

One Identity Manager

Web Designer and Web Portal Changes

This document provides an overview of changes in Web Designer and Web Portal in comparison to previous versions. It also provides assistance for migrating web projects from prior One Identity Manager versions.

Compiling web projects with customizations

To ensure the consistency of web project customizations made in prior versions, it is important to enable **enhanced checking** during compilation.

To enable enhanced checks, execute the following steps

1. Open Web Designer and connect to the database.
2. Open the **Connection | Settings** menu item.
3. Open the **Compiling and debugging** tab.
4. Check the box **Perform enhanced check**.
5. Close the dialog.
6. Compile the web project.
7. Open the **Tasks** tab and review the list of warnings and errors.

Updating web applications

NOTE: Starting with version 7.1 One Identity Manager includes a new full-text search engine for improved search performance. You must have an application server installed

with the search service. For more information, see [Integrating full-text search](#) on page 12.

NOTE: Ensure that the application server is updated before you update the Web Portal.

To update the Web Portal to version 8.1.2

- To update the Web Portal automatically, connect to the monitoring site `http://<servername>/<application>/monitor` in a browser and start the web application update.
- To manually update the Web Portal, uninstall the existing Web Portal and install the Web Portal again.

For more information, see the *One Identity Manager Installation Guide*.

Changes in Web Designer and in Web Portal 8.0

This section provides an overview of changes in Web Designer and in Web Portal 8.0 in comparison to previous versions.

[General changes](#)

[Known Issues](#)

[VI.WebRuntime.dll API changes](#)

[Changes to standard Web Portal components](#)

General changes

Change of UI component library

This version of the Web Portal ships without the Quest User Interface (QUI) framework.

While QUI was never part of the public API surface and was not intended to be called directly, any custom code that does call the QUI API must be rewritten.

Hyper Views

The contents of a Hyper View have historically been defined as Web Designer components (e.g. `VI_Common_HyperView_Person`). Because this has been the causes for code duplication and inconsistencies, the definition stored in the `DialogTree` table is now used directly for all Hyper Views displayed in the Web Portal. The definitions within the Web Designer components have been deleted.

However, the components themselves still exist. Any extensions should work as before. However, it is strongly recommended that you re-test any existing extension to Hyper View components.

Customizing CSS

The component `VI_Styles_RendererSettings_Default` stores most of the default CSS code. This component has changed as a consequence of product rebranding and the implementation of the mobile layout. Specifically, a lot of new CSS code supports mobile devices.

If you have a custom copy of this component, it is strongly recommended that you delete this custom copy and re-implement the changes using extensions.

As a general rule, this component should never be replaced, but only extended.

New grid control list view mode

The grid control now supports a new list view mode that supports mobile devices.

The grid control will render both table views and list views of the same data. Using CSS media queries, only one of the views is displayed at any given time, depending on the display size.

The list view mode can work in different ways, which can be defined on every grid.

- Automatic mode. In this mode, the control uses the column definitions to automatically create a list view.
- Manual mode. In this mode, you can explicitly define the contents of each list item, independently from the table view.
- Disabled mode. In this mode, the list view is not rendered.

If you need to disable list view mode globally for all grids, you can set the `VI_Common_GridListViewDisabled` configuration setting to **true**.

If you have defined any grid controls in custom code, or if you have made any modifications to existing grid controls, it is strongly recommended that you test the affected forms on mobile devices.

Secure Token Server

This version of Secure Token Server (STS) is a rebrand and is not backwards compatible. It contains breaking changes for any integrators.

The following is an overview of changed settings.

- Configuration Scope is now `urn:rsts:sts:config`.

- The OAuth2 scope for obtaining an Admin access token is now `rsts:sts:primaryproviderid:rsts`.
- Base type for Provider Configurations:
`RSts.Sts.ProviderConfiguration`
- Built-in Provider Configuration types:
`RSts.Sts.AzureConfiguration`
`RSts.Sts.DefenderCloudConfiguration`
`RSts.Sts.LdapConfiguration`
`RSts.Sts.RadiusConfiguration`
`RSts.Sts.ActiveDirectoryConfiguration`
- Base type for Claim Definitions:
`RSts.Sts.ClaimDefinition`
- Built-in Claim Definition types:
`RSts.Sts.RoleClaimDefinition`
`RSts.Sts.AttributeClaimDefinition`
- OData Configuration setting annotations:
`RSts.Sts.Required`
`RSts.Sts.Secret`
`RSts.Sts.Description`
- Login page CSS class name change to `.defenderCloudDiv`.
- Days until password expires claim type changed to `urn:rsts/daysUntilPasswordExpires`.
- Encrypted password claim type changed to `urn:rsts/encryptedPassword`.
- The Windows Service name changed to `RedistributableSecureTokenServer`.
- The Windows Event Log name changed to `Secure Token Server`.
- The Event Log Source names are `RSTS` and `RSTS Audit`. If you had written an `RSts.exe.config` file with `TraceListeners` in it, you would change to:

```
<source name="RSTS" switchValue="All">
```


and

```
<source name="RSTS Audit" switchValue="All">
```
- The Windows Performance Counter application category name changed to `STS`. And the providers category is just `STS Providers`.
- The default value of the STS Issuer Name setting is `urn:RSTS/identity`.
- The default SSL certificate (if one doesn't already exist) and Signing certificate is now `CN=Redistributable STS Demo`.

Default.aspx

The file `Default.aspx` is no longer being installed. Any bookmarks that directly point to this file are no longer valid.

This change has been made because the software updating mechanism cannot manage updates for this file.

Web SQL database search functions

The Web SQL `sqlsearch` functions are used to build `WHERE` clauses to filter a table by a search query. In the past, these filters were always evaluated by the database server. These functions now use the external search index where available.

The search results may be slightly different due to syntax differences between SQL and Lucene. The database filter functionality is only provided as a fallback and should be considered deprecated.

It is recommended to index all columns that were previously used in `sqlsearch` functions. To do this, set the **Index weighting** property on the column using the Designer tool.

SqlSearch used on a table

If at least one column of the specified table is indexed, then the search index is queried. Otherwise, the table is filtered on the database server using a `LIKE` query, using the columns that are part of the table's display pattern. The search column configuration in Web Designer is not used anymore.

SqlSearch used on a column

If the column is indexed, then the search index is queried. Otherwise, the SQL Server will filter the database using a `LIKE` query.

It is advised to test any usages of the `sqlsearch` functions in custom code after migration

Web control:

VI.WebRuntime.WebControls.GridBlock

The property `GridRowAlternateCssClass` has been removed. All current browsers support flexible styling of odd or even rows. The default value for this property was **GridRowAlternate**.

Replace any styles associated with this CSS class with CSS rules using the appropriate pseudo-class:

```
.Grid tr:nth-child(odd) { /* insert rules for odd rows */ }  
.Grid tr:nth-child(even) { /* insert rules for even rows */ }
```

Using the rebranding helper tool

Because of rebranding requirements, some identifiers used by the web controls and the Web Portal have changed:

- CSS class names
- HTML attribute names
- Javascript API names
- URL parameter names

As a general rule, the product now uses the **imx** prefix to identify One Identity Manager specific declarations.

There is a command-line tool that changes the identifiers in a web project. To use the tool, the web project must exist in a folder structure. The tool is called `WebDesigner.RebrandingHelper.exe` and is provided in the `QBM\dvd\AddOn\RebrandingHelper` folder. It does not need to be installed.

To preview the changes

- Use the **P** keyword (for preview), such as:
`WebDesigner.RebrandingHelper.exe "c:\some\path" "xml" "" P`

To change the files

- Run the same command again using the **C** keyword (for change):
`WebDesigner.RebrandingHelper.exe "c:\some\path" "xml" "" C`

Known Issues

Known issue with using the menu control in the Web Designer preview

When the mouse hovers over a main menu item, the item's child menu opens. In the Web Designer preview, the user has to click on the item to open the sub menu instead.

The background and foreground colors of the menu also show slight differences. Furthermore, opening one menu does not reliably close the other menu items.

This bug only occurs when using the Web Designer preview browser. A stand-alone Internet Explorer and all other supported browsers do not exhibit this bug. This is a bug in a third-party component.

VI.WebRuntime.dll API changes

The following public classes have been removed from the public API of VI.WebRuntime.dll.

- The namespaces VI.WebRuntime.Data.History and VI.WebRuntime.Data.UiAction have been moved to the Composition API, and renamed to QBM.CompositionApi.History and QBM.CompositionApi.UiAction respectively.
- The class VI.WebRuntime.WebControls.ReportControl has been removed. Any code that directly uses this class should use VI.WebRuntime.WebControls.ReportMobileControl instead.
- The class VI.WebRuntime.DbObjectSaveInfo has been removed. It was only intended for internal use.

Changes to standard Web Portal components

Removed components

Component name	Remarks
VI_BookmarksMenu_Mobile	Not used anymore.
VI_OptionsMenu	Not used anymore.

Changed components

VI_Employee_Edit and VI_Employee_PreEdit

The component VI_Employee_Edit is now used for creation of new employees only. If a UID_Person parameter of an existing person is passed to this component, the component will redirect to VI_Common_ObjectSheet_Person.

Any extensions to VI_Employee_Edit that should affect existing employee records must be moved/copied to VI_Employee_EditMasterData or VI_Common_ObjectSheet_Person.

The component VI_Employee_PreEdit is only called for new Person objects.

VI_ITShop_AnswerQuestions

A new parameter isRequest has been added to distinguish between IT Shop and attestation inquiries. Add this parameter to any custom usages of this module.

Deprecated components

The following Web Designer components should be considered deprecated. They may be removed in a future version.

- VI_Common_HyperView_ShapePerObject
- VI_Common_HyperView_ShapeList

Removed virtual functions

These virtual functions have been removed. You can safely remove any mapping that refers to one of these functions.

Component	Virtual function	Remarks
VI_Common_PropertyEditor	LayoutPanelLayoutName() ColumnCount()	These properties are no longer useful and have been removed.
VI_Common_PropertyViewer	LayoutPanelLayoutName() ColumnCount()	These properties are no longer useful and have been removed.
VI_Interfaces_ColumnEditor_Edit	DisplayAsLinkOnly()	This property was out of use.
VI_ITShop_ProlongateOrCancel	CanNavigateBack() IsForCancellation() IsForProlongation()	These properties were out of use.

Removed virtual nodes

Component	Node identifier
VI_ITShop_ProlongateOrCancel	NavigateBack
QBM_FormTemplate_Base	ContentMobile

Removed virtual collections

These virtual collections have been removed. You can safely remove any mapping collections that refer to these collections.

Component	Collection name	Remarks
VI_ITShop_DirectDecision	PWODecisionStep	Changed to QERWorkingStep.

Removed virtual properties

Component	Property	Remarks
VI_ITShop_DirectDecision	uid_pwodecisionstep	Renamed to uid_qerworkingstep.

Added virtual properties

Component	Property	Remarks
VI_Attestation_History_Filter	headline	This property should contain the title to be displayed within the component.
VI_Compliance_ExceptionApproval_Filter		
VI_ITShop_OrderHistory_Filter		
VI_QERPolicy_ExceptionApproval_Filter		
VI_Roles_RoleAssignmentNew	objectkeyparent	This property should contain the object key of the owner object.

Removed configuration keys

The following configuration keys have been removed

- VI_Common_DarkIcons
- VI_ITShop_FieldsInOverview

The following configuration keys manage colors that are not used in the rebranded design. Any customizations to one of these keys will be lost.

- VI_Common_Color_DarkGray
- VI_Common_Color_Blue_4
- VI_Common_Color_Blue_5
- VI_Common_Color_Blue_6
- VI_Common_Color_Blue_7
- VI_Common_Color_Blue_8

- *_Color_Palette_Blue_Dark
- *_Color_Palette_Blue_Med
- *_Color_Palette_Blue_Bright
- *_Color_Palette_Blue_Tile_BG
- *_Color_Palette_Blue_Rollover
- *_Color_Palette_Gray_Light
- *_Color_Palette_Gray_ButtonOrBG
- *_Color_Palette_Gray_Inactive
- *_Color_Palette_Gray
- *_Color_Palette_Gray_Med
- *_Color_Palette_Gray_Dark_Alerting
- *_Color_Palette_Gray_Dark_Panel
- *_Color_Palette_Gray_Dark
- *_Color_Palette_Gray_Framework
- *_Color_Palette_Gray_Darkest

Changed configuration keys

The following keys have been renamed.

Old name	New name
*_Color_Palette_Violet	QBM_Primary_Violet
*_Color_Palette_Red	QBM_Primary_Red
*_Color_Palette_Orange	QBM_Primary_Orange
*_Color_Palette_Yellow	QBM_Primary_Yellow
*_Color_Palette_Green	QBM_Primary_Green
_Color_Palette_Blue_	QBM_Primary_Blue
*_Color_Palette_White	QBM_Primary_White
*_Color_Palette_Black	QBM_Primary_Black
*_Color_Palette_Gray	QBM_Primary_Gray

Changes in Web Designer and in Web Portal 7.1

This section provides an overview of changes in Web Designer and in Web Portal 7.1 in comparison to previous versions.

[Updating assembly bindings to One Identity Manager version 7.1](#)

[Integrating full-text search](#)

[General changes](#)

[Changes to standard objects](#)

[VI.WebRuntime.dll API changes](#)

Updating assembly bindings to One Identity Manager version 7.1

This section applies only if you did not install a new web application using the One Identity Manager installation tool. For more information, see [Updating web applications](#) on page 1.

If you did not install a new web application using the One Identity Manager installation tool, it is necessary to edit any existing web applications' web.config file when upgrading to version One Identity Manager 7.1.

Open the web.config file in a text editor and insert the following text beneath the tag <assemblyBinding>:

```
<dependentAssembly>
  <assemblyIdentity name="System.Web.Http" publicKeyToken="31bf3856ad364e35"
    culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-5.2.2.0" newVersion="5.2.3.0" />
</dependentAssembly>
<dependentAssembly>
  <assemblyIdentity name="System.Net.Http.Formatting"
    publicKeyToken="31bf3856ad364e35" culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-5.2.2.0" newVersion="5.2.3.0" />
</dependentAssembly>
<dependentAssembly>
  <assemblyIdentity name="Microsoft.Data.OData" publicKeyToken="31bf3856ad364e35"
    culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-5.6.0.0" newVersion="5.6.4.0" />
</dependentAssembly>
```

```
<dependentAssembly>
  <assemblyIdentity name="Microsoft.Data.Edm" publicKeyToken="31bf3856ad364e35"
    culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-5.6.0.0" newVersion="5.6.4.0" />
</dependentAssembly>
<dependentAssembly>
  <assemblyIdentity name="System.Spatial" publicKeyToken="31bf3856ad364e35"
    culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-5.6.0.0" newVersion="5.6.4.0" />
</dependentAssembly>
```

Integrating full-text search

This section applies only if you did not install a new web application using the One Identity Manager installation tool.

One Identity Manager includes a new full-text search engine for improved search performance. The Web Portal queries the search index service which must be installed as part of a new application server installation.

There are two options to migrate an existing Web Portal:

1. If the Web Portal uses an application server connection, make sure that the application server installation is running a search index service. Otherwise, search will not work.
2. If the Web Portal uses a direct database connection, open `WebDesigner.ConfigFileEditor.exe` and connect to the application. On the "Search service" tab, you have the option to define an application server to connect to. The web application will continue using the direct database connection and will only use the application server connection when a search operation is performed.

General changes

Using entities in Web Designer collections

Web projects now compile directly to the asynchronous `VI.DB` entity layer. Database-bound Web Designer collections now contain entities (`IEntity` objects) instead of `IColElem`/`ISingleDbObject` objects.

Choosing the entity load type

Selecting the correct entity load type can have a significant impact on application performance. The new Web Designer node type `Database access options` allows selecting the best load type for every use case. See the *One Identity Manager Web Designer Reference Guide* for this node type for more information on the different options.

It is recommended to load read-only entities first, and then convert an entity to a writable or interactive entity only when required. The new Web Designer action node type `Create interactive entities` can be used to convert an entity of any type into an interactive one, which can be used to set values or call customizer methods.

Change to Web SQL `GetObjectKey()` function

When calling this function on proxy objects (such as `UNSGroup`), in previous versions the key of the proxy entity was returned (such as `<Key><T>UNSGroup</T><P>...</P></Key>`).

Because this key is an artificial key that mostly cannot be used for database comparisons, the function was changed to return the key of the proxy entity instead, which will include the actual object type (such as `<Key><T>ADSGroup</T><P>...</P></Key>`).

Removed `ListView` control

The `ListView` control has been removed. Any `ListView` nodes will still be displayed in Web Designer, however the compiler will generate an error when it encounters a `ListView` node.

It is strongly recommended to replace any usage of the `ListView` control in custom web components before migrating to One Identity Manager 7.1.

ComboBox control behavior

The combobox control now automatically refreshes itself when the associated filter condition changes.

Properties moved to attributes

Some Web Designer properties correspond directly to attributes of the resulting HTML nodes. Those properties have been moved to attribute nodes. The following node types are affected:

- ALT attribute: Image
- TITLE attribute: Container, Label, Button, LinkContainer
- STYLE attribute: Container, Label, Image, Cell, Button, LinkContainer

Enhanced object sheets for groups

In previous versions, the `VI_Common_ObjectSheet` module offered a read-only view of a group object. The module uses `VI_Interfaces_ObjectSwitch_ObjectSheet` components to show a view depending on the type of the group.

The group components have been upgraded to allow for editing if the logged on user is an owner of the group. By default, this means that the user must be either:

- A member of the owner role associated with the group's service item, or
- A member of the administrator role for the group's target system or container.

This affects all groups that map to the `UNSGroup` view.

Passing entities to components

If you have defined an object sheet component for an object type, an entity of that type is passed to the component at runtime.

The actual loading usually happens in the `VI_Common_ObjectSheet` module. For performance reasons, this module now loads entities in read-only mode instead of the interactive mode.

This is a change that may require modifications to existing components. If your object sheet component implements editing functionality, you will need to convert the read-only entity into an interactive entity first.

As a general rule, components should not presume that entities being passed were loaded in any particular mode. Furthermore, converting to an interactive entity should happen at the last possible moment.

If your component requires an entity to be interactive, your component should create an interactive entity. If the entity is already an interactive entity, no actual conversion happens, so there are no side effects of doing multiple conversions.

Change to compliance simulation in the shopping cart

By default, users will not be able to submit a shopping cart if at least one request will cause a non-approvable compliance violation.

To restore the behavior of previous versions, set the configuration parameter `VI_ITShop_AllowRequestsWithNonResolvableViolations` to **true**.

Changes to standard objects

Removed components

Component name	Description	
VI_History_ChangeLog_CommonView	These components were unused and obsolete in Web Designer 7.0 after the re-implementation of the history views. The components have been removed in this release.	
VI_History_ChangeLog_Generic		
VI_History_ChangeLog_Person		
VI_History_StateComparison_Org		
VI_History_StateComparison_Person		
VI_History_StateComparison_Person_MasterData		
VI_History_StateComparison_Person_Memberships_Generic		
VI_History_StateComparison_Person_Responsibilities_Generic		
VI_History_StateOverview		
VI_History_StateOverview_Grid		
VI_History_StateOverview_Person		
VI_Compliance_ExceptionApproval_ListView		These components are obsolete due to the removal of the ListView control.
VI_Common_ListView_Item		
VI_Common_WebPart_HighRiskOverview	Not used anymore.	
VI_Attestation_Edit_EmployeeCertification	The functionality of this component has been moved to the new component VI_Attestation_Detail_EmployeeCertification.	

Removed virtual functions

Component	Virtual function	Remarks
VI_Attestation_AttestationRuns_SendReminder	OptionalFilterAttestationRuns() ()	Use the function ListOfAttestationRuns () to pass a list of attestation runs.
VI_Attestation_AttestationRun_	UID_AttestationPolicy()	Use the function UID_

Component	Virtual function	Remarks
PendingApprovers	RunDate()	AttestationRun() instead.
VI_ITShop_ProductSelection_ReferenceUser	PersonForOrder()	Use the collection PersonsForOrder to supply the request recipients.

Removed virtual nodes

Component	Node identifier	Remarks
VI_ITShop_AccProductGroupSelector	AdditionalControls	Not supported anymore.
VI_ITShop_AccProductGroupList	SetSearchType	Not required anymore.

Removed virtual collections

These virtual collections have been removed. You can safely remove any mapping collections that refer to these collections.

Component	Collection name	Remarks
VI_Attestation_AttestationRuns_SendReminder	attestationruns	Use the function ListOfAttestationRuns() to pass a list of attestation runs.

Removed virtual properties

Component	Property	Remarks
VI_Attestation_AttestationRuns_SendReminder	uid_attestationpolicy rundate	Use the function ListOfAttestationRuns() to pass a list of attestation runs.
VI_Interfaces_ObjectSwitch_Object_Overview	xobjectkey	Not required anymore.
VI_ITShop_AccProductGroupList	accproductwhereclause	Not required anymore.

Component	Property	Remarks
VI_ITShop_AccProductGroupSelector	accproductwhereclause searchtype	Not required anymore.
VI_Roles_AttestationCasesForObject	xobjectkey	Not required anymore.
VI_Roles_Membership	tablename pk display xobjectkey isforitshop	Not required anymore.
VI_Roles_UNSGroupInUNSGroup	tablename	Not required anymore.

VI.WebRuntime.dll API changes

The API changes were made mainly to support the new object layer including asynchronous operations. This has a couple of effects:

1. Methods have been replaced by asynchronous implementations, and renamed to include an **async** suffix. For some methods, synchronous wrapper methods are available as extension methods. To use these methods, add a using `VI.WebRuntime.Compatibility;` directive to the consuming code.
2. Synchronous events have been replaced by `IAsyncObservable<T>` implementations of the observer pattern.
3. Methods that expect legacy argument types have been changed to support their successors:
 - a. `IColDef` --> `IMetaColumn`
 - b. `ITableDef` --> `IMetaTable`
 - c. `IConnection` --> `ISession`

Note that this list is not exhaustive. For example, it does not include overriding method implementations that have been removed.

The list also does not include new interface members. For more information on a specific interface member, see the *One Identity Manager Web Runtime documentation*.

Removed types

The following publicly visible types have been removed.

Full Name	Migration path
VI.WebRuntime.AjaxLoadMoreCommand	Support for the ListView control has been removed.
VI.WebRuntime.BaseActionSegment	Removed because of asynchronous implementation.
VI.WebRuntime.BaseMultipartLoop	Removed because of asynchronous implementation.
VI.WebRuntime.Communication .BaseRequestSessionHandler	Not intended to be called directly.
VI.WebRuntime.Communication.IRouteConfig	Not intended to be called directly.
VI.WebRuntime.Data .CollectionHierarchyDataSource	Use the IControlDataSource implementations to obtain hierarchical data.
VI.WebRuntime.Data .DataSourceListViewFilterFormMethod	Support for the ListView control has been removed.
VI.WebRuntime.Data.DbHierarchyDataSource	Use the IControlDataSource implementations to obtain hierarchical data.
VI.WebRuntime.Data.History.DeleteAction	Moved to the VI.WebRuntime.Data.UiAction namespace.
VI.WebRuntime.Data.History.DeleteDirectAssignment	Moved to the VI.WebRuntime.Data.UiAction namespace.
VI.WebRuntime.Data.History.HistoryRestoreActionRunner	Moved to the VI.WebRuntime.Data.UiAction namespace and renamed ActionRunner.
VI.WebRuntime.Data.History .IHistoryRestoreActionContext	Moved to the VI.WebRuntime.Data.UiAction namespace and renamed IUiActionContext.
VI.WebRuntime.Data.History.IHistoryRollbackAction	Moved to the VI.WebRuntime.Data.UiAction namespace and renamed IUiAction.
VI.WebRuntime.Data.History .ImpossibleRollbackAction	Moved to the

Full Name	Migration path
VI.WebRuntime.Data.History .PropertyRollback	VI.WebRuntime.Data.UiAction namespace and renamed ImpossibleAction.
VI.WebRuntime.Data.IHierarchyDataSource	Moved to the VI.WebRuntime.Data.UiAction namespace and renamed SetPropertyAction.
VI.WebRuntime.Data.IWhereClauseFilterCondition	Use the IControlDataSource implementations to obtain hierarchical data.
VI.WebRuntime.Data.WhereClauseFilterCondition	Use IWhereClauseFilter instead.
VI.WebRuntime.DataRowLoop	Use IWhereClauseFilterCondition instead.
VI.WebRuntime.GenericActionSegment	Removed because of asynchronous implementation.
VI.WebRuntime.IActionSegment	Removed because of asynchronous implementation.
VI.WebRuntime.IListViewHost	Removed because of asynchronous implementation.
VI.WebRuntime.IoC .UnityPerCallContextLifetimeManager<T>	Support for the ListView control has been removed.
VI.WebRuntime.ListViewHost	Not intended to be called directly.
VI.WebRuntime.MultiSelectCollection	Support for the ListView control has been removed.
VI.WebRuntime.RowValueUpdateHandler	Removed, not in use.
VI.WebRuntime.Services .ClientErrorLogService	Removed, not in use because the events have been replaced by observers.
VI.WebRuntime.WebControls.IFilterInformationBar	Renamed to ClientLogService,
	Not intended to be called directly.

Full Name	Migration path
VI.WebRuntime.WebControls .IListViewAction	Support for the ListView control has been removed.
VI.WebRuntime.WebControls .IListViewColumn	Support for the ListView control has been removed.
VI.WebRuntime.WebControls .JqmDataTransitionType	jQuery Mobile is not used anymore.
VI.WebRuntime.WebControls .JqmGenericControl	jQuery Mobile is not used anymore.
VI.WebRuntime.WebControls .JqmLinkButtonControl	jQuery Mobile is not used anymore.
VI.WebRuntime.WebControls .JqmListViewControl	jQuery Mobile is not used anymore.
VI.WebRuntime.WebControls .JqmPopupControl	jQuery Mobile is not used anymore.
VI.WebRuntime.WebControls.Angular .ListViewFilterControl	Support for the ListView control has been removed.
VI.WebRuntime.WebControls.IListView	Support for the ListView control has been removed.
VI.WebRuntime.WebControls.IListViewItem	Support for the ListView control has been removed.
VI.WebRuntime.WebControls.ListView	Support for the ListView control has been removed.
VI.WebRuntime.WebControls.ListViewAction	Support for the ListView control has been removed.
VI.WebRuntime.WebControls.ListViewColumn	Support for the ListView control has been removed.
VI.WebRuntime.WebControls.ListViewItem	Support for the ListView control has been removed.
VI.WebRuntime.WhileLoop	Removed because of asynchronous implementation.

Removed type members

Full Name	Migration path
VI.WebRuntime.ActionQueue..ctor()	Not intended to be called externally.

Full Name	Migration path
VI.WebRuntime.ActionQueue.QueueAction (Action)	The method expects an additional parameter (HttpContextBase). Also note support for tasks in the QueueTask method.
VI.WebRuntime.AdhocColumn . DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.AdhocColumn .RaiseRowUpdated(IDataRow,DbVal,DbVal)	Replaced by RaiseRowUpdatedAsync.
VI.WebRuntime.AdhocColumn .RowInvalidated (EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.AdhocColumn.GetValue (IDataRow)	Replaced by GetValueAsync.
VI.WebRuntime.AdhocColumn.RowUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ApplicationInfo .GetMenuStructure	Type changed IMenuStructureProvider.
VI.WebRuntime.ApplicationInfo .UserSessionCount	Removed.
VI.WebRuntime.ApplicationInfoGroup..ctor ()	Not intended to be called externally.
VI.WebRuntime.BaseRuntimeTable . ObjectSaved(EventHandler<EventArgs <IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable . PermissionChanged(EventHandler <EventArgs<DataCell>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable . RowDeleted(EventHandler<EventArgs <IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable . RowDisplayChanged(EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable . RowInserted(EventHandler<EventArgs <IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable . RowInserting(EventHandler<EventArgs <IDataRow>>)	Event replaced by IAsyncObservable object.

Full Name	Migration path
VI.WebRuntime.BaseRuntimeTable .RowUpdated(RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable .RowUpdating(RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable .AddDeferredDeletion(Func<IDataRow, Boolean>)	Removed. Call DeleteDbObjectsAsync with DeletionMode.DeleteDeferred to add a deferred deletion.
VI.WebRuntime.BaseRuntimeTable .BulkOperation(EventHandler <EventArgs<IRuntimeTable>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable .DatabaseTable	Moved to the IDbObjectStore interface.
VI.WebRuntime.BaseRuntimeTable .DataStateChanged(EventHandler <EventArgs<IRuntimeTable>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable .EnsurePrimaryKeyFilled(IDataRow)	Replaced by EnsurePrimaryKeyFilledAsync.
VI.WebRuntime.BaseRuntimeTable .Invalidated(EventHandler<EventArgs <IRuntimeTable>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseRuntimeTable .OnRowInserted(IDataRow)	Replaced by OnRowInsertedAsync.
VI.WebRuntime.BaseRuntimeTable .OnRowInserting(IDataRow)	Replaced by OnRowInsertingAsync.
VI.WebRuntime.BaseRuntimeTable .RaiseDataStateChanged(IRuntimeTable)	Replaced by RaiseDataStateChangedAsync.
VI.WebRuntime.BaseRuntimeTable .RaisePermissionChanged(DataCell)	Replaced by RaisePermissionChangedAsync.
VI.WebRuntime.BaseRuntimeTable .RaiseRowUpdated(IDataRow, IDataColumn, DbVal, DbVal)	Replaced by RaiseRowUpdatedAsync.
VI.WebRuntime.BaseRuntimeTable .RaiseTableInvalidated()	Replaced by RaiseTableInvalidatedAsync.
VI.WebRuntime.BaseRuntimeTable .SetDeletionFlag(Func<IDataRow, Boolean>)	Removed. Call Entity.MarkForDeletion to mark an entity for deletion.
VIWebRuntime.BaseRuntimeTable.ThrowBulkEventsChanged(EventHandler <EventArgs<IRuntimeTable>>)	Event replaced by IAsyncObservable object.

Full Name	Migration path
VI.WebRuntime.BaseRuntimeTable.Delete (Func<IDataRow,Boolean>,DeletionMode)	Removed. Use one of the Delete extension methods in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.BaseRuntimeTable.Delete (IDataRow)	Removed. Use one of the Delete extension methods in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.BaseRuntimeTable.Insert (ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.BaseRuntimeTable.Insert (ResultSet,Int32,ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.BaseRuntimeTable.Save (IEnumerable<IDataRow>)	Removed. Use one of the Save extension methods in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.BaseRuntimeTable.SuspendBulkEvents()	Return type changed to IAsyncActionBlock.
VI.WebRuntime.BaseTableRow .RowValueUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseTableRow .RowValueUpdating (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.BaseTableRow .BeginTransaction()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.BaseTableRow .CommitTransaction()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.BaseTableRow .DatabaseTable	Moved to the IDbObjectRow interface.
VI.WebRuntime.BaseTableRow .RollbackTransaction()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.BaseTableRow .TransactionOpen	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.BaseTableRow.SetForeignKey (IDbObjectRow,IDataColumn)	Replaced by an extension method SetForeignKey[Async] in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.BaseTableRow.SetValue (String,Object)	Replaced by SetValueAsync.

Full Name	Migration path
VI.WebRuntime.BaseTableRow.SetValueSafe (IDataRow,IDataColumn,Object)	Replaced by SetValueSafeAsync.
VI.WebRuntime.CallActionFormMethod .Execute(HttpContextBase,IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.ColumnEditorService .GetComponent(IDataColumn,RowCursor ,Boolean)	Replaced by GetComponentAsync.
VI.WebRuntime.ColumnEditorService .RegisterComponent(IColDef,Boolean ,String)	Argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.ColumnEditorService.GetKey (IColDef,Boolean)	Argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.ColumnEditorService.GetKey (IDataColumn,IDataRow,Boolean)	Replaced by GetKeyAsync.
VI.WebRuntime.ColumnInsertValues .ChangeRow(IDataRow)	Replaced by ChangeRowAsync.
VI.WebRuntime.ColumnProxy . RowInvalidated (EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ColumnProxy .DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.ColumnProxy .ForeignKeyColumn	Type changed to IMetaColumn.
VI.WebRuntime.ColumnProxy .RaiseRowUpdated(IDataRow,DbVal,DbVal)	Replaced by RaiseRowUpdatedAsync.
VI.WebRuntime.ColumnProxy .SetBaseColumnValue(IDataRow)	Replaced by SetBaseColumnValueAAsync.
VI.WebRuntime.ColumnProxy . RowUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ColumnProxy .GetValue (IDataRow)	Replaced by GetValueAsync.
VI.WebRuntime.ColumnProxy .SetValue (IDataRow,Object)	Replaced by SetValueAsync.
VI.WebRuntime.Communication .IRequestModuleInfo.UserSession	Converted to a get-only property.
VI.WebRuntime.Compilation .CodeGenProjectBase .GetAssemblyVersionString()	Replaced by AssemblyVersion.

Full Name	Migration path
VI.WebRuntime.Compilation .CodeGenProjectBase .SourceCodeDirectory	Converted to a get-only property.
VI.WebRuntime.Compilation .CodeGenProjectBase.GetTempBaseDirectory ()	Not intended to be called directly.
VI.WebRuntime.Compilation .CodeGenProjectBase.IsRtmCompatible	Removed because this flag was only meaningful for version 7.0.1. Assume value to be false.
VI.WebRuntime.ConfigDocument .CookieDomain	Support for cookie domains has been removed.
VI.WebRuntime.ConfigDocumentMono .CookieDomain	Support for cookie domains has been removed.
VI.WebRuntime.ConfigurableAdhocColumn .ForeignKeyColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.CRViewTable .LoadRelatedObjects()	Replaced by LoadRelatedObjectsAsync.
VI.WebRuntime.CRViewTable.Load (DBObjectLoadInfo, ILoadTableOptions)	Replaced by LoadAsync.
VI.WebRuntime.CRViewTable.SetSource (IColDef, IRuntimeTable, IEnumerable <IDataRow>)	Replaced by SetSourceAsync.
VI.WebRuntime.Data .ColumnValueBasicValidator.Validate (IAjaxResponse)	Replaced by ValidateAsync. Also, the parameter has been removed and return type has been changed to Task<IValidationErrorMessage>.
VI.WebRuntime.Data .DynamicColumnValidator.Validate (IDataRow)	Replaced by ValidateAsync.
VI.WebRuntime.Data.CollectionDataSource .BeginIndex	Moved to the IDataSourceLevel1 interface.
VI.WebRuntime.Data.CollectionDataSource .CanHaveChildren(IDataRow)	Replaced by CanHaveChildrenAsync.
VI.WebRuntime.Data.CollectionDataSource .CurrentPage	Moved to the IDataSourceLevel1 interface.
VI.WebRuntime.Data.CollectionDataSource .GetData(Int32, Int32, Boolean, Boolean , IEnumerable<IDataSourceFilter>, IDataRow , Boolean)	Moved to the IDataSourceLevel1 interface and replaced by GetDataAsync.

Full Name	Migration path
VI.WebRuntime.Data.CollectionDataSource .GetHierarchyDataSource()	Removed, not supported anymore.
VI.WebRuntime.Data.CollectionDataSource .GetTotalCount()	Moved to the IDataSourceLevel1 interface and replaced by GetTotalCountAsync.
VI.WebRuntime.Data.CollectionDataSource .GetTotalCount(IDataRow)	Moved to the IDataSourceLevel1 interface and replaced by GetTotalCountAsync.
VI.WebRuntime.Data.CollectionDataSource .HasAnyData()	Moved to the IDataSourceLevel1 interface and replaced by HasAnyDataAsync.
VI.WebRuntime.Data.ColumnFilterProvider .Columns	Type changed to IEnumerable<IMetaColumn>.
VI.WebRuntime.Data.ColumnFilterValueList .ToSqlFilterString(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.Data.ColumnHierarchyFilter .ToSqlFilterString(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.Data.ColumnValueValidator .Validate(IAjaxResponse)	Replaced by ValidateAsync. Also, the parameter has been removed and return type has been changed to Task<IValidationErrorResponse>.
VI.WebRuntime.Data.ComparisonFilterValue .ToSqlFilterString(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.Data.DbDataSource .CanHaveChildren(IDataRow)	Replaced by CanHaveChildrenAsync.
VI.WebRuntime.Data.DbDataSource .CurrentPage	Moved to the IDataSourceLevel1 interface.
VI.WebRuntime.Data.DbDataSource .GetHierarchyDataSource()	Removed, not supported anymore.
VI.WebRuntime.Data.DbDataSource .GetTotalCount()	Moved to the IDataSourceLevel1 interface and replaced by GetTotalCountAsync.
VI.WebRuntime.Data.DbDataSource .HasAnyData()	Moved to the IDataSourceLevel1 interface and replaced by HasAnyDataAsync.
VI.WebRuntime.Data.DbDataSource .Table	Type changed to IDbObjectStore.
VI.WebRuntime.Data.DbDataSource..ctor (IDataSourceTable<DbObjectLoadInfo> ,String,Boolean,Int32)	First argument type changed to IDbObjectStore.
VI.WebRuntime.Data.DbDataSource.GetData (Int32,Int32,Boolean,Boolean,IEnumerable	Moved to the IDataSourceLevel1 interface and replaced by GetDataAsync.

Full Name	Migration path
<IDataSourceFilter>,IDataRow,Boolean)	
VI.WebRuntime.Data.DynamicValidator.Validate(IAjaxResponse)	Replaced by ValidateAsync(IDataRow row).
VI.WebRuntime.Data.History.AssignColumn.GetComparisonAsync(ISession,String,DateTime,IEnumerable<IHistoryEvent>,ITranslator,CancellationTokens)	Parameter 2 type changed from string to IEnumerable<IDbObjectKey>.
VI.WebRuntime.Data.IColumnValidator.Validate(IDataRow)	Replaced by ValidateAsync.
VI.WebRuntime.Data.IControlDataSource.GetHierarchyDataSource()	Removed, not supported anymore.
VI.WebRuntime.Data.IControlDataSource.HasAnyData()	Moved to the IDataSourceLevel1 interface and replaced by HasAnyDataAsync.
VI.WebRuntime.Data.IDataFilterProvider.Columns	Type changed to IEnumerable<IMetaColumn>.
VI.WebRuntime.Data.IDataGroupDescription.GetCountInGroupAsync(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.Data.IDataGroupProvider.GetDataGroupDescriptions()	Replaced by GetDataGroupDescriptionsAsync.
VI.WebRuntime.Data.IDataGroupProvider.GetDataGroupDescriptions(Boolean)	Replaced by GetDataGroupDescriptionsAsync.
VI.WebRuntime.Data.IDataSourceBase.CanHaveChildren(IDataRow)	Replaced by CanHaveChildrenAsync.
VI.WebRuntime.Data.IDataSourceBase.GetData(Int32,Int32,Boolean,Boolean,IEnumerable<IDataSourceFilter>,IDataRow,Boolean)	Moved to the IDataSourceLevel1 interface and replaced by GetDataAsync.
VI.WebRuntime.Data.IDataSourceBase.GetTotalCount(IDataRow)	Moved to the IDataSourceLevel1 interface and replaced by GetTotalCountAsync.
VI.WebRuntime.Data.IDataSourceFilter.ToSqlFilterString(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.Data.IGroupedDataSource.GetGroups()	Replaced by GetGroupsAsync.
VI.WebRuntime.Data.IGroupedDataSource.GetGroups(Boolean)	Replaced by GetGroupsAsync.
VI.WebRuntime.Data.IPagedData	Replaced by GetTotalCountAsync.

Full Name	Migration path
.GetTotalCount()	
VI.WebRuntime.Data.IValidationGroup .Validate()	Replaced by ValidateAsync.
VI.WebRuntime.Data.IValueValidator .Validate (IAjaxResponse)	Replaced by ValidateAsync. Additionally, the parameter has been removed and the return type has been changed to Task<IValidationErrorResponse>.
VI.WebRuntime.Data.StringFilterBase .GetSqlFilterString(IConnection,IColDef ,String)	Parameter type changed from IConnection to ISession and from IColDef to IMetaColumn.
VI.WebRuntime.Data.StringFilterBase .ToSqlFilterString(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.Data.StringPrefixFilter .GetSqlFilterString(IConnection,IColDef ,String)	Parameter type changed from IConnection to ISession and from IColDef to IMetaColumn.
VI.WebRuntime.Data.StringSuffixFilter .GetSqlFilterString(IConnection,IColDef ,String)	Parameter type changed from IConnection to ISession and from IColDef to IMetaColumn.
VI.WebRuntime.Data.StringTermsFilter .GetSqlFilterString(IConnection,IColDef ,String)	Parameter type changed from IConnection to ISession and from IColDef to IMetaColumn.
VI.WebRuntime.Data.ValidationGroup .Validate ()	Replaced by ValidateAsync.
VI.WebRuntime.Data.WhereClauseFilter .ToSqlFilterString(IConnection)	Parameter type changed to ISession.
VI.WebRuntime.DatabaseColumn . RowInvalidated(EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.DatabaseColumn . RowUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.DatabaseColumn .DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.DatabaseColumn .RaiseRowUpdated(IDataRow,DbVal,DbVal)	Replaced by RaiseRowUpdatedAsync.
VI.WebRuntime.DatabaseColumn.GetValue (IDataRow)	Replaced by GetValueAsync.

Full Name	Migration path
VI.WebRuntime.DatabaseColumn.SetValue (IDataRow,Object)	Replaced by SetValueAsync.
VI.WebRuntime.DataRowStore.AddToRowList (IDataRow)	Replaced by AddToRowListAsync.
VI.WebRuntime.DataRowStore.Insert (ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.DbObject . PermissionChanged (EventHandler <EventArgs<DataCell>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.DbObject . TransactionOpen	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.DbObject .CreateFromObjectWithParameter (IRuntimeTable,ISingleDbObject ,IDialogParameter)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject .GetDatabaseObjectDisplay(Boolean)	Use Entity.Display or Entity.LongDisplay.
VI.WebRuntime.DbObject .HasColElemFkDisplayValues	Removed because IColElem is not used anymore.
VI.WebRuntime.DbObject .HasSingleObject	Removed because ISingleDbObject is not used anymore. Examine the wrapped entity to determine whether it is fully loaded, writable or interactive.
VI.WebRuntime.DbObject .RollbackTransaction ()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.DbObject. Saved (EventHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.DbObject..ctor (IRuntimeTable,DbObjectKey)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject..ctor (IRuntimeTable,IColDbObject,IColElem)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject..ctor (IRuntimeTable,ISingleDbObject)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject.BeginTransaction()	Replaced by the IAsyncTransactable implementation.

Full Name	Migration path
VI.WebRuntime.DbObject.ColElem	Removed because IColElem is not used anymore. Use the Entity property to obtain the entity.
VI.WebRuntime.DbObject.Collection	Removed because IColDbobject is not used anymore.
VI.WebRuntime.DbObject.CommitTransaction() ()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.DbObject.CreateFrom (IRuntimeTable,DbObjectKey)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject.CreateFrom (IRuntimeTable,IColDbObject,IColElem)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject.CreateFrom (IRuntimeTable,IDbObjectRow)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject.CreateFrom (IRuntimeTable,ISingleDbObject ,DialogParameter)	Not intended to be directly called. Use IDbObjectStore.InsertEntityAsync to add an entity to a collection.
VI.WebRuntime.DbObject.DatabaseTable	Removed; use the Entity.Tablename property to obtain the database table name.
VI.WebRuntime.DbObject.GetObjectWalker (String)	Removed, use the generic method.
VI.WebRuntime.DbObject.GetObjectWalker<T >(String)	Replaced by the extension method VI.WebRuntime.Compatibility.GetObjectWalkerAsync.
VI.WebRuntime.DbObject .IsDeleted()	Removed, use the Entity.IsDeleted property.
VI.WebRuntime.DbObject.IsLoaded()	Removed, use the Entity.IsLoaded property.
VI.WebRuntime.DbObject.Save()	Removed. Use one of the Save extension methods in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.DbObject.SetValue(String ,Object)	Replaced by SetValueAsync.

Full Name	Migration path
VI.WebRuntime.DbObject.SingleObject	Removed, use the Entity property to obtain the wrapped entity. Also see the section "Using entities in Web collections".
VI.WebRuntime.DbObject.TableDef	Renamed to DatabaseTable; type changed to IMetaTable.
VI.WebRuntime.DbObjectLoadInfo .DbTableName	Removed, use the Select property instead.
VI.WebRuntime.DbObjectLoadInfo .DbWhereClause	Removed, use the Select property instead.
VI.WebRuntime.DbObjectLoadInfo .DisplayPattern	Removed, use the Select property instead.
VI.WebRuntime.DbObjectLoadInfo .OrderBy	Removed, use the Select property instead.
VI.WebRuntime.DbObjectRelationViewBase .InsertRowIfNew(IColElem,IColDbObject ,IDbObjectRow&)	Not intended to be called externally.
VI.WebRuntime.DbObjectRelationViewBase .Invalidate()	Replaced by InvalidateAsync.
VI.WebRuntime.DbObjectRelationViewBase .LoadRelatedObjects()	Replaced by LoadRelatedObjectsAsync.
VI.WebRuntime.DbObjectRelationViewBase .Recalculate()	Replaced by RecalculateAsync.
VI.WebRuntime.DbObjectRelationViewBase .RelationColumn	Removed; now implemented by the inheriting classes as required.
VI.WebRuntime.DbObjectRelationViewBase .SetSourceInternal (IRuntimeTable,IEnumerable<IDataRow>,IColDef,Func <ITableDef>)	Replaced by SetSourceInternalAsync. Type argument changed from IcolDef to ImetaColumn, and ItableDef to ImetaTable.
VI.WebRuntime.DbObjectStore . DbCollection	Removed because IColDbObject is not used anymore.
VI.WebRuntime.DbObjectStore .AddDeferredDeletion(Func<IDataRow ,Boolean>)	Removed. Call DeleteDbObjectsAsync with DeletionMode.DeleteDeferred to add a deferred deletion.
VI.WebRuntime.DbObjectStore .AddRowsFromHistory(DateTime,String ,Object)	Replaced by AddRowsFromHistoryAsync.

Full Name	Migration path
VI.WebRuntime.DbObjectStore .CreateDialogParameters(IDbObjectRow ,String,DbObjectStore,String)	Replaced by an extension method: <pre>public static void CreateDialogParameters(this IDbObjectStore psetTable, IDbObjectRow row, string objectKeyParent, IDbObjectStore parameterTable, string newParameterSetName)</pre>
VI.WebRuntime.DbObjectStore .FillFromFKRelation(IDataRow,IDataColumn ,String,ILoadTableOptions)	Moved to the extension method FillFromFKRelationAsync.
VI.WebRuntime.DbObjectStore .HandleDeferredDeletions()	Replaced by HandleDeferredDeletionsAsync.
VI.WebRuntime.DbObjectStore .InsertFromXml (String,Int32 ,ColumnInsertValues)	Use the new XmlDbObjectImporter class to import entities from a XML representation.
VI.WebRuntime.DbObjectStore .JumpToHistoryDate(DateTime,IEnumerable <IDataRow>)	Replaced by JumpToHistoryDateAsync.
VI.WebRuntime.DbObjectStore .LoadDbCollection(ILoadTableOptions ,IColDbObject)	Call LoadAsync to load entities and add them to the collection.
VI.WebRuntime.DbObjectStore .PrepareDbColObject(IColDbObject)	Removed because IColDbObject is not used anymore.
VI.WebRuntime.DbObjectStore .SetDeletionFlag(Func<IDataRow,Boolean>)	Removed. Call Entity.MarkForDeletion to mark an entity for deletion.
VI.WebRuntime.DbObjectStore.AddRows (IColDbObject,DbObjectLoadInfo ,ILoadTableOptions)	Call InsertRowAsync to add a new entity to the collection; or LoadAsync to load entities and add them to the collection.
VI.WebRuntime.DbObjectStore.AddRows (IColDbObject,ILoadTableOptions)	Call InsertRowAsync to add a new entity to the collection; or LoadAsync to load entities and add them to the collection.
VI.WebRuntime.DbObjectStore.Insert (ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.DbObjectStore.InsertRow (IDbObjectRow)	Replaced by InsertRowAsync.
VI.WebRuntime.DbObjectStore.Load (DbObjectLoadInfo,ILoadTableOptions)	Replaced by LoadAsync.

Full Name	Migration path
VI.WebRuntime.DbObjectStore.Loaded (EventHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.DbObjectStore.OnLoaded()	Replaced by OnDataLoadedAsync.
VI.WebRuntime.DialogParameterObjectProxy	This class no longer inherits from IDbObjectRow and is not intended to be called directly.
VI.WebRuntime.DialogParameterValueColumn .SetValue(IDataRow, Object)	Replaced by SetValueAsync.
VI.WebRuntime.DownloadFormMethod.Execute (HttpContextBase, IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.ExpressionColumnBase . RowInvalidated(EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ExpressionColumnBase . RowUpdated(RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ExpressionColumnBase .DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.ExpressionColumnBase .GetValue(IDataRow)	Replaced by GetValueAsync.
VI.WebRuntime.ExpressionColumnBase .Invalidate(IDataRow)	Replaced by InvalidateAsync.
VI.WebRuntime.ExpressionColumnBase .InvalidateAllRows()	Replaced by InvalidateAllRowsAsync.
VI.WebRuntime.ExpressionColumnBase .SetValue(IDataRow, Object)	Replaced by SetValueAsync.
VI.WebRuntime.ExpressionColumnBase.query	Type changed to Func<IDataRow, Cancellation token, Task<object>>.
VI.WebRuntime.ExpressionColumnUpdater .RaiseDataEventOccurred(IRuntimeTable , IDataColumn, IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.ExpressionEvent .RaiseDataEventOccurred(IRuntimeTable , IDataColumn, IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.ExtensionMethods .GetDbCount (IConnection, String, String , IEnumerable<IDataParameter>, Boolean)	Replaced by GetDbCountAsync; first argument type changed from IConnection to ISession.

Full Name	Migration path
VI.WebRuntime.ExtensionMethods .InvalidateAllColumnsData(IDataRow)	Replaced by InvalidateAllColumnsDataAsync.
VI.WebRuntime.ExtensionMethods .PreloadDisplayValueCache(IDataColumn ,Boolean,CancellationTokens)	Replaced by PreloadDisplayValueCacheAsync.
VI.WebRuntime.ExtensionMethods .PreloadDisplayValueCache(IDataColumn ,IEnumerable<IDataRow>,Boolean ,CancellationTokens)	Replaced by PreloadDisplayValueCacheAsync.
VI.WebRuntime.ExtensionMethods .PreventEnhancingByJqm (AttributeCollection)	jQuery Mobile is not used anymore.
VI.WebRuntime.ExtensionMethods .PreventEnhancingByJqm(HtmlTextWriter)	jQuery Mobile is not used anymore.
VI.WebRuntime.ExtensionMethods.Delete (IRuntimeTable)	Use one of the Delete/DeleteAsync extension methods in VI.WebRuntime.Compatibility; see also the section "Using entities in Web Designer collections".
VI.WebRuntime.ExtensionMethods.Save (IEnumerable<IDataRow>)	Use one of the Save/SaveAsync extension methods in VI.WebRuntime.Compatibility; see also the section "Using entities in Web Designer collections".
VI.WebRuntime.FileUploadFormMethod .Execute(HttpContextBase,IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.FKViewTable .LoadRelatedObjects()	Replaced by LoadRelatedObjectsAsync.
VI.WebRuntime.FKViewTable.SetSource (IRuntimeTable,IEnumerable<IDataRow> ,IColDef)	Replaced by SetSourceAsync.
VI.WebRuntime.FormMethods .MultiFormMethod.Execute(HttpContextBase ,IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.FormMethods .MultiFormMethod.Execute(IDictionary <String,String>,IFormRenderPart ,IAjaxResponse)	Replaced by ExecuteAsync.
VI.WebRuntime.FormMethods .ReportExportPdfMethod.Execute	Replaced by ExecuteAsync.

Full Name	Migration path
(HttpContextBase, IFormRenderPart)	
VI.WebRuntime.FormMethods .ReportExportXlsMethod.Execute (HttpContextBase, IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.GenericControlFormMethod .Execute(HttpContextBase, IFormRenderPart)	
VI.WebRuntime.GenericRowListUpdater .RaiseDataEventOccurred(IRuntimeTable , IDataColumn, IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.GenericRowListUpdater. .ctor (Action, Func<IEnumerable<IDataRow>>)	First argument type changed to Func<Cancellation token, Task>.
VI.WebRuntime.IApplicationInfo .GetMenuStructure	Type changed to IMenuStructureProvider.
VI.WebRuntime.IApplicationInfo .UserSessionCount	Removed.
VI.WebRuntime.IConfigDocument.CookieDomain	Support for cookie domains has been removed.
VI.WebRuntime.IDataColumn . RowInvalidated (EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IDataColumn .DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.IDataColumn .RaiseRowUpdated(IDataRow, DbVal, DbVal)	Replaced by RaiseRowUpdatedAsync.
VI.WebRuntime.IDataColumn. RowUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IDataColumn.GetValue (IDataRow)	Replaced by GetValueAsync.
VI.WebRuntime.IDataColumn.SetValue (IDataRow, Object)	Replaced by SetValueAsync.
VI.WebRuntime.IDataRow . RowValueUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IDataRow . RowValueUpdating (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IDataRow.DatabaseTable ()	Moved to the IDbObjectRow interface.
VI.WebRuntime.IDataRow.SetForeignKey (IDbObjectRow, IDataColumn)	Replaced by an extension method SetForeignKey[Async] in the

Full Name	Migration path
	VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.IDataRow.SetValue(String, Object)	Replaced by SetValueAsync.
VI.WebRuntime.IDataSourceTable<T> .Loaded(EventHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IDataSourceTable<T>.Load(T, ILoadTableOptions)	Replaced by LoadAsync.
VI.WebRuntime.IDbColumnAdapterFactory .GetColumnAdapter(IColDef, IRuntimeTable)	First argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.IDbObjectRow .HasColElemFkDisplayValues()	Removed because IColElem is not used anymore. Use the Entity property.
VI.WebRuntime.IDbObjectRow .Collection	Removed because IColDbObject is not used anymore. Use the Entity property.
VI.WebRuntime.IDbObjectRow .GetDatabaseObjectDisplay(Boolean)	Call Entity.Display or Entity.LongDisplay.
VI.WebRuntime.IDbObjectRow .GetObjectWalker<T>(String)	Replaced by the extension method GetObjectWalkerAsync.
VI.WebRuntime.IDbObjectRow .HasSingleObject	Removed because IColDbObject is not used anymore. Use the Entity property.
VI.WebRuntime.IDbObjectRow .PermissionChanged(EventHandler <EventArgs<DataCell>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IDbObjectRow .SingleObject	Removed because IColDbObject is not used anymore. Use the Entity property.
VI.WebRuntime.IDbObjectRow.ColElem	Removed because IColElem is not used anymore.
VI.WebRuntime.IDbObjectRow.IsDeleted	Removed, use the Entity.IsDeleted property.
VI.WebRuntime.IDbObjectRow.IsLoaded	Removed, use the Entity.IsLoaded property.
VI.WebRuntime.IDbObjectRow.Save()	Removed. Use one of the Save extension methods in the VI.WebRuntime.Compatibility

Full Name	Migration path
	namespace.
VI.WebRuntime.IDbObjectRow.Saved (EventHandler)	Event replaced by <code>IAsyncObservable</code> object.
VI.WebRuntime.IForm.ListViewHost	Support for the <code>ListView</code> control has been removed.
VI.WebRuntime.IForm.RedirectToMenuItem (String, Boolean, IEnumerable<KeyValuePair<String, String>>)	Replaced by <code>RedirectToMenuItemAsync</code> .
VI.WebRuntime.IFormMethod.Execute (HttpContextBase, IFormRenderPart)	Replaced by <code>ExecuteAsync</code> .
VI.WebRuntime.IFormMethodAjax.Execute (IDictionary<String, String>, IFormRenderPart, IAjaxResponse)	Replaced by <code>ExecuteAsync</code> .
VI.WebRuntime.ILoadTableOptions .PrepareBulkLoad	Use <code>EntityCollectionLoadType.Bulk</code> to load complete entities.
VI.WebRuntime.IMenuItem.Tooltip	Changed to <code>get-only</code> for consistency reasons.
VI.WebRuntime.IModule .GetRequestParameter (String)	Moved to an extension method in <code>VI.WebRuntime</code> .
VI.WebRuntime.IModule.Recalculate()	Replaced by <code>RecalculateAsync</code> .
VI.WebRuntime.IoC.IObjectFactory .RouteHandlerFunc	Not intended to be called directly.
.WebRuntime.IoC.IObjectFactory FormFunc()	Not intended to be called directly.
VI.WebRuntime.IRuntimeTable .RowDeleted (EventHandler<EventArgs<IDataRow>>)	Event replaced by <code>IAsyncObservable</code> object.
VI.WebRuntime.IRuntimeTable .BulkOperation (EventHandler<EventArgs<IRuntimeTable>>)	Event replaced by <code>IAsyncObservable</code> object.
VI.WebRuntime.IRuntimeTable .DatabaseTable	Moved to the <code>IDbObjectStore</code> interface.
VI.WebRuntime.IRuntimeTable .DataStateChanged (EventHandler<EventArgs<IRuntimeTable>>)	Event replaced by <code>IAsyncObservable</code> object.
VI.WebRuntime.IRuntimeTable .Invalidated (EventHandler<EventArgs<IRuntimeTable>>)	Event replaced by <code>IAsyncObservable</code> object.
VI.WebRuntime.IRuntimeTable .ObjectSaved (EventHandler<EventArgs<IDataRow>>)	Event replaced by <code>IAsyncObservable</code> object.

Full Name	Migration path
VI.WebRuntime.IRuntimeTable .PermissionChanged(EventHandler <EventArgs<DataCell>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IRuntimeTable .RowDisplayChanged(EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IRuntimeTable .RowInserted (EventHandler<EventArgs <IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IRuntimeTable .RowInserting (EventHandler<EventArgs <IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IRuntimeTable .RowUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IRuntimeTable .RowUpdating (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IRuntimeTable.Delete (IDataRow)	Removed. Use one of the Delete extension methods in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.IRuntimeTable.Delete(Func <IDataRow, Boolean>, DeletionMode)	Removed. Use one of the Delete extension methods in the VI.WebRuntime.Compatibility namespace.
VI.WebRuntime.IRuntimeTable.Insert (ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.IRuntimeTable.Insert (ResultSet, Int32, ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.IRuntimeTable.SuspendBulkEv ents()	Return type changed to IAsyncActionBlock.
VI.WebRuntime.ISqlHelper .GetSqlSearchColumns(ITableDef)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.ISqlHelper .GetSqlSearchString(String, IColDef ,String)	Argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.ISqlHelper .GetSqlSearchString(String, IColDef ,String, Boolean, Boolean)	Argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.ISqlHelper .GetSqlSearchString(String, ITableDef ,String)	Argument type changed from ITableDef to IMetaTable.

Full Name	Migration path
VI.WebRuntime.ISqlHelper .GetSqlSearchString(String,ITableDef)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.ITablePermission . PermissionsChanged(EventHandler <EventArgs<ITablePermission>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ITablePermissions . PermissionsChanged(EventHandler <EventArgs<ITablePermission>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ITableStore . DataSourceChanged(EventHandler <EventArgs<IDataSourceBase>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ITableStore .NotifyDataSourceUpdated(IDataSourceBase)	Replaced by NotifyDataSourceUpdatedAsync.
VI.WebRuntime.ITableStore.Recalculate()	Replaced by RecalculateAsync.
VI.WebRuntime.ITableStore.SetValue (ITableStoreColumn,IDataRow,Object)	Replaced by SetValueAsync.
VI.WebRuntime.IValueChangeEvent .ValueChanged(RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IValueChangeEvent .ValueChanging(RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.IValueMetadata .ForeignKeyColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.IValueMetadata .ForeignKeyDynamicTables	Type changed to IEnumerable<IMetaTable>.
VI.WebRuntime.IValueMetadata .GetDataState (IDataRow)	Replaced by GetDataStateAsync.
VI.WebRuntime.IValueMetadata .GetDisplayValue(IDataRow)	Replaced by GetDisplayValueAsync.
VI.WebRuntime.IView.Invalidate()	Replaced by InvalidateAsync.
VI.WebRuntime.IView.Recalculate()	Replaced by RecalculateAsync.
VI.WebRuntime.JavascriptBaseFormMethod .Execute(HttpContextBase,IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.LoadTableOptions .PrepareBulkLoad	Use EntityCollectionLoadType.Bulk to load complete entities.
VI.WebRuntime.LocalColumn .RowInvalidated (EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.

Full Name	Migration path
VI.WebRuntime.LocalColumn.GetValue (IDataRow)	Replaced by GetValueAsync.
VI.WebRuntime.LocalColumn.RowUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.LocalColumn.SetValue (IDataRow,Object)	Replaced by SetValueAsync.
VI.WebRuntime.MemberRelation .ChangeMemberStatus(IDataRow,String ,Boolean)	Replaced by ChangeMemberStatusAsync.
WebRuntime.MemberRelation .MemberStatusChangedBulk (EventHandler<EventArgs<MemberRelation>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.MemberRelation .RaiseMemberStatusBulk()	Replaced by RaiseMemberStatusBulkAsync.
VI.WebRuntime.MemberRelation .ResetChanges()	Replaced by ResetChangesAsync.
VI.WebRuntime.MemberRelation..ctor (IConnection,IDataRow,IDataColumn ,DBObjectStore,String,String)	Argument type changed from IConnection to ISession and from IDataRow to IDbObjectRow.
VI.WebRuntime.MemberRelation.LoadMembers ()	Replaced by LoadMembersAsync.
VI.WebRuntime.MemberRelation.Owner	Type changed to IDbObjectRow.
VI.WebRuntime.MemberRelation.SaveChanges ()	Replaced by SaveChangesAsync.
VI.WebRuntime.MemberRelationColumn .CandidateTable	Type changed to string.
VI.WebRuntime.MemberRelationColumn .GetMemberRelation(IDataRow)	Replaced by GetMemberRelationAsync.
VI.WebRuntime.MemberRelationColumn .SetSource(String,String)	Replaced by SetSourceAsync.
VI.WebRuntime.MemberRelationColumn ColumnRelation	Type changed to VI.WebRuntime.ColumnRelation.
VI.WebRuntime.MenuItem.Tooltip	Type changed to Func<string>.
VI.WebRuntime.MenuItemBase .MenuItemClicked(EventHandler)	Event replaced by the IAsyncObservable property Click.

Full Name	Migration path
<EventArgs<MenuItemClick>>)	
VI.WebRuntime.MenuItemBase.Condition	Removed because the value was not used.
VI.WebRuntime.MenuItemBase.ExecuteAction (IForm,IDataRow)	Replaced by ExecuteActionAsync.
VI.WebRuntime.MenuItemBase.Redirect (IForm,IDataRow,Boolean,IEnumerable <KeyValuePair<String,String>>)	Replaced by RedirectAsync.
VI.WebRuntime.MenuStructure.FindItem (String)	Moved to an extension method in VI.WebRuntime.
VI.WebRuntime.Module.GetRequestParameter (String)	Moved to an extension method in the VI.WebRuntime namespace.
VI.WebRuntime.Module.Recalculate()	Replaced by RecalculateAsync.
VI.WebRuntime.ObjectImageFormMethod .Execute(HttpContextBase,IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.ObjectKeyBulkLoader .Dispose ()	This class does not implement IDisposable anymore; call LoadAsync instead to load the results.
VI.WebRuntime.ObjectKeyBulkLoader .InsertObjectKey(IDataRow ,ResolveMNOBJECTMODE,IDataColumn ,ColumnInsertValues)	Replaced by InsertObjectKeyAsync. The ResolveMNOBJECTMODE argument has been moved to the ResolveObjectMode property.
VI.WebRuntime.ObjectKeyBulkLoader .InsertObjectKey(ResolveMNOBJECTMODE ,DBObjectKEY,ColumnInsertValues,String)	Replaced by InsertObjectKeyAsync. The ResolveMNOBJECTMODE argument has been moved to the ResolveObjectMode property.
VI.WebRuntime.ObjectKeyBulkLoader..ctor (DBObjectStore)	Type changed to IDBObjectStore.
VI.WebRuntime.ObjectKeyBulkLoader.Table	Type changed to IDBObjectStore.
VI.WebRuntime.ObjectReferenceService .GetComponent(String,String)	Replaced by GetComponentAsync.
VI.WebRuntime.ObjectReferenceService .GetComponentName(String,DBObjectKEY)	Replaced by an extension method.
VI.WebRuntime.ObjectReferenceService .GetComponentName(String,String)	Replaced by GetComponentNameAsync.

Full Name	Migration path
VI.WebRuntime.ObjectReferenceService .Setup (IApplicationInfo)	Not intended to be called directly.
VI.WebRuntime.ObjectReferenceService. .ctor (IApplicationInfo)	Not intended to be called directly.
VI.WebRuntime.ObjectUpdater<TObj,TProp> .RaiseDataEventOccurred(IRuntimeTable ,IDataColumn,IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.ObjectViewProperty<T> . RowInvalidated(EventHandler <EventArgs<IDataRow>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ObjectViewProperty<T> . RowUpdated(RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ObjectViewProperty<T> .DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.ObjectViewProperty<T> .GetValue(IDataRow)	Replaced by GetValueAsync.
VI.WebRuntime.ObjectViewProperty<T> .RaiseRowUpdated(IDataRow,DbVal,DbVal)	Replaced by RaiseRowUpdatedAsync.
VI.WebRuntime.ObjectViewProperty<T> .SetValue(IDataRow,Object)	Replaced by SetValueAsync.
VI.WebRuntime.ObjectViewRow<T> . TransactionOpen	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.ObjectViewRow<T> .BeginTransaction()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.ObjectViewRow<T> .CommitTransaction()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.ObjectViewRow<T> .DatabaseTable	Moved to the IDbObjectTable interface.
VI.WebRuntime.ObjectViewRow<T> .RollbackTransaction()	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.ObjectViewTable<T> .Insert (ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.PopupHost. PopupClosed (EventHandler<EventArgs<PopupInfo>>)	Type changed to EventHandler<EventArgs<PopupCloseInfo>>.
VI.WebRuntime.PopupHost.ClosePopup()	Added argument of type

Full Name	Migration path
	HttpContextBase.
VI.WebRuntime.PopupInfo . PopupClosedWithPendingActions (EventHandler<EventArgs<PopupInfo>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.PopupInfo . PopupClosed (EventHandler<EventArgs<PopupInfo>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.RedirectUserMenu. Execute()	Method now returns a Task object.
VI.WebRuntime.ResultSetRow . RowValueUpdated (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ResultSetRow . RowValueUpdating (RowValueUpdateHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.ResultSetRow .TransactionOpen	Replaced by the IAsyncTransactable implementation.
VI.WebRuntime.RuntimeDisplayColumn .DatabaseSourceColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.Services .SessionTempFileHandler.ProcessRequest (HttpContextBase)	Replaced by ProcessRequestAsync.
VI.WebRuntime.SessionModule GetMenuStructure	Type changed to IMenuStructureProvider.
VI.WebRuntime.SessionModule .ReloadMenuStructure()	Replaced by ReloadMenuStructureAsync.
VI.WebRuntime.SqlHelper .GetDisplayPatternColumns(String ,ITableDef)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.SqlHelper .GetSqlSearchColumns(ITableDef)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.SqlHelper .GetSqlSearchString (String,IColDef ,String)	Argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.SqlHelper .GetSqlSearchString (String,IColDef ,String,Boolean,Boolean)	Argument type changed from IColDef to IMetaColumn.
VI.WebRuntime.SqlHelper .GetSqlSearchString (String,ITableDef ,Boolean,Boolean)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.SqlHelper .GetSqlSearchString (String,ITableDef ,String)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.SqlHelper .GetSqlSearchString (String,ITableDef ,String,Boolean,Boolean)	Argument type changed from ITableDef to IMetaTable.

Full Name	Migration path
VI.WebRuntime.SqlHelper .GetSqlSearchString (String,ITableDef)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.TablePermissionBase .PermissionsChanged(EventHandler <EventArgs<ITablePermission>>)	Event replaced by IAsyncObservable object.
VI.WebRuntime.TablePermissionBase .RaisePermissionsChanged()	Replaced by RaisePermissionsChangedAsync.
VI.WebRuntime.TimedAction.ExecuteAction	Type changed to Func<CancellationToken, Task>.
VI.WebRuntime.UpdateFormMethod.Execute (HttpContextBase,IFormRenderPart)	Replaced by ExecuteAsync.
VI.WebRuntime.UpdateFormMethod.Execute (IDictionary<String,String> ,IFormRenderPart,IAjaxResponse)	Replaced by ExecuteAsync.
VI.WebRuntime.UpdaterBase .RaiseDataEventOccurred(IRuntimeTable ,IDataColumn,IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.UserSession.Log(LogLevel ,String)	Removed; call the extension method VI.WebRuntime.IUserSession.GetLogSession to obtain a session-specific logger.
VI.WebRuntime.UserTable .GetLimitedSqlDataReader(String ,IEnumerable<QueryParameter>)	Replaced by GetLimitedSqlAsync.
VI.WebRuntime.UserTable. Loaded (EventHandler)	Event replaced by IAsyncObservable object.
VI.WebRuntime.UserTable.Load(IDataReader ,ILoadTableOptions)	Replaced by LoadAsync.
VI.WebRuntime.ValueMetadata .ForeignKeyColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.ValueMetadata .ForeignKeyColumnFunc(Func<IColDef>)	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.ValueMetadata .GetDisplayValue(IDataRow)	Replaced by GetDisplayValueAsync.
VI.WebRuntime.ValueMetadata.GetDataState (IDataRow)	Replaced by GetDataStateAsync.
VI.WebRuntime.VariableTable.Delete (IDataRow)	Replaced by DeleteAsync.

Full Name	Migration path
VI.WebRuntime.VariableTable.Insert (ColumnInsertValues)	Replaced by InsertAsync.
VI.WebRuntime.View.Invalidate()	Replaced by InvalidateAsync.
VI.WebRuntime.View.Recalculate()	Replaced by RecalculateAsync.
VI.WebRuntime.ViewUpdater .RaiseDataEventOccurred(IRuntimeTable ,IDataColumn,IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.WebControls.ActivatorControl.ActionToExecute	Return type changed to Func<CancellationTokentoken, Task>.
VI.WebRuntime.WebControls .ControlChangeUpdaterBase<T> .RaiseDataEventOccurred(IRuntimeTable ,IDataColumn,IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.WebControls .DialogImageSource..ctor(ITableDef ,ImageSize)	Replaced argument type ITableDef by IMetaTable.
VI.WebRuntime.WebControls .FilterByTableSelection .ParentConnectColumn	Type changed from IColDef to IMetaColumn.
VI.WebRuntime.WebControls .FilterByTableSelection.TargetTable	Type changed from ITableDef to IMetaTable.
VI.WebRuntime.WebControls .GridBlockContent.DataSourceBase	Type changed to IDataSourceLevel1.
VI.WebRuntime.WebControls .IFilterByTableSelection.TargetTable	Type changed from ITableDef to IMetaTable.
VI.WebRuntime.WebControls.IWebDesignerGeneratedControl.InitComponent	A second optional parameter of type CancellationTokentoken was added.
VI.WebRuntime.WebControls .RenderPartIterationUpdater .RaiseDataEventOccurred(IRuntimeTable ,IDataColumn,IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.WebControls .RenderPartUpdater .RaiseDataEventOccurred (IRuntimeTable ,IDataColumn,IDataRow)	Replaced by RaiseDataEventOccurredAsync.
VI.WebRuntime.WebControls.Angular .IMasterDetailControl .FixedDetailPane	Removed, not supported anymore.
VI.WebRuntime.WebControls.Angular .IMasterDetailControl .HasFixedOrientation	Removed because the orientation of the master-detail view is managed by CSS.

Full Name	Migration path
VI.WebRuntime.WebControls.Angular .IMasterDetailControl.Orientation	Removed because the orientation of the master-detail view is managed by CSS.
VI.WebRuntime.WebControls.Angular .MasterDetailControl .TabletScreensMediaQuery	Removed because the sizing of the master-detail view is managed entirely by CSS.
VI.WebRuntime.WebControls.Angular .MasterDetailControl.HasFixedOrientation	Removed because the orientation of the master-detail view is managed by CSS.
VI.WebRuntime.WebControls.Angular .MasterDetailControl.SmallScreensMediaQuery	Removed because the sizing of the master-detail view is managed entirely by CSS.
VI.WebRuntime.WebControls.Angular .MasterDetailControl.FixedDetailPane	Removed, not supported anymore.
VI.WebRuntime.WebControls.Angular .MasterDetailControl.Orientation	Removed because the orientation of the master-detail view is managed by CSS.
VI.WebRuntime.WebControls.BaseScripts. LocalIpForRemoteDebugging	Direct support for remote client debugging has been removed.
VI.WebRuntime.WebControls.BaseScripts EnableRemoteDebugging	Direct support for remote client debugging has been removed.
VI.WebRuntime.WebControls.TreeBlock .DataSource	Type changed from IHierarchyDataSource to IControlDataSource.
VI.WebRuntime.WebSql.GetMRCandidateTable (IConnection,ITableDef,String)	Argument type changed from ITableDef to IMetaTable.
VI.WebRuntime.WebSql.SendFileDownload (IFormRenderPart,String,String,Boolean ,String)	Additional optional parameter HttpContextBase appended.
VI.WebRuntime.WebSql.SendFileDownload (IFormRenderPart,String,String,Boolean ,Byte [])	Additional optional parameter HttpContextBase appended.
VI.WebRuntime.WebSql.WriteFile(String ,String)	Additional optional parameter HttpContextBase appended.

Changes in Web Designer and in Web Portal 7.0

This section provides an overview of changes in Web Designer and Web Portal 7.0 in comparison to previous versions.

Migrating customized projects

Loading objects that were migrated from versions prior to 6.1

Web Designer 7.0 supports loading objects that were last saved in Web Designer 6.1. However, when attempting to load an object that was last saved in a version prior to 6.1, you may see the following error message:

“The model version <X> of the object <Y> is unknown. Please verify that you are using the correct product version for your database.”

To migrate these objects, execute the following steps

1. Open Web Designer and authenticate to the database.
2. Select the menu item **Migration assistant**.
3. In the migration assistant, there will be some items that start with **Migrate XML document**. Select these items and click **Run selected tasks**.
4. You are now able to load the objects in Web Designer.

Table-dependent references

Table-dependent references have been replaced by object-dependent references. This may affect all web projects that were defined in versions up to (and including) 6.1.

To create object-dependent references for these web projects, execute the following steps

1. Open Web Designer and authenticate to the database.
2. Select the menu item **Migration assistant**.
3. In the migration assistant, there will be some items that start with **Create object-dependent references**. Select these items and click **Run selected tasks**.

Handling compiler error messages

This section provides details about some common error messages that you may see when first compiling a web project with custom components.

Message: The return type of the function <x> is ambiguous.

Details: All functions must have well-defined return types. In some cases, the types of the argument must also be made explicit.

Solution: Double-click the error message to select the function definition. Select the proper return data type of the function.

If the function has any arguments, define the argument type in the function signature. For example, if the function has the signature `Process(inputParameter)`, add the argument type **string**. The new function signature reads: `Process(string inputParameter)`.

Message: There is no cursor defined on the collection <x>

Details: In previous versions, it was allowed for cursors to be used implicitly. All cursors must now be explicitly declared.

Solution: You may see this message when you are referencing an implicit cursor below one of these node types:

- Virtual node mapping
- Local component
- Function

Double-click the error message to select the node, then look for the closest parent node that corresponds to one of these node types. Note that this can be the node itself. Right-click the node and add a cursor declaration for the collection.

Message: The component must implement the interface <x>.

Details: In Web Designer 8.1.2, components that are used as object-dependent references or column editors must explicitly implement the corresponding component interface.

Solution: Double-click the error message to select the node. Look for the **Component interface** node and select it. Assign the interface that is named in the error message.

Message: The module <x> was not found

Details: Some modules have been removed from the Web Portal.

Solution: Remove the reference to the module.

Message: The function {0} is not marked as a property iteration function.

Details: Functions that are used in property iterations must be explicitly marked as such.

Solution: Double-click the error message to select the function definition. Mark the function as a property iteration function.

Changes to Web SQL functions

Removed Web SQL functions

- The function `GetToXML` has been removed.
- The function `Eval` has been removed.

Semantic changes

The following Web SQL functions have changed their internal implementation. This may cause unexpected behavior in custom components that rely on details of the old implementation.

- The function `TranslateColumn` does not recognize `%` as a place holder character anymore. This change was made to increase performance for the general case, where place holders are not required.
- The function `Concat` now behaves differently when the input set is empty. In the case where the input does not contain any rows, the result is not `NULL` but an empty string.
- The function `GetViPrimaryKey` has been removed. It is no longer necessary to handle the special case where an artificial internal primary key was created for database objects. Use the function `PrimaryKey` instead.

Changed features

Execution of arbitrary SQL statements

Due to security considerations, it is not recommended to execute arbitrary SQL statements from the web application layer. When the web application is installed on top of the new application server, execution of arbitrary SQL statements generates an exception.

This change affects all Database query type collections defined in Web Designer components.

All SQL statements must be encapsulated as **predefined SQL statements** (QBMLimitedSQL-type objects) and stored in the database. Refer to the One Identity Manager Configuration Guide for more information on how to define predefined SQL statements.

For additional security, predefined SQL statements must be assigned to permission groups. Execution of predefined SQL statements always occurs with the permissions of the user who is logged in to the web application can.

Note that WHERE clauses, as well as the WebSQL functions Exists and GetSingleProperty, are not considered arbitrary SQL statements and may be used as before.

Removed features

- The drop down container control has been removed.
- The drop down button control does not support sub menu items anymore.
- The control events Click a control, Move parent container, Expand/collapse grids and Flush client commands have been removed.
- The property Allow special HTML sequences on the label control has been removed.
- The property Skip condition for this band on the grid control has been removed.
- The property Filter condition for nodes to be expanded on the grid control has been removed.
- On the Fill collection action, the option Load candidates for M:N relation has been removed.
- Special pages that were defined as raw HTML are no longer supported. Specifically, the login page, logout page and the error page cannot be defined as raw HTML.
- Configuration parameters with the type Expression are no longer supported.
- The Report Viewer node type does not support directly exporting a PDF file. To directly transmit a compiled report as a PDF file, see the component VI_Common_ShowReport for an implementation example.
- The ability to record and run scripts in Web Designer has been removed.

Property write protection

The semantics of the write protection functionality has been modified.

When defining a write protection on a property, the write protection will affect all controls bound to that property. Previously, only controls defined in the sub tree of the write protection node were affected. This change was made to reduce ambiguity in some situations, and to allow for cleaner separation of UI and permission-related code.

If you have both read-only and read-write use cases in a component, you can define a so-called proxy property to build a read-only proxy property.

Boolean values and NULL

In versions up to 6.1, the internal data storage allowed for Boolean-typed values to be NULL. This is no longer the case, and the C# non-nullable type System.Boolean is now used.

WebSQL expressions such as IsNull(BooleanFunction(), true) are now considered invalid and will generate the following compiler error:

"The expression is of type System.Boolean and is never NULL."

Modified session timeout handling

Previous versions included a heartbeat functionality to keep a user's session open as long as the browser window was open. This was implemented by sending a ping from the browser to the server. As this may cause heavy server load in certain scenarios, this feature has been removed.

As a consequence, the session timeout configured for the web application will now cause inactive sessions to time out, even if the user's browser window is still open.

Selectable IT Shop products

There are two minor changes to the requestable products filter in the Web Portal, managed by the configuration key `VI_ITShop_AccProduct_Filter`.

- By default, IT Shop products (AccProduct) that are not assigned to any service category are not displayed as requestable products (e.g. the delegation product).
- Assignment resources (objects in the QERAssign table) are displayed as requestable products in the IT Shop.

Object collection and selection components

As mentioned above, the grid and tree controls now support true on-demand loading of data.

In previous versions, it was necessary to load a collection before displaying its content in a grid or a tree. This is no longer necessary. If you configure a control to use the database as its data source, the control will then automatically load the required data. (The control will never remove objects from the collection, it only adds new ones.)

- The grid control implements page-based loading. By default, only the first page of objects is loaded. Grouping works the same way: only the data required for the selected group is loaded.
- The tree control implements hierarchical loading. By default, only the first level of the hierarchical structure objects is loaded. When expanding a tree node, the next level of the data structure is loaded.

It is important to note two consequences of this new mode.

- At any time, the collection will only hold a subset of the logical data set. Depending on how the user has configured paging and filtering on the control, the collection will only hold the data which is displayed. For example, the logical data set could consist of all PersonWantsOrg objects for a specific employee. If you configure a database-

bound grid, only PersonWantsOrg required to display the first grid page will be present in the collection after the page is first rendered. If you need to perform an operation on all PersonWantsOrg objects, you will need to manually load them.

- Data events will be raised for every load operation. Specifically, insert and bulk-load events will be raised whenever a paging, sorting or filtering operation causes new objects to be added to the collection.

The standard components have been changed as follows in order to benefit from performance improvements.

- Object collection components are now expected to load the data that they need to display. There is no need to load data into a collection before the collection is given to an object collection component. Instead, just supply the `TableName()` and `DbWhereClause()` functions and the object collection component will load and display the data accordingly.
- Object selection components do not actually load the data anymore. Object selection components are expected to provide a WHERE clause, without loading data.

Object selection and collection components are designed to work well with each other. The WHERE clause supplied by the object selection component can be directly mapped to the object collection component, so that a change to the WHERE clause triggers a refresh of the data display.

The standard component `VI_Edit_FK` provides an implementation example of this model.

For more information about the data source API, refer to the documentation for the `VI.WebRuntime.Data.DbDataSource` class.

Removed control layout properties

The following control layout properties are no longer supported. References to these properties will result in Web Designer compilation errors. In some cases, the properties have been renamed and references will need to be updated to reflect the new name.

Control	Property
Grid band	ImagePagingBack
	ImagePagingForward
	CsvSeperator
	Image_SortDesc
	Image_SortAsc
	Image_SortNone
	PagesInSelector
	CssClassNameContainerResizeInactive
	CssClassNameContainerResize

Control	Property
	UseSplitTables
	IconPlus
	IconWhite
	IconMinus
	RenderSortingWithCss
	CssClassNameSortImage
	CssClassNameNaviCell
	CssClassNameAutoFilterText
	CssClassNameFilter
	InteriorNaviCell
	AutoFilterMode
	AutoNormalizeLayout (is now called DoNotAutoNormalizeLayout)
Tree band	TreeLevelCssClassPrefix
	LineMode
	ShowRootLines
	IndentationWidth
	CssClassNameContent
	CssClassNameBlockDiv
Label	AppendImage
Button	AppendImage
	CssClassNameContainer
Table	WrapInContainer
	CssClassNameWrapper
Container	Wrap (is now called AutoGenSubContainers)
TextBox	CssClassNameWatermark
	CssClassNameTextarea
	CssClassNameMandatory (is now called CssClassNameIfMandatory)
	CssClassName (is now called CssClass)
Drop down list	CssClassName (is now called CssClass)
Radio button	CssClassName (is now called CssClass)
Date control	CssClassNameTableNavigation

Control	Property
	CssClassNameTableDays
	CssClassNameHeaderRow
	CssClassNameCellDefault
	CssClassNameCellSelected
	CssClassNameCellCurrent
	CssClassNameCellCurrentAndSelected
	CssClassNameDayLink
	CalendarTable
	DateSelectBox
	CalendarNavigationlink
	CalendarTableDays
	CalHeaderCell
	CalHeaderRow
	ButtonCSSName
	NoEnterDateManual
	CssClassNametextBox
	CssclassNameButton
	CssClassNameCalendar
Layout panel	RowCssClass (is now called CssClassNameRow)
	CellCssClass (is now called CssClassNameCell)
	RenderMode (is now called IsTableMode)
Tab control	CssClassNameNaviCell
	RenderMode

Changes to standard objects

Removed configuration keys

The following configuration keys have been removed.

Configuration key	Remarks
VI_History_StateOverview_Person_MembershipTables	The list of tables is now

Configuration key	Remarks
	defined as a function.
VI_History_StateOverview_Person_EntitlementTables	The list of tables is now defined as a function.
VI_Reporting_DefaultTemplate	The name of the default report template is now stored in the configuration parameter QER RPS DefaultReportTemplate .
VI_Common_Color_SmallButton_Font	The configuration keys have been removed due to layout changes.
VI_Common_Color_LinkAndHeader	
VI_Common_Color_Link_Hover	
VI_Common_Color_Gradient_WebPart1	
VI_Common_Color_Gradient_WebPart2	
VI_Common_Color_Gradient_Generic1	
VI_Common_Color_Gradient_Generic2	
VI_Common_Color_HeadingColor	
VI_Common_Color_Border_LightGray	
VI_Common_Color_Border_DarkGray	
VI_Common_Color_Border_Gray	
VI_Common_Color_Border_Blue	
VI_Common_Color_Border_Header	
VI_Common_Color_Border_SmallGroup	
VI_Common_Color_Border_SeparatorLine	
VI_Common_Color_PagingComponent_Border	
VI_Common_Color_VeryLightGray	
VI_Common_Color_Deactivated	
VI_Common_Color_Background_Signaled	
VI_Common_Color_ShapeSignaled	
VI_Common_Color_FKControl_LimitExceeded	
VI_Common_Color_Hyperview_Separator	
VI_Common_Color_Combobox_Item_Hover_Background	
VI_Common_Color_Combobox_Item_Hover_Font	
VI_Common_Color_ErrorBox_TitleFont	
VI_Common_Color_UserNotification_Background	

Configuration key**Remarks**

VI_Common_Color_UserNotification_Border
VI_Common_Color_Background_Calendar_Day
VI_Common_Color_Background_Calendar_Header
VI_Common_Color_Font_Calendar_Header
VI_Common_Color_Font_Calendar_Day_Hover
VI_Common_Color_TreeItemSelected_or_Hover
VI_Common_Color_TreeItemHover_Border
VI_Common_Color_DropDown_Calendar_Background
VI_Common_Color_Popup_Background
VI_Common_Color_Popup_Border
VI_Common_ChartTitle
VI_Common_Chart_Background
VI_Common_Color_GridRow_Hover_Background
VI_Common_Color_GridRow_StaticBelow_Background
VI_Common_Color_GridHeader_Font
VI_Common_Color_GridHeader_Border
VI_Common_Color_GridTable_Border
VI_Common_Color_Tab_Inactive_Gradient_Color1
VI_Common_Color_Tab_Inactive_Gradient_Color2
VI_Common_Color_Tab_Inactive_Background
VI_Common_Color_Tab_Inactive_Font
VI_Common_Color_Tab_Border
VI_Common_Color_Tab_Active_Background
VI_Common_Color_Tab_Active_Link
VI_Common_Color_Tab_Link
VI_Common_Color_NoScript_Link
VI_Common_Color_NoScript_ButtonFont
VI_Common_Color_Dojo_DropDownButton_Hover
VI_Common_Color_Dojo_DropDownButton_Focused
VI_Common_Color_Dojo_Menu_Border
VI_Common_Color_Dojo_Menu_Background
VI_Common_Color_Dojo_Menu_Focused_Font
VI_Common_PleaseWait_Border

Configuration key

Remarks

VI_Common_Color_Menu_Font
VI_Common_Color_MenuOpenItems_Font
VI_Common_Color_ToolbarButton_Font
VI_Common_Color_ColorBaseHighlight
VI_Common_Color_Button_Border
VI_Common_Color_Button_Font
VI_Common_Color_Button_Font_Secondary
VI_Common_Color_Button_Font_Disabled
VI_Common_Color_Button_Blue_Background_Default1
VI_Common_Color_Button_Blue_Background_Default2
VI_Common_Color_Button_Blue_Background_Hover1
VI_Common_Color_Button_Blue_Background_Hover2
VI_Common_Color_Button_Blue_Background_Disabled
VI_Common_Color_Button_Blue_Border_Focused
VI_Common_Color_Button_Silver_Background_Default1
VI_Common_Color_Button_Silver_Background_Default2
VI_Common_Color_Button_Silver_Background_Hover1
VI_Common_Color_Button_Silver_Background_Hover2
VI_Common_Color_Button_Silver_Background_Disabled
VI_Common_Color_Button_Silver_Border_Focused
VI_Common_Color_Button_Silver_Border_default

Removed modules

Module name

Description

VI_ActionHistory_Main	All modules corresponding to overview pages in version 6.1 have been removed from the application.
VI_Approvals_Main	
VI_Audit_Main	
VI_BusinessOwner_Main	
VI_Compliance_Main	The navigation control has been updated to allow for direct access to all menu items.
VI_Compliance_Policies_Main	
VI_Compliance_RuleViolations_Main	
VI_Employee_Main	

Module name	Description
VI_Infosystem_Main	
VI_ITShop_Main	
VI_MyData_Main	
VI_MyData_MyPassword_Main	
VI_QAM_Taxonomy_Main	
VI_RelatedApplications_Main	
VI_Infosystem_DashboardWithObject	The Infosystem module has been removed from the application.
VI_Reporting_Subscription	The module is not referenced anymore. Instead of referencing this module, use a reference to the component VI_Reporting_Subscriptions.
VI_Roles_SPSResources	Not required anymore.
VI_QAM_CategorizationManager	Not required anymore.

Removed components

Component name	Description
VI_Attestation_Display_WizardParameter_Domain	The attestation policy wizard has been updated to use more generic components. As a result, these components are not required anymore.
VI_Attestation_Edit_WizardParameter_AERole	
VI_Attestation_Edit_WizardParameter_Department	
VI_Attestation_Edit_WizardParameter_Domain	
VI_Attestation_Edit_WizardParameter_ESet	
VI_Attestation_Edit_WizardParameter_Locality	
VI_Attestation_Edit_WizardParameter_Org	
VI_Attestation_Edit_WizardParameter_Person	
VI_Attestation_Edit_WizardParameter_ProfitCenter	
VI_Attestation_Edit_WizardParameter_QAMCategory	
VI_Attestation_Edit_WizardParameter_QAMClassification	
VI_Attestation_Edit_WizardParameter_QAMDuG	
VI_Attestation_Edit_WizardParameter_QAMTrustee	

Component name	Description
VI_Attestation_Edit_WizardParameter_UNSAccount	
VI_Attestation_Edit_WizardParameter_UNSContainer	
VI_Attestation_Edit_WizardParameter_UNSGroup	
VI_Attestation_Edit_WizardParameter_XProxyContext	
VI_Common_HyperView_Default	This component is now named VI_HyperView_Default.
VI_Common_HyperView_Ressource	The Ressource table has been replaced with the new tables QERResource, QERReuse, QERReuseUS and QERAssign.
VI_Common_ObjectSelection_Default	This component is now named VI_ObjectSelection_Default.
VI_Common_ObjectSheet_Default	This component is now named VI_ObjectSheet_Default.
VI_Common_SearchPerson	These components implement object searching, which has been replaced as much as possible by flexible filtering. As a result, these search components are not required anymore.
VI_Common_SearchPersonPopup	
VI_Common_SelectDocumentNumber	
VI_Common_SelectorHardware	
VI_Common_SelectorPerson	
VI_QAM_QAMDUG_Search	
VI_QAM_QAMDUG_SearchListing	
VI_Reporting_SelectorPerson	
VI_Common_UserMessages	The notification messages are now rendered in the form type objects. This component is not used anymore.
Vi_Delegation_GridColumns_SecondLevel	The component is not used anymore.
VI_Edit_FK_Generic	The component is not used anymore.
VI_Edit_FK_Generic_Tree	The component is not used anymore.

Component name	Description
VI_Edit_Special_AttestationPolicy_UID_AttestationObject	The component is not used anymore.
VI_Edit_Special_URL	The component is not used anymore.
VI_Common_ObjectQuickInfo	The component is not used anymore.
VI_Infosystem_ChartWithObject	The info system menu item has been removed. See the new component VI_Common_ChartWithObject.
VI_Infosystem_DashboardLoadData	The info system menu item has been removed. See the new component VI_Common_DashboardLoadData.
VI_Infosystem_Dashboard_List	The info system menu item has been removed. See the new component VI_Common_Chart.
VI_ITShop_ObjectDisplayCheckStatus	
VI_ITShop_PwoDecisionHistory	The request workflow history view has been updated. This component is not used anymore.
VI_ObjectEdit_Ressource	Removed because the Ressource table has been removed from the data model.
VI_ObjectEdit_QAMClassificationRule	Classification rule editing is not implemented in the classification web server component.
VI_ProductEditor_Clone_ShoppingCartItem_Default VI_ProductEditor_Details_AccProduct_Default VI_ProductEditor_Details_PersonWantsOrg_Default VI_ProductEditor_Details_PersonWantsOrg_NewManager VI_ProductEditor_Details_ShoppingCartItem_Default VI_ProductEditor_Insert_ShoppingCartItem_Default	The names of these components have changed. The "ProductEditor_" part of the component names has been removed to improve component name consistency.

Component name	Description
VI_ProductEditor_Verify_ShoppingCartItem_Default	
VI_ProductEditor_Verify_ShoppingCartItem_RequestNewGroup	
VI_QAMDUG_Search	The component is not used anymore.
VI_QAM_Dashboard_CategoryAndFileType	This component is now named VI_QCS_Dashboard_CategoryAndFileType.
VI_QAM_Dashboard_ResourceByFileType	This component is now named VI_QCS_Dashboard_ResourceByFileType.
VI_QAM_QAMDUGExplorer	The component is not used anymore.
VI_QAM_QAMDUGOmnibox	The component is not used anymore.
VI_QAM_QAMDUG_PathLoader	The component is not used anymore.
VI_QAM_ResourceAccessRequests	The component is not used anymore.
VI_QAM_RiskIndexEditor	The component is not used anymore.
VI_Common_LanguageSelector	Removed because the application now supports multiple languages.

Removed virtual functions

Component	Virtual function	Remarks
VI_Common_CompactLabel	UsageKey()	Not required anymore.
VI_Edit_Multiselect	IntroducingText()	Not required anymore.
VI_Edit_Multiselect	UsageKey()	Not required anymore.
VI_Edit_Multiselect_SelectRecursive	UsageKey()	Not required anymore.
VI_Edit_MR	RemoveOldCandidates()	Not required anymore.

Component	Virtual function	Remarks
VI_Attestation_History_Filter	popupTitle()	Not required anymore.
VI_Common_DecisionControl	CanDecide()	
VI_Common_DecisionControl	AllowApprove()	Now stored as a flag on the EditTable collection.
VI_Common_DecisionControl	AllowDeny()	Now stored as a flag on the EditTable collection.
VI_Common_PropertyEditor	SizeType()	Use the LayoutPanelLayoutName() function to dynamically assign different property editor layouts.
VI_Responsibilities_Memberships	externalFilter()	Not required anymore.
VI_Responsibilities_Memberships	externalOrgRootFilter()	Not required anymore.
VI_Action_AdditionalApprover	PopupStyles()	Not required anymore.
VI_Action_DenyApproval	PopupStyles()	Not required anymore.
VI_Action_InsteadOfApprover	PopupStyles()	Not required anymore.
VI_ITShop_AccProductGroupList	PersonForOrder()	Removed as it is now possible to make a request for more than one employee.
VI_ITShop_AccProductGroupSelector	PersonForOrder()	Removed as it is now possible to make a request for more than one employee.
VI_ITShop_CartPatternApply	PersonForOrder()	Removed as it is now possible to make a request for more than one employee.
VI_ITShop_PersonWantsOrg_GridColumns	DisplayPersonInserted() ShowCancelOrder() ShowRequestInformation()	Not supported anymore.

Component	Virtual function	Remarks
VI_ITShop_ApprovalItem	AdditionalFilter()	Not supported anymore.
VI_ITShop_AssignmentInfo	CustomStyles()	Not supported anymore.
VI_Attestation_Display_WizardParameter_Generic	OWExpressionDisplay() ColumnNamePK() TableName() MultiLine()	Not required anymore.
VI_ITShop_CartCalculateCheckStatus	ITShopOrgForShoppingCartItem() ()	Not required anymore.
VI_Popup	PopupStyles()	Use the CcssClass() function to dynamically assign a CSS class to the popup.
VI_ITShop_WebPart_Top5Attestation	DecisionType() HeadLine() HideDecisionButton()	Removed because the component does not directly trigger actions anymore.
VI_ITShop_WebPart_Top5Decisions	DecisionType() HeadLine() HideDecisionButton()	Removed because the component does not directly trigger actions anymore.
VI_ITShop_CartOverview	ShowHeader() ShowToForLater()	Removed because these actions are implemented in the detail pane, not in the grid view.
VI_MessageBox	MessageBoxStyle()	Removed property, use the IsDestructiveAction() function to switch between different styles.
VI_ITShop_PwoOverviewGrid	PWOWhereClause()	Renamed to DbWhereClause() for consistency.
VI_ITShop_PwoOverviewGrid	NoItemsText() TitleText() ShowCancelOrder()	Not required anymore.
VI_Common_HyperView_ShapeList	ShapeBaseColor()	The ability to set colors for individual HyperView shapes has been

Component	Virtual function	Remarks
		removed. Use the ColorKey() function instead.
VI_Common_HyperView_Shape	ShapeBaseColor()	The ability to set colors for individual HyperView shapes has been removed. Use the ColorKey() function instead.
VI_Attestation_AttestationCase	DisplayLinkToApprovalPage()	Not required anymore.
VI_ITShop_AddCartItem	autoITShopOrg()	Not required anymore.
VI_ITShop_AccProduct_GridColumns	HideServiceCategory()	The column is not displayed by default.
VI_Roles_EntitlementDetective	TypeFilter()	Not required anymore.
VI_Roles_EntitlementUsageDetective	EvaluationContext()	Not required anymore.
VI_QAM_ResourceView	Filter()	Replaced by the virtual function QamDugWhereClause().
VI_Compliance_ExceptionApprovalGrid	showaudit() setuseraudit() showhistory()	Not required anymore.

Removed virtual nodes

Component	Node identifier	Remarks
VI_ITShop_CartOverview	MoveToForLater MoveToOrder	Removed because actions are implemented in the detail pane, not in the grid view.
VI_ITShop_PersonWantsOrg_GridColumns	AdditionalActions	Removed because actions are implemented in the detail pane, not in the grid view.
VI_ITShop_PwoOverviewGrid	AdditionalActions	Removed because actions are implemented in the detail pane, not in the grid view.

Component	Node identifier	Remarks
VI_ITShop_ QueryToPerson	OnAfterQueryToPerson	
VI_Roles_ AttestationCasesForObject	HeadLineIfData	

Removed virtual collections

These virtual collections have been removed. You can safely remove any mapping collections that refer to these collections.

Component	Collection name
VI_ITShop_ApprovalItem	Accproduct Pwodecisionhistory Pwohelperpwo pwodecisionstep
VI_ITShop_PersonWantsOrg_GridColumns	Accproduct Pwodecisionhistory pwohelperpwo
VI_ITShop_PersonWantsOrg_Information	Pwodecisionhistory
VI_Attestation_AttestationCase	AttestationPolicy
VI_Attestation_History_Filter	AttestationCase AttestationPolicy
VI_Compliance_ExceptionApproval_Filter	Personinnoncompliance Compliancerule
VI_Compliance_ExceptionApprovalGrid	compliancerule
VI_Compliance_GridColumn_RuleSeverity	compliancerule
VI_Compliance_RuleSeverityIndicator	compliancerule
VI_ITShop_AccProductGroupList	Accproductcandidate
VI_ITShop_AccProductGroupSelector	Accproductcandidate parentvars
VI_ITShop_CartOverview	Accproduct Personwantsorg Compliancechecksimpleobjects

Component	Collection name
	compliancechecksimple
VI_ITShop_OrderHistory_Filter	Accproductgroup Personwantsorg
VI_ITShop_PwoOverviewGrid	Pwodecisionhistory Accproduct
VI_QERPolicy_ExceptionApproval_Filter	Qerpolicyhasobject
VI_UNNS_RequestNewGroup	shoppingcartitem
VI_QAM_RequestAccess	shoppingcartitem

Removed virtual properties

Component	Property	Remarks
VI_ITShop_AccProductGroupList	Anzahl	Not required anymore.
VI_ITShop_AccProductGroupSelector	Anzahl Showonlythis canorderforsomebodyelse	Not required anymore.
VI_Common_DecisionHistory	DecisionRule	Renamed to UID_DecisionRule for consistency reasons.
VI_ITShop_PersonWantsOrg_GridColumns	UID_AccProduct	Not required anymore.
VI_ITShop_PwoOverviewGrid	UID_AccProduct	Not required anymore.
VI_Compliance_ExceptionApproval_Filter	DisplayStatus DisplayNonCompliance	Not required anymore.
VI_Edit_Multiselect	UsageKey	Not required anymore.
VI_Delegation_HistoryGrid	TableName	Not required anymore.
VI_ITShop_CalculatePWOComplianceCheck	DbObjectKey1 DbObjectKey2	Replaced by the new property DbObjectKey.
VI_Reporting_ParameterEdit	HasPredefinedRange QueryWhereClauseResulting	Not required anymore.
Vi_Delegation_GridColumns_	Display	Not required anymore.

Component	Property	Remarks
FirstLevel	TooltipDisplay	
VI_Common_CompactLabel	Display	Not required anymore.
VI_MyData_ChallengeLossRoleMembership	uid_accproduct checkstatus	Not required anymore.
VI_Roles_RoleAssignmentNew	pkselected display uid_unscontainer tablename	Not required anymore.

About us

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales and other inquiries, such as licensing, support, and renewals, visit <https://www.oneidentity.com/company/contact-us.aspx>.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at <https://support.oneidentity.com/>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product

Copyright 2020 One Identity LLC.

ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of One Identity LLC .

The information in this document is provided in connection with One Identity products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of One Identity LLC products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, ONE IDENTITY ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ONE IDENTITY BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ONE IDENTITY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. One Identity makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. One Identity does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

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

Refer to our Web site (<http://www.OneIdentity.com>) for regional and international office information.


Patents


One Identity is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <http://www.OneIdentity.com/legal/patents.aspx>.

Trademarks

One Identity and the One Identity logo are trademarks and registered trademarks of One Identity LLC. in the U.S.A. and other countries. For a complete list of One Identity trademarks, please visit our website at www.OneIdentity.com/legal. All other trademarks are the property of their respective owners.

Legend

-  **WARNING: A WARNING icon highlights a potential risk of bodily injury or property damage, for which industry-standard safety precautions are advised. This icon is often associated with electrical hazards related to hardware.**

-  **CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.**

One Identity Manager Web Designer and Web Portal Changes
Updated - January 2020
Version - 8.1.2