

Quest® Unified Communications Analytics
8.4.2
Resource Kit



© 2017 Quest Software Inc.

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 Quest Software Inc.

The information in this document is provided in connection with Quest Software 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 Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE 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 QUEST SOFTWARE 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 QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software 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. Quest Software 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:

Quest Software Inc.
Attn: LEGAL Dept.
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our website (www.quest.com) for regional and international office information.


Patents

Quest Software 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 www.quest.com/legal.

Trademarks

Quest and the Quest logo are trademarks and registered trademarks of Quest Software Inc. in the U.S.A. and other countries. For a complete list of Quest Software trademarks, please visit our website at www.quest.com/legal. Microsoft, Active Directory, ActiveSync, Excel, Lync, and Skype are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. All other trademarks, servicemarks, registered trademarks, and registered servicemarks are the property of their respective owners.

Legend

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

 **IMPORTANT NOTE, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

Contents

Using the resource kit to extract data	4
Using the bulk data exporter	4
Listing all available entities and fields	5
Common entities in the UC Analytics data model	5
Examples of bulk exporter commands	5
Why would the SendDate field be empty?	6
Why would fields that link to Sender not be set?	6
Using Microsoft Power BI to explore the OData endpoint	7
Prerequisites	7
Sample scenarios	8
About query performance	12
About columns with multiple values	12
Limitations	12
Using the OData client library	13
Prerequisites	13
About the sample files	13

Using the resource kit to extract data

UC Analytics provides a resource kit to give you easier access to the data you have collected. The resource kit includes the following tools:

- A bulk data exporter that allows you to retrieve large amounts of data from UC Analytics. It allows you to export your collected data to a .csv file. The bulk data exporter is automatically installed with the UC Analytics Data Engine.

[Using the bulk data exporter](#)

- An Open Data Protocol (OData) endpoint. You can access the OData endpoint using third party OData tools such as Microsoft Power BI or PowerView.

[Using Microsoft Power BI to explore the OData endpoint](#)

- An OData client library. You can use the OData library to create ad hoc queries using PowerShell or C#.

[Using the OData client library](#)

Using the bulk data exporter

Using a command line interface, you can use the bulk exporter to extract specific data from your UC Analytics collected data. You run the bulk exporter on the directory that hosts the Data Engine.

To run the bulk exporter

- 1 Navigate to the Data Engine directory located in the UC Analytics installation directory on the server on which the Data Engine is installed.
- 2 Enter the executable name (UC.Analytics.Insights.DataEngine.BulkDataExport.exe) followed by the parameters that meet your inquiry.

Table 1. Parameters that can be used with the bulk exporter.

Available parameters	Definition
/FilePath	File path and file name of the output file
/ColumnDelimiter	Default character is , (comma). You can specify a different delimiter character (optional)
/Entity	Target entity name such as EmailMessage
/Fields	Comma separated field names such as SendDate or Sender.EmailAddress.Person.Name
/ObfuscatedFields	Comma separated field names (optional)
/Filter	Filter expression (optional)
/StartDate	Start date (optional)
/EndDate	End date (optional)
/PageSize	Buffer size (The default value is 1000)

Table 1. Parameters that can be used with the bulk exporter.

Available parameters	Definition
/MaximumNumberOfResults	Number of objects (optional)
/OrderAscending	Sets the order of the results. True or false (default value is false)
/OrderByField	Field name (optional)
/ShowDataModel	Provides a complete up-to-date list of all the entities, fields, and their data types. Cannot be combined with other parameters

Listing all available entities and fields

To see a list of all the available entities, fields and their data types, use the /ShowDataModel switch on the command line and send the output to a file. For example, you could enter the following command:

```
D:\Program Files\Quest\UC Analytics\Data Engine>  
UC.Analytics.Insights.DataEngine.BulkDataExport.exe /ShowDataModel > dataModel.txt
```

i | **TIP:** When you enter parameters, remember that field names are case-sensitive and must exactly match the names as shown by the /ShowDataModel switch.

Common entities in the UC Analytics data model

The following is a list of common entities that comprise the UC Analytics data model:

- EmailMessage
- ActiveDirectoryObjectSnapshot
- LyncConference
- LyncPeerToPeerSession
- LyncQoeSession
- CiscoConference
- CiscoPeerToPeerSession

Additionally, you can retrieve the full metadata from OData ([http://<server name>:1336/\\$metadata](http://<server name>:1336/$metadata)).

The resource kit supports obfuscated data so that you can do data analysis without revealing private data. Data obfuscation is a form of data masking where data is purposely scrambled to prevent unauthorized access to sensitive materials.

Examples of bulk exporter commands

Your query can be simple or complex, depending on the depth of data that you want to extract.

In a simple example, you can extract the subject, send date, and sender name for all the email message in your database:

```
UC.Analytics.Insights.DataEngine.BulkDataExport.exe /Entity EmailMessage /FilePath  
output.csv /Fields "Subject, SendDate, Sender.EmailAddress.Person.Name"
```

In a more complex example, you could filter the query to include only the subjects from email messages that were sent by Stuart Murgal as of January 1, 2015.

```
UC.Analytics.Insights.DataEngine.BulkDataExport.exe /Entity EmailMessage /FilePath
output.csv /Filter "Sender.EmailAddress.Person.Name == \"Stuart Mughal\" "/Fields
"Subject" /StartDate "2015-01-01"
```

Why would the SendDate field be empty?

Why would some fields such as SendDate be empty?

Answer

The Send Date is affected by the types of data collections you have scheduled, the sources from which you are collecting, and by the message origins. If you only collect from the Exchange Tracking Logs data source, but do not collect from mailboxes using the Exchange Mailbox Contents data source, the Send Date may be approximate or missing for some messages. For example, the SendDate may not be set if you are not collecting all the tracking logs from all your Exchange mailbox and hub servers.

If you collect from both the Exchange Mailbox Contents and the Exchange Tracking Logs data sources, the Send Date is always set by the Mailbox Contents data collection. To always collect the exact Send Date, you must collect using both the Tracking Logs and the Mailbox Contents data sources.

When Send Date is approximate

If you collect only from the Exchange Tracking Logs data source, the Send Date for SMTP or inbound messages is approximate. For messages submitted through SMTP or originating from outside your Exchange organization, the Delivery Time is not calculated until the Exchange Calculation job has run.

For multi-hop message delivery, a message might appear in the tracking logs on several Exchange servers. Since the Send Date will be set by an event (StoreDriver Submit or SMTP Receive) which can appear in any one of the tracking logs, the Send Date is approximate.

Why would fields that link to Sender not be set?

Sometimes Sender and fields that follow links such as Sender.DeliveryTime are not set. Why?

Answer

The SenderDeliveryTime field is set only if senders send the message to themselves. The RecipientsDeliveryTime field is usually set.

The SenderFirstResponseTime is the duration between the receipt of the message, and the first reply to the message. So the field is mainly applicable to Recipients (rather than Sender) and is set only for messages that are collected from mailboxes using the Mailbox Contents data source (not Exchange Tracking Logs).

The SenderSendTimeOfDay and SentAfterHours fields are also set only for messages collected using the Mailbox Contents data source.

If you want the sender name rather than the key, you can use Sender.EmailAddress.DisplayName field instead.

Using Microsoft Power BI to explore the OData endpoint

You can use Microsoft Power BI Desktop to access the UC Analytics OData endpoint to extract and view your collected data. For performance reasons, it is recommended that you always edit and filter the query before loading the data. For example, you might filter by limiting the query to a short period of time or limiting the query to selected people.

The workflow for using Power BI can be divided into the following procedures:

- 1 Connecting to OData.
- 2 Selecting and downloading the data to your local computer.
- 3 Creating a visualization of the data.

The procedure to connect to OData is provided followed by a sample scenario to show the procedures for selecting and downloading data and for creating a visualization.

Prerequisites

You must install Microsoft Power BI Desktop on your computer. You can download Power BI Desktop from the following Microsoft web site: <https://powerbi.microsoft.com>

You also must have unrestricted access to all the data (configured in the Security settings in the UC Analytics Admin Setting) that you want to extract. For information about granting access to the UC Analytics data, see the *UC Analytics Deployment Guide*.

To connect to OData

- 1 Open Power BI Desktop.
- 2 On the home ribbon, click **Get Data**.

Data types are organized in the following categories:

- All
- File
- Database
- Azure
- Other

- 3 Select the **All** category which includes all data connection types from all categories.
- 4 In the All list, scroll down to locate the OData Feed.
- 5 Select **OData Feed** and click **Connect**.
- 6 Under URL, enter the URL for the server on which the UC Analytics Data Engine is installed and specify port 1336.

For example, for server MyAnalytics, you would enter: `http://MyAnalytics:1336`

- 7 Click **OK**.

Power BI Desktop makes the connection to OData Feed and, in the Navigator, presents a list of the available tables.

TIP: Do not select the Load button at the bottom of the Navigator pane before you have edited your query. If you simply select a table and click **Load**, all the data in the table is loaded into local RAM on your computer. For example, if you selected **EmailMessages** and clicked **Load**, data for every email message stored in UC Analytics would be downloaded to your computer. The download could take a very long time.

- 8 To edit the query before loading data, select the **Edit** button.

Sample scenarios

The following sample scenarios explain how to download selected data and to create visualizations using Power BI Desktop. By following the steps in these scenarios, you can become familiar with some of the Power BI Desktop functions.

For more information about using Power BI Desktop, see the knowledge base at the Microsoft Power BI support site: <https://support.powerbi.com/knowledgebase>.

Using Power BI Desktop to recreate a graph from the Email - Activity insight

The following example shows you how to recreate a graph from the Email - Activity insight that shows the number of email messages over time. To recreate the graph, you need two fields from the EmailMessages table: Key and Timestamp.

To select email message data over time

- 1 After you have connected to OData and displayed the tables in the Navigator, select **EmailMessages** and click **Edit**.
- 2 In the Query Editor ribbon, click **Choose Columns**.
- 3 Clear (**Select All Columns**).
- 4 Select **Key** and **Timestamp**, and click **OK**.
- 5 Right-click **Timestamp** and select **Change Type**.
- 6 Select **Date**.

Email messages are stored using the exact date and time of each message. You change the Timestamp column from Date/Time to Date so that messages can be aggregated by day to produce a reasonable line graph.

- 7 In the Query Editor ribbon, click **Close & Apply**.

Your changes are saved. When you first load data in Power BI Desktop, you will see Report View with a blank canvas. The Visualizations and Fields panes should be displayed on the right.

Optional: Using the Advanced Editor

If you want to see the code that Query Editor is creating with each step, or want to create your own code, you can use the Advanced Editor.

To launch the advanced editor, select **View** from the ribbon and select **Advanced Editor**. A window appears, showing the existing Query code.

For example, you could select the Key and Timestamp fields from the EmailMessages table using the Advanced Editor by entering the following sample query.

To enter the selection code in the advanced editor

- 1 After you have connected to OData and displayed the tables in the Navigator, right-click **EmailMessages** and select **Advanced Editor**.
- 2 Paste the following query into the Advanced Editor:

```
let
    Source = OData.Feed("http://MyAnalytics:1336"),
```



```

EmailMessages_table =
Source{ [Name="EmailMessages",Signature="table"]} [Data],

#"Removed Other Columns" = Table.SelectColumns(EmailMessages_table, {"Key",
"Timestamp"}),

#"Changed Type" = Table.TransformColumnTypes("#Removed Other
Columns",{{"Timestamp", type date}})

in

#"Changed Type"

```

- 3 Click **Done**.

To create a graph that shows email messages over time

After you have downloaded the email message data by date, you can create a visualization of the data.

- 1 In the Visualizations pane, select the line graph icon.
Under the Visualizations pane, a secondary pane shows the line graph parameters such as Axis, Legend, and Values.
- 2 Drag the **Key** and **Timestamp** fields, listed in the Fields pane, to the Axis and Values labels respectively.
The resulting line graph should closely resemble the graph in the Email - Activity insight.

Using Power BI Desktop to create a graph for specific user email activity

The following example shows you how to recreate a graph that shows the number of email messages over time for a specific user. You need to use two different tables: EmailMessages and EmailMessageParticipants.

To select email message data for a specific user over time

- 1 After you have connected to OData and displayed the tables in the Navigator, select **EmailMessages** and **EmailMessageParticipants** and click **Edit**.
- 2 When the Query Editor has the EmailMessages table in preview, in the Query Editor ribbon click **Choose Columns**.
- 3 Clear (**Select All Columns**).
- 4 Select **Key** and **Timestamp** and **Sender**, and click **OK**.
- 5 Right-click **Timestamp** and select **Change Type**.
- 6 Select **Date**.
Email messages are stored using the exact date and time of each message. You change the Timestamp column from Date/Time to Date so that messages can be aggregated by day to produce a reasonable line graph.
- 7 Click the down arrow at the top right of the column header to expand the **Sender** column.
- 8 Clear (**Select All Columns**).
- 9 Select **Key**.
- 10 When the Query Editor has the EmailMessageParticipants table in preview, in the Query Editor ribbon click **Choose Columns**.
- 11 Clear (**Select All Columns**).
- 12 Select **Key** and **EmailAddress** and click **OK**.
- 13 Click the down arrow at the top right of the column header to expand the **EmailAddress** column.

- 14 Select **DisplayName**.
- 15 Right-click on the **DisplayName** column and rename the column to **Participant Name**.
- 16 In the Query Editor ribbon, click **Close & Apply**.

Your changes are saved. When you first load data in Power BI Desktop, you will see Report View with a blank canvas. The Visualizations and Fields panes should be displayed on the right.

- 17 In the Power BI ribbon, click **Manage Relationships** and select **Autodetect...**

This step detects the relationships between the data.

To create a graph that shows email messages for a specific user

After you have downloaded the email message data by date and the email participant data, you can create a visualization of the data.

- 1 In the Visualizations pane, select the line graph icon.

Under the Visualizations pane, a secondary pane shows the line graph parameters such as Axis, Legend, and Values.
- 2 Drag **Key** and **Timestamp** fields, listed in the Fields pane, from the EmailMessages entity into the Values and Axis fields respectively.
- 3 Drag **Participant Name** field, listed in the Fields pane, from the EmailMessageParticipants entity into the Legend field for the line graph.

You should see the email activity trends for all the participants.
- 4 To analyze a specific participant, expand Participant Name under the Filters, and select the name of the participant that you want to see.

The resulting line graph should show the email activity over time for a specific user.

Using Power BI Desktop to create a bar chart for Exchange public folder item counts

The following example shows you how to create a vertical bar chart for public folders that shows the number of items in each public folder on a particular day.

To select public folder item count for a specific date

- 1 After you have connected to OData and displayed the tables in the Navigator, select **ExchangePublicFoldersSnapshot** and click **Edit**.
- 2 In the Query Editor ribbon, click **Choose Columns**.
- 3 Clear (**Select All Columns**).
- 4 Select **Key**, **Fullpath**, **ItemCount**, **Name**, and **SnapshotDate** and click **OK**.
- 5 Right-click **SnapshotDate** and select **Change Type**.
- 6 Select **Date**.

Messages are stored in public folders using the exact date and time of each message. You change the SnapshotDate column from Date/Time to Date so that messages can be aggregated by day to produce a total item count for the specified date.

- 7 At the top of the SnapshotDate column, click the down arrow to see a list of snapshot dates.
- 8 Select the snapshot date that you want.
- 9 In the Query Editor ribbon, click **Close & Apply**.

Your changes are saved. When you first load data in Power BI Desktop, you will see Report View with a blank canvas. The Visualizations and Fields panes should be displayed on the right.

To create a bar chart that shows public folder item count for a specific date

After you have downloaded the public folder items and snapshot date, you can create a visualization of the data.

- 1 In the Visualizations pane, select the vertical bar chart icon.
Under the Visualizations pane, a secondary pane shows the bar chart parameters such as Axis and Values.
- 2 Drag the **Name** and **ItemCount** fields, listed in the Fields pane, to the Axis and Values labels respectively.
- 3 Under Filters, click the down arrow to expand **Name(All)**.
- 4 Locate **IPM_SUBTREE** and clear the check box.
- 5 Under Visualizations, click the paintbrush icon to display customization options.
- 6 Set Data Labels to **On**.

The resulting bar chart will show all the public folders and the number of items in each for the selected day.

Using Power BI Desktop to create a trend for item counts in an Exchange public folder

The following example shows you how to create a trend that shows the number of items in a specific public folder over time.

To select public folder item counts over time

- 1 After you have connected to OData and displayed the tables in the Navigator, select **ExchangePublicFoldersSnapshot** and click **Edit**.
- 2 In the Query Editor ribbon, click **Choose Columns**.
- 3 Clear (**Select All Columns**).
- 4 Select **Key**, **Fullpath**, **ItemCount**, **Name**, and **SnapshotDate** and click **OK**.
- 5 Right-click **SnapshotDate** and select **Change Type**.
- 6 Select **Date**.

Messages are stored in public folders using the exact date and time of each message. You change the **SnapshotDate** column from Date/Time to Date so that messages can be aggregated by day to produce a total item count for the each date.

- 7 In the Query Editor ribbon, click **Close & Apply**.

Your changes are saved. When you first load data in Power BI Desktop, you will see Report View with a blank canvas. The Visualizations and Fields panes should be displayed on the right.

To create a trend that shows the item counts in a public folder over time

After you have downloaded the public folder item count data, you can create a visualization of the data.

- 1 In the Visualizations pane, select the line graph icon.
Under the Visualizations pane, a secondary pane shows the line graph parameters such as Axis, Legend, and ItemCount.
- 2 Drag the **SnapshotDate**, **Name**, and **ItemCount** fields, listed in the Fields pane, to the Axis, Legend, and ItemCount labels respectively.
- 3 Under Filters, click the down arrow to expand **Name(All)**.
- 4 Clear (**All**) and check the public folder name that you want to trend.
- 5 Under Visualizations, click the paintbrush icon to display customization options.
- 6 Set Data Labels to **On**.

The resulting trend graph will show the public folder you selected and the number of items in the folder for each day in the snapshot dates.

About query performance

If you query for a large amount of data from the tables, the query can take a significant amount of time to complete. It is recommended that, when you edit queries, you only select the fields that you need.

Table 2. Examples of load times for different query sizes.

Number of Rows	Number of Columns	Query Size (MB)	Query Load Time (seconds)
30,000	3	4.5	13
30,000	23	18	47

About columns with multiple values

By default, in OData each row contains a single entity such as one message, one conference, and so on. So for columns that can contain multiple values (such as participant), if you select Count for the Count of Key, you will see duplicate values.

For example, if you selected participants and a message has several participants, the same message appears in multiple rows, one entry for each participant. In this case, change the Count of Key from Count to Count (Distinct) to get a count of distinct keys.

Limitations

Only scalar values can be expanded. This means that if you attempt to expand a nested collection or a dynamic field, you get an error or null values.

Workaround

To get around the limitation of being unable to expand a nested collection, you could query and download the primary entity and the entity you want to expand. Then you can allow Power BI to autodetect the relationship between the entities using foreign keys.

An example of this workaround is shown in the scenario [Using Power BI Desktop to create a graph for specific user email activity](#) on page 9.

In this scenario `Participants.EmailAddress.Name` could not be expanded since `EmailAddress` is a collection of a collection (`Participants`). So how do we get `EmailAddress.DisplayName`? We must also query for the `EmailMessageParticipants` entity which has the property `EmailAddress`. Here we can expand and get `EmailAddress.DisplayName` because `DisplayName` is a scalar property of `EmailAddress`.

Using the OData client library

Using the OData client library you can create ad hoc queries using PowerShell or C#. The OData client library complements the bulk data exporter but is slower and is meant to support a smaller amount of data. Unlike the bulk data exporter, you can create queries to get data from calculated fields (such as direction or chargeback amounts) and you can also retrieve aggregated data.

While the bulk data exporter must be run locally, the OData library is secure and can be run remotely.

The OData client library is installed in the Resource Kit folder on the server that hosts the Data Engine. By default, the path is as follows:

```
C:\Program Files\Quest\UC Analytics\Resource Kit
```

In the Resource Kit folder are the .dll files that you must copy to the server from which you want to run your queries.

Prerequisites

You access the OData library using the .NET interface and must be proficient in either PowerShell or C# to write your own queries.

About the sample files

In the Resource Kit folder there is a folder named Samples that contains examples of both PowerShell and C# queries. For example, the AggregateQuerySamples files for C# and for PowerShell (.cs and .ps1) provide samples that show you how to create queries for aggregated data such as:

- the number of Inbound messages sent to a specific department
- the number of messages received by a specific email address grouped by the offices that sent the most messages
- the average message delivery times grouped by country
- the number of messages sent to a distribution group, grouped by month, for a defined number of months

To see individual data records, the ObjectQuerySamples files provide a sample query to show a list of the individual inbound messages sent to a specific department.

To get a dump of all the entities and fields that comprise the UC Analytics data model, you can use the DataModelSamples files.