konfiguration des One Identity Manager Service

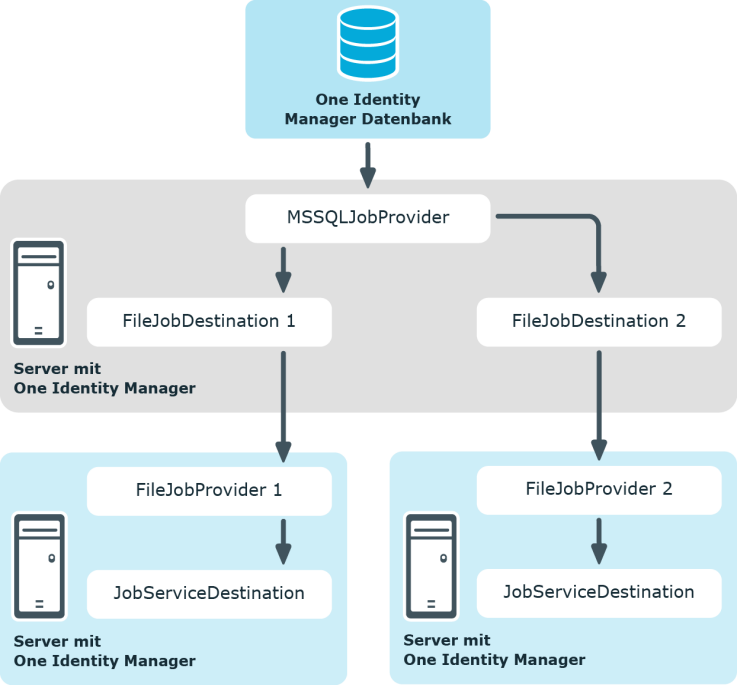


Tabelle 96: Provider des One Identity Manager Service

Provider	Beschreibung
MSSQLJobProvider	Der **MSSQLJobProvider** holt die Prozessschritte von der One Identity Manager-Datenbank unter SQL Server ab und sendet diese an eine Jobdestination.
FileJobProvider	Im **FileJobProvider** werden Prozessanfragen und Ergebnisse aus Dateien gelesen und in Dateien geschrieben. Diese Dateien können vom **FileJobGate** (**FileJobDestination** oder **FTPJobDestination**) verarbeitet werden. Die Übertragung der Daten erfolgt mittels Dateien.

Provider	Beschreibung
FTPJobProvider	Der **FTPJobProvider** basiert auf der Funktionalität des **FileJobProviders**. Im **FTPJobProvider** werden Prozessanfragen und Ergebnisse aus Dateien gelesen und in Dateien geschrieben. Nachdem die Dateien auf dem lokalen Verzeichnis erstellt sind, verbindet sich der **FTPJobProvider** zum FTP Server und verschiebt die Dateien auf den Server. Bei einer Signalisierung wird auch eine Verbindung zum FTP Server erstellt und die Daten werden geholt.
HTTPJobProvider	Der **HTTPJobProvider** empfängt Prozessschritte von einem übergeordneten Jobserver. Die Datenübertragung erfolgt über HTTP.
AppServerJobProvider	Der **AppServerJobProvider** holt die Prozessschritte vom Anwendungsserver ab und sendet diese an eine Jobdestination.

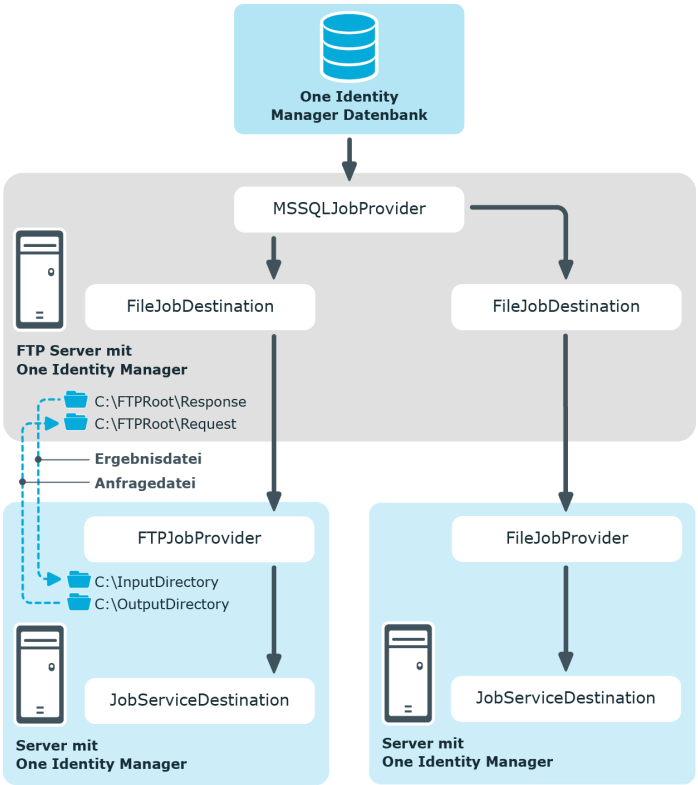
Tabelle 97: Jobdestinations des One Identity Manager Service

Jobdestination	Beschreibung
JobServiceDestination	Die **JobServiceDestination** ist die Komponente des One Identity Manager Service, welche die eigentliche Abarbeitung der Prozessschritte übernimmt. Sie fordert die Prozessschritte beim Jobprovider an, verarbeitet diese mit den Prozesskomponenten und liefert das Verarbeitungsergebnis zurück.
FileJobDestination	Die **FileJobDestination** verarbeitet die vom **FileJobGate** (**FileJobProvider** oder **FTPJobProvider**) zur Verfügung gestellten Prozessschritte und liefert die Ergebnisse an den Jobprovider zurück.
FTPJobDestination	Die **FTPJobDestination** verarbeitet die vom **FileJobGate** (**FileJobProvider** oder **FTPJobProvider**) zur Verfügung gestellten Prozessschritte und liefert die Ergebnisse an den Jobprovider zurück.
HTTPJobDestination	Die **HTTPJobDestination** versendet Prozessschritte an einen nachgeordneten Jobserver. Die Datenübertragung erfolgt über HTTP.

Tabelle 98: Jobgates des One Identity Manager Service

Jobgate	Beschreibung
HTTPJobGate	Bestehend aus **HTTPJobProvider** und **HTTPJobDestination**.
FileJobGate	Bestehend aus **FileJobProvider**, **FileJobDestination**, **FTPJobProvider** und **FTPJobDestination**. Jobprovider und Jobdestinations können untereinander kombiniert werden.

Abbildung 28: Beispielkonfiguration für FileJobGate



Detaillierte Informationen zum Thema

- Konfigurationsdateien des One Identity Manager Service auf Seite 312
- One Identity Manager Service Konfiguration für einen Jobserver anpassen auf Seite 313
- Modul Prozessabholung auf Seite 317
- Modul Jobdestination auf Seite 324
- Modul Konfiguration auf Seite 333
- Modul Logwriter auf Seite 335
- Modul Dispatcher auf Seite 338
- Modul Connection auf Seite 340
- Modul HTTP-Authentifizierung auf Seite 341
- Modul Plugins auf Seite 342
- Modul Datei mit privatem Schlüssel auf Seite 346

Konfigurationsdateien des One Identity Manager Service

Den One Identity Manager Service und seine Plugins konfigurieren Sie über eine Konfigurationsdatei. Diese Datei muss im gleichen Verzeichnis wie die Datei `viNetworkService.exe` liegen. Die Konfigurationsdatei ist sowohl für den One Identity Manager Service auf einem windowsbasierten Betriebssystem als auch für den Linux-Deamon notwendig.

Es werden zwei Formate der Konfigurationsdatei unterstützt:

- Jobservice.cfg

  Die `Jobservice.cfg` ist eine XML-Konfigurationsdatei im eigenen Format. Vorteil dieser Datei ist, dass ein Nachladen im laufenden Betrieb unterstützt wird.

- viNetworkService.exe.config

  Die `viNetworkService.exe.config` ist die Standard-Konfigurationsdatei für .NET-exe's und hat das dafür vorgegebene Format.

Bei der Ermittlung der Einstellungen wird ein Parameter zuerst in der Konfigurationsdatei `Jobservice.cfg` gesucht. Wird der Parameter nicht gefunden, wird automatisch die Konfigurationsdatei `viNetworkService.exe.config` benutzt. Somit kann der One Identity Manager Service auch nur mit der Konfigurationsdatei `viNetworkService.exe.config` arbeiten.

Die Konfiguration des One Identity Manager Service bearbeiten Sie im Designer in der **Kategorie Basisdaten > Installationen > Jobserver** oder mit dem Programm Job Service Configuration.

Für die verschiedenen Module innerhalb des One Identity Manager Service gibt es jeweils eine eigene Sektion innerhalb der Konfigurationsdatei.

Tabelle 99: Module des One Identity Manager Service

Modul	Beschreibung
Prozessabholung	In diesem Modul legen Sie den Jobprovider fest.
Jobdestination	In diesem Modul definieren Sie die Jobdestination.
Konfiguration	In diesem Modul nehmen Sie die allgemeinen Konfigurationseinstellungen für den One Identity Manager Service vor.
Logwriter	Dieses Modul schreibt die Meldungen des One Identity Manager Service in eine Protokolldatei.
Dispatcher	Mit diesem Modul konfigurieren Sie den One Identity Manager Service als Dispatcher. Die Prozessanforderungen vom nachgeschalteten Jobserver werden dann von diesem Jobserver zwischengespeichert, verarbeitet und weitergeleitet.
Connection	Mit diesem Modul nehmen Sie spezielle Konfigurationseinstellungen

Modul	Beschreibung
	zum Verhalten des One Identity Manager Service vor.
HTTP-Authentifizierung	Über dieses Modul legen Sie fest, wie die Authentifizierung am HTTP Server erfolgt, um auf die erweiterten Dienste zuzugreifen, beispielsweise zur Anzeige der Protokolldatei oder der Statusanzeige.
Plugins	In diesem Modul legen Sie fest, welche Plugins installiert werden sollen.
Datei mit privatem Schlüssel	In diesem Modul erfassen Sie die Informationen zu Dateien mit einem privaten Schlüssel. Verwenden Sie das Modul, wenn Sie mit mehreren privaten Schlüsseln arbeiten.

Detaillierte Informationen zum Thema

- Konfiguration des One Identity Manager Service auf Seite 309
- One Identity Manager Service Konfiguration für einen Jobserver anpassen auf Seite 313
- Modul Prozessabholung auf Seite 317
- Modul Jobdestination auf Seite 324
- Modul Konfiguration auf Seite 333
- Modul Logwriter auf Seite 335
- Modul Dispatcher auf Seite 338
- Modul Connection auf Seite 340
- Modul HTTP-Authentifizierung auf Seite 341
- Modul Plugins auf Seite 342
- Modul Datei mit privatem Schlüssel auf Seite 346
- Konfigurationsdateien des One Identity Manager Service auf Seite 560

One Identity Manager Service Konfiguration für einen Jobserver anpassen

Die Konfiguration erzeugen Sie bereits bei der Installation des One Identity Manager Service. Verwenden Sie den Jobservereditor, um einzelne Konfigurationseinstellungen anzupassen. Alle Konfigurationseinstellungen können Sie auch im Programm Job Service Configuration anpassen.

HINWEIS: Bevor Sie die Konfiguration der One Identity Manager Service ändern, stellen Sie sicher, dass die Konfiguration in die Datenbank eingelesen ist. Konfigurieren und aktivieren Sie im Designer den Zeitplan **Konfigurationsdatei vom Jobserver holen und in die Jobserverkonfiguration schreiben**.

Um die Konfiguration des One Identity Manager Service auf einem Jobserver zu ändern

1. Wählen Sie im Designer die Kategorie **Basisdaten > Installationen > Jobserver**.
2. Starten Sie den Jobservereditor über die Aufgabe **Jobserver bearbeiten**.
3. Aktivieren Sie die Ansicht **One Identity Manager Service konfigurieren**.
4. Wählen Sie in der Jobserverübersicht den Jobserver zur Bearbeitung aus.
5. Bearbeiten Sie die Konfigurationseinstellungen.

   TIPP: Über die Schaltflächen und können Sie die Darstellung der Konfigurationsdaten ändern.
6. Speichern die Sie die Konfigurationsdatei über die Schaltfläche .
7. Prüfen Sie die Konfiguration über die Schaltfläche .
8. Übertragen Sie die geänderte Konfiguration an den Jobserver über den Menüeintrag **Jobserver > Konfiguration auf den Jobserver übertragen**.

   Es wird ein Prozess generiert, der die Konfigurationsdatei auf dem Jobserver aktualisiert.

TIPP: Über den Menüeintrag **Jobserver > HTTP Anfrage starten** wird für einen Jobserver der HTTP Server des One Identity Manager Service angesprochen und die verschiedenen Dienste des One Identity Manager Service werden angezeigt.

Verwandte Themen

- Konfiguration des One Identity Manager Service auf Seite 309
- Vorlage für die Konfigurationsdatei auf Seite 314
- Modultypen auswählen und Parameter bearbeiten auf Seite 315
- Gültigkeitskontrolle der Konfigurationsdatei auf Seite 316

Vorlage für die Konfigurationsdatei

HINWEIS: Die Vorlage ist nur im Programm Job Service Configuration verfügbar.

Für die Konfiguration des One Identity Manager Service mit einer direkten Datenbankverbindung wird die Vorlage **SQL Server direkt** mitgeliefert.

Diese Vorlage enthält bereits die wichtigsten Module mit ihren Einstellungen für eine einfache Konfiguration des One Identity Manager Service mit direkter Anbindung an einen SQL Server. Die Vorlage laden Sie über den Menüeintrag **Vorlagen > SQL Server direkt**. Nach dem Laden müssen Sie die Vorlage weiter anpassen.

Verwandte Themen

- One Identity Manager Service Konfiguration für einen Jobserver anpassen auf Seite 313

- Modultypen auswählen und Parameter bearbeiten auf Seite 315
- Gültigkeitskontrolle der Konfigurationsdatei auf Seite 316

Modultypen auswählen und Parameter bearbeiten

In der Modulliste der One Identity Manager Service Konfiguration sind die einzelnen Konfigurationssektionen aufgeführt. Für einige Module steht eine Auswahl von Modultypen zur Verfügung.

Um einen Modultyp auszuwählen

1. Klicken Sie auf das Modul in der Modulliste.
2. Öffnen Sie die Auswahlliste der Modultypen über **Einfügen**.
3. Wählen Sie aus der Auswahlliste den gewünschten Modultyp aus und fügen diesen per **OK** ein.

Um die Bezeichnung eines Modultyps zu ändern

1. Klicken Sie auf das Modul in der Modulliste.
2. Wählen Sie den Modultyp und klicken Sie **Umbenennen**.
3. Ändern Sie die Bezeichnung des Modultyps.
4. Drücken Sie die Eingabetaste.

Um einen Modultyp zu entfernen

1. Klicken Sie auf das Modul in der Modulliste.
2. Wählen Sie den Modultyp und klicken Sie **Entfernen**.

Um den Wert eines Parameters zu bearbeiten

1. Wählen Sie den Parameter in der Spalte **Eigenschaft** aus.
2. Klicken Sie **Bearbeiten**.

Bei Auswahl eines Eintrags in der Modulliste werden die möglichen Parameter und deren Werte angezeigt. Einige Werte können Sie direkt per Mausklick in das Eingabefeld oder die Optionsschaltfläche in der Spalte **Wert** ändern.

HINWEIS: Die Beschreibung der Parameter in den einzelnen Modulen enthält in Klammern zusätzlich die Bezeichnung der Parameter, die in der Konfigurationsdatei verwendet wird.

Es werden die folgenden Symbole verwendet.

Tabelle 100: Bedeutung der Symbole für die Parameter

Symbol	Bedeutung
	Der Wert wird als Zeichenkette übergeben.
	Pflichtangabe. Der Parameter muss angepasst werden. Der Wert wird als Zeichenkette übergeben.
	Der Wert wird als Zahl übergeben.
	Der Parameter kann aktiviert oder deaktiviert werden.
	Dieser Parameter wird im laufenden Betrieb übernommen. Ein Neustart des One Identity Manager Service ist nicht erforderlich.
	Dieser Parameter wird erst nach dem Neustart des One Identity Manager Service wirksam.

Verwandte Themen

- One Identity Manager Service Konfiguration für einen Jobserver anpassen auf Seite 313
- Vorlage für die Konfigurationsdatei auf Seite 314
- Gültigkeitskontrolle der Konfigurationsdatei auf Seite 316

Gültigkeitskontrolle der Konfigurationsdatei

In der Gültigkeitskontrolle wird überprüft, ob die Mindestanforderungen an eine Konfigurationsdatei erfüllt sind.

Um die Gültigkeitskontrolle in der One Identity Manager Service Konfiguration zu starten

- Prüfen Sie die Konfiguration über die Schaltfläche .

  Fehler und Warnungen werden in einem separaten Dialogfenster ausgegeben.

Tabelle 101: Fehlerausgabe bei Gültigkeitsprüfung

Fehler	Ausgabe als
Kein Jobprovider vorhanden.	Fehler
Kein Logwriter vorhanden.	Fehler
Pflichtfeld nicht ausgefüllt.	Fehler
Keine Jobdestination vorhanden.	Warnung
Keine Plugins vorhanden.	Warnung

Verwandte Themen

- One Identity Manager Service Konfiguration für einen Jobserver anpassen auf Seite 313
- Vorlage für die Konfigurationsdatei auf Seite 314
- Modultypen auswählen und Parameter bearbeiten auf Seite 315

Modul Prozessabholung

In diesem Modul definieren Sie den Jobprovider. Der Jobprovider stellt einer Jobdestination Prozessschritte zur Verfügung und wertet das Ergebnis aus. Als Modultypen stehen zur Auswahl:

- MSSQLJobProvider
- FileJobProvider
- FTPJobProvider
- HTTPJobProvider
- AppServerJobProvider

Innerhalb einer Konfigurationsdatei können Sie beliebig viele Jobprovider konfigurieren. Die zugehörige Konfigurationssektion wird über den Namen ermittelt. Die zugefügten Jobprovider sollten Sie dazu umbenennen.

MSSQLJobProvider

Der **MSSQLJobProvider** verarbeitet Prozessanforderungen an eine One Identity Manager-Datenbank auf einem SQL Server.

Tabelle 102: Parameter des MSSQLJobProvider

Parameter	Beschreibung
Verbindungsparameter (`ConnectString`)	Zugangsdaten zum Datenbankserver und die zu verwendende Datenbank. Für die Prozessabholung wird mindestens ein Benutzer mit der Berechtigungsebene für Konfigurationsbenutzer benötigt. Einige Funktionen können einen administrativen Benutzer erfordern. • **Server**: Datenbankserver. • (Optional) **Windows Authentifizierung**: Gibt an, ob integrierte Windows-Authentifizierung verwendet wird. Die Verwendung dieser Authentifizierung wird nicht empfohlen. Sollten Sie dieses Verfahren dennoch einsetzen, stellen Sie sicher, dass Ihre Umgebung

Parameter	Beschreibung
	Windows-Authentifizierung unterstützt. • **Nutzer**: SQL Server Anmeldename des Benutzer. • **Kennwort**: Kennwort für die SQL Server Anmeldung des Benutzer. • **Datenbank**: Wählen Sie die Datenbank.
Max. Anzahl anhängiger Anfragen (`RequestQueueLimit`)	Die Prozessanforderungen werden intern gecached. Der Parameter enthält die maximale Anzahl an Cacheeinträgen. Der Standardwert ist **1000**.
Max. Anzahl anhängiger Anfragen (`RequestQueueLimit`)	Die Verarbeitungsergebnisse werden intern gecached. Der Parameter enthält die maximale Anzahl an Cacheeinträgen. Der Standardwert ist **10000**.
Timeout für das Zurückschreiben von Results beim Shutdown in Sekunden (`FlushTimeoutSeconds`)	Der Dienst kann Resultate weiter in die Datenbank schreiben, bis dieses Timeout erreicht wurde. Die Angabe erfolgt in Sekunden. Standardwert ist **10** Sekunden.

FileJobProvider

Im **FileJobProvider** erfolgt die Datenübertragung mittels Dateien. Prozessanfragen und Ergebnisse werden in Dateien geschrieben oder aus Dateien gelesen. Diese Dateien können von der **FileJobDestination** verarbeitet werden.

Tabelle 103: Parameter des FileJobProvider

Parameter	Beschreibung
Sicherung übertragener Dateien (`BackupFiles`)	Ist diese Option aktiviert, werden alle Dateien, mit oder ohne Fehler, in ein Unterverzeichnis `Backup` verschoben. Ist die Option nicht aktiviert (Standard), werden nur Dateien mit Fehlern gesichert.
Dateiindex überprüfen (`CheckInputIndex`)	Ist diese Option aktiviert, wird kontrolliert, ob der Dateinamensindex größer wird. Dateien mit gleichem oder niedrigerem Index werden nicht verarbeitet. Standardmäßig ist diese Option nicht aktiviert.
Max. Anzahl der Prozessbäume in einer Transferdatei (`MaxListCount`)	Anzahl der Prozessschritte, die maximal zu einer Datei zusammengefasst werden. Damit kann die Dateigröße begrenzt werden.
Verschlüsselung verwenden (`UseEncryption`)	Gibt an, ob die Daten verschlüsselt in die Dateien geschrieben werden.

Parameter	Beschreibung
	HINWEIS: Die Einstellung zur Verschlüsselung muss im Jobprovider und in der zugehörigen Jobdestination gleich konfiguriert sein.
Benachrichtigungsverfahren (`EventTypes`)	Der Jobprovider unterstützt drei verschiedene Verfahren, um über neue Daten informiert zu werden. • **Timer**: Die Abfrage neuer abgelegter Daten erfolgt im festgelegten Zeitintervall. • **HTTP**: Der Provider fragt über HTTP den übergeordneten Jobserver und verarbeitet nach dessen Antwort die abgelegten Daten. • **FSEvent**: Die Abfrage neuer abgelegter Daten erfolgt nach einem Ereignis vom Dateisystem. Die Benachrichtigungsverfahren können, durch Komma getrennt, kombiniert werden. Beispiel: `TIMER,FSEVENT`
Zielrechner für HTTP Benachrichtigung (`HostName`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** geben Sie hier den Namen des Zielrechners an, zu dem die Anfragen übertragen werden.
Port für HTTP Benachrichtigung (`Port`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** tragen Sie hier den Port zur Übertragung ein.
Überwachungsintervall des Eingangsverzeichnisses (`TimerInterval`)	Bei Verwendung des Benachrichtigungsverfahren **TIMER** tragen Sie hier das Zeitintervall in Millisekunden ein.
Eingangsverzeichnis (`InputDirectory`)	Das Modul liest und verarbeitet die Prozessdateien (`*.fjg`) aus diesem Verzeichnis. HINWEIS: Stellen Sie sicher, dass der Jobprovider und die zugehörige Jobdestination das gleiche Verzeichnis verwenden. Eingangsverzeichnis und Ausgangsverzeichnis kehren sich dabei entsprechend um.
Ausgangsverzeichnis (`OutputDirectory`)	Verzeichnis in das die verarbeiteten Dateien geschrieben werden.
Unterverzeichnisse (`SubDirectories`)	Hier können Sie eine Pipe-getrennte (|) Liste von Verzeichnisnamen eintragen. Alle angegebenen Verzeichnisse werden überwacht und entsprechend verarbeitet. Es wird folgende Verzeichnisstruktur erwartet:

Parameter	Beschreibung

	`SubDirectories = "ServerA|ServerB"`
	`...`
	`Request`
	`ServerA`
	`ServerB`
	`Response`
	`ServerA`
	`ServerB`
	wobei `Request` und `Response` die in den Parametern **Eingangsverzeichnis** (`InputDirectory`) und **Ausgangsverzeichnis** (`OutputDirectory`) angegebenen Verzeichnisse sind.
	HINWEIS: Es kann nur das Benachrichtigungsverfahren **Timer** verwendet werden. Die Benachrichtigungsverfahren **HTTP** und **FSEvent** stehen nicht zur Verfügung!

Automatische Erkennung von Unterverzeichnissen (`AutoSubDirectories`)	Ist diese Option aktiviert, so verarbeitet das Modul automatisch die Dateien aller Unterverzeichnisse. Die Verarbeitung erfolgt nicht rekursiv.

Verwandte Themen

- FTPJobProvider auf Seite 320
- FileJobDestination auf Seite 327
- FTPJobDestination auf Seite 329

FTPJobProvider

Nachdem die Dateien auf dem lokalen Verzeichnis erstellt sind, verbindet sich der **FTPJobProvider** zum FTP Server und verschiebt die Dateien auf den Server. Bei einer Signalisierung wird eine Verbindung zum FTP Server erstellt und die Daten werden geholt. Auf dem FTP Server werden zwei Unterverzeichnisse `Request` und `Response` erwartet. Diese Namen sind fest definiert und können nicht geändert werden. In diesen Verzeichnissen legen die Komponenten (Jobprovider/Jobdestination) die Dateien ab oder holen sie. Der FTP Benutzer benötigt dazu die Berechtigungen zum Erstellen, Umbenennen und Löschen von Dateien.

Tabelle 104: Parameter des FTPJobProvider

Parameter	Beschreibung
Sicherung übertragener Dateien (`BackupFiles`)	Ist diese Option aktiviert, werden alle Dateien, mit oder ohne Fehler, in ein Unterverzeichnis `Backup` verschoben. Ist die Option nicht aktiviert (Standard), werden nur Dateien mit Fehlern gesichert.
Dateiindex überprüfen (`CheckInputIndex`)	Ist diese Option aktiviert, wird kontrolliert, ob der Dateinamensindex größer wird. Dateien mit gleichem oder niedrigerem Index werden nicht verarbeitet. Standardmäßig ist diese Option nicht aktiviert.
Max. Anzahl der Prozessbäume in einer Transferdatei (`MaxListCount`)	Anzahl der Prozessschritte, die maximal zu einer Datei zusammengefasst werden. Damit kann die Dateigröße begrenzt werden.
Verschlüsselung verwenden (`UseEncryption`)	Gibt an, ob die Daten verschlüsselt in die Dateien geschrieben werden. HINWEIS: Die Einstellung zur Verschlüsselung muss im Jobprovider und in der zugehörigen Jobdestination gleich konfiguriert sein.
Benachrichtigungsverfahren (`EventTypes`)	Der Jobprovider unterstützt drei verschiedene Verfahren, um über neue Daten informiert zu werden. • **Timer**: Die Abfrage neuer abgelegter Daten erfolgt im festgelegten Zeitintervall. • **HTTP**: Der Provider fragt über HTTP den übergeordneten Jobserver und verarbeitet nach dessen Antwort die abgelegten Daten. • **FSEvent**: Die Abfrage neuer abgelegter Daten erfolgt nach einem Ereignis vom Dateisystem. Die Benachrichtigungsverfahren können, durch Komma getrennt, kombiniert werden. Beispiel: `TIMER,FSEVENT`
Zielrechner für HTTP Benachrichtigung (`HostName`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** geben Sie hier den Namen des Zielrechners an, zu dem die Anfragen übertragen werden.
Port für HTTP Benachrichtigung (`Port`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** tragen Sie hier den Port zur Übertragung ein.
Überwachungsintervall des Eingangsverzeichnisses (`TimerInterval`)	Bei Verwendung des Benachrichtigungsverfahren **TIMER** tragen Sie hier das Zeitintervall in Millisekunden ein.

Parameter	Beschreibung
Eingangsverzeichnis (`InputDirectory`)	Das Modul liest und verarbeitet die Prozessdateien (`*.fjg`) aus diesem Verzeichnis. HINWEIS: Stellen Sie sicher, dass der Jobprovider und die zugehörige Jobdestination das gleiche Verzeichnis verwenden. Eingangsverzeichnis und Ausgangsverzeichnis kehren sich dabei entsprechend um.
Ausgangsverzeichnis (`OutputDirectory`)	Verzeichnis in das die verarbeiteten Dateien geschrieben werden.

Unterverzeichnisse (`SubDirectories`)	Hier können Sie eine Pipe-getrennte (|) Liste von Verzeichnisnamen eintragen. Alle angegebenen Verzeichnisse werden überwacht und entsprechend verarbeitet. Es wird folgende Verzeichnisstruktur erwartet:
	`SubDirectories = "ServerA|ServerB"`
	`...`
	`Request`
	`ServerA`
	`ServerB`
	`Response`
	`ServerA`
	`ServerB`
	wobei `Request` und `Response` die in den Parametern **Eingangsverzeichnis** (`InputDirectory`) und **Ausgangsverzeichnis** (`OutputDirectory`) angegebenen Verzeichnisse sind.
	HINWEIS: Es kann nur das Benachrichtigungsverfahren **Timer** verwendet werden. Die Benachrichtigungsverfahren **HTTP** und **FSEvent** stehen nicht zur Verfügung!

Automatische Erkennung von Unterverzeichnissen (`AutoSubDirectories`)	Ist diese Option aktiviert, so verarbeitet das Modul automatisch die Dateien aller Unterverzeichnisse. Die Verarbeitung erfolgt nicht rekursiv.
FTP Server (`FTPServer`)	Name oder die IP-Adresse des FTP Servers.
FTP Port (`FTPPort`)	Port zur FTP Übertragung. Standardport ist Port 21.
FTP Benutzerkonto (`FTPUser`)	Benutzerkonto zur FTP Anmeldung.
FTP Kennwort (`FTPPassword`)	Kennwort für das Benutzerkonto zur FTP Anmeldung.

Verwandte Themen

- FileJobProvider auf Seite 318
- FileJobDestination auf Seite 327
- FTPJobDestination auf Seite 329

HTTPJobProvider

Der **HTTPJobProvider** empfängt Prozessschritte von einem übergeordneten Jobserver. Die Datenübertragung erfolgt über HTTP.

Tabelle 105: Parameter des HTTPJobProvider

Parameter	Beschreibung
Empfänger-Port (`ParentPort`)	HTTP Port des übergeordneten Jobservers.
Empfänger-Server (`ParentServer`)	DNS-Name oder die IP-Adresse des übergeordneten Jobservers.
Wiederholversuche (`Retries`)	Anzahl der Wiederholversuche, die das Modul vornimmt, wenn die Datenübertragung scheiterte.
Wartezeit im Wiederholfall (`RetryDelay`)	Angabe, wie lange das Modul nach einer gescheiterten Prozessübertragung wartet, bis ein erneuter Versuch unternommen wird. Format der Zeitangabe: `Tage.Stunden:Minuten:Sekunden`
Zieldomäne (`RemoteDomain`)	Domäne des Benutzerkontos auf dem entfernten HTTP Server.
Zielbenutzerkonto (`RemoteUser`)	Benutzerkonto für die Anmeldung am HTTP Server.
Zielkennwort (`RemotePassword`)	Kennwort für das Benutzerkonto zur Anmeldung am HTTP Server.

Verwandte Themen

- HTTPJobDestination auf Seite 332

AppServerJobProvider

Der **AppServerJobProvider** holt die Prozessschritte vom Anwendungsserver ab und sendet diese an eine Jobdestination.

Tabelle 106: Parameter des AppServerJobProvider

Parameter	Beschreibung
Authentifizierungsdaten (`AuthenticationString`)	Wählen Sie das Authentifizierungsmodul. Abhängig vom Authentifizierungsmodul können weitere Daten, wie beispielsweise Benutzer und Kennwort erforderlich sein. Ausführliche Informationen zu den One Identity Manager Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.
Max. Anzahl anhängiger Anfragen (`RequestQueueLimit`)	Die Prozessanforderungen werden intern gecached. Der Parameter enthält die maximale Anzahl an Cacheeinträgen. Der Standardwert ist **1000**.
Max.Anzahl anhängiger Ergebnisse (`ResultQueueLimit`)	Die Verarbeitungsergebnisse werden intern gecached. Der Parameter enthält die maximale Anzahl an Cacheeinträgen. Der Standardwert ist **10000**.
Verbindungsparameter (`ConnectString`)	Webadresse (URL) des Anwendungsservers.

Modul Jobdestination

In diesem Modul definieren Sie die Jobdestination. Diese verarbeitet die Prozessschritte und liefert ein Ergebnis an den Jobprovider zurück. Als Modultypen stehen zur Auswahl:

- JobServiceDestination
- FileJobDestination
- FTPJobDestination
- HTTPJobDestination

HINWEIS: Innerhalb einer Konfigurationsdatei können Sie beliebig viele Jobdestinations konfigurieren. Die zugehörige Konfigurationssektion wird über den Namen ermittelt. Die zugefügten Jobdestinations sollten Sie dazu umbenennen.

JobServiceDestination

Das Modul **JobServiceDestination** des One Identity Manager Service übernimmt die eigentliche Abarbeitung der Prozessschritte. Eine **JobServiceDestination** fordert die Prozessschritte beim Jobprovider an, verarbeitet diese mit den Prozesskomponenten und liefert das Verarbeitungsergebnis zurück.

Tabelle 107: Parameter der JobServiceDestination

Parameter	Beschreibung
Anzahl externer Slots (`ExternalSlots`)	Anzahl der externen Prozesse (`StdioProcessor.exe`), die der One Identity Manager Service maximal zur Abarbeitung von Prozesskomponenten öffnet.
Umgebungsvariablen für externe Slots (`ExternalSlotEnvironment`)	Liste von Umgebungsvariablen, die für Prozesse der externen Slots gesetzt werden sollen. Die Variablen übergeben Sie in einer Pipe-getrennten (|) Liste. Syntax: `Variable1=Wert1|Variable2=Wert2...`
Anzahl externer 32-Bit-Slots (`ExternalSlots32`)	Anzahl der externen Prozesse im 32-Bit Speicherbereich (`StdioProcessor32.exe`), die der One Identity Manager Service maximal zur Abarbeitung von Prozesskomponenten öffnet.
Umgebungsvariablen für externe 32-Bit-Slots (`ExternalSlotEnvironment32`)	Liste von Umgebungsvariablen, die für Prozesse der externen 32-Bit-Slots gesetzt werden sollen. Die Variablen übergeben Sie in einer Pipe-getrennten (|) Liste. Syntax: `Variable1=Wert1|Variable2=Wert2...`
Anzahl der internen Slots (`InternalSlots`)	Anzahl der internen Prozesse, die der One Identity Manager Service für die interne Abarbeitung von Prozesskomponenten bereitstellt.
Datei mit privatem Schlüssel (`PrivateKey`)	Datei mit den Verschlüsselungsinformationen. Standardwert ist **private.key**. Die Schlüsseldatei muss auf allen Servern mit aktivem One Identity Manager Service im Installationsverzeichnis des Dienstes vorhanden sein. Findet der One Identity Manager Service beim Start einen privaten Schlüssel, so legt er diesen im benutzerbezogenen Schlüsselcontainer ab und löscht die Datei auf der Festplatte. Zur Erzeugung einer Schlüsseldatei und Verschlüsselung von Datenbankinformationen nutzen Sie das Programm Crypto Configuration. HINWEIS: Wenn Sie mit einer verschlüsselten One Identity Manager-Datenbank arbeiten, beachten Sie die im *One Identity Manager Installationshandbuch* aufgeführten Hinweise zum Arbeiten mit einer verschlüsselten Datenbank.
Verschlüsselungsverfahren	Eingesetztes Verschlüsselungsverfahren.

Parameter	Beschreibung
(`EncryptionScheme`)	Zulässige Werte sind: • **RSA**: RSA-Verschlüsselung mit AES für größere Daten (Standard). • **FIPSCompliantRSA**: FIPS zertifizierter RSA mit AES für größere Daten. Das Verfahren ist einzusetzen, wenn die Verschlüsselung dem FIPS 140-2 Standard entsprechen muss. Die lokale Sicherheitsrichtlinie **Use FIPS compliant algorithms for encryption, hashing, and signing** muss aktiviert sein.
ID des Jobproviders (`ProviderID`)	Werden durch den One Identity Manager Service mehrere Jobprovider abgearbeitet, tragen Sie den Namen des zu verwendenden Jobproviders ein. Ist die Angabe leer, wird der erste Jobprovider genutzt.
ID des privaten Schlüssels (`PrivateKeyId`)	ID des privaten Schlüssels. Ist keine ID angegeben, wird nach der Datei `private.key` gesucht. Verwenden Sie den Parameter, wenn Sie mit mehreren privaten Schlüsseln arbeiten, beispielsweise wenn der One Identity Manager Service Daten zwischen zwei verschlüsselten One Identity Manager-Datenbanken übertragen muss. Die privaten Schlüssel erfassen Sie im Modul **Datei mit privatem Schlüssel** Wenn der One Identity Manager nur eine verschlüsselte Datenbank verwendet, können Sie alternativ die Schlüsseldatei im Parameter **Datei mit privatem Schlüssel** (`PrivateKey`) erfassen.
Queue (`Queue`)	Bezeichnung der Queue. Mit dieser Queue-Bezeichnung werden die Prozessschritte an der Jobqueue angefordert. Für jede Queue muss ein Jobserver in der One Identity Manager-Datenbank bekannt sein.
Timeout für Prozessanfragen (`RequestTimeout`)	Gibt an, wann eine Prozessanforderung als gescheitert gilt und erneut abgeschickt wird. Format der Zeitangabe: `Tage.Stunden:Minuten:Sekunden`
Prozessanfrageintervall (`StartInterval`)	Intervall in Sekunden, nach dem der One Identity Manager Service neue Prozessschritte anfordert. Der Standardwert liegt bei **90** Sekunden. Aus den Statistikinformationen eines Jobservers werden Konfigurationsvorschläge für das Intervall ermittelt.
Zeitintervall für Statis-	Intervall in Sekunden, in welchem der One Identity

Parameter	Beschreibung
tikberechnungen (`StatisticInterval`)	Manager Service Statistikinformationen zur Verarbeitungsgeschwindigkeit an die Datenbank liefert. Der Standardwert wird auf das 4-fache des Prozessanfrageintervalls gesetzt. Aus den Statistikinformationen eines Jobservers werden Konfigurationsvorschläge für das Intervall ermittelt.
Max. Nachnutzung externer Prozessoren (`MaxExternalSlotReuse`)	Gibt an, wie oft ein externer Prozessor nach genutzt werden kann, bevor der Prozess entladen und neu gestartet wird. Der Wert **0** besagt, dass der Prozess erst entladen wird, wenn er nicht mehr benutzt wird. Der Standardwert ist **100**.

Verwandte Themen

- Einrichten von Jobservern auf Seite 289
- Eigenschaften von Jobservern auf Seite 291
- Statistikinformationen eines Jobservers auf Seite 297
- Modul Datei mit privatem Schlüssel auf Seite 346
- Ausführung externer Prozesse durch den StdioProcessor auf Seite 308

FileJobDestination

Die **FileJobDestination** verarbeitet die vom **FileJobGate** (**FileJobProvider** oder **FTPJobProvider**) zur Verfügung gestellten Prozessschritte und liefert die Ergebnisse an den Jobprovider zurück.

Tabelle 108: Parameter der FileJobDestination

Parameter	Beschreibung
Sicherung übertragener Dateien (`BackupFiles`)	Ist diese Option aktiviert, werden alle Dateien, mit oder ohne Fehler, in ein Unterverzeichnis `Backup` verschoben. Ist die Option nicht aktiviert (Standard), werden nur Dateien mit Fehlern gesichert.
Dateiindex überprüfen (`CheckInputIndex`)	Ist diese Option aktiviert, wird kontrolliert, ob der Dateinamensindex größer wird. Dateien mit gleichem oder niedrigerem Index werden nicht verarbeitet. Standardmäßig ist diese Option nicht aktiviert.
Max. Anzahl der Prozessbäume in einer Transferdatei (`MaxListCount`)	Anzahl der Prozessschritte, die maximal zu einer Datei zusammengefasst werden. Damit kann die Dateigröße begrenzt werden.

Parameter	Beschreibung
Verschlüsselung verwenden (`UseEncryption`)	Gibt an, ob die Daten verschlüsselt in die Dateien geschrieben werden. HINWEIS: Die Einstellung zur Verschlüsselung muss im Jobprovider und in der zugehörigen Jobdestination gleich konfiguriert sein.
Benachrichtigungsverfahren (`EventTypes`)	Der Jobprovider unterstützt drei verschiedene Verfahren, um über neue Daten informiert zu werden. • **Timer**: Die Abfrage neuer abgelegter Daten erfolgt im festgelegten Zeitintervall. • **HTTP**: Der Provider fragt über HTTP den übergeordneten Jobserver und verarbeitet nach dessen Antwort die abgelegten Daten. • **FSEvent**: Die Abfrage neuer abgelegter Daten erfolgt nach einem Ereignis vom Dateisystem. Die Benachrichtigungsverfahren können, durch Komma getrennt, kombiniert werden. Beispiel: **TIMER,FSEVENT**
Zielrechner für HTTP Benachrichtigung (`HostName`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** geben Sie hier den Namen des Zielrechners an, zu dem die Anfragen übertragen werden.
Port für HTTP Benachrichtigung (`Port`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** tragen Sie hier den Port zur Übertragung ein.
Überwachungsintervall des Eingangsverzeichnisses (`TimerInterval`)	Bei Verwendung des Benachrichtigungsverfahren **TIMER** tragen Sie hier das Zeitintervall in Millisekunden ein.
Eingangsverzeichnis (`InputDirectory`)	Das Modul liest und verarbeitet die Prozessdateien (`*.fjg`) aus diesem Verzeichnis. HINWEIS: Stellen Sie sicher, dass der Jobprovider und die zugehörige Jobdestination das gleiche Verzeichnis verwenden. Eingangsverzeichnis und Ausgangsverzeichnis kehren sich dabei entsprechend um.
Ausgangsverzeichnis (`OutputDirectory`)	Verzeichnis in das die verarbeiteten Dateien geschrieben werden.
Unterverzeichnisse (`SubDirectories`)	Hier können Sie eine Pipe-getrennte (|) Liste von Verzeichnisnamen eintragen. Alle angegebenen Verzeichnisse werden überwacht und entsprechend

Parameter	Beschreibung

	verarbeitet. Es wird folgende Verzeichnisstruktur erwartet:
	`SubDirectories = "ServerA|ServerB"`
	`...`
	`Request`
	`ServerA`
	`ServerB`
	`Response`
	`ServerA`
	`ServerB`
	wobei `Request` und `Response` die in den Parametern **Eingangsverzeichnis** (`InputDirectory`) und **Ausgangsverzeichnis** (`OutputDirectory`) angegebenen Verzeichnisse sind.
	HINWEIS: Es kann nur das Benachrichtigungsverfahren **Timer** verwendet werden. Die Benachrichtigungsverfahren **HTTP** und **FSEvent** stehen nicht zur Verfügung!

Automatische Erkennung von Unterverzeichnissen (`AutoSubDirectories`)	Ist diese Option aktiviert, so verarbeitet das Modul automatisch die Dateien aller Unterverzeichnisse. Die Verarbeitung erfolgt nicht rekursiv.
ID des Jobproviders (`ProviderID`)	Werden durch den One Identity Manager Service mehrere Jobprovider abgearbeitet, tragen Sie den Namen des zu verwendenden Jobproviders ein. Ist die Angabe leer, wird der erste Jobprovider genutzt.

Verwandte Themen

- FileJobProvider auf Seite 318
- FTPJobProvider auf Seite 320
- FTPJobDestination auf Seite 329

FTPJobDestination

Die **FTPJobDestination** verarbeitet die im **FileJobGate** (**FileJobProvider** oder **FTPJobProvider**) zur Verfügung gestellten Prozessschritte und liefert die Ergebnisse an den Jobprovider zurück.

Tabelle 109: Parameter der FTPJobDestination

Parameter	Beschreibung
Sicherung übertragener Dateien (`BackupFiles`)	Ist diese Option aktiviert, werden alle Dateien, mit oder ohne Fehler, in ein Unterverzeichnis `Backup` verschoben. Ist die Option nicht aktiviert (Standard), werden nur Dateien mit Fehlern gesichert.
Dateiindex überprüfen (`CheckInputIndex`)	Ist diese Option aktiviert, wird kontrolliert, ob der Dateinamensindex größer wird. Dateien mit gleichem oder niedrigerem Index werden nicht verarbeitet. Standardmäßig ist diese Option nicht aktiviert.
Max. Anzahl der Prozessbäume in einer Transferdatei (`MaxListCount`)	Anzahl der Prozessschritte, die maximal zu einer Datei zusammengefasst werden. Damit kann die Dateigröße begrenzt werden.
Verschlüsselung verwenden (`UseEncryption`)	Gibt an, ob die Daten verschlüsselt in die Dateien geschrieben werden. HINWEIS: Die Einstellung zur Verschlüsselung muss im Jobprovider und in der zugehörigen Jobdestination gleich konfiguriert sein.
Benachrichtigungsverfahren (`EventTypes`)	Der Jobprovider unterstützt drei verschiedene Verfahren, um über neue Daten informiert zu werden. • **Timer**: Die Abfrage neuer abgelegter Daten erfolgt im festgelegten Zeitintervall. • **HTTP**: Der Provider fragt über HTTP den übergeordneten Jobserver und verarbeitet nach dessen Antwort die abgelegten Daten. • **FSEvent**: Die Abfrage neuer abgelegter Daten erfolgt nach einem Ereignis vom Dateisystem. Die Benachrichtigungsverfahren können, durch Komma getrennt, kombiniert werden. Beispiel: **TIMER,FSEVENT**
Zielrechner für HTTP Benachrichtigung (`HostName`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** geben Sie hier den Namen des Zielrechners an, zu dem die Anfragen übertragen werden.
Port für HTTP Benachrichtigung (`Port`)	Bei Verwendung des Benachrichtigungsverfahren **HTTP** tragen Sie hier den Port zur Übertragung ein.
Überwachungsintervall des Eingangsverzeichnisses (TimerInterval)	Bei Verwendung des Benachrichtigungsverfahren **TIMER** tragen Sie hier das Zeitintervall in Millisekunden ein.

Parameter	Beschreibung
Eingangsverzeichnis (`InputDirectory`)	Das Modul liest und verarbeitet die Prozessdateien (`*.fjg`) aus diesem Verzeichnis. HINWEIS: Stellen Sie sicher, dass der Jobprovider und die zugehörige Jobdestination das gleiche Verzeichnis verwenden. Eingangsverzeichnis und Ausgangsverzeichnis kehren sich dabei entsprechend um.
Ausgangsverzeichnis (`OutputDirectory`)	Verzeichnis in das die verarbeiteten Dateien geschrieben werden.

Unterverzeichnisse (`SubDirectories`)	Hier können Sie eine Pipe-getrennte (|) Liste von Verzeichnisnamen eintragen. Alle angegebenen Verzeichnisse werden überwacht und entsprechend verarbeitet. Es wird folgende Verzeichnisstruktur erwartet:
	`SubDirectories = "ServerA|ServerB"`
	`...`
	`Request`
	`ServerA`
	`ServerB`
	`Response`
	`ServerA`
	`ServerB`
	wobei `Request` und `Response` die in den Parametern **Eingangsverzeichnis** (`InputDirectory`) und **Ausgangsverzeichnis** (`OutputDirectory`) angegebenen Verzeichnisse sind.
	HINWEIS: Es kann nur das Benachrichtigungsverfahren **Timer** verwendet werden. Die Benachrichtigungsverfahren **HTTP** und **FSEvent** stehen nicht zur Verfügung!

Automatische Erkennung von Unterverzeichnissen (`AutoSubDirectories`)	Ist diese Option aktiviert, so verarbeitet das Modul automatisch die Dateien aller Unterverzeichnisse. Die Verarbeitung erfolgt nicht rekursiv.
ID des Jobproviders (`ProviderID`)	Werden durch den One Identity Manager Service mehrere Jobprovider abgearbeitet, tragen Sie den Namen des zu verwendenden Jobproviders ein. Ist die Angabe leer, wird der erste Jobprovider genutzt.
FTP Server (`FTPServer`)	Name oder die IP-Adresse des FTP Servers.

Parameter	Beschreibung
FTP Port (`FTPPort`)	Port zur FTP Übertragung. Standardport ist Port 21.
FTP Benutzerkonto (`FTPUser`)	Benutzerkonto zur FTP Anmeldung.
FTP Kennwort (`FTPPassword`)	Kennwort für das Benutzerkonto zur FTP Anmeldung.

Verwandte Themen

- FileJobProvider auf Seite 318
- FTPJobProvider auf Seite 320
- FileJobDestination auf Seite 327

HTTPJobDestination

Eine **HTTPJobDestination** versendet Prozessschritte an einen nachgeordneten Jobserver. Die Datenübertragung erfolgt über HTTP.

Tabelle 110: Parameter der HTTPJobDestination

Parameter	Beschreibung
Empfänger-Port (`ChildPort`)	HTTP Port des nachgeordneten Jobservers.
ID des Jobproviders (`ProviderID`)	Werden mehrere Jobprovider abgearbeitet, tragen Sie den Namen des zu verwendenden Jobproviders ein. Ist die Angabe leer, wird der erste Jobprovider genutzt.
Wiederholversuche (`Retries`)	Anzahl der Wiederholversuche, die das Modul vornimmt, wenn die Datenübertragung scheiterte.
Wartezeit im Wiederholfall (`RetryDelay`)	Gibt an, wie lange das Modul nach einer gescheiterten Prozessübertragung wartet, bis ein erneuter Versuch unternommen wird. Format der Zeitangabe: `Tage.Stunden:Minuten:Sekunden`
Zieldomäne (`RemoteDomain`)	Domäne des Benutzerkontos auf dem entfernten HTTP Server.
Zielbenutzerkonto (`RemoteUser`)	Benutzerkonto für die Anmeldung am HTTP Server.
Zielkennwort (`RemotePassword`)	Kennwort für das Benutzerkonto zur Anmeldung am HTTP Server.

Verwandte Themen

- HTTPJobProvider auf Seite 323

Modul Konfiguration

Mit diesem Modul werden die allgemeinen Konfigurationseinstellungen des One Identity Manager Services vorgenommen.

Tabelle 111: Parameter des Moduls Konfiguration

Parameter	Beschreibung
Ausführliche Protokollierung (`VerboseLogging`)	Aktivieren Sie den Parameter, um ausführlichere Meldungen zum Starten und Stoppen des One Identity Manager Service zu erhalten.
Debugmodus (`DebugMode`)	Im Debugmodus schreibt der One Identity Manager Service zusätzliche Informationen in die Protokolldatei, zum Beispiel alle an eine Komponente übergebenen Parameter und Ergebnisse der Prozessverarbeitung. HINWEIS: Der Parameter wird zur Fehlerlokalisierung eingesetzt. Die Aktivierung des Parameters wird im normalen Betrieb aus Performancegründen nicht empfohlen.
Debugmodus der Komponenten (`ComponentDebugMode`)	Einzelne Prozesskomponenten des One Identity Manager Service schreiben zusätzliche Verarbeitungsinformationen in die Protokolldatei. HINWEIS: Der Parameter wird zur Fehlerlokalisierung eingesetzt. Die Aktivierung des Parameters wird im normalen Betrieb aus Performancegründen nicht empfohlen.
HTTP Header (`HTTPHeaders`)	HTTP Header für die Statusseite. Pipe (|)-getrennte Liste von Headern in der Form: `"name1: value1|name2: value2"`. Unterstützte Werte sind: • X-Frame-Options: SAMEORIGIN • X-Content-Type-Options: nosniff • Content-Security-Policy: default-src 'self';script-src 'self' 'unsafe-inline';style-src 'self' 'unsafe-inline';img-src 'self' data:;font-src 'self' data: • X-XSS-Protection: 1; mode=block Beispiel:

Parameter	Beschreibung
	"X-Frame-Options: SAMEORIGIN|X-Content-Type-Options:nosniff"
IP-Adresse des HTTP Servers (`HTTPAddress`)	Läuft der One Identity Manager Service auf einem Rechner mit mehreren Netzwerkkarten, so können Sie hier angeben, an welcher IP-Adresse der Service arbeiten soll. Ist keine IP-Adresse angegeben, so wird mit allen gearbeitet.
Port des HTTP Servers (`HTTPPort`)	Jeder One Identity Manager Service arbeitet automatisch als HTTP Server. Mit diesem Parameter bestimmen Sie den Port, an dem der One Identity Manager Service arbeitet. Standard ist der Port **1880**. Angesprochen wird der HTTP Server über: http://<Servername>:<Portnummer>
Protokollierung Jobprovider und ausführende Instanz (`LogDestinationAndProviderId`)	Gibt an, ob die ID des Jobproviders und der ausführenden Instanz in den Protokollmeldungen des Prozessschrittes ausgegeben werden.
Sprache (`Language`)	Sprache der Fehlermeldungen und Ausgaben aus dem One Identity Manager Service. Zulässige Werte sind **deutsch** und **english**. Standardwert ist **english**.
SSL verwenden (`UseSSL`)	Gibt an, ob der HTTP Server sichere Verbindungen bereitstellen soll. Ist die Option aktiviert, erreichen Sie den Server im Browserfrontend mittels HTTPS. Der One Identity Manager Service nutzt `System.Net.HttpListener` für die Webschnittstelle. Ausführliche Informationen wie Sie Zertifikate konfigurieren finden Sie unter How to: Configure a Port with an SSL Certificate.
Verschlüsselte Konfigurationswerte nicht zusätzlich schützen (`DoNotProtectCryptedValues`)	Normalerweise werden verschlüsselte Werte aus der `Jobservice.cfg` zusätzlich per Data Protection API geschützt. Dies verhindert die Nutzung durch andere Konten oder Server. Diese Option schaltet den zusätzlichen Schutz ab, um die Nutzung zum Beispiel auf anderen Cluster-Nodes zu ermöglichen. HINWEIS: Das Setzen dieser Option führt zu Problemen, wenn die Datenbank, gegen die der One Identity Manager Service arbeitet, nicht verschlüsselt ist. Deshalb ist sicherzustellen, dass die Verschlüsselung der Datenbank aktiviert ist.
Wartezeit bei fehlerhaftem Start (`WaitTimeOnFailedStart`)	Zeit, die nach einem fehlerhaften Start gewartet wird, bevor ein Wiederholversuch unternommen wird.

Parameter	Beschreibung
	Standardwert ist **90** Sekunden. Format der Zeitangabe: `Stunden:Minuten:Sekunden`
Wiederholversuche bei fehlgeschlagenem Start (`RetriesOnFailedStart`)	Anzahl der Wiederholversuche für den Startvorgang des One Identity Manager Service. Standardwert ist **5**.
Private Schlüssel nicht schützen (`DoNotProtectPrivateKeys`)	Wenn der One Identity Manager Service beim Start einen privaten Schlüssel im Installationsverzeichnis findet, so legt er diesen im Windows internen Schlüsselcontainer seines Dienstkontos ab und löscht die Datei auf der Festplatte. Ist diese Option aktiviert, werden die Schlüsseldateien nicht in den Schlüsselcontainer verschoben.
Schreibe die Konfiguration nicht zurück in die Datenbank (`DoNotWriteConfigBack`)	Im Standard wird die Konfiguration des Dienstes in die Datenbank übernommen. Um dies zu verhindern, aktivieren Sie die Option.

Secrets, die für die Ersetzung zulässig sind (`SecretsAllowList`)	Kommagetrennte Liste von Secret-Namen, die für eine Ersetzung in Parametern zulässig sind. Im Verzeichnis, das unter `SecretsFolder` angegeben ist, muss es eine Datei mit dem Namen des Secrets geben, die den Wert enthält.
	Syntax:
	`&SECRET(Name)&`
	Beispiel:
	`&SECRET(API_KEY)&`
	Im Verzeichnis `%SECRETS%` muss es eine Datei `API_KEY` geben, die den Wert enthält.

Secrets-Verzeichnis (`SecretsFolder`)	Pfad zum Verzeichnis, in dem die Secrets-Dateien liegen, die in Parametern genutzt werden können. Der Pfad kann Umgebungsvariablen in der Form `%Name%` enthalten. Standardwert ist **%SECRETS%**.

Modul Logwriter

Dieses Modul schreibt die Meldungen des One Identity Manager Service. Als Modultypen stehen zur Auswahl:

- EventLogLogWriter
- FileLogWriter

EventLogLogWriter

Der **EventLogLogWriter** schreibt die Meldungen des One Identity Manager Service in ein Ereignisprotokoll. Das Ereignisprotokolls können Sie beispielsweise über die Ereignisanzeige in der Microsoft Management Console anzeigen.

Tabelle 112: Parameter des EventLogLogWriter

Parameter	Beschreibung
Ereignisprotokoll (`EventLog`)	Name des Ereignisprotokolls, in das die Meldungen geschrieben werden. Mit dem Standardwert **Application** werden die Meldungen in das Anwendungsprotokoll geschrieben. HINWEIS: Wenn mehrere One Identity Manager Service auf einem Server Ereignisprotokolle schreiben, achten Sie darauf, dass die ersten 8 Buchstaben des Protokollnamens auf diesem Server eindeutig sind.
Schweregrad (`LogSeverity`)	Informationsgrad der protokollierten Meldungen. Zulässige Werte sind: • **Info**: Alle Meldungen werden in das Ereignisprotokoll geschrieben. Das Ereignisprotokoll wird schnell groß und unübersichtlich. • **Warning**: Nur Warnungen und Ausnahmefehler werden in das Ereignisprotokoll geschrieben (Standard). • **Serious**: Nur Ausnahmefehler werden in das Ereignisprotokoll geschrieben.
Ereignis-ID (`EventID`)	ID mit der Meldungen in das Ereignisprotokoll geschrieben werden.
Kategorie (`Category`)	Kategorie mit der Meldungen in das Ereignisprotokoll geschrieben werden.
Quelle (`Source`)	Bezeichnung für die Quelle mit der Meldungen in das Ereignisprotokoll geschrieben werden.

Der One Identity Manager Service protokolliert Meldungen standardmäßig nur im Ereignisprotokoll **Application**.

Um ein Ereignisprotokoll mit einem anderen Namen zu verwenden

1. Legen Sie die Datei, in die der One Identity Manager Service schreiben soll, manuell auf dem Jobserver an. Nutzen Sie dafür beispielsweise Windows PowerShell.
   a. Führen Sie auf dem Jobserver Windows PowerShell als Administrator aus.
   b. Führen Sie folgendes CmdLet aus:

```
New-EventLog -Source "Foobar" -LogName "<Dateiname>"
```

2. Geben Sie diesen Dateinamen in der Konfigurationsdatei des One Identity Manager Service im Modul **EventLogLogWriter** als Namen für das Ereignisprotokoll an.
3. Starten Sie den Computer neu.
4. Starten Sie den One Identity Manager Service neu.

Verwandte Themen

- FileLogWriter auf Seite 337

FileLogWriter

Der **FileLogWriter** schreibt die Meldungen des One Identity Manager Service in eine Protokolldatei. Die Anzeige der Protokolldatei ist über ein Browserfrontend möglich.

Der Aufruf der Protokolldatei erfolgt mit der entsprechenden URL.

http://<Servername>:<Portnummer>

Standard ist der Port 1880.

Tabelle 113: Parameter des FileLogWriter

Parameter	Beschreibung
Protokolldatei (`OutputFile`)	Name der Protokolldatei inklusive Verzeichnisangabe. In die angegebene Datei werden die Informationen des One Identity Manager Service geschrieben. WICHTIG: Es ist sicherzustellen, dass das angegebene Verzeichnis für die Datei existiert. Kann die Datei nicht erzeugt werden, ist keine Fehlerausgabe möglich. Fehlermeldungen erscheinen dann unter Windows Betriebssystemen im Ereignisprotokoll oder unter Linux Betriebssystemen in `/var/log/messages`.
Umbenennungsintervall der Protokolldatei (`LogLifeTime`)	Um unnötig große Protokolldateien zu vermeiden, unterstützt das Modul die Funktionalität zum Wechseln der Protokolldatei und einer Historie. Mit dem Intervall legen Sie die maximale Laufzeit einer Protokolldatei fest, bevor diese in das Backup umbenannt wird. Hat eine Protokolldatei ihr maximales Alter erreicht, wird die Datei umbenannt (zum Beispiel `JobService.log_20040819-083554`) und eine neue Protokolldatei wird angefangen. Format der Zeitangabe: `Tage.Stunden:Minuten:Sekunden`
Prozessschritt-Protokoll Lebensdauer	Aufbewahrungszeit für Protokolle der Prozessschritte. Nach Ablauf dieser Zeit werden die Protokolle gelöscht.

Parameter	Beschreibung

(`JobLogLifeTime`)	Format der Zeitangabe:
	`Tage.Stunden:Minuten:Sekunden`
	Im Job Queue Info können Sie zu Testzwecken die Protokollierung einzelner Prozessschritte aktivieren. Dabei werden die Verarbeitungsmeldungen des Prozessschrittes mit dem NLog Informationsgrad **Debug** in ein separates Protokoll geschrieben. Die Dateien werden im Protokollverzeichnis abgelegt.
	Ablagestruktur:
	`<Protokollverzeichnis>\JobLogs\<Erste 4 Stellen der UID_Job>\Job_<UID_Job>_<yyyymmdd>_<Timestamp>.log`

Max. Anzahl archivierter Protokolldateien (`HistorySize`)	Maximale Anzahl der Protokolldateien. Sind mehrere Protokolldateien vorhanden, wird bei der Erzeugung einer neuen Protokolldatei das älteste Backup gelöscht.
Max. Protokolldateigröße (MB) (`MaxLogSize`)	Maximale Größe der Protokolldatei in MB. Hat die Protokolldatei diese Größe erreicht, wird sie in eine Backup-Datei umbenannt und eine neue Protokolldatei wird angelegt.
Max. Länge der Parameter (`ParamMaxLength`)	Maximale Anzahl der Zeichen in einem Prozessschrittparameter, damit dieser in die Protokolldatei geschrieben wird.
Schweregrad (`LogSeverity`)	Informationsgrad der protokollierten Meldungen. Zulässige Werte sind: • **Info**: Alle Meldungen werden in das Ereignisprotokoll geschrieben. Das Ereignisprotokoll wird schnell groß und unübersichtlich. • **Warning**: Nur Warnungen und Ausnahmefehler werden in das Ereignisprotokoll geschrieben (Standard). • **Serious**: Nur Ausnahmefehler werden in das Ereignisprotokoll geschrieben.
Servernamen hinzufügen (`AddServerName`)	Gibt an, ob der Servername zu den Protokolleinträgen hinzugefügt werden soll.

Modul Dispatcher

In einer hierarchischen Serverstruktur kann ein Server als Proxyserver für andere Server dienen. Der Proxyserver fragt in einem definierten Zeitintervall nach, ob für einen Server Prozessschritte zur Verarbeitung bereitstehen und verschickt diese an den nachfolgenden

Server. Die Einrichtung eines Proxyservers empfiehlt sich, wenn die Anfragelast zur Datenbank minimiert werden soll.

Tabelle 114: Parameter des Moduls Dispatcher

Parameter	Beschreibung
Wirkt als Proxy für andere Server (`IsProxy`)	Gibt an, ob der Server als Proxyserver arbeiten soll. Für einen Server, der als Proxyserver arbeiten soll, aktivieren Sie diese Option.
Proxy-Abfrageintervall (`ProxyIntervall`)	Zeitraum in Sekunden, nachdem der Proxyserver stellvertretend für andere Server eine erneute Abfrage an die Datenbank starten soll.

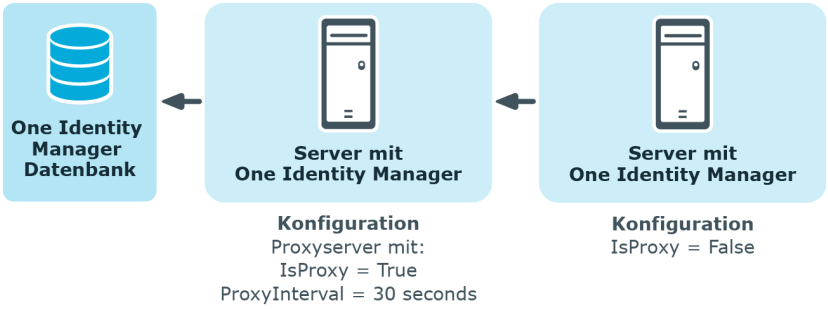
Die folgenden Richtwerte dienen zur Orientierung für die Konfiguration der Abfrageintervalle des One Identity Manager Service in einer kaskadierenden Umgebung:

Tabelle 115: Richtwerte für die Abfrageintervalle des One Identity Manager Service

Parameter	Kopfserver (direkt an die Datenbank angebunden)	Blattserver (per HTTP oder File angebunden)
JobServiceDestination.StartInterval	90 Sekunden	600 Sekunden
JobServiceDestination.StatisticInterval	360 Sekunden	600 Sekunden
Dispatcher.ProxyInterval	180 Sekunden	
Dispatcher.IsProxy	True	False

Der Proxymodus des Kopfservers sorgt dafür, dass dieser stellvertretend für die Blattserver Prozessschritte im kürzeren Proxyintervall abfragt. Beim Neustart des Kopfservers kann es zwar etwas dauern, bis alle Blattserver ihre erste Anfrage gestellt haben (in diesem Fall maximal 600 Sekunden), das System schwingt sich dann aber ein.

Abbildung 29: Konfigurationsbeispiel für einen Dispatcher



Modul Connection

Mit diesem Modul nehmen Sie spezielle Konfigurationseinstellungen zum Verhalten des One Identity Manager Service vor.

Tabelle 116: Parameter des Moduls Connection

Parameter	Beschreibung
Verzeichnis für Generierungsprotokolle (`JobGenLogDir`)	Verzeichnis der Protokolldateien, in denen die vom One Identity Manager Service erzeugten Anweisungen zur Prozessgenerierung aufgezeichnet werden.
Nachladeton unterdrücken (`NoReloadBeep`)	Mit Aktivierung des Parameters wird das akustische Signal beim Nachladen von gecachten Systemdaten abgeschaltet.
BLOB-Lese-Operationen protokollieren (`LogBlobReads`)	Gibt an, ob Lese-Operationen auf `Text`- und Binär-LOB (`BLOB`)-Felder in das SQL Protokoll geschrieben werden.
Cache-Typ (`CacheType`)	Gibt an, wie die Daten gecached werden. Der Standardwert ist **MultipleFiles**.
Nachladeintervall für Cache (`CacheReloadInterval`)	Intervall in Sekunden, nach dem der lokale Cache aktualisiert werden soll. Dieser Parameter überschreibt die Einstellung des Konfigurationsparameters **Common | CacheReload | Interval**.
Regulärer Ausdruck für Stacktrace-Positionen (`ObjectDumpStackExpression`)	Der reguläre Ausdruck legt fest, wann ein Stacktrace zusätzlich in das Objektprotokoll geschrieben werden soll. Wenn die aktuelle Zeile des Objektprotokolls auf den regulären Ausdruck passt, wird der Stacktrace mit in das Objektprotokoll geschrieben. Beispielausdruck: **"Lastname"** Enthält die aktuelle Zeile den Wert **"Lastname"**, so wird zusätzlich der Stacktrace in das Protokoll übernommen. HINWEIS: Der Parameter wird zur Fehlerlokalisierung eingesetzt. Die Aktivierung des Parameters wird im normalen Betrieb aus Performancegründen nicht empfohlen.
Fingerabdruck des Tokenzertifikats (`TokenCertificateThumbprint`)	Fingerabdruck des zu verwendenden Zertifikates zur Prüfung des Sicherheitstokens.
Tokenzertifikatdatei (`TokenCertificateFile`)	Zertifikatsdatei des zu verwendenden Zertifikates zur Prüfung des Sicherheitstokens. Das Zertifikat muss RSA-Verschlüsselung mit SHA1, SHA256 oder SHA512 unterstützen und den privaten Schlüssel enthalten.

Parameter	Beschreibung
Unterstütze Read-Only-Replikate in Azure (`SupportReadScaleOut`)	Gibt an, ob ein zweiter Pool für Read-Only-Anfragen in Azure unterstützt wird. Wenn die Option aktiviert ist, werden Read-Only-Anfragen über die Read-Only-Replikate verteilt. Diese Funktion steht in den Tarifen **Premium** und **Unternehmenskritisch** von Azure zur Verfügung. Ausführliche Informationen finden Sie unter https://docs.microsoft.com/en-us/azure/azure-sql/database/read-scale-out.
Direkte Verbindung ohne Verfügbarkeitstest (`DirectConnection`)	Gibt an, ob direkt zur Zieldatenbank verbunden werden soll, ohne vorher die Verfügbarkeit oder den Status zu testen. Dies erlaubt Tools, die keine Datenbankwechsel innerhalb der Verbindung zulassen, die Verbindung zu tracen. HINWEIS: Die Option kann Migrationen beeinflussen, da die Verbindungen immer geöffnet werden.

Modul HTTP-Authentifizierung

Jeder One Identity Manager Service arbeitet automatisch als HTTP Server. Welche Dienste der One Identity Manager Service zur Verfügung stellt, hängt von der Konfiguration der Plugins ab. Über dieses Modul legen Sie fest, wie die Authentifizierung am HTTP Server erfolgt, um auf die Dienste zuzugreifen, beispielsweise zur Anzeige der Protokolldatei oder der Statusanzeige.

Als Modultypen stehen zur Auswahl:

- BasicHttpAuthentication

  Mit diesem Authentifizierungstyp geben Sie ein bestimmtes Benutzerkonto und das zugehörige Kennwort an für den Zugriff auf den HTTP Server an.

- WindowsHttpAuthentication

  Über diesen Authentifizierungstyp legen Sie eine Active Directory Gruppe fest, deren Benutzer sich am HTTP Server authentifizieren können. Es kann entweder eine Sicherheits-ID (SID) oder der Active Directory Gruppenname in der Domäne des Jobservers angegeben werden. Befindet sich die Active Directory nicht in der Domäne des Jobservers, muss die SID verwendet werden

HINWEIS: Wenn kein Modul angegeben ist, dann ist keine Authentifizierung erforderlich. In diesem Fall können alle Benutzer auf die Dienste zugreifen.

Modul Plugins

Plugins sind Programmklassen, die der One Identity Manager Service lädt und die den Service um Funktionalitäten erweitern. Als Plugins stehen derzeit zur Verfügung:

- HTTPLogPlugin
- ScheduleCommandPlugin
- RequestWatchDogPlugin
- PerformanceCounterPlugin
- DebugMailPlugin
- ShareInfoPlugin
- RemoteConnectPlugin
- DatabaseAgentPlugin

HTTPLogPlugin

Das Plugin schreibt eine Protokolldatei, in welcher die HTTP Anfragen des One Identity Manager Service aufgezeichnet werden.

Als Parameter ist anzugeben:

- Ausgabedatei (`LogFile`)

  Tragen Sie den Namen der Datei ein, in welche die Meldungen protokolliert werden. Die Datei wird im Apache HTTP Server Combined Log Format geschrieben.

ScheduleCommandPlugin

Das Plugin ruft in regelmäßigen Abständen ein externes Programm auf. Dies ist zum Beispiel nützlich, wenn Prozessschritte über eigene Übertragungswege geleitet werden sollen.

Tabelle 117: Parameter des ScheduleCommandPlugin

Parameter	Beschreibung
Auszuführendes Kommando (`Command`)	Auszuführendes Kommando inklusive Kommandozeilenoptionen. Dieses Kommando wird in einer `cmd` ausgeführt, damit sind Build-In-Befehle möglich.
Kommando bei Servicestart (`StartCommand`)	Kommando, das beim Starten des One Identity Manager Service ausgeführt wird.

Parameter	Beschreibung
Kommando bei Servicestopp (`StopCommand`)	Kommando, das beim Stoppen des One Identity Manager Service ausgeführt wird.
Ausführungsintervall (`Interval`)	Intervall in Sekunden, in welchen Abständen der Befehl aufgerufen werden soll. Während das Kommando läuft, wird der Timer angehalten, sodass die Aufrufe sich nicht überschneiden. Der Standardwert ist **60**.
Kommandoausgabe in die Protokolldatei (`OutputToLog`)	Gibt an, ob im Erfolgsfall die Kommandoausgaben protokolliert werden. Ist der Parameter aktiviert, wird die Ausgabe des Kommandos auch im Erfolgsfall in die Protokolldatei des One Identity Manager Service geschrieben. Ist der Parameter deaktiviert, werden nur Fehler in die Protokolldatei geschrieben.
Informationsgrad (`LogSeverity`)	Meldungstypen, mit dem die Meldung im Erfolgsfall in der Protokolldatei erscheint. Zulässige Werte sind **Info**, **Warning** und **Serious**. Der Standardwert ist **Info**.

RequestWatchDogPlugin

Dieses Plugin startet den One Identity Manager Service neu, wenn innerhalb eines einstellbaren Intervalls weniger als eine definierte Anzahl von Anfragen gestellt wird.

Tabelle 118: Parameter des RequestWatchDogPlugin

Parameter	Beschreibung
Aktion (`Action`)	Aktion, die ausgeführt werden soll, wenn es zum Stillstand der Anfragen kommt. Zulässige Werte sind: • **Restart** (Standard): Meldungen werden protokolliert. Der Dienst wird neu gestartet. • **Log**: Meldungen werden protokolliert. Es erfolgt kein Neustart des Dienstes.
Überwachungsintervall (`Interval`)	Überwachungsintervall in Sekunden. Der Standardwert ist **600**.
Min. Anzahl der Anfragen (`MinRequests`)	Anzahl der Anfragen, die mindestens innerhalb des angegebenen Intervalls gestellt werden müssen. Der Standardwert ist **2**.

PerformanceCounterPlugin

Dieses Plugin exportiert Statuswerte des One Identity Manager Service als Leistungszähler. Damit ist die Überwachung mittels Systemmonitor möglich. Die Liste der aktuell verfügbaren Leistungszähler wird unter http://<servername>:1880/PerfCounter angezeigt.

Tabelle 119: Parameter des PerformanceCounterPlugin

Parameter	Beschreibung
Auszugebende Wertetypen (`CounterType`)	Wertetypen, die als Leistungszähler zur Verfügung gestellt werden. `Int`- und `Long`-Werte werden direkt ausgegeben, `TimeSpan`-Werte werden als `Long`-Wert (Anzahl Millisekunden) ausgegeben.
Polling-Intervall (`PollingInterval`)	Intervall in Sekunden, in welchem die Leistungszähler exportiert werden.
Kategorie (`Category`)	Kategorie, unter der die Leistungszähler des One Identity Manager Service angezeigt werden. Die Angabe ist erforderlich, wenn auf dem Server mehrere One Identity Manager Service mit diesem Plugin aktiv sind.

HINWEIS: Wenn nach dem Neustart des One Identity Manager Service ein Fehler der Art `Mindestens ein Dienst konnte nicht gestartet werden` auftritt, tragen Sie den WMI Performance Adapter Dienst als Abhängigkeit des One Identity Manager Service ein.

DebugMailPlugin

Ist dieses Plugin aktiv, werden die vom One Identity Manager Service generierten E-Mail Benachrichtigung nicht versendet, sondern in einem Ablageverzeichnis abgelegt. Die Dateinamen enthalten dabei den Zeitstempel. Wenn eine Mail HTML Text enthält, wird zusätzlich zur beschreibenden `*.txt`-Datei eine gleichnamige `*.html`-Datei mit dem Body abgelegt. Auch E-Mail Anhänge werden nach diesem Schema gespeichert.

HINWEIS:

- Das Plugin arbeitet nur für intern im One Identity Manager Service ausgeführte Prozesse.
- Mit Aktivierung des Plugins werden keine E-Mail Benachrichtigung durch den One Identity Manager Service versendet. Dieses Plugin dient nur zur Fehlerlokalisierung und sollte im normalen Betrieb nicht eingesetzt werden.

Als Parameter ist anzugeben:

- Ablageverzeichnis (`DropFolder`)

  Verzeichnis, in dem die E-Mail-Benachrichtigungen abgelegt werden.

ShareInfoPlugin

Dieses Plugin wird zur Auflösung von Samba-Freigaben (`smb.conf`) unter Linux Betriebssystemen benötigt. Das Plugin löst UNC-Pfade in lokale Pfade auf. Dieses Plugin benötigt keine weiteren Parameter.

HINWEIS: Installieren Sie das Plugin, wenn der One Identity Manager Service Kopieraktionen zwischen Servern mit Linux Betriebssystemen ausführt.

RemoteConnectPlugin

Um die Synchronisation mit einem Zielsystem zu konfigurieren, muss der One Identity Manager Daten aus dem Zielsystem auslesen. Dabei kommuniziert der One Identity Manager direkt mit dem Zielsystem. Mitunter ist der direkte Zugriff von der Arbeitsstation, auf welcher der Synchronization Editor installiert ist, nicht möglich, beispielsweise aufgrund der Firewall-Konfiguration oder weil die Arbeitsstation nicht die notwendigen Hard- oder Softwarevoraussetzungen erfüllt. Wenn der direkte Zugriff von der Arbeitsstation nicht möglich ist, kann eine Remoteverbindung eingerichtet werden. Voraussetzung dafür ist, dass auf dem Jobserver das **RemoteConnectPlugin** installiert ist.

Tabelle 120: Parameter des RemoteConnectPlugins

Parameter	Wert	Beschreibung
Authentifizierungsverfahren (`AuthenticationMethod`)	ADGroup	Verfahren, mit dem eingehende Anfragen authentifiziert werden sollen. Zulässige Werte: **ADGroup**
Erlaubte AD Gruppe (`ADGroupAuthPermittedGroup`)		Definierter Name oder Objekt SID der Active Directory Gruppe, deren Mitglieder berechtigt sind, eine Remoteverbindung zu nutzen. Dieser Parameter wird nur für das Authentifizierungsverfahren **ADGroup** benötigt.
Port (`Port`)	2880	Port, über den der Server erreichbar ist.

HINWEIS: Die Authentifizierung der Remoteverbindung ist nur über eine Active Directory Gruppe möglich!

DatabaseAgentPlugin

Durch dieses One Identity Manager Service Plugin wird der Database Agent Service bereitgestellt. Der Database Agent Service steuert die Verarbeitung der DBQueue Prozessor Aufträge. Das Plugin sollte auf dem Jobserver konfiguriert sein, der die Serverfunktion **Aktualisierungsserver** übernimmt. Für die Datenbankverbindung im Jobprovider muss ein administrativer Benutzer verwendet werden.

Als Parameter ist anzugeben:

- IDs der Jobprovider (`ProviderIDs`)

  IDs der zu nutzenden Jobprovider. Einzutragen ist eine mit dem Pipe-Symbol (|) getrennte Liste der Namen der Jobprovider. Ist die Angabe leer, wird der erste Jobprovider genutzt. Bei Angabe von * werden alle Jobprovider genutzt.

Verwandte Themen

- MSSQLJobProvider auf Seite 317
- Verarbeitung der DBQueue Prozessor Aufträge durch den Database Agent Service auf Seite 553

Modul Datei mit privatem Schlüssel

In diesem Modul erfassen Sie die Informationen zu Dateien mit einem privaten Schlüssel. Verwenden Sie das Modul, wenn Sie mit mehreren privaten Schlüsseln arbeiten, beispielsweise wenn der One Identity Manager Service Daten zwischen zwei verschlüsselten One Identity Manager-Datenbanken übertragen muss.

Ist hier kein Schlüssel eintragen, wird die privaten Schlüsseldatei aus dem Parameter **Datei mit privatem Schlüssel** (`PrivateKey`) der **JobServiceDestination** verwendet.

Um eine Datei mit einem privaten Schlüssel zu erfassen

1. Klicken Sie **Neu** und erfassen Sie folgende Informationen:
   - **Eigenschaft**: Tragen Sie die ID des privaten Schlüssels ein. Die ID wird in der **JobServiceDestination** im Parameter **ID des privaten Schlüssels** (`PrivateKeyId`) erwartet. Der Standardschlüssel hat die ID **Default**.
   - **Wert**: Erfassen Sie den Pfad zur privaten Schlüsseldatei. Die Pfadangabe kann absolut oder relativ zum One Identity Manager Service erfolgen.

Beispiel: Konfiguration in der Datei Jobservice.cfg

```
configuration>
     <category name="privatekeys">
          <value name="Default">private.key</value>
          <value name="Key2">key2.key</value>
          <value name="OtherKey">C:\Path\To\Other.key</value>
     </category>
</configuration>
```

Verwandte Themen

- JobServiceDestination auf Seite 324

10

Prozessüberwachung zur Nachverfolgung von Änderungen

Um der Berichtspflicht gegenüber internen Gremien und der Erfüllung der Nachweispflicht aufgrund gesetzlicher Auflagen nachzukommen, bietet der One Identity Manager die Möglichkeit den Änderungsverlauf an Objekten und deren Eigenschaften zu protokollieren. Innerhalb des One Identity Manager können unterschiedliche Verfahren zur Nachverfolgung von Änderungen genutzt werden. Über die Kombination der Verfahren können alle Änderungen im One Identity Manager nachvollzogen werden.

- Aufzeichnung von Datenänderungen

  Die Aufzeichnung von Datenänderungen kann für Neuanlagen oder Löschoperationen von Objekten bis hin zu Änderungen einzelner Objekteigenschaften erfolgen.

- Aufzeichnung von Prozessinformationen

  Die Aufzeichnung der Prozessinformationen erlaubt die Nachverfolgung aller Prozesse und Prozessschritte bei deren Verarbeitung durch den One Identity Manager Service.

- Aufzeichnung von Meldungen in der Prozesshistorie

  In der Prozesshistorie werden Erfolgs- und Fehlermeldungen zur Verarbeitung der einzelnen Prozessschritte in der Jobqueue durch den One Identity Manager Service aufgezeichnet.

Alle im One Identity Manager protokollierten Aufzeichnungen werden zunächst in der One Identity Manager-Datenbank gespeichert. Der Anteil der historisierten Daten am Gesamtvolumen einer One Identity Manager-Datenbank sollte maximal 25 % betragen. Anderenfalls kann es zu Performance-Problemen kommen. Die Aufzeichnungen sollten in regelmäßigen Abständen aus der One Identity Manager-Datenbank entfernt und archiviert werden. Ausführliche Informationen zur Archivierung von Daten finden Sie im *One Identity Manager Administrationshandbuch für die Datenarchivierung*.

Detaillierte Informationen zum Thema

- Grundlagen zur Prozessüberwachung auf Seite 349
- Aufzeichnen von Datenänderungen auf Seite 350
- Aufzeichnen von Prozessinformationen in der Prozessverarbeitung auf Seite 352

- Aufzeichnung von Meldungen in der Prozesshistorie auf Seite 357
- Archivieren und Löschen von Aufzeichnungen auf Seite 364

Grundlagen zur Prozessüberwachung

Um die Prozessüberwachung im One Identity Manager zu nutzen

1. Prüfen Sie im Designer, ob der Konfigurationsparameter aktiviert **Common | ProcessState** ist. Anderenfalls aktivieren Sie den Konfigurationsparameter.

   Ist der Konfigurationsparameter aktiviert, können Sie die Prozessüberwachung konfigurieren. Zusätzlich wird im Manager die Prozessansicht aktiviert.

2. Den Umfang der Aufzeichnungen steuern Sie über die Konfiguration der einzelnen Verfahren.

Die vom One Identity Manager eingesetzten Verfahren erlauben die Beobachtung aller im System ausgeführten Änderungen, die durch eine Benutzeraktion ausgelöst werden. Jede Aktion im One Identity Manager wird durch eine unikale ID-Nummer gekennzeichnet. Diese ID-Nummer wird als GenProcID bezeichnet. Alle Änderungen, die auf eine Ursache zurückzuführen sind, erhalten dieselbe GenProcID und werden auf diese Art zusammengefasst. Wird keine GenProcID von einer vorgelagerten Aktion an die aktuelle Aktion übergeben, so wird automatisch eine neue ID gebildet.

Wird eine Aktion über die Objektschicht des One Identity Manager ausgelöst, so wird die GenProcID in die Kontextinformationen der Datenbankverbindung geschrieben. Der angemeldete Benutzer wird ebenfalls in den Kontextinformationen vermerkt und steht somit zur Verfügung.

Wird eine Aktion direkt auf der Datenbank oder über eine Anwendung, die ohne die One Identity Manager-Objektschicht arbeitet, vorgenommen, so wird durch die Trigger eine neue GenProcID erzeugt. Diese GenProcID ist für die Dauer der Datenbankverbindung gültig, das bedeutet alle Änderungen gehören zur selben Aktion und verweisen somit auf dieselbe GenProcID. Die Benutzerinformationen werden aus dem Namen des Datenbankbenutzers, der MAC-Adresse und dem Namen der Arbeitsstation sowie dem Namen der Anwendung gebildet.

Alle Aktionen (Prozessauslöser), die eine Änderung am System auslösen, und deren aktuelle Statusinformationen werden intern in der Statustabelle `DialogProcess` protokolliert. Diese Protokollierung erfolgt unabhängig vom eingesetzten Verfahren zur Änderungsverfolgung. Somit ist ein Einstiegspunkt für die Beobachtung gegeben und die Zusammenfassung der Änderungen, die auf einer Aktion beruhen, möglich.

Zu einer Aktion werden die folgenden Informationen aufgezeichnet:

- ID-Nummer (GenProcID)
- Anzeigename für die ausgeführte Aktion
- Basisobjekt, für dass die Aktion ausgelöst wurde
- Benutzer, der die Aktion ausgelöst hat

- Zeitpunkt, an dem die Aktion ausgelöst wurde
- Objektschlüssel zur Selektion des Prozessauslösers
- Kommentar zur Aktion
- aktueller Verarbeitungsstatus

HINWEIS: Die Darstellung der Informationen erfolgt im Manager in der Prozessansicht. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Detaillierte Informationen zum Thema

- Aufzeichnen von Datenänderungen auf Seite 350
- Aufzeichnen von Prozessinformationen in der Prozessverarbeitung auf Seite 352

Aufzeichnen von Datenänderungen

HINWEIS: Die Darstellung der Informationen erfolgt im Manager in der Prozessansicht. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Um Datenänderungen aufzuzeichnen

- Prüfen Sie im Designer, ob der Konfigurationsparameter **Common | ProcessState** aktiviert ist. Anderenfalls aktivieren Sie den Konfigurationsparameter.
- Aktivieren Sie im Designer den Konfigurationsparameter **Common | ProcessState | PropertyLog** und kompilieren Sie die Datenbank.

  Ist der Konfigurationsparameters aktiviert, werden Änderungen einzelner Werte aufgezeichnet und im Manager in der Prozessansicht angezeigt.

  Wenn Sie den Konfigurationsparameter zu einem späteren Zeitpunkt deaktivieren, werden die nicht benötigten Modellbestandteile und Skripte deaktiviert. SQL Prozeduren und Trigger werden weiterhin ausgeführt. Ausführliche Informationen zum Verhalten präprozessorrelevanter Konfigurationsparameter und zur bedingten Kompilierung finden Sie unter Bedingte Kompilierung mittels Präprozessorbedingungen auf Seite 369.
- (Optional) Um für den Systemanteil Änderungen an Eigenschaften aufzuzeichnen, die zu einem alternativen Schlüssel gehört, aktivieren Sie im Designer den Konfigurationsparameter **Common | ProcessState | PropertyLog | AutoTrackAlternatePK**.
- (Optional) Um für den Nutzdatenanteil Änderungen an Eigenschaften aufzuzeichnen, die zu einem alternativen Schlüssel gehört, aktivieren Sie im Designer den Konfigurationsparameter **Common | ProcessState | PropertyLog | AutoTrackAlternatePK | PayLoad**.
- Kennzeichnen Sie die Spalten, für welche Änderungen aufzuzeichnen sind.

- Kennzeichnen Sie die Spalten, die beim Löschen eines Objektes aufzuzeichnen sind.

  TIPP: Wenn Sie im Designer den Konfigurationsparameter **Common | ProcessState | PropertyLog | AllDefaultPropertiesForModel** aktivieren, werden Spalten des One Identity Manager Schemas bereits zum Aufzeichnen von Änderungen und zum Aufzeichnen beim Löschen gekennzeichnet. Welche Spalten betroffen sind, ist in der Tabelle `QBMVDefaultHistoryColumns` definiert.

Die Aufzeichnung kann für Neuanlagen, Änderungen oder Löschoperationen auf Objekten erfolgen. Für jede Aufzeichnung wird die auslösende GenProcID mitgeführt, somit können die Änderungen eines Objektes zusammengefasst werden. Die Aufzeichnungen der Datenänderungen werden in den Tabellen `DialogWatchOperation` und `DialogWatchProperty` abgelegt. Zusätzlich wird für die auslösende Aktion ein Eintrag in der Statustabelle `DialogProcess` erzeugt.

Für die einzelnen Operationen werden die folgenden Informationen erfasst:

- Anlegen eines Objektes

  Bei Neuanlage eines Objektes werden der Objektschlüssel, der Anzeigename des Objektes, das Erstellungsdatum und der Benutzer aufgezeichnet.

- Ändern eines Objektes

  Bei Änderung einer Spalte werden der alte Wert, das Änderungsdatum und der Benutzer aufgezeichnet. Abhängig von den Konfigurationsparametern **Common | ProcessState | PropertyLog | AutoTrackAlternatePK** und **Common | ProcessState | PropertyLog | AutoTrackAlternatePK | PayLoad** werden Änderungen an Eigenschaften, die zu einem alternativen Schlüssel gehören, aufgezeichnet.

- Löschen eines Objektes

  Beim Löschen eines Objektes werden die beim Löschen aufzuzeichnenden Spalten sowie alle Primärschlüsselspalten aufgezeichnet. Es werden der Wert, das Löschdatum und der Benutzer aufgezeichnet.

Verwandte Themen

- Spalten für die Aufzeichnung von Datenänderungen kennzeichnen auf Seite 351
- Grundlagen zur Prozessüberwachung auf Seite 349
- Aufzeichnen von Prozessinformationen in der Prozessverarbeitung auf Seite 352

Spalten für die Aufzeichnung von Datenänderungen kennzeichnen

TIPP: Wenn Sie im Designer den Konfigurationsparameter **Common | ProcessState | PropertyLog | AllDefaultPropertiesForModel** aktivieren, werden Spalten des One Identity Manager Schemas bereits zum Aufzeichnen von Änderungen und zum Aufzeichnen beim Löschen gekennzeichnet. Welche Spalten betroffen sind, ist in der

Datenbanksicht `QBMVDefaultHistoryColumns` definiert.

Um eine Spalte für die Aufzeichnung zu kennzeichnen

1. Wählen Sie im Designer die Kategorie **One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Wählen Sie die Spalte und wählen Sie die Ansicht **Spalteneigenschaften**.
4. Wählen Sie den Tabreiter **Sonstiges** und bearbeiten Sie folgende Eigenschaften.
    - **Aufzeichnen von Änderungen**: Um Datenänderungen der Spalte aufzuzeichnen, aktivieren Sie die Option.
    - **Aufzeichnen beim Löschen**: Um die Spalte beim Löschen des Objektes aufzuzeichnen, aktivieren Sie die Option.

Verwandte Themen

- Aufzeichnen von Datenänderungen auf Seite 350
- Eigenschaften von Spaltendefinitionen auf Seite 99

Aufzeichnen von Prozessinformationen in der Prozessverarbeitung

HINWEIS: Die Darstellung der Informationen erfolgt im Manager in der Prozessansicht. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Um Prozessinformationen aufzuzeichnen

- Prüfen Sie im Designer, ob der Konfigurationsparameter **Common | ProcessState** aktiviert ist. Anderenfalls aktivieren Sie den Konfigurationsparameter.
- Prüfen Sie im Designer, ob der Konfigurationsparameter **Common | ProcessState | ProgressView** aktiviert ist. Anderenfalls aktivieren Sie den Konfigurationsparameter. Wählen Sie über die Option des Konfigurationsparameters den Umfang der Aufzeichnungen.

  Zulässige Werte sind:

    - **1**: Komplette Prozessverfolgung. Es werden Prozessinformationen aller Prozesse aufgezeichnet, die für die Prozessverfolgung markiert sind.
    - **2**: Web Portal Verfolgung. Es werden nur Prozessinformationen der Prozesse aufgezeichnet, die für die Prozessverfolgung im Web Portal markiert sind. (Standard).
- Kennzeichnen Sie die Prozesse und die Prozessschritte für die Prozessverfolgung und definieren Sie die Vorlagen für die Prozessinformationen für Ereignisse, Prozesse und

Prozessschritte.

Die Vorlagen für die Erzeugung von Prozessinformationen für Prozesse, Prozessschritte und Ereignisse richten Sie im Designer mit dem Prozesseditor ein. Für die sprachabhängige Definition der Prozessinformationen verwenden Sie die #LD-Notation.

Ist der Konfigurationsparameter **Common | ProcessState | ProgressView** aktiviert, erzeugt der Jobgenerator bei der Prozessgenerierung für Prozesse, Prozessschritte und Ereignisse mit Prozessinformationen Einträge in den Statustabellen.

Der Jobgenerator nutzt beim Start die GenProcID für den auszuführenden Generierungsvorgang. Wird keine GenProcID per Laufzeit übergeben, so wird automatisch eine neue GenProcID gebildet. Diese ID wird vor dem Generierungsprozess auf die globale Variable **GenProcID** des aktuellen Datenbankverbindungsobjektes geschrieben und ist somit in allen Prozessen verwendbar. Alle Teilschritte, die über einen Generierungsvorgang ausgelöst wurden, werden auf diese Art zusammengefasst und aufgezeichnet. Eine Ausnahme bilden Massenoperationen wie Synchronisationen und CSV-Importe. Hier wird zur Nachverfolgung von Änderungen an den einzelnen Objekten, anstelle einer GenProcID für den gesamten Prozess, für jeden Einzelschritt eine neue GenProcID erzeugt.

Für jeden zur Prozessverfolgung markierten Prozessschritt wird ein Eintrag in der Statustabelle `DialogProcessStep` erstellt. Für jeden Prozess, in welchem mindestens ein solcher Prozessschritt gefunden wird, erfolgt ein Eintrag in die Statustabelle `DialogProcessChain`. Für jeden Generierungsvorgang, der einen Eintrag in die Statustabelle `DialogProcessChain` veranlasst hat, wird ein Eintrag in die Statustabelle `DialogProcess` geschrieben. Dabei bildet der Jobgenerator den Anzeigenamen für die Prozessansicht durch Ausführung des angegebenen VB.Net-Ausdruckes für die Prozessinformationen.

Die möglichen Verarbeitungszustände und die verfügbaren Zusatzinformationen zum jeweiligen Verarbeitungsstatus sind in den folgenden Tabellen aufgeführt.

Tabelle 121: Mögliche Prozesszustände

Prozesszustand	Beschreibung
Initial	<generated> ::= "G"
Ende der Bearbeitung	<finalstate> ::= <ended> | <failed> | <not executed> wobei: <ended> ::= "E" (Verarbeitung korrekt beendet) <failed> ::= "F" (Verarbeitung fehlerhaft beendet) <not executed> ::= "N" (Im Verlauf der Verarbeitung nicht mehr erreichbar)
In Bearbeitung	<workingstate> ::= <delayed> | <processing> [<ProcessStateAddON>] wobei: <delayed> ::= "D" (Zur Verarbeitung zurückgestellt) <Long delayed>::="L" (Verarbeitung wurde eingefroren)

Prozesszustand	Beschreibung
	<processing> ::= "P" (In Verarbeitung) <ProcessStateAddON> (optionale Zusatzinformationen)

Tabelle 122: Mögliche Zusatzinformationen

Zusatzinformationen	Beschreibung
Verarbeitung zurückgestellt bis	<datetime> ::= <YYYY> - <MM> - <DD> <HH> : <NN> : <SS> wobei: <YYYY> ::= 1980..9999 <MM> ::= 01..12 <DD> ::= 01..31 <HH> ::= 00..23<NN> ::= 00..59 <SS> ::= 00..59
Wiederholversuche	<retryinfo> ::= 1..99

Verwandte Themen

- Prozessinformationen für Prozesse bearbeiten auf Seite 354
- Prozessinformationen für Prozessschritte bearbeiten auf Seite 355
- Prozessinformationen für Ereignisse bearbeiten auf Seite 356
- Grundlagen zur Prozessüberwachung auf Seite 349
- Aufzeichnen von Datenänderungen auf Seite 350

Prozessinformationen für Prozesse bearbeiten

Um Prozessinformationen für einen Prozess zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess aus.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozess.
4. Bearbeiten Sie in der Ansicht **Prozesseigenschaften** auf dem Tabreiter **Allgemein** folgende Information.
    - **Prozessinformation**: Wählen Sie wie die Aufzeichnung der Prozessinformation erfolgen soll.

Zulässige Werte sind:

- **Keine**: Die Prozessinformationen werden nicht aufgezeichnet.
- **Komplette Prozessverfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager angezeigt.
- **Web Portal Verfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager und Web Portal angezeigt.

5. Erfassen Sie in der Ansicht **Prozesseigenschaften** auf dem Tabreiter **Prozessverfolgung** folgende Information.
   - **Prozessinformation:** Bildungsvorschrift für die Prozessinformationen als VB.Net-Ausdruck. Für die sprachabhängige Definition der Prozessinformationen verwenden Sie die #LD-Notation.

Verwandte Themen

- Verwendung der #LD-Notation auf Seite 389
- Eigenschaften für Prozesse auf Seite 254
- Prozessinformationen für Prozessschritte bearbeiten auf Seite 355
- Prozessinformationen für Ereignisse bearbeiten auf Seite 356

Prozessinformationen für Prozessschritte bearbeiten

Um Prozessinformationen für einen Prozessschritt zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess aus.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozessschritt.
4. Bearbeiten Sie in der Ansicht **Prozessschritteigenschaften** auf dem Tabreiter **Allgemein** folgende Information.
   - **Prozessinformation**: Wählen Sie wie die Aufzeichnung der Prozessinformation erfolgen soll.

     Zulässige Werte sind:

     - **Keine**: Die Prozessinformationen werden nicht aufgezeichnet.
     - **Komplette Prozessverfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager angezeigt.
     - **Web Portal Verfolgung**: Die Prozessinformationen werden aufgezeichnet und werden im Manager und Web Portal angezeigt.
5. Erfassen Sie in der Ansicht **Prozessschritteigenschaften** auf dem Tabreiter **Prozessverfolgung** folgende Informationen.

- **Detailtiefe**: Wählen Sie die Detailtiefe der Prozessinformationen. Zur Auswahl stehen **grundlegende Informationen**, **erweiterte Informationen** und **vollständige Informationen**.

  Über die Detailtiefe wird die Darstellung der Prozessinformationenim Manager in der Prozessansicht geregelt. Abhängig von den Programmeinstellungen des Manager, werden dem Benutzer unterschiedliche detaillierte Sichten auf die Prozessinformationen zur Verfügung gestellt. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.
- **Prozessinformation**: Erfassen Sie die Bildungsvorschrift für die Prozessinformationen als VB.Net-Ausdruck. Für die sprachabhängige Definition der Prozessinformationen verwenden Sie die #LD-Notation.

Verwandte Themen

- Verwendung der #LD-Notation auf Seite 389
- Eigenschaften für Prozessschritte auf Seite 256
- Prozessinformationen für Prozesse bearbeiten auf Seite 354
- Prozessinformationen für Ereignisse bearbeiten auf Seite 356

Prozessinformationen für Ereignisse bearbeiten

WICHTIG: Um Prozessinformationen für Ereignisse zu generieren, muss mindestens ein Prozess des Ereignisses für die Prozessverfolgung aktiviert sein.

Um Prozessinformationen für Ereignisse zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Prozess-Orchestrierung** den Prozess aus.
2. Starten Sie den Prozesseditor über die Aufgabe **Prozess bearbeiten**.
3. Klicken Sie im Prozessdokument auf das Element für den Prozess.
4. Wählen Sie in der Ansicht **Ereignisse** das Ereignis und klicken Sie .
5. Erfassen Sie folgende Information.
   - **Prozessinformation für Ereignis**: Bildungsvorschrift für die Prozessinformationen als VB.Net-Ausdruck. Für die sprachabhängige Definition der Prozessinformationen verwenden Sie die #LD-Notation.

     Ist keine Bildungsvorschrift hinterlegt, werden die Prozessinformationen folgendermaßen ermittelt:

     ```
     <Tabelle> - <Ereignis> - <Anzeigename des Objektes>
     ```

Weisen mehrere Prozesse auf ein Ereignis, wird das Ereignis mit der niedrigsten Generierungsreihenfolge ermittelt, für das eine Bildungsvorschrift für die Prozessinformation hinterlegt ist. Diese Bildungsvorschrift wird ausgewertet und im

Manager in der Prozessansicht dargestellt. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Verwandte Themen

- Verwendung der #LD-Notation auf Seite 389
- Ereignisse für Prozesse erstellen auf Seite 265
- Prozessinformationen für Prozesse bearbeiten auf Seite 354
- Prozessinformationen für Prozessschritte bearbeiten auf Seite 355

Aufzeichnung von Meldungen in der Prozesshistorie

In der Prozesshistorie (Tabelle `JobHistory`) werden die verarbeiteten Prozesse aufgezeichnet. Die Prozesshistorie können Sie im Job Queue Info analysieren. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.

Um Meldungen in der Prozesshistorie aufzuzeichnen

- Prüfen Sie im Designer, ob der Konfigurationsparameter **Common | ProcessState** aktiviert ist. Anderenfalls aktivieren Sie den Konfigurationsparameter.
- Prüfen Sie im Designer, ob der Konfigurationsparameter **Common | ProcessState | JobHistory** aktiviert ist. Anderenfalls aktivieren Sie den Konfigurationsparameter. Wählen Sie über die Option des Konfigurationsparameters den Umfang der Aufzeichnungen.

Tabelle 123: Zulässige Werte des Konfigurationsparameters Common | ProcessState | JobHistory

Wert	Bedeutung
NO	Es werden keine Meldungen in der Prozesshistorie aufgezeichnet.
ALL	Alle verarbeiteten Prozessschritte werden in Prozesshistorie aufgezeichnet.
ERROR	Es werden nur fehlerhafte Prozessschritte in Prozesshistorie aufgezeichnet.
ERRORorSELECTED	Fehlerhafte Prozessschritte und Prozessschritte, die mit der Option **Prozesshistorie** gekennzeichnet sind, werden in der Prozesshistorie aufgezeichnet.

Wert	Bedeutung
SELECTED	Nur Prozessschritte, die mit der Option **Prozesshistorie** gekennzeichnet sind, werden in der Prozesshistorie aufgezeichnet.

- Legen Sie über den Konfigurationsparameter **Common | ProcessState | JobHistory | TrimLongParameters** fest, wie lang der Wert von Prozessparametern sein darf, die in der Prozesshistorie aufgezeichnet werden.

Die Aufzeichnungen in der Prozesshistorie sollten in regelmäßigen Abständen aus der One Identity Manager-Datenbank entfernt und archiviert werden. Der One Identity Manager bietet dazu verschiedene Verfahren an. Weitere Informationen finden Sie unter Archivieren und Löschen von Aufzeichnungen auf Seite 364.

Prozessverfolgung für Operationen des DBQueue Prozessor

Um Vererbungsberechnungen als Folge einer Änderung am System zu verfolgen, wird die GenProcID in den Operationen des DBQueue Prozessor mitgeführt. Für eventuelle Folgeoperationen muss sichergestellt sein, dass es je Operation und Objekt nur einen Eintrag in der DBQueue gibt. Um derartige Prozesse abzubilden, wird für diese eine neue GenProcID vergeben und in weiteren Prozessen verwendet. Die kollidierten Prozesse und ihre GenProcID's werden in der Tabelle `DialogProcessSubstitute` abgebildet.

Bei der Erzeugung neuer GenProcID für kollidierte Prozesse gelten folgende Regeln:

- Mehrere gleiche Operationen des DBQueue Prozessor auf einem Objekt werden zu einem Prozess (einer GenProcID) zusammengeführt. Dabei werden vorhandene Ersatzprozesse genutzt, wenn die Menge ihrer Vorgänger, bezogen auf Basisprozesse, identisch ist.
- Sollte es in der Folge zu weiteren Kollisionen kommen, so werden schon ersetzte GenProcID zuerst in ihre Ursprünge aufgelöst und dann ein neuer Ersatz gebildet.
- Ein Ersatz gilt genau für eine Menge an Ursprungsprozessen.

Der Konfigurationsparameter **QBM | DBQueue | GenProcIDReplaceLimit** definiert den Grenzwert für Prozessersetzungen. Diese Maximalzahl von kollidierenden Prozessen wird in der Tabelle `DialogProcessSubstitute` abgebildet. Bei Bedarf können Sie den Konfigurationsparameter im Designer aktivieren und den Wert anpassen.

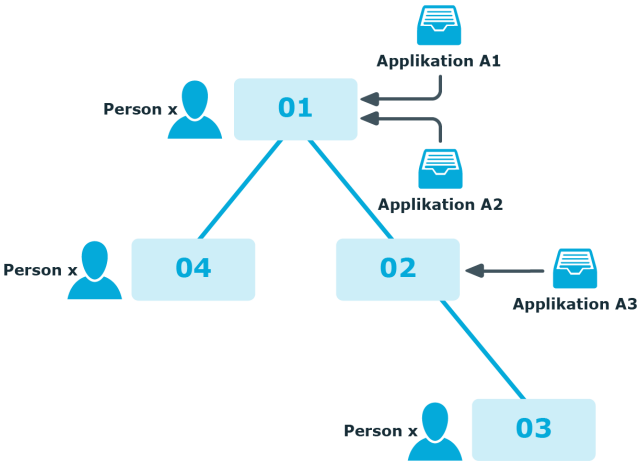
Verwandte Themen

- Beispiel für die Ersetzung der GenProcID auf Seite 359
- Verarbeitung von DBQueue Aufträgen auf Seite 547

Beispiel für die Ersetzung der GenProcID

Es existiert eine hierarchische Rollenstruktur bestehend aus 4 Rollen O1, O2, O3 und O4. Die Person X ist den Rollen O1, O4 und O3 zugeordnet. Nachfolgend wird die Zuweisung von Software an die Rollen betrachtet.

Abbildung 30: Abbildung der Rollenstruktur laut Beispiel



Zwischen zwei Ausführungen des DBQueue Prozessor laufen drei Prozesse mit jeweils eigener GenProcID an der Oberfläche ab:

- P1: Zuweisung Software A1 an die Rolle O1
- P2: Zuweisung Software A2 an die Rolle O1
- P3: Zuweisung Software A3 an die Rolle O2

Im Ergebnis stehen folgende Operationen in der DBQueue (Tabelle `DialogDBQueue`) und der Prozessinformation:

Operation	Objekt	GenProcID
OrgHasApp	O1	P1
OrgHasApp	O1	P2
OrgHasApp	O2	P3

Die Operation `OrgHasApp` bezüglich O1 kann nicht aufgeteilt werden, da für O1 die Vereinigungsmenge der Software errechnet wird. Zu diesem Zeitpunkt ist auch keine Information mehr verfügbar, welche GenProcID durch die Zuordnung welcher Software eingetragen wurde.

Um Eindeutigkeit bezüglich der Kombination Operation und Objekt zu erreichen, wird eine neue GenProcID P4 eingeführt und die beiden Operationen bezüglich O1 auf diese verdichtet. In der Tabelle `DialogProcessSubstitute` wird vermerkt, dass P4 die möglichen (aber in der Einzelaktion nicht eindeutigen) Vorgänger P1 und P2 hat.

Operation	Objekt	GenProcID
OrgHasApp	O1	P4
OrgHasApp	O2	P3

Abhängig, ob `OrgHasApp` eine Operation ist, die im Einzelschritt- oder im Bulkverfahren abgearbeitet wird, kann es jetzt zu folgenden Konstellationen kommen:

- Fall 1) Die Berechnung für O1 wird ausgeführt, dann die Operation für O2.
- Fall 2) Die Berechnung für O2 wird ausgeführt, dann die Operation für O1.
- Fall 3) Die Berechnungen für O1 und O2 werden in einer Bulkoperation gleichzeitig ausgeführt.

Nach Ausführung dieser Operationen und unter der Annahme, dass sie alle zu Änderungen an den betroffenen totalen Mengen führen, ergeben sich folgende Zustände:

Fall 1) Die Berechnung für O1 wird ausgeführt, dann die Operation für O2.

Operation	Objekt	GenProcID
OrgHasApp	O2	P3
OrgHasApp	O4	P4
OrgHasApp	O2	P4
OrgHasApp	O3	P4
PersonHasApp	X	P4

Vor dem nächsten Lauf des DBQueue Prozessors muss wieder eine Verdichtung der GenProcID's vorgenommen werden, da für die Operation `OrgHasApp` für Objekt O2 keine Eindeutigkeit gegeben ist. P5 wird eingeführt, mit den möglichen Vorgängern P4 und P3.

Operation	Objekt	GenProcID
OrgHasApp	O2	P5
OrgHasApp	O4	P4
OrgHasApp	O3	P4
PersonHasApp	X	P4

Jetzt wird die Berechnung für O2 ausgeführt:

Operation	Objekt	GenProcID
OrgHasApp	O3	P5
PersonHasApp	X	P5
OrgHasApp	O4	P4
OrgHasApp	O3	P4
PersonHasApp	X	P4

Da für O3 keine Eindeutigkeit gegeben ist, wird P6 eingeführt mit den möglichen Vorgängern P4 und P5.

Operation	Objekt	GenProcID
OrgHasApp	O3	P6
PersonHasApp	X	P5
OrgHasApp	O4	P4
PersonHasApp	X	P4

Nach den Berechnungen für O3 und O4 liegt folgende Situation vor:

Operation	Objekt	GenProcID
PersonHasApp	X	P6
PersonHasApp	X	P5
PersonHasApp	X	P4

Für das Objekt X ist keine Eindeutigkeit gegeben, so dass P7 mit den möglichen Vorgängern P4, P5 und P6 eingeführt wird.

Fall 2) Die Berechnung für O2 wird ausgeführt, dann die Operation für O1.

Operation	Objekt	GenProcID
OrgHasApp	O1	P4
OrgHasApp	O2	P3

Nach der Ausführung stehen folgende Einträge in der DBQueue:

Operation	Objekt	GenProcID
OrgHasApp	O1	P4
OrgHasApp	O3	P3

Nach dem nächsten Ausführungsschritt liegt folgende Situation vor:

Operation	Objekt	GenProcID
OrgHasApp	O3	P3
OrgHasApp	O4	P4
OrgHasApp	O2	P4
OrgHasApp	O3	P4
PersonHasApp	X	P4

Für die Eindeutigkeit von O3 muss ein Prozess P5 mit den möglichen Vorgängern P3 und P4 erzeugt werden:

Operation	Objekt	GenProcID
OrgHasApp	O3	P5
OrgHasApp	O4	P4
OrgHasApp	O2	P4
PersonHasApp	X	P4

Nach den Berechnungen liegt folgende Situation vor:

Operation	Objekt	GenProcID
PersonHasApp	X	P5
PersonHasApp	X	P4

Für das Objekt X ist keine Eindeutigkeit gegeben, so dass P6 mit den möglichen Vorgängern P4 und P5 eingeführt wird.

Fall 3) Die Berechnungen für O1 und O2 werden in einer Bulkoperation gleichzeitig ausgeführt.

Operation	Objekt	GenProcID
OrgHasApp	O1	P4
OrgHasApp	O2	P3

Nach dem ersten Berechnungsschritt stehen folgende Informationen in der DBQueue:

Operation	Objekt	GenProcID
OrgHasApp	O4	P4
OrgHasApp	O2	P4
OrgHasApp	O3	P4
OrgHasApp	O3	P3
PersonHasApp	X	P4

Für O3 wird durch Prozess P5 mit den möglichen Vorgängern P3 und P4 Eindeutigkeit hergestellt:

Operation	Objekt	GenProcID
OrgHasApp	O4	P4
OrgHasApp	O2	P4
OrgHasApp	O3	P5
PersonHasApp	X	P4

Nach dem nächsten Berechnungsschritt ist folgender Inhalt zu finden:

Operation	Objekt	GenProcID
OrgHasApp	O3	P4
PersonHasApp	X	P4
PersonHasApp	X	P5

Nachdem im nächsten Durchlauf O3 errechnet wurde und dieser keinen neuen PersonHasApp-Eintrag erzeugt hat, da X mit P4 schon existiert, steht zum Schluss nur X mit P4 und P5.

Operation	Objekt	GenProcID
PersonHasApp	X	P4
PersonHasApp	X	P5

Für das Objekt X ist keine Eindeutigkeit gegeben, so dass P6 mit den möglichen Vorgängern P4 und P5 eingeführt wird.

Archivieren und Löschen von Aufzeichnungen

Alle im One Identity Manager protokollierten Aufzeichnungen werden zunächst in der One Identity Manager-Datenbank gespeichert. Der Anteil der historisierten Daten am Gesamtvolumen einer One Identity Manager-Datenbank sollte maximal 25 % betragen. Anderenfalls kann es zu Performance-Problemen kommen. Die Aufzeichnungen sollten in regelmäßigen Abständen aus der One Identity Manager-Datenbank entfernt und archiviert werden.

Um die aufgezeichneten Daten in regelmäßigen Abständen aus der One Identity Manager-Datenbank zu entfernen, werden folgende Verfahren angeboten:

- Die Daten können direkt aus der One Identity Manager-Datenbank in eine One Identity Manager History Database übernommen werden. Dieses ist das Standardverfahren für die Datenarchivierung. Wählen Sie dieses Verfahren, wenn die Server auf denen die One Identity Manager-Datenbank und die One Identity Manager History Database liegen einander sehen.
- Die Daten werden ohne Archivierung nach einer festgelegten Zeitspanne aus der One Identity Manager-Datenbank gelöscht.

Ausführliche Informationen zur Einrichtung der Archivierung von Daten in einer History Database finden Sie im *One Identity Manager Administrationshandbuch für die Datenarchivierung*.

Detaillierte Informationen zum Thema

- Löschen der Aufzeichnungen in der One Identity Manager-Datenbank ohne Archivierung auf Seite 365
- Aufbewahrungszeiten für Aufzeichnungen festlegen auf Seite 365
- Performance-Optimierung zum Löschen von Aufzeichnungen auf Seite 366

Löschen der Aufzeichnungen in der One Identity Manager-Datenbank ohne Archivierung

Sollen die Aufzeichnungen einzelner Teilbereiche für einen gewissen Zeitraum in der One Identity Manager-Datenbank gehalten werden, jedoch keine spätere Archivierung erfolgen, dann haben Sie folgende Möglichkeiten:

- Um einen einzelnen Teilbereich von der Archivierung auszuschließen, konfigurieren Sie diesen Teilbereich nicht für den Export, sondern legen nur den Aufbewahrungszeitraum fest.
- Um alle Teilbereiche ohne Archivierung direkt zu löschen, legen Sie die Aufbewahrungszeiten fest. Aktivieren Sie im Designer den Konfigurationsparameter **Common | ProcessState | ExportPolicy** und tragen Sie den Wert **NONE** ein.

Die Aufzeichnungen werden nach Ablauf der Aufbewahrungszeit durch den DBQueue Prozessor aus der One Identity Manager-Datenbank gelöscht. Zusätzlich werden alle Einträge für ausgelöste Aktionen gelöscht, zu denen es keine Aufzeichnungen in den Teilbereichen gibt.

HINWEIS: Wenn Sie keine Aufbewahrungszeiten festlegen, werden die Aufzeichnungen dieser Teilbereiche innerhalb der täglichen Wartungsaufträge des DBQueue Prozessor aus der One Identity Manager-Datenbank gelöscht.

Verwandte Themen

- Aufbewahrungszeiten für Aufzeichnungen festlegen auf Seite 365
- Weitere Informationen finden Sie unter Performance-Optimierung zum Löschen von Aufzeichnungen auf Seite 366.

Aufbewahrungszeiten für Aufzeichnungen festlegen

Die Aufzeichnungen werden, abhängig vom gewählten Archivierungsverfahren, nach Ablauf der Aufbewahrungszeiten aus der One Identity Manager-Datenbank exportiert oder gelöscht. Für die Teilbereiche, deren Aufzeichnungen exportiert werden, sollte eine längere Aufbewahrungszeit gewählt werden, als für die Teilbereiche, deren Aufzeichnungen gelöscht werden.

HINWEIS: Wenn Sie keine Aufbewahrungszeiten festlegen, werden die Aufzeichnungen dieser Teilbereiche täglich innerhalb der tägliche Wartungsaufträge des DBQueue Prozessors aus der One Identity Manager-Datenbank gelöscht.

Die Aufzeichnungen werden erst exportiert, wenn die Aufbewahrungszeiten aller Teilbereiche abgelaufen ist und keine weiteren aktiven Prozesse für die Prozessgruppe (GenProcID) in der DBQueue, der Prozesshistorie oder als geplante Operation existieren.

Die Aufbewahrungszeiten für die einzelnen Bereiche legen Sie über Konfigurationsparameter fest.

Tabelle 124: Konfigurationsparameter für die Behandlung der aufgezeichneten Datenänderungen

Konfigurationsparameter	Bedeutung
Common | ProcessState | PropertyLog | IsToExport	Die aufgezeichneten Datenänderungen sollen exportiert werden. Ist der Konfigurationsparameter nicht aktiv, werden die Informationen nach Ablauf der Aufbewahrungszeit gelöscht.
Common | ProcessState | PropertyLog | LifeTime	Maximale Aufbewahrungszeit in Tagen für aufgezeichnete Datenänderungen in der Datenbank. Standardwert ist **30**.

Tabelle 125: Konfigurationsparameter für die Behandlung der Prozessinformationen

Konfigurationsparameter	Bedeutung
Common | ProcessState | ProgressView | IsToExport	Die Prozessinformationen sollen exportiert werden. Ist der Konfigurationsparameter nicht aktiv, werden die Informationen nach Ablauf der Aufbewahrungszeit gelöscht.
Common | ProcessState | ProgressView | LifeTime	Maximale Aufbewahrungszeit in Tagen für Prozessinformationen in der Datenbank. Standardwert ist **30**.

Tabelle 126: Konfigurationsparameter für die Behandlung der Prozesshistorie

Konfigurationsparameter	Bedeutung
Common | ProcessState | JobHistory | IsToExport	Die Informationen in der Prozesshistorie sollen exportiert werden. Ist der Konfigurationsparameter nicht aktiv, werden die Informationen nach Ablauf der Aufbewahrungszeit gelöscht.
Common | ProcessState | JobHistory | LifeTime	Maximale Aufbewahrungszeit in Tagen für Aufzeichnungen aus der Prozesshistorie in der Datenbank. Standardwert ist **30**.

Performance-Optimierung zum Löschen von Aufzeichnungen

Bei großen Datenmengen können Sie zur Performance-Optimierung die Menge der zu löschenden Objekte pro Operation und Verarbeitungslauf des DBQueue Prozessor

festlegen. Die Festlegung für die einzelnen Bereiche treffen Sie über Konfigurationsparameter.

Tabelle 127: Konfigurationsparameter für das Löschen der aufgezeichneten Datenänderungen

Konfigurationsparameter	Bedeutung
Common | ProcessState | PropertyLog | Delete	Erlaubt die Konfiguration des Löschverhaltens für aufgezeichnete Datenänderungen.
Common | ProcessState | PropertyLog | Delete | BulkCount	Anzahl der Einträge, die in einer Operation gelöscht werden sollen. Standardwert ist **200**.
Common | ProcessState | PropertyLog | Delete | TotalCount	Gesamtmenge der Einträge, die in einem Verarbeitungslauf gelöscht werden sollen. Standardwert ist **10000**.

Tabelle 128: Konfigurationsparameter für das Löschen der Prozessinformationen

Konfigurationsparameter	Bedeutung
Common | ProcessState | ProgressView | Delete	Erlaubt die Konfiguration des Löschverhaltens für Prozessinformationen.
Common | ProcessState | ProgressView | Delete | BulkCount	Anzahl der Einträge, die in einer Operation gelöscht werden sollen. Standardwert ist **200**.
Common | ProcessState | ProgressView | Delete | TotalCount	Gesamtmenge der Einträge, die in einem Verarbeitungslauf gelöscht werden sollen. Standardwert ist **10000**.

Tabelle 129: Konfigurationsparameter für das Löschen der Prozesshistorie

Konfigurationsparameter	Bedeutung
Common | ProcessState | JobHistory | Delete	Erlaubt die Konfiguration des Löschverhaltens für die Prozesshistorie.
Common | ProcessState | JobHistory | Delete | BulkCount	Anzahl der Einträge, die in einer Operation gelöscht werden sollen. Standardwert ist **200**.
Common | ProcessState | JobHistory | Delete | TotalCount	Gesamtmenge der Einträge, die in einem Verarbeitungslauf gelöscht werden sollen. Standardwert ist **10000**.

Tabelle 130: Konfigurationsparameter für das Löschen von Prozessstatus-Einträge

Konfigurationsparameter	Bedeutung
Common | ProcessState | Delete	Erlaubt die Konfiguration des Löschverhaltens für die Einträge zum Prozessstatus.
Common | ProcessState | Delete | BulkCount	Anzahl der Einträge, die in einer Operation gelöscht werden sollen. Standardwert ist **500**.
Common | ProcessState | Delete | TotalCount	Gesamtmenge der Einträge, die in einem Verarbeitungslauf gelöscht werden sollen. Standardwert ist **10000**.

Bedingte Kompilierung mittels Präprozessorbedingungen

Im One Identity Manager ist die bedingte Kompilierung von Programmcode integriert. Über die bedingte Kompilierung können Teile des Programmcodes übersetzt werden, während andere Teile von der Kompilierung ausgeschlossen werden.

Die bedingte Kompilierung bietet folgende Vorteile:

- Verkleinern der Assemblies
- Strukturieren der Systemkonfiguration
- Erhöhen der Übersicht im Modell und in den Berechtigungen
- Beschleunigen der Verarbeitung
- Ausblenden nicht benötigter Informationen in allen VB.Net-Ausdrücken
- Ausblenden von nicht benötigten Modellanteilen

Die bedingte Kompilierung im One Identity Manager wird über Präprozessorbedingungen gesteuert. Präprozessorbedingungen können angewendet werden in:

- Objekten, die die Eigenschaft **Präprozessorbedingung** besitzen
- VB.Net-Ausdrücken

HINWEIS: Mit Hilfe der Präprozessorbedingungen wird ein Modul nicht deaktiviert. So werden beispielsweise SQL Prozeduren und Trigger weiterhin ausgeführt, auch wenn die Objekte durch Präprozessorbedingungen deaktiviert sind. Um ein Modul zu deaktivieren, entfernen Sie das Modul aus der One Identity Manager-Datenbank. Ausführliche Informationen zum Entfernen von Modulen finden Sie im *One Identity Manager Installationshandbuch*.

Die möglichen Präprozessorbedingungen werden über Konfigurationsparameter und deren Optionen definiert. Um systemweit wirksam zu werden, erfordert jegliche Änderung an präprozessorrelvanten Konfigurationsparametern sowie Änderungen der Präprozessorbedingungen an Objekten und in VB.Net-Ausdrücken eine Kompilierung der One Identity Manager-Datenbank.

Detaillierte Informationen zum Thema

- Präprozessorrelevante Konfigurationsparameter auf Seite 370
- Präprozessorbedingungen in Objekten auf Seite 371
- Präprozessorbedingungen in VB.Net-Ausdrücken auf Seite 372
- Auswertung der Präprozessorbedingungen bei der Kompilierung auf Seite 373

Präprozessorrelevante Konfigurationsparameter

WICHTIG: Jede Änderung an einem präprozessorrelevanten Konfigurationsparameter und seinen Optionen erfordert eine Kompilierung der One Identity Manager-Datenbank.

Ein präprozessorrelevanter Konfigurationsparameter ist durch die Option **Präprozessorrelevanter Parameter** gekennzeichnet. Für die zugehörige Konfigurationsparameteroption ist ein Präprozessorausdruck eingetragen, der als Präprozessorbedingung verwendet werden kann.

Durch die Aktivierung eines präprozessorrelevanten Konfigurationsparameters gilt die Präprozessorbedingung systemweit als vorhanden. Erst durch die Kompilierung der Datenbank wird die Präprozessorbedingung wirksam.

HINWEIS: Vordefinierte präprozessorrelevante Konfigurationsparameter und Optionen werden bei Schemainstallationen überschrieben. Kundenspezifische präprozessorrelevante Konfigurationsparameter und Optionen definieren Sie im Designer unter dem Konfigurationsparameter **Custom**.

Um präprozessorrelevante Konfigurationsparameter anzuzeigen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Konfigurationsparameter**.
2. Wählen Sie im Konfigurationsparametereditor den Menüeintrag **Ansicht > Präprozessordefinitionen**.

   In der Ansicht **Präprozessordefinitionen** alle Präprozessorbedingungen angezeigt. Mit Maus-Doppelklick auf einen Eintrag wird der Konfigurationsparameter angezeigt.

HINWEIS: Einen Überblick über die vorhandenen Präprozessorabhängigkeiten erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Präprozessorabhängigkeiten**.

Verwandte Themen

- Kundenspezifische Konfigurationsparameter erstellen auf Seite 43
- Präprozessorbedingungen in Objekten auf Seite 371

- Präprozessorbedingungen in VB.Net-Ausdrücken auf Seite 372
- Auswertung der Präprozessorbedingungen bei der Kompilierung auf Seite 373

Präprozessorbedingungen in Objekten

WICHTIG: Jede Änderung der Präprozessorbedingungen an Objekten erfordert eine Kompilierung der One Identity Manager-Datenbank.

An einigen Objekten können Sie eine Präprozessorbedingung direkt angeben.

Um eine Präprozessorbedingung zu erfassen

- Tragen Sie in der Eigenschaft **Präprozessorbedingung** die Präprozessorausdrücke der Konfigurationsparameter ein. Die Präprozessorausdrücke können Sie über `AND`, `OR`, `NOT`, `()` miteinander verknüpfen.

Beispiel:

Die Spalte `Person.RiskIndexCalculated` soll in der Oberfläche nur angezeigt werden, wenn die Risikoberechnung aktiviert ist.

In der Spaltendefinition (Tabelle `DialogColumn`) wird folgende Präprozessorbedingung eingetragen.

Tabelle 131: Beispiel für Präprozessorbedingungen

Tabelle	Spalte	Präprozessorbedingung
Person	RiskIndexCalculated	RISKINDEX

Wird ein präprozessorrelvanter Konfigurationsparameter aktiviert oder deaktiviert, werden für den DBQueue Prozessor Aufträge zur kompletten Berechnung aller Präprozessoraufträge und Berechnungsaufträge für die betroffenen Objekte erzeugt. An den betroffenen Objekten wird die Option **Deaktiviert durch Präprozessor** aktualisiert. Führt die erneute Auswertung der Präprozessorbedingungen zu einer Änderung der Option, werden weitere Folgeaufträge zur Präprozessorauswertung an den abhängigen Objekten erzeugt. Die Benutzerberechtigungen können ebenfalls betroffen sein. Nach der Abarbeitung der Aufträge durch den DBQueue Prozessor ist eine Kompilierung der Datenbank erforderlich.

Die Auswertung der Präprozessorbedingungen hat folgende Auswirkungen:

- Ist eine Tabelle über eine Präprozessorbedingung deaktiviert, werden auch alle ihre Spalten, alle Objektdefinitionen, die sich auf die Tabelle beziehen sowie die Oberflächenformulare und die zugehörige Menüführung deaktiviert.

- Ist eine Primärschlüsselspalte deaktiviert, so werden alle Fremdschlüsselspalten deaktiviert, die sich auf diesen Primärschlüssel beziehen.
- Wird nach der vorhergehenden Regel ein Primärschlüssel-Mitglied deaktiviert, beispielsweise bei M:N-Tabellen, so werden auch die Tabelle dieses Primärschlüssels sowie alle weiteren Spalten dieser Tabelle deaktiviert.

  Dieses Verfahren hat den Vorteil, dass beispielsweise bei Deaktivierung der Tabelle `ADSGroup` automatisch auch alle Zuordnungen deaktiviert werden, wie beispielsweise die Tabelle `DepartmentHasADSGroup`.

HINWEIS: Einen Überblick über die vorhandenen Präprozessorabhängigkeiten erhalten Sie im Designer in der Kategorie **One Identity Manager Schema > Präprozessorabhängigkeiten**.

Verwandte Themen

- Präprozessorrelevante Konfigurationsparameter auf Seite 370
- Präprozessorbedingungen in VB.Net-Ausdrücken auf Seite 372
- Auswertung der Präprozessorbedingungen bei der Kompilierung auf Seite 373

Präprozessorbedingungen in VB.Net-Ausdrücken

WICHTIG: Jede Änderung der Präprozessorbedingungen in VB.Net-Ausdrücken erfordert eine Kompilierung der One Identity Manager-Datenbank.

Präprozessorbedingungen können in VB.Net-Ausdrücken verwendet werden. Skriptcode, der von einer Präprozessorbedingung abhängig ist, müssen Sie in einer `#If...Then...#Else` - Anweisung übergeben. Zur Formulierung der Präprozessorbedingung verwenden Sie die Präprozessorausdrücke der präprozessorrelevanten Konfigurationsparameter. Die Präprozessorausdrücke können Sie über `AND`, `OR`, `NOT`, `()` miteinander verknüpfen.

Die Auswertung der Präprozessorbedingungen erfolgt erst bei der Generierung der Skripte.

Syntax

```
#If <Präprozessorbedingung_1>
     ’ Code, der für diese Präprozessorbedingung gelten soll
#ElseIf <Präprozessorbedingung_2> Then
     ’ Code, der für diese Präprozessorbedingung gelten soll
#Else
’ Code, der sonst gelten soll
#Endif
```

Beispiel:

Für die Spalte `ADSGroup.IsForITShop` wurde in der Spaltendefinition (Tabelle `DialogColumn`) die Präprozessorbedingung `ITSHOP` eingetragen. In der Bildungsregel der Spalte `ADSGroup.DisplayName` soll die Spalte `IsForITShop` referenziert werden. Um kompilierfähig zu bleiben, muss für die Bildungsregel folgende Konstruktion verwendet werden:

```
#If ITSHOP Then
     If $IsForITShop:Bool$ And $UID_AccProduct$ <> "" Then
          Value = $FK(UID_AccProduct).Ident_AccProduct$
     Else
          value = $cn$
     End If
#Else
     value = $cn$
#End If
```

Verwandte Themen

- Präprozessorrelevante Konfigurationsparameter auf Seite 370
- Präprozessorbedingungen in Objekten auf Seite 371
- Auswertung der Präprozessorbedingungen bei der Kompilierung auf Seite 373

Auswertung der Präprozessorbedingungen bei der Kompilierung

Um systemweit wirksam zu werden, erfordert jegliche Änderung an präprozessorrelvanten Konfigurationsparametern sowie Änderungen der Präprozessorbedingungen an Objekten und in VB.Net-Ausdrücken eine Kompilierung der One Identity Manager-Datenbank.

Für die Kompilierung gilt Folgendes:

- Für Objekte mit einer Präprozessorbedingung wird intern ein Programmcode mit einer entsprechenden `#If...Then...#Else` - Anweisung erzeugt. Der Programmcode, der in Sektionen liegt, deren Präprozessorbedingung nicht zutrifft, ist für den Compiler nicht vorhanden und wird nicht übersetzt. Diese Objekte gelten als nicht vorhanden.

- VB.Net-Ausdrücke, die Präprozessorbedingungen enthalten, werden kompiliert. Der Programmcode ist vorhanden. Die Auswertung der Präprozessorbedingungen erfolgt erst bei der Generierung der Skripte.

Für das Kompilieren der Bildungsregeln gilt:

- Bildungsregeln auf Spalten, die per Präprozessorbedingung deaktiviert sind, werden nicht kompiliert und die daraus resultierenden Beziehungen werden nicht in der Tabelle `DialogNotification` gespeichert. Diese Spalten sind also komplett als nicht vorhanden anzusehen.
- Bildungsregeln, die sich auf deaktivierte Spalten beziehen, werfen beim Kompilieren eine Fehlermeldung, falls der entsprechende Code-Teil nicht auch in eine Präprozessoranweisung eingebunden ist.

Skripte im One Identity Manager

Skripte im One Identity Manager dienen zur Überwachung und Erhaltung der Datenkonsistenz und der Businesslogik des Kunden in der Datenbank. Skripte können eingesetzt werden zum:

- Überprüfen von Spaltenwerten
- Auslösen von Ereignissen
- Erzeugen, Verändern und Löschen von Objekten und damit zur Manipulation der Datenbank

Detaillierte Informationen zum Thema

- Verwendung von Visual Basic .NET-Skripten auf Seite 375
- Hinweise zur Ausgabe von Meldungen auf Seite 376
- Hinweise zur Verwendung von Datumswerten auf Seite 376
- Hinweise zur Verwendung von Windows PowerShell Skripten auf Seite 378
- Verwendung der $-Notation auf Seite 378
- Verwendung von base Objekt auf Seite 385
- Aufruf von Funktionen auf Seite 385
- Prä-Skripte zur Verwendung in Prozessen und Prozessschritten auf Seite 385
- Verwendung von Session Services auf Seite 386
- Verwendung der #LD-Notation auf Seite 389
- Skriptbibliothek auf Seite 393

Verwendung von Visual Basic .NET-Skripten

Die One Identity Manager-Skripte werden in VB.Net-Syntax abgelegt, was die Nutzung aller VB.Net-Funktionen zulässt. Die zu verarbeitenden Werte werden als

Präprozessoranweisungen angegeben.

> HINWEIS: Umfangreiche Beispiele für die Syntax und die Verwendung von Skripten finden Sie auf dem Installationsmedium im Verzeichnis `QBM\dvd\AddOn\SDK\ScriptSamples`.

Skripte können Sie verwenden in:

- Bildungsregeln und Formatierungsskripten (Tabelle `DialogColumn`)
- Tabellenskripte (Tabelle `DialogTable`)
- Skriptbibliothek (Tabelle `DialogScript`)
- Methoden (Tabelle `DialogMethod`)
- Auswahlskripten für Objektdefinitionen (Tabelle `DialogObject`)
- Auswahlskripten für Datenbanksichten (Tabelle `DialogTable`)
- Skripten zur Serverermittlung zur Ausführung von Prozessschritten (Tabelle `Job`)
- Parametern von Prozessschritten (Tabelle `Jobrunparameter`)
- Benachrichtigungen zur Prozessverarbeitung (Tabelle `Job`)
- Generierungsbedingungen von Prozessschritten und Prozessen (Tabellen `Job` und `JobChain`)
- Prä-Skripten von Prozessschritten und Prozessen (Tabellen `Job` und `JobChain`)
- Prozessinformationen (Tabellen `Job`, `JobChain` und `JobEventgen`)
- Mailvorlagen (Tabelle `DialogRichMailBody`)

Hinweise zur Ausgabe von Meldungen

Auf Servern dürfen niemals die VB.Net-Funktionen `Msgbox` und `Inputbox` verwendet werden. Nutzen Sie die Funktionen `VID_Write2Log`, `RaiseMessage` oder `AppData.Instance.RaiseMessage`.

Beispiele für Ausgaben in die Protokolldatei des One Identity Manager Service entnehmen Sie den Skriptbeispielen auf dem Installationsmedium im Verzeichnis `QBM\dvd\AddOn\SDK\ScriptSamples`.

Hinweise zur Verwendung von Datumswerten

- Ist explizit kein Datum angegeben, wird intern das Datum 30.12.1899 verwendet. Dies ist bei Wertevergleichen zu beachten, beispielsweise bei der Verwendung in

Berichten.

Beispiel: Ausdruck für die Darstellung der Datenspalten in Berichten

```
{IIF(Person.ExitDate.ToString() = "12/30/1899 12:00:00 AM","-
",Person.ExitDate)}
```

- Datumszeitwerte, wie beispielsweise Erstellungsdatum oder Änderungsdatum, werden in der Datenbank mit der jeweils aktuellen UTC vermerkt. Die Objektschicht wandelt diese Zeitinformationen beim Laden eines Objektes in die jeweils gültige Zeitzoneninformationen um. Der Anwender sieht also alle Werte in seiner lokalen Zeit. Beim Speichern eines Objektes werden die aktuellen Zeitzoneninformationen in UTC Zeitinformationen umgewandelt.

  HINWEIS: Die Verwendung von `DateTime.Now` in Skripten sollte kritisch geprüft werden. Um den Wert für den Anwender anzuzeigen sollte statt `DateTime.Now` besser `DateTime.UTCNow` verwendet werden.

- Es ist fehlerbehaftet, in Skripten Datumswerte in deutscher Notation vom Datentyp `String` in den Datentyp `DateTime` zu konvertieren:

  ```
  Value = CDate("31.12.2014")
  ```

  Dies führt immer dann zu Problemen, wenn das Skript in nicht-deutschen Systemen läuft. Im günstigsten Fall erhält man eine Fehlermeldung `"Cast from string...to type Date is not valid."`. Im ungünstigsten Fall wird das falsche Datum geliefert, da Monat und Tag vertauscht werden (aus dem 12.3.2014 wird der 3.12.2014).

  Wenn möglich, sollte man in diesem Fall Zeichenkettenkonvertierungen vermeiden. Der Typ `DateTime` stellt für diesen Zweck verschiedene Konstruktoren zur Verfügung. Für das oben genannte Beispiel wäre das:

  ```
  Value = new DateTime(2014, 12, 31)
  ```

  Wenn trotzdem der Datentyp `String` verwendet werden soll, so ist die ISO-Notation des Datumswertes zu verwenden, da diese in allen Einstellungen korrekt umgewandelt wird:

  ```
  Value = CDate("2014-12-31")
  ```

  ```
  Value = CDate("2014-12-31 15:22:12")
  ```

  Die komplizierte Variante ist die Angabe der zu benutzenden Kultur beziehungsweise die Angabe des Formats, in dem das Datum vorliegt:

  ```
  Value = DateTime.Parse("31.12.2014", new CultureInfo("de-DE"))
  ```

  ```
  Value = DateTime.ParseExact("31.12.2014", "dd.MM.yyyy",
  CultureInfo.InvariantCulture)
  ```

Hinweise zur Verwendung von Windows PowerShell Skripten

Beispiele für die Syntax und die Verwendung von Windows PowerShell Skripten im One Identity Manager finden Sie unter https://github.com/OneIdentity/IdentityManager.PoSh. Beachten Sie die dort beschriebenen Installationsvoraussetzungen und Hinweise.

Standardmäßig versucht das Windows PowerShell Modul, alle referenzierten DLLs von einer gültigen One Identity Manager Installation zu laden. In der Standardinstallation wird der One Identity Manager installiert unter:

- `%ProgramFiles(x86)%\One Identity` (auf 32-Bit Betriebssystemen)
- `%ProgramFiles%\One Identity` (auf 64-Bit Betriebssystemen)

Verwendung der $-Notation

Die $-Notation wird im One Identity Manager verwendet, um auf die Eigenschaften von Objekten zuzugreifen.

Syntax

```
$<Definition>:<Datentyp>{<Format>}$
```

Bei Verwendung der $-Notation ist darauf zu achten, dass der Wert den korrekten Datentyp erhält. Die $-Notation liefert als Standard den Typ `String` zurück.

Als Datentypen sind zulässig:

```
Binary
Bool
Byte
Date
Decimal
Double
Int
Long
Short
String (Standard)
Text
```

Die Formatangabe ist optional. Wenn das Format angegeben wird, ist der Zieltyp des Ausdrucks ein `String`. Wird das Format nicht angegeben, ist es der angegebene Datentyp.

Die Formatangaben entsprechen den Formatzeichenfolgen der Funktion `String.Format` für die einzelnen Typen. Ausführliche Informationen zu den Formatzeichenfolgen finden Sie unter https://docs.microsoft.com/en-us/dotnet/standard/base-types/composite-formatting#format-string-component.

Beispiele:

`$MaxValidDays:Int{000}$`

ergibt für den Wert **42** die Ausgabe "**042**"

`$XDateUpdated:Date{t}$`

ergibt "**11:16**"

HINWEIS: Wenn Sie in Skripten ein **$**-Zeichen verwenden möchten, ohne das es den Zugriff auf einen Spaltennamen repräsentieren soll, dann müssen Sie dieses Zeichen durch Verdopplung maskieren.

Beispiel:

In Windows PowerShell Skripten ist statt

```
theScript.AppendLine("foreach ($Domain in $Domains)")
```

Folgendes zu verwenden:

```
theScript.AppendLine("foreach ($$Domain in $$Domains)")
```

Detaillierte Informationen zum Thema

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Referenzen in Kommentaren auf Seite 383
- Zugriff auf Metawerte des lokalen Objektes auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf Spalten des lokalen Objektes

Syntax

```
$<Spaltenname>:<Datentyp>{<Format>}$
```

Beispiele zur Verwendung in Bildungsregeln

Der Anzeigename eines Active Directory Benutzers soll aus dem Vornamen und dem Nachnamen des Active Directory Benutzers gebildet werden. Die Bildungsregel auf `ADSAccount.Displayname` lautet:

```
If $Givenname$<>"" And $Surname$<>"" Then
     Value = $Surname$ & " " & $Givenname$
ElseIf $Givenname$<>"" Then
     Value = $Givenname$
ElseIf $Surname$<>"" then
     Value = $Surname$
End If
```

Wird eine Person deaktiviert, so soll das Austrittsdatum der Person gesetzt werden. Die Bildungsregel auf `Person.Exitdate` lautet:

```
If $IsInActive:bool$ Then
     Value = Date.Today
End If
```

Verwandte Themen

- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Referenzen in Kommentaren auf Seite 383
- Zugriff auf Metawerte des lokalen Objektes auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes

Derzeit ist als Beziehung nur die Fremdschlüsselbeziehung zugelassen.

Syntax

```
$FK(<Fremdschlüsselspalte>).<Spaltenname>:<Datentyp>{<Format>}$
```

Beispiel zur Verwendung in Bildungsregeln

Der Vorname eines Active Directory Benutzers soll aus dem Vornamen der zugeordneten Person gebildet werden. Die Bildungsregel auf `ADSAccount.Givenname` lautet:

```
Value = $FK(UID_Person).Firstname$
```

Verwandte Themen

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Referenzen in Kommentaren auf Seite 383
- Zugriff auf Metawerte des lokalen Objektes auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf den alten Wert einer Spalte

Syntax

```
$Spaltenname[o]$
```

Beispiel zur Verwendung in den Parametern der Prozessschritte

Optionale Parameter von Prozessschritten werden nicht generiert, wenn der Wert auf **Nothing** gesetzt wird oder in der Wertvorlage nicht zugewiesen wird. Damit ist es möglich die Parameterzahl bei Zielsystem-Komponenten einzuschränken. Soll ein solcher Wert geleert werden, ist statt **Nothing** ein Leerstring zu übergeben.

Eine Wertvorlage könnte beispielsweise so aussehen:

```
If $Lastname[o]$ <> $Lastname$ Then
     Value = $Lastname$
End If
```

> HINWEIS: Für einige der Standardspalten wie beispielsweise `XDateInserted`, `XDateUpdated`, `XUserInserted`, `XUserUpdated`, `XOrigin`, `XIsInEffect` und `XMarkedForDeletion` werden die neuen Werte erst nach dem Speichern des Objektes gebildet. Damit ist während der Verarbeitung der Bildungsregeln der neue Wert der Spalten immer gleich dem alten Wert (beispielsweise `$XDateUpdated[o]$ = $XDateUpdated$`).

Verwandte Themen

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Referenzen in Kommentaren auf Seite 383

- Zugriff auf Metawerte des lokalen Objektes auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf den Anzeigewert einer Spalte

Bei der Bildung des Anzeigewertes für eine Spalte werden die Eigenschaften **Mehrsprachig** (`IsMultiLanguage`) und **Liste zulässiger Werte** (`LimitedValues`) aufgelöst.

Syntax

```
$Spaltenname[D]$
```

Um auf den Anzeigewert eines alten Wertes einer Spalte zuzugreifen, können die Eigenschaften `[O]` und `[D]` kombiniert werden.

```
$Spaltenname[OD]$
```

```
$Spaltenname[DO]$
```

Beispiel zur Verwendung

Für die Beschränkungsart der HCL Domino Server Beschränkungen ist eine Liste zulässiger Werte definiert.

```
PrivateList=Run Personal Agent RestrictedList=Run Restricted Agent
UnrestrictedList=Run Unrestricted Agent
```

Für eine Serverbeschränkung mit den Wert **PrivateList** wird auf dem Informationsformular der Anzeigewert **Run Personal Agent** dargestellt.

Beispiel zur Verwendung in Bildungsregeln

Der Anzeigewert der konkreten Serverbeschränkung soll aus dem Namen des HCL Domino Benutzers und dem Anzeigewert des Beschränkungsart gebildet werden.

```
Value = vid_Left($FK(UID_NotesUser).FullName1st$,39) & " [" & vid_Left
($NotesAgentMgrType[D]$, 22) & "]"
```

Verwandte Themen

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf Referenzen in Kommentaren auf Seite 383
- Zugriff auf Metawerte des lokalen Objektes auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf Referenzen in Kommentaren

Der Präprozessor interpretiert auch Referenzen, die innerhalb von Kommentaren liegen, beispielsweise `'$Lastname$`. Die Referenzierung einer Spalte in einem Skript-Kommentar führt dazu, dass ein Skript bei Änderung des Spaltenwerts ausgeführt wird.

Beispiel zur Verwendung in Bildungsregeln

Das Eintrittsdatum einer Person wird mit einer Bildungsregel belegt. Diese Bildungsregel soll wirksam werden, wenn sich der Nachname der Person ändert. Die Bildungsregel auf `Person.Entrydate` ist dann:

```
'$Lastname$
```

```
Value = Date
```

Verwandte Themen

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Metawerte des lokalen Objektes auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf Metawerte des lokalen Objektes

Syntax

```
$[IsLoaded]:Bool$
```

Tabelle 132: Metawerte und ihre Bedeutung

Metawert	Bedeutung
IsLoaded	Dieser Wert gibt an, ob das Objekt aus der Datenbank geladen wurde.
IsChanged	Dieser Wert gibt an, ob ein Objekt oder eine Spalte seit dem Laden verändert wurde.
IsDifferent	Dieser Wert gibt an, ob sich der neue Wert und der alte Wert unterscheiden. Der Zugriff auf Spalten kann erfolgen über `Spaltenname[C]`.
IsDeleted	Dieser Wert gibt an, ob das Objekt zum Löschen markiert ist.

Verwandte Themen

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Referenzen in Kommentaren auf Seite 383
- Zugriff auf die Anzeigewerte eines Objektes auf Seite 384

Zugriff auf die Anzeigewerte eines Objektes

Syntax

Anzeigemuster entsprechend der Spalte `DisplayPattern` eines Objektes:

```
$[Display]$
```

Anzeigemuster entsprechend der Spalte `DisplayPatternLong` eines Objektes:

```
$[LongDisplay]$
```

Beispiel zur Verwendung am Basisobjekt

```
$[Display]$
```

Beispiel zur Verwendung über Fremdschlüsselbeziehung

Für Zuweisungen von Active Directory Gruppen an Abteilungen soll der `DisplayPatternLong` der `ADSGroup` verwendet werden.

```
$FK(UID_ADSGroup).[LongDisplay]$
```

Verwandte Themen

- Zugriff auf Spalten des lokalen Objektes auf Seite 379
- Zugriff auf Spalten eines über eine Beziehung verbundenen Objektes auf Seite 380
- Zugriff auf den alten Wert einer Spalte auf Seite 381
- Zugriff auf den Anzeigewert einer Spalte auf Seite 382
- Zugriff auf Referenzen in Kommentaren auf Seite 383
- Zugriff auf Metawerte des lokalen Objektes auf Seite 383

Verwendung von base Objekt

Mit `base.` wird immer das aktuell geladene Objekt angesprochen. Das `base.`-Objekt kann in Methoden, Auswahlskripten für Objektdefinitionen und in Einfügewerten verwendet werden. In Bildungsregeln, Formatierungsskripten und Prozessen kann das `base.` Objekt nicht eingesetzt werden.

Syntax

- Einfache Wertzuweisung

  ```
  Base.PutValue("<Spalte>", <Wert>)
  ```

- Wertzuweisung mit Variablenersetzung (Wert muss eine Zeichenkette sein)

  ```
  Base.PutValue("<Spalte>", context.Replace(<Wert>))
  ```

Beispiel

```
Base.PutValue("IsForITShop", 1)
Base.PutValue("UID_ADSContainer", context.Replace("%cont%"))
```

Aufruf von Funktionen

Aufrufe von Funktionen werden in der Skriptbibliothek (Tabelle `DialogScript`) abgelegt.

Beispiel für eine Funktion in der Skriptbibliothek

```
Public Function BuildInternalName(ByVal Firstname As String,ByVal Lastname As
String) As String
     BuildInternalName = Lastname & Firstname
End Function
```

Verwendung der Funktion in einer Bildungsregel auf `Person.Internalname`

```
Value = BuildInternalName($Firstname$, $Lastname$)
```

Prä-Skripte zur Verwendung in Prozessen und Prozessschritten

Der Code von Prä-Skripten wird vor der Ausführung von anderen Skripten ausgeführt. Es können prozesslokale Variablen definiert werden. Prozesslokale Variablen sind lokale

Datenspeicher bei der Generierung eines Prozesses. Sie dienen zur einmaligen Bestimmung von Werten innerhalb eines Prä-Skriptes, die dann innerhalb der Prozesse und ihrer Prozessschritte weiterverwendet werden können, beispielsweise in Generierungsbedingungen, Serverauswahlskripten oder in den Parametern.

> HINWEIS: Es wird empfohlen, die prozesslokalen Variablen nur im Prä-Skript zu besetzen und bei der weiteren Verwendung lesend darauf zuzugreifen.

Syntax im Prä-Skript eines Prozesses

```
values("Name") = "Wert"
```

Verwendung in den Codeteilen des Prozesses und seiner Prozessschritte

```
Value = values("Name")
```

Verwandte Themen

- Verwenden von prozesslokalen Variablen und globalen Variablen für die Prozessdefinition auf Seite 272
- Abfrage von globalen Variablen des Session-Objektes auf Seite 388

Verwendung von Session Services

Das `Session`-Objekt ist die Instanz, welche die Daten zu einer Benutzersitzung zur Verfügung stellt. Dazu gehören der angemeldete Benutzer, seine Berechtigungsgruppen und seine Programmfunktionen. Des Weiteren stellt das `Session`-Objekt verschiedene Services zum Zugriff auf Daten bereit. Die angebotenen Services des `Session`-Objektes werden über ein generisches Interface (`Resolve (Of Service)()`) zur Verfügung gestellt. Nachfolgend werden Beispiele für häufig verwendete Services erläutert.

> HINWEIS: Die vollständige Beschreibung aller Services entnehmen Sie der Dokumentation zur `VI.DB.DLL`.

Detaillierte Informationen zum Thema

- Abfrage von Konfigurationsparametern auf Seite 386
- Testen auf die Existenz eines bestimmten Datenbank-Eintrages auf Seite 387
- Abfrage von globalen Variablen des Session-Objektes auf Seite 388

Abfrage von Konfigurationsparametern

Bei der Abfrage von Konfigurationsparametern muss immer der gesamte Pfad zum Konfigurationsparameter angegeben werden.

Syntax

```
Session.Config().GetConfigParm("<vollständiger Pfad>")
```

Wenn in einer Generierungsbedingung in VB.Net-Syntax ein Konfigurationsparameter getestet wird, so liefert die Funktion eine Zeichenkette. Soll dieser Wert gegen einen numerischen Wert verglichen werden, so klappt das nur so lange, wie der Konfigurationsparameter aktiv ist und einen numerischen Inhalt hat. Das liegt an der impliziten Wertart-Konvertierung von VB.Net. Ist der Konfigurationsparameter nicht aktiv, so liefert die Funktion eine leere Zeichenkette (""), die nicht gegen einen numerischen Wert verglichen werden kann, was in einen Laufzeitfehler von VB.Net mündet. Konfigurationsparameterwerte werden daher immer gegen Zeichenketten verglichen.

Nicht zu verwenden:

```
Session.Config().GetConfigParm("QER\Person\User\DeleteOptions\Homedir")=1
```

Stattdessen zu verwenden:

```
Session.Config().GetConfigParm("QER\Person\User\DeleteOptions\Homedir")="1"
```

Um sicher zu gehen, dass immer ein logischer Wert geliefert wird, sollte die Funktion `VID_IsTrue` verwendet werden.

Beispiel

```
If VID_IsTrue(Session.Config().GetConfigParm
("QER\Person\User\DeleteOptions\Homedir")) Then ...
```

Verwandte Themen

- Testen auf die Existenz eines bestimmten Datenbank-Eintrages auf Seite 387
- Abfrage von globalen Variablen des Session-Objektes auf Seite 388

Testen auf die Existenz eines bestimmten Datenbank-Eintrages

HINWEIS: Der Test sollte ohne Beachtung von Bearbeitungsberechtigungen erfolgen.

Syntax

```
Session.Source().Exists("<Tablename>","<WhereClause>")
```

Beispiel

```
Session.Source().Exists("Person", "CentralAccount = '" & accnt & "' and uid_person
<> '" & uid_person & "'")
```

Verwandte Themen

- Abfrage von Konfigurationsparametern auf Seite 386
- Abfrage von globalen Variablen des Session-Objektes auf Seite 388

Abfrage von globalen Variablen des Session-Objektes

Globale Variablen werden durch das einstellende Programm besetzt. Zusätzlich zu den vordefinierten Variablen können alle Umgebungsvariablen und alle am `Session`-Objekt definierten kundenspezifischen Variablen verwendet werden. Kundenspezifische `Session`-Variablen können beispielsweise über Skripte, Methoden oder Customizer definiert werden.

> HINWEIS: Wird eine kundenspezifische `Session`-Variable definiert, dann sollte sie anschließend wieder entfernt werden. Andernfalls bleibt sie bei Nachnutzung der Session bestehen und es werden unter Umständen falsche Prozesse generiert.

Syntax

```
Variables("<Variablenname>")
```

Beispiel zur Verwendung in den Parametern der Prozessschritten

```
Value = Variables("GENPROCID")
Value = CBool(Session.Variables("FULLSYNC"))
```

Tabelle 133: Zulässige vordefinierte globale Variablen

Variable	Bedeutung
EnvUserName	Name des Benutzers in der Umgebung, gegen die authentifiziert wurde, beispielsweise **Domain\User** bei Active Directory.
FullSync	Die Variable wird von allen Synchronisatoren gesetzt. Werte sind **True** und **False**.
GenProcID	Unikale ID-Nummer für den Prozess.
LogonUser	`DialogUser.Username` des angemeldeten Benutzers.
DialogUserUID	`DialogUser.UID_DialogUser` des angemeldeten Benutzers.
UserName	Der Name, der in `XUserInserted` bzw. `XUserUpdated` abgebildet wird.
UserUID	Die `UID_Person` des angemeldeten Benutzers, falls ein

Variable	Bedeutung
	personenbezogenes Authentifizierungsmodul benutzt wurde.
ShowCommonData	Gibt an, ob Systemdaten angezeigt (Wert = **1**) oder nicht angezeigt (Wert = **0**) werden. Die Variable wird im Designer über die Programmeinstellung **Systeminformationen anzeigen** ausgewertet.
SessionType	Gibt an, ob eine direkte Datenbankverbindung oder eine Verbindung über Anwendungsserver unterstützt wird. Nur direkte Datenbankverbindung: `'%SessionType%' = 'Direct'` Nur Verbindung über Anwendungsserver: `'%SessionType%' = 'AppServer'`
Feature_<Featurename>	Abfrage zusätzlicher Programmfunktionen (`QBMFeature`), die für den Benutzer freigeschaltet sind. Der Wert ist **1**, wenn die Programmfunktion verfügbar ist, ansonsten ist die Variable nicht gesetzt.
ManageOutstandingOperation	Die Variable wird bei der Zielsystemsynchronisation zur Unterscheidung der auszuführenden Operationen bei der Nachbehandlung ausstehender Objekte. Zulässige Werte sind **Delete**, **DeleteState** und **Publish**.

Verwandte Themen

- Abfrage von Konfigurationsparametern auf Seite 386
- Testen auf die Existenz eines bestimmten Datenbank-Eintrages auf Seite 387

Verwendung der #LD-Notation

Die #LD-Notation wird zur sprachabhängigen Abbildung von Informationen genutzt. Die #LD-Notation wird vorrangig in der Prozessverfolgung und der Benachrichtigung zur Prozessverarbeitung eingesetzt, kann aber auch in Skripten, die in der Skriptbibliothek abgelegt sind, verwendet werden.

Syntax

```
Value=#LD[<Sprache>|<Sprachkultur>](<Schlüssel>,{<Parameter>}*)#
```

wobei:

`<Sprache>|<Sprachkultur>`	(Optional) Sprache oder Sprachkultur für die Ausgabe.
`<Schlüssel>`	Basiszeichenkette mit Platzhaltern. Die Syntax der Platzhalter entspricht einem Format-Platzhalter in VB.Net (`{0}` bis `{9}`).
`<Parameter>`	Parameter für die Ersetzung der Platzhalter (kommagetrennt).

Tabelle 134: Verwendung der #LD-Notation

Kontext	Tabelle.Spalte	Bemerkungen
Prozessverfolgung	Job.ProcessDisplay	Abbildung auf DialogProcessStep.DisplayName
	JobChain.ProcessDisplay	Abbildung auf DialogProcessChain.DisplayName
	JobEventgen.ProcessDisplay	Abbildung auf DialogProcess.DisplayName
Benachrichtigung zur Prozessverarbeitung	Job.NotifyAddress	
	Job.NotifyAddressSuccess	
	Job.NotifyBody	
	Job.NotifyBodySuccess	
	Job.NotifySender	
	Job.NotifySenderSuccess	
	Job.NotifySubject	
	Job.NotifySubjectSuccess	
	JobRunParameter.ValueTemplate	Nur in der Prozesskomponente `MailComponent`.
Bildungsregeln	DialogColumn.Template	
	DialogColumn.CustomTemplate	
Formate	DialogColumn.FormatScript	
	DialogColumn.CustomFormatScript	
Methodendefinitionen	DialogMethod.MethodScript	

Kontext	Tabelle.Spalte	Bemerkungen
Einfügewerte	DialogObject.InsertValues	
	DialogTable.InsertValues	
	DialogTree.ListInsertValues	
	DialogSheet.InsertValues	
Auswahlskripte	DialogTable.SelectScript	
	DialogObject.SelectScript	
Skripte für die Prozessgenerierung	Job.GenCondition	
	Job.PreCode	
	Job.ServerDetectScript	
	JobChain.GenCondition	
	JobChain.PreCode	

Verwandte Themen

- Verwendung der #LD-Notation in der Prozessverfolgung auf Seite 391
- Beispiel für die Angabe der Sprache oder der Sprachkultur auf Seite 392

Verwendung der #LD-Notation in der Prozessverfolgung

Voraussetzung für die sprachabhängige Darstellung der Prozessinformationen ist die Definition einer entsprechenden Vorlage für die Ausgabetexte in den möglichen aktiven Sprachen.

Die Vorlagen für sprachabhängige Ausgabetexte werden bei der Kompilierung der Skripte in die Tabelle `DialogMultiLanguage` eingetragen. In dieser Tabelle wird ein Schlüssel (Spalte `Entrykey`), die Sprache und die sprachabhängige Ersetzung (Spalte `EntryValue`) eingetragen. Der Schlüssel sollte dabei dem Wert in der Standardsprache entsprechen. Ist für eine Sprache keine Vorlage hinterlegt, so wird der Schlüssel als Ausgabetext verwendet. Um die Vorlagen in weitere Sprachen zu übersetzten, verwenden Sie den Wörterbucheditor.

Beispiel:

Für eine Person wird eine Änderung vorgenommen. Die sprachabhängige Bildung der Prozessinformation könnte folgendermaßen formuliert werden.

- Bildungsvorschrift für die Prozessinformation am Ereignis `Update`

  ```
  Value = #LD("Change of properties of person {0}.", $InternalName$)#
  ```

- Vorlagen für die Ausgabetexte in der Tabelle `DialogMultiLanguage`

Schlüssel	Sprache	Wert
Change of properties of person {0}.	English - United States [en-US]	Change of properties of person {0}.
Change of properties of person {0}.	German - Germany [de-DE]	Änderung der Daten der Person {0}.

Mit einem `InternalName = JBasset` ergeben sich in der Prozessansicht folgende Ausgabetexte.

Sprache des angemeldeten Benutzers	Ausgabetext in der Prozessansicht
English - United States [en-US]	Change of properties of person JBasset.
German - Germany [de-DE]	Änderung der Daten der Person JBasset.

Verwandte Themen

- Übersetzungen im Wörterbucheditor anzeigen auf Seite 238

Beispiel für die Angabe der Sprache oder der Sprachkultur

Die #LD-Notation unterstützt die Angabe einer Sprache oder einer Sprachkultur. Dies ist besonders in den Fällen sinnvoll, in denen Anwender eine Nachricht des Systems in ihrer bevorzugten Sprache erhalten sollen.

Beispiele

- Ausgabe in der Standardsprache

```
Value = #LD("Test: {0}", <Parameter>)#
Value = #LD[""]("Test: {0}", <Parameter>)#
```

- Ausgabe immer auf englisch

```
Value = #LD["en-US"]("Test: {0}", <Parameter>)#
Value = #LD["english"]("Test: {0}", <Parameter>)#
```

- Nutzung einer Variable

```
Dim lang As String = "en-US"
Value = #LD[lang]("Test: {0}", <Parameter>)#
```

Die Angabe der Sprache in eckigen Klammern ist optional. Wichtig ist, dass der Sprachausdruck immer ein `String`-Ausdruck sein muss. Ist die Sprache nicht angegeben oder ergibt der `String`-Ausdruck einen Leerstring oder **Nothing**, wird zur Übersetzung die derzeit eingestellte Sprache der Anwendung verwendet.

Skriptbibliothek

Die Skriptbibliothek enthält den Quellcode der im One Identity Manager verwendeten Skripte. Die mitgelieferten Standardskripte sind nicht bearbeitbar. Während der Schemainstallation werden diese Skripte, auch bei Verwendung in kundenspezifischen Skripten, überschrieben.

> HINWEIS: Umfangreiche Beispiele für die Syntax und die Verwendung von Skripten finden Sie auf dem Installationsmedium im Verzeichnis `QBM\dvd\AddOn\SDK\ScriptSamples`. Beispiele für Unittests finden Sie unter `QBM\dvd\AddOn\SDK\UnitTestSamples`.

Die Skripte werden im Designer in der Kategorie **Skriptbibliothek** abgebildet. Auf dem Überblicksformular eines Skriptes erhalten Sie Informationen über dessen Verwendung beispielsweise in Spaltendefinitionen, Prozessen oder anderen Skripten.

Skripte erstellen, bearbeiten und testen Sie mit dem Skripteditor. Um umfangreichere Debug- und Bearbeitungsmöglichkeiten des Visual Studio zu nutzen, können die Skripte im den System Debugger bearbeiten und testen.

> HINWEIS: Skripte für Synchronisationsprojekte aus der Skriptbibliothek des Synchronization Editors stehen im Designer nicht zur Verfügung. Ausführliche Informationen zur Skriptbibliothek des Synchronization Editors finden Sie im *One Identity Manager Referenzhandbuch für die Zielsystemsynchronisation*.

Detaillierte Informationen zum Thema

- Unterstützung bei der Bearbeitung von Skripten im Skripteditor auf Seite 394
- Skripte mit dem Skripteditor erstellen und bearbeiten auf Seite 401

- Skripte im Skripteditor kopieren auf Seite 402
- Kompilieren eines Skriptes im Skripteditor testen auf Seite 404
- Skripte im Skripteditor testen auf Seite 403
- Überschreiben von Skripten auf Seite 404
- Berechtigungen zu Ausführen von Skripten auf Seite 405
- Bearbeiten und Testen von Skriptcode mit dem System Debugger auf Seite 406
- Erweitertes Debugging im Object Browser auf Seite 413

Unterstützung bei der Bearbeitung von Skripten im Skripteditor

Für die Bearbeitung von Skripten wird ein spezielles Eingabefeld verwendet. Das Eingabefeld besitzt einen erweiterten Bearbeitungsmodus, in dem zusätzliche Aktionen möglich sind.

Um in den erweiterten Bearbeitungsmodus zu wechseln

- Verwenden Sie **Strg + Alt + Eingabe** oder die Schaltfläche am rechten unteren Rand.

Abbildung 31: Direkte Eingabe einer Datenbankabfrage

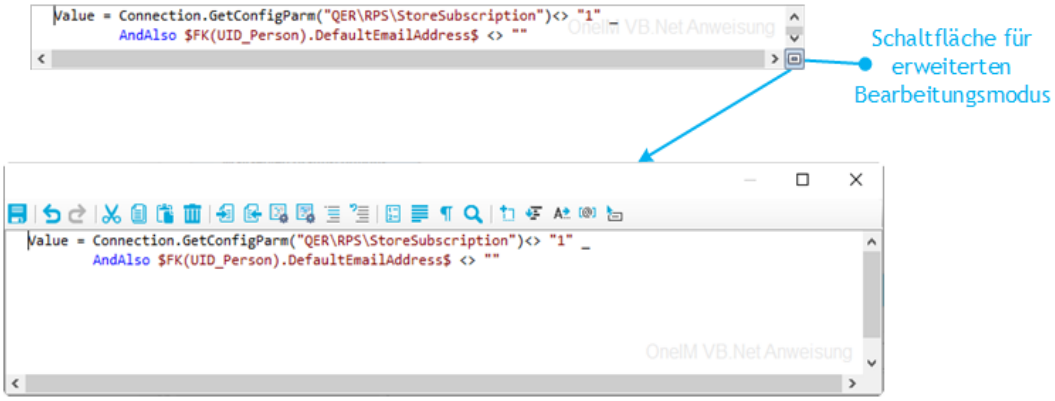


Tabelle 135: Bedeutung der Symbole des erweiterten Bearbeitungsmodus

Symbol	Bedeutung
	Beenden des erweiterten Bearbeitungsmodus.
	Zurücksetzen der letzten Änderung.
	Wiederherstellen der letzten Änderung.

Symbol	Bedeutung
	Ausschneiden des markierten Codes.
	Kopieren des markierten Codes in die Zwischenablage.
	Einfügen des Codes aus der Zwischenablage.
	Löschen des markierten Codes.
	Einzug verkleinern.
	Einzug vergrößern.
	Text automatisch formatieren.
	Textauswahl automatisch formatieren.
	Zeilennummern einblenden oder ausblenden.
	Kommentarzeichen einfügen.
	Kommentarzeichen entfernen.
	Zeilennummer einfügen oder entfernen.
	Automatischen Zeilenumbruch einfügen oder entfernen.
	Leerzeichen und Tabstoppzeichen einblenden oder ausblenden.
	Suchen innerhalb des Codes.
	Code-Ausschnitt einfügen.
	Liste der Objekteigenschaften anzeigen.
	Liste mit Wortvervollständigung anzeigen.
	Liste mit Parameterinformationen anzeigen.
	Zusätzliche Informationen anzeigen.

Tabelle 136: Tastenkombinationen für die Bearbeitung von Skripten

Tastenkombination	Beschreibung
Strg + C	In die Zwischenablage kopieren.
Strg + Einfg	In die Zwischenablage kopieren.
Strg + X	Ausschneiden und in die Zwischenablage übernehmen.
Umschalt + Entf	Ausschneiden und in die Zwischenablage übernehmen.
Strg + L	Zeile ausschneiden und in die Zwischenablage übernehmen.
Strg + V	Aus der Zwischenablage einfügen.
Umschalt + Einfg	Aus der Zwischenablage einfügen.

Tastenkombination	Beschreibung
Strg + Y	Aktion wiederholen.
Strg + Umschalt + Z	Aktion wiederholen.
Strg + Z	Aktion rückgängig machen.
Rücktaste	Zeichen vor dem Cursor entfernen.
Umschalt + Rücktaste	Zeichen vor dem Cursor entfernen.
Strg + Rücktaste	Wort vor dem Cursor entfernen.
Entf	Zeichen am Cursor entfernen.
Strg + Umschalt + L	Zeile entfernen.
Strg + Entf	Zeichen bis zum nächsten Wort entfernen.
Eingabe	Zeilenumbruch einfügen.
Umschalt + Eingabe	Zeilenumbruch einfügen.
Strg + Eingabe	Zeile oberhalb einfügen.
Strg + Umschalt + Eingabe	Zeile unterhalb einfügen.
Strg + Leertaste	Automatische Vervollständigung.
Strg + Umschalt + Leertaste	Liste mit Parameterinformationen anzeigen.
Tab	Einzug/Tabstopp einfügen.
Umschalt + Tab	Einzug/Tabstopp entfernen.
Strg + U	Markierte Zeichen in Kleinbuchstaben ändern.
Strg + Umschalt + U	Markierte Zeichen in Großbuchstaben ändern
Einfg	Überschreibmodus einschalten oder ausschalten.
Strg + T	Zeichen vor und nach dem Cursor tauschen.
Strg + Umschalt + T	Wörter vor und nach dem Cursor tauschen.
Umschalt + Alt + T	Zeile mit der vorherigen Zeile tauschen.
Alt + Bild-Auf	Zeile nach oben bewegen.
Alt + Bild-Ab	Zeile nach unten bewegen.
Bild-Ab	Nach oben bewegen.
Bild-Auf	Nach unten bewegen.
Nach-Links-Taste	Nach links bewegen.
Nach-Rechts-Taste	Nach rechts bewegen.

Tastenkombination	Beschreibung
Strg + Nach-Links-Taste	Zum vorherigen Wort bewegen.
Strg + Nach-Rechts-Taste	Zum nächsten Wort bewegen.
Pos1	Zum Zeilenanfang bewegen.
Ende	Zum Zeilenende bewegen.
Strg + Pos1	Zum Anfang des Skripts bewegen.
Strg + Ende	Zum Ende des Skripts bewegen.
Bild-Auf	Seite nach oben bewegen.
Bild-Ab	Seite nach unten bewegen.
Strg + Bild-Auf	Zum sichtbaren Anfang bewegen.
Strg + Bild-Ab	Zum sichtbaren Ende bewegen.
Strg +]	Zur letzten passenden Klammer bewegen. (nicht auf deutscher Tastatur erreichbar)
Strg + Nach-Unten-Taste	Nach unten scrollen.
Strg + Nach-Oben-Taste	Nach oben scrollen.
Strg + F	Suchendialog öffnen.
F3	Weitersuchen.
Strg + F3	Vorwärts suchen.
Umschalt + F3	Rückwärts suchen.
Strg + Umschalt + F3	Vorherige Auswahl finden.
Strg + H	Ersetzen.
Strg + I	Inkrementelle Suche vorwärts.
Strg + Umschalt + I	Inkrementelle Suche rückwärts.
Strg + Umschalt + Num-	Auswahl des Code-Blocks verringern.
Strg + Umschalt + Num+	Auswahl des Code-Blocks erweitern.
Escape	Auswahl entfernen.
Umschalt + Nach-Unten-Taste	Auswahl nach unten erweitern.
Umschalt + Nach-Oben-Taste	Auswahl nach oben erweitern.
Umschalt + Nach-Links-	Auswahl nach links erweitern.

Tastenkombination	Beschreibung
Taste	
Umschalt + Nach-Rechts-Taste	Auswahl nach rechts erweitern.
Strg + Umschalt + Nach-Links-Taste	Auswahl bis zum vorherigen Wort erweitern.
Strg + Umschalt + Nach-Rechts-Taste	Auswahl bis zum nächsten Wort erweitern.
Umschalt + Pos1	Auswahl bis zum Zeilenanfang erweitern.
Umschalt + Ende	Auswahl bis zum Zeilenende erweitern
Strg + Umschalt + Pos1	Auswahl bis zum Anfang des Skriptes erweitern.
Strg + Umschalt + Ende	Auswahl bis zum Ende des Skriptes erweitern.
Umschalt + Bild-Auf	Auswahl eine Seite nach oben erweitern.
Umschalt + Bild-Ab	Auswahl eine Seite nach unten erweitern.
Strg + Umschalt + Bild-Auf	Auswahl bis zum sichtbaren Anfang erweitern.
Strg + Umschalt + Bild-Ab	Auswahl bis zum sichtbaren Ende erweitern.
Strg + A	Alles auswählen.
Strg + Umschalt + W	Wort auswählen.
Strg + Umschalt +]	Auswahl bis zur nächsten Klammer erweitern. (nicht auf deutscher Tastatur erreichbar)
Umschalt + Alt + Nach-Unten-Taste	Ausgewählten Block nach unten erweitern.
Umschalt + Alt + Nach-Oben-Taste	Ausgewählten Block nach oben erweitern.
Umschalt + Alt + Nach-Links-Taste	Ausgewählten Block nach links erweitern.
Umschalt + Alt + Nach-Rechts-Taste	Ausgewählten Block nach rechts erweitern.
Strg + Umschalt + Alt + Nach-Links-Taste	Ausgewählten Block bis zum vorherigen Wort erweitern.
Strg + Umschalt + Alt + Nach-Rechts-Taste	Ausgewählten Block bis zum nächsten Wort erweitern.
Strg + Mausrad	Schriftgröße anpassen.

Für die Erstellung des Skriptcodes werden zusätzliche Eingabehilfen angeboten.

Syntaxhervorhebung

Die Eingabefelder unterstützen abhängig vom Syntaxtyp die Syntaxhervorhebung.

Automatische Vervollständigung

Bei der Erstellung von Skriptcode kann die automatische Vervollständigung genutzt werden. Durch die Einblendung verwendbarer Bezeichnungen von Eigenschaften oder Funktionen, wird die Menge des manuell einzutragenden Skriptcodes verringert. Die automatische Vervollständigung könne Sie über die Tastenkombination **Strg + Leertaste** an geeigneten Stellen der Eingabefelder aufrufen. Welche Inhalte zur Auswahl angeboten werden, wird über Schlüsselwörter im Skriptcode bestimmt.

Eingabe von Code-Ausschnitten

Eingabefelder, die den Syntaxtyp VB.Net erfordern, unterstützen die Eingabe von Code-Ausschnitten. In der Kategorie **Visual Basic** werden allgemeine Code-Ausschnitte angeboten. Die Kategorie **Object Layer** enthält spezielle Code-Ausschnitte für die One Identity Manager-Objektschicht. In der Kategorie **SQL Formatter** werden Code-Ausschnitte für Datenbankabfragen aus VB.Net heraus.

Code-Ausschnitte können Sie über folgende Varianten einfügen:

1. Eingabe über Symbol
   a. Wählen Sie in der Menüleiste das Symbol .
   b. Wählen Sie die Kategorie **Object Layer**, **SQL Formatter** oder **Visual Basic**.
   c. Wählen Sie den Code-Ausschnitt aus.
2. Eingabe über Tastenkombination
   a. Drücken Sie die Taste **F2**.
   b. Wählen Sie die Kategorie **Object Layer**, **SQL Formatter** oder **Visual Basic**.
   c. Wählen Sie den Code-Ausschnitt aus.
3. Einfügen über Verknüpfungsnamen
   a. Geben Sie einen Verknüpfungsnamen ein.
   b. Über **Tab** wird der Code-Ausschnitt eingefügt.

   HINWEIS: Beachten Sie bei der Eingabe der Verknüpfungsnamen die Groß- und Kleinschreibung.

HINWEIS: Bei direkter Auswahl eines Code-Ausschnittes über Tastenkombination oder über das Symbol werden eine Kurzbeschreibung sowie der Verknüpfungsname als Tooltip angezeigt.

Kundendefinierte Code-Ausschnitte

Sie können kundenspezifische Code-Ausschnitte verwenden. Erstellen Sie dazu im Installationsverzeichnis des One Identity Manager ein Verzeichnis `CustomSnippets`, in welches Sie die Dateien für die Code-Ausschnitte einfügen. Für die Entwicklung eigener Code-Ausschnitte nutzen Sie die Dokumentation von Visual Studio.

Für eine Sortierung der kundenspezifische Code-Ausschnitte verwenden Sie in der Snippet-Datei im Titel der Code-Ausschnitte eine Sortierung in folgender Syntax. Bei der Anzeige währender der Eingabe eine Code-Ausschnitts wird die Nummerierung unterdrückt.

```
<Title>(1) Ihr Titel für den Code-Ausschnitt</Title>
<Title>(2) Ihr Titel für den Code-Ausschnitt</Title>
```

Eingabe von Werten in $-Notation

In Eingabefeldern, die einen VB.Net-Ausdruck erwarten, wird bei Eingabe von **$** eine Eingabehilfe geöffnet. Alle Eigenschaften des aktuellen Objektes werden angezeigt. Zusätzlich wird ein Tooltip mit einer näheren Beschreibung der Eigenschaft eingeblendet. Wenn Sie eine Fremdschlüssel (FK)-Spalte auswählen, können Sie mit den Pfeiltasten zu den Spalten der zugehörigen Tabelle navigieren. Beenden Sie die Auswahl auf der Zielspalte mit **Eingabe** oder Doppelklick. In der Eingabespalte wird nun die vollständige $-Notation für Ihre Auswahl angezeigt. Um die Eingabehilfe ohne Datenübernahme zu schließen, verwenden Sie **Esc** oder verlassen Sie das Eingabefeld.

Abbildung 32: Eingabehilfe für $-Notation

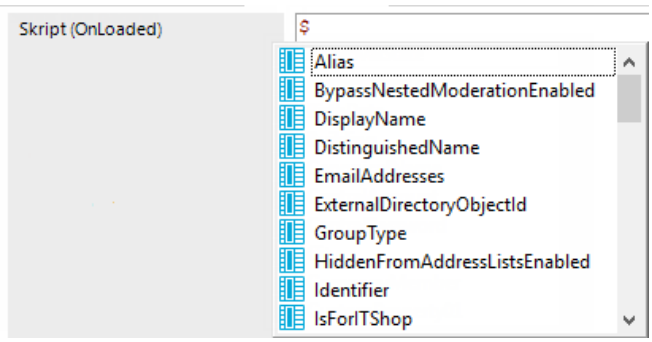


Tabelle 137: Bedeutung der verwendeten Symbole in der Eingabehilfe für $-Notation

Symbol	Bedeutung
	Eigenschaft des aktuellen Objektes.
	Primärschlüssel (PK).
	Fremdschlüssel (FK).
	Dynamischer Fremdschlüssel.
	Tabelle.
	Spezielle Eigenschaften.
	Skript.

Tabelle 138: Funktionen der Eingabehilfe für die $-Notation

Tastenkombination	Aktion
Nach-Unten-Taste	Öffnen der Eingabehilfe.
Nach-Oben-Taste, Nach-Unten-Taste	Wechsel zum vorherigen oder nächsten Eintrag.
Nach-Links-Taste, Nach-Rechts-Taste	Wechsel über Fremdschlüssel zum übergeordneten Objekt oder zum untergeordneten Objekt.
Eingabe	Übernahme des Wertes in $-Notation.

Skripte mit dem Skripteditor erstellen und bearbeiten

WICHTIG: Nach der Erstellung und Bearbeitung eines Skriptes, testen Sie die Kompilierung des Skriptes. Damit ein Skript wirkt, kompilieren Sie die Skripte der Skriptbibliothek.

HINWEIS: Umfangreiche Beispiele für die Syntax und die Verwendung von Skripten finden Sie auf dem Installationsmedium im Verzeichnis `QBM\dvd\AddOn\SDK\ScriptSamples`.

Um ein neues Skript zu erstellen

1. Wählen Sie im Designer die Kategorie **Skriptbibliothek**.
2. Starten Sie den Skripteditor über die Aufgabe **Ein neues Skript erstellen**.
3. Bearbeiten Sie die Stammdaten.

Tabelle 139: Stammdaten des Skript

Eigenschaft	Beschreibung
Skript	Name des Skriptes. Kennzeichnen Sie kundenspezifische Skripte mit dem Präfix **CCC_**.
Beschreibung	Nähere Beschreibung zur Funktion des Skriptes
Skriptcode	Die One Identity Manager-Skripte werden in VB.Net-Syntax abgelegt, was die Nutzung aller VB.Net-Funktionen zulässt. Die zu verarbeitenden Werte werden als Präprozessoranweisungen angegeben.
Gesperrt	Gibt an, ob das Skript gesperrt ist und nicht verwendet werden darf. Eine Sperrung ist beispielsweise sinnvoll, wenn das Skript noch in Bearbeitung ist.

Um ein Skript zu bearbeiten

1. Wählen Sie im Designer in der Kategorie **Skriptbibliothek** das Skript.
2. Wählen Sie die Aufgabe **Skript bearbeiten**.
3. Bearbeiten Sie die Stammdaten des Skriptes.

Detaillierte Informationen zum Thema

- Skripte im Skripteditor kopieren auf Seite 402
- Überschreiben von Skripten auf Seite 404
- Skripte im Skripteditor testen auf Seite 403
- Verwendung von Visual Basic .NET-Skripten auf Seite 375

Skripte im Skripteditor kopieren

WICHTIG: Nach der Erstellung und Bearbeitung eines Skriptes, testen Sie die Kompilierung des Skriptes. Damit ein Skript wirkt, kompilieren Sie die Skripte der Skriptbibliothek.

Um ein vorhandenes Skript zu kopieren

1. Wählen Sie im Designer in der Kategorie **Skriptbibliothek** das Skript, welches Sie kopieren möchten.
2. Wählen Sie die Aufgabe **Skript kopieren**.
3. Prüfen Sie im Dialog **Skript kopieren** die folgenden Informationen und passen Sie diese bei Bedarf an.

   Tabelle 140: Kopieren eines Skriptes

Eigenschaft	Beschreibung
Alter Skript-name	Name des kopierten Skriptes.
Skript	Der Name des neuen Skriptes wird aus dem Präfix **CCC_** und dem Namen des alten Skriptes gebildet. Sie können den Namen anpassen. Kennzeichnen Sie kundenspezifische Skripte mit dem Präfix **CCC_**.
Skriptcode	Der Skriptcode des originalen Skriptes wird übernommen. Bei Bedarf können Sie den zu kopierenden Skriptcode bereits anpassen.

4. Um die Kopie zu erstellen, klicken Sie **OK**.
5. Bearbeiten Sie im Skripteditor die Stammdaten des Skriptes.

Verwandte Themen

- Skripte mit dem Skripteditor erstellen und bearbeiten auf Seite 401

Skripte im Skripteditor testen

Zum Testen eines Skriptes können Sie den Skripteditor verwenden.

Um ein Skript zu testen

1. Wählen Sie im Designer in der Kategorie **Skriptbibliothek** das Skript.
2. Wählen Sie die Aufgabe **Skript bearbeiten**.
3. Wählen Sie den Menüeintrag **Ansicht > Skript testen**.
4. Wählen Sie in der Ansicht **Skript testen** in der Auswahlliste das Skript und passen Sie die Parameter für das Skript an.

   Alle Übergabeparameter des Skriptes und deren Datentyp werden angezeigt und können mit Werten belegt werden. Zusätzlich können die Variablen `Base` und `Value` der Skript-Basisklasse als Eingabeparameter vorbelegt werden, um im Skript damit zu arbeiten.
5. Wählen Sie über die **Optionen** eine oder mehrere der folgenden Optionen zur Ausführung des Tests.
   - **Hauptdatenbankverbindung verwenden**: Gibt an, ob die Skript-Tests gegen die Hauptdatenbank oder gegen die interne SQLite-Datenbank getestet werden. Skripte, die sich auf den Anwendungsanteil des One Identity Manager-Datenmodells beziehen, sollten Sie immer gegen die Hauptdatenbank ausführen. Skripte für den Systemanteil können Sie gegen die Hauptdatenbank oder die interne SQLite-Datenbank testen.
   - **Transaktion verwenden**: Gibt an, ob die Ausführung des Skriptes innerhalb einer Transaktion mit anschließendem Rollback erfolgt oder ob das Skript direkt gegen die Datenbank ausgeführt wird.
   - **SQL Protokoll aufzeichnen**: Gibt an, ob bei der Ausführung des Skriptes die Datenbankaktionen im SQL Protokoll aufzuzeichnen sind. Die Ausgabe erfolgt in einem separaten Dialogfenster. Zusätzlich zum ausgeführten Statement wird die Ausführungszeit des Skriptes ausgegeben.
6. Starten Sie den Test über die Schaltfläche **Start**.

   Nach Ausführung eines Skriptes wird das Ergebnis im Eingabefeld **Ergebnis** angezeigt.

Verwandte Themen

- Bearbeiten und Testen von Skriptcode mit dem System Debugger auf Seite 406

Kompilieren eines Skriptes im Skripteditor testen

Haben Sie ein neues Skript erstellt, müssen Sie dieses Skript kompilieren. Erst durch die Kompilierung wird das Skript generierbar. Im Skripteditor können Sie die Skriptkompilierung testen.

Um die Skriptkompilierung zu testen

- Wählen Sie im Designer in der Kategorie **Skriptbibliothek** das Skript.
- Wählen Sie die Aufgabe **Skript bearbeiten**.
- Starten Sie den Kompiliervorgang über das Symbol , den Menüeintrag **Skript > Skript kompilieren** oder **F9**.

Bei der Kompilierung werden alle Skripte übersetzt. Bei der Kompilierung werden die Assemblies erstellt und lokal auf der Arbeitsstation, auf der die Generierung durchgeführt wird, abgelegt. Während der Übersetzung wird eine Gültigkeitskontrolle des Skriptcodes durchgeführt. Der Vorgang kann daher etwas Zeit in Anspruch nehmen.

Fehlermeldungen werden in der Ansicht **Compilerfehler** ausgegeben. Durch Maus-Doppelklick auf eine Fehlermeldung in der Ansicht wird im Skriptcode zum entsprechenden Eintrag gesprungen. Dieser kann hier angepasst werden.

WICHTIG: Nach dem Testen des Skriptes übernehmen Sie das Skript in die One Identity Manager-Datenbank und führen eine Skriptkompilierung mit dem Database Compiler durch. Ausführliche Informationen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Überschreiben von Skripten

Das Überschreiben von Skripten bietet sich an, wenn nur geringe kundenspezifische Änderungen an den Standardskripten erforderlich sind. Überschreibbare Skripte sind mit der Eigenschaft `overridable` gekennzeichnet.

HINWEIS: Es können nur die mitgelieferten Standardskripte überschrieben werden. Kundenspezifische Skripte können nicht überschrieben werden, da diese in einer Skriptklasse `Custom scripts` gespeichert werden.

Um ein Skript zu überschreiben

1. Wählen Sie im Designer in der Kategorie **Skriptbibliothek > Überschreibbare Skripte** das Skript, dass Sie überschreiben möchten.
2. Wählen Sie die Aufgabe **Skript kopieren**.
3. Passen Sie im Dialog **Skript kopieren** die folgenden Informationen an.

- **Skript**: Der Name des neuen Skriptes wird aus dem Präfix **CCC_** und dem Namen des alten Skriptes gebildet. Sie können den Namen anpassen. Kennzeichnen Sie kundenspezifische Skripte mit dem Präfix **CCC_**.

4. Um die Kopie zu erstellen, klicken Sie **OK**.
5. Im Skripteditor ersetzen Sie im Header des Skriptes die Eigenschaft `overridable` mit `overrides`.
6. Passen Sie den weiteren Skriptcode entsprechend kundenspezifisch an.

WICHTIG: Nach der Erstellung und Bearbeitung eines Skriptes, testen Sie die Kompilierung des Skriptes. Damit ein Skript wirkt, kompilieren Sie die Skripte der Skriptbibliothek.

Syntaxbeispiel

```
Public overridable Function My_Function() as Boolean
     'Skriptcode der originalen Funktion
End Function
Public overrides Function My_Function() as Boolean
     'Kundenspezifisch angepasster Skriptcode
End Function
```

Verwandte Themen

- Skripte mit dem Skripteditor erstellen und bearbeiten auf Seite 401
- Skripte im Skripteditor kopieren auf Seite 402
- Kompilieren eines Skriptes im Skripteditor testen auf Seite 404
- Skripte im Skripteditor testen auf Seite 403
- Verwendung von Visual Basic .NET-Skripten auf Seite 375

Berechtigungen zu Ausführen von Skripten

Die grundlegende Berechtigung zum Ausführen von Skripten erhält der angemeldete Benutzer über die Programmfunktion **Erlaubt das Ausführen von Skripten im Frontend** (`Common_StartScripts`).

Wird ein Skript zusätzlich mit einer Programmfunktion versehen (Tabelle `QBMScriptHasFeature`), so kann ein Benutzer dieses Skript nur noch ausführen, wenn er auch die nötige Programmfunktion über seine Berechtigungsgruppen besitzt. Besitzt der Benutzer die Programmfunktion nicht, so wird beim Ausführungsversuch eine Fehlermeldung geworfen.

Ausführliche Informationen zum Steuern von Berechtigungen zum Ausführen von Skripten über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Bearbeiten und Testen von Skriptcode mit dem System Debugger

Mit dem System Debugger haben Sie die Möglichkeit Skripte, Bildungsregeln, Formatierungsskripte, Methoden und Tabellenskripte zu testen. Es stehen Ihnen hier die Debug- und Bearbeitungsmöglichkeiten des Visual Studio zur Verfügung.

Um den System Debugger zu nutzen, muss folgende Software installiert sein:

- Visual Studio 2017 mit aktuellem Service Pack
- Microsoft .NET Framework Version 4.7.2 Developer Pack oder höher

HINWEIS: Um den System Debugger ohne das Starten des Visual Studio mit privilegierten Berechtigungen betreiben zu können, müssen die One Identity Manager-Komponenten lokal in einem Verzeichnis installiert werden, dass nicht über die Benutzerkontensteuerung verwaltet wird.

Detaillierte Informationen zum Thema

- Systembibliothek laden auf Seite 406
- Hinweise zum Bearbeiten von Skriptcode im System Debugger auf Seite 408
- Aufzeichnen der Datenbankabfragen und Objektaktionen auf Seite 409
- Testen von Skriptcode im System Debugger auf Seite 409
- Änderungen in die Datenbank speichern auf Seite 413

Systembibliothek laden

Beim Aufruf des System Debuggers wird im Visual Studio ein Solutiontemplate `SystemLibrary.sln` mit der Solution **SystemLibrary** zum Bearbeiten und Testen der Skripte geladen.

Innerhalb der Solution sind die folgenden Projekte definiert.

Tabelle 141: Projektdateien der Solution

Projekt	Skriptdatei	Beschreibung
Methods	Methods.vb	Die Skriptdatei enthält alle Methoden.

Projekt	Skriptdatei	Beschreibung
Scripts	VIScripts.vb	Die Skriptdatei enthält alle vordefinierten Skripte des Modellanteils.
	VIDScripts.vb	Die Skriptdatei enthält alle vordefinierten Skripte des Systemdatenanteils.
	CustomerScripts.vb	Die Skriptdatei enthält kundenspezifische Skripte. Hier werden neue Skripte eingefügt.
SystemDebugger	Main.vb	Das Startprojekt.
Tables	Tables.vb	Die Skriptdatei enthält alle Tabellenskripte.
Templates	Templates.vb	Die Skriptdatei enthält alle Bildungsregeln und Formatierungskripte.

Um die Systembibliothek zu laden

1. Führen Sie die Datei `SystemLibrary.sln` im One Identity Manager Installationsverzeichnis aus.
2. Prüfen Sie, ob das Projekt **SystemDebugger** im Visual Studio als Startprojekt eingetragen ist.
3. Starten Sie das Solution über **F5** im Visual Studio.
4. Stellen Sie eine Verbindung zur Datenbank her.
5. Prüfen Sie das Verzeichnis der Solutiondatei und die Optionen für die Erzeugung der Skriptbibliothek.

Optionen	Bedeutung
Systemskripte exportieren	Gibt an, ob vordefinierte Skripte in die Systembibliothek geladen werden.
Kundenskripte exportieren	Gibt an, ob kundenspezifische Skripte in die Systembibliothek geladen werden.
Gesperrte Skripte exportieren	Gibt an, ob nur die aktiven Skripte oder zusätzlich auch die gesperrten Skripte aus der Skriptbibliothek geladen werden.
Projektreferenzen aktualisieren	Gibt an, ob in Skripten verwendete Referenzen mit geladen werden.
Backups von existierenden Dateien herstellen	Gibt an, Sicherungen bestehender Dateien erzeugt werden

6. Klicken Sie **OK**.

Die Dateien der Systembibliothek werden mit den Informationen aus der Datenbank befüllt.

7. Bestätigen Sie im Visual Studio das Nachladen der einzelnen Projekte.
8. Starten Sie die Solution nochmals über **F5** im Visual Studio.

   Die generierten Quellen der Solution werden kompiliert.
9. Stellen Sie erneut die Verbindung zur Datenbank her.

   Der System Debugger wird gestartet.

Verwandte Themen

- Testen von Skriptcode im System Debugger auf Seite 409

Hinweise zum Bearbeiten von Skriptcode im System Debugger

Nach dem Laden der Systembibliothek können Sie die Skripte, Bildungsregeln, Formatierungsskripte, Methoden und Tabellenskripte im System Debugger bearbeiten und testen.

Beachten Sie Folgendes:

- Im Quellcode dürfen die `VI-KEY`-Kommentare nicht bearbeitet oder entfernt werden, da diese die einzelnen Codeblöcke kennzeichnen und für die Rücksicherung der Skripte in die Datenbank benötigt werden.
- Beim Laden der Bildungsregeln und Formatierungsskripte wird die `$`-Notation in einen Methodenaufruf `GetTriggerValue` konvertiert. Alle Aufruf der `GetTriggerValue`-Methode werden beim Speichern der Änderungen in die Datenbank in die `$`-Notation konvertiert.

  Beispiel:

  `$FK(UID_Person).IsExternal:Bool$` wird beim Laden konvertiert zu `GetTriggerValue ("FK(UID_Person).IsExternal").Bool`
- Zur Erstellung von Skripten nutzen Sie den Skripteditor im Designer. Im Skripteditor geben Sie den Namen des Skriptes und mindestens den Skript-Body an, damit der Export des Skriptes in die Systembibliothek erfolgen kann und das Skript mit dem System Debugger bearbeitbar ist.
- Bildungsregeln, Formatierungsskripte, Methoden und Tabellenskripte erstellen Sie im Designer. Nach dem Export in die Systembibliothek können Sie diese Elemente mit dem System Debugger bearbeiten.

Verwandte Themen

- Testen von Skriptcode im System Debugger auf Seite 409
- Änderungen in die Datenbank speichern auf Seite 413

Aufzeichnen der Datenbankabfragen und Objektaktionen

Nutzen Sie im System Debugger die Aufzeichnung der Datenbankabfragen und Objektaktionen für die Fehlersuche und die Optimierung der Skripte während der Entwicklung. Aufgezeichnet werden die Ausführungszeit und das ausgeführte Kommando.

- SQL Protokoll

  Öffnen Sie das Aufzeichnungsfenster über den Menüeintrag **Ansicht > SQL Protokoll**.

- Objektprotokoll

  Öffnen Sie das Aufzeichnungsfenster über den Menüeintrag **Ansicht > Objektprotokoll**.

Tabelle 142: Funktionen für die Aufzeichnung von Datenbankabfragen und Objektaktionen

Symbol	Bedeutung
	Die Aufzeichnung wird gestartet.
	Die Aufzeichnung wird gestoppt.
	Die Aufzeichnungen werden in die Zwischenablage kopiert.
	Die Aufzeichnungen werden in einer Datei gespeichert.
	Die Aufzeichnungen werden gelöscht.

Testen von Skriptcode im System Debugger

Mit dem System Debugger haben Sie die Möglichkeit Skripte, Bildungsregeln, Formatierungsskripte, Methoden und Tabellenskripte zu testen. Es stehen Ihnen hier die Debug- und Bearbeitungsmöglichkeiten des Visual Studio zur Verfügung.

Detaillierte Informationen zum Thema

- Skripte im System Debugger testen auf Seite 410
- Bildungsregeln und Formatierungsskripte im System Debugger testen auf Seite 411
- Methoden im System Debugger testen auf Seite 412
- Tabellenskripte im System Debugger testen auf Seite 412

Skripte im System Debugger testen

Um ein Skript zu testen

1. Wählen Sie im System Debugger im Bereich **Skripte** das Skript aus.
2. Sofern erforderlich, geben Sie die Werte für die Parameter des Skriptes an.
3. Prüfen Sie die Optionen für die Ausführung des Skriptes.
   - **Debug Methode verwenden**: Es wird in den Quellcode gesprungen. Somit können alle Debug-Möglichkeiten des Visual Studio genutzt werden.
   - **Base Eigenschaften definieren**: Die Variablen `Base` und `Value` der Skript-Basisklasse können als Eingabeparameter vorbelegt werden, um im Skript damit zu arbeiten.

     Beispiel:

     `Base` wird mit einem DB-Object-Key initialisiert um `base.GetValue ("SpaltenName").String` zu verwenden.
   - **Transaktion mit Rollback**: Mit dieser Option legen Sie fest, ob die Ausführung des Skriptes innerhalb einer Transaktion mit anschließendem Rollback erfolgt oder ob das Skript direkt gegen die Datenbank ausgeführt wird.
4. Wählen Sie **Start**.

   Die Ausführung des Skriptes wird gestartet. Nach Ausführung eines Skriptes werden das Ergebnis und die Ausführungszeit des Skriptes angezeigt.

TIPP: Um Skripte leichter zu finden, können Sie im Bereich **Skripte** die folgenden Funktionen nutzen.

- Um Skripte zu filtern, geben Sie im Eingabefeld **Skript suchen** die Zeichenkette ein, nach der gefiltert werden soll.
- Geänderte Skripte werden im System Debugger mit einem * gekennzeichnet.
- Um alle geänderten Skripte zu finden, klicken Sie auf 🔍 und verwenden Sie im Menü den Eintrag **Geänderte Skripte**.

Verwandte Themen

- Hinweise zum Bearbeiten von Skriptcode im System Debugger auf Seite 408
- Änderungen in die Datenbank speichern auf Seite 413
- Skripte im Skripteditor testen auf Seite 403

Bildungsregeln und Formatierungsskripte im System Debugger testen

Um eine Bildungsregel zu testen

1. Wählen Sie im System Debugger im Bereich **Bildungsregeln** die Spalte mit Bildungsregel aus.
2. Wählen Sie unter **Datenspalte für Bildungsregel**, die Spalte deren Bildungsregel Sie testen.
3. Wählen Sie unter **Datenbankobjekt** das Objekt, auf welches die Bildungsregel angewendet werden soll.
4. Prüfen Sie die Option **Transaktion mit Rollback** für die Ausführung der Bildungsregel.

   Mit dieser Option legen Sie fest, ob die Ausführung der Bildungsregel innerhalb einer Transaktion mit anschließendem Rollback erfolgt oder ob die Bildungsregel direkt gegen die Datenbank ausgeführt wird.
5. Wählen Sie eine der folgenden Aktionen zum Testen der Bildungsregel.

Aktion	Bedeutung
Speichern	Das Objekt wird gespeichert.
Verwerfen	Die Änderungen am Objekt werden verworfen.
Laden	Das Objekt wird erneut geladen.
Neu	Ein neues Objekt wird erzeugt.
Ausführen	Die Bildungsregel wird erneut ausgeführt.

Um ein Formatierungsskript zu testen

1. Wählen Sie im System Debugger im Bereich **Formate** die Spalte mit Formatierungsskript aus.
2. Wählen Sie unter **Datenbankobjekt** das Objekt, auf welches das Formatierungsskript angewendet werden soll.

Verwandte Themen

- Hinweise zum Bearbeiten von Skriptcode im System Debugger auf Seite 408
- Änderungen in die Datenbank speichern auf Seite 413

Methoden im System Debugger testen

Um eine Methode zu testen

1. Wählen Sie im System Debugger im Bereich **Dialogmethoden** die Methode aus.
2. Wählen Sie unter **Basisobjekt** das Objekt, auf welches die Methode angewendet werden soll.
3. Prüfen Sie die Option **Transaktion mit Rollback** für die Ausführung der Methode.

   Mit dieser Option legen Sie fest, ob die Ausführung der Methode innerhalb einer Transaktion mit anschließendem Rollback erfolgt oder ob die Methode direkt gegen die Datenbank ausgeführt wird.
4. Wählen Sie **Start**.

   Die Ausführung der Methode wird gestartet.

Verwandte Themen

- Hinweise zum Bearbeiten von Skriptcode im System Debugger auf Seite 408
- Änderungen in die Datenbank speichern auf Seite 413

Tabellenskripte im System Debugger testen

Um Tabellenskripte zu testen

1. Wählen Sie im System Debugger im Bereich **Tabellenskripte** die Tabelle mit Tabellenskript aus.
2. Wählen Sie unter **Datenbankobjekt** das Objekt, für welches die Ausführung der Tabellenskripte getestet wird.
3. Prüfen Sie die Option **Transaktion mit Rollback** für die Ausführung der Tabellenskripte.

   Mit dieser Option legen Sie fest, ob die Ausführung der Tabellenskripte innerhalb einer Transaktion mit anschließendem Rollback erfolgt oder ob die Tabellenskripte direkt gegen die Datenbank ausgeführt wird.
4. Wählen Sie eine der folgenden Aktionen zum Testen der Tabellenskripte.

Aktion	Bedeutung
Speichern	Das Objekt wird gespeichert. (`OnSaved`, `OnSaving`)
Verwerfen	Die Änderungen am Objekt werden verworfen. (`OnDiscarded`, `OnDiscarding`)
Laden	Das Objekt wird erneut geladen. (`OnLoaded`)
Neu	Ein neues Objekt wird erzeugt.

Verwandte Themen

- Hinweise zum Bearbeiten von Skriptcode im System Debugger auf Seite 408
- Änderungen in die Datenbank speichern auf Seite 413

Änderungen in die Datenbank speichern

Um Änderungen in die Datenbank zu speichern

1. Wählen Sie im System Debugger das Skript, die Bildungsregel, das Formatierungsskript, die Methode oder das Tabellenskript.
2. Wählen Sie das Menü **Skripte > Skript speichern**.

   Es wird ein Dialogfenster geöffnet, in welchem der Skriptname, das Datenbankobjekt, die Datenbankverbindung und der einzufügende Skriptcode dargestellt werden.
3. Wählen Sie unter **Änderungskennzeichen** ein Änderungskennzeichen unter dem Ihre Änderungen zusammengefasst werden.
4. Klicken Sie **Speichern**.

TIPP: Um mehrere Skripte im Bereich **Skripte** zum Speichern auszuwählen, halten Sie die **Strg**-Taste gedrückt, klicken Sie die Skripte an und wählen Sie das Menü **Skripte > Skript speichern**.

HINWEIS: Achten Sie darauf, nach Änderungen die Datenbank neu zu kompilieren.

Erweitertes Debugging im Object Browser

Der Object Browser unterstützt das Debuggen von Skripten, Bildungsregeln, Formatierungsskripten, Tabellenskripten, Prozessen und Methoden. Es stehen Ihnen hier die Debug-Möglichkeiten des Visual Studio zur Verfügung. Eine Bearbeitung von Skripten, Bildungsregeln, Formatierungsskripten, Tabellenskripten, Prozessen und Methoden ist nicht möglich. Korrigieren Sie Fehler im Designer.

Voraussetzungen

- Um die Debug-Funktion im Object Browser zu nutzen, muss folgende Software installiert sein:
  - Visual Studio 2017 mit aktuellem Service Pack
  - Microsoft .NET Framework Version 4.7.2 Developer Pack oder höher

- Der Benutzer benötigt die Programmfunktion **Erlaubt das Erstellen lokaler Debug-Assemblies** (`Common_CompileForDebug`). Der Benutzer hat im Configuration Wizard damit eine zusätzliche Kompilieroption zum Erzeugen lokaler Debug-Assemblies.

  Weisen Sie im Designer die Programmfunktion an eine kundenspezifische Berechtigungsgruppe zu und nehmen Sie den Systembenutzer in die Berechtigungsgruppe auf. Ausführliche Informationen Steuern von Berechtigungen über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.
- Auf der lokalen Arbeitsstation des Benutzers sind lokale Debug-Assemblies vorhanden.

Detaillierte Informationen zum Thema

- Lokale Debug-Assemblies erstellen auf Seite 414
- Im Object Browser debuggen auf Seite 414
- Fehlerbehebung für das Debugging im Object Browser auf Seite 416

Lokale Debug-Assemblies erstellen

Um lokale Debug-Assemblies zu erzeugen

1. Aktivieren Sie im Database Compiler auf der Seite **Kompiliervorgaben** die Option **Debuginformationen erzeugen**.
2. Wählen Sie die Kompilervorgabe **Skripte inkl. aller Abhängigkeiten**.
3. Starten Sie die Kompilierung.

   Während der Kompilierung werden zusätzlich Meldungen ausgegeben, die auf die lokale Erstellung der Debug-Assemblies hinweisen. Einige Kompilierschritte, wie beispielsweise die Kompilierung der Webprojekte, werden übersprungen, da diese nicht lokal debuggt werden können.

Der Database Compiler erzeugt auf dem lokalen Computer im Verzeichnis `%USERPROFILE%\AppData\Local\One Identity\One Identity Manager\AssemblyCache` die Assemblies, die zugehörigen PDB Dateien.

Der Quellcode wird im Verzeichnis `%USERPROFILE%\AppData\Local\One Identity\One Identity Manager\AssemblyCache\Sources` erzeugt.

Die Assemblies werden dabei nicht in die Datenbank übertragen.

Im Object Browser debuggen

HINWEIS: Das lokale Debuggen kann so lange verwendet werden bis auf der Arbeitsstation selbst Assemblies ohne Debug-Informationen erzeugt werden oder über die

Datenbankverbindung neuer Assemblies geladen werden.

Um im Object Browser zu debuggen

1. Starten Sie den Object Browser und wählen Sie das Menü **Debug > Debugger starten/stoppen**.

   Das Visual Studio gestartet und der Object Browser wird mit dem Debugger verbunden. Dieser Vorgang kann etwas Zeit in Anspruch nehmen. Ist die Verbindung zum Visual Studio hergestellt, wird Dialog **Breakpoints verwalten** geöffnet.

2. Im Dialog **Breakpoints verwalten** können Sie verschiedene Breakpoints für Skripte, Bildungsregeln, Formatskripten, Tabellenskripten, Prozessen und Methoden definieren.

   - Um einen neuen Breakpoint zu definieren, klicken Sie und wählen Sie eine der folgenden Optionen
     - **Skriptausführung**: Ein neuer Breakpoint vom Typ **Skript** wird hinzugefügt. Wählen Sie unter **Breakpoint-Operation** das Skript.
     - **Spaltenmethoden**: Ein Breakpoint vom Typ **Spalte** wird hinzugefügt. Wählen Sie unter **Breakpoint-Operation** das Formatierungsskript, die Bildungsregel oder das Skript zum bedingten Entfernen von Berechtigungen.
     - **Tabellenmethoden**: Ein neuer Breakpoint vom Typ **Tabellenskipt** wird hinzugefügt. Wählen Sie unter **Breakpoint-Operation** das Tabellenskript.
     - **Prozessgenerierung**: Ein Breakpoint vom Typ **Prozess** wird hinzugefügt. Wählen Sie unter **Breakpoint-Operation** den Prozess.
     - **Objektmethode**: Ein Breakpoint vom Typ **Objektmethode** wird hinzugefügt. Wählen Sie unter **Breakpoint-Operation** die Methodendefinition.
   - Um einen vorhandenen Breakpoint zu verwenden, wählen die den Breakpoint aus der Liste.
   - Um einen Breakpoint zu entfernen, wählen Sie den Breakpoint in der Liste und klicken Sie .
   - Um alle Breakpoints zu entfernen, klicken Sie .

3. Klicken Sie OK.

   Der Dialog **Breakpoints verwalten** wird geschlossen. Die Breakpoint-Definitionen werden an das Visual Studio übertragen.

   HINWEIS: Sie können den Dialog **Breakpoints verwalten** über das Menü **Debug > Breakpoints konfiguieren** wieder erreichen.

4. Führen Sie nun im Object Browser die Aktionen aus, die Sie debuggen möchten, beispielsweise das Skript aufrufen, die Bildungsregel einer Spalte ausführen oder den Prozess generieren.

   Sobald die Aktion ausgeführt werden soll, wird die Visual Studio Anwendung in den Vordergrund geholt und zeigt die Quellcodestelle mit dem gewählten Breakpoint. Ab

diesen Punkt stehen Ihnen die umfangreichen Debug-Möglichkeiten von Visual Studio im vollen Umfang zur Verfügung. Mit **F10** können Sie beispielsweise Zeile für Zeile den Quellcode durchlaufen und mit **F5** die Anwendung wieder fortsetzen.

5. Haben Sie das Debugging abgeschlossen, wählen Sie das Menü **Debug > Debugger starten/stoppen**.

   Das Visual Studio wird von Object Browser getrennt und geschlossen.

Verwandte Themen

- Lokale Debug-Assemblies erstellen auf Seite 414
- Fehlerbehebung für das Debugging im Object Browser auf Seite 416

Fehlerbehebung für das Debugging im Object Browser

Problem

Im Database Compiler wird die Option **Debuginformationen erzeugen** nicht angezeigt.

Ursache

Dem verwendeten Systembenutzer ist die Programmfunktion **Erlaubt das Erstellen lokaler Debug-Assemblies** (`Common_CompileForDebug`) nicht über seine Berechtigungsgruppen zugewiesen.

Lösung

Weisen Sie im Designer die Programmfunktion an eine kundenspezifische Berechtigungsgruppe zu und nehmen Sie den Systembenutzer in die Berechtigungsgruppe auf. Ausführliche Informationen Steuern von Berechtigungen über Programmfunktionen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Problem

Im Object Browser wird das Menü **Debug** nicht angezeigt.

Mögliche Ursachen

- Das Visual Studio ist nicht im erwarteten Umfang installiert.
- Die geladenen Assemblies enthalten keine Debug-Informationen.

Mögliche Lösungen

- Überprüfen Sie die Installation vom Visual Studio auf der lokalen Arbeitsstation. Weitere Informationen finden Sie unter Erweitertes Debugging im Object Browser auf Seite 413.
- Prüfen Sie die Datenbankverbindung. Die Debug-Assemblies gehören immer zu einer konkreten Datenbank. Bei Auswahl einer anderen Datenbankverbindung steht die Debug-Informationen nicht zur Verfügung.
- Prüfen Sie, ob eventuell bereits neuere Assemblies aus der Datenbank nachgeladen wurden. Das Änderungsdatum von DLL und PDB Dateien darf sich nicht unterscheiden.
- Kompilieren Sie gegebenenfalls die Assemblies erneut mit Debug-Informationen. Weitere Informationen finden Sie unter Lokale Debug-Assemblies erstellen auf Seite 414.

Problem

Die Breakpoints im Visual Studio werden als inaktiv angezeigt.

Ursache

Breakpoints werden als inaktiv angezeigt, wenn die Assembly mit der zu debuggenden Funktion noch nicht im Prozessraum geladen ist. Beispielsweise werden die Assemblies zur Prozessgenerierung erst zum Zeitpunkt der Generierung in die Anwendung geladen. Ab dann wird der Breakpoint als aktiv erkannt und angesprungen.

Problem

Die Breakpoints im Visual Studio werden immer als inaktiv angezeigt und werden auch nicht angesprungen.

Mögliche Ursachen

- Der Object Browser hat noch falsche Assemblies geladen.
- Der Object Browser konnte zu den Assemblies keine Debug-Informationen finden.

Mögliche Lösung

Ist das Visual Studio mit dem Object Browser verbunden, wechseln Sie in das verbundene Visual Studio und öffnen die Ansicht **Debug > Windows > Modules**. Hier finden Sie die Auflistung aller geladenen Modulen und Zusatzinformationen.

13

One Identity Manager Abfragesprache

Die One Identity Manager Abfragesprache (auch One Identity Manager Query Language) kann zum Erstellen von Abfragen oder Where-Klausel-Ausdrücke gegen die One Identity Manager-Objektschicht verwendet werden.

Die One Identity Manager Abfragesprache wird beispielsweise für die Kommunikation zwischen Anwendungsserver und Client genutzt. Sie können die One Identity Manager Abfragesprache derzeit im Object Browser im Abfragefenster verwenden.

Die Abfragesprache unterscheidet nicht zwischen Groß- und Kleinschreibung. Schlüsselwörter können in Groß- oder Kleinschreibung geschrieben werden. Um leichter erkennbar zu sein, werden Schlüsselwörter in den folgenden Beispielen immer in Großbuchstaben geschrieben.

Die Abfragesyntax in vielen Teilen an die SQL-Syntax angelehnt. Das macht es einfacher, einfache Where-Klauseln von SQL-Syntax in die Abfragesyntax zu konvertieren.

Detaillierte Informationen zum Thema

- Sprachelemente der One Identity Manager Abfragesprache auf Seite 418
- Formulieren von Abfragen in der One Identity Manager Abfragesprache auf Seite 423

Sprachelemente der One Identity Manager Abfragesprache

Die One Identity Manager Abfragesprache unterstützt verschiedene Sprachelemente.

Detaillierte Informationen zum Thema

- Kommentare auf Seite 419
- Bezeichner auf Seite 419

- Literale für Werte auf Seite 420
- Referenzen auf Parameter auf Seite 422
- Vorformatierte Where-Klauseln auf Seite 422

Kommentare

In der One Identity Manager Abfragesprache gibt es zwei Arten von Kommentaren, die analog zu Kommentaren in SQL-Syntax sind.

Zeilenkommentare

Zeilenkommentare beginnen mit der Markierung `--` und enden mit dem Zeilenende.

```
-- This is a line comment
```

Zeilenkommentare müssen nicht am Anfang der Zeile stehen.

```
FROM Person -- This is a line comment too
```

Blockkommentare

Blockkommentare beginnen mit den Zeichen `/*` und enden mit `*/`. Blockkommentare können sich über mehr als eine Zeile erstrecken.

```
/* This is a block comment
that spreads over
more than one line */
```

Bezeichner

In der One Identity Manager Abfragesprache beginnen Bezeichner immer mit einem ASCII-Zeichen aus dem Bereich `A` bis `Z` oder einem Unterstrichzeichen (_). Ziffern sind nur nach der ersten Stelle gültig. Bezeichner können mehrere durch Punkte (`.`) getrennte Teile enthalten.

Beispiele für gültige Bezeichner

```
Name
Name1
_Name
Name_1
Alias.Name
```

Bezeichner können mit dem folgenden regulären Ausdruck verglichen werden.

```
[A-Z_][A-Z0-9_.]*
```

Literale für Werte

Literale definieren Werte von bestimmten Datentypen in der Abfrage. In der One Identity Manager Abfragesprache werden folgenden Literale für Werte unterstützt.

Detaillierte Informationen zum Thema

- Zeichenkettenwerte auf Seite 420
- Ganzzahlige Werte auf Seite 421
- Dezimalwerte auf Seite 421
- Datumswerte und Zeitwerte auf Seite 421

Zeichenkettenwerte

In der One Identity Manager Abfragesprache werden Zeichenketten in einfache Anführungszeichen gesetzt. Einfache Anführungszeichen innerhalb von Zeichenketten werden durch Verdoppelung des einfachen Anführungszeichens maskiert. Alle Zeichenfolgeninhalte sind Unicode. Das `N`-Zeichen am Anfang der Zeichenkette ist erlaubt, um eine gewisse Kompatibilität zur SQL-Syntax zu gewährleisten. Es hat keine besondere Bedeutung.

Beispiele für zulässige Zeichenketten

```
''
```

```
N''
```

```
'A string'
```

```
'A ''string'''
```

```
N'A string'
```

Zeichenketten können mit dem folgenden regulären Ausdruck verglichen werden.

```
N?'([^']|'')*'
```

Ein Spezialfall von Zeichenketten sind mehrwertige Eigenschaften. Dies sind Zeichenketten, die durch ASCII-7-Zeichen getrennt sind. Mehrwertige Eigenschaften können in der Abfragesyntax in dieser Form ausgedrückt werden:

```
MVP('Value 1', 'Value 2', 'Value 3')
```

Der resultierende Wert kann an den meisten Stellen verwendet werden, an denen Zeichenketten gültig sind.

Ganzzahlige Werte

In der One Identity Manager Abfragesprache enthalten ganzzahlige Werte (Integer-Werte) Ziffern und ein optionales Pluszeichen (+) oder Minuszeichen (-).

Beispiele für zulässige Werte

42

+42

-42

Ganzzahlige Werte können mit dem folgenden regulären Ausdruck verglichen werden.

```
[+-]?\d+
```

Dezimalwerte

In der One Identity Manager Abfragesprache enthalten Dezimalwerte Ziffern und ein optionales Pluszeichen (+) oder Minuszeichen (-). Die Nachkommaanteil wird durch einen Punkt (.) abgeteilt.

Beispiele für zulässige Werte

3.14

+3.14

-3.14

Dezimalwerte können mit dem folgenden regulären Ausdruck verglichen werden.

```
[+-]?\d*\.\d+
```

Datumswerte und Zeitwerte

In der One Identity Manager Abfragesprache werden Datumswerte und Zeitwerte in der ISO-Notation spezifiziert. Der Angabe des Zeitanteils ist optional. Wenn eine Zeit angegeben wird, sind die Sekundenanteile und die Millisekundenanteile ebenfalls optional.

Standardmäßig werden die Zeitwerte in der UTC-Zeitzone angegeben. Optional kann eine Zeitzone mit dem Schlüsselwort `TIMEZONE` und einer Zeichenkette angegeben werden, die den Bezeichner der Zeitzone enthält. In Datumsbereichen steht der `TIMEZONE`-Bezeichner hinter dem gesamten Bereich und wirkt sich auf beide Daten aus.

Beispiele für zulässige Werte

2020-04-01

2020-4-1

```
2020-04-01 12:30
2020-04-01 12:30:24
2020-04-01 12:30:24.523
2020-04-01 12:30:24 TIMEZONE 'W. Europe Standard Time'
```

Datumswerte können mit dem folgenden regulären Ausdruck verglichen werden.

```
d{4}-\d{1,2}-\d{1,2} \d{1,2}:\d{1,2}(:\d{1,2}(.\d{1,3}))?
```

Referenzen auf Parameter

In der One Identity Manager Abfragesprache werden Abfrageparameter in SQL-Syntax spezifiziert.

```
@ParameterName
```

Der Parametername stimmt mit der Definition von Bezeichnern überein.

Beispiele für gültige Referenzen auf Parameter

```
@Parameter
@Parameter1
@Parameter_1
@Parameter.SecondPart
```

Parameter können mit dem folgenden regulären Ausdruck verglichen werden.

```
@[A-Z_][A-Z0-9_.]*
```

Verwandte Themen

- Bezeichner auf Seite 419

Vorformatierte Where-Klauseln

Da nicht alle möglichen Where-Klauseln in der eingeschränkten One Identity Manager Abfragesprache geschrieben werden können, können vorformatierte Where -Klauseln in der SQL-Syntax des zugrunde liegenden Datenbanksystems verwendet werden.

Vorformatierte Where-Klauseln werden in eckige Klammern (`[` und `]`) eingeschlossen. Öffnende eckige Klammern in der Klausel können mit `[[]` maskiert werden.

Beispiel für eine vorformatierte Where-Klausel

```
[isnull(lastname, '') = N'Harris']
```

Vorformatierte Where-Klauseln können mit dem folgenden regulären Ausdruck verglichen werden.

```
[([^]]|[]])*\]
```

Verwandte Themen

- Verwenden vorformatierter Where-Klauseln auf Seite 436

Formulieren von Abfragen in der One Identity Manager Abfragesprache

Die One Identity Manager Abfragesprache kann zum Erstellen von Abfragen oder Where-Klausel-Ausdrücke gegen die One Identity Manager-Objektschicht verwendet werden.

Die Abfragesprache unterscheidet nicht zwischen Groß- und Kleinschreibung. Schlüsselwörter können in Groß- oder Kleinschreibung geschrieben werden. Um leichter erkennbar zu sein, werden Schlüsselwörter in den folgenden Beispielen immer in Großbuchstaben geschrieben. Die Abfragesyntax in vielen Teilen an die SQL-Syntax angelehnt.

Detaillierte Informationen zum Thema

- Abfrage-Header auf Seite 424
- Where-Klauseln auf Seite 424
- Search-Klauseln auf Seite 425
- Select-Klauseln auf Seite 425
- Order by-Klauseln auf Seite 426
- Klauseln für Paging auf Seite 427
- Klauseln für Anzeigewerte auf Seite 427
- Abfragehinweise auf Seite 429
- Bedingungen auf Seite 430
- Vergleichen von Spalten auf Seite 431
- Vergleichen mittels IN-Klauseln und NOT IN-Klauseln auf Seite 432
- Vergleichen von Datumsdifferenzen auf Seite 433
- Vergleichen von Datumsbereichen auf Seite 434
- Vergleichen von festen Werten auf Seite 435
- Vergleichen von Parametern auf Seite 435
- Verwenden vorformatierter Where-Klauseln auf Seite 436

Abfrage-Header

Datenabfragen in der One Identity Manager Abfragesprache beginnen immer mit dem Schlüsselwort `FROM` und einem Tabellennamen. Ein optionaler Alias für die Tabelle kann nach dem Schlüsselwort `AS` angegeben werden.

```
FROM <table name>
```

```
FROM <table name> AS <table alias>
```

Eine Alternative ist die direkte Abfrage über einen Objektschlüssel.

Beispiel für die Langform der Abfrage

```
FROM Table WHERE PRIMARYKEY '<Key><T>Table</T><P>UID</P></Key>'
```

Beispiel für die Kurzform der Abfrage

```
FROM '<Key><T>Table</T><P>UID</P></Key>'
```

Nach dem Abfrage-Header kann eine Liste von Klauseln angegeben werden. Die Klauseltypen müssen nicht in einer bestimmten Reihenfolge stehen.

Where-Klauseln

Where-Klauseln in der One Identity Manager Abfragesprache beginnen mit dem Schlüsselwort `WHERE` und enthalten eine Bedingung, die resultierenden Daten entsprechen müssen. Anders als in der SQL-Syntax können in der One Identity Manager Abfragesprache mehrere Where-Klauseln in einer Abfrage kombiniert werden.

Beispiele für Where-Klauseln

```
WHERE Lastname = 'Harris'
```

```
WHERE Lastname = 'Harris' AND Firstname = 'Clara'
```

```
WHERE [isnull(lastname, '') = N'Harris']
```

```
WHERE Lastname IN ('Harris', 'Basset')
```

```
WHERE XDateInserted IN RANGE LAST MONTH
```

Verwandte Themen

- Bedingungen auf Seite 430

Search-Klauseln

Für die Suche nach allen passenden Einträgen aus dem Volltextindex wird in der One Identity Manager Abfragesprache das Schlüsselwort `SEARCH` verwendet. Um diesen Klauseltyp zu verwenden muss ein Volltextindex installiert und von der Anwendung erreichbar sein.

```
SEARCH 'Search string'
```

Select-Klauseln

Select-Klauseln der One Identity Manager Abfragesprache beschreiben die Daten, die aus der zugrunde liegenden Datenbank geholt werden. Sie definieren die Mindestmenge der zurückgegebenen Daten. Die Objektschicht kann mehr Daten auswählen, um beliebige Anforderungen zu erfüllen. Ein Beispiel ist der Primärschlüssel, der immer ausgewählt wird, oder spezielle Spalten wie die X-Spalten.

Mehrere Select-Klauseln können in einer Abfrage kombiniert werden.

Es werden verschiedene Select-Varianten unterstützt.

Auswählen bestimmter Spalten einer Tabelle

```
SELECT COLUMNS <list of columns>
```

Beispiel

```
FROM Person
SELECT COLUMNS Firstname, Lastname, CentralAccount
```

Auswählen aller Spalten einer Tabelle

```
SELECT ALL
```

Auswählen aller Anzeigewerte einer Tabelle

```
SELECT DISPLAYS
```

Anzeigespalten, die selektiert werden sind:

- Anzeigemuster der Tabelle (`DialogTable.DisplayPattern`)
- Anzeigemuster (lang) der Tabelle (`DialogTable.DisplayPatternLong`)
- Primärschlüssel der Tabelle

Für die zurückgegebenen Einträge kann das Anzeigemuster und das Anzeigemuster (lang) der Tabelle mit einer Klausel für Anzeigewerte überschrieben werden. Weitere Informationen finden Sie unter Klauseln für Anzeigewerte auf Seite 427.

Auswählen der Anzeigemuster einer Tabelle

`SELECT DISPLAYPATTERN`

Wählt nur die Spalten aus dem Anzeigemuster der Tabelle oder der `DISPLAY`-Klausel, wenn diese verwendet wird. Dies kann eine Leistungsverbesserung gegenüber `SELECT DISPLAYS` sein, da weniger Spalten ausgewählt werden.

Auswählen aller Spalten einer Tabelle, die nicht als BLOB-Feld gekennzeichnet sind

`SELECT NONLOBS`

Selektiert alle Spalten der Tabelle, die nicht als sehr langes Binärobjekt oder Textobjekt durch die Spalte `DialogColumn.IsBlobExternal` gekennzeichnet sind.

Existiert ein passender Eintrag

`EXISTS`

Ermittelt, ob ein Eintrag gemäß den WHERE-Klauseln vorhanden ist. Die Exist-Klausel überschreibt alle anderen Select-Klauseln außer der Count-Klausel.

Ermitteln der Anzahl passender Einträge

`COUNT`

Zählt die Einträge, die mit den Where-Klauseln übereinstimmen. Die Count-Klausel überschreibt alle anderen Klauseln.

Order by-Klauseln

Mit einer Order by-Klausel wird in der One Identity Manager Abfragesprache festgelegt, in welcher Reihenfolge die Einträge zurückgegeben werden.

```
ORDER BY <list of columns>
```

Die Reihenfolge der Spalten kann angegeben werden über

- `ASC` oder `ASCENDING` (Standard)
- `DESC` oder `DESCENDING`

Wenn Anzeigewerte ausgewählt werden, wird eine standardmäßige Order by-Klausel über die Anzeigespalten erzeugt.

Beispiele für Order by-Klauseln

```
ORDER BY Lastname, Firstname
```

```
ORDER BY EntryDate DESC
```

Es können Fallback-Spalten angegeben werden, die für die Sortierung verwendet werden, wenn die vorherigen Werte **NULL** sind. Diese Fallback-Werte werden durch den Null-Koaleszenz Operator `??` angegeben.

Beispiele für die Angabe einer Fallback-Spalte

```
ORDER BY DisplayName ?? CN
```

```
ORDER BY DisplayName ?? CN DESC, XDateInserted
```

Klauseln für Paging

Die Paging-Operatoren der One Identity Manager Abfragesprache ermöglichen es, nur eine Teilmenge der ausgewählten Einträge zurückzugeben.

Take-Klausel

Die Take-Klausel gibt die Anzahl der Einträge an, die maximal zurückgegeben werden sollen. Wenn mehr als eine Take-Klausel angegeben wurde, ist nur die letzte Take-Klausel wirksam.

```
TAKE <integer>
```

Beispiel

```
-- Return only the first ten persons from the result set
```

```
FROM Person SELECT DISPLAYS TAKE 10
```

Skip-Klausel

Die Skip-Klausel gibt an, wie viele Einträge am Anfang des Ergebnisses übersprungen werden sollen, bevor Einträge zurückgegeben werden.

```
SKIP <integer>
```

Beispiel

```
-- Skip 50 persons and return the following 15
```

```
FROM Person SELECT DISPLAYS SKIP 50 TAKE 15
```

Klauseln für Anzeigewerte

Die Klauseln für Anzeigewerte ermöglichen die Definition eines benutzerdefinierten Anzeigemusters. Für die zurückgegebenen Einträge kann das Anzeigemuster und das Anzeigemuster (lang) der Tabelle mit einer Klausel für Anzeigewerte überschrieben werden.

```
DISPLAY 'Display pattern'
LONGDISPLAY 'Display pattern'
```

Parameter ist in beiden Fällen eine Zeichenkette, die das Anzeigemuster mit Platzhaltern in der Form `%Spaltenname%` enthält.

Beispiel

```
FROM Person
SELECT DISPLAYS
DISPLAY '%Lastname%, %Firstname%'
LONGDISPLAY '%Lastname%, %Firstname% - %CentralAccount%'
```

Verwandte Themen

- Select-Klauseln auf Seite 425

Klauseln für Abfrageparameter

Klauseln für Abfrageparameter ermöglichen in der One Identity Manager Abfragesprache die Übergabe von Parameterwerten, die in Where-Klauseln verwendet werden. Es kann entweder ein einzelner Parameter oder eine Liste von Parametern übergeben werden.

Syntax für einen einzelnen Parameter

```
PARAM <Parameter name> [ OF <Type> ] = <Value>
```

Syntax für mehrere Parameter

```
PARAMS
     <Param1> [ OF <Type> ] = <Value>,
     <Param2> [ OF <Type> ] = <Value>
```

Als Parametername sind zulässig:

- Syntax für Parameterreferenzen mit einem qualifizierenden `@`-Zeichen
- Bezeichner

Gültige Typen sind die in `ValType`-Enumeration definierten .Net-Datentypen der Objektschicht. Wenn der Typ aus dem Wert abgeleitet werden kann, ist die Angabe eines Typs optional.

Beispiele für Abfrageparameter

```
PARAM Parameter1 = 'Harris'
PARAM @Parameter2 = 'Harris'
```

```
PARAM Parameter3 OF String = 'Harris'
PARAM @Parameter4 OF String = 'Harris'
PARAM Parameter5 = 42
PARAM Parameter6 OF Int = 42
PARAMS
     Parameter7 OF Double = 3.14,
     Parameter8 OF Date = 2020-04-30
```

Beispiel: Vollständige Abfrage mit Parameterreferenz und Definition

```
FROM Person
WHERE LastName = @Param1
SELECT DISPLAYS
PARAM Param1 = 'Harris'
```

Verwandte Themen

- Bezeichner auf Seite 419
- Referenzen auf Parameter auf Seite 422

Abfragehinweise

Abfragehinweise (Schlüsselwort `HINT`) können in der One Identity Manager Abfragesprache verwendet werden, um den Abfrageprozessor mit zusätzlichen Daten zu versorgen. Abfragehinweise werden intern verwendet, um einen Kontext für zusätzliche Berechtigungsabfragen zum Anzeigen von Spalten bereitzustellen, die nur verfügbar sind, wenn sie als Anzeigewerte für Fremdschlüssel einer anderen Tabelle geladen werden.

```
HINT 'Name' = 'Value'
```

Beispiel: Laden eines Person-Objekts, das von einem Department-Objekt als UID_PersonHead referenziert wird

```
FROM Person
WHERE PRIMARYKEY '<Key><T>Person</T><P>99918ef1-113f-480a-8e6e-
704b1a3cf73a</P></Key>'
SELECT DISPLAYS
```

```
HINT 'SourceContext' = 'Org'
```

Bedingungen

Bedingungen kombinieren alle Ausdrücke, die die resultierenden Entitäten nach einem oder mehreren Kriterien filtern.

In der One Identity Manager Abfragesprache können Bedingungen mit den Operatoren `AND` und `OR` verknüpft werden. Die Vorrangigkeit der Operatoren kann mit geschweiften Klammern (`{ }`) außer Kraft gesetzt werden. Bedingungen können durch das Schlüsselwort `NOT` invertiert werden.

Beispiel für eine Bedingung

```
WHERE Lastname = 'Harris'
AND (Firstname = 'Fred' OR Firstname = 'Clara')
```

Verwandte Themen

- Besondere Bedingungen auf Seite 430

Besondere Bedingungen

Auswählen einer Entität über den Primärschlüssel

Mit dem Schlüsselwort `PRIMARYKEY` wählen Sie in der One Identity Manager Abfragesprache die Entität mit dem passenden Primärschlüssel aus. Der Primärschlüssel muss in Objektschlüssel-Notation angegeben werden.

```
WHERE PRIMARYKEY 'Object Key'
```

Beispiel

```
FROM Person
WHERE PRIMARYKEY '<Key><T>Person</T><P>99918ef1-113f-480a-8e6e-
704b1a3cf73a</P></Key>'
SELECT DISPLAYS
```

Auswählen einer Entität über einen Schlüssel

Mit dem Schlüsselwort `KEY` wählen Sie die Entität mit dem passenden Schlüssel aus. Schlüssel können Objektschlüssel oder alternative Objektschlüssel sein, jeweils in ihrer XML-Notation.

```
WHERE KEY 'Key'
```

Beispiel

```
FROM ADSAccount
WHERE KEY '<Key><Table Name="ADSAccount" Key="c149784b-6386-45d7-a38d-3c6e8e1b69d4"><Prop Name="UID_ADSAccount"><Value>c149784b-6386-45d7-a38d-3c6e8e1b69d4</Value></Prop></Table></Key>'
SELECT COLUMNS cn
```

Auswählen über vordefiniertes SQL

Mit dem Schlüsselwort `LIMITEDSQL` verwenden Sie eine vorformatierte Where-Klausel aus der Tabelle `QBMLimitedSQL`. Eventuell darin verwendete Parameter können mit den Klauseln `PARAMETER` oder `PARAMETERS` angegeben werden.

```
WHERE LIMITEDSQL 'Identifier'
```

Vergleichen von Spalten

Spalten können gegen einen anderen Satz von Zielen verglichen werden. In der One Identity Manager Abfragesprache beginnen alle diese Vergleiche mit dem Spaltennamen.

```
WHERE <Column> <Operator> <Operand>
```

Die folgenden Operatoren werden beim Vergleich von Spalten unterstützt.

Operator	Anwendung	Typ des Operanden
=	Gleich	Wert, der mit dem Typ der Spalte, dem Parameter oder einer anderen Spalte übereinstimmt
<>	Nicht gleich	Wert, der mit dem Typ der Spalte, dem Parameter oder einer anderen Spalte übereinstimmt
<	Kleiner als	Wert, der mit dem Typ der Spalte, dem Parameter oder einer anderen Spalte übereinstimmt
>	Größer als	Wert, der mit dem Typ der Spalte, dem Parameter oder einer anderen Spalte übereinstimmt
<=	Kleiner als oder gleich	Wert, der mit dem Typ der Spalte, dem Parameter oder einer anderen Spalte übereinstimmt

Operator	Anwendung	Typ des Operanden
>=	Größer als oder gleich	Wert, der mit dem Typ der Spalte, dem Parameter oder einer anderen Spalte übereinstimmt
LIKE	Übereinstimmung mit einem angegeben Muster (wie SQL Operator `like`)	Zeichenkette oder mehrwertige Eigenschaft
NOT LIKE	Keine Übereinstimmung mit einem angegeben Muster	Zeichenkette oder mehrwertige Eigenschaft
STARTSWITH	Zeichenkette beginnt mit	Zeichenkette oder mehrwertige Eigenschaft
ENDSWITH	Zeichenkette endet mit	Zeichenkette oder mehrwertige Eigenschaft
CONTAINS	Zeichenkette enthält	Zeichenkette oder mehrwertige Eigenschaft
BITSSET	Die angegebenen Bit-Positionen sind gesetzt	Ganzzahliger Wert oder Parameter
BITSNOTSET	Die angegebenen Bit-Positionen sind nicht gesetzt	Ganzzahliger Wert oder Parameter

Beispiele für Spaltenvergleiche

```
WHERE Lastname = 'Einstein'
WHERE XDateInserted > 2020-02-01
WHERE Lastname STARTSWITH 'Ein'
WHERE XMarkedForDeletion BITSSET 2
```

Vergleichen mittels IN-Klauseln und NOT IN-Klauseln

Vergleiche mit den Schlüsselworten `IN` und `NOT IN` ermöglichen in der One Identity Manager Abfragesprache den Vergleich einer Spalte mit einer Menge von Werten.

Die Werte können angegeben werden

- in Form einer Werteliste
- als mehrwertige Zeichenkette, getrennt durch ein ASCII-7-Zeichen
- als Parameter, der eine durch ein ASCII-7-Zeichen getrennte mehrwertige Zeichenkette enthält

Wertelisten

```
WHERE <Column> IN ( <Value>, <Value>, ...)
WHERE <Column> NOT IN ( <Value>, <Value>, ...)
```

Alle Werte müssen vom gleichen Typ sein und müssen in den Typ der Spalte konvertierbar sein.

Beispiele für Wertlisten

```
WHERE StringColumn IN ('Value 1', 'Value 2', 'Value 3')
WHERE StringColumn NOT IN ('Value 1', 'Value 2', 'Value 3')
```

Mehrwertige Zeichenkette

```
WHERE <Column> IN 'Separated string value'
WHERE <Column> NOT IN 'Separated string value'
```

Beispiel für mehrwertige Zeichenketten

```
WHERE Lastname IN 'Harris Basset Bloggs'
```

Parameter

Mehrwertigen Zeichenketten können über Parameter an Abfragen übergeben werden.

```
WHERE <Column> IN @Parameter
```

Beispiel für Parameter

```
WHERE Lastname IN @Lastnames
```

Die MVP-Syntax kann ebenfalls verwendet werden. Die normale IN-Klausel-Syntax sollte in diesem Fall bevorzugt werden. Weitere Informationen finden Sie unter Zeichenkettenwerte auf Seite 420.

Vergleichen von Datumsdifferenzen

Ein Spezialfall von Spaltenvergleichen in der One Identity Manager Abfragesprache sind die Vergleiche von Datumsdifferenzen. Diese vergleichen den Wert einer Spalte mit einem Zeitbereich, der auf der aktuellen Uhrzeit basiert.

```
WHERE <Column> <Operator> DATE <Integer> <Unit> AGO
```

Zeiteinheit	Schlüsselwort
Jahre	YEARS YEAR Y
Monate	MONTHS MONTH M
Wochen	WEEKS WEEK
Tage	DAYS DAY D
Stunden	HOURS HOUR H
Minuten	MINUTES MINUTE MIN
Sekunden	SECONDS SECOND S
Millisekunden	MILLISECONDS MILLISECOND MS

Beispiele

```
WHERE XDateInserted < DATE 3 MONTHS AGO
WHERE XDateInserted > DATE 5 MIN AGO
```

Vergleichen von Datumsbereichen

Vergleiche von Datumsbereichen in der One Identity Manager Abfragesprache prüfen, ob ein Datum in einen bestimmten Datumsbereich fällt.

Syntax für benannte Bereiche

```
WHERE <Column> IN RANGE <Range name> [ TIMEZONE 'Timezone ID'] [CULTURE
'Culture ID']
```

Syntax für einen bestimmten Zeitbereich

```
WHERE <Column> IN RANGE <Start time> TO <End time> [TIMEZONE <Timezone ID>]
```

Die Startzeit von Bereichen ist eingeschlossen, die Endzeit ist ausgeschlossen.

Zulässige Bezeichnungen sind:

```
TODAY
YESTERDAY
THIS WEEK
THIS MONTH
THIS YEAR
LAST WEEK
LAST MONTH
LAST YEAR
```

```
LAST <integer> DAYS
LAST <integer> DAY
```

Die Namen der Zeitzonen müssen mit den Bezeichnern auf dem System übereinstimmen, das die Abfrage ausführt. Wenn keine Zeitzone angegeben wird, wird die UTC-Zone verwendet.

Die Kultur gibt den Wochenanfangstag an. Sie ist nur sinnvoll, wenn einer der Wochenbereiche angegeben wird.

Beispiele

```
WHERE XDateInserted IN RANGE YESTERDAY
WHERE XDateInserted IN RANGE YESTERDAY
     TIMEZONE 'W. Europe Standard Time'
WHERE XDateInserted IN RANGE LAST YEAR
WHERE XDateInserted IN RANGE THIS WEEK
     TIMEZONE 'W. Europe Standard Time'
     CULTURE 'de-DE'
WHERE XDateInserted IN RANGE 2020-01-01 TO 2020-02-01
     TIMEZONE 'W. Europe Standard Time'
```

Verwandte Themen

- Datumswerte und Zeitwerte auf Seite 421

Vergleichen von festen Werten

In der One Identity Manager Abfragesprache unterstützen Vergleiche zwischen festen Werten nur die Operatoren gleich (=) und nicht gleich (<>).

Beispiel

```
FROM Person
WHERE 1 = 0
SELECT DISPLAYS
```

Vergleichen von Parametern

In der One Identity Manager Abfragesprache können Parameter nur gegen feste Werte verglichen werden. Alle Vergleiche, die für Spalten gültig sind, können auch für Parameter verwendet werden.

HINWEIS: IN-Klauseln und NOT-IN-Klauseln können nicht verwendet werden.

Beispiele

```
WHERE @StringParameter = ''
WHERE @IntParameter > 5
WHERE @IntParameter BITSSET 4
```

Verwandte Themen

- Vergleichen von Spalten auf Seite 431

Verwenden vorformatierter Where-Klauseln

Komplexere Where-Klauseln können in der One Identity Manager Abfragesprache im Format des zugrunde liegenden Datenbanksystems angegeben werden. Sie werden bei der Ausführung auf SQL-Injection-Versuche geprüft. Vorformatierte Where-Klauseln werden in eckige Klammern ([und]) eingeschlossen.

```
WHERE [pre-formatted clause]
```

Beispiel

```
FROM Person
WHERE [isnull(LastName, N'') = N'Einstein']
SELECT DISPLAYS
```

Verwandte Themen

- Vorformatierte Where-Klauseln auf Seite 422

Berichte im One Identity Manager

Der One Identity Manager bietet die Möglichkeit Berichte über mehrere Objekte, einschließlich Summen und anderer Aggregatfunktionen zu erstellen und auszuführen. Gruppenbildungen und die graphische Darstellung von Werten ist ebenfalls möglich. Bei der Schemainstallation werden vordefinierte Berichte mitgeliefert. Kundenspezifische Berichte erstellen und bearbeiten Sie mit dem Report Editor.

Berichte können Sie über zeitlich geplante Berichtsabonnements an definierte E-Mail Adressen versenden. Sie können Berichte über den aktuellen Zustand oder über einen definierten Zeitraum erstellen. Zu jedem Bericht können Sie verschiedene abonnierbare Berichte erstellen, die von den Benutzern des Web Portal bestellt werden. Zusätzlich können Sie den Bericht in die Benutzeroberfläche des Managers oder des Designers einbinden.

Ausführliche Informationen zu Berichtsabonnements finden Sie im *One Identity Manager Administrationshandbuch für Berichtsabonnements* und im *One Identity Manager Web Designer Web Portal Anwenderhandbuch*>.

Detaillierte Informationen zum Thema

- Berichte im Report Editor erstellen und bearbeiten auf Seite 443
- Beispiel für einen einfachen Bericht mit Gruppierung der Daten auf Seite 468
- Übersetzen von Berichten auf Seite 471
- Berichte in die Benutzeroberfläche einbinden auf Seite 472
- Berichte zyklische erzeugen und exportieren auf Seite 474

Arbeiten mit dem Report Editor

Der Report Editor ist das Programm zur Erstellung und Bearbeitung von Berichten. Das Programm nutzt StimulReport.Net-Komponenten für den Entwurf der Berichte. Die genaue Beschreibung und Funktionalität zu den einzelnen Komponenten entnehmen Sie der Online-Hilfe der Firma Stimulsoft (www.stimulsoft.com).

HINWEIS: Beim ersten Start des Report Editors können Sie den Konfigurationstyp (**Basis-Bericht**, **Standard** oder **Professionell**) für die Berichte wählen. Der Konfigurationstyp entscheidet über den Umfang der dargestellten Eigenschaften bei der Bearbeitung eines Berichtes. Den Konfigurationstyp können Sie nachträglich in der Bearbeitungsansicht über das Kontextmenü des Eigenschaftenfensters ändern.

HINWEIS: Berichte mit historischen Datenänderungen werten Datenänderungen aus, die sich in einer One Identity Manager History Database befinden. Ist die One Identity Manager History Database über eine Kennung im TimeTrace der One Identity Manager-Datenbank eingebunden, muss die Anmeldung am Report Editor über einen Anwendungsserver erfolgen, für den diese Kennung in der Konfigurationsdatei (`web.config`) eingetragen ist. Ausführliche Informationen zur Verbindung zur One Identity Manager History Database über Anwendungsserver und die erforderliche Konfiguration finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Menüeinträge im Report Editor

Tabelle 143: Bedeutung der Einträge in der Menüleiste

Menü	Menüeintrag	Bedeutung
Datenbank	Neue Verbindung	Es wird eine Datenbankverbindung hergestellt.
	Einstellungen	Es können allgemeine Programmeinstellungen konfiguriert werden.
	Beenden	Das Programm wird beendet.
Bericht	Erstellen	Es wird ein neuer Bericht erstellt.
	Speichern	Der aktuelle Bericht wird in der Datenbank gespeichert.
	Löschen	Der aktuelle Bericht wird gelöscht.
	Bearbeiten	Der Eigenschaftendialog des aktuellen Berichtes wird geöffnet.
	Daten neu laden	Die Daten des Berichtes werden neu aus der Datenbank geladen.
	Neue virtuelle Datenquelle	Der Dialog zur Erstellung einer virtuellen Datenquelle wird geöffnet.
Hilfe	Community	Die Webseite der One Identity Manager Community wird geöffnet.
	Support Portal	Die Webseite des One Identity Manager Produkt Support wird geöffnet.
	Training	Die Webseite zum One Identity Manager Trainingsportal wird geöffnet.

Menü	Menüeintrag	Bedeutung
	Online-Dokumentation	Die Webseite der One Identity Manager Dokumentation wird geöffnet.
	Suchen	Die Hilfe wird zur Suche geöffnet.
	Hilfe zum Report Editor	Die Hilfe zum Programm wird geöffnet.
	Info	Die Versionsinformationen zum Programm werden angezeigt.

Tabelle 144: Bedeutung der Einträge in der allgemeinen Symbolleiste

Symbol	Bedeutung
	Es wird ein neuer Bericht erstellt.
	Der aktuelle Bericht wird gelöscht.
	Der aktuelle Bericht wird in der Datenbank gespeichert.
	Der Dialog zum Bearbeiten eines Änderungskennzeichens wird geöffnet.
	Das aktuelle Änderungskennzeichen wird als Standard definiert und automatisch verwendet.
	Der Eigenschaftendialog des aktuellen Berichtes wird geöffnet.
	Die Daten des aktuellen Berichtes werden erneut geladen.
	Der Dialog zur Erstellung einer neuen virtuellen Datenquelle wird geöffnet.

Tabelle 145: Funktionen der Symbolleiste in der Berichtliste

Symbol	Bedeutung
	Es werden alle Berichte angezeigt.
	Mit einer Filterbedingung kann die Menge der angezeigten Berichte eingeschränkt werden.
	Der Filter wird ausgeführt und es werden alle Berichte angezeigt, die der Filterbedingung genügen. Die Filterbedingung wird intern als ein `Like`-Vergleich interpretiert.
	Die Berichtliste wird aktualisiert.

Tabelle 146: Einträge im Kontextmenü der Berichtliste

Eintrag im Kontextmenü	Bedeutung
Neu	Es wird ein neuer Bericht erstellt.

Eintrag im Kontext-menü	Bedeutung
Bearbeiten	Der ausgewählte Bericht wird geladen.
Eigenschaften bearbeiten	Der Eigenschaftendialog des ausgewählten Berichtes wird geladen.
Kopieren	Der ausgewählte Bericht wird kopiert.
Löschen	Der ausgewählte Bericht wird gelöscht.

Ansichten im Report Editor

Der Report Editor verfügt über verschiedene Ansichten zur Bearbeitung der Berichte.

Tabelle 147: Ansichten des Report Editor

Ansicht	Beschreibung
Berichtliste	Es werden alle Berichte zusammengefasst nach Kategorien aufgelistet. Mit einer Filterbedingung kann die Menge der angezeigten Berichte eingeschränkt werden.
Bearbeitungsansicht für die Berichte	Im Bearbeitungsbereich werden die Berichte mit dem Report-designer entworfen. Über die Werkzeugpalette des Report-designers können Sie die gewünschten Steuerelemente auf dem Berichtsformular platzieren. HINWEIS: Für den Entwurf der Berichte sollten Sie die Online-Hilfe zu StimulReport.Net der Firma Stimulsoft (www.stimulsoft.com) als Grundlage verwenden.
Eigenschaftendialog	In der Ansicht bearbeiten Sie die Eigenschaften des ausgewählten Berichtes. Für Eingabefelder ist ein Standard-kontextmenü verfügbar.
SQL Protokoll	In der Ansicht werden Datenbankabfragen aufgezeichnet. Nutzen Sie die Aufzeichnung für die Fehlersuche und die Optimierung der Berichte während der Entwicklung. Weitere Informationen finden Sie unter Aufzeichnen der Datenbankabfragen auf Seite 441.

Programmeinstellungen des Report Editors

Allgemeine Konfigurationseinstellungen werden in einer Konfigurationsdatei `ReportEdit2.exe.config` vorgegeben. Zusätzlich können global gültige Konfigurationseinstellungen über eine globale Konfigurationsdatei `Global.cfg` im One Identity Manager-eigenen Format definiert werden. Die Konfigurationsdateien sind im

Programmverzeichnis abgelegt. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Prozessüberwachung und Fehlersuche*.

Um im Report Editor die Programmeinstellungen zu ändern

1. Wählen Sie im Report Editor den Menüeintrag **Datenbank > Einstellungen**.
2. Im Bereich **Spracheinstellungen** passen Sie folgende Einstellungen an:
   - **Allgemeine Kultur**: Sprache für die Formatierung von Daten wie beispielsweise Datumsformate, Zeitformate oder Zahlenformate.
   - **Andere Sprache der Benutzeroberfläche**: Sprache für die Benutzeroberfläche. Bei der ersten Anmeldung am Programm wird die Systemsprache zur Anzeige der Benutzeroberfläche verwendet. Die Änderung der Sprache wird mit dem Neustart des Programms wirksam. Die Sprache wird global für alle Programme des One Identity Manager festgelegt, somit muss die Spracheinstellung nicht in jedem Programm erneut vorgenommen werden.
3. Im Bereich **Verhalten** passen Sie folgende Einstellungen an:
   - **Code-Tabreiter anzeigen**: Gibt an, ob der Tabreiter zur Bearbeitung des Skriptcodes im Reportdesigner eingeblendet wird.
   - **Beim Speichern ohne Änderungskennzeichen nachfragen**: Änderungen an Berichten sollten Sie auf ein Änderungskennzeichen buchen. Setzen Sie die Option, damit beim Speichern von Änderungen ohne Änderungskennzeichen eine Hinweismeldung angezeigt wird.
   - **Maximale Anzahl an Vorschausätzen**:Anzahl der Datensätze, die maximal bei der Vorschauanzeige des Berichtes verwendet werden.
4. Übernehmen Sie die Einstellungen mit **OK**.

Aufzeichnen der Datenbankabfragen

Nutzen Sie im Report Editor die Aufzeichnung der Datenbankabfragen für die Fehlersuche und die Optimierung der Berichte während der Entwicklung. Aufgezeichnet werden die Ausführungszeit und das ausgeführte Kommando.

Um die Aufzeichnungen zu starten und zu stoppen

1. Öffnen Sie das Aufzeichnungsfenster im Report Editor über den Eintrag **SQL Protokoll** im unteren Bereich des Programms.
2. (Optional) Legen Sie über das Kontextmenü **Spalten wählen** fest, welche Spalten im Protokoll angezeigt werden.
3. Um die Aufzeichnung zu starten, wählen Sie das Kontextmenü **Start/Stopp**.
4. Um die Aufzeichnung zu stoppen, wählen Sie erneut das Kontextmenü **Start/Stopp**.

TIPP: Sie können die Darstellung der Meldungen im Protokoll konfigurieren. Schalten Sie dazu die Anzeige des Protokolls über die Schaltfläche am rechten Rand der Tabelle in den erweiterten Modus.

Tabelle 148: Bedeutung der Symbole im Protokoll

Symbol	Bedeutung
!	Es werden alle kritischen Fehlermeldungen aufgezeichnet. (Informationsgrad **Fatal**)
	Es werden alle Informationen aufgezeichnet. (Informationsgrad **Info**)
	Es werden alle Warnungen aufgezeichnet. (Informationsgrad **Warning**)
	Es werden alle Fehlermeldungen aufgezeichnet. (Informationsgrad **Error**)
	Debugger-Ausgaben werden aufgezeichnet. Diese Einstellung sollte nur zu Testzwecken verwendet werden. (Informationsgrad **Debug**)
	Es erfolgt die Ausgabe sehr ausführlicher Informationen. Diese Einstellung sollte nur zu Analysezwecken verwendet werden. Das Protokoll wird schnell groß und unübersichtlich. (Informationsgrad **Trace**)
	Es kann eine benutzereigene Filterbedingung definiert und angewendet werden.
	Die Filterbedingung wird gelöscht.
	Es wird nach einem Begriff gesucht.
	Es wird weiter nach dem Begriff gesucht.
	Alle Meldungen mit einem bestimmten Begriff werden markiert.
Puffergröße	Die Puffergröße für die Aufzeichnung der Meldungen kann eingestellt werden. Der Füllstand des Puffers wird neben dem Eingabefeld angezeigt.
	Der Pufferinhalt wird gelöscht.
	Die Aufzeichnung der Meldungen wird gestoppt.
	Die Aufzeichnung der Meldungen wird gestartet.
	Das Protokoll wird in eine Datei gespeichert.
	Es kann festgelegt werden, welche Spalten zur Anzeige im Fehlerprotokoll verwendet werden.
	Die ausgewählten Meldungen werden in die Zwischenablage kopiert.
	Das Fehlerprotokoll wird in einem Texteditor geöffnet.

Berichte im Report Editor erstellen und bearbeiten

HINWEIS: Berichte mit historischen Datenänderungen werten Datenänderungen aus, die sich in einer One Identity Manager History Database befinden. Ist die One Identity Manager History Database über eine Kennung im TimeTrace der One Identity Manager-Datenbank eingebunden, muss die Anmeldung am Report Editor über einen Anwendungsserver erfolgen, für den diese Kennung in der Konfigurationsdatei (`web.config`) eingetragen ist. Ausführliche Informationen zur Verbindung zur One Identity Manager History Database über Anwendungsserver und die erforderliche Konfiguration finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Berichte erstellen und bearbeiten Sie im Programm Report Editor. Berichte werden in der Datenbanktabelle `DialogReport` abgelegt. Die Erstellung eines Berichtes beinhaltet die folgenden Schritte:

1. Definieren der Berichtseigenschaften, Datenquellen und Berichtsparameter
2. Entwerfen des Berichtsformulars mit dem Reportdesigner

Standardmäßig werden mit dem One Identity Manager bereits definierte Berichte mitgeliefert, die während einer Schemainstallation automatisch angepasst werden. Sind kundenspezifischen Anpassungen an den mitgelieferten Berichten erforderlich:

1. Erstellen Sie eine Kopie des Berichtes.
2. Bearbeiten Sie die benötigten Berichtseigenschaften.
3. Verwenden Sie im Weiteren die kundenspezifische Version des Berichtes.

Beim Einfügen oder Kopieren eines Berichtes wird zunächst der Eigenschaftendialog geöffnet, über welchen Sie die allgemeinen Daten zum Bericht, die notwendigen Datenquellen sowie eventuelle Parameter für die Berichtdefinition erfassen. Anschließend wird im Bearbeitungsbereich mit dem Reportdesigner eine neues Berichtsformular erstellt. Dieses bildet die Grundlage für den Entwurf des Berichtes. Über die Werkzeugpalette des Reportdesigners können Sie die gewünschten Steuerelemente auf dem Berichtsformular platzieren.

HINWEIS: Für den Entwurf der Berichte sollten Sie die Online-Hilfe zu StimulReport.Net der Firma Stimulsoft (www.stimulsoft.com) als Grundlage verwenden.

Um die einen neuen Bericht zu erstellen

- Wählen Sie im Report Editor den Menüeintrag **Bericht > Erstellen**.

Um einen Bericht zu kopieren

- Wählen Sie im Report Editor den Bericht in der Berichtliste und wählen Sie den Kontextmenüeintrag **Kopieren.**

  Es wird eine neuer Bericht erstellt und der Eigenschaftendialog zur Bearbeitung angezeigt. Die Eigenschaften des neuen Berichtes werden aus dem Originalbericht übernommen.

Um einen Bericht zu bearbeiten

1. Wählen Sie im Report Editor den Bericht in der Berichtliste und öffnen Sie den Bericht per Maus-Doppelklick oder über den Kontextmenüeintrag **Bearbeiten**.

   Das Berichtsformular wird im Reportdesigner geöffnet.
2. Um den Eigenschaftendialog zu öffnen, wählen Sie anschließend den Menüeintrag **Bericht > Bearbeiten**.

Um die Berichtseigenschaften zu bearbeiten ohne den Bericht im Reportdesigner zu laden

- Wählen Sie im Report Editor den Bericht in der Berichtliste und wählen Sie das Kontextmenü **Eigenschaften bearbeiten**.

  Der Eigenschaftendialog wird geöffnet.

HINWEIS: Nachdem Sie einen Bericht angepasst haben, können Sie ein Änderungskennzeichen festlegen, mit dem der Bericht gekennzeichnet wird. Diese Änderungskennzeichen werden im Database Transporter bei der Erstellung eines Kundentransportpaketes als Exportkriterium angeboten.

Detaillierte Informationen zum Thema

- Allgemeinen Berichteigenschaften bearbeiten auf Seite 444
- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458
- Virtuellen Datenquellen verwenden auf Seite 465
- Bearbeiten des Berichtsformulars auf Seite 465

Allgemeinen Berichteigenschaften bearbeiten

Um die allgemeinen Berichteigenschaften zu bearbeiten

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Eigenschaften**.
3. Bearbeiten Sie die allgemeinen Eigenschaften.
4. Speichern Sie die Änderungen.

Tabelle 149: Allgemeine Eigenschaften eines Berichtes

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung des Berichtes. Kennzeichnen Sie kunden-

Eigenschaft	Bedeutung
	spezifische Berichte mit dem Präfix **CCC_**.
Anzeigename	Anzeigename des Berichtes. Der Anzeigename steht bei der Erstellung eines Berichtes als ReportAlias zu Verfügung und kann beispielsweise zur Bildung des Berichttitels oder des Dateinamens beim Exportieren eines Berichtes im Webfrontend verwendet werden. Übersetzen Sie den eingegebenen Text über die Schaltfläche. Der Anzeigename eines Berichtes kann Variablen enthalten, zulässig sind Systemvariablen sowie die Berichtsparameter. Die Übergabe der Variablen erfolgt mittels Prozentzeichen. Beispiel: `Name des Berichtes %Variable%`
Max. Laufzeit [Sekunden]	Maximale Anzahl Sekunden, die für die Generierung des Berichtes zur Verfügung stehen. Bei Überschreitung der Zeit wird die Generierung des Berichtes abgebrochen.
Beschreibung	Beschreibung des Berichtes. Übersetzen Sie den eingegebenen Text über die Schaltfläche.
Filterkriterium	Filterkriterium für die Anzeige der Berichte im Webfrontends.
Basistabelle	Basistabelle des Berichtes.
Kategorie	Kategorie dient zur Klassifizierung von Berichten. Zulässige Werte sind die Kategorien **Common**, **Mail**, **Attestation** und **Dashboard**.
Präprozessorbedingung	Berichte können Sie mit einer Präprozessorbedingung versehen. Damit ist ein Bericht nur verfügbar, wenn die Präprozessorbedingung erfüllt ist.
Benutzerdefinierte Felder > Freies Feld Nr. 01 ... Freies Feld Nr. 10	Zusätzliche unternehmensspezifische Informationen. Die Anzeigenamen, Formate und Bildungsregeln für die Eingabefelder können Sie mit dem Designer an Ihre Anforderungen anpassen.
Erweiterte Eigenschaften > Bericht	UID, unter welcher der Bericht in der Datenbank abgelegt ist.

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenquellen erstellen und bearbeiten

Für jeden Bericht müssen Sie eine Datenquelle erstellen aus der die darzustellenden Daten des Berichtes ausgelesen werden. In der Regel genügt eine Datenquelle für einen Bericht, Sie können aber mehrere Datenquellen pro Bericht definieren. Während der Bearbeitung einer Datenquelle können Sie das resultierende Ergebnis prüfen.

Um eine Datenquelle zu bearbeiten

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Datenquelle**.
3. Wählen Sie in der Liste **Vorhandene Datenquellen** die Datenquelle.

   - ODER -

   Klicken Sie die Schaltfläche **Hinzufügen**.

   Eine neue Datenquelle wird erstellt.
4. Bearbeiten Sie die Eigenschaften der Datenquelle.
5. Speichern Sie die Änderungen.

Um eine Datenabfrage zu testen

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Datenquelle**.
3. Wählen Sie in der Liste **Vorhandene Datenquellen** die Datenquelle.
4. Klicken Sie die Schaltfläche neben der Auswahlliste **Abfragemodul**.

   Das Ergebnis einer Datenquelle wird in einem separaten Dialogfenster geöffnet.

HINWEIS: Beim Kopieren der Datenabfrage in die Zwischenablage wird eine Datenbankabfrage in SQL Syntax generiert, die Sie mit einem geeigneten Programm zur Ausführung von SQL Abfragen in der Datenbank ausführen können. Um die Datenabfrage zu kopieren, verwenden Sie die Schaltfläche neben der Auswahlliste **Abfragemodul**.

Um eine Datenquelle zu löschen

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Datenquelle**.
3. Wählen Sie in der Liste **Vorhandene Datenquellen** die Datenquelle.
4. Klicken Sie die Schaltfläche **Löschen**.
5. Speichern Sie die Änderungen.

Detaillierte Informationen zum Thema

- Datenabfrage über eine SQL-Abfrage auf Seite 447
- Datenabfrage über eine Datenbanksicht auf Seite 448

- Datenabfrage über ein Objekt auf Seite 449
- Datenabfrage über die Historie eines Einzelobjektes auf Seite 451
- Datenabfrage über die Historie mehrerer Objekte auf Seite 452
- Datenabfrage über historische Zuordnungen auf Seite 454
- Datenabfrage für Simulationsdaten auf Seite 456

Verwandte Themen

- Virtuellen Datenquellen verwenden auf Seite 465

Datenabfrage über eine SQL-Abfrage

Datenabfragen mit dem Abfragemodul **SQL** erfolgen ohne Überprüfung der Zugriffsberechtigungen der Benutzer direkt auf der Datenbank. Das bedeutet, hat ein Benutzer auf eine der verwendeten Spalten keine Berechtigungen, wird die Spalte im Bericht trotzdem angezeigt.

Tabelle 150: Eigenschaften für Datenquelle SQL

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Max. Zeilen	Maximale Anzahl von Ergebniszeilen dieser Abfrage. HINWEIS: Es wird maximal diese Anzahl an Ergebnissen ermittelt und im Bericht angezeigt auch wenn die Anzahl überschritten werden sollte. Im Standard wird keine Fehlermeldung oder Hinweismeldung ausgegeben. Eventuelle Meldungen müssen im Bericht kundenspezifisch implementiert werden.
Übergeordnete Abfrage	Wird nicht verwendet.
Abfragemodul	Wählen Sie das Abfragemodul **SQL**.
Abfrage	Vollständige Datenbankabfrage in SQL-Syntax. Die Abfrage muss alle Spalten enthalten, die im Bericht verwendet werden. In der Abfrage können Sie SQL Parameter verwenden. Diese Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen. Syntax für Parameter: `@<Parametername>` Syntax für Parameter vom Datentyp **Date** und einem Wertebereich (Zeitraum von bis):

Eigenschaft	Bedeutung

	`@<Parametername>Start`
	`@<Parametername>End`
	Beispiel für die Verwendung in der SQL-Abfrage:
	`and StartDate between @<Parametername>Start and @<Parametername>End`

Beispiel:

Die Abfrage soll die Personen (Tablle `Person`) liefern die einer Abteilung zugeordnet sind. Die Abteilung (`UID_Department`) wird über den Objektschlüssel (`XObjektKey`) ermittelt. Dieser wird als Parameter an den Bericht übergeben. Abgefragt werden Vorname (`Firstname`) und Nachname (`Lastname`) der Person sowie der Name der Abteilung (`Departmentname`).

```
Select Firstname, Lastname, Departmentname
     from person join Department
     on person.uid_Department = department.uid_Department
     where Department.XObjectKey = @ObjectKeyBase
```

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenabfrage über eine Datenbanksicht

Mit dem Abfragemodul **Sicht** erstellen Sie die Datenabfrage über vordefinierte Datenbanksichten und steuern so die Zugriffsberechtigungen der Benutzer.

Tabelle 151: Eigenschaften für Datenquelle Sicht

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Max. Zeilen	Maximale Anzahl von Ergebniszeilen dieser Abfrage. Bei Überschreitung der Anzahl wird die Berichtserstellung abgebrochen.
Übergeordnete	Wird nicht verwendet.

Eigenschaft	Bedeutung
Abfrage	
Abfragemodul	Wählen Sie das Abfragemodul **Sicht**.
Name der Sicht	Bezeichnung der Datenbanksicht.
Bedingung	Bedingung zum Einschränken der Datensätze, die über die Datenbanksicht geliefert werden. Die Bedingung formulieren Sie als gültige Where-Klausel für Datenbankabfragen. In der Bedingung können Sie SQL Parameter verwenden. Diese Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen. Syntax für Parameter: `@<Parametername>`
Sortierung	Spalten der Datenbanksicht, nach denen die Datenabfrage sortiert wird.

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenabfrage über ein Objekt

Datenabfragen mit dem Abfragemodul **Objekt** erfolgen über die Objektschicht und somit unter vollständiger Berücksichtigung der Zugriffsberechtigungen der Benutzer.

Tabelle 152: Eigenschaften für Datenquelle Objekt

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Max. Zeilen	Maximale Anzahl von Ergebniszeilen dieser Abfrage. HINWEIS: Es wird maximal diese Anzahl an Ergebnissen ermittelt und im Bericht angezeigt auch wenn die Anzahl überschritten werden sollte. Im Standard wird keine Fehlermeldung oder Hinweismeldung ausgegeben. Eventuelle Meldungen müssen im Bericht kundenspezifisch implementiert werden.
Übergeordnete Abfrage	In einer übergeordneten Abfrage werden Einschränkungen der Datensätze getroffen, die an die nachfolgenden Abfragen weitergegeben werden, zum Beispiel alle Personen einer Abteilung. Parameter, die in der übergeordneten Abfrage definiert werden, sind auch in den

Eigenschaft	Bedeutung
	nachfolgenden Abfragen verfügbar.
Abfragemodul	Wählen Sie das Abfragemodul **Objekt**.
Tabelle	Tabelle, aus der die Objekte ermittelt werden.
Spalten	Spalten der Tabelle, die im Bericht verwendet werden. Einige Spalten werden immer in die Berichtdefinition übernommen und müssen hier nicht explizit angegeben werden. Dazu gehören: • die Primärschlüsselspalten der Tabelle • alle Spalten, die im Anzeigemuster der Tabelle verwendet werden • Hilfsspalten (`_Display` und `_DisplayLong`), die das Anzeigemuster der Tabelle liefern • Für Fremdschlüsselspalten und Spalten mit einer Liste definierter Werte oder mehrsprachiger Eingabe, wird eine zusätzliche Spalte für den Anzeigewert erstellt (`<Spalte>_Display`).
Fremdschlüssel auflösen	Setzen Sie diese Option, wenn in der `<Spalte>_Display` statt der `UID` der Anzeigewert des referenzierten Objektes zurückgegeben werden soll.

Bedingung	Bedingung zum Einschränken der Datensätze, die über die Tabelle geliefert werden. Die Bedingung formulieren Sie als gültige Where-Klausel für Datenbankabfragen. In der Bedingung können Sie SQL Parameter verwenden. Diese Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen.
	Syntax für Parameter:
	`@<Parametername>`
	Syntax für Spalten einer übergeordneten Abfrage:
	`@<Bezeichnung der übergeordneten Abfrage>.<Spalte der übergeordneten Abfrage>`

Sortierung	Spalten der Tabelle, nach denen die Datenabfrage sortiert wird.

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenabfrage über die Historie eines Einzelobjektes

Datenabfragen mit dem Abfragemodul **Historie eines Einzelobjektes** verwenden Sie um Berichte mit historische Verläufen über ein Einzelobjekt, beispielsweise eine einzelne Person, zu erstellen.

Tabelle 153: Eigenschaften für Datenquelle Historie eines Einzelobjektes

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Max. Zeilen	Maximale Anzahl von Ergebniszeilen dieser Abfrage. HINWEIS: Es wird maximal diese Anzahl an Ergebnissen ermittelt und im Bericht angezeigt auch wenn die Anzahl überschritten werden sollte. Im Standard wird keine Fehlermeldung oder Hinweismeldung ausgegeben. Eventuelle Meldungen müssen im Bericht kundenspezifisch implementiert werden.
Übergeordnete Abfrage	In einer übergeordneten Abfrage werden Einschränkungen der Datensätze getroffen, die an die nachfolgenden Abfragen weitergegeben werden, zum Beispiel alle Personen einer Abteilung. Parameter, die in der übergeordneten Abfrage definiert werden, sind auch in den nachfolgenden Abfragen verfügbar.
Abfragemodul	Wählen Sie das Abfragemodul **Historie eines Einzelobjektes**.
Objektschlüssel	Der Objektschlüssel kann direkt oder per Parameter abgefragt werden. Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen. Spalten einer übergeordneten Abfrage werden nach folgender Syntax angegeben: `<Name der übergeordneten Abfrage>.<Spalte der übergeordneten Abfrage>`
Min. Datum oder Zeitraum	Mit dem minimalen Datum legen Sie den Zeitpunkt fest, ab dem die historischen Daten ermittelt werden. Sie können das Datum direkt oder über einen den Parameter festlegen. Per Parameter wird das minimale Datum aller betroffenen Einträge in den verbundenen One Identity Manager History Database-Datenbanken ermittelt. Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen.
Spalten	Spalten, für welche die Änderungen ermittelt werden.
Fremdschlüssel auflösen	Setzen Sie diese Option, wenn statt der `UID` der Anzeigewert des referenzierten Objektes zurückgegeben werden soll.

Die Datenabfrage liefert folgende Spalten.

Tabelle 154: Spalten der Datenabfrage für die Historie eines Einzelobjektes

Spalte	Bedeutung
ChangeID	Eindeutige Kennung (`UID`) der Aufzeichnung.
ObjectKey	Objektschlüssel der Aufzeichnung.
ObjectUID	Eindeutige Kennung (`UID`) des geänderten Objektes.
User	Name des Benutzers, der die Änderung veranlasst hat.
ChangeTime	Zeitpunkt der Änderung.
ChangeType	Änderungstyp (`Insert`, `Update`, `Delete`).
Columnname	Name der Spalte, deren Wert sich geändert hat.
ColumnDisplay	Anzeigename der Spalte, deren Wert sich geändert hat.
OldValue	Alter Wert der Spalte.
OldValueDisplay	Anzeigewert des alten Wertes.
NewValue	Neuer Wert der Spalte.
NewValueDisplay	Anzeigewert des neuen Wertes.

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenabfrage über die Historie mehrerer Objekte

Datenabfragen mit dem Abfragemodul **Historie mehrerer Objekte** verwenden Sie um Berichte mit historische Verläufen über mehrere Objekte zu erstellen, die über ein zusätzliches Kriterium weiter eingeschränkt werden, beispielsweise alle Personen mit einem bestimmten Nachnamen.

Tabelle 155: Eigenschaften für Datenquelle Historie mehrerer Objekte

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Max. Zeilen	Maximale Anzahl von Ergebniszeilen dieser Abfrage. HINWEIS: Es wird maximal diese Anzahl an Ergebnissen ermittelt

Eigenschaft	Bedeutung
	und im Bericht angezeigt auch wenn die Anzahl überschritten werden sollte. Im Standard wird keine Fehlermeldung oder Hinweismeldung ausgegeben. Eventuelle Meldungen müssen im Bericht kundenspezifisch implementiert werden.
Übergeordnete Abfrage	Wird nicht verwendet.
Abfragemodul	Wählen Sie das Abfragemodul **Historie mehrerer Objekte**.
Tabelle	Tabelle, aus der die Objekte ermittelt werden.
Min. Datum oder Zeitraum	Mit dem minimalen Datum legen Sie den Zeitpunkt fest, ab dem die historischen Daten ermittelt werden. Sie können das Datum direkt oder über einen den Parameter festlegen. Per Parameter wird das minimale Datum aller betroffenen Einträge in den verbundenen One Identity Manager History Database-Datenbanken ermittelt. Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen.
Spalten	Spalten, für welche die Änderungen ermittelt werden.
Kriterien	Spalte, der Tabelle und Wert, über die die ermittelten Objekte weiter eingeschränkt werden. Der Wert kann direkt oder als Parameter abgefragt werden. Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen.

Die Datenabfrage liefert folgende Spalten.

Tabelle 156: Spalten der Datenabfrage für die Historie eines Einzelobjektes

Spalte	Bedeutung
ChangeID	Eindeutige Kennung (`UID`) der Aufzeichnung.
ObjectKey	Objektschlüssel der Aufzeichnung.
ObjectUID	Eindeutige Kennung (`UID`) des geänderten Objektes.
User	Name des Benutzers, der die Änderung veranlasst hat.
ChangeTime	Zeitpunkt der Änderung.
ChangeType	Änderungstyp (`Insert`, `Update`, `Delete`).
Columnname	Name der Spalte, deren Wert sich geändert hat.
ColumnDisplay	Anzeigename der Spalte, deren Wert sich geändert hat.
OldValue	Alter Wert der Spalte.
OldValueDisplay	Anzeigewert des alten Wertes.

Spalte	Bedeutung
NewValue	Neuer Wert der Spalte.
NewValueDisplay	Anzeigewert des neuen Wertes.

Beispiel:

Es soll die Historie aller Personen mit dem Nachnamen "Harris" erstellt werden. Die Berichtdaten können folgendermaßen definiert werden:

Tabelle:	Person
Min. Datum:	MinDate
Kriterien: Spalte	Lastname
Kriterien: Wert	Harris

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenabfrage über historische Zuordnungen

Datenabfragen mit dem Abfragemodul **Historische Zuordnungen** verwenden Sie um Berichte mit historische Verläufen von Zuweisungen zu Objekten, beispielsweise alle Rollenmitgliedschaften einer Person, zu erstellen. Das Abfragemodul wird sowohl für Abfragen über Fremdschlüssel-Beziehungen als auch über Zuweisungstabellen (M:N-Tabellen) und Child-Beziehungen eingesetzt.

Tabelle 157: Eigenschaften für Datenquelle Historische Zuordnungen

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Max. Zeilen	Maximale Anzahl von Ergebniszeilen dieser Abfrage. HINWEIS: Es wird maximal diese Anzahl an Ergebnissen ermittelt und im Bericht angezeigt auch wenn die Anzahl überschritten werden sollte. Im Standard wird keine Fehlermeldung oder Hinweismeldung ausgegeben. Eventuelle

Eigenschaft	Bedeutung
	Meldungen müssen im Bericht kundenspezifisch implementiert werden.
Übergeordnete Abfrage	In einer übergeordneten Abfrage werden Einschränkungen der Datensätze getroffen, die an die nachfolgenden Abfragen weitergegeben werden, zum Beispiel alle Personen einer Abteilung. Parameter, die in der übergeordneten Abfrage definiert werden, sind auch in den nachfolgenden Abfragen verfügbar.
Abfragemodul	Wählen Sie das Abfragemodul **Historische Zuordnungen**.
Zuordnungsrichtung	Zuordnungen, die im Bericht verwendet werden sollen. Zulässig sind die Werte **Zuweisungen (CR & MN)** und **Referenzierte Objekte (FK)**.
Tabelle	Tabelle, über welche die Zuweisung erfolgt ist.
Min. Datum oder Zeitraum	Mit dem minimalen Datum legen Sie den Zeitpunkt fest, ab dem die historischen Daten ermittelt werden. Sie können das Datum direkt oder über einen den Parameter festlegen. Per Parameter wird das minimale Datum aller betroffenen Einträge in den verbundenen One Identity Manager History Database-Datenbanken ermittelt. Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen.
Auswahlspalte	Spalte der Tabelle, über die auf das Basisobjekt verwiesen wird.
Auswahlwert	Wert der Auswahlspalte kann direkt oder per Parameter abgefragt werden. Parameter müssen Sie anschließend auf dem Tabreiter **Parameter** zum Bericht hinzufügen. Spalten einer übergeordneten Abfrage werden nach folgender Syntax angegeben: `<Name der übergeordneten Abfrage>.<Spalte der übergeordneten Abfrage>`
Abzufragender Fremdschlüssel	Fremdschlüsselspalte, um historische Zuordnungen zu erhalten.
Deaktivierungsspalte	Einige Tabellen enthalten Spalten mit denen ein Objekt deaktiviert werden kann, beispielsweise die Spalte `AccountDisabled` in der Tabelle `ADSAccount`. Geben Sie diese Spalten an, wenn im Bericht die Deaktivierung einer Zuweisung als Löschen und die Aktivierung als Einfügen aufgeführt werden soll.
Zusätzliche Objektspalten	Spalten der Tabelle, die zusätzlich im Bericht verfügbar sein sollen.
Zusätzliche Kriterien	Spalte, der Tabelle und Wert, zur weiteren Einschränkung des Basisobjektes.

Die Datenabfrage liefert folgende Spalten. Zusätzlich werden die Spalten geliefert, die als zusätzliche Objektspalten erfasst sind.

Tabelle 158: Spalten der Datenabfrage über historische Zuordnungen

Spalte	Bedeutung
BaseKey	Objektschlüssel des Basisobjektes der Zuweisung.
BaseUID	Eindeutige Kennung des Basisobjektes.
ObjectKey	Objektschlüssel der Zuweisung.
DestinationKey	Objektschlüssel des Zielobjektes der Zuweisung.
DestinationUID	Eindeutige Kennung des Zielobjektes.
Display	Anzeigewert des Zielobjektes.
CreationUser	Nutzer der die Zuweisung erstellt hat.
CreationTime	Zeitpunkt der Zuweisung.
DeletionUser	Nutzer, der die Zuweisung gelöscht hat.
DeletionTime	Zeitpunkt des Löschens.
Type	Nähere Spezifizierung der Zuweisung, beispielsweise Name der Zuweisungstabelle oder Zielsystemtyp.
Origin	Bitmaske zur Abbildung der Art der Zuweisung.
OriginDisplay	Anzeigename der Bitmaske zur Abbildung der Art der Zuweisung.

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446
- Berichtsparameter auf Seite 458

Datenabfrage für Simulationsdaten

Um die Simulationsdaten, die während der Simulation im Manager erzeugt werden, in einem Bericht auszuwählen, verwenden Sie folgende Abfragemodule:

- **Ergebnis Frontendsimulation**: Das Abfragemodul können Sie für alle Teilbereiche einer Simulation, außer Auswertung der Regelverletzungen, einsetzen.
- **Ergebnis Frontendsimulation Compliance**: Dieses Abfragemodul verwenden Sie, um die Auswertung der Regelverletzungen im Bericht auszugeben.

Tabelle 159: Eigenschaften für Datenquelle Ergebnis Frontendsimulation

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.

Eigenschaft	Bedeutung
Beschreibung	Beschreibung der Datenquelle.
Abfragemodul	Wählen Sie das Abfragemodul **Ergebnis Frontendsimulation**.
Übergeordnete Abfrage	Wird nicht verwendet.
Simulationsauswertung	Angabe, welcher Teilbereich der Simulationsauswertung im Bericht angezeigt wird. Zulässige Werte sind: • **Überblick**: Zeigt im Überblick, welche Aktionen durch die vorgenommenen Änderungen während der Simulation ausgelöst wurden. • **Geänderte Objekte**: Zeigt alle Objekte und ihre Eigenschaften, die von den Änderungen während der Simulation betroffen waren. • **DBQueue**: Zeigt die Berechnungsaufträge für den DBQueue Prozessor, die aus den Änderungen während der Simulation resultieren. • **Triggeränderungen**: Zeigt alle Änderungen an Objekten, die während der Simulation durch Trigger eingestellt wurden. • **Generierte Prozesse**: Zeigt die Prozesse und Prozessschritte, die aufgrund der Änderungen während der Simulation generiert wurden.

Tabelle 160: Eigenschaften für Datenquelle Ergebnis Frontendsimulation Compliance

Eigenschaft	Bedeutung
Bezeichnung	Bezeichnung der Datenquelle.
Beschreibung	Beschreibung der Datenquelle.
Abfragemodul	Wählen Sie das Abfragemodul **Ergebnis Frontendsimulation Compliance**.
Übergeordnete Abfrage	Wird nicht verwendet.

Verwandte Themen

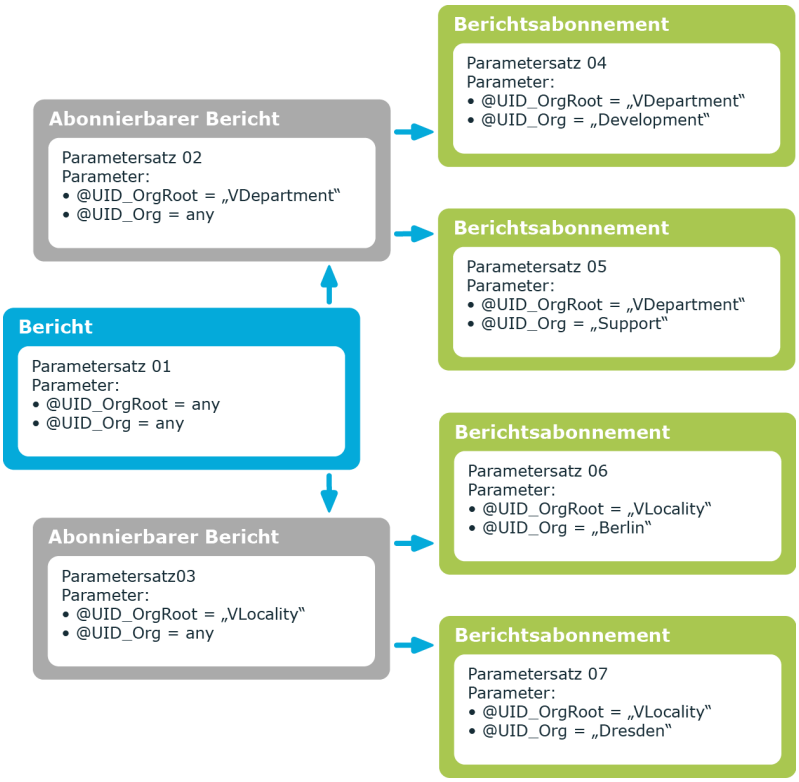
- Datenquellen erstellen und bearbeiten auf Seite 446

Berichtsparameter

Ein Bericht kann verschiedene Berichtsparameter enthalten, die bei der Erzeugung des Berichtes oder bei der Generierung der E-Mail Benachrichtigung ermittelt und dem Bericht übergeben werden. Der berechnete Bericht wird dann angezeigt oder entsprechend der eingerichteten Berichtsabonnements per E-Mail an den Abonnenten versendet. Berichtsparameter können vor der Anzeige des Berichtes vom Benutzer abgefragt werden. So kann beispielsweise der Bericht auf ein Zeitraum eingegrenzt oder eine bestimmte Abteilung übergeben werden, für die ein Bericht angezeigt wird.

Berichtsparameter werden intern in Parametersätzen zusammengefasst. Für jeden Bericht, jeden abonnierbaren Bericht und jedes Berichtsabonnement wird automatisch ein separater Parametersatz erzeugt. Über die Abfolge *Bericht->Abonnierbarer Bericht->Berichtsabonnement* werden die Berichtsparameter und deren Einstellungen vererbt.

Abbildung 33: Vererbung von Berichtsparametern



Berichtsparameter konfigurieren Sie an verschiedenen Stellen.

Berichtsparameter am Bericht

Welche Berichtsparameter in einem Bericht verwendet werden, definieren Sie bei der Erstellung des Berichtes im Report Editor. Hier legen Sie fest, welche Berichtsparameter bei der Erstellung eines abonnierbaren Berichtes sichtbar und überschreibbar sind und welche Berichtsparameter bereits definierte Werte enthalten.

Berichtsparameter am abonnierbaren Bericht

Sichtbare Berichtsparameter werden beim Einfügen eines abonnierbaren Berichtes im Manager angezeigt. Sie können diese Berichtsparameter, sofern sie auch überschreibbar sind, weiter konfigurieren. Das heißt Sie legen fest, welche Berichtsparameter für den Benutzer des Web Portals sichtbar und überschreibbar sind und definieren Parameterwerte.

Berichtsparameter am Berichtsabonnement

Berichtsparameter, die im abonnierbaren Bericht als sichtbar und überschreibbar gekennzeichnet sind, werden den Web Portal Benutzern bei der Erstellung ihrer persönlichen Berichtsabonnements angezeigt. Sofern die Berichtsparameter auch überschreibbar sind, kann der Web Portal Benutzer die Werte der Berichtsparameter weiter anpassen.

> HINWEIS: Alle Berichtsparameter die Anwendern zur Verfügung stehen sollen, beispielsweise bei der Anzeige der Berichte, der Erstellung abonnierbarer Berichte im Manager oder in den Berichtsabonnements im Web Portal, müssen Sie bereits im Bericht definieren.

Detaillierte Informationen zum Thema

- Berichtparameter bearbeiten auf Seite 459
- Allgemeine Parametereinstellungen bearbeiten auf Seite 460
- Definition der Parameterwerte bearbeiten auf Seite 461
- Einstellungen für Wertberechnung auf Seite 464

Berichtparameter bearbeiten

Um Berichtsparameter zu bearbeiten

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Parameter**.
3. Wählen Sie in der Liste **Vorhandene Parameter** den Berichtsparameter.

   - ODER -

   Klicken Sie die Schaltfläche **Hinzufügen**.

   Ein neuer Berichtsparameter wird erstellt.
4. Bearbeiten Sie die Eigenschaften des Berichtsparameters.
5. Speichern Sie die Änderungen.

Um einen Berichtsparameter zu löschen

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Parameter**.

3. Wählen Sie in der Liste **Vorhandene Parameter** den Berichtsparameter.
4. Klicken Sie die Schaltfläche **Löschen**.
5. Speichern Sie die Änderungen.

Verwandte Themen

- Berichtsparameter auf Seite 458
- Allgemeine Parametereinstellungen bearbeiten auf Seite 460
- Definition der Parameterwerte bearbeiten auf Seite 461
- Einstellungen für Wertberechnung auf Seite 464

Allgemeine Parametereinstellungen bearbeiten

Um die allgemeinen Parametereinstellungen zu bearbeiten

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Parameter**.
3. Wählen Sie in der Liste **Vorhandene Parameter** den Berichtsparameter und wählen Sie den Tabreiter **Allgemein**.

Tabelle 161: Allgemeine Parametereinstellungen

Eigenschaft	Beschreibung
Parametername	Bezeichnung des Parameters. HINWEIS: Der Name muss mit dem Namen des Parameters in der Datenabfrage übereinstimmen.
Parametertyp	Typ des Parameters. Zulässige Werte sind: • **Fest**: Es wird ein fester Parameterwert verwendet. Erfassen Sie auf dem Tabreiter **Wertdefinition** den Parameterwert. • **Benutzerabfrage**: Der Benutzer muss zur Laufzeit einen Parameterwert über einen Abfragedialog auswählen. • **Berechnung**: Der Parameterwert wird bei Erstellung des Berichtes zur Laufzeit dynamisch berechnet. Erfassen Sie auf dem Tabreiter **Wertberechnung** die Tabellenspalte und die Bedingung für die Berechnung des Wertes. Abhängig vom Typ werden die weiteren Einstellungen eingeblendet oder ausgeblendet.
Anzeigename	Benutzerfreundliche Bezeichnung des Parameters. Um einen sprach-

Eigenschaft	Beschreibung
	abhängigen Anzeigenamen anzuzeigen, übersetzen Sie den eingegebenen Text über die Schaltfläche .
Beschreibung	Freitextfeld für zusätzliche Erläuterungen. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
Reihenfolge	Position des Parameters in der Anzeige im abonnierbaren Bericht und im Web Portal.
Pflichtparameter	Gibt an, ob es sich um einen Pflichtparameter handelt. Ein Pflichtparameter muss zwingend mit einem Wert bestückt werden.
Sichtbar	Gibt an, ob der Parameter bei der Erstellung eines Berichtsabonnements im Web Portal angezeigt wird.
Überschreibbar	Gibt an, ob der Wert des Parameters durch Web Portal Benutzer bei der Erstellung eines Berichtsabonnements überschrieben werden kann.

Verwandte Themen

- Berichtparameter bearbeiten auf Seite 459
- Definition der Parameterwerte bearbeiten auf Seite 461
- Einstellungen für Wertberechnung auf Seite 464

Definition der Parameterwerte bearbeiten

Legen Sie den Parameterwert fest und definieren wie der Parameterwert aussehen soll. Abhängig von den Werten der Parameterdefinition werden die weiteren Eingaben ausgeblendet oder eingeblendet.

Um die Definition der Parameter zu bearbeiten

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Parameter**.
3. Wählen Sie in der Liste **Vorhandene Parameter** den Berichtsparameter und wählen Sie den Tabreiter **Wertdefiniton**.

HINWEIS: Die Eingaben **Beispielwert** und **Standardwert** werden stark durch die Werte der Parameterdefinition beeinflusst. Dies äußert sich einerseits durch die dynamische Anpassung der Steuerelemente zur Auswahl eines Beispielwertes oder Standardwertes und anderseits durch die dynamische Anpassung der auswählbaren Werte selbst. Es wird daher empfohlen, diese Werte zuletzt zu bearbeiten.

Tabelle 162: Wertdefinition

Eigenschaft	Beschreibung
Datentyp	Datentyp des Parameters.
Datumszusatz	Zusätzliche Informationen zur Berechnung von Datums- und Zeitangaben für die Anzeige in der Benutzeroberfläche. Der Wert kann bearbeitet werden, wenn der Datentyp **Date** ausgewählt wurde.
Wertebereich	Gibt an, ob der Wert des Parameters innerhalb des definierten Wertebereichs liegen muss. Bei **Ja** werden zusätzliche Eingabefelder angezeigt.
Mehrwertig	Gibt an, ob der Parameter mehrere Werte annehmen kann. Wenn **Ja**, können Benutzer mehrere Werte aus einer Auswahlliste wählen.
Mehrzeilig	Gibt an, ob der Inhalt des Parameters mehrzeilig sein kann. Wenn **Ja**, sind Zeilenumbrüche zulässig.
Datenquelle	Typ der Datenquelle. Zulässige Werte sind: • **Kein**: Der Benutzer kann einen beliebigen Parameterwert angeben. • **Tabelle**: Der Benutzer wählt einen Wert aus der festgelegten Tabellenspalte aus. • **Liste zulässiger Werte**: Der Benutzer wählt einen Wert aus einer vorgegebenen Werteliste aus. Abhängig von der gewählten Datenquelle können zusätzliche Angaben erforderlich sein.
Tabellenspalte (Wertabfrage)	Zusätzliche Angabe für die Datenquelle **Tabelle**. Tabellenspalte zur Auswahl des Parameterwertes. Die Benutzer können einen Wert aus dieser Tabellenspalte auswählen. Ist der Parameter zusätzlich mehrwertig, können mehrere Werte dieser Tabellenspalte ausgewählt werden.
Anzeigemuster	Zusätzliche Angabe für die Datenquelle **Tabelle**. Anzeigemuster für Tabellenelemente in Listen in der Notation `%Spalte%`. Zulässig ist der `??`-Operator. Damit wird bei leerem Wert der Spalte, der Wert einer anderen Spalte angezeigt. Beispiel: `%Spalte1??Spalte2??Spalte3%`
Bedingung (Wertabfrage)	Zusätzliche Angabe für die Datenquelle **Tabelle**. Einschränkende Bedingung (Where-Klausel) zur Auswahl des Parameterwertes über eine Tabellenspalte. Die Benutzer können einen Wert aus der Ergebnismenge auswählen. Ist der Parameter zusätzlich mehrwertig, können mehrere Werte dieser Ergebnismenge ausgewählt werden.

Eigenschaft	Beschreibung

	In der Bedingung können andere Parameter nach folgender Syntax referenziert werden:
	`$PC(<Parametername>)$`
	Beispiel:
	`UID_Database = $PC(UID)$`
	wobei `UID` der Name des referenzierten Berichtsparameters ist.

Liste zulässiger Werte	Zusätzliche Angabe für die Datenquelle **Liste zulässiger Werte**. Liste, der in diesem Parameter zulässigen Werte in der Notation **Wert=Anzeigename**. Ist kein **=** angegeben, zählt der Eintrag als Wert und als Anzeigename. Beispiel: `1=intern 2=extern` Um sprachabhängige Anzeigenamen anzuzeigen, übersetzen Sie die einzelnen Anzeigenamen über die Schaltfläche .
Leerer Wert überschreibend	Gibt an, ob ein leerer Parameterwert den Standardwert überschreibt. Wenn die Option deaktiviert ist, wird der Standardwert verwendet, wenn kein Parameterwert angegeben ist.
Beispielwert	Beispielwert des Parameters. Der Beispielwert wird genutzt, um die Vorschau des Berichts zu erstellen. Wenn ein Wertebereich angegeben werden soll, werden die Eingabefelder **Beispielwert (von)** und **Beispielwert (bis)** angezeigt.
Standardwert	Standardwert des Parameters. Dieser wird verwendet, wenn der Benutzer keinen Parameterwert festlegt und die Option **Leerer Wert überschreibend** deaktiviert ist. Wenn ein Wertebereich angegeben werden soll, werden die Eingabefelder **Standardwert (von)** und **Standardwert (bis)** angezeigt.

Verwandte Themen

- Berichtparameter bearbeiten auf Seite 459
- Allgemeine Parametereinstellungen bearbeiten auf Seite 460
- Einstellungen für Wertberechnung auf Seite 464
- Anzeigemuster für die Listendarstellung auf Seite 146

Einstellungen für Wertberechnung

Um die Einstellungen für die Wertberechnung zu bearbeiten

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie im Eigenschaftendialog den Tabreiter **Parameter**.
3. Wählen Sie in der Liste **Vorhandene Parameter** den Berichtsparameter und wählen Sie den Tabreiter **Wertberechnung**.

Tabelle 163: Skripte zur Wertberechnung

Eigenschaft	Beschreibung
Tabellenspalte (kalk.)	Zusätzliche Angabe für den Parametertyp **Berechnung**. Tabellenspalte zur Auswahl des Parameterwertes. Der Parameterwert wird bei Erstellung des Berichtes zur Laufzeit ermittelt.
Bedingung (kalk.)	Zusätzliche Angabe für den Parametertyp **Berechnung**. Einschränkende Bedingung (Where-Klausel) zur Auswahl des Wertes über eine Tabellenspalte. Der Parameterwert wird bei Erstellung des Berichtes zur Laufzeit ermittelt. Ist der Parameter zusätzlich mehrwertig, werden mehrere Werte ermittelt. Wenn keine Bedingung angegeben ist und der Parameter nicht mehrwertig ist, wird der erste Wert verwendet, der über die Tabellenspalte ermittelt wird. Wenn der Parameter mehrwertig ist und keine Bedingung angegeben ist, werden alle ermittelten Werte verwendet.
Skript für Wertermittlung	Skript in VB.Net-Syntax für die Bearbeitung des Parameterwertes. Das Skript kann wie ein Formatierungsskript genutzt werden und den vorhandenen Parameterwert bearbeiten oder es kann den Parameterwert neu setzen.
Skript für Wertüberprüfung	Skript in VB.Net-Syntax für die Überprüfung zulässiger Werte des Parameters. Erfassen Sie ein Skript, welches die Benutzereingabe überprüft.

Verwandte Themen

- Berichtparameter bearbeiten auf Seite 459
- Allgemeine Parametereinstellungen bearbeiten auf Seite 460
- Definition der Parameterwerte bearbeiten auf Seite 461

Virtuellen Datenquellen verwenden

Virtuelle Datenquellen verwenden Sie, wenn Sie eine definierte Datenquelle innerhalb eines Berichtes mehrfach verwenden möchten, jedoch weitere Einschränkungen oder eine andere Datensortierung benötigen.

Um eine virtuelle Datenquelle zu erstellen

1. Öffnen Sie den Bericht im Report Editor.
2. Wählen Sie den Menüeintrag **Bericht > Neue virtuelle Datenquelle**.

   Es wird eine Dialogfenster geöffnet, indem alle vorhandenen Datenquellen des Berichtes angezeigt werden.
3. Konfigurieren Sie die Eigenschaften der virtuellen Datenquelle.

Verwandte Themen

- Datenquellen erstellen und bearbeiten auf Seite 446

Bearbeiten des Berichtsformulars

In der Bearbeitungsansicht des Report Editors erstellen und bearbeiten Sie die Formulare des Berichtes. In die Bearbeitungsansicht ist der Stimulsoft Reports.Ultimate Reportdesigner der integriert. Die genaue Beschreibung und Funktionalität zu den einzelnen Komponenten entnehmen Sie der Online-Hilfe der Firma Stimulsoft (www.stimulsoft.com).

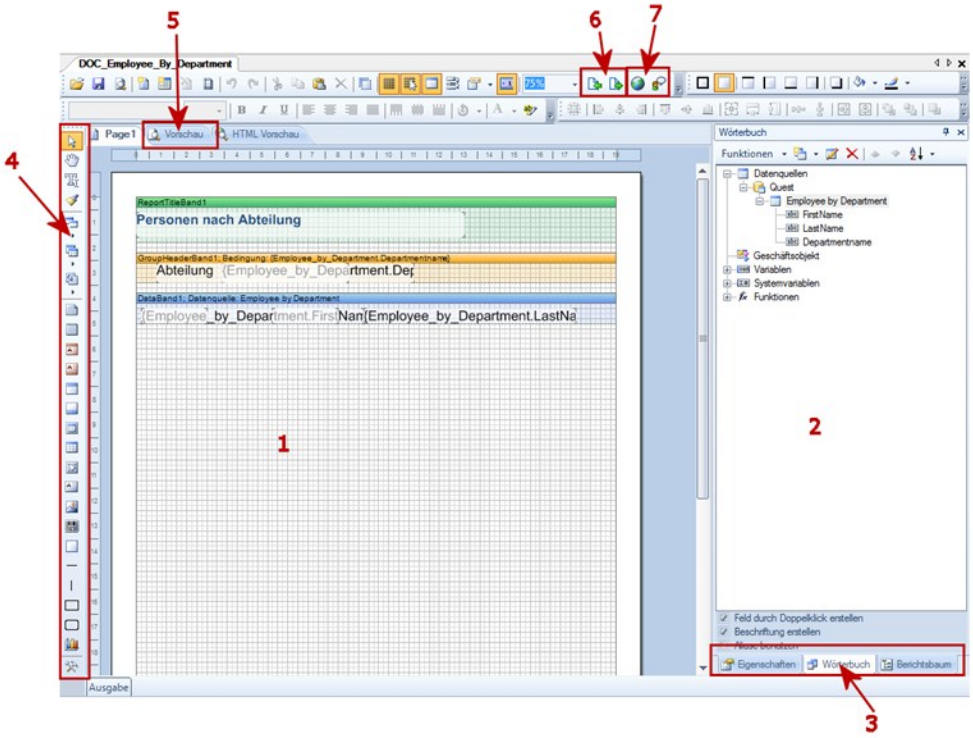
> HINWEIS: Beim ersten Start des Report Editors können Sie den Konfigurationstyp (**Basis-Bericht**, **Standard** oder **Professionell**) für die Berichte wählen. Der Konfigurationstyp entscheidet über den Umfang der dargestellten Eigenschaften bei der Bearbeitung eines Berichtes. Den Konfigurationstyp können Sie nachträglich in der Bearbeitungsansicht über das Kontextmenü des Eigenschaftenfensters ändern.

Die Symbolleiste des Stimulsoft Reports.Ultimate Reportdesigners wurde um folgende Funktionen erweitert.

Tabelle 164: Erweiterungen der Symbolleiste des Stimulsoft Reports.Ultimate Reportdesigners

Symbol	Bedeutung
	Importieren eines Berichtes (XML Format).
	Exportieren eines Berichtes (XML Format).
	Globalisierungseditor. Der Globalisierungseditor des Reportdesigners wird geöffnet.
	Der Dialog **Übersetzungen bearbeiten** wird geöffnet.

Abbildung 34: Reportdesigner mit Berichtsformular (1), Anzeige Wörterbuch/Eigenschaften (2), Umschalten zwischen Wörterbuch/Eigenschaften (3), Werkzeugpalette (4), Vorschau (5), Importieren/Exportieren von Berichtseiten (6), Übersetzen des Berichtes (7)



Detaillierte Informationen zum Thema

- Datenfelder in das Berichtsformular einfügen auf Seite 466
- Übersetzen von Berichten auf Seite 471
- Beispiel für einen einfachen Bericht mit Gruppierung der Daten auf Seite 468

Datenfelder in das Berichtsformular einfügen

Zur Anzeige der Daten im Bericht müssen Sie die gewünschten Steuerelemente auf dem Berichtsformular einfügen und mit den Spalten der Datenquellen verknüpfen. Nachdem Sie die Datenquellen erstellt haben, werden diese mit allen verwendeten Spalten im Wörterbuch des Reportdesigners unterhalb des Eintrags **Quest** aufgelistet. Die Berichtsparameter stehen als Variablen unterhalb des Eintrags **Quest** zur Verfügung.

Die genaue Beschreibung und Funktionalität zu den einzelnen Komponenten entnehmen Sie der Online-Hilfe der Firma Stimulsoft (www.stimulsoft.com).

Um Datenfelder in das Berichtsformular einzufügen

1. Wählen Sie die Spalte, die Sie in den Bericht einfügen möchten, im Wörterbuch (Tabreiter **Wörterbuch**) aus.
2. Platzieren Sie die Spalte per Drag and Drop auf dem Berichtsformular.

   Es wird ein neues Steuerelement auf dem Berichtsformular erzeugt und mit einigen Eigenschaften vorbelegt.

   TIPP: Weitere Steuerelemente können Sie nach Bedarf über die Werkzeugpaletten des Reportdesigners hinzufügen.
3. Über das Eigenschaftenfenster des Reportdesigners (Tabreiter **Eigenschaften**) passen Sie die einzelnen Steuerelemente an.
4. Über die **Vorschau** können Sie den Bericht während der Erstellung einsehen. Die Vorschau nutzt dabei die angegebenen Beispielwerte der Berichtsparameter für die Ermittlung der Daten.

Verwandte Themen

- Beispiel für einen einfachen Bericht mit Gruppierung der Daten auf Seite 468

Hinweise zur Verwendung von Datumsangaben in Berichten

Ist explizit kein Datum angegeben, wird intern das Datum 30.12.1899 verwendet. Dies ist bei Wertevergleichen zu beachten, beispielsweise bei der Verwendung in Berichten.

Um anstelle des internen Wertes **30.12.1899** eine andere Zeichenkette anzuzeigen, haben Sie folgende Möglichkeiten.

- Für Datenquellen mit dem Abfragemodul **SQL** können Sie die Datumskonvertierung in der Datenquelle anpassen.

  Beispiel:

  ```
  select ISNULL(convert(varchar, Person.ExitDate, 121), '-') as date_
  substituted
  ```
- Passen Sie bei der Erstellung des Berichtformulars den Ausdruck für die Darstellung der Datenspalten an.

  Beispiel:

  ```
  {IIF(Person.ExitDate.ToString() = "12/30/1899 12:00:00 AM","-
  ",Person.ExitDate)}
  ```
- Definieren Sie bei der Erstellung des Berichtformulars eine Bedingung für die Darstellung der Datenspalte.

  Bedingungen können Sie über das Symbol in der Symbolleiste im Reportdesigner hinzufügen.

Beispiel für einen einfachen Bericht mit Gruppierung der Daten

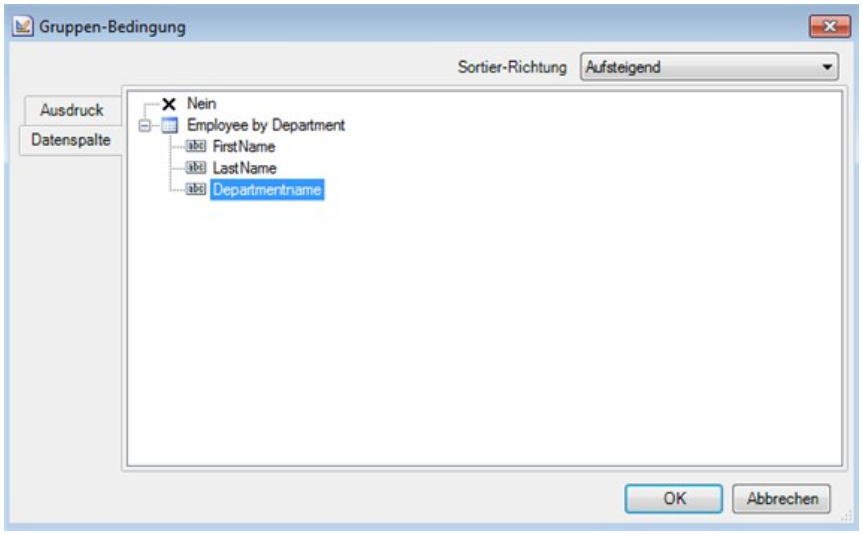
Es soll ein Bericht erstellt werden, der alle Personen gruppiert nach ihren Abteilungen auflistet.

1. Es wird ein neuer Bericht erzeugt.
   - Als Bezeichnung des Berichtes wird **CCC_Employee_by_Department** festgelegt. Als Anzeigename wird **Personen nach Abteilungen %UID%** festgelegt.
   - Es wird eine Datenquelle **Employee by Department** mit dem Abfragemodul **SQL** für den Bericht erstellt. Die Datenabfrage soll die Personen liefern die einer Abteilung zugeordnet sind. Die Abteilung wird über den Objektschlüssel (`XObjektKey`) ermittelt. Dieser wird als Parameter an den Bericht übergeben. Abgefragt werden Vorname (`Firstname`) und Nachname (`Lastname`) der Person sowie der Name der Abteilung (`Departmentname`).

     ```
     Select Firstname, Lastname, Departmentname
          from person join Department
          on person.uid_Department = department.uid_Department
          where Department.XObjectKey = @UIDDepartment
     ```
   - Der Parameter `UIDDepartment` wird zum Bericht hinzugefügt und mit einem Beispielwert für die Vorschau belegt.
2. Die Steuerelemente für die Datenbankspalten werden auf dem Berichtsformular angeordnet.

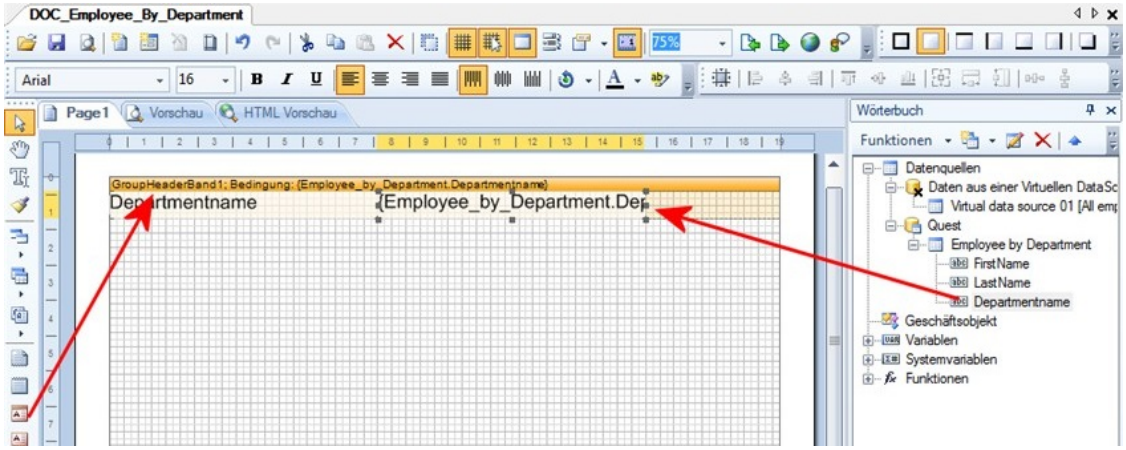
   Für die Gruppierung wird über die Werkzeugpalette des Reportdesigners ein Band vom Typ **Gruppenkopf** in das Berichtsformular eingefügt. Als Gruppen-Bedingung wird der Spaltenname, nach dem gruppiert werden soll, angegeben, im Beispiel `Departmentname`.

Abbildung 35: Festlegen der Gruppen-Bedingung



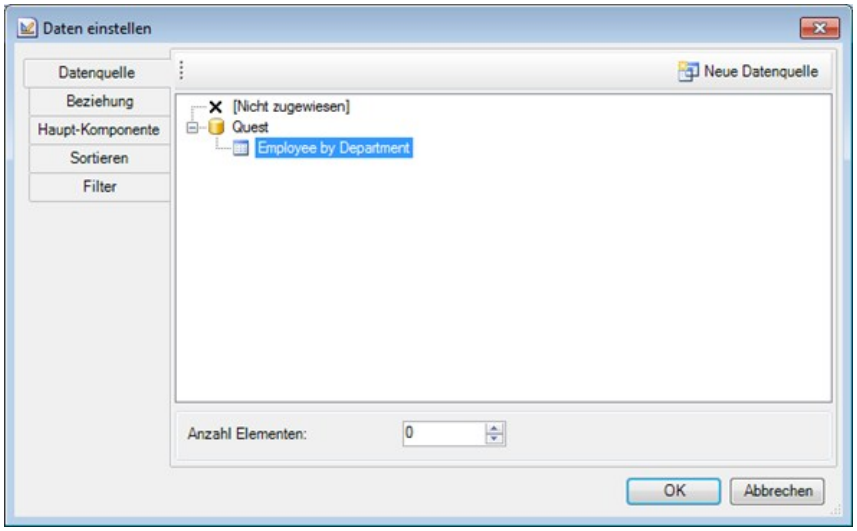
3. Die Spalte `Departmentname` wird per Drag and Drop aus dem Wörterbuch des Reportdesigners (Tabreiter **Wörterbuch**) in den Gruppenkopf eingefügt. Dabei wird ein neues Steuerelement auf dem Berichtsformular erzeugt.

Abbildung 36: Erstellen einer Gruppierung



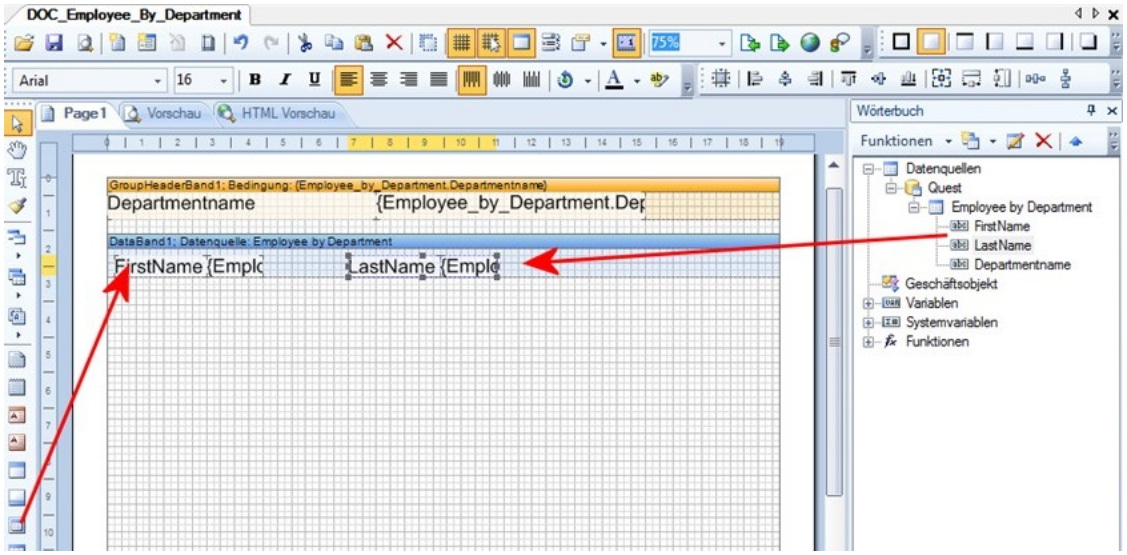
4. Für die Darstellung der Personen wird über die Werkzeugpalette des Reportdesigners ein neues Band vom Typ **Datenband** in das Berichtsformular eingefügt. Als Datenquelle wird die Datenquelle **Employee by Department** festgelegt.

Abbildung 37: Festlegen der Datenquelle



5. Die Spalten `Lastname` und `Firstname` werden per Drag and Drop aus dem Wörterbuch des Reportdesigners (Tabreiter **Wörterbuch**) in das Datenband eingefügt. Dabei werden die entsprechenden Steuerelemente auf dem Berichtsformular erzeugt.

Abbildung 38: Anordnung der Steuerelemente auf dem Berichtsformular



6. Weitere Steuerelemente beispielsweise ein Titel (**PageHeader**) können nach Bedarf mit dem Reportdesigner hinzugefügt werden. Über das Eigenschaftenfenster des Reportdesigners (Tabreiter **Eigenschaften**) werden die einzelnen Steuerelemente angepasst.
7. Über die Vorschau kann der Bericht während der Erstellung eingesehen werden. Die Vorschau nutzt dabei den angegebenen Beispielwert des Parameters für die Ermittlung der Daten.

Detaillierte Informationen zum Thema

- Allgemeinen Berichteigenschaften bearbeiten auf Seite 444
- Datenquellen erstellen und bearbeiten auf Seite 446
- Hinweise zur Verwendung von Datumsangaben in Berichten auf Seite 467
- Berichtsparameter auf Seite 458
- Bearbeiten des Berichtsformulars auf Seite 465

Übersetzen von Berichten

Ein Bericht kann verschiedene Elemente enthalten, die für die sprachabhängige Anzeige des Berichtes zu übersetzen sind.

- Datenbankspalten, die in der Berichtsdefinition verwendet werden.

  Datenbankspalten übersetzen Sie im Wörterbucheditor des Designers.
- Anzeigename/ ReportAlias des Berichtes.

  Der Anzeigename des Bericht kann bei der Erstellung eines Berichtes als **ReportAlias** verwendet werden. Der Anzeigename wird im Eigenschaftendialog des Berichtes eingetragen. Übersetzen Sie den eingegebenen Text über die Schaltfläche .
- Text-Elemente auf dem Berichtsformular.

  Die Text-Elemente übersetzen Sie direkt im Report Editor mit dem Globalisierungseditor.

Um alle Text-Elemente eines Berichtes zu übersetzen

1. Wählen Sie den Bericht in der Berichtliste und öffnen Sie den Bericht per Maus-Doppelklick oder über den Kontextmenüeintrag **Bearbeiten**.

   Das Berichtsformular wird im Reportdesigner geöffnet.
2. Starten Sie den Globalisierungseditor.
   - Klicken Sie auf die Schaltfläche in der Symbolleiste des Reportdesigners.

     - ODER -
   - Wählen Sie im Eigenschaftenfenster des Reportdesigners auf den Tabreiter **Eigenschaften** in der Auswahlliste den Bericht aus und Sie öffnen den Globalisierungseditor über den Eintrag **Globalisierungs-Zeichenfolgen**.

     HINWEIS: Den Globalisierungseditor können Sie über das Eigenschaftenfenster des Reportdesigners nur starten, wenn Sie den Konfigurationstyp **Professionell** gewählt haben. Den Konfigurationstyp können Sie nachträglich in der Bearbeitungsansicht über das Kontextmenü des Eigenschaftenfensters ändern.

3. Stellen Sie sicher, dass die Schaltfläche **Bericht beim Starten automatisch lokalisieren** aktiviert ist.

   Damit wird der Bericht in der jeweiligen Sprachkultur generiert.

4. Fügen Sie über **Kultur hinzufügen** die Kultur für die Sprache ein und übersetzen Sie die einzelnen Einträge.

Um einzelne Text-Elemente zu übersetzen

1. Wählen Sie den Bericht in der Berichtliste und öffnen Sie den Bericht per Maus-Doppelklick oder über den Kontextmenüeintrag **Bearbeiten**.

   Das Berichtsformular wird im Reportdesigner geöffnet.

2. Wählen Sie das Text-Element auf dem Berichtsformular.
3. Starten Sie den Dialog über die Schaltfläche in der Symbolleiste des Reportdesigners.
4. Übersetzen Sie den Text und übernehmen Sie die Änderungen mit **OK**.

Verwandte Themen

- Übersetzungen im Wörterbucheditor anzeigen auf Seite 238
- Allgemeinen Berichteigenschaften bearbeiten auf Seite 444

Berichte in die Benutzeroberfläche einbinden

Um einen Bericht in One Identity Manager-Werkzeugen, wie beispielsweise dem Manager, darzustellen, müssen Sie den Bericht als kundenspezifisches Oberflächenformular in die Benutzeroberfläche einbinden.

Berichte, die Sie im Report Editor erzeugen, können Sie für Statistiken im Infosystem des Managers anzeigen. Dazu müssen Sie die Benutzeroberfläche für die Anwendung **Manager** anpassen. Der Bericht wird bei Doppelklick auf die Kopfzeile der Statistik geöffnet.

Um ein Oberflächenformular zu erstellen

1. Wählen Sie im Designer die Kategorie **Benutzeroberfläche > Formulare > Oberflächenformulare**.
2. Wählen Sie die Aufgabe **Formulare bearbeiten**.
3. Wählen Sie den Menüeintrag **Formular > Einfügen**.
4. Bearbeiten Sie die Stammdaten des Oberflächenformulars.

   Beachten Sie die folgenden Besonderheiten:

- Verwenden Sie die Formulardefinition **VI_Report**.

  Diese Formulardefinition ist zur Anzeige in der grafischen Benutzeroberfläche und zur Anzeige in Webapplikationen konfiguriert. Damit müssen Sie nur ein Oberflächenformular einrichten. Anhand des Einsatzzweckes wird dynamisch entschieden, welche der Formularvorlagen zur Darstellung des Oberflächenformulars genutzt wird.

- In den Konfigurationsdaten des Formulars übergeben Sie in der Sektion `Properties` den Namen des auszuführenden Berichtes sowie die Berichtsparameter.

  Syntax:

```
<DialogSheetDefinition FormatVersion="1.0">
    <Properties>
        <Property Name="ReportName">ReportName aus Tabelle
        DialogReport</Property>
        <Property Name="ReportParameter.Parameter1">Wert1</Property>
        <Property Name="ReportParameter.Parameter2">Wert2</Property>
    </Properties>
</DialogSheetDefinition>
```

  Beispiel:

```
<DialogSheetDefinition FormatVersion="1.0">
    <Properties>
        <Property Name="ReportName">CCC_Employee_by_
        Department</Property>
        <Property Name="ReportParameter.UIDDepartment">%UID_
        Department%</Property>
    </Properties>
</DialogSheetDefinition>
```

5. Weisen Sie das Oberflächenformular an die Anwendungen und an die Berechtigungsgruppen zu.
6. (Optional) Weisen Sie das Oberflächenformular an die Objektdefinitionen zu.
7. (Optional) Weisen Sie das Oberflächenformular an die Menüeinträge zu.

Verwandte Themen

- Bearbeiten von Oberflächenformularen auf Seite 155
- Oberflächenformulare erstellen
- Oberflächenformulare an Anwendungen zuweisen
- Oberflächenformulare an Berechtigungsgruppen zuweisen
- Oberflächenformulare an Menüeinträge zuweisen auf Seite 161

- Oberflächenformulare an Objektdefinitionen zuweisen auf Seite 160
- Berichte in Statistiken verwenden auf Seite 197

Berichte zyklische erzeugen und exportieren

Um die Erstellung von Berichten und den Export auch zyklisch oder ereignisgesteuert vorzunehmen, erstellen Sie kundenspezifische Prozesse.

Berichte können Sie mit der Prozesskomponente `ReportComponent` erstellen und in verschiedene Dateiformate exportieren. Es werden die Formate `HTML`, `PDF`, `RTF`, `TEXT`, `XLS`, `TIFF`, `XML`, `CSV`, `XPS`, `DOCX` und `XLSX` unterstützt.

Um Berichte im CSV Format zu exportieren, können Sie auch die Prozesskomponente `ScriptComponent` mit den Prozessfunktionen `CSVExport` oder `CSVExportSingle` verwenden.

> HINWEIS: Nutzen Sie in den Prozessen den Standard Berichtserver als ausführenden Server.

Verwandte Themen

- Definieren von Prozessen auf Seite 248

15

Erweiterung des One Identity Manager Schemas um kundenspezifische Tabellen oder Spalten

Mit der im One Identity Manager verwendeten Objekttechnologie ist es möglich, das bestehende Anwendungsdatenmodell kundenspezifisch um Spalten und Tabellen auf Datenbankebene zu erweitern, so dass diese Erweiterungen auf der Objektebene mit allen Funktionen zur Verfügung stehen. Die kundenspezifische Erweiterung des Systemdatenmodells ist nicht empfohlen.

Voraussetzung für die Schemaerweiterung sind Grundkenntnisse des eingesetzten Datenbanksystems SQL Server. Es wird vorausgesetzt, dass Sie das Konzept sowie die Architektur des One Identity Manager verstanden haben.

Setzen Sie zur kundenspezifischen Erweiterung des One Identity Manager Schemas das Programm Schema Extension ein. Folgende Erweiterungen können Sie im Schema Extension ausführen:

- Erstellen neuer Tabellen
- Erstellen neuer Zuordnungstabellen
- Erstellen neuer Spalten
- Erstellen neuer Sichten
- Erstellen neuer Indexe
- Entfernen kundenspezifischer Schemaerweiterungen

> HINWEIS: Kundenspezifische Schemaerweiterungen können Sie nur auf Datenbanken mit einer Staging-Ebene **Testumgebung** oder **Entwicklungssystem** entfernen. Ein Transport der Änderungen ist nicht möglich.

Durch das Programm Schema Extension werden die Schemaerweiterungen in der Datenbank angelegt und die notwendigen Erweiterungen im One Identity Manager Schema sichergestellt. Es werden die grundlegenden Tabellendefinitionen und Spaltendefinitionen der kundenspezifischen Tabellen in den Tabellen `DialogTable`, `DialogColumn`, `QBMRelation` und `DialogValidDynamicRef` eingetragen. Die Eigenschaften müssen Sie im Designer weiter an die gewünschten Anforderungen anpassen.

Im Designer finden Sie verschiedene Konsistenzprüfungen. Führen Sie nach einer Schemaerweiterung sowie nach Änderungen der Tabellendefinitionen und Spaltendefinitionen die Konsistenzprüfungen aus und verwenden Sie die Reparaturmethoden. Ausführliche Informationen zum Prüfen der Datenkonsistenz finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Kundenspezifische Funktionen, Trigger oder Datenbankprozeduren können Sie nicht mit dem Programm Schema Extension erzeugen. Sollten Sie kundenspezifische Funktionen, Trigger oder Datenbankprozeduren benötigen, fügen Sie diese in einem geeigneten Programm zur Ausführung von SQL Abfragen in die Datenbank ein. Halten Sie die folgenden Konventionen für die Bezeichnung der Datenbankbestandteile ein.

- Die Bezeichnungen beginnen mit der Zeichenkette **CCC_**.
- Alle Bezeichnungen sind maximal 30 Zeichen lang.
- Für die Schreibweise von Bezeichnungen wird UpperCamelCase empfohlen.

Detaillierte Informationen zum Thema

- Neue Tabelle erstellen auf Seite 477
- Neue Zuordnungstabelle erstellen auf Seite 489
- Tabelle erweitern auf Seite 478
- Datenbanksichten mit dem Typ Read only erstellen auf Seite 486
- Datenbanksichten mit dem Typ Union erstellen auf Seite 488
- Indexe erstellen auf Seite 490
- Kundenspezifische Schemaerweiterungen entfernen auf Seite 491
- Berechtigungen für die Schemaerweiterungen auf Seite 493
- Änderungskennzeichen für die Schemaerweiterungen auf Seite 494
- Schemaerweiterungen in die Datenbank übernehmen auf Seite 494

Verwandte Themen

- Grundlagen zum One Identity Manager Schema auf Seite 47
- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52
- Empfehlungen für die weitere Konfiguration kundenspezifischer Schemaerweiterungen auf Seite 495
- Verwaltung von kundenspezifischen Datenbankobjekten innerhalb der Datenbank auf Seite 498

Neue Tabelle erstellen

Mit dieser Aufgabe erstellen Sie eine einfache Tabelle im One Identity Manager Schema.

Technische Details

- Die technische Bezeichnung der Tabelle wird automatisch nach dem Schema `CCC<Tabellenname>` gebildet.
- Folgende Spalten werden automatisch erzeugt:
  - Primärschlüsselspalte

    Die Primärschlüsselspalte wird automatisch als UID übernommen. Der Name der Primärschlüsselspalte wird nach dem Schema `UID_CCC<Tabellenname>` gebildet.
  - X-Spalten (`XUserInserted`, `XUserUpdated`, `XDateInserted`, `XDateUpdated`, `XTouched`, `XObjectKey`, `XMarkedForDeletion`)

Um im Schema Extension eine einfache Tabelle zu erstellen

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.
2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank.
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Neue Tabelle**.
6. Auf der Seite **Neue Tabelle erstellen** erfassen Sie die folgenden Informationen.

   Tabelle 165: Tabelleneigenschaften

Eigenschaft	Beschreibung
Tabelle	Technische Bezeichnung der Tabelle.
Anzeigename	Anzeigename der Tabelle. Der Anzeigename wird beispielsweise in der Datenbanksuche oder der Fehlerausgabe zur Anzeige der Tabelle verwendet.
Beschreibung	Bemerkungen zur Verwendung der Tabelle.

7. Auf der Seite **Spalten der Tabelle definieren** erstellen Sie die neuen Spalten. Weitere Informationen finden Sie unter Definieren von Spalten auf Seite 478.

Verwandte Themen

- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52
- Tabelle erweitern auf Seite 478
- Neue Zuordnungstabelle erstellen auf Seite 489
- Datenbanksichten mit dem Typ Read only erstellen auf Seite 486
- Datenbanksichten mit dem Typ Union erstellen auf Seite 488

Tabelle erweitern

Um im Schema Extension eine bestehende Tabelle zu erweitern

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.
2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank.
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Tabelle erweitern**.
6. Auf der Seite **Tabelle erweitern** wählen Sie in der Auswahlliste **Tabelle** die Tabelle, die Sie erweitern möchten.
7. Auf der Seite **Spalten der Tabelle definieren** erstellen Sie die neuen Spalten. Weitere Informationen finden Sie unter Definieren von Spalten auf Seite 478.

Verwandte Themen

- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52
- Neue Tabelle erstellen auf Seite 477
- Neue Zuordnungstabelle erstellen
- Datenbanksichten mit dem Typ Read only erstellen auf Seite 486
- Datenbanksichten mit dem Typ Union erstellen auf Seite 488

Definieren von Spalten

Auf der Seite **Spalten der Tabelle definieren** im Schema Extension wird zur gewählten Tabelle angezeigt, welche Spalten bereits vorhanden sind und wie viele Ressourcen für

neue Spalten frei sind.

HINWEIS: Beachten Sie bei der Erweiterung einer Tabelle die maximal zulässige Größe einer Tabelle.

Detaillierte Informationen zum Thema

- Einfache Spalte erstellen auf Seite 479
- Fremdschlüsselspalte erstellen auf Seite 480
- Dynamische Fremdschlüsselspalte erstellen auf Seite 482
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483
- Erweiterte Konfiguration der Spalten auf Seite 483

Verwandte Themen

- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52

Einfache Spalte erstellen

Technische Details

- Die technische Bezeichnung der Spalte wird automatisch nach dem Schema `CCC_<Spaltenname>` gebildet.

Um im Schema Extension eine einfache Spalte zu erstellen

1. Auf der Seite **Spalten der Tabelle definieren** klicken Sie die Schaltfläche .
2. Wählen Sie die Option **Einfache Spalte** und erfassen Sie unter **Spaltenname** die Bezeichnung der Spalte.
3. Klicken Sie **OK**.
4. Auf der Seite **Spalten der Tabelle definieren** erfassen Sie mindestens die folgenden Informationen.

Tabelle 166: Allgemeine Spalteneigenschaften

Eigenschaft	Beschreibung
Pflichtfeld	Gibt an, ob die Spalte ein Pflichtfeld ist.
Datentyp	Datentyp der Spalte. Angeboten werden die zulässigen .Net-Datentypen. Diese werden intern in SQL-Datentypen abgebildet.
Länge	Länge der Spalte. Die Angabe der Länge erfolgt nur beim .Net-Datentyp `String`.
Anzeigename	Anzeigename der Spalte.

5. (Optional) Klicken Sie um weitere Spalteneigenschaften zu konfigurieren. Weitere Informationen finden Sie unter Erweiterte Konfiguration der Spalten auf Seite 483.

Verwandte Themen

- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483
- Fremdschlüsselspalte erstellen auf Seite 480
- Dynamische Fremdschlüsselspalte erstellen auf Seite 482

Fremdschlüsselspalte erstellen

Einschränkungen

- Die referenzierte Tabelle hat einen einspaltigen Primärschlüssel.

Technische Details

- Die technische Bezeichnung der Spalte wird automatisch nach dem Schema `CCC_<Spaltenname>` gebildet.
- Fremdschlüsselspalten werden mit dem Datentyp `String` und einer Länge von 38 Zeichen erstellt.
- Die Tabellenbeziehungen und Spaltenbeziehungen werden automatisch erzeugt.
  - Die Beziehungs-ID's folgen der Namenskonvention:

    `CCC-<Datenbank-ID>-<4-stellige laufende Nummer>`
  - Wird eine Datenbanksicht um eine Fremdschlüsselspalte erweitert, folgen die Beziehungs-ID's der Namenskonvention:

    `CCC-<Datenbank-ID>-<4-stellige laufende Nummer> <Name der referenzierten Tabelle>`
  - Wird eine Spalte aus einer Basistabelle mit dem Tabellentyp **Basistabelle** referenziert, werden die Tabellenbeziehungen und Spaltenbeziehungen auch für die Basistabelle erzeugt.
  - Die Überprüfung der referentiellen Integrität erfolgt per DLL oder Trigger.

Um im Schema Extension eine Fremdschlüsselspalte zu erstellen

1. Auf der Seite **Spalten der Tabelle definieren** klicken Sie die Schaltfläche .
2. Wählen Sie die Option **Fremdschlüsselspalte** und erfassen Sie folgende Informationen.

- **Spaltenname**: Erfassen Sie die Bezeichnung der Spalte. Die Bezeichnung der Fremdschlüsselspalte sollte, soweit möglich, der Bezeichnung der Primärschlüssel der referenzierten Tabelle entsprechen.
- **aus Tabelle**: Wählen Sie die referenzierte Tabelle.

3. Klicken Sie **OK**.
4. Auf der Seite **Spalten der Tabelle definieren** erfassen Sie mindestens die folgenden Informationen.
   - **Pflichtfeld**: Gibt an, ob die Spalte ein Pflichtfeld ist.
   - **Anzeigename**: Anzeigename der Spalte.
5. Klicken Sie und erfassen Sie auf dem Tabreiter **Key-Spaltenwerte** folgende Informationen.

Tabelle 167: Eigenschaften für Fremdschlüsselspalten

Eigenschaft	Bemerkungen
Fremdschlüssel	Gibt an, ob es sich um eine Fremdschlüsselspalte handelt. Aktivieren Sie die Option.
aus Tabelle	Referenzierte Tabelle bei Fremdschlüsselbeziehungen.
Lösch-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Löschen eines Objektes.
Einfüge-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Einfügen eines Objektes.

6. (Optional) Konfigurieren Sie weitere Spalteneigenschaften.Weitere Informationen finden Sie unter Erweiterte Konfiguration der Spalten auf Seite 483.

Verwandte Themen

- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52
- Tabellenbeziehungen auf Seite 109
- Einfache Spalte erstellen auf Seite 479
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483
- Dynamische Fremdschlüsselspalte erstellen auf Seite 482

Dynamische Fremdschlüsselspalte erstellen

Technische Details

- Die technische Bezeichnung der Spalte wird automatisch nach dem Schema `CCC_<Spaltenname>` gebildet.
- Dynamische Fremdschlüsselspalten werden mit dem Datentyp `String` und einer Länge von 138 Zeichen erstellt.

Um im Schema Extension eine dynamische Fremdschlüsselspalte zu erstellen

1. Auf der Seite **Spalten der Tabelle definieren** klicken Sie die Schaltfläche .
2. Wählen Sie die Option **Dynamischer Fremdschlüssel** und erfassen Sie unter **Spaltenname** die Bezeichnung der Spalte
3. Klicken Sie **OK**.
4. Auf der Seite **Spalten der Tabelle definieren** erfassen Sie mindestens die folgenden Informationen.
   - **Pflichtfeld**: Gibt an, ob die Spalte ein Pflichtfeld ist.
   - **Anzeigename**: Anzeigename der Spalte.
5. Klicken Sie und erfassen Sie auf dem Tabreiter **Dynamischer Fremdschlüssel** folgende Informationen.

Tabelle 168: Eigenschaften für Fremdschlüsselspalten

Eigenschaft	Bemerkungen
Referenzierte Tabelle	Wählen Sie die Tabellen, die referenziert werden sollen. Ist keine Einschränkung festgelegt, sind alle Tabellen zulässig.
Lösch-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Löschen eines Objektes.
Einfüge-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Einfügen eines Objektes.

1. (Optional) Konfigurieren Sie weitere Spalteneigenschaften.Weitere Informationen finden Sie unter Erweiterte Konfiguration der Spalten auf Seite 483.

Verwandte Themen

- Tabellenarten und Standardspalten im One Identity Manager Datenmodell auf Seite 52
- Dynamische Fremdschlüssel auf Seite 113
- Einfache Spalte erstellen auf Seite 479

- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483
- Fremdschlüsselspalte erstellen auf Seite 480

Neue Spalte für Datenbanksichten mit dem Typ View erstellen

Handelt es sich bei der zu erweiternden Tabelle um eine Datenbanksicht mit dem Tabellentyp **View**, wird die Auswahl für neue Spalten auf die kundenspezifischen Spalten der zugrunde liegenden Basistabelle beschränkt. Es werden nur die kundenspezifischen Spalten der Basistabelle zur Auswahl angeboten, die in der Datenbanksicht noch nicht verwendet werden.

- Erweitern Sie zunächst die Basistabelle um die neue Spalte (einfache Spalte oder Fremdschlüsselspalte).
- Anschließend erweitern Sie die Datenbanksicht um die neuen Spalten.

Um im Schema Extension eine neue Spalte für eine Datenbanksicht mit dem Typ View zu erstellen

1. Auf der Seite **Spalten der Tabelle definieren** klicken Sie die Schaltfläche .
2. Erfassen Sie folgende Informationen.
   - **Spaltenname**: Erfassen Sie die Bezeichnung der Spalte.
   - **Basisspalte**: Wählen Sie die Spalte der Basistabelle, die in die Datenbanksicht eingefügt werden soll.
3. Klicken Sie **OK**.
4. (Optional) Klicken Sie um weitere Spalteneigenschaften zu konfigurieren.Weitere Informationen finden Sie unter Erweiterte Konfiguration der Spalten auf Seite 483.

Verwandte Themen

- Einfache Spalte erstellen auf Seite 479
- Datenbanksichten vom Typ View auf Seite 59

Erweiterte Konfiguration der Spalten

Um die Spalteneigenschaften zu bearbeiten

1. Wählen Sie auf der Seite **Spalten der Tabelle definieren** die Spalte aus und wählen Sie die Schaltfläche .
2. Konfigurieren Sie die Spalteneigenschaften.

3. Auf dem Tabreiter **Erweitert** erfassen Sie folgende Informationen.

Tabelle 169: Erweiterte Konfiguration der Spalten

Eigenschaft	Bemerkungen
Name	Technische Bezeichnung der Spalte. Der Name der Spalte wird gebildet aus `CCC_<Spaltenname>`.
Datentyp	Datentyp der Spalte. Angeboten werden die zulässigen .Net-Datentypen. Diese werden intern in SQL-Datentypen abgebildet. Als Datentypen sind nur die bereits im Datenmodell des One Identity Manager verwendeten Datentypen zulässig.
Länge	Länge der Spalte. Die Angabe der Länge erfolgt nur beim .Net-Datentyp `String`. Für Spalten die UID's enthalten, tragen Sie den Wert 38 ein.
Spalte enthält UID's	Gibt an, ob es sich bei der Spalte um eine UID handelt. Die Angabe ist nur zulässig für Spalten mit dem .Net-Datentyp `String` und einer Länge von 38 Zeichen.
Spalte enthält Unicode	Gibt an, ob die Spalte Unicode enthält. Die Angabe ist nur zulässig für den .Net-Datentypen `String`.

4. Auf dem Tabreiter **Basiswerte** erfassen Sie folgende Informationen.

Tabelle 170: Basiswerte der Spalten

Eigenschaft	Bemerkungen
Primärschlüssel	Gibt an, ob die Spalte als Primärschlüssel verwendet wird.
Pflichtfeld	Gibt an, ob die Spalte ein Pflichtfeld ist.
Anzeigename	Anzeigename der Spalte.
Anzeige im Assistenten	Gibt an, ob die Spalte im Regeleditor für Complianceregeln zum Erstellen von Abfragen und in tabellarischen Übersichten im Web Portal zur Anzeige angeboten wird.
Berechtigungen nicht automatisch erweitern	Für kundenspezifische Spalte an einer vordefinierten Tabelle erfolgt keine automatische Berechtigung von vordefinierten >n, obwohl der Konfigurationsparameter **Common | AutoExtendPermissions** gesetzt ist.
Kommentar	Hinweise zur Nutzung der Spalte.
Initialwert	Initialer Wert für die Spalte. Dieser Wert wird in die bestehenden Datensätze der erweiterten Tabelle übernommen. Für nummerische Datentypen ist der Initialwert **0**. Für den Datentyp `Bool` ist der Initialwert **False**.

Eigenschaft	Bemerkungen
Sortierreihenfolge	Positionierung der Spalte zur Darstellung auf einem generischen Formular sowie auf dem kundenspezifischen Tabreiter der Standardformulare. Spalten, deren Wert kleiner **1** ist, werden nicht auf den Formularen angezeigt.

5. Auf dem Tabreiter **Key-Spaltenwerte** erfassen Sie folgende Informationen für Fremdschlüsselspalten.

Tabelle 171: Eigenschaften für Fremdschlüsselspalten

Eigenschaft	Bemerkungen
Fremdschlüssel	Gibt an, ob es sich um eine Fremdschlüsselspalte handelt.
aus Tabelle	Referenzierte Tabelle bei Fremdschlüsselbeziehungen.
Lösch-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Löschen eines Objektes.
Einfüge-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Einfügen eines Objektes.

6. Auf dem Tabreiter **Dynamischer Fremdschlüssel** erfassen Sie folgende Informationen für dynamische Fremdschlüssel.

Tabelle 172: Eigenschaften für Fremdschlüsselspalten

Eigenschaft	Bemerkungen
Referenzierte Tabelle	Wählen Sie die Tabellen, die referenziert werden sollen. Ist keine Einschränkung festgelegt, sind alle Tabellen zulässig.
Lösch-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Löschen eines Objektes.
Einfüge-Beschränkungen	Beschränkung für die Prüfung der referentiellen Integrität beim Einfügen eines Objektes.

7. Klicken Sie **OK**.

Verwandte Themen

- Einfache Spalte erstellen auf Seite 479
- Fremdschlüsselspalte erstellen auf Seite 480
- Dynamische Fremdschlüsselspalte erstellen auf Seite 482
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483
- Tabellenbeziehungen auf Seite 109
- Dynamische Fremdschlüssel auf Seite 113

Datenbanksichten mit dem Typ Read only erstellen

Technische Details

- Die technische Bezeichnung der Datenbanksicht wird automatisch nach dem Schema `CCC<Tabellenname>` gebildet.
- Die erste Spalte der Datenbankabfrage (Sichtdefinition) wird als Primärschlüsselspalte der Datenbanksicht verwendet.

  HINWEIS: Es wird empfohlen die Primärschlüsselspalte der abgefragten Tabelle in der Sichtdefinition als erste Spalte zu referenzieren. Sollte dies nicht möglich sein, ist ein möglichst eindeutiges Merkmal zu wählen.
- Enthält eine Datenbanksicht Fremdschlüsselspalten, legen Sie fest, welche Zieltabelle referenziert werden soll. Die Tabellen-und Spaltenbeziehungen werden automatisch erzeugt.
- Soll die Datenbanksicht für die Volltextsuche indiziert werden, müssen die Spalten `XDateInserted`, `XDateUpdated` und `XObjectKey` vorhanden sein.

Um im Schema Extension eine Datenbanksicht mit dem Typ Read only zu erstellen

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.
2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank.
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Neue Sicht**.
6. Auf der Seite **Erstellen einer Sicht** erfassen Sie folgende Informationen.

Tabelle 173: Eigenschaften für Datenbanksichten

Eigenschaft	Beschreibung
Tabelle	Technische Bezeichnung der Datenbanksicht.
Anzeigename	Anzeigename der Datenbanksicht. Der Anzeigename wird beispielsweise in der Datenbanksuche oder der Fehlerausgabe zur Anzeige der Datenbanksicht verwendet.
Beschreibung	Bemerkungen zur Verwendung der Datenbanksicht.
Sichtdefinition	Tragen Sie die Datenbankabfrage als Select-Anweisung ein.

7. (Optional) Auf der Seite **Erzeugen von FK-Beziehungen für Sichten** erzeugen Sie die Fremdschlüsselbeziehungen.
    - Doppelklicken Sie auf das Symbol vor dem Spaltennamen und wählen Sie in der Spalte **Zieltabelle** die Zieltabelle.
8. Auf der Seite **Spalten der Tabelle definieren** erfassen Sie den Anzeigenamen der Spalten.
9. (Optional) Klicken Sie um weitere Spalteneigenschaften zu konfigurieren. Weitere Informationen finden Sie unter Erweiterte Konfiguration der Spalten auf Seite 483.

Verwandte Themen

- Datenbanksichten vom Typ Read only auf Seite 66
- Common Table Expressions in Read Only-Datenbanksichten verwenden auf Seite 487
- Datenbanksichten mit dem Typ Union erstellen auf Seite 488
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483

Common Table Expressions in Read Only-Datenbanksichten verwenden

Mit One Identity Manager 7.0 wurde das Verhalten bei Verwendung von Common Table Expressions mit Schlüsselwort `with` als Bedingung für Sichtdefinitionen in Datenbanksichten vom Typ **Read only** geändert. Die Bedingungen für Sichtdefinitionen werden in eine zusammenfassende Abfrage eingebettet. Damit ist nicht mehr gewährleistet, dass Common Table Expressions als allererster Ausdruck innerhalb einer Abfrage stehen.

Mögliche Fehlermeldung

```
(execute slot single)50000 0 re-throw in Procedure QBM_ZViewBuildR, Line 1050000
0 rethrow in Procedure QBM_PViewBuildR_intern, Line 10250000 0 re-throw in
Procedure QBM_PViewBuildR_intern, Line 8250000 0 re-throw in Procedure QBM_
PViewBuild_FromAddOn, Line 6550000 0 re-throw in Procedure QBM_PSQLCreate, Line
26156 0 detected in (...) Procedure ..., Line 6156 0 Incorrect syntax near the
keyword 'with'
```

Empfohlenes Vorgehen

1. Erstellen Sie eine Datenbanksicht unter Verwendung der Common Table Expression.

    Beispiel:

    ```
    create view CCC_Vxy as
    with myWithClause (column1, column2) as (
    ```

```
select 1 as column1, 2 as column2
)
select * from myWithClause
go
```

2. Verwenden Sie die Datenbanksicht in der zusätzlichen Sichtdefinition (QBMViewAddon) der Datenbanksichten vom Typ **Read only**.

```
select * from CCC_Vxy
```

Datenbanksichten mit dem Typ Union erstellen

Technische Details

- Die technische Bezeichnung der Datenbanksicht wird automatisch nach dem Schema `CCC<Tabellenname>` gebildet.
- Als erste Spalte der Datenbankabfrage (Sichtdefinition) muss der Objektschlüssel (`XObjectKey`) referenziert werden. Über den Objektschlüssel ist der schnellere Zugriff auf ein Einzelobjektes mit den dafür geltenden Berechtigungen möglich.
- Soll die Datenbanksicht für die Volltextsuche indiziert werden, müssen die Spalten `XDateInserted`, `XDateUpdated` und `XObjectKey` vorhanden sein.

Um im Schema Extension eine Datenbanksicht mit dem Typ Union zu erstellen

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.
2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank.
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Vereinigungssicht erstellen**.
6. Auf der Seite **Vereinigungssicht erstellen** erfassen Sie folgende Informationen.

Tabelle 174: Eigenschaften für Datenbanksichten

Eigenschaft	Beschreibung
Tabelle	Technische Bezeichnung der Datenbanksicht.
Anzeigename	Anzeigename der Datenbanksicht. Der Anzeigename wird beispielsweise in der Datenbanksuche oder der Fehlerausgabe

Eigenschaft	Beschreibung
	zur Anzeige der Datenbanksicht verwendet.
Beschreibung	Bemerkungen zur Verwendung der Datenbanksicht.
Sichtdefinition	Tragen Sie die Datenbankabfrage als Select-Anweisung ein.

Verwandte Themen

- Datenbanksichten vom Typ Union auf Seite 63
- Datenbanksichten mit dem Typ Read only erstellen auf Seite 486
- Neue Spalte für Datenbanksichten mit dem Typ View erstellen auf Seite 483

Neue Zuordnungstabelle erstellen

Technische Details

- Die technische Bezeichnung der Tabelle wird automatisch nach dem Schema `CCC<Tabellenname>` gebildet.
- Die Spalten `XObjectKey` und `XMarkedForDeletion` werden automatisch erzeugt.
- Die Tabellenbeziehungen und Spaltenbeziehungen werden automatisch erzeugt.

Um im Schema Extension eine Zuordnungstabelle (M:N-Tabelle) zu erstellen

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.
2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank.
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Neue Zuordnungstabelle**.
6. Auf der Seite **Zuordnungstabelle erstellen** erfassen Sie folgende Informationen.

Tabelle 175: Eigenschaften für Zuordnungstabellen

Eigenschaft	Beschreibung
Tabelle	Technische Bezeichnung der Tabelle.
Anzeigename	Anzeigename der Tabelle. Der Anzeigename wird beispielsweise in der Datenbanksuche oder der Fehler-

Eigenschaft	Beschreibung
	ausgabe zur Anzeige der Tabelle verwendet.
Beschreibung	Bemerkungen zur Verwendung der Tabelle.
XOrigin-Spalte anlegen (für Zuweisungsbestellungen)	Die Herkunftsspalte (`XOrigin`) können Sie optional erzeugen. In dieser Spalte wird die Herkunft einer Zuweisung als Verknüpfung von Bit-Positionen abgelegt. Bei jedem Eintrag in die Zuordnungstabelle erfolgt entsprechend der Zuweisungsart eine Änderung der Bit-Positionen.
Verbundene Tabellen	Über die Auswahllisten **Linke Tabelle** und **Rechte Tabelle** legen Sie fest, welche Tabellen an der Zuordnungstabelle beteiligt sind.
Spaltennamen	Geben Sie für jede Tabellenseite in den Eingabefeldern **Spaltennamen** die beteiligten Spalten an. Es wird die Primärschlüsselspalte der Tabelle ausgewählt.

7. Auf der Seite **Spalten der Tabelle definieren** erfassen Sie den Anzeigenamen der Spalten.
8. (Optional) Klicken Sie um weitere Spalteneigenschaften zu konfigurieren. Weitere Informationen finden Sie unter Erweiterte Konfiguration der Spalten auf Seite 483.

Verwandte Themen

- Neue Tabelle erstellen auf Seite 477
- Datenbanksichten mit dem Typ Read only erstellen auf Seite 486
- Datenbanksichten mit dem Typ Union erstellen auf Seite 488

Indexe erstellen

Definieren Sie Indexe um den Zugriff auf die Datenbankspalten zu optimieren. Ein Index kann eine oder mehrere Datenbankspalten enthalten.

> HINWEIS: Für Tabellen, die Sie mit dem Programm Schema Extension erstellen, werden automatisch Indexe für die Primärschlüsselspalte und die Objektschlüsselspalte (`XObjectKey`) erstellt.

Um im Schema Extension einen neuen Index zu erstellen

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.

2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank an.
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Neuen Index**.
6. Auf der Seite **Tabelle erweitern** wählen Sie unter **Tabelle** die Tabelle, für die Sie einen Index erstellen möchten.
7. Auf der Seite **Index erzeugen** legen Sie die Spalten für die Indexdefinition fest.
   a. Klicken Sie die Schaltfläche .

      Es wird ein Dialogfenster geöffnet, in dem Sie die Spalten für den Index festlegen. Auf der rechten Seite des Dialogfensters werden alle Spalten der Tabelle angezeigt. Auf der linken Seite des Dialogfensters werden alle Spalten angezeigt, die zum Index gehören.
   b. Geben Sie im Eingabefeld **Indexname** den Namen des Indexes an.

      Es wird bereits ein Name vorgeschlagen. Diesen können Sie bei Bedarf ändern.
   c. Wählen Sie auf der rechten Seite des Dialogfensters die Spalte, die Sie zum Index hinzufügen möchten.
   d. Fügen Sie die Spalte über die Schaltfläche zum Index hinzu.

      Ändern Sie bei Bedarf die Reihenfolge der Spalten in der Indexdefinition oder entfernen Sie eine Spalte aus dem Index über die entsprechenden Schaltflächen.
   e. Klicken Sie **OK**.

Kundenspezifische Schemaerweiterungen entfernen

HINWEIS: Kundenspezifische Schemaerweiterungen können Sie nur auf Datenbanken mit einer Staging-Ebene **Testumgebung** oder **Entwicklungssystem** entfernen. Ein Transport der Änderungen ist nicht möglich.

Um im Schema Extension kundenspezfische Schemaerweiterungen zu entfernen

1. Starten Sie das Launchpad und melden Sie sich an der One Identity Manager-Datenbank an.
2. Öffnen Sie das Launchpad und wählen Sie den Eintrag **One Identity Manager Schema erweitern**. Das Programm Schema Extension wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.

4. Auf der Seite **Datenbankverbindung** prüfen Sie die Verbindungsdaten zur One Identity Manager-Datenbank .
5. Auf der Seite **Erweiterungsmethode** wählen Sie **Erweiterung entfernen**.
6. Auf der Seite **Erweiterungen entfernen** wählen Sie die kundenspezifische Schemaerweiterung , die Sie entfernen möchten.
7. Um eine kundenspezifische Tabelle zu entfernen, wählen Sie in der Auswahlliste **Tabelle** die Tabelle und aktivieren Sie die Option **Die ganze Tabelle entfernen**.
8. Um kundenspezifische Spalten zu entfernen, wählen Sie in der Auswahlliste **Tabelle** die Tabelle und wählen Sie unter **Zu entfernende Spalten** die Spalten.
9. Klicken Sie **Weiter**.
10. Bestätigen Sie die Sicherheitsabfrage mit **Ja**.
11. Auf der Seite **Systemanpassungen** werden die Schemaänderungen angezeigt. Weitere Informationen finden Sie unter Schemaerweiterungen in die Datenbank übernehmen auf Seite 494.

Verwandte Themen

- Mögliche Fehlermeldungen beim Entfernen kundenspezifischer Schemaerweiterungen auf Seite 492

Mögliche Fehlermeldungen beim Entfernen kundenspezifischer Schemaerweiterungen

Fehlermeldungen, die beim Entfernen von Schemaerweiterungen auftreten, werden im Schema Extension angezeigt und im Systemprotokoll aufgezeichnet.

Tabelle 176: Mögliche Fehlermeldungen

Schritt	Meldung	Lösung
Checking Table empty	Table is not empty	Entfernen Sie alle Objekte aus der Tabelle, bevor Sie die Schemaänderung durchführen.
Checking template references	Column is referenced in template <TableName>.<ColumnName>	Ändern Sie im Designer alle Bildungsregeln und Formatierungsskripte, die sich auf die zu löschende Spalte beziehen. Weitere Informationen finden Sie unter Bildungsregeln bearbeiten auf Seite 83 und Formatierungsskripte erstellen auf Seite 90.
Checking	Column is referenced as parent in	Entfernen Sie die Tabellenbeziehung

Schritt	Meldung	Lösung
referential integrity	RI <RelationID>	aus der Tabelle `QBMRelation`, bevor Sie die Schemaänderung durchführen.
Checking dynamic RI	Column is referenced as parent in <TableName>.<ColumnName>	Die zu löschende Spalte ist als Ziel für dynamisch referenzierte Tabellen definiert. Verwenden Sie den Designer, um diese Referenz aufzuheben. Weitere Informationen finden Sie unter Dynamische Fremdschlüssel auf Seite 113.
Checking indexes	Column is contained in index <TableName>.<IndexName>	Ändern oder löschen Sie den Index, so dass die Spalte nicht mehr referenziert wird, bevor Sie die Schemaänderung durchführen.

Berechtigungen für die Schemaerweiterungen

HINWEIS: Auf Datenbankebene wird für kundenspezifische Schemaerweiterungen die Datenbankrolle **Rolle für Endbenutzer** berechtigt.

Für den ersten Zugriff auf die Schemaerweiterungen über die One Identity Manager-Werkzeuge wählen Sie die Berechtigungsgruppen, die Berechtigungen für die Schemaerweiterungen erhalten. Nachdem Sie alle Änderungen in die Datenbank übernommen haben, können Sie zusätzliche Berechtigungen über den Berechtigungseditor im Programm Designer vergeben. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Um Berechtigungsgruppen im Schema Extension festzulegen

- Wählen Sie auf der Seite **Berechtigungen** über die Auswahllisten
  - mindestens eine kundenspezifische Berechtigungsgruppe, die Leseberechtigungen und Schreibberechtigungen erhält
  - mindestens eine kundenspezifische Berechtigungsgruppe, die nur Leseberechtigungen erhält

Änderungskennzeichen für die Schemaerweiterungen

Weisen Sie die Schemaerweiterungen an ein Änderungskennzeichen zu. Änderungskennzeichen werden im Database Transporter bei der Erstellung eines Kundentransportpaketes als Exportkriterium angeboten.

Um ein Änderungskennzeichen im Schema Extension zuzuweisen

- Wählen Sie auf der Seite **Änderungskennzeichen zuordnen** eine der folgenden Optionen.
  - **Kein Änderungskennzeichen vergeben**
  - **Neues Änderungskennzeichen anlegen**: Erfassen Sie im Eingabefeld **Änderungskennzeichen** die Bezeichnung des Änderungskennzeichens.
  - **Bestehendes Änderungskennzeichen verwenden**: Wählen Sie in der Auswahlliste **Änderungskennzeichen** ein Änderungskennzeichen.

Ausführliche Informationen zum Arbeiten mit Änderungskennzeichen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Schemaerweiterungen in die Datenbank übernehmen

In diesem Schritt übernehmen Sie die Schemaerweiterungen in die One Identity Manager-Datenbank.

Um die Schemaerweiterungen im Schema Extension zu übernehmen

1. Auf der Seite **Systemanpassungen** werden die Schemaänderungen angezeigt.
   a. Wenn die Anweisungen an eine bestehende Datei angefügt werden sollen, aktivieren Sie die Option **Statements an eine bestehende Datei anhängen**.
   b. Wählen Sie über die Schaltfläche **In Datei speichern** den Ablagepfad und geben Sie den Dateinamen an. Die Anweisungen werden als XML-Datei gespeichert.
2. Klicken Sie **Weiter**.
3. Bestätigen Sie die Sicherheitsabfrage mit **Ja**.

   Die Schemaerweiterungen in der Datenbank angelegt und die notwendigen Erweiterungen im One Identity Manager-Systemdatenmodell vorgenommen. Diese Aktion kann einige Zeit in Anspruch nehmen.

4. Auf der Seite **Systemverarbeitung** werden die aktuellen Berechnungsaufträge für den DBQueue Prozessor angezeigt. Nachdem die Verarbeitung der Berechnungsaufträge abgeschlossen ist, klicken Sie **Weiter**.
5. Auf der Seite **Kompilierung** klicken Sie **Weiter**.

   Der Kompiliervorgang kann einige Zeit in Anspruch nehmen.
6. Nach Beenden der Kompilierung, klicken Sie **Weiter**.
7. Auf der letzten Seite starten Sie weitere Schemaerweiterungen oder klicken Sie **Fertig**, um das Programm zu beenden.

Nach Abschluss der Schemaerweiterungen können Sie mit den One Identity Manager-Werkzeugen auf die Erweiterungen zugreifen und weitere Änderungen vornehmen.

Verwandte Themen

- Empfehlungen für die weitere Konfiguration kundenspezifischer Schemaerweiterungen auf Seite 495
- Verwaltung von kundenspezifischen Datenbankobjekten innerhalb der Datenbank auf Seite 498

Empfehlungen für die weitere Konfiguration kundenspezifischer Schemaerweiterungen

Nachdem Sie das One Identity Manager Schema um kundenspezifische Tabellen oder Spalten erweitert haben, sind weitere Schritte erforderlich, um die Erweiterungen in der Benutzeroberfläche des Manager anzuzeigen.

Allgemeine Empfehlungen

- Bearbeiten Sie die Objektschicht mit den One Identity Manager-Werkzeugen. So ist sichergestellt, dass die Daten im erwarteten Format erzeugt werden.
- Bearbeiten Sie die Objektschicht immer in der Standardsprache einer One Identity Manager-Installation **English - United States [en-US]**. Wählen Sie dazu in den One Identity Manager-Werkzeugen die Anmeldesprache **Englisch (USA)**.
- Im Designer finden Sie verschiedene Konsistenzprüfungen. Führen Sie nach einer Schemerweiterung sowie nach Änderungen der Tabellendefinitionen und Spaltendefinitionen die Konsistenzprüfungen aus und verwenden Sie die Reparaturmethoden. Ausführliche Informationen zum Prüfen der Datenkonsistenz finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Empfehlungen für Tabellendefinitionen, Spaltendefinitionen und Tabellenbeziehungen

Zu den Eigenschaften gehören beispielsweise Anzeigenamen, Beschreibungen, Anzeigemuster für Tabellen und Spalten sowie Bildungsregeln, Formatierungen, Pflichtfelddefinitionen. Weitere Informationen finden Sie unter Grundlagen zum One Identity Manager Schema auf Seite 47.

- Verwenden Sie den Schemaeditor des Designer um die Tabellendefinitionen und Spaltendefinitionen weiter zu bearbeiten.
- Legen Sie im Designer den Nutzungstyp der Tabellen fest. Der Nutzungstyp der Tabelle dient als Grundlage für Berichte und die Auswahl von Aufträgen für tägliche Wartungsarbeiten.
- Bearbeiten Sie im Designer für die Tabellen den Anzeigenamen und das Symbol. Diese Eigenschaften werden verwendet, wenn Sie Objektdefinitionen für die Tabelle erstellen.
- Definieren Sie im Designer ein Anzeigemuster für die Darstellung der Tabelleneinträge beispielsweise in der Ergebnisliste der One Identity Manager-Werkzeuge oder in Berichten.

  > HINWEIS: Für M:N-Tabellen müssen Sie kein Anzeigemuster eintragen. Das Anzeigemuster wird in diesem Fall von der `viDB.DLL` aus den Fremdschlüsseln gebildet.

- Gibt es für eine Tabelle eine Kombination von Spalten, die eindeutig sein müssen, definieren Sie im Designer mehrspaltige Eindeutigkeiten.
- Ordnen Sie die Tabellen im Designer in der Schemaübersicht des Schemaeditor an. Erfolgt dies nicht, so werden in der Schemaübersicht alle neuen Tabellen in der oberen linken Ecke des Moduls angezeigt. Die farbige Hinterlegung des Moduls wird bei einem erneuten Laden der Objekte automatisch angepasst.
- Erfassen Sie im Designer für jede Spalte den Anzeigenamen und eine Bemerkung für die Anzeige in den One Identity Manager-Werkzeugen.
- Spalten, die Kennwörter enthalten, kennzeichnen Sie im Designer mit der Option **verschlüsselt**.
- Der Syntaxtyp der Spaltendefinition wird genutzt, um in den One Identity Manager-Werkzeugen die entsprechende Syntaxhervorhebung oder Eingabehilfen zur Verfügung zu stellen.
- Spalten, die einen Benutzerkontennamen enthalten, kennzeichnen Sie im Designer in der Eigenschaft **Tabellenübergreifende Suche** mit dem Wert **Zentrales Benutzerkonto**.
- Spalten, die eine E-Mail-Adresse enthalten, kennzeichnen Sie im Designer in der Eigenschaft **Tabellenübergreifende Suche** mit dem Wert **E-Mail-Adresse**.
- Für Tabellen, die bei der Ermittlung von Personen für Benutzerkonten oder E-Mail-Adressen berücksichtigt werden sollen, erfassen Sie im Designer in der Eigenschaft **Pfad zum Person-Objekt für tabellenübergreifende Suche** den Pfad zum `Person`-Objekt. Die ermittelten Daten werden in `QBMSplittedLookup.SplittedElement`

abgebildet. Für Tabellen, für die keine Ermittlung einer Person erfolgen soll, erfassen Sie den Wert **no**.

Empfehlungen für Berechtigungen

Bei der Schemaerweiterung mit dem Programm Schema Extension vergeben Sie bereits Berechtigungen für Berechtigungsgruppen. Sie können die Berechtigungen im Berechtigungseditor des Designer weiter bearbeiten und zusätzliche Berechtigungsgruppen mit dem Benutzer-& Berechtigungsgruppeneditor erstellen. Berechtigungsgruppen können mit Anwendungsrollen verknüpft werden. Die Benutzer werden an Anwendungsrollen zugewiesen und erhalten somit die Berechtigungen, die Sie benötigen. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Empfehlungen für Objektdefinitionen

Die Darstellung der Daten in den Benutzeroberflächen erfolgt über Objekte. Eine allgemein gültige Objektdefinition ohne einschränkende Auswahlkriterien wird bereits mit dem Programm Schema Extension erzeugt. Zusätzlich können Sie weitere einschränkende Objektdefinitionen erstellen. Die Objektdefinitionen erstellen Sie im Designer. Weitere Informationen finden Sie unter Objektdefinitionen für die Benutzeroberfläche auf Seite 119.

Empfehlungen für die Menüführung

Damit die Daten im Manager angezeigt werden, erweitern Sie die Menüführung. Verwenden Sie den Oberflächeneditor des Designer, um Menüeinträge für die Navigation und Ergebnislisten zu erzeugen. Ausführliche Informationen finden Sie unter Menüführung der Benutzeroberfläche auf Seite 124 und Empfehlungen zur Bearbeitung der Menüführung auf Seite 127.

Empfehlungen für Formulare für die Benutzeroberfläche

Für die Anzeige und die Bearbeitung im Manager erstellen oder erweitern Sie die Formulare. Ausführliche Informationen finden Sie unter Empfehlungen für die Bearbeitung von Formularen auf Seite 153, Bearbeiten von Oberflächenformularen auf Seite 155, Formulare für kundenspezifische Erweiterungen auf Seite 163 und Bearbeiten von Überblickformularen auf Seite 179.

Empfehlungen für Methodendefinitionen

Wollen Sie im Manager bestimmte Aufgaben für die Objekte anbieten, erstellen Sie im Designer Methodendefinitionen. Weitere Informationen finden Sie unter Methodendefinitionen für die Benutzeroberfläche auf Seite 212.

- Erstellen Sie bei Bedarf neue Methodendefinitionen.
- Methodendefinitionen werden für Objektdefinitionen erstellt, so dass abhängig vom ausgewählten Objekt unterschiedliche Methoden in der Benutzeroberfläche

angeboten werden. Erstellen Sie bei Bedarf weitere Objektdefinitionen.

- Weisen Sie die Methodendefinitionen an die Berechtigungsgruppen für die nicht-rollenbasierte Anmeldung und die rollenbasierte Anmeldung zu.
- Weisen Sie bei Bedarf der Methodendefinition eine Programmfunktion zu. Ausführliche Informationen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Empfehlungen für Auswertungen

Für Auswertung von Daten erstellen Sie Statistikdefinitionen und Berichte und binden Sie diese in die Benutzeroberfläche ein. Weitere Informationen finden Sie unter Statistiken im One Identity Manager auf Seite 190 und Berichte im One Identity Manager auf Seite 437.

Empfehlungen für die Lokalisierung der Texte

Für die sprachabhängige Anzeige von Texten im Manager wie beispielsweise Spaltenbezeichnungen, Kommentaren, Menüeinträgen, Formularbezeichnungen übersetzen Sie die Texte mit dem Wörterbucheditor des Designer. Weitere Informationen finden Sie unter Sprachabhängige Abbildung von Informationen auf Seite 233.

Verwaltung von kundenspezifischen Datenbankobjekten innerhalb der Datenbank

Für die Erstellungen von Transportpaketen mit dem Programm Database Transporter sowie zur Erstellung von Berichten über die Systemkonfiguration werden die Informationen über Datenbankobjekte wie kundenspezifische Datenbanktabellen und Datenbankspalten, Datenbankprozeduren, Funktionen, Trigger, Indizes oder Viewdefinitionen in der Datenbank abgelegt. Der DBQueue Prozessor prüft und aktualisiert diese Informationen.

> HINWEIS: Eine manuelle Bearbeitung der Daten ist in der Regel nicht erforderlich, lediglich die Bemerkungen können Sie für die Nutzung in Berichten anpassen.

Um Datenbankobjekte anzupassen

1. Wählen Sie im Designer die Kategorie **Basisdaten > Erweitert > SQL Anpassungen**.
2. Wählen Sie das Datenbankobjekt.
3. Bearbeiten Sie die **Bemerkungen**.

Tabelle 177: Eigenschaften für Datenbankobjekte

Eigenschaft	Beschreibung
Bearbeitungsstatus	Der Bearbeitungsstatus wird bei der Erstellung von Kundenkonfigurationspaketen genutzt.
Bemerkungen	Zusätzliche Bemerkungen, zum Beispiel für die Nutzung in Berichten zur Systemkonfiguration.
Bezeichnung	Name des Datenbankobjektes.
Geändert	Gibt an, ob eine Änderung am Datenbankobjekt erfolgt ist.
Reihenfolge	Sortierreihenfolge.
Typ	Typ des Datenbankobjektes, zum Beispiel Prozedur, Funktion, Trigger, Index, View, Kundentabelle, Kundenspalte.

Ausführliche Informationen zur Erstellung von Transportpaketen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Webservice Integration

Der One Identity Manager bietet die Möglichkeit zur Integration von Webservices. Webservices können Sie beispielsweise nutzen, um Daten in Anwendungen zu schreiben, die nicht als Standard-Zielsystem an den One Identity Manager angebunden werden können.

Daten für externe Anwendungen können aus beliebigen Tabellen des One Identity Manager Schemas stammen. Sie können beispielsweise als kundendefiniertes Zielsystem abgebildet sein.

Beispiel:

Die Stammdaten einer Telefonanlage sollen aus den Personenstammdaten im One Identity Manager ermittelt werden. Die Telefonanlage wird im One Identity Manager als kundendefiniertes Zielsystem abgebildet. Ein Anschluss in der Telefonanlage entspricht einem Benutzerkonto im One Identity Manager.

Sobald ein neuer Mitarbeiter im One Identity Manager angelegt wird, soll in der Telefonanlage ein neuer Anschluss bereitgestellt werden. Per Kontendefinition wird ein neues Benutzerkonto angelegt. Über einen Webservice werden die Stammdaten des Benutzerkontos an die Telefonanlage übermittelt. Dort wird ein neuer Teilnehmer angelegt und eine Telefonnummer generiert. Diese Telefonnummer übergibt der Webservice als Rückgabewert an den One Identity Manager. Die Telefonnummer soll an die Personenstammdaten übernommen werden.

So können Sie vorgehen

1. Richten Sie im One Identity Manager ein kundendefiniertes Zielsystem ein.
    - Für die Eigenschaft **Synchronisiert durch** wählen Sie den Wert **Synchronisation per Skript**.
2. Richten Sie einen Server für die Provisionierung der Daten ein.
    - Weisen Sie diesen Server dem kundendefinierten Zielsystem als Synchronisationsserver zu.

3. Richten Sie eine Kontendefinition ein, um Benutzerkonten in diesem Zielsystem automatisch zu verwalten.
4. Erfassen Sie die benötigten IT Betriebsdaten.
5. Binden Sie den Webservice an den One Identity Manager an. Nutzen Sie dabei den generischen Webserviceaufruf.

   Der Webservice-Integrationsassistent unterstützt Sie beim Erstellen der Skripte für die Provisionierung von Daten bei den Standardereignissen **Insert**, **Update** und **Delete**. Die Prozesse zur Provisionierung werden standardmäßig durch den One Identity Manager bereitgestellt.
6. Um den Rückgabewert des Webservice zu verarbeiten, erstellen Sie zusätzliche Skripte und Prozesse.

   > TIPP: Beim Einfügen, Ändern oder Löschen von Containern, Benutzerkonten und Gruppen in einem kundendefinierten Zielsystem werden die Rückgabewerte standardmäßig als Objekt GUID in der Datenbank gespeichert.
   >
   > Erstellen Sie einen Prozess, um die Telefonnummer aus der Objekt GUID an die Personenstammdaten zu übernehmen.

Ausführliche Informationen zum Einrichten eines kundendefinierten Zielsystems, zu Kontendefinitionen, IT Betriebsdaten und zum Einrichten eines Servers finden Sie im *One Identity Manager Administrationshandbuch für die Anbindung kundendefinierter Zielsysteme*.

Detaillierte Informationen zum Thema

- Generischer Webserviceaufruf auf Seite 502
- Webservice-Lösung mit dem Webservice-Integrationsassistenten erstellen auf Seite 504
- Webservice-Lösung ändern auf Seite 508
- Abbildung von Prozessen im One Identity Manager auf Seite 244
- Skripte im One Identity Manager auf Seite 375

Anbinden eines Webservices

Für die Integration eines Webservices in den One Identity Manager erstellen Sie kundenspezifische Skripte. Dabei unterstützt Sie ein Assistent. Der Webservice-Integrationsassistent ermittelt alle Methoden, die der Webservice nutzt, und erstellt Skripte, um die benötigten Methoden aufzurufen. Die Daten aus dem One Identity Manager werden als Parameter an die Methoden übergeben. Welche Operationen in der externen Anwendung ausgeführt werden können, wird durch die im Webservice definierten Methoden bestimmt. Der Assistent erzeugt neue Einträge in den Tabellen `DialogWebService` und `DialogScript`.

Der Webservice-Integrationsassistent unterscheidet verschiedene Typen des Methodenaufrufs. Jeder Typ unterstützt in unterschiedlichem Maße die Definition des Methodenaufrufs und damit die Erzeugung der Skripte.

Generischer Webserviceaufruf

Den generischen Webserviceaufruf nutzen Sie, wenn Daten aus einem kundendefinierten Zielsystem über einen Webservice in eine externe Anwendung publiziert werden sollen. Der Webservice-Integrationsassistent fragt alle benötigten Parameter ab und generiert daraus den Skriptcode.

Voraussetzungen

Die Daten der externen Anwendung sind im One Identity Manager als kundendefiniertes Zielsystem abgebildet.

- Ein kundendefiniertes Zielsystem (Tabelle `UNSRootB`) ist angelegt. Die Eigenschaft **Synchronisiert durch** hat den Wert **Synchronisation per Skript**.
- Ein Server für die Provisionierung der Daten ist eingerichtet und als Synchronisationsserver am kundendefinierten Zielsystem hinterlegt.

Ausführliche Informationen zum Einrichten der Skript-gesteuerten Provisionierung finden Sie im*One Identity Manager Administrationshandbuch für das Zielsystem-Basismodul*.

Standardprozesse

Zur Provisionierung von Daten aus einem kundendefinierten Zielsystem an einen Webservice stellt der One Identity Manager Standardprozesse bereit.

Um diese Prozesse zu nutzen, müssen die mit dem Webservice-Integrationsassistenten erstellten Skripte folgender Namenskonvention entsprechen:

`<Kundenpräfix>_<Tabelle>_<Ident_UNSRoot>_<Ereignis>`.

> WICHTIG: Enthält ihr Zielsystem einen Bindestrich (-) im Namen, müssen Sie diesen in den Skriptfunktionen im Bestandteil `<Ident_UNSRoot>` entfernen. Anderenfalls können Fehler in der Skriptverarbeitung auftreten.

Einige dieser Prozesse verarbeiten auch die Rückgabewerte des Webservice.

Tabelle 178: Standardprozesse für die Synchronisation per Skript

Objekt im kundendefinierten Zielsystem (Tabelle)	Prozess	Speicherung der Rückgabewerte
Container (`UNSContainerB`)	VI_UnsContainer_Generic	UNSContainerB.ObjectGUID
Benutzerkonten	VI_UnsAccountB_Generic	UNSAccountB.ObjectGUID

Objekt im kundendefinierten Zielsystem (Tabelle)	Prozess	Speicherung der Rückgabewerte
(UNSAccountB)		
	VI_UnsAccountInGroup_Generic_Del	-
	VI_UnsAccountInGroup_Generic_Add	-
Gruppen (UNSGroupB)	VI_UnsGroup_Generic	UNSGroupB.ObjectGUID
	VI_UNSGroupBInUNSGroupB_Generic_Del	-
	VI_UnsGroupBInUNSGroupB_Generic_Add	-
Berechtigungselemente (UNSItemB)	VI_UnsItem_Generic	-
	VI_UnsGroupHasItem_Generic_Del	-
	VI_UnsGroupHasItem_Generic_Add	-
	VI_UnsAccountHasItem_Generic_Del	-
	VI_UnsAccountHasItem_Generic_Add	-

Direkter Webserviceaufruf

Der Webservice-Integrationsassistent ermittelt alle Parameter, die in der Methode definiert sind, und generiert daraus den Skriptcode. Die Parameter werden im Funktionsaufruf übergeben. Änderungen an den Parametern sind möglich.

Um die Skripte auszuführen

- Erstellen Sie kundenspezifische Prozesse und übergeben Sie die Skripte und Parameter an die Prozessschritte.

Verwandte Themen

- Abbildung von Prozessen im One Identity Manager auf Seite 244

Selbstdefinierter Webserviceaufruf

Der Webservice-Integrationsassistent ermittelt alle Parameter, die in der Methode definiert sind. Die Übergabe der Parameter muss komplett selbst definiert werden.

Um die Skripte auszuführen

- Erstellen Sie kundenspezifische Prozesse und übergeben Sie die Skripte und Parameter an die Prozessschritte.

Verwandte Themen

- Abbildung von Prozessen im One Identity Manager auf Seite 244

Webservice-Lösung mit dem Webservice-Integrationsassistenten erstellen

Voraussetzung

- Um einen Webservice mit dem Diensttyp **WCF** anbinden zu können, muss im Installationsverzeichnis des One Identity Manager die Datei `SvcUtil.exe` vorhanden sein.

  Informieren Sie sich bei Microsoft, wie Sie diese Datei erwerben können.
- Um einen Webservice mit dem Diensttyp **SOAP** anbinden zu können, muss die Datei `WSDL.exe` auf dem Server, der die Provisionierung ausführt, vorhanden sein.

  Informieren Sie sich bei Microsoft, wie Sie diese Datei erwerben können.

Um einen neuen Webservice zu integrieren

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Webservices**.
2. Wählen Sie die Aufgabe **Neuen Webservice integrieren**.

   Der Webservice-Integrationsassistent wird gestartet.
3. Auf der Startseite klicken Sie **Weiter**.
4. Auf der Seite **Neuen Webservice integrieren** erfassen Sie die Zugangsdaten und allgemeinen Eigenschaften des Webservice.

   Tabelle 179: Allgemeine Eigenschaften eines Webservice

Eigenschaft	Beschreibung
Name des Webservice	Anzeigename für den Webservice im One Identity Manager.

Eigenschaft	Beschreibung
Beschreibung	Freitextfeld für zusätzliche Erläuterungen.
.NET Namespace für den Proxycode	Eindeutige Bezeichnung des .NET Namespace.
URL des Webservice	URL, unter welcher der Webservice ausgeführt wird.
URL der WSDL-Datei	URL, unter der die `WSDL.exe` des Webservices erreicht werden kann. Wenn die `WSDL.exe` nicht öffentlich verfügbar ist, kann sie auch lokal gespeichert werden. HINWEIS: Wenn der Webservice-Betreiber die WSDL-Datei ändert, führen Sie den Webservice-Integrationsassistenten erneut aus, um die Änderungen nutzen zu können.
Diensttyp	Typ des Webservices.
Gesperrt	Gibt an, ob der Webservice genutzt werden kann.
Benutzername	Benutzername für die Anmeldung am Webservice.
Domäne des Benutzers	Domäne des Benutzers.
Kennwort des Benutzers und Kennwortbestätigung	Kennwort für die Anmeldung am Webservice.
Proxycodegenerator	Pfad und Dateiname zum Proxycodegenerator. • Wenn der Diensttyp **WCF** ausgewählt ist, Pfad zur Datei `SvcUtil.exe`. • Wenn der Diensttyp **SOAP** ausgewählt ist, Pfad zur Datei `WSDL.exe`.

Tabelle 180: Erweiterte Eigenschaften eines Webservices

Eigenschaft	Beschreibung
URL des Proxyservers	URL des Proxyservers, wenn die Kommunikation über einen Proxyserver geleitet wird.
Benutzername des Proxyservers	Benutzername für die Anmeldung am Proxyserver.
Domäne des Proxy-servers	Domäne des Proxyservers.
Kennwort des Proxy-servers und Kennwort-	Kennwort und Kennwortbestätigung für die Anmeldung am Proxyserver.

Eigenschaft	Beschreibung
bestätigung	
Timeout für WSDL.exe	Timeout für den Zugriff auf die WSDL-Datei.

Benutzerdefinierte Kommandozeile	Kommandozeile für den Aufruf des Proxycodegenerators. Die Kommandozeile kann um zusätzliche Parameter erweitert werden, falls diese benötigt werden.
	Standardkommando:
	`/nologo /language:VB "/namespace:%Namespace%" "/out:{0}" %WsdlUrl%`
	Beispiel:
	`/nologo /language:VB "/namespace:EnricoHolidayWebservice" "/out:{0}" http://kayaposoft.com/enrico/ws/v1.0/index.php?wsdl`

a. Klicken Sie **Überprüfen**.

   Der Zugriff auf den Webservice wird getestet.

b. Wenn der Test erfolgreich war, klicken Sie **Weiter**.

5. Auf der Seite **Proxycode erzeugen** wird der generierte Proxycode angezeigt.

   Der Proxycode enthält alle Webservicemethoden, die in der WSDL-Datei definiert sind, und stellt sie der One Identity Manager Skriptkomponente zur Verfügung.

6. Auf der Seite **Serviceklasse wählen** wählen Sie die Serviceklasse, die Sie nutzen möchten.

   Werden von dem Webservice mehrere Serviceklassen bereitgestellt, muss für jede Serviceklasse, die genutzt werden soll, eine separate Webserviceanbindung erstellt werden.

7. Auf der Seite **Methodenaufrufe wählen** wählen Sie die Webservicemethoden aus, die Sie im One Identity Manager nutzen möchten. Für jede gewählte Methode wird im nächsten Schritt ein Skript erzeugt.

8. Auf der Seite **Webserviceaufruf generieren** wird ein Skript erzeugt, welches die ausgewählte Methode aufruft. Erfassen Sie alle dafür benötigten Parameter und Eigenschaften.

- Klicken Sie , um den Typ des Methodenaufrufs festzulegen.

Tabelle 181: Typ des Methodenaufrufs

Typ	Beschreibung
Selbstdefinierter Webserviceaufruf	Weitere Informationen finden Sie unter Selbstdefinierter Webserviceaufruf auf Seite 504.
Direkter Webserviceaufruf	Weitere Informationen finden Sie unter Direkter Webserviceaufruf auf Seite 503.
Generischer Webserviceaufruf	Weitere Informationen finden Sie unter Generischer Webserviceaufruf auf Seite 502.

Tabelle 182: Skripteigenschaften

Eigenschaft	Beschreibung
Skriptname	Name des Skripts. Kennzeichnen Sie kundenspezifische Skripte mit dem Präfix **CCC_**. Skriptnamen für den generischen Webserviceaufruf müssen nach folgendem Muster gebildet werden: `<Kundenpräfix>_<Tabelle>_<Zielsystem>_<Ereignis>` Wählen Sie die Tabelle, das Zielsystem und das Ereignis, für die das Skript erstellt werden soll. Parameter, Wertart und Datentabelle werden automatisch aus der gewählten Tabelle ermittelt.
Parameter	Name des Parameters.
Wertart	Datentyp des Parameters.
Datentabelle	Datenbanktabelle, die die Daten enthält, die an den Webservice übergeben werden sollen.
Rückgabetyp	Datentyp des Parameters, der den Rückgabewert enthält.

Tabelle 183: Datenübergabe

Eigenschaft	Beschreibung
Parameter	Parameter, der an den Webservice übergeben wird.
Wertart	Datentyp des Parameters.
Zuordnung von	Parameter aus den definierten Skripteigenschaften. Öffnen Sie die Auswahlliste und ordnen Sie den zugehörigen Parameter zu. Wenn erforderlich, wählen Sie die Spalte aus der Datentabelle, welche die Übergabewerte enthält.

Im Bereich **Skriptcode** sehen Sie das generierte Skript. Sie können den erweiterten Bearbeitungsmodus nutzen, um das Skript zu bearbeiten.

> TIPP: Das Skript ruft die Funktion `VID_GetWcfWebService` auf, welche wiederum die Funktionen `GetWcfBinding` und `GetWcfEndpointAddress` nutzt. Diese drei Funktionen sind überschreibbar.

9. Um den Webservice-Integrationsassistenten zu beenden, klicken Sie **Fertig**.
10. Speichern Sie die Änderungen.
11. Kompilieren Sie die Datenbank.

Verwandte Themen

- Skripte im One Identity Manager auf Seite 375
- Überschreiben von Skripten auf Seite 404

Webservice-Lösung ändern

Eine bestehende Webservice-Lösung können Sie jederzeit ändern oder erweitern. Dabei wird ein bestehendes Skript überschrieben. Neue Skripte werden hinzugefügt.

Um eine Webservice-Lösung zu erweitern

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Webservices**.
2. Wählen Sie im Listeneditor den Webservice.
3. Wählen Sie die Aufgabe **Webservice Aufrufe hinzufügen**.

   Der Webservice-Integrationsassistent wird gestartet.
4. Folgen Sie den Anweisungen des Assistenten.
5. Speichern Sie die Änderungen.
6. Kompilieren Sie die Datenbank.

Um eine Webservice-Lösung zu bearbeiten

1. Wählen Sie im Designer die Kategorie **Basisdaten > Allgemein > Webservices**.
2. Wählen Sie im Listeneditor den Webservice.
3. Wählen Sie die Aufgabe **Webservice bearbeiten**.

   Der Webservice-Integrationsassistent wird gestartet.
4. Folgen Sie den Anweisungen des Assistenten.

   Auf der Seite **Proxycode erzeugen** können Sie den generierten Proxycode bearbeiten.

- Um die Änderungen zu überprüfen, aktivieren Sie **Mit vorhandener Version vergleichen**.

5. Speichern Sie die Änderungen.
6. Kompilieren Sie die Datenbank.

HINWEIS: Wenn der Webservice-Integrationsassistent erneut ausgeführt wird, wird der Proxycode neu generiert. Alle manuellen Änderungen am Proxycode werden dabei überschrieben.

Detaillierte Informationen zum Thema

- Webservice-Lösung mit dem Webservice-Integrationsassistenten erstellen auf Seite 504

Webservice-Lösung löschen

Um eine Webservice-Lösung aus der Datenbank zu entfernen

1. Löschen Sie den Webservice.
2. Löschen Sie alle zugehörigen kundenspezifischen Skripte.
3. Ermitteln Sie alle weiteren kundenspezifischen Elemente Ihrer Webservice-Lösung und löschen Sie diese.
4. Kompilieren Sie die Datenbank.

17

One Identity Manager als SCIM 2.0 Serviceprovider

One Identity Manager stellt eine Schnittstelle entsprechend der System for Cross-domain Identity Management (SCIM) 2.0 zur Verfügung. Die Schnittstelle ermöglicht den Import und den Export von One Identity Manager-Objekten über einen SCIM Client. Die Schnittstelle kann beispielsweise eingesetzt werden für:

- den Import von Daten zu Identitäten, Abteilungen, Kostenstellen und Standorten aus einem HR-System
- den Import von Daten zu Benutzerkonten und Gruppen aus einem Cloud-System
- den Abgleich von kundendefinierten Zielsystemen, die einen SCIM V2.0 Client haben
- die Bereitstellung von Informationen zu Identitäten über eine standardisierte Schnittstelle für die Weiterverarbeitung in Fremdsystemen

Der SCIM 2.0 Serviceprovider für One Identity Manager wird als Plugin für den API Server bereitgestellt und kann bei der Installation des API Servers ausgewählt werden. Ein weitere Konfiguration des SCIM Plugin selbst ist nicht notwendig. Die erfolgreiche Installation und Aktivierung des SCIM Plugins kann im Administrationsportal des API Server überprüft werden. Ausführliche Informationen zur Installation eines API Servers finden Sie im *One Identity Manager Installationshandbuch*.

Detaillierte Informationen zum Thema

- Endpunkte und Basis-URL auf Seite 511
- SCIM Plugin Features auf Seite 511
- Authentifizierung von SCIM Clients auf Seite 512
- Authentifizierung des SCIM Plugins am One Identity Manager auf Seite 512
- Besonderheiten bei der Generierung des SCIM Schemas auf Seite 513
- Anfragen an das SCIM Plugin auf Seite 514

Endpunkte und Basis-URL

HINWEIS: Die an der Schnittstelle bereitgestellten Provider und Endpunkte sind fest vorgegeben und können nicht konfiguriert werden.

Die SCIM 2.0 Schnittstelle ist im API Server Installationsverzeichnis unter dem festen Unterverzeichnis `/ApiServer/scim/v2` erreichbar. Die Basis-URL wird folgendermaßen gebildet:

```
<http | https>://[< subdomain | servername>.]<domainname>/ApiServer/scim/v2
```

Es gibt weitere fest definierte Endpunkte. Die Endpunkte werden ohne weitere URL-Parameter aufgerufen.

- /ServiceproviderConfig

  Der Endpunkt gibt Auskunft über die implementierten Möglichkeiten des Serviceprovider wie beispielsweise die angebotenen Authentifizierungsarten. Der Endpunkt kann ohne Authentifizierung aufgerufen werden.

- /Schemas

  Die Rückgabestruktur des Endpunktes definiert alle Datenobjekte und ihre Eigenschaften, die vom Provider unterstützt werden.

- /ResourceTypes

  Beim Aufruf des Endpunktes werden die Metadaten zu den über den Endpunkt `/Schemas` publizierten Datenobjekten geliefert. Diese werden mit einem Endpunkt basierend auf der Basis-URL verknüpft, unter dem der jeweilige Provider für einen bestimmten Datenobjekttyp erreichbar ist.

Verwandte Themen

- SCIM Plugin Features auf Seite 511
- Authentifizierung von SCIM Clients auf Seite 512
- Besonderheiten bei der Generierung des SCIM Schemas auf Seite 513
- Anfragen an das SCIM Plugin auf Seite 514

SCIM Plugin Features

Die Rückgabestruktur des Endpunktes `/ServiceproviderConfig` definiert unter anderem die Features, die vom SCIM Plugin unterstützt werden.

- `patch`: Bei der Übertragung von Änderungen werden nur Einzeloperationen auf Eigenschaften übertragen, nicht das komplette zu ändernde Objekt.
- `filter`: Die Filterung von Objekten direkt bei der Anfrage oder in `patch`-Operationen wird unterstützt.

Authentifizierung von SCIM Clients

In der über den Endpunkt `/ServiceproviderConfig` abgerufenen Struktur ist eine `authenticationSchemes`-Liste enthalten. Die Liste publiziert die möglichen Authentifizierungsmethoden und Autorisierungsmethoden, mit denen das SCIM Plugin genutzt werden kann.

Unterstützt werden:

- HTTP Basic access authentication
- NTLM Authentication and Kerberos
- OAuth 2.0 client authorization

Verwandte Themen

- Authentifizierung von SCIM Clients auf Seite 512

Authentifizierung des SCIM Plugins am One Identity Manager

Zum Zugriff auf die One Identity Manager-Datenbank muss sich das SCIM Plugin authentifizieren. Die Authentifizierung erfolgt über die One Identity Manager Authentifizierungsmodule. Ausführliche Informationen erhalten Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

Die Authentifizierungsmodule werden in der folgenden Reihenfolge überprüft und das erste erfolgreiche Authentifizierungsmodul zur Anmeldung verwendet. Stellen Sie sicher, dass das mindestens ein Authentifizierungsmodul aktiviert und konfiguriert ist.

1. Person (`Person`)
2. Active Directory Benutzerkonto (`ADSAccount`)
3. Person (rollenbasiert) (`RoleBasedPerson`)
4. Active Directory Benutzerkonto (rollenbasiert) (`RoleBasedADSAccount`)
5. HTTP Header (rollenbasiert) (RoleBasedHTTPHeader)
6. HTTP Header (HTTPHeader)
7. OAuth 2.0/OpenID Connect (rollenbasiert) (`OAuthRoleBased`)

Verwandte Themen

- Authentifizierung von SCIM Clients auf Seite 512

Besonderheiten bei der Generierung des SCIM Schemas

Das am Endpunkt `/Schemas` exportierte SCIM 2.0 Schema wird aus dem One Identity Manager Schema generiert. Die zu berücksichtigenden Tabellendefinitionen und zu publizierenden M:N-Abbildungen sind vorgegeben. Es entsteht pro Tabelle eine Datenobjektbeschreibung mit einfachen und komplexen Eigenschaften.

Spalten einer Tabelle

Die Spalten einer Tabelle werden auf einfache Eigenschaften integraler Typen abgebildet.

Fremdschlüssel-Beziehungen

Die Fremdschlüssel-Beziehungen einer Tabelle werden nur dann in das Schema aufgenommen, wenn die Zieltabelle der Referenz ebenfalls Bestandteil des Schemas ist. In diesem Fall wird eine komplexe Eigenschaft mit dem Spaltennamen des Fremdschlüssels publiziert. Diese komplexe Eigenschaft besitzt die Eigenschaften `value`, `$ref` und `displayName`.

Die komplexe Eigenschaft wird im Schema mit dem Attribut „`returned“ : „request“` gekennzeichnet. Um diese Daten lesen zu können, muss die Eigenschaft vom SCIM Client explizit über den URL–Parameter `attributes` angefordert werden.

Beispiel:

```
https://<servername>.<domainname>/ApiServer/scim/v2/Locality/0294ce3c-8286-
4641-bc13-9bcd4a2fd714?attributes=cn,City,UID_PersonHead
```

M:N-Tabellen

M:N-Tabellen werden unter der komplexen Eigenschaft `members` im Schema publiziert. Dies gilt auch, wenn mehrere M:N-Tabellen zu berücksichtigen sind. Diese komplexe Eigenschaft definiert ein Array von Unterelementen, die die Eigenschaften `value`, `type`, `$ref` und `display` besitzen.

Die komplexe Eigenschaft `members` wird im Schema mit dem Attribut „`returned“ : „request“` gekennzeichnet. Um diese Daten lesen zu können, muss die Eigenschaft vom SCIM Client explizit über den URL–Parameter `attributes` angefordert werden.

Beispiel:

```
http://<servername>.<domainname>/ApiServer/scim/v2/UNSGroupB/94bbe614-0a6e-
4659-8fe9-20da94d967c8?attributes=cn,members
```

Bei mehreren zusammengefassten M:N-Tabellen erfolgt die Unterscheidung, aus welcher Tabelle das jeweilige Element stammt, anhand des Wertes in der Eigenschaft `type`. Bei schreibenden Zugriffen auf die Eigenschaft ist darauf zu achten, dass der Wert in der Eigenschaft `type` mit übergeben wird. Die als korrekt akzeptierten Werte sind im Schema am jeweiligen `type`-Subattribut als Liste in `canonicalValues` definiert.

Ist der Wert für `type` für den SCIM Client nicht ermittelbar, so kann dieser leer gelassen werden und nicht mit im `PUT`- oder `PATCH`-Request übertragen werden. Das SCIM Plugin versucht den korrekten `type` zu ermitteln. Dabei wird das Element anhand seiner in der Eigenschaft `value` übergebenen ID in allen One Identity Manager Tabellen gesucht, die Teil der `members`-Definition sind. Wird das Objekt dabei gefunden, kann die Operation ausgeführt werden.

Anfragen an das SCIM Plugin

Anfragen an die Basis-URL

Die SCIM 2.0 Spezifikation sieht optionale Anfragen an die Basis-URL des SCIM Serviceproviders vor. Diese Anfragen können optional einen Filterausdruck enthalten. Dies dient vorwiegend der Suche nach Objekten, wenn deren Endpunkt nicht genau bekannt ist und somit der Endpunkt-übergreifenden Suche.

Das SCIM Plugin unterstützt diese Anfragen. Im Filter sind nur logische `OR`-Verknüpfungen und die Vergleichsoperatoren `eq`, `sw`, `ew` sowie `co` zugelassen, die sich auf die Metainformation `Resourcetype` beziehen müssen.

Beispiel:

```
https://<servername>.<domainname>/ApiServer/scim/v2?filter=
(meta.Resourcetype eq “Locality”) or (meta.Resourcetype sw “D”)
```

Das Ergebnis kann eine Liste von Objekten verschiedenen Typs enthalten, wobei die Anzahl der zurückgelieferten Elemente 10.000 aus Last- und Performance-Gründen nicht überschreiten darf. Andernfalls wird eine Fehlermeldung von Typ `tooMany` zurückgegeben. Die Suchbedingung sollte verfeinert werden und das Ergebnis stärker einschränken.

Anfragen an eine Endpunkt-URL

Die SCIM 2.0 Spezifikation sieht bei Anfragen an die über `/ResourceTypes` definierten Endpunkte optional die Parameter `filter`, `attributes`, `count` und `startIndex` vor. Bei Anfragen mit der ID eines konkreten Objektes (die URL enthält die `id` des Objektes) sind die Parameter `excludedAttributes` und `attributes` erlaubt. Das SCIM Plugin unterstützt diese Parameter.

Bei Anfragen an einen Endpunkt wird eine Liste aller Elemente (oder aller Elemente, die dem Filter entsprechen) zurückgegeben. So kann vom SCIM Client ein indexbasiertes Paging mit Angabe der gewünschten Anzahl von Sätzen pro Seite initiiert werden (Parameter `count` und `startIndex`).

Beispiel: Anfrage an einen Endpunkt

```
http://<servername>.<domainname>/ApiServer/scim/v2/Person
```

Beispiel: Anfrage der ersten 100 Elemente an einen Endpunkt mit Paging

```
http://<servername>.<domainname>/ApiServer/scim/v2/Person?startindex=1&count
=100
```

Beispiel: Anfrage an einen Endpunkt mit Filter

```
http://<servername>.<domainname>/ApiServer/scim/v2/Person?filter=InternalNam
e co "Y"
```

Beispiel: Anfrage an einen Endpunkt mit Filter und Ausgabe von zwei zusätzlichen Eigenschaften

```
http://<servername>.<domainname>/ApiServer/scim/v2/Person?filter=InternalNam
e co "Y"&attributes=ExitDate,TechnicalEntryDate
```

Beispiel: Anfrage eines Objektes an einen Endpunkt

```
http://<servername>.<domainname>/ApiServer/scim/v2/UNSGroupB/94bbe614-0a6e-
4659-8fe9-20da94d967c8
```

Beispiel: Anfrage von konkreten Eigenschaften eines Objektes an einen Endpunkt

```
http://<servername>.<domainname>/ApiServer/scim/v2/UNSGroupB/94bbe614-0a6e-
4659-8fe9-20da94d967c8?attributes=cn,members
```

18

SOAP Web Service

Der SOAP Web Service des One Identity Manager stellt eine SOAP Schnittstelle zum Zugriff auf das One Identity Manager Objektmodell bereit. Der SOAP Web Service verwaltet einen Verbindungspool. Nicht jeder Aufruf öffnet eine neue Verbindung. Mit dem SOAP Web Service werden nicht alle Funktionen der Objektschicht unterstützt. Der SOAP Web Service stellt die Methoden für Einzelobjekte, Objektlisten und Funktionsaufrufe zur Verfügung.

Tabelle 184: Methoden für Einzelobjekte

Methode	Beschreibung
CreateSingleObject	Anlegen eines neuen einzelnen Objektes.
GetCompleteSingleObject	Laden eines einzelnen vollständigen Objektes aus der Datenbank mit allen Parametern.
GetCompleteSingleObjectEx	Funktionalität analog zu `GetCompleteSingleObject` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
GetSingleObject	Laden eines einzelnen Objektes aus der Datenbank.
GetSingleObjectEx	Funktionalität analog zu `GetSingleObject` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
ChangeSingleObject	Speichern der Veränderungen eines einzelnen Objektes.
ChangeSingleObjectEx	Funktionalität analog zu `ChangeSingleObject` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
DeleteSingleObject	Löschen eines einzelnen Objektes.
DeleteSingleObjectEx	Funktionalität analog zu `DeleteSingleObject` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
Exists	Existiert ein bestimmtes Einzelobjekt?
ExistsEx	Funktionalität analog zu `Exists` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
GetSingleProperty	Holen eines Einzelwertes eines Objektes.
GetSinglePropertyEx	Funktionalität analog zu `GetSingleProperty` mit der Unterstützung eines mehrspaltigen Primärschlüssels.

Tabelle 185: Methoden für Objektlisten

Methode	Beschreibung
GetListObject	Laden einer Liste von Objekten.
GetListObjectWithDisplays	Laden einer Liste von Objekten mit Angabe zusätzlich zum Primärschlüssel zu ladender Spalten.

Tabelle 186: Methoden für Funktionsaufrufe

Funktion	Beschreibung
InvokeCustomizer	Aufruf einer Customizer-Methode auf einem Objekt.
InvokeCustomizerEx	Funktionalität analog zu `InvokeCustomizer` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
InvokeDialogMethod	Aufruf einer Dialogmethode auf einem Dialogobjekt.
InvokeDialogMethodEx	Funktionalität analog zu `InvokeDialogMethod` mit der Unterstützung eines mehrspaltigen Primärschlüssels.
FireGenEvent	Generieren der Prozesse eines bestimmten Ereignisses.
FireGenEventEx	Funktionalität analog zu `FireGenEvent` mit der Unterstützung eines mehrspaltigen Primärschlüssels.

Detaillierte Informationen zum Thema

- Installation und Konfiguration des SOAP Web Service auf Seite 518
- Beispiele für Aufrufe auf Seite 526

Installation und Konfiguration des SOAP Web Service

Für die Installation des SOAP Web Service muss ein Server zur Verfügung gestellt werden, auf dem die nachfolgend genannte Software bereits installiert ist.

- Windows Betriebssysteme

  Unterstützt werden die Versionen:

  - Windows Server 2022
  - Windows Server 2019
  - Windows Server 2016

    - Windows Server 2012 R2
    - Windows Server 2012
- Microsoft .NET Framework Version 4.7.2 oder höher
- Microsoft Internet Information Services 10 oder 8.5 oder 8 oder 7.5 oder 7 mit ASP.NET 4.7.2 und den Role Services:
    - Web Server > Common HTTP Features > Static Content
    - Web Server > Common HTTP Features > Default Document
    - Web Server > Application Development > ASP.NET
    - Web Server > Application Development > .NET Extensibility
    - Web Server > Application Development > ISAPI Extensions
    - Web Server > Application Development > ISAPI Filters
    - Web Server > Security > Basic Authentication
    - Web Server > Security > Windows Authentication
    - Web Server > Performance > Static Content Compression
    - Web Server > Performance > Dynamic Content Compression

Benötigte Berechtigungen

- Das Benutzerkonto unter dem der Internet Information Service läuft benötigt schreibenden Zugriff (**MODIFY**) auf das Installationsverzeichnis.
- Für die automatische Aktualisierung sind folgende Berechtigungen erforderlich:
    - Das Benutzerkonto für die Aktualisierung benötigt die Berechtigung zum Schreiben auf das Anwendungsverzeichnis.
    - Das Benutzerkonto für die Aktualisierung benötigt die lokale Sicherheitsrichtlinie **Anmelden als Stapelverarbeitungsauftrag**.
    - Das Benutzerkonto, unter dem der Anwendungspool läuft, benötigt die lokalen Sicherheitsrichtlinien **Ersetzen eines Tokens auf Prozessebene** und **Anpassen von Speicherkontingenten für einen Prozess**.

Detaillierte Informationen zum Thema

- SOAP Web Service installieren auf Seite 519
- Konfigurieren des SOAP Web Service auf Seite 523
- Status eines SOAP Web Service anzeigen auf Seite 525
- SOAP Web Service deinstallieren auf Seite 525

SOAP Web Service installieren

WICHTIG: Starten Sie die Installation des SOAP Web Service lokal auf dem Server.

Um den SOAP Web Service zu installieren

1. Starten Sie die Datei `autorun.exe` aus dem Basisverzeichnis des One Identity Manager Installationsmediums.
2. Auf der Startseite des Installationsassistenten:
   a. Wechseln Sie zum Tabreiter **Installation**.
   b. Im Bereich **Web-basierte Komponenten** klicken Sie **Installieren**.

   Der Web Installer wird gestartet.
3. Auf der Startseite des Web Installer wählen Sie **SOAP Web Service installieren** und klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** nehmen Sie eine der folgenden Aktionen vor:

   TIPP: Es wird die Verwendung einer Verbindung über einen Anwendungsserver empfohlen.

   - Um eine bestehende Verbindung zur One Identity Manager-Datenbank zu verwenden, wählen Sie in der Auswahlliste **Datenbankverbindung auswählen** die entsprechende Verbindung aus.

     - ODER -
   - Um eine neue Verbindung zur One Identity Manager-Datenbank zu verwenden, klicken Sie **Neue Verbindung erstellen** und geben Sie eine neue Verbindung an.
5. Unter **Authentifizierungsverfahren** geben Sie das Verfahren und die Anmeldedaten an, mit denen Sie sich an der Datenbank anmelden möchten.
6. Auf der Seite **Installationsziel wählen** nehmen Sie die folgenden Einstellungen vor und klicken Sie **Weiter**.

Tabelle 187: Einstellungen für das Installationsziel

Einstellung	Beschreibung
Anwendungsname	Name, der als Anwendungsname zum Beispiel in der Titelzeile des Browsers verwendet werden soll.
Zielpfad im IIS	Webseite auf dem Internet Information Services, auf dem die Anwendung installiert wird.
SSL erzwingen	Gibt an, ob sichere oder unsichere Webseiten zur Installation angeboten werden. Ist die Option aktiviert, können nur Seiten, die per SSL gesichert sind, zur Installation verwendet werden. Diese Einstellung ist der Standardwert. Ist die Option nicht aktiviert, können unsichere Webseiten zur Installation verwendet werden.
URL	Uniform Resource Locator (URL) der Anwendung.
Dedizierten Anwen-	Gibt an, ob für jede Anwendung ein eigener Anwendungspool

Einstellung	Beschreibung
dungspool einrich-ten	installiert werden soll. Diese Option ermöglicht es, Anwen-dungen unabhängig voneinander einzustellen. Ist die Option aktiviert, wird jede Anwendung in ihren eigenen Anwen-dungspool installiert.
Anwendungspool	Anwendungspool, der verwendet werden soll. Die Angabe ist nur möglich, wenn die Option **Dedizierter Anwendungspool einrichten** deaktiviert ist. Wenn Sie den Standardwert **DefaultAppPool** verwenden, wird der Anwendungspool nach folgender Syntax gebildet: `<Anwendungsname>_POOL`
Identität	Berechtigung für die Ausführung des Anwendungspool. Sie können eine Standard-Identität oder ein benutzerdefiniertes Benutzerkonto verwenden. Wenn Sie den Standardwert **ApplicationPoolIdentity** verwenden, wird das Benutzerkonto nach folgender Syntax gebildet: `IIS APPPOOL\<Anwendungsname>_POOL` Wenn Sie einen anderen Benutzer berechtigen möchten, klicken Sie **...** neben dem Eingabefeld, aktivieren Sie die Option **Benutzerdefiniertes Konto** und erfassen Sie den Benutzer und sein Kennwort.
Web-Authen-tifizierung	Authentifizierungsart gegenüber der Webanwendung. Zur Auswahl stehen: • **Windows Authentifizierung (Single Sign-On)** Der Benutzer wird gegenüber dem Internet Information Services mithilfe seines Windows-Benutzerkontos authentifiziert und die Webanwendung meldet die diesem Benutzerkonto zugeordnete Person rollenbasiert an. Sollte dieses Single Sign-on nicht möglich sein, wird der Benutzer auf eine Anmeldeseite umgeleitet. Diese Authentifizierung ist nur wählbar, wenn die Windows-Authentifizierung installiert ist. • **Anonym** Eine Anmeldung ohne Windows-Authentifizierung ist möglich. Der Benutzer wird gegenüber dem Internet Information Services und der Webanwendung anonym authentifiziert und die Webanmeldung leitet auf eine Anmeldeseite um.
Datenbank-Authen-	HINWEIS: Dieser Bereich wird Ihnen nur angezeigt, wenn

Einstellung	Beschreibung
tifizierung	Sie auf der Seite **Datenbankverbindung** eine SQL-Datenbankverbindung ausgewählt haben. Authentifizierungsart gegenüber der One Identity Manager-Datenbank. Zur Auswahl stehen: • **Windows Authentifizierung** Die Webanwendung authentifiziert sich gegenüber der One Identity Manager-Datenbank mit dem Windows-Benutzerkonto, unter dem ihr Anwendungspool läuft. Eine Anmeldung ist über ein benutzerdefiniertes Benutzerkonto oder einer Standard-Identität für den Anwendungspool möglich. • **SQL-Authentifizierung** Die Authentifizierung erfolgt mittels SQL Server-Anmeldung und Kennwort. Es wird die SQL Server-Anmeldung aus der Verbindung zur Datenbank verwendet. Über die Schaltfläche [**...**] können Sie eine abweichende SQL-Anmeldung angeben, beispielsweise wenn die Anwendung mit einer Berechtigungsebene für Endbenutzer ausgeführt werden soll. Diese Zugangsdaten werden maschinenspezifisch verschlüsselt in der Konfiguration der Webanwendung gespeichert.

7. Auf der Seite **Setze das Konto für Aktualisierung** legen Sie das Benutzerkonto für die automatische Aktualisierung des Anwendungsservers fest.

1. Das Benutzerkonto wird verwendet, um die Dateien im Anwendungsverzeichnis anzulegen oder auszutauschen.
   - Wenn Sie die Benutzerkonto, unter welcher der Anwendungspool ausgeführt wird, für die Aktualisierungen nutzen möchten, setzen Sie die Option **Nutze die IIS-Berechtigungen für Aktualisierungen**.
   - Wenn Sie ein anderes Benutzerkonto verwendet möchten, setzen Sie die Option **Nutze ein spezielles Konto für die Aktualisierungen** und geben Sie die Domäne, den Benutzernamen und das Kennwort des Benutzers an.

8. Auf der Seite **Installation läuft** werden die einzelnen Installationsschritte angezeigt. Nachdem der Installationsvorgang abgeschlossen wurde, klicken Sie **Weiter**.

   Der Web Installer generiert die Webanwendung und die entsprechenden Konfigurationsdateien (`web.config`) zu jedem Verzeichnis.

9. Auf der letzten Seite klicken Sie **Fertig**, um das Programm zu beenden.

Konfigurieren des SOAP Web Service

Die Konfiguration des SOAP Web Service befindet sich im Installationsverzeichnis des Webservice in der XML-Datei `web.config`. Diese Datei können Sie mit einem beliebigen Texteditor bearbeiten.

Tabelle 188: Konfigurierbare Optionen der Konfigurationsdatei Web.config

Sektion	Option	Gültige Werte	Beschreibung
connectionString			Verbindungsparameter zur Datenbank.
runtimedirs	key="Cache"	value = "<Pfad>"	Verzeichnis zur Ablage der Cache Verzeichnisse. Standard: `value="C:\inetpub\wwwroot\<web service name>\App_Data\Cache\DB"`
	key="AssemblyCache"	value = "<Pfad>"	Verzeichnis zur Ablage der Cache Verzeichnisse. Standard: `value="C:\inetpub\wwwroot\<web service name>\App_Data\Cache\Assemblies"`
settings	key="timeout"	value="<Zeitangabe>"	Timeout für die Verbindungen im Anwendungspool. Standard: `value="00:05:00"`
	key="maxconnectionlifetime"	value="<Zeitangabe>"	Maximale Zeit, die die Verbindungen aufrecht erhalten werden. Nach Ablauf der Zeit werden Verbindungen geschlossen auch wenn das Timeout noch nicht erreicht wurde. Standard: `value="00:05:00"`
	key="usepropertybag"	value = "True" value = "False"	Gibt an, ob der PropertyBag genutzt wird. Der PropertyBag wird

Sektion	Option	Gültige Werte	Beschreibung
			benutzt, um bei der Befüllung von Objekteigenschaften die von Nebenwirkungen oder Bildungsregeln geforderte Bestückungsreihenfolge einzuhalten. Zulässige Werte sind: • **False**: Werte werden in der Reihenfolge, in der sie angegeben wurden, im Objekt gesetzt. • **True**: Die Bestückungsreihenfolge wird aus den Metadaten bestimmt. Standard: `value="True"`
	key="ignoreinvisiblevalues"	value = "True" value = "False"	Gibt an, ob Werte, die der Benutzer nicht sehen darf, nicht zurückgegeben werden. Zulässige Werte sind: • **False**: Bei Werten, die der Benutzer nicht sehen darf, wird eine Fehlermeldung erzeugt. • **True**: Werte, die der Benutzer nicht sehen darf, werden nicht zurückgegeben. Das Setzen dieses Wertes verhindert, dass der Benutzer eine Fehlermeldung erhält.

Sektion	Option	Gültige Werte	Beschreibung
			Standard: `value="True"`
	key = "logdirectory"	value = "<Pfad>"	Verzeichnis für die Protokolle. Standard: `value = "C:\inetpub\wwwroot\<web service name>\App_Data\Logs`
	key = "allowwebservicemethods"	value = "Liste des Methoden"	Semikolon-getrennte Liste der erlaubten Webservice Methoden.
	key = "allowfunctions"	value = "Liste der Funktionen"	Liste der erlaubten Funktionen per `CallFunction`-Methode. Wenn keinen Funktion angegeben ist, sind alle Funktionen erlaubt.

Verwandte Themen

- Spaltenabhängigkeiten zur Wertebestückung auf Seite 91

Status eines SOAP Web Service anzeigen

Der SOAP Web Service ist über einen Browser erreichbar unter:

http://<Server>/<Anwendungsname>

https://<Server>/<Anwendungsname>

> TIPP: Sie können die Statusanzeige des Webservers im Job Queue Info öffnen. Wählen Sie dazu im Job Queue Info das Menü **Ansicht > Serverstatus** und öffnen Sie auf dem Tabreiter **Webserver** die Statusanzeige des Webservers über das Kontextmenü **Im Browser öffnen**.

Zusätzlich ist hier eine API Dokumentation verfügbar.

SOAP Web Service deinstallieren

Um eine Webanwendung zu deinstallieren

1. Starten Sie die Datei `autorun.exe` aus dem Basisverzeichnis des One Identity Manager Installationsmediums.

2. Auf der Startseite des Installationsassistenten:
   a. Wechseln Sie zum Tabreiter **Installation**.
   b. Im Bereich **Web-basierte Komponenten** klicken Sie **Installieren**.

   Der Web Installer wird gestartet.
3. Auf der Startseite des Web Installer klicken Sie **Deinstallieren einer Webanwendung** und klicken Sie **Weiter**.
4. Auf der Seite **Deinstallieren einer Webanwendung** doppelklicken Sie die Webanwendung, die Sie entfernen möchten.

   Vor der Anwendung wird das Symbol angezeigt.
5. Klicken Sie **Weiter**.
6. Auf der Seite **Datenbankverbindung** wählen Sie die Datenbankverbindung und das Authentifizierungsverfahren und geben Sie die entsprechenden Anmeldedaten ein.
7. Klicken Sie **Weiter**.
8. Bestätigen Sie die Sicherheitsabfrage mit **Ja**.
9. Auf der Seite **Installation läuft** werden die einzelnen Schritte zur Deinstallation angezeigt.
10. Nachdem der Installationsvorgang abgeschlossen wurde, klicken Sie **Weiter**.
11. Auf der Seite **Beenden des Assistenten** klicken Sie **Fertig**.
12. Schließen Sie das Autorun-Programm.

Beispiele für Aufrufe

Einen Überblick über die bereitgestellten Methoden finden Sie unter SOAP Web Service auf Seite 517. Nachfolgend einige Beispiele für Aufrufe eines Webservice Clients in der Programmiersprache C#.

Vorbereitung

Die Authentifizierung erfolgt über eine Authentifizierungszeichenkette, die ein Authentifizierungsmodul und die zu verwendenden Anmeldeinformationen enthält. Sie müssen eine Instanz des Webservices und die Objekt für die Anmeldeinformationen zur Systemanmeldung erzeugen. Diese Anmeldeinformation wird an weitere Aufrufe weitergereicht.

Beispiel:

```
var svc = new Q1IMServiceSoapClient();

var login = new LoginInformation

     { AuthString = "Module=DialogUser;User=viadmin;Password=" };
```

Tabelle 189: Beispiele für Authentifizierung

Authentifizierungsmodul	Beispiel
Systembenutzer	Module=DialogUser;User=<user name>;Password=<password>
Person	Module=Person;User=<central user account>;Password=<password>
Active Directory Benutzerkonto (rollenbasiert)	Module=RoleBasedADSAccount
Active Directory Benutzerkonto (manuelle Eingabe/rollenbasiert)	Module=RoleBasedManualADS;User=<AD user name>;Password=<AD password>

Ausführliche Informationen zu den One Identity Manager Authentifizierungsmodulen finden Sie im *One Identity Manager Handbuch zur Autorisierung und Authentifizierung*.

GetListObject

Diese Methode gibt ein Array von Objekten zurück, die der angegebenen Where-Klausel entsprechen. Das zurückgegebene Array enthält die Primärschlüssel der Objekte und einen speziellen Schlüssel `[DISPLAY]`, der den Anzeigewert des Objektes enthält.

Beispiel:

```
Q1IMService.KeyValuePair[][] objects = svc.GetListObject(login, "Person",
     "FirstName like 'Hal%'");
```

GetListObjectWithDisplays

Diese Methode arbeitet analog zu `GetListObject` und erlaubt die Angabe von zusätzlich zu ladenden Spalten.

Beispiel:

Im Beispiel stehen die Spalten `FirstName` und `LastName` im Ergebnis zur Verfügung.

```
Q1IMService.KeyValuePair[][] objects = svc.GetListObjectWithDisplays(login,
"Person",
     "FirstName like 'Hal%'",
     new [] {"FirstName", "LastName"});
```

GetCompleteSingleObject

Über die Methode werden alle Eigenschaften des Objektes, welches über die Primärschlüsselspalte (Spalte und Inhalt der Spalte) definiert ist, geladen.

Beispiel:

```
Q1IMService.KeyValuePair[] singleValues = svc.GetCompleteSingleObject(login,
     "Person", "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05");
```

GetSingleObject

Über diese Methode werden Eigenschaften eines einzelnen Objektes geladen.

Beispiel:

Im Beispiel werden die Spalten `FirstName`, `LastName` und der Anzeigewert geladen. Der Anzeigewert wird im Schlüssel `[DISPLAY]` ausgegeben.

```
Q1IMService.KeyValuePair[] values = svc.GetSingleObject(login, "Person",
     "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05",
     new[] { "FirstName", "LastName" });
```

ChangeSingleObject

Über diese Methode werden einzelne Eigenschaften eines Objektes geändert.

Beispiel:

Im Beispiel wird die Spalte `Description` der Person mit der entsprechenden `UID_Person` geändert.

```
var values = new[]
     {
          new Q1IMService.KeyValuePair
               {
                    Key = "Description",
                    Value = "Created by webservice"
               }
     };
svc.ChangeSingleObject(login, "Person", "UID_Person",
     "746a5662-054b-4531-a889-1c135dad4c05", values);
```

ChangeSingleObjectEx

Die Änderung eines Objektes über diese Methode erfolgt analog zu `ChangeSingleObject`, nur werden hier die Primärschlüsselwerte als Key-Value-Pair-Array übergeben.

Beispiel:

```
var values = new[]
     {
          new Q1IMService.KeyValuePair
               {
                    Key = "Description",
                    Value = "Created by webservice"
               }
```

```
    };
var keys = new[]
    {
        new Q1IMService.KeyValuePair
            {
                Key = "UID_Person",
                Value = "746a5662-054b-4531-a889-1c135dad4c05"
            }
    };
svc.ChangeSingleObjectEx(login, "Person", keys, values);
```

DeleteSingleObject

Über diese Methode wird ein Objekt gelöscht.

Beispiel:

Im Beispiel wird die Person mit der entsprechenden UID aus der Datenbank gelöscht.

```
svc.DeleteSingleObject(login, "Person",
    "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05");
```

DeleteSingleObjectEx

Mit dieser Methode ist es möglich, Objekte mit mehrspaltigem Primärschlüssel, zum Beispiel aus M:N-Tabellen, zu löschen.

Beispiel:

```
svc.DeleteSingleObjectEx (
    login,
    "OrgHasApp",
    new []
    {
        new Q1IMService.KeyValuePair { Key = "UID_Org", Value = <UID> },
        new Q1IMService.KeyValuePair { Key = "UID_Application", Value = <UID>}
    });
```

CreateSingleObject

Über diese Methode wird ein neues Objekt in der Datenbank erzeugt.

Beispiel:

Im Beispiel wird die Person "John Doe" erzeugt.

```
var values = new[]
```

```
    {
        new Q1IMService.KeyValuePair {Key = "FirstName", Value = "John"},
        new Q1IMService.KeyValuePair {Key = "LastName", Value = "Doe"}
    };
svc.CreateSingleObject(login, "Person", values);
```

Exists

Diese Methode überprüft die Existenz eines Objektes.

Beispiel:

```
bool exists = svc.Exists(login, "Person",
    "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05");
```

GetSingleProperty

Diese Methode kann eingesetzt werden, um eine einzelne Eigenschaft zu ermitteln.

Beispiel:

```
string description = svc.GetSingleProperty(login, "Person",
    "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05",
    "Description");
```

InvokeCustomizer

Der SOAP Web Service unterstützt eine `InvokeCustomizer`-Methode , die eine Funktion auf einem Objekt in der Datenbank aufruft. Die ersten drei Parameter spezifizieren das Objekt, auf dem die Methode aufgerufen wird. Der Parameter `customizerName` gibt den Namen der Funktion an. Es folgt ein Array von Strings, in welchem die voll qualifizierten Namen der Datentypen der Parameter stehen, die der aufzurufenden Funktion übergeben werden sollen. Das folgende Array von Strings enthält die textuelle Repräsentation der Parameter.

Arbeitsweise der Funktion:

- Als erstes wird die Datenbank geöffnet und das Objekt, welches durch `objectType`, `pkName` und `pkValue` spezifiziert worden ist, geholt.
- Anschliessend werden die Runtime-Datentypen ermittelt, welche durch `parameterTypes` angegeben werden.
- Danach werden die Text-Repräsentationen der Parameter aus dem Werte Array in korrespondierende der Runtime-Datentypen konvertiert.
- Mit diesen Werten wird die Funktion aufgerufen.

Hat die aufzurufende Funktion keine Parameter, so kann der Funktion für die Parameter `parameterTypes` und `parameters` der Wert **null** übergeben werden.

Beispiel:

Im Beispiel wird die Methode "TestMethod" auf einem Objekt vom Typ `Person` mit der Primärschlüsselspalte `UID_Person` und dem angegebenen Wert, aufgerufen. Dabei werden die beiden Parameter vom Typ `System.String` und `System.Int32` mit den Werten "Foo" und "4711" übergeben.

```
svc.InvokeCustomizer (login, "Person",
     "UID_Person", "0000644F-C139-4B25-8D1C-5ECB93067E79",
     "TestMethod",
     new [] {"System.String", "System. Int32"},
     new [] {"foo", "4711"});
```

InvokeDialogMethod

Die Methode kann eine Dialogmethode auf einem Objekt aufrufen. Dialogmethoden besitzen keine Parameter und keine Rückgabewerte. Der Aufruf ähnelt dem Aufruf von `InvokeCustomizer`.

Beispiel:

Im Beispiel wird die Methode "TestDialogMethod" für eine bestimmte Person aufgerufen. "TestDialogMethod" ist die Bezeichnung der Methode entsprechend `DialogMethod.MethodName`.

```
svc.InvokeDialogMethod (login,
     "Person",
     "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05",
     "TestDialogMethod");
```

FireGenEvent

Über diese Methode werden die Prozesse eines bestimmten Ereignisses generiert. Es besteht die Möglichkeit, zusätzliche Generierungsparameter anzugeben.

```
public void FireGenEvent(
     string objectType, string pkName, string pkValue,
     strincolumng eventName, KeyValuePair[] parameters);
```

Beispiel:

Im Beispiel wird das Ereignis "EXPORT_DATA" ohne zusätzliche Parameter generiert.

```
svc.FireGenEvent(login, "Person",
     "UID_Person", "746a5662-054b-4531-a889-1c135dad4c05",
     "EXPORT_DATA", new Q1IMService.KeyValuePair[] { });
```

CallFunction

Diese Methode ruft eine Funktion der One Identity Manager Skriptbibliothek auf.

Beispiel:

Im Beispiel wird das Skript `VI_BuildInitials` aufgerufen.

```
svc.CallFunction(login, "VI_BuildInitials",
     new string [] {"John", "Doe"});
```

One Identity Manager als SPML Provisioning Service Provider

Der One Identity Manager ermöglicht den Datenaustausch mit Systemen anderer Hersteller über SPML. SPML steht für Service Provisioning Markup Language und definiert eine standardisierte Schnittstelle für den Austausch von Provisioning Informationen. Die Version 2 von SPML (SPLMLv2) wurde im April 2006 von der Organization for the Advancement of Structured Information Standards (OASIS, www.oasis-open.org) veröffentlicht. Die Schnittstelle dient dazu den Datenaustausch im Zusammenhang mit komplexen Provisioning Lösungen und Umgebungen zu vereinfachen und zu standardisieren.

Der One Identity Manager kann als SPML Client und als SPML Provider betrieben werden. An dieser Stelle wird näher auf die Konfiguration des One Identity Manager als SPML Provider eingegangen. Der SPML Provider unterstützt das komplette One Identity Manager Schema. Welche Objekte und Beziehungen über den SPML Provider verwaltet werden, kann kundenspezifisch konfiguriert werden.

Detaillierte Informationen zum Thema

- SPML Webservice auf Seite 533
- Installation und Konfiguration des SPML Webservice auf Seite 534
- Konfiguration des One Identity Manager Schemas auf Seite 542
- Testen der Funktionalität des SPML Webservice auf Seite 544

SPML Webservice

Für die Funktion als SPML Service Provider wird ein Webservice, der SPML Webservice, zur Verfügung gestellt. Der SPML Webservice ist SPMLv2 konform und wurde auf der Grundlage der von OASIS veröffentlichten Dokumente implementiert. Er stellt die Hauptoperationen, wie das Hinzufügen, Löschen, Ändern von Objekten sowie Erweiterungen zum Suchen und Referenzieren von Objekten zur Verfügung.

Der SPML Webservice unterstützt folgende definierte Funktionen von SPMLv2.

Tabelle 190: Unterstützte Funktionen von SPMLv2

Funktion	Beschreibung
listTargetsRequest	Gibt die Zielsysteme des Providers mit ihrem spezifischen Schema zurück. Der SPML Provider unterstützt ausschließlich das Schema One Identity Manager.
addRequest	Legt ein neues Objekt im angegebenen Zielsystem des Providers mit den übermittelten Eigenschaften an.
lookupRequest	Liefert die Eigenschaften eines durch einen Schlüssel identifiziertes Objektes.
modifyRequest	Ändert die übermittelten Eigenschaften eines durch einen Schlüssel identifiziertes Objektes im angegebenen Zielsystem des Providers.
deleteRequest	Löscht ein durch einen Schlüssel identifiziertes Objekt im Zielsystem des Providers.
searchRequest	Liefert alle auf die Suchkriterien zutreffenden Objekte des Zielsystems des Providers zurück.
iterateRequest	Liefert bei einer Suche weitere Datensätze sofern noch nicht alle Suchergebnisse an den Client übermittelt wurden.
closeIteratorRequest	Schließt eine aktive Suche ab und teilt dem Provider mit, dass keine weiteren Ergebnisse gewünscht werden.

Mit der Erweiterung `Reference` ist es möglich, Referenzen zwischen verschiedenen Objekten des Zielsystems des Providers zu pflegen. Dabei sind zwei Typen von Referenzen zu unterscheiden.

- Referenz vom Typ **owner**

  Referenzen des Typs **owner** resultieren im One Identity Manager in Fremdschlüsselbeziehungen.

- Referenz vom Typ **memberOf**

  Referenzen des Typs **memberOf** resultieren im One Identity Manager in M:N-Zuordnungen.

Installation und Konfiguration des SPML Webservice

Für die Installation des SPML Webservice muss ein Server zur Verfügung gestellt werden, auf dem die nachfolgend genannte Software bereits installiert ist.

- Windows Betriebssysteme

  Unterstützt werden die Versionen:

  - Windows Server 2022
  - Windows Server 2019
  - Windows Server 2016
  - Windows Server 2012 R2
  - Windows Server 2012
- Microsoft .NET Framework Version 4.7.2 oder höher
- Microsoft Internet Information Services 10 oder 8.5 oder 8 oder 7.5 oder 7 mit ASP.NET 4.7.2 und den Role Services:
  - Web Server > Common HTTP Features > Static Content
  - Web Server > Common HTTP Features > Default Document
  - Web Server > Application Development > ASP.NET
  - Web Server > Application Development > .NET Extensibility
  - Web Server > Application Development > ISAPI Extensions
  - Web Server > Application Development > ISAPI Filters
  - Web Server > Security > Basic Authentication
  - Web Server > Security > Windows Authentication
  - Web Server > Performance > Static Content Compression
  - Web Server > Performance > Dynamic Content Compression

Benötigte Berechtigungen

- Das Benutzerkonto unter dem der Internet Information Service läuft benötigt schreibenden Zugriff (`MODIFY`) auf das Installationsverzeichnis.
- Für die automatische Aktualisierung sind folgende Berechtigungen erforderlich:
  - Das Benutzerkonto für die Aktualisierung benötigt die Berechtigung zum Schreiben auf das Anwendungsverzeichnis.
  - Das Benutzerkonto für die Aktualisierung benötigt die lokale Sicherheitsrichtlinie **Anmelden als Stapelverarbeitungsauftrag**.
  - Das Benutzerkonto, unter dem der Anwendungspool läuft, benötigt die lokalen Sicherheitsrichtlinien **Ersetzen eines Tokens auf Prozessebene** und **Anpassen von Speicherkontingenten für einen Prozess**.

Detaillierte Informationen zum Thema

- SPML Webservice installieren auf Seite 536
- Konfigurieren des SPML Webservice auf Seite 539

SPML Webservice installieren

WICHTIG: Starten Sie die Installation des SPML Webservice lokal auf dem Server.

Um den SPML Webservice zu installieren

1. Starten Sie die Datei `autorun.exe` aus dem Basisverzeichnis des One Identity Manager Installationsmediums.
2. Auf der Startseite des Installationsassistenten:
   a. Wechseln Sie zum Tabreiter **Installation**.
   b. Im Bereich **Web-basierte Komponenten** klicken Sie **Installieren**.

   Der Web Installer wird gestartet.
3. Auf der Startseite des Web Installer wählen Sie **SPML Webservice installieren** und klicken Sie **Weiter**.
4. Auf der Seite **Datenbankverbindung** nehmen Sie eine der folgenden Aktionen vor:

   TIPP: Es wird die Verwendung einer Verbindung über einen Anwendungsserver empfohlen.

   - Um eine bestehende Verbindung zur One Identity Manager-Datenbank zu verwenden, wählen Sie in der Auswahlliste **Datenbankverbindung auswählen** die entsprechende Verbindung aus.

     - ODER -
   - Um eine neue Verbindung zur One Identity Manager-Datenbank zu verwenden, klicken Sie **Neue Verbindung erstellen** und geben Sie eine neue Verbindung an.
5. Auf der Seite **Installationsziel wählen** nehmen Sie die folgenden Einstellungen vor und klicken Sie **Weiter**.

Tabelle 191: Einstellungen für das Installationsziel

Einstellung	Beschreibung
Anwendungsname	Name, der als Anwendungsname zum Beispiel in der Titelzeile des Browsers verwendet werden soll.
Zielpfad im IIS	Webseite auf dem Internet Information Services, auf dem die Anwendung installiert wird.
SSL erzwingen	Gibt an, ob sichere oder unsichere Webseiten zur Installation angeboten werden. Ist die Option aktiviert, können nur Seiten, die per SSL gesichert sind, zur Installation verwendet werden. Diese Einstellung ist der Standardwert. Ist die Option nicht aktiviert, können unsichere Webseiten zur Installation verwendet werden.

Einstellung	Beschreibung
URL	Uniform Resource Locator (URL) der Anwendung.
Dedizierten Anwendungspool einrichten	Gibt an, ob für jede Anwendung ein eigener Anwendungspool installiert werden soll. Diese Option ermöglicht es, Anwendungen unabhängig voneinander einzustellen. Ist die Option aktiviert, wird jede Anwendung in ihren eigenen Anwendungspool installiert.
Anwendungspool	Anwendungspool, der verwendet werden soll. Die Angabe ist nur möglich, wenn die Option **Dedizierter Anwendungspool einrichten** deaktiviert ist. Wenn Sie den Standardwert **DefaultAppPool** verwenden, wird der Anwendungspool nach folgender Syntax gebildet: `<Anwendungsname>_POOL`
Identität	Berechtigung für die Ausführung des Anwendungspool. Sie können eine Standard-Identität oder ein benutzerdefiniertes Benutzerkonto verwenden. Wenn Sie den Standardwert **ApplicationPoolIdentity** verwenden, wird das Benutzerkonto nach folgender Syntax gebildet: `IIS APPPOOL\<Anwendungsname>_POOL` Wenn Sie einen anderen Benutzer berechtigen möchten, klicken Sie **...** neben dem Eingabefeld, aktivieren Sie die Option **Benutzerdefiniertes Konto** und erfassen Sie den Benutzer und sein Kennwort.
Web-Authentifizierung	Authentifizierungsart gegenüber der Webanwendung. Zur Auswahl stehen: • **Windows Authentifizierung (Single Sign-On)** Der Benutzer wird gegenüber dem Internet Information Services mithilfe seines Windows-Benutzerkontos authentifiziert und die Webanwendung meldet die diesem Benutzerkonto zugeordnete Person rollenbasiert an. Sollte dieses Single Sign-on nicht möglich sein, wird der Benutzer auf eine Anmeldeseite umgeleitet. Diese Authentifizierung ist nur wählbar, wenn die Windows-Authentifizierung installiert ist. • **Anonym** Eine Anmeldung ohne Windows-Authentifizierung ist möglich. Der Benutzer wird gegenüber dem Internet Information Services und der Webanwendung anonym authentifiziert und die Webanmeldung leitet auf eine

Einstellung	Beschreibung
	Anmeldeseite um.
Datenbank-Authentifizierung	HINWEIS: Dieser Bereich wird Ihnen nur angezeigt, wenn Sie auf der Seite **Datenbankverbindung** eine SQL-Datenbankverbindung ausgewählt haben. Authentifizierungsart gegenüber der One Identity Manager-Datenbank. Zur Auswahl stehen: • **Windows Authentifizierung** Die Webanwendung authentifiziert sich gegenüber der One Identity Manager-Datenbank mit dem Windows-Benutzerkonto, unter dem ihr Anwendungspool läuft. Eine Anmeldung ist über ein benutzerdefiniertes Benutzerkonto oder einer Standard-Identität für den Anwendungspool möglich. • **SQL-Authentifizierung** Die Authentifizierung erfolgt mittels SQL Server-Anmeldung und Kennwort. Es wird die SQL Server-Anmeldung aus der Verbindung zur Datenbank verwendet. Über die Schaltfläche [**...**] können Sie eine abweichende SQL-Anmeldung angeben, beispielsweise wenn die Anwendung mit einer Berechtigungsebene für Endbenutzer ausgeführt werden soll. Diese Zugangsdaten werden maschinenspezifisch verschlüsselt in der Konfiguration der Webanwendung gespeichert.

6. Auf der Seite **Setze das Konto für Aktualisierung** legen Sie das Benutzerkonto für die automatische Aktualisierung des Anwendungsservers fest.

   Das Benutzerkonto wird verwendet, um die Dateien im Anwendungsverzeichnis anzulegen oder auszutauschen.

   - Wenn Sie die Benutzerkonto, unter welcher der Anwendungspool ausgeführt wird, für die Aktualisierungen nutzen möchten, setzen Sie die Option **Nutze die IIS-Berechtigungen für Aktualisierungen**.
   - Wenn Sie ein anderes Benutzerkonto verwendet möchten, setzen Sie die Option **Nutze ein spezielles Konto für die Aktualisierungen** und geben Sie die Domäne, den Benutzernamen und das Kennwort des Benutzers an.

7. Auf der Seite **Installation läuft** werden die einzelnen Installationsschritte angezeigt. Nachdem der Installationsvorgang abgeschlossen wurde, klicken Sie **Weiter**.

   Der Web Installer generiert die Webanwendung und die entsprechenden Konfigurationsdateien (`web.config`) zu jedem Verzeichnis.

8. Auf der letzten Seite klicken Sie **Fertig**, um das Programm zu beenden.

Konfigurieren des SPML Webservice

Die Konfiguration des SPML Webservice befindet sich im Installationsverzeichnis des Webservice in der XML-Datei `web.config`. Diese Datei können Sie mit einem beliebigen Texteditor bearbeiten.

HINWEIS:

- Nach der Standardinstallation passen Sie auf jeden Fall die Option `AuthenticationString` in der Sektion `configuration\application` an.
- Die Schemadateien `QOIM_Schema.xsd` und `QOIM_SpmlTargetSchema.xsd` erzeugen Sie mit dem Schemaeditor des Designers. Weitere Informationen finden Sie unter Schemadateien erstellen auf Seite 543. Legen Sie die Schemadateien im Verzeichnis des SPML Webservice abund geben Sie den Ablageort der Schemadateien in der Konfigurationsdatei über die Optionen `ProviderSchema` und `SpmlTargetSchema` bekannt. Standardmäßig erfolgt die Ablage im Verzeichnis `Schemas` des Installationsverzeichnisses.
- Soll der SPML Webservice nur über eine verschlüsselte SSL Verbindung erreichbar sein, so konfigurieren Sie dies in den Einstellungen des Internet Information Services für die jeweilige Applikation. Informationen dazu entnehmen Sie der Dokumentation Ihres Internet Information Services.

Tabelle 192: Konfigurierbare Optionen der Konfigurationsdatei Web.config

Sektion	Option	Gültige Werte	Beschreibung
connectionString			Verbindungsparameter zur Datenbank.
runtimedirs	key="Cache"	value = "<Pfad>"	Verzeichnis zur Ablage der Cache Verzeichnisse. Standard: `value="C:\inetpub\wwwroot\<web service name>\App_Data\Cache\DB"`
	key="AssemblyCache"	value = "<Pfad>"	Verzeichnis zur Ablage der Cache Verzeichnisse. Standard: `value="C:\inetpub\wwwroot\<web service name>\App_Data\Cache\Assemblies"`
application	key = "ProviderSchema"	value = "<Pfad>"	Relativer Pfad zum SPML Schema (`QOIM_Schema.xsd`). Das Schema definiert alle Objekte und Eigenschaften die über den Webservice verwaltet werden können. Die Datei wird vom Designer

Sektion	Option	Gültige Werte	Beschreibung
			erzeugt. Alle Anfragen gegen den Webservice werden gegen diese Datei verifiziert. Standard: `value=".\Schemas\QOIM_Schema.xsd"`
	key = "SpmlTargetSchema"	value = "<Pfad>"	Relativer Pfad zum SPML Targetschema (`QOIM_SpmlTargetSchema.xsd`). Das Schema definiert die Antwort auf den `listTargetsRequest`. Die Datei wird vom Designer erzeugt. Standard: `value=".\Schemas\QOIM_SpmlTargetSchema.xsd"`
	key = "MaxConnections"	value = "<Integer>"	Anzahl der möglichen gleichzeitigen Verbindungen. Anzahl der Clients. Standard: `value ="1"`
	key = "AuthenticationString"	value="Module=;User=; Password="	Authentifizierungsmodul und Anmeldeinformationen mit denen die Anmeldung und alle Operationen des Webservice durchgeführt werden. Standard: `value="Module=DialogUser;User=DIALOGUSER;Password=PASSWORD"`
	key = "DebugMode"	value = "True" value = "False"	Erweiterte Ausgaben im Protokoll. Standard: `value="true"`
	key = "LogAllRequests"	value = "True" value = "False"	Anfragen werden immer protokolliert. Standard: `value="false"`
	key = "LogDirectory"	value = "<Pfad>"	Verzeichnis für das Protokoll. Standard: `value=".\Log"`
	key = "MaxSearchResults"	value = "<Integer>"	Maximal zulässige Anzahl an Suchergebnissen für die Iteration. Standard: `value="10000"`
	key = "ConcurrentSearchResponseObjects"	value = "<Integer>"	Anzahl der Objekte die pro Iteration über das Suchergebnis an den Client zurückgegeben werden sollen.

Sektion	Option	Gültige Werte	Beschreibung
			Standard: `value="10"`
	key = "CheckForUnusedResultsInterval"	value = "<Integer>"	Aller wie viel Sekunden wird nach verwaisten Suchergebnissen gescannt. Standard: `value="30"`
	key = "KeepSearchResultsFor"	value = "<Integer>"	Wie viel Sekunden hat der Client für die nächste Iteration der Ergebnismenge bevor die Ergebnismenge verworfen wird. Standard: `value="60"`

HINWEIS: Um die Verbindungsparameter (`ConnectionString`) zu verschlüsseln, verwenden Sie `aspnet_regiis.exe`.

Aufrufbeispiel:

```
c:\windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis.exe -pe
"application" -app "/<web service name>" -prov
"DataProtectionConfigurationProvider"
```

```
wobei: <web service name> = Pfad des Webservice auf dem Internet
Information Services
```

SPML Webservice deinstallieren

Um eine Webanwendung zu deinstallieren

1. Starten Sie die Datei `autorun.exe` aus dem Basisverzeichnis des One Identity Manager Installationsmediums.
2. Auf der Startseite des Installationsassistenten:
   a. Wechseln Sie zum Tabreiter **Installation**.
   b. Im Bereich **Web-basierte Komponenten** klicken Sie **Installieren**.

   Der Web Installer wird gestartet.
3. Auf der Startseite des Web Installer klicken Sie **Deinstallieren einer Webanwendung** und klicken Sie **Weiter**.
4. Auf der Seite **Deinstallieren einer Webanwendung** doppelklicken Sie die Webanwendung, die Sie entfernen möchten.

   Vor der Anwendung wird das Symbol angezeigt.
5. Klicken Sie **Weiter**.

6. Auf der Seite **Datenbankverbindung** wählen Sie die Datenbankverbindung und das Authentifizierungsverfahren und geben Sie die entsprechenden Anmeldedaten ein.
7. Klicken Sie **Weiter**.
8. Bestätigen Sie die Sicherheitsabfrage mit **Ja**.
9. Auf der Seite **Installation läuft** werden die einzelnen Schritte zur Deinstallation angezeigt.
10. Nachdem der Installationsvorgang abgeschlossen wurde, klicken Sie **Weiter**.
11. Auf der Seite **Beenden des Assistenten** klicken Sie **Fertig**.
12. Schließen Sie das Autorun-Programm.

Konfiguration des One Identity Manager Schemas

Der SPML Webservice unterstützt das komplette One Identity Manager Schema. Um Objekte und deren Beziehungen über den SPML Webservice zu verwalten, ist es notwendig, die zu verwaltenden Objekte und Eigenschaften sowie Beziehungen im One Identity Manager Schema zu definieren. Erst wenn Objekte und Eigenschaften sowie Referenzen im One Identity Manager Schema für die Verwaltung mittels SPML definiert wurden, kann der SPML Webservice genutzt werden. Nach erfolgter Definition werden zwei Schemadateien erzeugt, die vom SPML Webservice zur Validierung benötigt werden. Die Dateien sind im entsprechenden Verzeichnis des SPML Webservice auszutauschen.

Detaillierte Informationen zum Thema

- Vorbereiten des One Identity Manager Schemas für den Export in das SPML-Schema auf Seite 542
- Schemadateien erstellen auf Seite 543

Vorbereiten des One Identity Manager Schemas für den Export in das SPML-Schema

Um Objekte mit einzelnen Eigenschaften und die Beziehungen zwischen verschiedenen Objekttypen mit dem SPML Webservice zu verwalten, kennzeichnen Sie die korrespondierenden Tabellen, Spalten sowie die Tabellenbeziehungen des One Identity Manager Schema für den Export in das SPML-Schema.

Um Objekte und deren Eigenschaften über den SPML Webservice zu verwalten

1. Wählen Sie im Designer die Kategorie**One Identity Manager Schema**.
2. Wählen Sie die Tabelle aus und starten Sie den Schemaeditor über die Aufgabe **Tabellendefinition anzeigen**.
3. Aktivieren Sie auf dem Tabreiter **Tabelle** die Option **Export für SPML-Schema**.
4. Wählen Sie im Schemaeditor die Spalte.
5. Aktivieren Sie auf dem Tabreiter **Sonstiges** die Option **Export für SPML-Schema**.

HINWEIS: Sollen Referenzen zwischen verschiedenen Objekttypen des One Identity Manager Schemas über den SPML Webservice verwaltet werden, so müssen beide beteiligten Objekte für die Verwaltung über SPML markiert sein, das bedeutet beide Tabelle sind mit der Option **Export für SPML-Schema** zu kennzeichnen.

Referenzen zwischen Objekttypen werden in One Identity Manager über Fremdschlüsselbeziehungen und über M:N Zuordnungen abgebildet.

- Für das Verwalten von Fremdschlüsselbeziehungen mittels SPML ist es ausreichend, wenn Sie die entsprechende Spalte im One Identity Manager Schema mit der Option **Export für SPML-Schema** markieren.

  HINWEIS: Beachten Sie, dass nur jeweils eine Fremdschlüsselbeziehung zwischen zwei Objekttypen per SPML verwaltet werden kann. So kann beispielsweise der Verantwortliche einer Geschäftsrolle (`Org.UID_PersonHead`) per SPML gepflegt werden, jedoch nicht gleichzeitig den stellvertretenden Verantwortlichen (`Org.UID_PersonHeadSecond`)
- Für die Konfiguration von M:N Zuordnungen für die Nutzung mittels SPML, wählen Sie die entsprechende M:N-Tabelle aus und kennzeichnen die Tabellenbeziehung mit der Option **Export für SPML-Schema**.

Verwandte Themen

- Tabellendefinitionen auf Seite 57
- Tabellenbeziehungen auf Seite 109

Schemadateien erstellen

Haben Sie alle Tabellen, Spalten und Tabellenbeziehungen, welche mittels SPML verwaltet werden sollen, entsprechend gekennzeichnet, müssen Sie die benötigten Schemadateien für den SPML Webservice erzeugen.

WICHTIG:

- Stellen Sie sicher, dass Sie vor dem Export alle Änderungen im Designer in die Hauptdatenbank übernommen haben und alle offenen Berechnungsaufträge für den DBQueue Prozessor abgearbeitet wurden.

- Ändern Sie zu einem späteren Zeitpunkt weitere SPML-relevante Einstellungen am One Identity Manager Schema, müssen Sie die Schemadateien erneut erzeugen.

Um die Schemadateien zu erstellen

1. Starten Sie im Designer in der Kategorie **One Identity Manager Schema** den Schemaeditor.
2. Wählen Sie den Menüeintrag **Schema > SPML-Schemainformationen exportieren**.
3. Bestätigen Sie die Sicherheitsabfrage mit **OK**.
4. Geben Sie im Dialog **Ordner suchen** ein Verzeichnis an, in welchem die Schemadatei erzeugt werden.
5. Klicken Sie **OK**.

   Der Export wird gestartet. Der Export kann je nach Umfang der Änderungen einige Zeit in Anspruch nehmen.
6. Klicken Sie **OK**.

Die Schemadateien `QOIM_Schema.xsd` und `QOIM_SpmlTargetSchema.xsd` legen Sie im Verzeichnis des SPML Webservice ab. Geben Sie den Ablageort der Schemadateien in der Konfigurationsdatei des SPML Webservice an. Standardmäßig erfolgt die Ablage im Verzeichnis `Schemas` des Installationsverzeichnisses.

Verwandte Themen

- Konfigurieren des SPML Webservice auf Seite 539

Testen der Funktionalität des SPML Webservice

Um die Grundfunktionalität des SPML Webservice zu testen, wird eine einfaches Testfrontend mitgeliefert. Voraussetzung für die Nutzung des Testfrontends ist eine korrekte Installation und Konfiguration des SPML Webservice. Überprüfen Sie mit einem Browser ob der SPML Webservice funktionsfähig und korrekt konfiguriert ist.

Der SPML Webservice ist über einen Browser erreichbar unter:

http://<Server>/<Anwendungsname>

https://<Server>/<Anwendungsname>

TIPP: Sie können die Statusanzeige des Webservers im Job Queue Info öffnen. Wählen Sie dazu im Job Queue Info das Menü **Ansicht > Serverstatus** und öffnen Sie auf dem Tabreiter **Webserver** die Statusanzeige des Webservers über das Kontextmenü **Im Browser öffnen**.

Detaillierte Informationen zum Thema

- Konfiguration des SPML Testfrontends auf Seite 545
- SPML Testfrontend auf Seite 545

Konfiguration des SPML Testfrontends

Diese Konfigurationseinstellung dient dazu, die spezifische URL des SPML Webservice für die Nutzung im SPML Testfrontend anzugeben.

1. Kopieren Sie die Dateien `VI.SPMLTestFrontend.exe` und `VI.SPMLTestFrontend.exe.config` vom Installationsmedium aus dem Verzeichnis `QBM\dvd\AddOn\SPML\Testfrontend` in das One Identity Manager Installationsverzeichnis.
2. Um dem Testfrontend den konfigurierten Webservice bekanntzugeben, passen Sie die Datei `VI.SPMLTestFrontend.exe.config` an.
3. Anschließend können Sie die Datei `VI.SPMLTestFrontend.exe` starten.

SPML Testfrontend

Mit dem SPML Testfrontend ist es möglich die Funktionsweise des SPML Webservices zu testen und zu analysieren. Das Frontend dient ausschließlich der Analyse des SPML Webservice und dem Testen der Funktionalität.

HINWEIS: Ein dauerhafter Einsatz des Frontends zur Steuerung des SPML Webservice ist nicht vorgesehen.

Um die Funktionen des SPML Webservices zu testen

1. Starten Sie das SPML Testfrontend über die Datei `VI.SPMLTestFrontend.exe`.
2. Wählen Sie in der Auswahlliste **Choose Request** die zu testende Funktion des SPML Webservice aus.

   Die zugehörige XML Anfrage wird im Textfeld **SPML Request (XML)** angezeigt.

   Die XML Anfrage können Sie vor dem Absenden an den SPML Webservice entsprechend bearbeiten. Die vordefinierten Ausschnitte der XML Anfrage sollten Sie immer überprüfen und dem im Zielsystem definierten Schema für die SPML Unterstützung anpassen. Die vordefinierten Ausschnitte dienen lediglich als Hilfestellung für das Formulieren von SPML konformen Anfragen.
3. Wenn die in der XML Anfrage übergebene Request-ID automatisch hochgezählt werden soll, setzen Sie die Option **Increment request IDs automatically**. Standardmäßig ist die Option deaktiviert und es wird die eingetragene Request-ID verwendet.

4. Senden Sie die Abfrage an den SPML Webservice über die Schaltfläche **Access** ab.

   Das Ergebnis wird im Textfeld **SPML Response (XML)** dargestellt.

Wird ein neues Objekt im Zielsystem angelegt, so wird dessen Schlüssel in die Auswahlliste der bekannten Objekte (**Known Objects**) übernommen. Wird eine Suche mit begrenzter Ergebnismenge durchgeführt und dadurch eventuell die Ergebnisliste nicht vollständig zurückgegeben, wird ein Iterator zurückliefert. Dieser Iterator wird der Auswahlliste für bekannte Iteratoren (**Known Iterators**) hinzugefügt. Mit diesem Iterator kann die Suche fortgesetzt werden.

Ausführliche Fehlermeldungen entnehmen Sie der Protokolldatei. Diese liegt im Verzeichnis, welches Sie über die Option `LogDirectory` in der Konfigurationsdatei des SPML Webservice festgelegt haben.

Verwandte Themen

- Konfigurieren des SPML Webservice auf Seite 539

20

Verarbeitung von DBQueue Aufträgen

Die in die DBQueue eingestellten Aufträge resultieren aus Triggerverarbeitung, Änderungen an Konfigurationsparametern, wie beispielsweise Änderung der Konfigurationsparameter zur Vererbung oder durch die Ausführung zeitgesteuerter Aufträge. Der DBQueue Prozessor verarbeitet die Aufträge aus der DBQueue. Für die parallele Ausführung der Aufträge werden durch den DBQueue Prozessor mehrere Slots verwendet.

Sie können während der Installation oder Aktualisierung einer Datenbank mit dem Configuration Wizard wählen, welcher Agenttyp die Verarbeitung der internen Aufträge übernehmen soll.

- **Database Agent Service**: Die Verarbeitung der internen Aufträge erfolgt durch den Database Agent Service. Stellen Sie sicher, dass der Database Agent Service installiert und konfiguriert ist. Der Database Agent Service wird über ein Plugin des One Identity Manager Service bereitgestellt.

  > WICHTIG: Dies ist eine EXPERIMENTELLE Funktion. Ihre Auswirkungen auf die Performance von Produktionssystemen sind noch nicht bekannt. Daher wird diese Funktion noch nicht vom Support abgedeckt. Sie können die Funktion jedoch gern ausprobieren (vorzugsweise auf nicht produktiven Systemen) und Ihre Anmerkungen an OneIM.Beta@oneidentity.com senden.

- **SQL Server Agent**: Die Verarbeitung der internen Aufträge erfolgt durch den SQL Server Agent. Es werden die benötigten Datenbankschedules erzeugt. Stellen Sie sicher, dass der SQL Server Agent gestartet ist.

Detaillierte Informationen zum Thema

- Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen auf Seite 548
- Benachrichtigungsverhalten für die DBQueue Prozessor Initialisierung konfigurieren auf Seite 549
- Erneutes Initialisieren des DBQueue Prozessor auf Seite 550
- Mengenverarbeitung im DBQueue Prozessor auf Seite 550

- Verarbeitung der DBQueue Prozessor Aufträge durch den SQL Server Agent auf Seite 551
- Verarbeitung der DBQueue Prozessor Aufträge durch den Database Agent Service auf Seite 553
- Steuerung der Verarbeitung von DBQueue Prozessor Aufträgen auf Seite 554
- Verarbeiten der DBQueue Prozessor Aufträge auf Seite 555
- Kommunikation des zentralen Dispatchers mit den einzelnen Slots auf Seite 557

Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen

Über die Staging-Ebene der One Identity Manager-Datenbank legen Sie fest, ob es sich um eine Testdatenbank, Entwicklungsdatenbank oder produktive Datenbank handelt. Über die Staging-Ebene werden einige Datenbankeinstellungen gesteuert.

Wenn Sie die Staging-Ebene der Datenbank ändern, werden die folgenden Einstellungen konfiguriert.

Tabelle 193: Standardeinstellungen für Entwicklungsumgebung, Testumgebung und Produktivumgebung

Einstellung	Entwicklungsumgebung	Testumgebung	Produktivumgebung
Maximale Laufzeit DBQueue Prozessor	20 Minuten	40 Minuten	120 Minuten
Maximale Anzahl der Slots für DBQueue Prozessor	5	7	Maximale Anzahl der Slots laut Hardwarekonfiguration

Die Standardeinstellungen für den DBQueue Prozessor sind für einen Normalbetrieb ausgelegt und müssen in der Regel nicht angepasst werden.

Wenn mehrere Datenbanken in einer verwalteten Instanz in Azure SQL-Datenbank betrieben werden, können Sie die Anzahl der Slots fest vorgeben. Passen Sie im Designer den folgenden Konfigurationsparameter an.

- **QBM | DBServerAgent | CountSlotAgents**: Genaue Anzahl der Slots. Ist der Konfigurationsparameter aktiviert, wird immer die Anzahl der angegebenen Slot

eingerichtet. Es erfolgt keine interne Berechnung der Slotanzahl anhand der Hardwarekonfiguration. Die Änderung der Konfiguration des Servers hat keinen Einfluss. Es wird der Wert **15** empfohlen.

> HINWEIS: Für den Einsatz einer Datenbank auf einem SQL Server wird der Konfigurationsparameter nicht empfohlen. Für den Einsatz einer Datenbank auf einem SQL Server hat sich die Ermittlung der Slots über die Hardwarekonfiguration bewährt.

Für Testumgebungen und Entwicklungsumgebungen sind die Konfigurationseinstellungen reduziert, da sich mehrere Datenbanken auf einem Server befinden können. Müssen aus Performancegründen die Einstellungen für Testumgebungen und Entwicklungsumgebungen angepasst werden, ändern Sie im Designer die Einstellungen der folgenden Konfigurationsparameter an.

- **QBM | DBQueue | CountSlotsMax**: Anzahl der maximal zu verwendenden Slots.

  Nutzen Sie den Konfigurationsparameter um die Anzahl der Slots bei Bedarf zu reduzieren. Werte kleiner als **5** sind nicht zulässig.

  Ausnahme: Für die Nutzung der maximal verfügbaren Slots laut Hardwarekonfiguration geben Sie den Wert **0** an.

- **QBM | DBQueue | KeepAlive**: Maximale Laufzeit des zentralen Dispatchers. Nach Ablauf der Laufzeit werden die Aufträge aktuell verwendeter Slots noch abgearbeitet. Anschließend werden die Slots gestoppt und der zentrale Dispatcher beendet.

  Der minimal zulässige Wert für die Laufzeit ist **5 Minuten**, der maximal zulässige Wert ist **720 Minuten**.

Verwandte Themen

- Staging-Ebene der Datenbank ändern auf Seite 40

Benachrichtigungsverhalten für die DBQueue Prozessor Initialisierung konfigurieren

Treten bei der Initialisierung des DBQueue Prozessors Fehler auf, werden die Meldungen im Anwendungsprotokoll protokolliert. Das Anwendungsprotokoll können Sie beispielsweise über die Ereignisanzeige in der Microsoft Management Console anzeigen.

Über den Konfigurationsparameter **QBM | DBServerAgent | CreateNotification** können Sie konfigurieren, in welchen Fällen Meldungen in das Anwendungsprotokoll geschrieben werden. Passen Sie den Konfigurationsparameter bei Bedarf im Designer an.

Zulässige Werte sind:

- **0**: Es erfolgt keine Protokollierung.
- **1**: Es werden nur Erfolgsmeldungen protokolliert.
- **2**: Es werden nur Fehlermeldungen protokolliert (Standard).
- **3**: Es werden alle Meldungen protokolliert.

Erneutes Initialisieren des DBQueue Prozessor

WICHTIG: Verwenden Sie für die Ausführung der SQL Abfragen den Benutzer, den Sie auch für die Migration der Datenbank verwenden.

- Die Prozedur `QBM_PDBQueuePrepare` ist einmalig manuell auszuführen, bei Erweiterung der Serverhardware und bei Erstellung kundenspezifischer DBQueue Prozessor Aufträge.
- Bei Einrichtung einer Referenzdatenbank für eine Test- oder Entwicklungsumgebung sind die Prozeduren `QBM_PDBQueuePrepare` und `QBM_PWatchDogPrepare` einmalig auszuführen.

  Führen Sie mit einem geeigneten Programm zur Ausführung von SQL Abfragen einmalig die folgenden Prozeduren in der Referenzdatenbank aus.

  ```
  exec QBM_PWatchDogPrepare
  ```

  ```
  exec QBM_PDBQueuePrepare 0,1
  ```

Mengenverarbeitung im DBQueue Prozessor

Tabelle 194: Konfigurationsparameter für die Mengenverarbeitung im DBQueue Prozessor

Konfigurationsparameter	Bedeutung
QBM | DBQueue | DefaultRuntime	Der Konfigurationsparameter legt fest, wie groß das Laufintervall des DBQueue Prozessors sein soll. Der Standardwert ist **90** Sekunden.
QBM | DBQueue | ChangeLimitMin	Der Konfigurationsparameter definiert den unteren Grenzwert für Änderungen (Einfügen, Ändern oder Löschen) innerhalb einer einzelnen Operation. Der Standardwert ist **3000**.
QBM | DBQueue |	Der Konfigurationsparameter definiert den oberen

Konfigurationsparameter	Bedeutung
ChangeLimitMax	Grenzwert für Änderungen (Einfügen, Ändern oder Löschen) innerhalb einer einzelnen Operation. Der Standardwert ist **50000**.

Um die Gesamtzeit der Verarbeitung der DBQueue Aufträge zu reduzieren, sind einige der Prozeduren des DBQueue Prozessors für die Mengenverarbeitung gekennzeichnet. Sind mehrere Einträge für einen derart gekennzeichneten Auftrag in der DBQueue vorhanden, dann schaltet der DBQueue Prozessor von Einzelverarbeitung auf Mengenverarbeitung.

Es ist ein Mechanismus implementiert, anhand dessen entschieden wird, ob die Umstellung auf Mengenverarbeitung gegenüber der Einzelverarbeitung zu einer Zeitersparnis führen würde. Dazu werden zunächst 25 Einzelverarbeitungen eines Auftrages ausgeführt und die Verarbeitungszeiten ermittelt. Alle weiteren Einträge eines Auftrages, werden über Mengenverarbeitung abgearbeitet und die minimale und maximale Ladezeit für eine rentable Mengenverarbeitung bestimmt. Die Aktualisierung der Ladezeiten erfolgt durch ein selbstoptimierendes Berechnungsverfahren. Das eingesetzte Verfahren hat zur Folge, dass sich der DBQueue Prozessor erst einschwingen muss, insbesondere nach einer initialen Schemainstallation oder nach Systemänderungen wie beispielsweise Speicherzuwachs des Datenbankservers.

Über den Konfigurationsparameter **QBM | DBQueue | DefaultRuntime** können Sie festlegen, wie groß das Laufintervall des DBQueue Prozessors sein soll. Der Standardwert ist **90** Sekunden. Dies entspricht dem Zeitraum, für den über das Berechnungsverfahren eine gute Auslastung erzielt wird.

Als Schutz vor Überladung bei zu großen Datenmengen können Grenzwerte für die zu bearbeitende Ergebnismenge definiert werden. Die Steuerung erfolgt über die Konfigurationsparameter **QBM | DBQueue | ChangeLimitMin** und **QBM | DBQueue | ChangeLimitMax**.

Verarbeitung der DBQueue Prozessor Aufträge durch den SQL Server Agent

WICHTIG: Ändern oder Löschen Sie keinen der eingerichteten Datenbankschedules, da dies zu unerwarteten Fehlerzuständen führen kann.

Für die Verarbeitung der internen Aufträge durch den SQL Server Agent, wird der DBQueue Prozessor während der Schemainstallation initialisiert. Während der Initialisierungsphase werden die folgenden Datenbankschedules erzeugt:

- **QBM_PWatchDog on <database>**

  Dieser Datenbankschedule übernimmt mehrere Funktionen im One Identity Manager.

- Er prüft, ob der zentrale Dispatcher des DBQueue Prozessor aktiv ist und startet diesen neu.
- Er übernimmt die Prüfung und den Start der Zeitpläne.
- Er startet den Datenbankschedule zum Entfernen der bereits verarbeiteten DBQueue Aufträge.
- Er startet den Datenbankschedule für Wartungsarbeiten.
- Er prüft in regelmäßigen Abständen, ob der Einzelbenutzermodus der Datenbank noch erforderlich ist und setzt gegebenenfalls die Angabe zurück.

Dieser Datenbankschedule verfügt über einen aktiven Zeitplan mit einem Intervall von 1 Minute.

- **QBM_PDBQueueProcess_Main on <database>**

  Dieser Datenbankschedule ist der zentrale Dispatcher des DBQueue Prozessor. Der zentrale Dispatcher übernimmt die Steuerung der Verarbeitung und verteilt die Aufträge der DBQueue an die einzelnen Slots. Bei jedem Lauf des zentralen Dispatchers wird die Anzahl der aktuell verfügbaren Slots ermittelt, die für den aktuellen Lauf verwendet werden können. Der zentrale Dispatcher startet die Datenbankschedules der aktuell verfügbaren Slots einmalig.

  Es wird maximal ein Datenbankschedule für den zentralen Dispatcher gestartet. Der Datenbankschedule des zentralen Dispatchers besitzt keinen aktiven Zeitplan, sondern wird durch den Datenbankschedule **QBM_PWatchDog on <database>** gestartet.

- **QBM_PDBQueueProcess<SlotNumber> on <database>**

  Während der Initialisierungsphase des DBQueue Prozessor wird die Anzahl der maximal verfügbaren Slots ermittelt.

  - Für produktive Datenbanken ist die Anzahl der maximalen Slots abhängig von der Anzahl der Prozessoren auf dem Datenbankserver.
  - Für Testumgebungen und Entwicklungsumgebungen ist die maximale Anzahl der Slots reduziert.
  - Ist der Konfigurationsparameter **QBM | DBServerAgent | CountSlotAgents** aktiviert, wird immer die genaue Anzahl der angegebenen Slot eingerichtet. Es erfolgt keine interne Berechnung der Slotanzahl anhand der Hardwarekonfiguration.

  Weitere Informationen finden Sie unter Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen auf Seite 548.

  Für jeden Slot wird ein zugehöriger Datenbankschedule **QBM_PDBQueueProcess<SlotNumber> on <database>** eingerichtet. Jeder Datenbankschedule wird mit einem Prozess eingerichtet, der die Aufträge der DBQueue für genau diesen Slot ausführt. Die Datenbankschedules der einzelnen Slots besitzen keine aktiven Zeitpläne. Der Start erfolgt über den zentralen Dispatcher.

- **QBM_PDBQueueProcess_Del on <database>**

Dieser Datenbankschedule entfernt die verarbeiteten DBQueue Aufträge. Der Datenbankschedule besitzt keinen aktiven Zeitplan, sondern wird durch den Datenbankschedule **QBM_PWatchDog on <database>** gestartet.

- **QBM_PDBQueueProcess_Mnt on <database>**

  Dieser Datenbankschedule verarbeitet die Wartungsaufträge. Die Wartungsaufträge übergeben dem Datenbankschedule ihre Aufgaben, statt sie selbst durchzuführen. Damit muss an der Zeitsteuerung für die Wartungsaufträge nichts geändert werden. Der Datenbankschedule besitzt keinen aktiven Zeitplan, sondern wird durch den Datenbankschedule **QBM_PWatchDog on <database>** gestartet. Ausführliche Informationen zu Wartungsaufträgen finden Sie im *One Identity Manager Administrationshandbuch für betriebsunterstützende Aufgaben*.

Verwandte Themen

- Steuerung der Verarbeitung von DBQueue Prozessor Aufträgen auf Seite 554
- Verarbeiten der DBQueue Prozessor Aufträge auf Seite 555
- Kommunikation des zentralen Dispatchers mit den einzelnen Slots auf Seite 557
- Verarbeitung der DBQueue Prozessor Aufträge durch den Database Agent Service auf Seite 553

Verarbeitung der DBQueue Prozessor Aufträge durch den Database Agent Service

Für die Verarbeitung der internen Aufträge durch den Database Agent Service, stellen Sie sicher, dass der Database Agent Service installiert und konfiguriert ist. Der Database Agent Service wird über ein Plugin des One Identity Manager Service bereitgestellt. Das Plugin sollte auf dem Jobserver konfiguriert sein, der die Serverfunktion **Aktualisierungsserver** übernimmt. Für die Datenbankverbindung im Jobprovider muss ein administrativer Benutzer verwendet werden.

Der Database Agent Service übernimmt folgende Aufgaben:

- Steuerung der Verarbeitung und Verteilung der Aufträge der DBQueue an die einzelnen Slots (zentraler Dispatcher)
- Prüfen und Starten von zeitgesteuerten Aufträgen
- Entfernen von bereits verarbeiteten DBQueue Aufträgen
- Starten von Wartungsarbeiten
- Befüllen und Aktualisieren der Jobqueue Übersicht
- Archivieren oder Löschen von verarbeiteten Prozessen aus der Jobqueue
- Archivieren oder Löschen von Aufzeichnungen aus der Prozessverarbeitung

HINWEIS: Wenn der Database Agent Service nicht arbeitet, wird in allen Administrationswerkzeugen eine Meldung in der Statuszeile angezeigt. Um diese Meldung zu sehen, benötigen die Benutzer mindestens die Berechtigungsebene für Konfigurationsbenutzer.

Verwandte Themen

- Steuerung der Verarbeitung von DBQueue Prozessor Aufträgen auf Seite 554
- Verarbeiten der DBQueue Prozessor Aufträge auf Seite 555
- Kommunikation des zentralen Dispatchers mit den einzelnen Slots auf Seite 557
- DatabaseAgentPlugin auf Seite 345
- Verarbeitung der DBQueue Prozessor Aufträge durch den SQL Server Agent auf Seite 551

Steuerung der Verarbeitung von DBQueue Prozessor Aufträgen

Der zentrale Dispatcher übernimmt die Steuerung der Verarbeitung und verteilt die Aufträge der DBQueue an die einzelnen Slots.

Es wird zunächst die Anzahl der aktuell verfügbaren Slots ermittelt, die verwendet werden können. Je mehr Last auf dem Datenbankserver vorhanden ist, desto weniger Slots können aktuell verwendet werden, mindestens werden jedoch 5 Slots verwendet.

Die Anzahl der aktuell verfügbaren Slots ergibt sich aus :

```
Anzahl aktuell verfügbarer Slots = Anzahl der maximal verfügbaren Slots - Betrag
für alle Prozesse der eigenen Datenbank - Betrag für Prozesse auf anderen
Datenbanken des Servers
```

HINWEIS: Die Anzahl der verfügbaren Slots ist weiterhin über den Konfigurationsparameter **QBM | DBQueue | CountSlotsMax** beeinflussbar. Sollte die Anzahl der Anzahl der verfügbaren Slots laut Berechnung größer sein, als der Wert im Konfigurationsparameter, wird der Wert des Konfigurationsparameters verwendet. Weitere Informationen finden Sie unter Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen auf Seite 548.

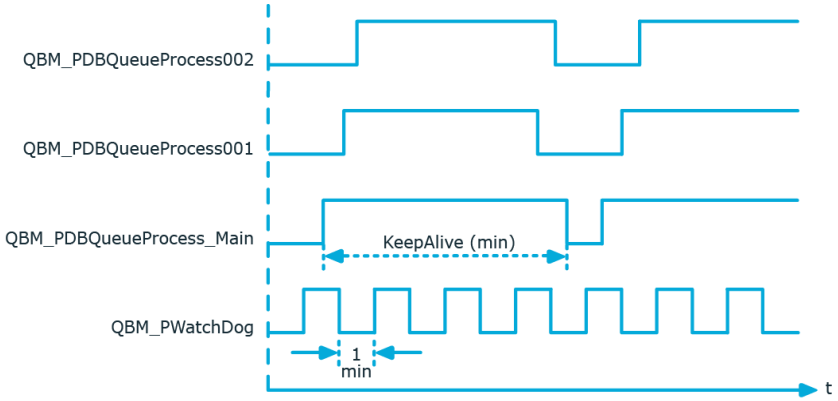
Der zentrale Dispatcher startet die aktuell verfügbaren Slots einmalig. Für jeden Slot ist ein Prozess eingerichtet, der die Aufträge für genau diesen Slot ausführt.

Sobald Aufträge in die DBQueue eingetragen werden, wird der zentrale Dispatcher benachrichtigt. Der zentrale Dispatcher verteilt die Aufträge in die einzelnen Slots und benachrichtigt die Prozesse der Slots, dass Aufträge zur Verarbeitung anstehen. Jeder Prozess verarbeitet die Aufträge, die für seinen Slot eingestellt werden. Nach Beendigung seines Auftrags sendet jeder Prozess eine Benachrichtigung an den zentralen Dispatcher und wartet auf neue Aufträge.

Der zentrale Dispatcher prüft in definierten Abständen, ob die Slots noch aktiv sind und verteilt neue Aufträge an die Slots. Sind keine Aufträge in der DBQueue vorhanden, geht der zentrale Dispatcher in den Wartezustand und wartet auf die Benachrichtigung über neue Aufträge.

Nach Ablauf der Laufzeit werden die Aufträge aktuell verwendeter Slots noch abgearbeitet. Anschließend werden die Slots gestoppt. Weitere Informationen finden Sie unter Kommunikation des zentralen Dispatchers mit den einzelnen Slots auf Seite 557.

Abbildung 39: Steuerung der Verarbeitung über SQL Server Agent



Verarbeiten der DBQueue Prozessor Aufträge

Der zentrale Dispatcher ermittelt die Einträge der DBQueue (Tabelle `DialogDBQueue`) und verschiebt die Aufträge in die Tabelle `QBMDBQueueCurrent` mit der Zuordnung Auftrag pro Slot.

Beispiel für Einträge in den Tabellen DialogDBQueue und QBMDBQueueCurrent

Tabelle 195: Einträge in Tabelle DialogDBQueue (Auszug)

Name des Auftrags	Objekt
OrgRoot	A
OrgRoot	B
ADSAccountInADSGroup	X
ADSAccountInADSGroup	Y
ADSAccountInADSGroup	Z

Tabelle 196: Einträge in Tabelle QBMDBQueueCurrent (Auszug)

Slotnummer	Name des Auftrags	Objekt
001	OrgRoot	A
001	OrgRoot	B
002	ADSAccountInADSGroup	X
002	ADSAccountInADSGroup	Y
002	ADSAccountInADSGroup	Z

Jeder Prozess verarbeitet die Aufträge, die für seinen Slot in die Tabelle `QBMDBQueueCurrent` eingestellt werden. Folgeaufträge, die aus der Verarbeitung resultieren, werden in die Tabelle `DialogDBQueue` eingestellt.

Wenn ein Prozess seine Aufträge abgearbeitet hat und keine weiteren Aufträge anstehen, wird die Slotnummer in der Tabelle `QBMDBQueueCurrent` durch den Prozess selbst auf **0** gesetzt. Der Eintrag verbleibt zunächst in der Tabelle `QBMDBQueueCurrent`, wird jedoch nicht mehr beachtet (da Slot 0 kein aktiver Slot ist).

In regelmäßigen Abständen werden alle Einträge mit Slotnummer 0 aus der Tabelle `QBMDBQueueCurrent` gelöscht.

Tabelle 197: Bedeutung der Slotnummern in Tabelle QBMDBQueueCurrent

Slotnummer	Bedeutung
001 - n	Nummer des Slots, in dem ein Auftrag zu verarbeiten ist.
0	Zustand nach ordnungsgemäßer Erledigung des Auftrags.
-1	Bei der Verarbeitung des Auftrags ist ein Fehler aufgetreten oder Verarbeitung wird zurückgestellt, zum Beispiel wegen laufender Synchronisation. Der zentrale Dispatcher reaktiviert diesen Auftrag. HINWEIS: Das Zurückstellen von DBQueue Aufträgen wird im Systemprotokoll aufgezeichnet.
-2	Bei der Verarbeitung des Auftrags ist ein Fehler aufgetreten oder Verarbeitung wird zurückgestellt, zum Beispiel wegen Blockierungen. Der zentrale Dispatcher reaktiviert diesen Auftrag.
-3	Bei der Verarbeitung des Auftrags ist ein Fehler aufgetreten oder Verarbeitung wird zurückgestellt, zum Beispiel wegen noch vorhandener Einträge in der Jobqueue. Der zentrale Dispatcher reaktiviert diesen Auftrag.

Nutzung des DBQueue Puffers

Um Blockaden bei der Verarbeitung der Aufträge in der DBQueue durch langlaufende Aktionen, wie beispielsweise Synchronisationen, zu verhindern, wird ein DBQueue Puffer (Tabelle `QBMDBQueuePond`) genutzt. Synchronisationen schreiben ihre DBQueue Prozessor Aufträge zunächst in die Tabelle `QBMDBQueuePond`. Nach Abschluss der Synchronisation

werden die Einträge aus der Tabelle `QBMDBQueuePond` in die Tabelle `DialogDBQueue` verschoben.

Sollte eine langlaufende Aktion über einen definierten Zeitraum keine weiteren Einträge im DBQueue Puffer erzeugen, weil beispielsweise eine Synchronisation nicht korrekt beendet wurde, werden verbliebene Einträge der Tabelle `QBMDBQueuePond` in die Tabelle `DialogDBQueue` verschoben. Der Zeitraum ist im Konfigurationsparameter **QBM | DBQueue | BufferTimeout** festgelegt (Standard: **120** Minuten). Die Übernahme der Einträge erfolgt innerhalb der täglichen Wartungsaufträge.

Kommunikation des zentralen Dispatchers mit den einzelnen Slots

Die Kommunikation des zentralen Dispatchers mit den einzelnen Slots erfolgt über die Tabelle `QBMDBQueueSlot`. Die Anzahl der maximal verfügbaren Slots wird während der Initialisierung des DBQueue Prozessor ermittelt. Pro Slot wird ein Eintrag in der Tabelle `QBMDBQueueSlot` erzeugt. Die Tabelle enthält Informationen über die einzelnen Slots und deren Status sowie die momentan laufenden Aufträge.

Tabelle 198: Bedeutung des Status in Tabelle QBMDBQueueSlot

Status	Bedeutung
0	Keine Aktivität vorhanden. Initialzustand (gesetzt durch Initialisierung) oder Endzustand (gesetzt durch Prozess).
1	Der Prozess wird zur Arbeit aufgefordert, beispielsweise zur Vorbereitung zentraler temporärer Tabellen.
2	Betriebsbereit. Der Prozess hat die Arbeit aufgenommen, aber es sind momentan keine Aufträge vorhanden. Das ist der Zustand in dem Aufträge eingestellt werden können.
3	Übergabe an die Tabelle `QBMDBQueueCurrent`. Der Prozess hat die Aufträge zur Verarbeitung bekommen und soll tätig werden.
4	Der Prozess hat die Aufträge erkannt und übernommen.
5	Der Prozess verarbeitet die Aufträge.
-1	Der Prozess wurde zur Beendigung aufgefordert, Stopp-Verhalten nach maximaler Laufzeit oder Abbruch bei Prozessfehlern.

Stopp-Verhalten nach Ablauf der maximalen Laufzeit

Nach Ablauf der maximalen Laufzeit werden die Aufträge aktuell verwendeter Slots in der Tabelle `QBMDBQueueCurrent` noch abgearbeitet. Es werden keine neuen Aufträge aus der Tabelle `DialogDBQueue` übernommen. In der Tabelle `QBMDBQueueSlot` werden alle Slots mit einem Slotstatus **2** auf den Status **-1** gesetzt. Die Prozesse werden damit zur Beendigung

aufgefordert und beenden sich selbst. Der zentrale Dispatcher prüft, ob alle Prozesse beendet sind.

Verwandte Themen

- Beispiel für die Kommunikation während der Verarbeitung auf Seite 558
- Konfiguration des DBQueue Prozessor für Test-, Entwicklungs- oder Produktivumgebungen auf Seite 548

Beispiel für die Kommunikation während der Verarbeitung

Nachfolgendes Beispiel zeigt die Einträge in der Tabelle `QBMDBQueueSlot` während der Verarbeitung.

- Initialisierung der Slots.

Slotnummer	Status	Name des Auftrags
001	0	

- Start der Prozesse durch den zentralen Dispatcher.

Slotnummer	Status	Name des Auftrags
001	1	

- Der Prozess ist betriebsbereit. Die Vorbereitungen, wie beispielsweise Vorbereitung temporärer Tabelle, sind abgeschlossen. Es erfolgt die regelmäßige Prüfung des Slotstatus.

Slotnummer	Status	Name des Auftrags
001	2	

- Der zentrale Dispatcher verteilt die Aufträge. Der zentrale Dispatcher prüft die Betriebsbereitschaft der Slots und stellt die Aufträge der Tabelle `DialogDBQueue` in die Tabelle `QBMDBQueueCurrent` mit Slotnummer ein. Nach Übergabe an die Tabelle `QBMDBQueueCurrent` wird der Status der einzelnen Slots aktualisiert.

Slotnummer	Status	Name des Auftrags
001	3	OrgRoot

- Der Prozess hat anhand des Status einen Auftrag erkannt, startet die Verarbeitung und aktualisiert den Status seines Slots.

Slotnummer	Status	Name des Auftrags
001	4	OrgRoot

- Der Prozess hat die Verarbeitung abgeschlossen und setzt Slotnummer in der Tabelle `DialogDBQueueCurrent` auf **0**. Der Prozess ändert den Status seines Slots auf Betriebsbereit.

Slotnummer	Status	Name des Auftrags
001	2	

Anhang A

Konfigurationsdateien des One Identity Manager Service

Die Konfiguration des One Identity Manager Service erfolgt über eine Konfigurationsdatei. Die Konfigurationsdatei muss im gleichen Verzeichnis wie die `viNetworkService.exe` liegen. Es werden zwei Konfigurationsdateien unterstützt.

Detaillierte Informationen zum Thema

- Jobservice.cfg auf Seite 560
- viNetworkService.exe.config auf Seite 562

Jobservice.cfg

Die Datei `Jobservice.cfg` ist eine XML-Konfigurationsdatei im One Identity Manager eigenen einfacheren Format. Vorteil dieser Datei ist, dass ein Nachladen im laufenden Betrieb unterstützt wird. Für die verschiedenen Module innerhalb des One Identity Manager Service gibt es jeweils eine eigene Konfigurationssektion innerhalb der Datei.

> HINWEIS: Beachten Sie die Groß- und Kleinschreibung. Sowohl die Sektionen als auch der Namen der Werte müssen in Kleinbuchstaben geschrieben werden.

Die Wurzel in der XML-Datei heißt immer `configuration`. In der Sektion `category` wird jeweils ein Modul der Konfigurationsdatei und seine Werte definiert werden. Derzeit unterstützt das Programm nur den Sektionentyp `System.Configuration.NameValueSectionHandler`.

```
<configuration>
    <category name="serviceconfiguration">
        <value
        name="jobprovider">VI.JobService.MSSqlJobProvider,jobservice</value>
        <value name="HttpPort">1180</value>
        <value name="logwriter">VI.JobService.FileLogWriter,jobservice</value>
    </category>
```

```
</configuration>
```

Beispiel

Einfache Konfiguration mit:

- direkter Anbindung an SQL Server
- nur eine Jobdestination (JobProcessor)

```
<configuration>
    <category name="serviceconfiguration">
        <value
        name="jobprovider">VI.JobService.MSSqlJobProvider,jobservice</value>
        <value name="logwriter">VI.JobService.FileLogWriter,jobservice</value>
    </category>
    <category name="sqlprovider">
        <value name="connectstring">User ID=sa;initial Catalog=<Database>;Data
        Source=<SQL-Server>;Password=<Password></value>
    </category>
    <category name="filelogwriter">
        <value name="loglifetime">0.01:00:00</value>
        <value name="logseverity">Info</value>
    </category>
    <category name="dispatcher" />
    <category name="jobdestinations">
        <value
        name="queuex">VI.JobService.JobServiceDestination,jobservice</value>
    </category>
    <category name="queuex">
        <value name="queue">\%COMPUTERNAME%</value>
    </category>
</configuration>
```

Verwandte Themen

- viNetworkService.exe.config auf Seite 562

viNetworkService.exe.config

Die Datei `viNetworkService.exe.config` ist die Standard-Konfigurationsdatei für .NET-exe's und hat das dafür vorgegebene Format. Für die verschiedenen Module innerhalb des One Identity Manager Service gibt es jeweils eine eigene Konfigurationssektion innerhalb der Datei.

HINWEIS: Beachten Sie die Groß- und Kleinschreibung.

Die Wurzel in der XML-Datei heißt immer `configuration`. In der immer vorhandenen Sektion `configSections` müssen alle weiteren Sektionen der Konfigurationsdatei und deren Typ definiert werden. Derzeit unterstützt das Programm nur den Sektionentyp `System.Configuration.NameValueSectionHandler`.

```
<configuration>
    <configSections>
        <section name="sectionname"
        type="System.Configuration.NameValueSectionHandler" />
    </configSections>
    <sectionname>
        ...
    </sectionname>
</configuration>
```

Beispiel

Einfache Konfiguration mit:

- direkter Anbindung an SQL Server
- nur eine Jobdestination (JobProcessor)

```
<configuration>
    <configSections>
        <section name="serviceconfiguration"
        type="System.Configuration.NameValueSectionHandler" />
        <section name="sqlprovider"
        type="System.Configuration.NameValueSectionHandler" />
        <section name="filelogwriter"
        type="System.Configuration.NameValueSectionHandler" />
        <section name="dispatcher"
        type="System.Configuration.NameValueSectionHandler" />
        <section name="jobdestinations"
        type="System.Configuration.NameValueSectionHandler" />
```

```
        <section name="queuex"
        type="System.Configuration.NameValueSectionHandler" />
        <section name="plugins"
        type="System.Configuration.NameValueSectionHandler" />
    </configSections>
    <serviceconfiguration>
        <add key="jobprovider"
        value="VI.JobService.MSSqlJobProvider,jobservice" />
        <add key="logwriter" value="VI.JobService.FileLogWriter,jobservice" />
    </serviceconfiguration>
    <sqlprovider>
        <add key="ConnectString" value="User ID=sa;initial
        Catalog=<Database>;Data Source=<SQL-Server>;Password=<Password>" />
    </sqlprovider>
    <filelogwriter>
        <add key="LogLifeTime" value="0.01:00:00" />
        <add key="LogSeverity" value="Info" />
    </filelogwriter>
    <dispatcher />
    <jobdestinations>
        <add key="QueueX"
        value="VI.JobService.JobServiceDestination,jobservice" />
    </jobdestinations>
    <queuex>
        <add key="queue" value="\%COMPUTERNAME%" />
    </queuex>
</configuration>
```

Verwandte Themen

- Jobservice.cfg auf Seite 560

Über uns

One Identity Lösungen eliminieren die Komplexität und die zeitaufwendigen Prozesse, die häufig bei der Identity Governance, der Verwaltung privilegierter Konten und dem Zugriffsmanagement aufkommen. Unsere Lösungen fördern die Geschäftsagilität und bieten durch lokale, hybride und Cloud-Umgebungen eine Möglichkeit zur Bewältigung Ihrer Herausforderungen beim Identitäts- und Zugriffsmanagement.

Kontaktieren Sie uns

Bei Fragen zum Kauf oder anderen Anfragen, wie Lizenzierungen, Support oder Support-Erneuerungen, besuchen Sie https://www.oneidentity.com/company/contact-us.aspx.

Technische Supportressourcen

Technische Unterstützung steht für One Identity Kunden mit einem gültigen Wartungsvertrag und Kunden mit Testversionen zur Verfügung. Sie können auf das Support Portal unter https://support.oneidentity.com/ zugreifen.

Das Support Portal bietet Selbsthilfe-Tools, die Sie verwenden können, um Probleme schnell und unabhängig zu lösen, 24 Stunden am Tag, 365 Tage im Jahr. Das Support Portal ermöglicht Ihnen:

- Senden und Verwalten von Serviceanfragen
- Anzeigen von Knowledge Base Artikeln
- Anmeldung für Produktbenachrichtigungen
- Herunterladen von Software und technischer Dokumentation
- Anzeigen von Videos unter www.YouTube.com/OneIdentity
- Engagement in der One Identity Community
- Chat mit Support-Ingenieuren
- Anzeigen von Diensten, die Sie bei Ihrem Produkt unterstützen

Index

#

#LD-Notation 389

$

$-Notation 378
  Datentyp 378

*****

.CustomForms..vif 166
.Forms..vif 169

A

Anwendung
  Authentifizierungsmodul 219
  Berechtigungsgruppe 219
  einrichten 218-219
  Formular 219
  Konfigurationsdaten 219
  Menüeintrag 219
  Startmenüeintrag 219
  Systembenutzer 219
Anwendungsserver 18
  Suchindex
    aktualisieren 95
  Verbindungsinformation 299
AppServerJobProvider 323
  AuthenticationString 323
  ConnectString 323
  RequestQueueLimit 323
  ResultQueueLimit 323
Aufgabe
  deaktivieren 214
Authentifizierungsmodul
  Standard 35

B

Benutzeroberfläche
  Anwendung 218-219
  Bilder 221
  Formulare 153
  Menüführung 124
  Methodendefinition 212
  Objektdefinition 119
  Statistik 196
  Symbole 221
  Übersetzung 233
Berechtigungsgruppe
  kopieren 133
Bericht
  Anzeigename 444
  Basistabelle 444
  bearbeiten 443
  Berichtsformular 465
  Berichtsparameter 458
    Anzeigename 460
    bearbeiten 459
    Bedingung (kalk.) 464
    Bedingung (Wertabfrage) 461
    Beschreibung 460
    Datenquelle 461

Datentyp 461
Leerer Wert überschreibend 461
Liste zulässiger Werte 461
löschen 459
Mehrwertig 461
Mehrzeilig 461
Parameterdefinition 461
Parametertyp 460
Parameterwert 461
Pflichtparameter 460
Reihenfolge 460
Sichtbar 460
Skript 464
Standardwert 461
Tabellenspalte (kalk.) 464
Tabellenspalte (Wertabfrage) 461
Überschreibbar 460
Wertberechnung 464
Bezeichnung 444
Datenabfrage
Historie eines Einzelobjektes 451
Historie mehrerer Objekte 452
Historische Zuordnungen 454
Objekt 449
Sicht 448
Simulation 456
SQL 447
testen 446
Datenfeld 466
Datenquelle
bearbeiten 446
löschen 446
virtuell 465
einbinden 472
erstellen 443
exportieren 465, 474
importieren 465
kopieren 443
laden 443
mehrsprachig 471
Oberflächenformular 472
Präprozessorbedingung 444
ReportAlias 444
ReportName 472
ReportParameter 472
übersetzen 465, 471
Bildungsregel
Abhängigkeiten 51
Änderung verhindern 84
Ausführung limitieren 85
bearbeiten 83
Empfänger 82
lokal 86
objektübergreifend 86
Schwellwerte 85
Sender 82
testen 411
überschreibend 83
Wert einkürzen 83
Bundesland
aktivieren 231
Arbeitsstunden 231
bearbeiten 229, 231
Feiertage 229, 231
Sprachkultur 229, 231
Zeitzone 229, 231

C

Collection 22

Common Table Expressions 487
Connection
CacheReloadInterval 340
CacheType 340
DirectConnection 340
JobGenLogDir 340
LogBlobReads 340
NoReloadBeep 340
ObjectDumpStackExpression 340
SupportReadScaleOut 340
TokenCertificateFile 340
TokenCertificateThumbprint 340
Customizer 18, 80

D

Database Agent Service 345, 553
DatabaseAgentPlugin
ProviderIDs 345
Dateigruppen 115
Datenänderung
Aufbewahrungszeit 365
aufzeichnen 350-351
löschen 365
Datenbank
Authentifizierungsmodul 35
Dateigruppen 115
Datenbank-ID 35
Entwicklungsdatenbank 39, 548
Hauptdatenbank 35
Kundenname 35
Kundenpräfix 35
Land 35, 41
Öffentlicher Schlüssel 35
Produktive Datenbank 39, 548
Sprache 35
SQL Anpassungen 498
Staging-Ebene 35, 39
Staging Ebene 548
Testdatenbank 39, 548
Verbindungsdaten 35, 38
Versionstand 35
Datenbankschicht 18
DBQueue 18
DBQueue Prozessor 547
Auftrag verarbeiten 555
Datenbankschedule 551
GenProcID 358
Initialisierung 551
Kommunikation 557
Konfiguration 548
Laufzeit 548, 550
Mengenverarbeitung 550
Prozessüberwachung 358
QBM_PDBQueuePrepare 550
QBM_PDBQueueProcess 551
QBM_PDBQueueProcess_Del 551
QBM_PDBQueueProcess_Main 551
QBM_PDBQueueProcess_Mnt 551
QBM_PWatchDog 551
QBM_PWatchDogPrepare 550
Slots 548, 551, 557
Verarbeitung 554
Zentraler Dispatcher 551, 557
DebugMailPlugin 344
DropFolder 344
DialogColumn 47
DialogMultiLanguage 234
DialogNotification 82
DialogTable 47

Dispatcher
IsProxy 338
ProxyIntervall 338

E

Einfügewert
definieren 147
Entität 18
Delayed Logic 22
Discard 22
EntityLogic 18
EntitySource 18, 22
Interactive 22
MarkForDeletion 22
Read Only 22
UnitOfWork 18, 22
Ereignis
Assign 264
bearbeiten 264-265
Berechtigung 266
Delete 264
Execte 264
Insert 264
Objektereignis 264-265
Progammfunktion 265
Programmfunktion 264, 266
Prozessinformation 265, 356
Reihenfolge 265
Remove 264
Update 264
EventLogLogWriter 336
Category 336
EventID 336
EventLog 336
LogSeverity 336
Source 336

F

Feiertag
bearbeiten 227, 232
Bundesland 229, 231
Land 228, 230
FileJobDestination 327
AutoUpdateSubDirectories 327
BackupFiles 327
CheckInputIndex 327
EventTypes 327
Hostname 327
InputDirectory 327
MaxListCount 327
OutputDirectory 327
Port 327
ProviderID 327
SubDirectories 327
TimerInterval 327
UseEncryption 327
FileJobProvider 318
AutoSubDirectories 318
BackupFiles 318
CheckInputIndex 318
EventTypes 318
HostName 318
InputDirectory 318
MaxListCount 318
OutputDirectory 318
Port 318
SubDirectories 318
TimerInterval 318
UseEncryption 318

FileLogWriter 337
AddServerName 337
HistorySize 337
JobLogLifeTime 337
LogLifeTime 337
LogSeverity 337
MaxLogSize 337
OutPutFile 337
ParamMaxLength 337
Formatierungsskript 90
testen 411
Formatierungstyp 88
Formulararchiv 170
Formulardefinition 167, 169
benötigte Tabellen 172
Formularfolge 172
Konfiguration 172-173
VI_Common_ChildRelation_Grid 163
VI_ElementNavigation 163, 185
VI_Generic_MasterData 162-163
VI_Report 163, 472
VI_Wizard 163
Formulareditor 155
Formulartyp 170
Edit 170
Grid 170
Info 170
MemberRelation 165, 170
Report 170
Virtual 170
Wizard 170
Formularvorlage 169-170
Einsatzzweck 170
Formulararchiv 170
Formularquelle 170
Formulartyp 170
FrmCommonChildRelationGrid 163
FrmCommonOneChildAndMemberRelation 163
FrmCommonOneChildRelation 163
FrmCommonOneDynamicRelation 163
FrmCommonOneGenericRelation 163
FrmCommonOneMember AndChildRelation 163
FrmCommonOneMemberRelation 163
FrmElementNavigation 163
frmGeneric 163
ReportForm 163
WizardForm 163
Fremdschlüsselspalte 52
Dynamische 52
FTP Benutzer 320
FTP Server 320
FTPJobDestination 329
AutoUpdateSubDirectories 327
BackupFiles 327
CheckInputIndex 327
EventTypes 327
FTPPassword 329
FTPPort 329
FTPServer 329
FTPUser 329
Hostname 327
InputDirectory 327
MaxListCount 327
OutputDirectory 327
Port 327
ProviderID 327

SubDirectories 327
TimerInterval 327
UseEncryption 327
FTPJobProvider 320
AutoSubDirectories 318
BackupFiles 318
CheckInputIndex 318
EventTypes 318
FTPPassword 320
FTPPort 320
FTPServer 320
FTPUser 320
HostName 318
InputDirectory 318
MaxListCount 318
OutputDirectory 318
Port 318
SubDirectories 318
TimerInterval 318
UseEncryption 318

G

GenProcID 349
Ersatz 358
Globally Unique Identifier (GUID) 52, 68

H

HTTP-Authentifizierung
BasicHttpAuthentication 341
WindowsHttpAuthentication 341
HTTP Server 333
HTTPJobDestination 332
ChildPort 332
ProviderID 332
RemoteDomain 332
RemotePassword 332
RemoteUser 332
Retries 332
RetryDelay 332
HTTPJobProvider 323
ParentPort 323
ParentServer 323
RemoteDomain 323
RemotePassword 323
RemoteUser 323
Retries 323
RetryDelay 323
HTTPLogPlugin
LogFile 342
Protokolldatei 342

I

Infosystem
Ampel 200
Balkendiagramm 200
Kreisdiagramm 200
Liniendiagramm 200
Statistik 196
Statistikdefinition 190
Tabelle 200
Tachometer 200
Thermometer 200
InstallState.config 294
IsChanged 383
IsDeleted 383
IsLoaded 383

J

Job Service Configuration
- Gültigkeitskontrolle 316
- Modultyp 315

Jobdestination 309
- FileJobDestination 327
- FTPJobDestination 329
- HTTPJobDestination 332

Jobgate 309

JobGenLogDir 340

Jobprovider 309
- AppServerJobProvider 323
- FileJobProvider 318
- FTPJobProvider 320
- HTTPJobProvider 323
- MSSQLJobProvider 317

Jobqueue
- Queue 306, 324

JobQueue 18

Jobserver
- ausführender Server 291
- bearbeiten 289-290
- Dienstkonto 291
- erstellen 290
- Fallback Verbindung 299
- HTTP Anfrage starten 313
- keine direkte Datenbankverbindung 299, 301
- Konfiguration übertragen 313
- konfigurieren 313
- Maschinenrolle 296
- Queue 290-291
- Serverbetriebssystem 291
- Serverfunktion 290-291, 295
- Statistik 297
- Status 313
- Verbindungsinformation 299, 301

Jobservice.cfg 312, 560

JobServiceDestination
- EncryptionScheme 324
- ExternalSlotEnvironment 324
- ExternalSlotEnvironment32 324
- ExternalSlots 324
- ExternalSlots32 324
- InternalSlots 324
- MaxExternalSlotReuse 308, 324
- PrivateKey 324
- PrivateKeyId 324, 346
- ProviderID 324
- Queue 324
- RequestTimeout 324
- StartInterval 324
- StatisticInterval 324

K

Kompilierung
- bedingte 369
- Fehlermeldung 270

Konfiguration
- ComponentDebugMode 333
- DebugMode 333
- DoNotProtectCryptedValues 333
- DoNotProtectPrivateKeys 333
- DoNotWriteConfigBack 333
- HTTPAddress 333
- HTTPHeaders 333
- HTTPPort 333
- Language 333

LogDestinationAndProviderId 333
RetriesOnFailedStart 333
SecretAllowList 333
SecretFolder 333
UseSSL 333
VerboseLogging 333
WaitTimeOnFailedStart 333
Konfigurationsdatei 312
Konfigurationsparameter
aktivieren 42, 44
Anzeigename 44
bearbeiten 42
deaktivieren 44
erstellen 43
Option 46
Präprozessorausdruck 46, 370
Präprozessorbedingung 370
präprozessorrelevant 44
verschlüsselt 44
Wert 42, 44
Konfigurationsspeicher 28

L

Land
aktivieren 226
Arbeitsstunden 230
bearbeiten 228
Feiertage 228, 230
Sprachkultur 228, 230
Zeitzone 228, 230
Länderinformation 226
Bundesland 229, 231
Feiertag 227, 232
Land 228, 230
Zeitzone 225
Language Pack
importieren 243
Launchpad 208
Aktion 211
Aktionen 210
Erweitern 208
Menüeintrag 211
NavigationNodeState 208-209
Link
einrichten 148
Liste
Anzeigemuster 144, 146
bearbeiten 144
Bedingung 144
Einfügewert 147
einzufügende Werte 144
Objekt 144
Sortierung 144
LogWriter
EventLogLogWriter 336
FileLogWriter 337
Löschverzögerung 22

M

M N Tabelle 73
MarkForDeletion 22
Maschinenrolle 294, 296
Menüeintrag
Anwendung zuweisen 137
Anzeigetext 139
Aufgabe 125
Aufgabenkategorie 125
bearbeiten 128
Bedingung 139, 142
Berechtigungsgruppe zuweisen 138

datenabhängiger 125, 142
Datenbankabfrage 142
Datenquelle 142
deaktivieren 139
Diagrammtyp 196, 200
eindeutig 142
Eintragsart 139
erstellen 128, 134, 136-137
fixer 125
freier 125
Konfigurationsschalter 139
kopieren 128, 135
laden 130
Launchpad 211
Link 125, 148
Liste 144
Menükategorie 125, 128, 134, 137
Oberflächenformular zuweisen 161
Overlay-Symbol 139, 208-209
Präprozessorbedingung 139
Reihenfolge 139
rekursiv 142, 144
Sortierung 142
Statistik 125, 196
Symbol 139
Variabelendefinition 151
Variablendefintion 149
zentrales Formularelement 125
Menüführung
auswählen 128
erstellen 134
kopieren 133
laden 128-131
Methodendefinition
Anzeigetext 215
bearbeiten 213
Berechtigung 213
Berechtigungsgruppe 213, 215
deaktivieren 214-215
freigeschaltet für 215
Name 215
Objekt 215
Objektdefinition 213
Programmfunktion 213, 215
Skript 215
Symbol 215
testen 412
Verhalten 215
Modul GUID 68
MSSQLJobProvider 317
ConnectString 317
FlushTimeoutSeconds 317
RequestQueueLimit 317
ResultQueueLimit 317

N

NavigationNodeState-Klasse 208-209

O

Oberflächenformular
Anwendung zuweisen 158
Anzeige aus der Navigationsstruktur 161
Anzeigetext 167
bearbeiten 155, 167
Berechtigungsgruppe zuweisen 159
Berichtsformular 472
deaktivieren 156, 167
Einfügewert 147

einzufügende Werte 167
ersetzen 166
erstellen 157
Formulardefinition 167
Formularname 167
Formularvorschau 155
generisches Formular 162
Konfiguration 167
kopieren 157
kundenspezifisch 162-163
Menüeintrag zuweisen 161
Objektdefinition 160
Objektdefinition zuweisen 160
Präprozessorbedingung 167
Reihenfolge 167
Stammdatenformular 162
Tabreiter 162
Überblickformular 179
Objekt
ändern 22, 26
Assign 22
base. 385
Delete 22
Discard 22
einfügen 22, 26
EntitySource 22
Handling 22
Insert 22
interaktiv 22
IsChanged 383
IsDeleted 383
IsLoaded 383
laden 22
Lebenszyklus 22
löschen 22, 26
MarkForDeletion 22
Remove 22
Transaktion 26
UnitOfWork 22
Update 22
verwerfen 22
verzögert 22
XMarkedForDeletion 22
zum Löschen markieren 22
Objektdefinition
Allgemeingültig 122
Anzeigemuster 122
Anzeigename 122
Anzeigetext (Formular) 121
Anzeigetext (Liste) 121
Anziegtext 122
Auswahlskript 119, 122
bearbeiten 122
Bedingung 119, 122
Einfügewert 147
Einzufügende Werte 122
Objekt 122
Präprozessorbedingung 122
Tabelle 122
Objektereignis 264-265
Berechtigung 266
Programmfunktion 264-266
Objektklasse 22
Objektschicht 18
One Identity Manager
Datenmodell 47
Softwarearchitektur 18
Systemkonfiguration 34
Berichte 31
Datenbankverbindung 38

Dokumentation 31
Konfigurationsparameter 42
Land aktivieren 226
Spracheinstellung 40
Zeitzone 225
One Identity Manager Abfragesprache 418
Abfrage-Header 424
Abfragehinweise 429
Abfrageparameter 428
Anzeigewerte 427
Bedingungen 430
Bezeichner 419
Blockkommentar 419
Datumsbereiche vergleichen 434
Datumsdifferenzen vergleichen 433
Datumswerte 421
Dezimalwerte 421
Display 427
Feste Werte vergleichen 435
Ganzzahlige Werte 421
In-Klausel 432
Kommentare 419
Literale 420
LongDisplay 427
Not In-Klausel 432
Order by-Klausel 426
Paging 427
Parameter 422, 435
Parameter vergleichen 435
Search-Klausel 425
Select-Klausel 425
Skip-Klausel 427
Spalten vergleichen 431
Sprachelemente 418
Take-Klausel 427
Vorformatierte Where-Klausel 422, 436
Where-Klausel 424
Zeichenkettenwerte 420
Zeilenkommentar 419
Zeitwerte 421
One Identity Manager Schema 47
Schemaübersicht 48
One Identity Manager Service
AppServerJobProvider 323
Arbeitsweise 305-308
Connection 340
DatabaseAgentPlugin 345
DebugMailPlugin 344
Dispatcher 338
einrichten 305
Ereignisanzeige 336
EventLogWriter 336
FileJobDestination 327
FileJobProvider 318
FileLogWriter 337
FTPJobDestination 329
FTPJobProvider 320
Generierungsprotokoll 340
HTTP Authentifizierung 341
HTTPJobDestination 332
HTTPJobProvider 323
HTTPLogPlugin 342
installieren 301
Jobdestination 324
JobServiceDestination 324
Konfiguration 333
Vorlage 314
Konfigurationsdatei 312, 560, 562

konfigurieren 312-313
Linux-Freigaben 345
Logwriter 335
MSSQLJobProvider 317
PerformanceCounterPlugin 344
Plugins 342
Protokolldatei 337
Prozessabholung 317
Prozesskomponente 305
Queue 290-291
remote installieren 301
RequestWatchDogPlugin 343
ScheduleCommandPlugin 342
ShareInfoPlugin 345
Sprache 333
Statistikinformation 324
StdioProcessor.exe 308, 324
StdioProcessor.log 308
One Identity Manager Service Modul
Connection 340
Dispatcher 338
HTTP-Authentifizierung 341
JobDestination 324
Konfiguration 333
Logwriter 335
Plugins 342
Private Key-Dateien 346
Prozessabholung 317

P

PerformanceCounterPlugin 344
Category 344
CounterType 344
PollingInterval 344
Plugins
DatabaseAgentPlugin 345
DebugMailPlugin 344
HTTPLogPlugin 342
PerformanceCounterPlugin 344
RemoteConnectPlugin 345
RequestWatchDogPlugin 343
ScheduleCommandPlugin 342
ShareInfoPlugin 345
Präprozessorausdruck 370
Präprozessorbedingung 371-372
auswerten 373
Bericht 444
Konfigurationsparameter 370
Menüeintrag 139
Oberflächenformular 167
Objekt 122
Prozess 254
Prozessschritt 256
Spalte 99
Statistik 192
Tabelle 73
Präsentationsschicht 18
Primärschlüsselspalte 52
Programmfunktion
Objektereignis 264
Propzesshistorie
Aufbewahrungszeit 365
löschen 365
Proxyserver 338
Prozess
Basisobjekt 254
bearbeiten 248-249
Benachrichtigung 273
Bezeichnung 254

Ereignis 264-265
erstellen 249
exportieren 254
Fehlerkontrolle 269
Generierungsbedingung 254
Gültigkeitsprüfung 269
importieren 254
kompilieren 270
kopieren 249
nicht generieren 254
Overlimit 273
Prä-Skript zur Generierung 254, 385
Präprozessorbedingung 254
Prozessinformation 254, 354
Schwellwert 254, 273
Simulation 267
Tabelle 254
UID 254
Variable 272
vergleichen 253
Prozessauftrag 278
ausführen 279-280
Basisobjekt 280
Bedingung 280
einrichten 279-280
Ereignis 280
Parameter 280
Status 278
Zeitplan 280
Prozessautomation 278
Prozesseditor
Layoutposition 246
Prozessdokument 246
Prozesselement 246
Prozessschrittelement 246
Prozessfunktion 281
Ausführungstyp 283, 286
Betriebssystemklasse 286
Bezeichnung 286
Maximale Instanzen 284, 286
Prozesskomponente 286
Prozessgenerierung
simulieren 267
testen 267
Prozesshistorie
aufzeichnen 357
Prozessinformation
aufzeichnen 352
Ausbewahrungszeit 365
Detailtiefe 355
für Ereignis 356
für Prozess 354
für Prozessschritt 355
löschen 365
Prozesskomponente 281, 305, 324
Anzeigename 286
AutoUpdateComponent 281
CommandComponent 281
ControlFilesComponent 281
DelayComponent 281
FileComponent 281
FtpComponent 281
HandleObjectComponent 281
LogComponent 281
MailComponent 281
Maximale Instanzen 284, 286
PowerShellComponent 281
ProjectorComponent 281
Prozessfunktion 286
ReportComponent 281

ScriptComponent 281
SQLComponent 281
WakeOnLanComponent 281
ZipComponent 281
Prozessparametervorlage 286
Prozessschritt
Aufzweigung der Verarbeitung 256
bearbeiten 250
Benachrichtigung 256, 276
Bezeichnung 256
erstellen 250
Fehler ignorieren 256
Fehler protokollieren 256
Generierungsbedingung 256
importieren 250, 252
kopieren 250-251
Parameter 260
bearbeiten 261
Bezeichnung 261
IN 261
INOUT 261
OUT 261
Out-Parameter 262
Parametercollection 262
Typ 261
verschlüsselt 261
versteckt 261
Vorlage 286
Wertbelegung 262
Wertvorlage 261
Prä-Skript zur Generierung 256, 385
Präprozessorbedingung 256
Priorität 256
Protokollierungsmodus 256
Prozessfunktion 256
Prozessinformation 256, 355
Server 274-275
Serverfunktion 256, 274-275
Skript zur Serverauswahl 256, 274-275
Stopp bei Fehlern 256
suchen 253
Wartemodus bei Fehlern 256
Wartezeit 256
Wiederholversuche 256
Prozessüberwachung 348-349
aktivieren 349
Aufbewahrungszeit 365
Datenänderung 350-351
DBQueue Prozessor 358
löschen 365
Prozessauslöser 349
Prozessinformation 352
Prozessverarbeitung 352
Prozessverarbeitung 244
PWatchDogPrepare 550

Q

QBM_PDBQueuePrepare 115, 550
QBM_PDBQueueProcess 551
QBM_PDBQueueProcess_Del 551
QBM_PDBQueueProcess_Main 551
QBM_PDBQueueProcess_Mnt 551
QBM_PDiskStorePhysicalSync 115
QBM_PTableMove 115
QBM_PWatchDog 551
QBM_PWatchDogPrepare 115
QBMRelation 47
QBMTranslationAddOnSource 237
QERCentralAccount 99

QERMailAddress 99

R

RemoteConnectPlugin 345
- ADGroupAuthPermittedGroup 345
- AuthenticationMethod 345
- Port 345

Report Editor 437
- Globalisierungseditor 465
- Programmeinstellung 440
- SQL Protokoll 441

RequestWatchDogPlugin 343
- Interval 343
- MinRequests 343

S

ScheduleCommandPlugin 342
- Command 342
- Interval 342
- LogSeverity 342
- OutputToLog 342
- StartCommand 342
- StopCommand 342

Schema Extension 475

Schemaeditor
- Schemaübersicht 48

Schemaerweiterung
- Änderungskennzeichen 494
- Berechtigungen 493
- Berechtigungsgruppe 493
- Datenbanksicht
  - erstellen 486, 488
  - Spalte 483
- DDL Anweisungen 494

Einfache Spalte
- erstellen 479
- Spaltenname 479

entfernen 491

Fremdschlüsselspalte
- aus Tabelle 480
- erstellen 480
- Spaltenname 480

In Datei speichern 494

Index 490
- Indexname 490

Index erzeugen 490

kundenspezifisch 475

Neue Tabelle erstellen 477

Read Only-Datenbanksicht
- Anzeigename 486
- Beschreibung 486
- Name 486
- Sichtdefinition 486

Schema Extension 475

Spalte
- Anzeigename 483
- aus Tabelle 480
- Basisspalte 483
- Berechtigung 483
- Datentyp 483
- definieren 478
- Einfache Spalte 479
- Einfügebeschränkung 483
- erstellen 479-480
- Fremdschlüsselspalte 480
- Initialwert 483
- Kommentar 483
- konfigurieren 483
- Löschbeschränkung 483

Name 483
Pflichtfeld 483
Primärschlüssel 477, 483
Spaltenname 479-480, 483
UID 483
Unicode 483
X-Spalten 477
Tabelle
Anzeigename 477
Beschreibung 477
erstellen 477
erweitern 478
Name 477
Primärschlüssel 477
X-Spalten 477
Tabelle erweitern 478
übernehmen 494
Vereinigungssicht
Anzeigename 488
Beschreibung 488
Name 488
Sichtdefinition 488
Zuordnungstabelle
Indexspalte 490
Schlüsseldatei 324
SCIM Plugin 510
Semaphor 29
Server
festlegen 274-275
Serverfunktion 294-295
Session
Session.Config.GetConfigParm 386
Session.Source.Exists 387
Session.Variables 388
Session Services 386
ShareInfoPlugin 345
Skript 375
#LD-Notation 389
$-Notation 378
base. 385
bearbeiten 401, 408
Berechtigung 405
Datentyp 378
Datumswerte 376
Eingabehilfe 394
Automatische Vervollständigung 394
erstellen 401
Funktion 385
kompilieren 404
kopieren 402
Meldung 376
overridable 404
overrides 404
Prä-Skript 385
Programmfunktion 405
RaiseMessage 376
Session Services 386
speichern 413
Syntax 375
testen 403, 406, 409-410
überschreiben 404
VID_Write2Log 376
Webserviceaufruf 504
Skriptbibliothek 393
laden 406
SOAP Web Service 517
ChangeSingleObject 517, 526
ChangeSingleObjectEx 517, 526
CreateSingleObject 517, 526

DeleteSingleObject 517, 526
DeleteSingleObjectEx 517, 526
Exists 517, 526
ExistsEx 517, 526
FireGenEvent 517, 526
FireGenEventEx 517, 526
GetCompleteSingleObject 517, 526
GetCompleteSingleObjectEx 517, 526
GetListObject 517, 526
GetListObjectWithDisplays 517, 526
GetSingleObject 517, 526
GetSingleObjectEx 517, 526
GetSingleProperty 517, 526
GetSinglePropertyEx 517, 526
installieren 519
InvokeCustomizer 517, 526
InvokeCustomizerEx 517, 526
InvokeDialogMethod 517, 526
InvokeDialogMethodEx 517, 526
Konfigurationsdatei 523
konfigurieren 523
Statusanzeige 525
web.config 523
Softwarearchitektur 18
Spalte
#LD Inhalt 235
Abbildung 81
Abhängigkeiten 91
Alternativer Primärschlüssel 99
Anzahl Kommastellen 87, 99
Anzeigename 99
Aufzeichnen 99
Basisspalte 99
bearbeiten 81, 99
Bearbeitung
sperren 30
zulassen 30
Bildungsregel 82
Bildunsgregel 99
Bitmaske 94, 99
Datentyp 99
Dynamischer Fremdschlüssel 99, 113
Eigenschaften 81
eindeutig 70
Empfänger 82
Export für SPML 99, 542
Format 99
Formatierungsskript 90
Formatierungstyp 88
Fremdschlüssel 99
Gruppe 99, 162
Hierachieinformationen 99
Index Wichtung 95, 99
Kein DB Transport 99
Keine Protokollierung 99
Länge 99
Maximale Länge 87, 99
mehrsprachig 235
Mehrsprachig 99
Mehrzeilig 99, 162
Minimale Länge 70, 87, 99
Multi-Value Spezifizierung 93, 99
MVP-Spalte 93, 99
Namensbestandteile 99
Pflichtfeld 87, 99
Präprossorbedingung 99
Primärschlüssel 99
Proxyview 99

Reihenfolge 99, 162
Rekursiver Schlüssel 99
Schwellwert 99
Sender 82
Sprachabhängigkeiten 235
Standardkonfiguration
ändern 28
wiederherstellen 28, 30
Standardwert 99
Syntax 99
Tabelle 99
Tabellenübergreifende Suche 99
Textspeicher 237
übersetzen 235
Übersetzungsquelle 235
Übersetzungsziel 235
verschlüsselt 99
Werteliste 92
Wertliste 99
zulässige Werte 92
Spaltenbeziehung 109
SPML Provisioning Service Provider 533
SPML Schema
exportieren 543
konfigurieren 542
SPML Webservice 533
installieren 534
Konfigurationsdatei 539
konfigurieren 534
Schemadatei 543
testen 544
web.config 539
Spracheinstellung
Anmeldesprache 40
Standardsprache 40
Sprachkultur 231
Bundesland 229
Land 228, 230
Standard GUID 68
Statistik 190
Abfrage 194
Aggregratfunktion 192
Anzeigename 192
Balkendiagramm 200
Bedingung 194
Berechnung 192
Bericht verwenden 197, 199
deaktivieren 196
deaktiviert 192
Diagrammtyp 200
ElementName 194, 200
ElementObjectKey 194
ElementObjectKey2 194
ElementOrder 194
ElementValue 194, 200
Historie 192
Importierte Statistikdaten 192
Kreisdiagramm 200
Liniendiagramm 200
Messwerte 194
Präprozessorbedingung 192
Schwellwert 192, 194
Sofortberechnung 192
Statistikdefinition bearbeiten 191
Tabelle 200
Tachometer 200
Thermometer 200
Zeitplan 192
Statitsik
Ampel 200

StdioProcessor.exe 308, 324
SwitchToModuleGuid() 68
SwitchToNormalGuid() 68
System Debugger 406
SQL Protokoll 409

T

Tabelle
Abbildung 57
Anzeigemuster 73
Anzeigename 73
Arbeitstabelle 52
Art 52
Basistabelle 58, 73
bearbeiten 57, 73
Bedingung für Transport 73
Cache Informationen 73
Customizer 73
Dateigruppe 115
Eigenschaften 57
Eindeutige Gruppe 70
einfache Tabelle 52
Einfügewert 147
Export für SPML 73, 542
Im Speicher vorhalten 73
Leerwerte ignorieren 70
Logischer Speicherort 73, 115
Löschverzögerung 71, 73
M allTabelle 52
M N-Tabelle 52
M N Tabelle 73
Mehrspaltige Eindeutigkeiten 70, 73
Modul GIUD 73
Modul GUID erforderlich 68
Modul GUID zulässig 68
Nutzungstyp 73
Pfad in der Hierarchie 73, 165
Physischer Speicherort 115
Präprozessorbedingung 73
Proxy 58, 61
Proxyview 73
Read only 58, 66
Common Table Expressions 487
Scope-Hierarchie 73
Sichtdefinition 73
Statistikinformationen 73
Symbol 73
Tabelle 58
Tabellenskript 67, 73
Tabellenübergreifende Suche 73
Typ 58
Union 58, 63
View 58-59
Zuweisung per Ereignis 73
Tabellenbeziehung 109
Änderungsdatum für Abhängigkeiten aktualisieren 109
Transport nur im Verbund 109
Tabellenskript 67, 73
testen 412
Transaktion 26

U

Überblickformular 179
Anzeigetext 181
Ausrichtung 186
Berechtigungsgruppe 181
deaktivieren 188
Entwurfsansicht 181
erstellen 181

Formulardefinition 185
Formularelement 181, 185
Formularelement deaktivieren 188
Formularelement einfügen 183
Formularelement löschen 189
Hintergrundfarbe 186
Kopfzeile 186
Liste 185-186
löschen 189
Menüeintrag 181
Objekt 181
OverviewNode 185
Produktzuordnung 181
Spalten 181, 186
VI_ElementNavigation 185
Vorschau 181, 188
zentrales Formularelement 181, 185
Überblicksformulareditor
Entwurfsansicht 181
Vorschau 181, 188
Übersetzung 233
bearbeiten 238, 240-241
Fallback 237
Grundlagen 234
importieren 243
Language Pack 243
Schlüsseltext ändern 242
Spalte 235
Textspeicher 237
Übersetzungsquelle 235
Übersetzungsziel 235

V

Variable
definieren 149
DialogUserUID 149, 388
EnvUserName 149, 388
Feature_ 388
FullSync 388
GenProcID 388
LogonUser 149, 388
ManageOutstandingOperation 388
Prä-Skript 272
Prozessdefinition 272
SessionType 149, 208-209
ShowCommonDialog 149, 388
UserName 149, 388
UserUID 149, 388
Verschlüsselung
PrivateKey 324
VI.DB.DLL 18
viNetworkService.exe 312
viNetworkService.exe.config 312, 562
Volltextsuche
Anwendungsserver 95
konfigurieren 95

W

Webservice
integrieren 500
Methodenaufruf
direkt 503
generisch 502
selbstdefiniert 504
Proxycode ändern 508
Skript erstellen 504
Skriptparameter 504
SOAP 504
Synchronisation per Skript 502
WCF 504

WSDL-Datei 504
Webservice-Integrationsassistent 504

X

XDateInserted 52
XDateSubItem 52
XDateUpdated 52
XIsInEffect 52
XMarkedForDeletion 22, 52
XObjectKey 52
XOrigin 52
XTouched 52
XUserInserted 52
XUserUpdated 52

Z

Zeitzone 225
Bundesland 229, 231
Land 228, 230
Zuordnungsformular 165
Konfigurationsdaten 173
Tabreiter 173