

QMATIC

MANUAL

DATA CONNECT

272.01B

Copyright notice

The information in this document is subject to change without prior notice and does not represent a commitment on the part of Q-MATIC AB.

All efforts have been made to ensure the accuracy of this manual, but Q-MATIC AB can not assume any responsibility for any errors and their consequences.

This manual is copyrighted and all rights are reserved.

Reproduction of any part of this manual, in any form, is not allowed, unless written permission is given by Q-MATIC AB.

COPYRIGHT © Q-MATIC AB, 2023.

ⓘ Please send any feedback or questions about this manual to documentation@qmatic.com.

Contents

1. What's new.....	1
2. Introduction	3
2.1. Overview	4
2.2. Requirements	5
2.3. Licensing	5
2.4. Installation components	5
3. Installation	7
3.1. Install Data Connect API	8
3.2. Install Data Connect App	11
3.3. Access rights	12
4. Using the data.....	13
4.1. Data time ranges	14
4.2. Import data from Power BI Desktop	14
4.3. Modeling	19
4.4. Quota	19
5. Exploring the data	21
5.1. Overview	22
5.2. Getting started	23
5.3. Exploring OData	24
5.4. OData Primitive Data Types	25
6. API Reference	27
6.1. Appointment	29
6.2. AppointmentArrivalStatus	34
6.3. AppointmentCancelledDate	35
6.4. AppointmentCreatedDate	37
6.5. AppointmentCreatedTime	40
6.6. AppointmentCustomer	42
6.7. AppointmentEndTime	43
6.8. AppointmentService	45
6.9. AppointmentStartTime	46
6.10. AppointmentStatus	48
6.11. Branch	49
6.12. CallTime	51
6.13. Date	53

6.14. DeliveredService	56
6.15. DeliveredServiceTime	60
6.16. EndTime	62
6.17. EnteredQueueTime	64
6.18. EntryPoint	66
6.19. FeedbackAnswer	67
6.20. FeedbackDate	68
6.21. FeedbackQuestion	71
6.22. FeedbackResult	72
6.23. FeedbackTime	75
6.24. Outcome	77
6.25. QmaticBiContainer	79
6.26. QuestionType	83
6.27. Queue	84
6.28. QueueType	85
6.29. Resource	86
6.30. Service	87
6.31. ServicePoint	88
6.32. Staff	89
6.33. StaffSession	90
6.34. StaffSessionEndTime	93
6.35. StaffSessionOutcome	95
6.36. StaffSessionStartTime	96
6.37. Visit	98
6.38. VisitCallType	100
6.39. VisitCustomer	101
6.40. VisitTransaction	102
6.41. VisitTransactionOutcome	109
6.42. WorkProfile	111

1. What's new

Version	Chapter	Change
01.A	• General	• Manual for Data Connect Created

2. Introduction

Topics

Overview	4
Requirements	5
Licensing.....	5
Installation components	5

2.1 Overview



With Data Connect, you get access to the customer journey management data from the convenience of your own chosen data visualization tool.

The data is exposed as an API which means you can import data and start creating meaningful dashboards right away. The data is safely encrypted during transit and you have full control over who has access to Data Connect through an administrative portal to make sure that your data does not end up in the wrong hands.

The API enables advanced analytics and storage in your system. You can incrementally pull all your data into your systems for further processing.

There are many existing client libraries for most platforms and programming languages which will reduce the development time required to build and maintain such integrations.

2.1.1 Supported data visualization tools

You can use any data visualization tool, as long as it supports Odata source.

Examples of supported tools are PowerBI, Tableau, Excel, Qlikview.

2.1.2 Supported OS and databases

To see the supported operating systems and databases, please refer to the supported operating systems and databases for Qmatic Orchestra, found in the Qmatic Orchestra datasheet.

2.2 Requirements

- Orchestra 7.3 original release, or later, is required.
- A signed SSL certificate is required for Data Connect to support visualization tools like Power BI.

2.3 Licensing

To use Data Connect, the Data Connect OnPrem license is required.

2.4 Installation components

For Data Connect, you need to install the following two components:

- Data Connect API: this is the Data Connect backend.
- Data Connect App: this is the user interface.

Both components require some manual procedures, see “3. Installation” on page 7.

3. Installation

Topics

Install Data Connect API.....8

Install Data Connect App11

Access rights12

3.1 Install Data Connect API

This section describes how to install the Data Connect backend.

When you copy a command from this document, make sure to remove any line breaks to have the whole command on one row before executing it.

3.1.1 Extract ZIP file (Windows only)

If using Windows, extract Data_Connect_Api.zip to a suitable directory (e.g. C:/qmatic/data-connect).

3.1.2 Create user and user permitted directory (Linux only)

If using Linux, create user and user permitted directory.

1. Run the commands below to create a qmatic user and the directory where Data Connect API should be installed. If you already have a qmatic user, you don't have to run the two first commands (adduser and passwd).

```
sudo adduser qmatic
sudo passwd qmatic
sudo mkdir -p /opt/qmatic/data-connect
```

2. Extract the Data_Connect_Api zip into the directory you created and run the command below.

```
sudo chown -R qmatic:qmatic /opt/qmatic/data-connect
```

3. Go to /opt/qmatic/data-connect directory and untar the java for Linux distribution as the qmatic user.

```
su qmatic -c "tar -xvf zulul7.42.19-ca-jre17.0.7-
linux_x64.tar"
```

3.1.3 Update application.yml

Some mandatory changes need to be made to application.yml. Locate the conf/application.yml file and change the settings according to the instructions below.

Server ports

Update http-port and https-port in section server with the desired ports.

```
server:
  http-port: 8088
  https-port: 8448
```

SSL

SSL is enabled by default to support tools like Power BI. Provide a signed certificate and configure the credentials in the section below. To disable SSL, set `require-ssl` to `false`.

❗ Refer to the Secure Communication section of the Orchestra Reference manual for more details.

```
server:
  ssl:
    enabled: true
    key-store: ${APP_HOME}/conf/security/keystore.jks
    key-store-password: changeit
    key-store-type: jks
    key-alias: orchestra
    key-password: changeit
  security:
    require-ssl: true
```

StatDB

Make sure the correct database section is uncommented (Postgres, Oracle, MSSQL) and update credentials and other settings to match your configuration.

```
insapi:
  db:
    driverClassName:com.microsoft.sqlserver.jdbc.SQLServerDriver
    url: jdbc:sqlserver://
    localhost:1433;databaseName=statdb
    username: stat
    password: stat
```

Orchestra Endpoint

If Orchestra is running on the same server, e.g. `http://127.0.0.1:8080`:

```
environment:
  orchestra:
    hostName: 127.0.0.1 #Can use localhost instead of
    127.0.0.1
    port: 8080
    protocol: http
```

If Orchestra is running on a different server, e.g. `https://10.20.0.1:8443`

```
environment:
```

```
orchestra:
  hostName: 10.20.0.1 #Can use host name of the server
                    instead of the IP
  port: 8443
  protocol: https
```

3.1.4 Install and start Windows service

1. Locate the bin\install-dataconnect.bat file in bin directory.
2. Run it with administrator privilege to install the Windows service.
3. Open Services in Windows and check if the new service called Qmatic Data Connect API is installed.
4. Start the service.

3.1.5 Install and start Linux service

1. If needed, give executable permission to the following files in the bin directory:

- install-data-connect.sh
- data-connect
- run.sh
- stop.sh

For example (assuming Data_Connect_Api.zip was unzipped to /opt/qmatic/data-connect):

```
sudo chmod u+x /opt/qmatic/data-connect/bin/install-
data-connect.sh
```

2. Locate the bin/install-data-connect.sh file in bin directory and run it with sudo, which copies a startup script to init.d.

❗ The library chkconfig needs to be installed as a prerequisite before you run this script

3. To start / restart / stop the service, execute following commands.

```
sudo /etc/init.d/data-connect start
sudo /etc/init.d/data-connect restart
sudo /etc/init.d/data-connect stop
```

❗ If there is any problem during the installation, you can check the logs in the directory where data-connect is extracted in the logs folder

3.2 Install Data Connect App

The Data Connect app consists of a landing page with links to data feeds. To access the data feeds, Data Connect API needs to be installed first.

When you copy a command from this document, make sure to remove any line breaks to have the whole command on one row before executing it.

This section describes how to manually install Data Connect App. If you install from Xtend, go to step 4.

❗ If you are running Orchestra 7.4 Original Release, please contact support to get help with the installation.

1. Copy data-connect.war to <Orchestra_install_dir>\system\custdeploy
2. If you are running Orchestra 7.4 original release or earlier, add the properties below to applicationMessages.properties

```
application.data-connect = Data Connect  
  
application.data-connect.description = Data Connect data  
feeds
```

3. Download the data_connect_app_db.zip and execute scripts for the correct database. Which scripts to use depends on the Orchestra version:
 - If you are running Orchestra 7.4 update 1 or later, use **db/updates** scripts.
 - If you are running Orchestra 7.4 original release or earlier, use **db/inserts** scripts.

4. Locate the shiro.ini file in /conf directory and add the following entry under the URL section:

```
/data-connect/** = saml2SecurityFilter, qpAuthc,  
modules[data-connect]
```

❗ When upgrading from Orchestra 7.4 Original Release to a later release, you might already have a row with data_connect in shiro.ini. If so, you need to update the row to 'data-connect'.

5. To point the Data Connect App to the Data Connect API, configure the environment variable by adding the variable below to the file standalone.bat (for Windows) or standalone.sh (for Linux) in <orchestra-home>/app/wildfly<version>/bin directory. Add it before debug configuration, i.e. before the row "By default debug mode is disabled".

Windows:

```
set DATA_CONNECT_BASE_URL=https://<host>:<port>/data-  
connect/v1
```

Linux:

```
export DATA_CONNECT_BASE_URL=https://<host>:<port>/  
data-connect/v1
```

Replace host and port with the correct values.

6. Restart Orchestra central.
7. Set up access rights for users and roles in Orchestra > **User Management**.
See “3.3 Access rights” on page 12.

3.3 Access rights

Users of Data Connect need to be assigned a role with the following access modules:

- Data Connect
- Connector ServicePoint

As from Orchestra 7.4, there is a predefined role, Data Connect, which has the correct modules. For Orchestra 7.3, you need to add a role called Data Connect and assign it with the modules above.

4. Using the data

Topics

Data time ranges 14

Import data from Power BI Desktop..... 14

Modeling 19

Quota 19

4.1 Data time ranges

There are two OData URLs that give you the same data structure and same schema, but with different data ranges.

The OData URLs can be found in the Data Connect landing page. There are two URLs, one for short term data, and one for long term data.

The screenshot shows a light blue background with two white rounded rectangular boxes. The top box is titled 'Short term data' and contains the text: 'Short term data URL gives you data from the last 60 days. This is more suitable for near real time reporting or dashboards. It is also suitable for short term analysis.' Below this text is a text input field containing the URL 'https://localhost:8080/onyxdev/v1/short-term-data' and a green button with a copy icon and the text 'COPY LINK'. The bottom box is titled 'Long term data' and contains the text: 'The long term data URL gives you data from a time range between 3 years before the start of the current year until the current day. This is more suitable for long term analysis and reporting.' Below this text is a text input field containing the URL 'https://localhost:8080/onyxdev/v1/long-term-data' and a green button with a copy icon and the text 'COPY LINK'.

4.1.1 Short term data URL

Short term gives you data from the last 60 days. This is more suitable for near real time reporting or dashboards. It is also suitable for short term analysis.

Example URL: <https://localhost:8080/onyxdev/v1/short-term-data>

4.1.2 Long term data URL

The long term URL gives you data from a time range between 3 years before the start of the current until the current day. This is more suitable for long term analysis and reporting. This means if the current date is 2022-05-01, the long term URL will give you data from 2019-01-01 until 2022-05-01.

Example URL: <https://localhost:8080/onyxdev/v1/long-term-data>

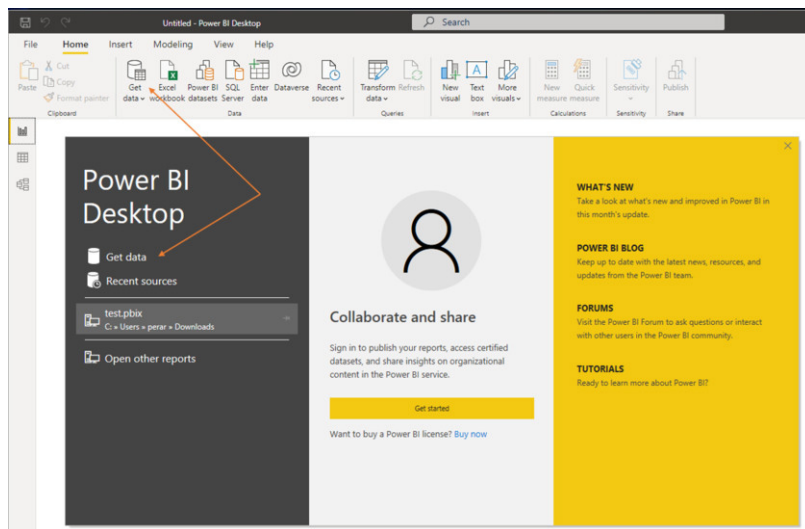
4.2 Import data from Power BI Desktop

1. Start the Power BI Desktop application from the start menu.

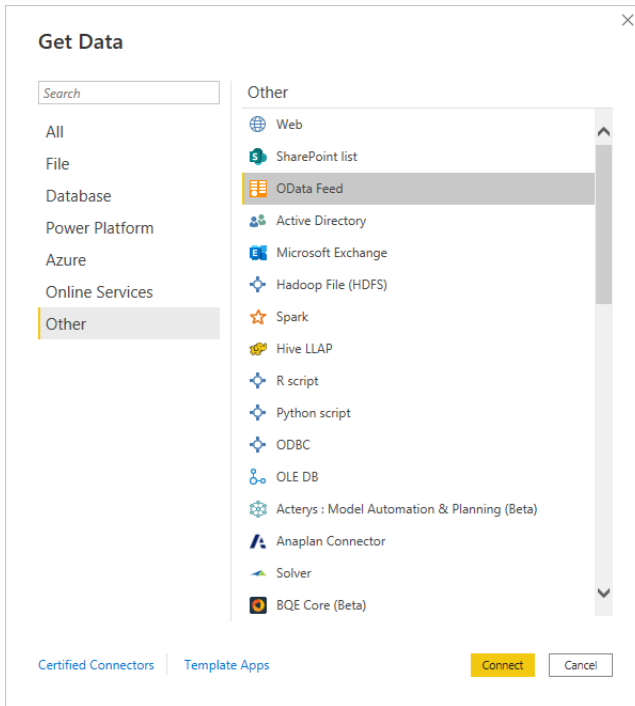


Power BI Desktop

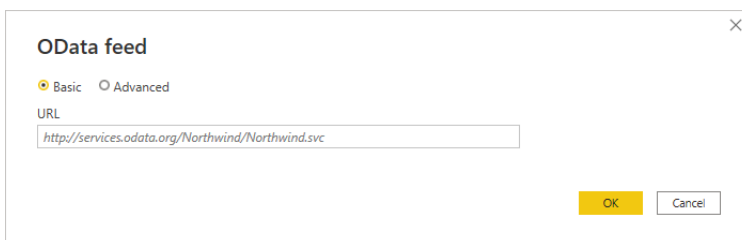
2. From the splash screen, click **Get data**.



3. To select your data source, select **Other > OData Feed**. Click **Connect** to proceed.



4. Enter the url to the API. If you are prompted to choose authentication, select **Basic**.



5. Select the check boxes next to the tables that you want to have imported to Power BI. When ready, click **Load** to start loading your data.

Navigator

Display Options ▾

- http://api-test-link [17]
- ☒ Branches
- ☒ Dates
- ☒ EntryPoints
- ☒ Genders
- ☒ Outcomes
- ☒ Queues
- ☒ QueueTypes
- ☒ ServicePoints
- ☒ Services
- ☒ Staffs
- ☒ Times
- ☒ VerticalMessages
- ☒ VisitCallTypes
- ☒ VisitOutcomes
- ☒ Visits
- ☒ VisitTransactions
- ☒ WorkProfiles

WorkProfiles

OriginId	EndDate	ClientKey	BranchOriginId	Name
6	null	1	1	Max Visit Lifetime
4	null	1	1	Appointment
5	null	1	1	Max Waiting Time
2	null	1	1	Information
1	null	1	1	Money
6	null	1	2	Max Visit Lifetime
1	null	1	2	Money
2	null	1	2	Information
7	null	1	3	WS_ALL
2	null	1	3	Information
7	null	1	1	WS_ALL
3	null	1	1	Safe Deposit
3	null	1	3	Safe Deposit
5	null	1	3	Max Waiting Time
2	null	1	4	Information
6	null	1	4	Max Visit Lifetime
6	null	1	3	Max Visit Lifetime
4	null	1	4	Appointment
5	null	1	2	Max Waiting Time
2	null	1	5	Information
6	null	1	5	Max Visit Lifetime
5	null	1	5	Max Waiting Time
4	null	1	5	Appointment

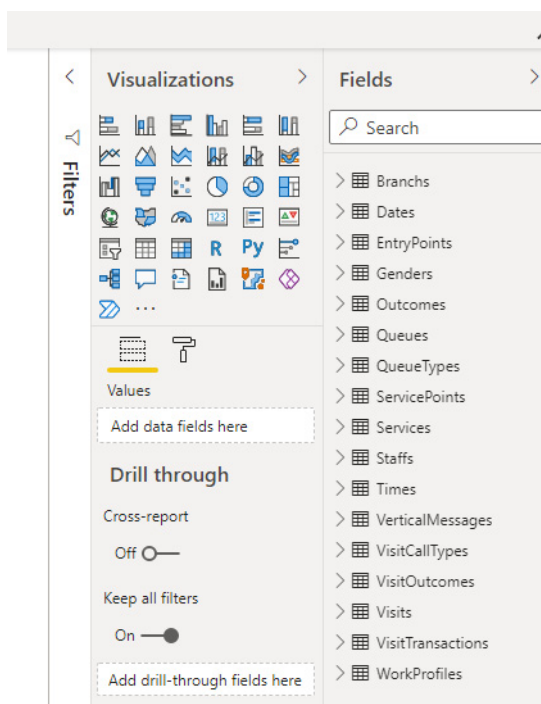
Select Related Tables

Load

Transform Data

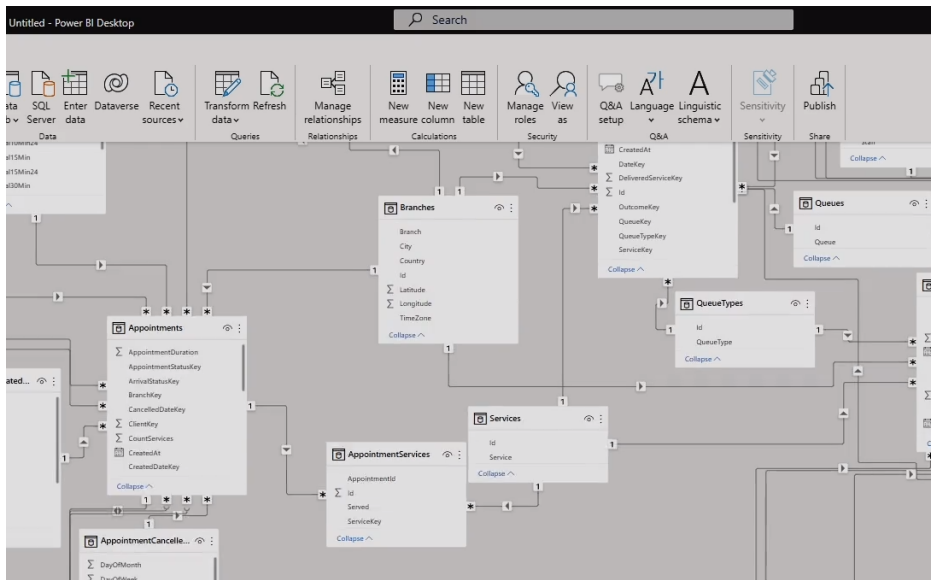
Cancel

6. Your data is now imported and you can start creating dashboards.



4.3 Modeling

Once you have imported the data, you will get a complete model of the data itself, so there is no need to create models.



4.4 Quota

There is a quota implemented in terms of the number of entities that you can fetch during a period of 24 hours. This protects the system from being overloaded. The current quota is 5 000 000 entities per 24 hours. An entity is for example

- one visit
- one branch
- one service
- etc.

If you exceed the quota, you will get the HTTP 429 error with an error message stating how long it will take until your quota is reset.

5. Exploring the data

Topics

Overview	22
Getting started	23
Exploring OData	24
OData Primitive Data Types	25

5.1 Overview

The core technology for Data Connect is the OData protocol, which is a standardized protocol with the ISO and OASIS organizations. It follows the RESTful principles of API design and is therefore very easy to adopt for developers that have worked with REST APIs. The difference between a typical REST implementation and OData is that it standardizes schemas, metadata and interactions. The closest technology would be a database protocol. You connect to a service and get the schema. From that schema, you can do querying and most CRUD operations.

5.1.1 Encryption

The service is encrypted using Transport Level Security (TLS).

5.1.2 Authentication

The OData specification requires Basic Authentication for the best interoperability, and the Data Connect follows the OData specification and implements that requirement. Please get in touch with an administrator for the Qmatic system deployed for your company to get the credentials.

5.1.3 Performance considerations

Data Connect allows you to read all data at once. It is, however, recommended to only read the changes since the last query. Doing this is especially important for the fact tables that contain large amounts of data. For more slowly changing entities, Qmatic recommends you to implement caching.

5.1.4 Change data format

The OData protocol supports both XML and JSON. You can control the format by changing the accept HTTP header.

5.1.5 Limitations

- **Read only:** The data in Data Connect is read-only. You can not perform any of the OData operations that modify data. An error will be returned when a request is made that attempts to write or modify data.
- **Rate limiting:** Data Connect has a rate limit, set by the Qmatic system administrator. If you exceed the rate limit, you will get an HTTP 429 error. If you get this error, you need to slow down your fetching process.

Due to the ETL process in aggregating the data, this is not a real-time API. Any system that integrates with this API should fetch data not faster than every 30 minutes. Choose a random starting time for the fetching process.

Ex: 13:47 is better than 13:00. This reduces any risk of disturbing your other processes.

5.2 Getting started

1. Read the OData beginner tutorial, see Resources below.
2. Look around with the service endpoint in your browser, Postman, or similar tools to get a feel for the look and feel of the protocol. You can obtain the service endpoint URL from your Qmatic system administrator.
3. Choose a library for the language and platform you use. Qmatic recommends using a client library for the best maintainability and compatibility, but you can also interact with Data Connect without a client library.

5.2.1 Resources

- OData Beginner Tutorial: <https://www.odata.org/getting-started/basic-tutorial/>
- Client Libraries: <https://www.odata.org/libraries/>

5.2.2 Exploring Data Connect

Normally, you would use a library or a BI tool/system to interact with the API, but for exploration purposes you can use one of the tools listed below.

Postman

To work with the API using Postman, you first need to install Postman. Postman is a web application that allows you to interact with web APIs.

- Postman OData Tutorial: <https://www.postman.com/odatatestmock/workspace/my-workspace/documentation/16737030-698fdeae-4db6-4d4c-8f4f-f322d109a0f1>

curl

To use curl to interact with Data Connect, you first need to install curl. curl is a command line tool that allows you to interact with web APIs. You can follow the instructions in the tutorial mentioned for Postman in the paragraph above.

- curl: <https://curl.se/>

5.3 Exploring OData

OData RESTful APIs are easy to consume. The OData metadata, a machine readable description of the data model of the APIs, enables the creation of powerful generic client proxies and tools. Some of them can help you interact with OData even without knowing anything about the protocol.

If you want to have a quick look at the protocol, here are some examples of how to use the OData protocol using simple http requests.

The following tools are good to explore the OData protocol:

- Postman
- curl
- REST Client for Visual Studio Code
- Any web browser

❗ The URLs below are not properly URL encoded for readability purposes. This needs to be done manually or through a tool like Postman or similar.

Table 1: Example requests

Request	URL
Get all branches	GET https://mydomain.com/odata/Branches HTTP/1.1
Get two branches	GET https://mydomain.com/odata/Branches?\$top=2 HTTP/1.1
Get a specific branch	GET https://mydomain.com/odata/Branches?\$filter=Branch eq 'Lone Tree' HTTP/1.1
Get a visit transaction with waiting time over 10 minutes in a specific branch	GET https://mydomain.com/odata/VisitTransactions?\$orderby=EnteredQueueTimeKey asc&\$top=10 HTTP/1.1
Get Services ordered by name	GET https://mydomain.com/odata/Services?\$orderby=Service asc HTTP/1.1
Get the 10 last visit transactions	GET https://mydomain.com/odata/VisitTransactions?\$orderby=EnteredQueueTimeKey asc&\$top=10 HTTP/1.1

5.4 OData Primitive Data Types

Structured types are composed of other structured types and primitive types. The primitive types that OData defines can be seen in the table below.

Table 2: Primitive types

Type	Meaning
Edm.Binary	Binary data
Edm.Boolean	Binary-valued logic
Edm.Byte	Unsigned 8-bit integer
Edm.Date	Date without a time-zone offset
Edm.DateTimeOffset	Date and time with a time-zone offset, no leap seconds
Edm.Decimal	Numeric values with decimal representation
Edm.Double	IEEE 754 binary64 floating-point number (15-17 decimal digits)
Edm.Duration	Signed duration in days, hours, minutes, and (sub)seconds
Edm.Guid	16-byte (128-bit) unique identifier
Edm.Int16	Signed 16-bit integer
Edm.Int32	Signed 32-bit integer
Edm.Int64	Signed 64-bit integer
Edm.SByte	Signed 8-bit integer
Edm.Single	IEEE 754 binary32 floating-point number (6-9 decimal digits)
Edm.Stream	Binary data stream
Edm.String	Sequence of characters
Edm.TimeOfDay	Clock time 00:00-23:59:59.999999999999

6. API Reference

Topics

Appointment.....	29
AppointmentArrivalStatus	34
AppointmentCancelledDate	35
AppointmentCreatedDate	37
AppointmentCreatedTime	40
AppointmentCustomer	42
AppointmentEndTime	43
AppointmentService.....	45
AppointmentStartTime	46
AppointmentStatus	48
Branch	49
CallTime.....	51
Date	53
DeliveredService.....	56
DeliveredServiceTime.....	60
EndTime	62
EnteredQueueTime	64
EntryPoint	66
FeedbackAnswer.....	67
FeedbackDate	68
FeedbackQuestion.....	71
FeedbackResult.....	72
FeedbackTime	75
Outcome	77
QmaticBiContainer.....	79
QuestionType	83
Queue	84
QueueType	85
Resource	86
Service.....	87
ServicePoint.....	88
Staff	89
StaffSession.....	90
StaffSessionEndTime	93

StaffSessionOutcome	95
StaffSessionStartTime	96
Visit	98
VisitCallType.....	100
VisitCustomer	101
VisitTransaction	102
VisitTransactionOutcome.....	109
WorkProfile	111

6.1 Appointment

For every Appointment that is created, there is an entity here. When the appointment is updated, the entity is updated with the new information. An Appointment that was served in a Visit is included in the stat message for the Visit and is inserted after the VisitTransaction table has been updated.

6.1.1 AppointmentCustomers

Type: AppointmentCustomer

6.1.2 AppointmentDuration

Number of minutes that the Appointment was planned to take.

Type: Edm.Int64

6.1.3 AppointmentService

Type: AppointmentService

6.1.4 AppointmentStatusKey

Note: The Appointment status is never updated from CREATED if the Visit is never put in an Appointment Queue.

Type: Edm.Int32

6.1.5 ArrivalStatus

Type: AppointmentArrivalStatus

6.1.6 ArrivalStatusKey

Reference to the id of the Arrival Status in the table dim_arrival_status.

Type: Edm.Int32

6.1.7 Branch

Type: Branch

6.1.8 BranchKey

Reference to the id of a Branch in the table dim_branch.

Indicates that this entry is a Branch, not a Branch Group. If this value is NULL, the entry is a Branch Group.

Type: Edm.Int64

6.1.9 CancelledDate

Type: AppointmentCancelledDate

6.1.10 CancelledDateKey

The date when the Appointment was canceled.

Type: Edm.Int32

6.1.11 ClientKey

Reference to the id of an entry in the table dim_client which holds the license.

Type: Edm.Int64

6.1.12 CountServices

The number of Services that were delivered.

Type: Edm.Int32

6.1.13 CreatedAt

The year, quarter, month and date that the appointment was created.

Type: Edm.DateTimeOffset

6.1.14 CreatedDate

Type: AppointmentCreatedDate

6.1.15 CreatedDateKey

The date when the Appointment was created (booked).

Type: Edm.Int32

6.1.16 CreatedTime

Type: AppointmentCreatedTime

6.1.17 CreatedTimeKey

Timestamp for when the Appointment was created (booked).

Type: Edm.Int32

6.1.18 Date

Type: Date

6.1.19 DateKey

Date for the Appointment.

Type: Edm.Int32

6.1.20 DaysCancelledBefore

Number of days before the Appointment start time that it was cancelled.

Type: Edm.Int32

6.1.21 DaysToAppointment

The number of days between when the Appointment was booked and the Appointment start time.

Type: Edm.Int64

6.1.22 EndTime

Type: AppointmentEndTime

6.1.23 EndTimeKey

End time for the Appointment.

Type: Edm.Int32

6.1.24 Id

The appointment identifier.

Type: Edm.Int32

6.1.25 Lateness

The time from when the Appointment should have started until the Customer was called.

Type: Edm.Int64

6.1.26 OriginId

The unique id for the Appointment, used to keep track of changes. This value is never changed.

Type: Edm.Int64

6.1.27 Resource

Type: Resource

6.1.28 ResourceKey

The Id of the Resource that was assigned to the Appointment.

Type: Edm.Int64

6.1.29 ServiceKey

ServiceKey remains for backwards compatibility and is populated with the id for the first service if it is a multi service Appointment.

Type: Edm.Int64

6.1.30 StartTime

Type: AppointmentStartTime

6.1.31 StartTimeKey

The start time for the Appointment.

Type: Edm.Int32

6.1.32 Status

Type: AppointmentStatus

6.1.33 Tardiness

The time from when the Customer arrived, until the Appointment should have started. This is always a positive value.

Note: Only used for backward compatibility. Lateness should be used instead!

Type: Edm.Int64

6.1.34 TransactionTime

The sum of all transaction times within the Appointment (with the same Appointment.Id)

Type: Edm.Int64

6.1.35 UpdatedCount

The number of times this Appointment has been changed (for example rescheduled).

Type: Edm.Int32

6.1.36 WaitingTime

The waiting time for the Appointment. The waiting time is exactly the same as for the Visit transaction (an Appointment becomes a Visit when the Customer arrives).

Type: Edm.Int64

6.1.37 WalkingTime

The walking time for the first transaction in the Appointment. We assume that all other transactions were delivered within the same Appointment, without any walking times.

Type: Edm.Int64

6.1.38 WrapupTime

The total wrap-up time for all transactions within the Appointment.

Type: Edm.Int64

6.2 AppointmentArrivalStatus

This table holds information about arrival status for Appointments.

The following values are defined:

Id	Name	Description
1	NOT_ARRIVED	The Customer has not yet arrived (no Visit has been created). Note: This is not updated when the Customer arrives, but when the transaction is ready.
2	ARRIVE_BEFORE_CALL_BEFORE	The Customer arrived before the Appointment time and was also called before the Appointment time.
3	ARRIVE_BEFORE_CALL_AFTER	The Customer arrived before the Appointment time and was called after the Appointment time.
4	ARRIVE_BEFORE_NOT_CALLED	Not used.
5	ARRIVE_LATE	The Customer arrived after the Appointment time.

6.2.1 Appointments

Type: Appointment

6.2.2 ArrivalStatus

The name of the Arrival Status, see table above.

Type: Edm.String

6.2.3 Id

The Arrival Status identifier.

Type: Edm.Int32

6.3 AppointmentCancelledDate

This entity holds information about the date that the appointment was canceled.

6.3.1 Appointments

Type: Appointment

6.3.2 DayOfMonth

The day of the month.

Example: For 2014-01-13, the day of month is 13.

Type: Edm.Int32

6.3.3 DayOfWeek

The day of the week, where Sunday is the first day of the week.

Example: 1=Sunday, 2=Monday, etc.

Type: Edm.Int32

6.3.4 DayOfWeekName

The name of the day of week:

Example: day.Monday

Type: Edm.String

6.3.5 DayOfWeekNameEn

The English name for the day of the week.

Example: Tuesday

Type: Edm.String

6.3.6 FullDate

The full Date: Year, Quarter, Month and Day.

Type: Edm.Date

6.3.7 Id

The Appointment Cancelled Date identifier.

Type: Edm.Int32

6.3.8 Month

The month. Numbers 1 to and including 12.

Example: 5=May

Type: Edm.Int32

6.3.9 MonthName

The name of the month.

Example: month.march

Type: Edm.String

6.3.10 MonthNameEn

The name of the month in English.

Example: January

Type: Edm.String

6.3.11 Quarter

The quarter of the year, presented as numbers 1 to and including 4.

Type: Edm.Int32

6.3.12 WeekNumber

The week number the week containing January 1 as the first week of the year. Presented as numbers ranging from 1 to 54.

The first day of the week is Sunday.

Type: Edm.Int32

6.3.13 WeekNumberIso

The week number according to ISO 8601, where the week containing the first Thursday of the year is defined as the first week of the year. Presented as numbers ranging from 1 to 53.

The first day of the week is Monday.

Type: Edm.Int32

6.3.14 Year

The year.

Example: 2014

Type: Edm.Int32

6.4 AppointmentCreatedDate

This entity holds information about the date that the appointment was created.

6.4.1 Appointments

Type: Appointment

6.4.2 DayOfMonth

The day of the month.

Example: For 2014-01-13, the day of month is 13.

Type: Edm.Int32

6.4.3 DayOfWeek

The day of the week, where Sunday is the first day of the week.

Example: 1=Sunday, 2=Monday, etc.

Type: Edm.Int32

6.4.4 DayOfWeekName

The name of the day of week:

Example: day.Monday

Type: Edm.String

6.4.5 DayOfWeekNameEn

The English name for the day of the week.

Example: Tuesday

Type: Edm.String

6.4.6 FullDate

The full Date: Year, Quarter, Month and Day.

Type: Edm.Date

6.4.7 Id

The Appointment Cancelled Date identifier.

Type: Edm.Int32

6.4.8 Month

The month. Numbers 1 to and including 12.

Example: 5=May

Type: Edm.Int32

6.4.9 MonthName

The name of the month.

Example: month.march

Type: Edm.String

6.4.10 MonthNameEn

The name of the month in English.

Example: January

Type: Edm.String

6.4.11 Quarter

The quarter of the year, presented as numbers 1 to and including 4.

Type: Edm.Int32

6.4.12 WeekNumber

The week number the week containing January 1 as the first week of the year.
Presented as numbers ranging from 1 to 54.

The first day of the week is Sunday.

Type: Edm.Int32

6.4.13 WeekNumberIso

The week number according to ISO 8601, where the week containing the first Thursday of the year is defined as the first week of the year. Presented as numbers ranging from 1 to 53.

The first day of the week is Monday.

Type: Edm.Int32

6.4.14 Year

The year.

Example: 2014

Type: Edm.Int32

6.5 AppointmentCreatedTime

This entity holds information about the time that the Appointment was created.

6.5.1 AmPm

The string AM or PM.

Type: Edm.String

6.5.2 Appointments

Type: Appointment

6.5.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.5.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.5.5 Id

The Appointment Created Time identifier.

Type: Edm.Int32

6.5.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.5.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.5.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.5.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.5.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.5.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.5.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.5.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.5.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.5.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.6 AppointmentCustomer

This entity holds information about the Customer of an Appointment.

6.6.1 Appointment

Type: Appointment

6.6.2 AppointmentId

Identifying the Appointment.

Type: Edm.Int64

6.6.3 Booked

Booked = 1 if the Customer was added before the Appointment become a Visit.

Type: Edm.Boolean

6.6.4 CustomerId

Identifying the Customer. No Customer data is stored in the stat table, so there is no connection to any stat table with Customer info. This is a unique id for every Customer that has been identified and where Customer data has been found in a Customer database.

Type: Edm.Int64

6.6.5 Id

The Appointment Customer table identifier.

Type: Edm.Int64

6.6.6 Served

Served = 1 if the Customer was included when the Service was delivered (if the Service was an Appointment transaction). Customers with Booked = 0 and Served = 1 were added to the Appointment after it became an Appointment transaction.

Type: Edm.Boolean

6.7 AppointmentEndTime

This entity holds information about the time that the Appointment ended.

6.7.1 AmPm

The string AM or PM.

Type: Edm.String

6.7.2 Appointments

Type: Appointment

6.7.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.7.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.7.5 Id

The End Time identifier.

Type: Edm.Int32

6.7.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.7.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.7.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.7.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.7.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.7.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.7.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.7.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.7.14 Minute

When the time is 13:37 the minute is 13:37.

Type: Edm.Int32

6.7.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.8 AppointmentService

This entity holds information about the service delivered during the Appointment.

6.8.1 Appointment

Type: Appointment

6.8.2 AppointmentId

The Appointment id identifies the Appointment.

Type: Edm.Int64

6.8.3 Id

The Appointment Service table identifier.

Type: Edm.Int64

6.8.4 Served

Indicates if the Services have been delivered.

Type: Edm.Boolean

6.8.5 Service

Type: Service

6.8.6 ServiceKey

Reference to the id of a Service in the table dim_service

Type: Edm.Int64

6.9 AppointmentStartTime

The string AM or PM.

6.9.1 AmPm

Type: Edm.String

6.9.2 Appointments

Type: Appointment

6.9.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.9.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.9.5 Id

The Start Time identifier.

Type: Edm.Int32

6.9.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.9.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.9.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.9.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.9.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.9.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.9.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.9.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.9.14 Minute

When the time is 13:37 the minute is 13:37.

Type: Edm.Int32

6.9.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.10 AppointmentStatus

The name of the Appointment Status, see table above.

6.10.1 Appointments

Type: Appointment

6.10.2 Id

The Appointment Status identifier.

Type: Edm.Int32

6.10.3 Status

Type: Edm.String

6.11 Branch

This entity holds the Branch information that will be used by all other entities. The Branch entity is updated when a Branch is saved or updated on the central server.

6.11.1 Branch

The name of the Branch.

Type: Edm.String

6.11.2 City

The name of the City where the Branch is located.

Type: Edm.String

6.11.3 Country

The name of the Country where the Branch is located.

Type: Edm.String

6.11.4 DeliveredServices

Type: DeliveredService

6.11.5 FactAppointments

Type: Appointment

6.11.6 FactStaffSessions

Type: StaffSession

6.11.7 FactVisitTransactions

Type: VisitTransaction

6.11.8 FeedbackResults

Type: FeedbackResult

6.11.9 Id

The Branch identifier.

Type: Edm.Int32

6.11.10 Latitude

The latitude of the Branch.

Type: Edm.Double

6.11.11 Longitude

The longitude of the Branch.

Type: Edm.Double

6.11.12 TimeZone

The time zone of the Branch.

Type: Edm.String

6.12 CallTime

This entity holds information about the time that the Customer was called.

6.12.1 AmPm

The string AM or PM.

Type: Edm.String

6.12.2 FactVisitTransactions

Type: VisitTransaction

6.12.3 Hour

When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.12.4 Hour24

When the time is 13:37 the hour is 24.

Type: Edm.Int32

6.12.5 Id

The Call Time identifier.

Type: Edm.Int32

6.12.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.12.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.12.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.12.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.12.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.12.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.12.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.12.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.12.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.12.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.13 Date

This entity holds date information.

6.13.1 Appointments

Type: Appointment

6.13.2 DayOfMonth

The day of the month.

Example: For 2014-01-13, the day of month is 13.

Type: Edm.Int32

6.13.3 DayOfWeek9

The day of the week, where Sunday is the first day of the week.

Example: 1=Sunday, 2=Monday, etc.

Type: Edm.Int32

6.13.4 DayOfWeekName

The name of the day of week:

Example: day.Monday

Type: Edm.String

6.13.5 DayOfWeekNameEn

The English name for the day of the week.

Example: Tuesday

Type: Edm.String

6.13.6 DeliveredServices

Type: DeliveredService

6.13.7 FactVisitTransactions

Type: VisitTransaction

6.13.8 FullDate

The full Date: Year, Quarter, Month and Day.

Type: Edm.Date

6.13.9 Id

The Date identifier.

Type: Edm.Int32

6.13.10 Month

The month. Numbers 1 to and including 12.

Example: 5=May

Type: Edm.Int32

6.13.11 MonthName

The name of the month.

Example: month.march

Type: Edm.String

6.13.12 MonthNameEn

The name of the month in English.

Example: January

Type: Edm.String

6.13.13 Quarter

The quarter of the year, presented as numbers 1 to and including 4.

Type: Edm.Int32

6.13.14 StaffSessions

Type: StaffSession

6.13.15 WeekNumber

The week number the week containing January 1 as the first week of the year.
Presented as numbers ranging from 1 to 54.

The first day of the week is Sunday.

Type: Edm.Int32

6.13.16 WeekNumberIso

The week number according to ISO 8601, where the week containing the first Thursday of the year is defined as the first week of the year. Presented as numbers ranging from 1 to 53.

The first day of the week is Monday.

Type: Edm.Int32

6.13.17 Year

The year.

Example: 2014

Type: Edm.Int32

6.14 DeliveredService

This entity holds information about delivered services added to Visits.

6.14.1 Branch

Type: Branch

6.14.2 BranchKey

Reference to the Branch identifier in the Branch dimension table.

Indicates that this entry is a Branch, not a Branch Group. If this value is NULL, the entry is a Branch Group.

Type: Edm.Int64

6.14.3 CreatedAt

The year, quarter, month and date that the delivered service was created.

Type: Edm.DateTimeOffset

6.14.4 Date

Type: Date

6.14.5 DateKey

Date for when the delivered service was added.

Type: Edm.Int32

6.14.6 DeliveredServiceKey

Reference to the Delivered Service Identifier in the Delivered Service dimension table.

Type: Edm.Int64

6.14.7 Id

The Delivered Service identifier.

Type: Edm.Int32

6.14.8 Outcome

Type: Outcome

6.14.9 OutcomeKey

Reference to the id of an Outcome in the Outcome dimension table.

Type: Edm.Int64

6.14.10 Queue

Type: Queue

6.14.11 QueueKey

Reference to the id of an Queue in the Queue dimension table.

Type: Edm.Int64

6.14.12 QueueType

Type: QueueType

6.14.13 QueueTypeKey

Reference to the id of an Queue **Type** in the Queue **Type** dimension table.

Type: Edm.Int32

6.14.14 Service

Type: Service

6.14.15 ServiceKey

Reference to the id of a Service in the Service dimension table.

Type: Edm.Int64

6.14.16 ServicePoint

Type: ServicePoint

6.14.17 ServicePointKey

Reference to the id of an Service Point in the Service Point dimension table.

Type: Edm.Int64

6.14.18 Staff

Type: Staff

6.14.19 StaffKey

Reference to the id of a Staff in the Staff dimension table.

Type: Edm.Int32

6.14.20 Time

Type: DeliveredServiceTime

6.14.21 TimeKey

Reference to the id of a Time in the Time dimension table.

Type: Edm.Int32

6.14.22 TimeSeconds

The seconds part of the time stamp.

Type: Edm.Int32

6.14.23 TransactionTime

The transaction time is from the Customer was called and until the Delivered Service was added, or between two added Delivered Services. The Delivered Service is added after it has been delivered.

Type: Edm.Int64

6.14.24 Visit

Type: Visit

6.14.25 VisitKey

Reference to the id of a Visit in the Visit dimension table.

Type: Edm.Int64

6.14.26 WaitingTime

There is no unique waiting time for a delivered service. It is the same as the for the transaction (FactVisitTransaction).

Type: Edm.Int64

6.15 DeliveredServiceTime

This entity holds information about the time that a service was delivered.

6.15.1 AmPm

The string AM or PM.

Type: Edm.String

6.15.2 DeliveredServices

Type: DeliveredService

6.15.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.15.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.15.5 Id

The Appointment Created Time identifier.

Type: Edm.Int32

6.15.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.15.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.15.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.15.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.15.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.15.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.15.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.15.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.15.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.15.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.16 EndTime

This table holds end time information.

6.16.1 AmPm

The string AM or PM.

Type: Edm.String

6.16.2 FactVisitTransactions

Type: VisitTransaction

6.16.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.16.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.16.5 Id

The End Time identifier.

Type: Edm.Int32

6.16.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.16.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.16.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.16.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.16.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.16.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.16.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.16.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.16.14 Minute

When the time is 13:37 the minute is 13:37.

Type: Edm.Int32

6.16.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.17 EnteredQueueTime

This entity holds information about the time that a Customer was placed in a Queue.

6.17.1 AmPm

The string AM or PM.

Type: Edm.String

6.17.2 FactVisitTransactions

Type: VisitTransaction

6.17.3 Hour

When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.17.4 Hour24

When the time is 13:37 the hour is 24.

Type: Edm.Int32

6.17.5 Id

The Entered Queue Time identifier.

Type: Edm.Int32

6.17.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.17.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.17.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.17.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.17.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.17.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.17.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.17.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.17.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.17.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.18 EntryPoint

This entity holds information about Entry Points that are configured in the system. Every time a new or updated Entry Point is encountered in an inserted Visit, a new entry is generated.

6.18.1 EntryPoint

The name of the Entry Point.

Type: Edm.String

6.18.2 FactVisitTransactions

Type: VisitTransaction

6.18.3 Id

The Entry Point identifier.

Type: Edm.Int32

6.19 FeedbackAnswer

This entity holds information about feedback answers.

6.19.1 FeedbackResults

Type: FeedbackResult

6.19.2 Id

The Feedback Answer identifier.

Type: Edm.Int32

6.19.3 NumericAnswer

The numeric answer from Customer Feedback.

Type: Edm.Int32

6.19.4 TextAnswer

The answer for the free text answers. This is NULL if it is a numeric answer.

Text answers cannot be used as dimensions. This option is also used for numeric answers that are not suitable to be used as dimensions, for example NPS.

Type: Edm.String

6.20 FeedbackDate

This entity holds information about the date that the feedback was submitted.

6.20.1 DayOfMonth

The day of the month.

Example: For 2014-01-13, the day of month is 13.

Type: Edm.Int32

6.20.2 DayOfWeek

The day of the week, where Sunday is the first day of the week.

Example: 1=Sunday, 2=Monday, etc.

Type: Edm.Int32

6.20.3 DayOfWeekName

The name of the day of week:

Example: day.Monday

Type: Edm.String

6.20.4 DayOfWeekNameEn

The English name for the day of the week.

Example: Tuesday

Type: Edm.String

6.20.5 FeedbackResults

Type: FeedbackResult

6.20.6 FullDate

The full Date: Year, Quarter, Month and Day.

Type: Edm.Date

6.20.7 Id

The Feedback Date identifier.

Type: Edm.Int32

6.20.8 Month

The month. Numbers 1 to and including 12.

Example: 5=May

Type: Edm.Int32

6.20.9 MonthName

The name of the month.

Example: month.march

Type: Edm.String

6.20.10 MonthNameEn

The name of the month in English.

Example: January

Type: Edm.String

6.20.11 Quarter

The quarter of the year, presented as numbers 1 to and including 4.

Type: Edm.Int32

6.20.12 WeekNumber

The week number the week containing January 1 as the first week of the year. Presented as numbers ranging from 1 to 54.

The first day of the week is Sunday.

Type: Edm.Int32

6.20.13 WeekNumberIso

The week number according to ISO 8601, where the week containing the first Thursday of the year is defined as the first week of the year. Presented as numbers ranging from 1 to 53.

The first day of the week is Monday.

Type: Edm.Int32

6.20.14 Year

The year.

Example: 2014

Type: Edm.Int32

6.21 FeedbackQuestion

This entity holds information about feedback questions.

6.21.1 FeedbackResults

Type: FeedbackResult

6.21.2 Id

The Feedback Question identifier.

Type: Edm.Int64

6.21.3 Question

The Feedback Question.

Type: Edm.String

6.21.4 StartDate

The date that the Feedback Question was created.

Type: Edm.Date

6.22 FeedbackResult

This entity holds information about the Customer Feedback results.

6.22.1 Answer

Type: FeedbackAnswer

6.22.2 AnswerKey

Reference to the id of an Answer in the Answer dimension table.

Type: Edm.Int32

6.22.3 Branch

Type: Branch

6.22.4 BranchKey

Reference to the id of a Branch in the Branch dimension table.

Type: Edm.Int32

6.22.5 Date

Type: FeedbackDate

6.22.6 DateKey

Reference to the id of a Date in the Date dimension table.

Type: Edm.Int32

6.22.7 EventTimestamp

The timestamp for when the Feedback Result was created.

Type: Edm.DateTimeOffset

6.22.8 Id

The feedback result identifier.

Type: Edm.Int64

6.22.9 Question

Type: FeedbackQuestion

6.22.10 QuestionKey

Reference to the id of a Question in the Question dimension table.

Type: Edm.Int32

6.22.11 QuestionType

The **type** of Feedback Question.

Type: QuestionType

6.22.12 QuestionTypeKey

Reference to the id of a Question **Type** in the Question **Type** dimension table.

Type: Edm.Int32

6.22.13 TextAnswer

The answer for the free text answers. This is NULL if it is a numeric answer.

Text answers cannot be used as dimensions. This option is also used for numeric answers that are not suitable to be used as dimensions, for example NPS.

Type: Edm.String

6.22.14 Time

Type: FeedbackTime

6.22.15 TimeKey

Reference to the id of a Time in the Time dimension table.

Type: Edm.Int32

6.22.16 TimeSeconds

The second that the Feedback Result was added to the Visit.

Type: Edm.Int32

6.22.17 Visit

Type: Visit

6.22.18 VisitKey

Reference to the id of a Visit in the Visit dimension table.

Type: Edm.Int64

6.23 FeedbackTime

This entity holds information about the time that the Feedback was submitted.

6.23.1 AmPm

The string AM or PM.

Type: Edm.String

6.23.2 FeedbackResults

Type: FeedbackResult

6.23.3 Hour

When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.23.4 Hour24

When the time is 13:37 the hour is 24.

Type: Edm.Int32

6.23.5 Id

The Feedback Time identifier.

Type: Edm.Int32

6.23.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.23.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.23.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.23.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.23.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.23.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.23.12 Interval5Min

Type: Edm.String

6.23.13 Interval5Min24

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.23.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.23.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.24 Outcome

This entity holds information about Outcomes that are configured in the Business Configuration application of the system. Every time a new or updated Outcome is encountered in an inserted Visit, a new entry is generated.

6.24.1 BranchOriginId

The origin id for the Branch.

Type: Edm.Int64

6.24.2 ClientKey

The id of an entry in the table dim_client which holds the license.

Type: Edm.Int64

6.24.3 DeliveredServices

Type: DeliveredService

6.24.4 EndDate

Time stamp (Year, quarter, month and date) for when this entity was deleted or updated in Orchestra. If this value is NULL, it means that it is the active instance of this entity.

Type: Edm.DateTimeOffset

6.24.5 FactVisitTransactions

Type: VisitTransaction

6.24.6 Id

The Outcome identifier.

Type: Edm.Int32

6.24.7 OriginId

The unique identifier of the Outcome, used to keep track of changes. This value is never changed.

Type: Edm.Int32

6.24.8 Outcome

The name of the Outcome.

Type: Edm.String

6.24.9 StartDate

The date that the Outcome was created.

Type: Edm.DateTimeOffset

6.25 QmaticBiContainer

6.25.1 AppointmentArrivalStatuses

Type: AppointmentArrivalStatus

6.25.2 AppointmentCancelledDates

Type: AppointmentCancelledDate

6.25.3 AppointmentCreatedDates

Type: AppointmentCreatedDate

6.25.4 AppointmentCreatedTimes

Type: AppointmentCreatedTime

6.25.5 AppointmentCustomers

Type: AppointmentCustomer

6.25.6 AppointmentEndTimes

Type: AppointmentEndTime

6.25.7 AppointmentServices

Type: AppointmentService

6.25.8 AppointmentStartTimes

Type: AppointmentStartTime

6.25.9 AppointmentStatuses

Type: AppointmentStatus

6.25.10 Appointments

Type: Appointment

6.25.11 Branches

Type: Branch

6.25.12 CallTimes

Type: CallTime

6.25.13 Dates

Type: Date

6.25.14 DeliveredServiceTimes

Type: DeliveredServiceTime

6.25.15 DeliveredServices

Type: DeliveredService

6.25.16 EndTimes

Type: EndTime

6.25.17 EnteredQueueTimes

Type: EnteredQueueTime

6.25.18 EntryPoints

Type: EntryPoint

6.25.19 FeedbackAnswers

Type: FeedbackAnswer

6.25.20 FeedbackDates

Type: FeedbackDate

6.25.21 FeedbackQuestions

Type: FeedbackQuestion

6.25.22 FeedbackResults

Type: FeedbackResult

6.25.23 FeedbackTimes

Type: FeedbackTime

6.25.24 Outcomes

Type: Outcome

6.25.25 QuestionTypes

Type: QuestionType

6.25.26 QueueTypes

Type: QueueType

6.25.27 Queues

Type: Queue

6.25.28 Resources

Type: Resource

6.25.29 ServicePoints

Type: ServicePoint

6.25.30 Services

Type: Service

6.25.31 StaffSessionEndTimes

Type: StaffSessionEndTime

6.25.32 StaffSessionOutcomes

Type: StaffSessionOutcome

6.25.33 StaffSessionStartTimes

Type: StaffSessionStartTime

6.25.34 StaffSessions

Type: StaffSession

6.25.35 Staffs

Type: Staff

6.25.36 VisitCallTypes

Type: VisitCallType

6.25.37 VisitCustomers

Type: VisitCustomer

6.25.38 VisitTransactionOutcomes

Type: VisitTransactionOutcome

6.25.39 VisitTransactions

Type: VisitTransaction

6.25.40 Visits

Type: Visit

6.25.41 WorkProfiles

Type: WorkProfile

6.26 QuestionType

This table holds information about the **type** of feedback question.

6.26.1 FeedbackResults

Type: FeedbackResult

6.26.2 Id

The Question **Type** identifier.

Type: Edm.Int32

6.26.3 QuestionType

The name of the feedback Question **Type**.

Type: Edm.String

6.27 Queue

This entity holds information about Queues that are configured in the Business Configuration application of the system. Every time a new or updated Queue is encountered for a specific Branch in an inserted Visit, a new entry is generated.

6.27.1 DeliveredServices

Type: DeliveredService

6.27.2 FactVisitTransactions

Type: VisitTransaction

6.27.3 Id

The Queue identifier.

Type: Edm.Int32

6.27.4 Queue

The name of the Queue.

Type: Edm.String

6.28 QueueType

This entity holds information about which Queue **type** is used for a Queue.

The following values are defined:

Id	Name
1	QUEUE
2	STAFF_QUEUE
3	SERVICE_POINT
4	STAFF_POOL
5	APPOINTMENT_QUEUE

6.28.1 FactVisitTransactions

Type: VisitTransaction

6.28.2 Id

The Queue **type** identifier.

Type: Edm.Int32

6.28.3 QueueType

The name of the Queue **type**.

Type: Edm.String

6.29 Resource

This entity holds information about Resources that are configured in the Calendar Admin application. Every time a new or updated Resource is encountered in an inserted Appointment, a new entry is generated in this table. When a resource is updated, an end date will be set to the previous version.

6.29.1 Appointments

Type: Appointment

6.29.2 BranchOriginId

The unique identifier of the Branch, used to keep track of changes. This value is never changed.

Type: Edm.Int64

6.29.3 Id

The Resource identifier.

Type: Edm.Int32

6.29.4 Name

The name of the Resource.

Type: Edm.String

6.29.5 OriginId

The unique identifier of the Resource, used to keep track of changes. This value is never changed.

Type: Edm.Int32

6.30 Service

This entity holds information about Services that are configured in the Business Configuration application of the system. Every time a new or updated Service is encountered in an inserted Visit, or Appointment, a new entry is generated.

6.30.1 AppointmentService

Type: AppointmentService

6.30.2 DeliveredServices

Type: DeliveredService

6.30.3 FactVisitTransactions

Type: VisitTransaction

6.30.4 Id

The Service identifier.

Type: Edm.Int32

6.30.5 Service

The name of the Service.

Type: Edm.String

6.31 ServicePoint

This entity holds information about Service Points that are configured in the system. Every time a new or updated Service Point is encountered for a Branch in an inserted Visit, a new entry is generated.

6.31.1 DeliveredServices

Type: DeliveredService

6.31.2 FactVisitTransactions

Type: VisitTransaction

6.31.3 Id

The Service Point identifier.

Type: Edm.Int32

6.31.4 ServicePoint

The name of the Service Point.

Type: Edm.String

6.32 Staff

This entity holds information about staff (Users) that are configured in the User Management application of the system. Every time a new or updated staff is encountered in an inserted Visit or Staff session, a new entry is generated.

The system can also contain a staff member with the name ?anonymous?. This identifies events that are missing a logged in user, for example a branch reset.

6.32.1 DeliveredServices

Type: DeliveredService

6.32.2 FactStaffSessions

Type: StaffSession

6.32.3 FactVisitTransactions

Type: VisitTransaction

6.32.4 FirstName

The first name of the staff member.

Type: Edm.String

6.32.5 Id

The Staff identifier.

Type: Edm.Int32

6.32.6 LastName

The last name of the Staff member.

Type: Edm.String

6.32.7 Staff

The full name of the Staff member.

Type: Edm.String

6.33 StaffSession

Contains the sessions of the users (Staff).

6.33.1 Branch

Type: Branch

6.33.2 BranchKey

Reference to the id of a Branch in the Branch Dimension table.

Indicates that this entry is a Branch, not a Branch Group. If this value is NULL, the entry is a Branch Group.

Type: Edm.Int64

6.33.3 CreatedAt

The year, quarter, month and date that the Staff Session was created.

Type: Edm.DateTimeOffset

6.33.4 Date

Type: Date

6.33.5 DateKey

Date for when the delivered service was added.

Type: Edm.Int32

6.33.6 EndTime

Type: StaffSessionEndTime

6.33.7 EndTimeKey

The end time for the Staff Session.

Type: Edm.Int32

6.33.8 EndTimestamp

The time stamp for the end of the Staff Session.

Type: Edm.DateTimeOffset

6.33.9 Id

The Staff Session identifier.

Type: Edm.Int32

6.33.10 IdleTime

The part of the session time not spent on any Customer.

Type: Edm.Int64

6.33.11 ServedCount

The number of Customers that were served during the session.

Type: Edm.Int32

6.33.12 ServingTime

The part of the session time that was transaction time.

Type: Edm.Int64

6.33.13 SessionOutcomeKey

Reference to the id of a Staff Session Outcome in the Staff Session Outcome dimension table.

Type: Edm.Int32

6.33.14 Staff

Type: Staff

6.33.15 StaffKey

Reference to the id of a Staff in the Staff dimension table.

Type: Edm.Int64

6.33.16 StaffSessionOutcome

Type: StaffSessionOutcome

6.33.17 StartTime

Type: StaffSessionStartTime

6.33.18 StartTimestamp

The timestamp for the start of the Staff Session.

Type: Edm.DateTimeOffset

6.33.19 StoreNextTime

The time, in seconds, that the Staff member spent in store next during the session.

Type: Edm.Int32

6.33.20 TimeKey

Reference to the id of a Time in the Time dimension table.

Type: Edm.Int32

6.33.21 TimeSeconds

The time, in seconds, that the staff member spent in store next during the session.

Type: Edm.Int32

6.33.22 TotalTime

The session time.

Type: Edm.Int64

6.33.23 VisitsServed

The number of transactions during the session, no matter of the outcome - for example, no shows are included.

Type: Edm.Int64

6.33.24 WrapupTime

The total wrap-up time for all transactions within the Staff Session.

Type: Edm.Int64

6.34 StaffSessionEndTime

This entity holds information about the end time of Staff Sessions.

6.34.1 AmPm

The string AM or PM.

Type: Edm.String

6.34.2 FactStaffSessions

Type: StaffSession

6.34.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.34.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.34.5 Id

The Staff Session end time identifier.

Type: Edm.Int32

6.34.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.34.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.34.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.34.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.34.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.34.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.34.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.34.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.34.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.34.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.35 StaffSessionOutcome

This entity holds information about the outcome of the Staff session.

The following values are defined:

Id	Name	Description
1	HIJACKED	A User is logged out because another User logs in to the Service Point.
2	LOGOUT	The User logs out from the Service Point.
3	LOGOUT_FORCE	The User makes a forced logout, logging out without being able to complete the transaction, for example not register any Delivered Service.
4	LOGOUT_BY_OTHER_USER	Make a forced logout on another User. The User has not finished the transaction and is not available to end the transaction, so another User must make a forced logout.
5	SESSION_TIMEOUT	The session ended by a User session timeout.
6	PUBLISH	The session ended because of a publish.
7	BRANCH_DELETED	The session ended because the Branch was deleted.
8	USER_DELETED	The session ended because the User was deleted.
9	RESET	The session ended because of a reset

6.35.1 FactStaffSessions

Type: StaffSession

6.35.2 Id

The Staff Session Outcome identifier.

Type: Edm.Int64

6.35.3 Name

The name of the outcome, see table above.

Type: Edm.String

6.36 StaffSessionStartTime

This entity holds information about the start time of Staff Sessions.

6.36.1 AmPm

The string AM or PM.

Type: Edm.String

6.36.2 FactStaffSessions

Type: StaffSession

6.36.3 Hour

The hour on a 12 h format.

Example: When the time is 13:37 the hour is 1.

Type: Edm.Int32

6.36.4 Hour24

The hour on a 24 h format.

Example: When the time is 13:37 the hour is 13.

Type: Edm.Int32

6.36.5 Id

The Staff Session end time identifier.

Type: Edm.Int32

6.36.6 Interval10Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.36.7 Interval10Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.36.8 Interval15Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.36.9 Interval15Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.36.10 Interval30Min

When the time is 13:37 the minute is 01:30.

Type: Edm.String

6.36.11 Interval30Min24

When the time is 13:37 the minute is 13:30.

Type: Edm.String

6.36.12 Interval5Min

When the time is 13:37 the minute is 01:35.

Type: Edm.String

6.36.13 Interval5Min24

When the time is 13:37 the minute is 13:35.

Type: Edm.String

6.36.14 Minute

When the time is 13:37 the minute is 37.

Type: Edm.Int32

6.36.15 Time

Example: 1900-01-01 13:37:00.000

Type: Edm.DateTimeOffset

6.37 Visit

This entity holds information about Visits.

6.37.1 AppointmentId

An identifier that indicates that this Visit started as an Appointment. The id refers to the Appointment table.

6.37.2

Type: Edm.Int64

6.37.3 CreatedTimestamp

The timestamp for when the Visit was created.

Type: Edm.Int64

6.37.4 Custom1

Custom information, used for notes.

Type: Edm.String

6.37.5 Custom2

Custom information.

Type: Edm.String

6.37.6 Custom3

Custom information.

Type: Edm.String

6.37.7 Custom4

Custom information.

Type: Edm.String

6.37.8 Custom5

Custom information.

Type: Edm.String

6.37.9 DeliveredServices

Type: DeliveredService

6.37.10 FactVisitTransactions

Type: VisitTransaction

6.37.11 FeedbackResults

Type: FeedbackResult

6.37.12 Id

The Visit Identifier.

Type: Edm.Int32

6.37.13 TicketId

The ticket number.

Type: Edm.String

6.37.14 VipLevel

The VIP level of the Customer.

Type: Edm.String

6.37.15 VisitCustomers

Type: VisitCustomer

6.37.16 VisitType

The name of the Visit **type**. <--Appointment, normal, multi-service? -->

Type: Edm.String

6.38 VisitCallType

This entity holds information about how the Customer was called to the Visit.

The following values are defined:

Id	Name	Description
1	CHERRY_PICK	The ticket number was cherry picked (a specific ticket number called directly from the Queue).
2	WALK_DIRECT	The Customer walked directly to the Service Point, without taking any ticket.
3	WORK_PROFILE	The Customer was called, according to the Work Profile.

6.38.1 FactVisitTransactions

Type: VisitTransaction

6.38.2 Id

The Visit Call **Type** identifier.

Type: Edm.Int64

6.38.3 VisitCallType

The name of this Visit Call **Type**.

Type: Edm.String

6.39 VisitCustomer

This entity holds information about the Visit Customer.

6.39.1 CustomerId

The Customer identifier, defined in the Customer database.

Type: Edm.Int64

6.39.2 Id

Identifies the Customer. No Customer data is stored in the stat table, so there is no connection to any stat table with Customer info. This is a unique id for every Customer that has been identified and where Customer data has been found in a Customer database.

Type: Edm.Int64

6.39.3 Visit

Type: Visit

6.39.4 VisitKey

Reference to the id of a Visit in the Visit dimension table.

Type: Edm.Int64

6.40 VisitTransaction

6.40.1 Branch

Type: Branch

6.40.2 BranchKey

Reference to the id of a Branch in the table dim_branch.

Indicates that this entry is a Branch, not a Branch Group. If this value is NULL, the entry is a Branch Group.

Type: Edm.Int64

6.40.3 CallTime

Type: CallTime

6.40.4 CallTimeKey

Reference to a Call Time identifier in the Call Time dimension table.

Type: Edm.Int32

6.40.5 CallTimeOfDay

The time that a Customer was called.

Type: Edm.Int64

6.40.6 CallTimestamp

The timestamp for when the Customer was called.

Type: Edm.DateTimeOffset

6.40.7 Date

Type: Date

6.40.8 DateKey

Reference to a Date identifier in the Date dimension table.

Type: Edm.Int32

6.40.9 EndTime

Type: EndTime

6.40.10 EndTimeKey

Reference to an End Time identifier in the End Time dimension table.

Type: Edm.Int32

6.40.11 EndTimeOfDay

The Visit Transaction end time.

Type: Edm.Int64

6.40.12 EnterQueueTimeOfDay

The time that a Customer entered the Queue.

Type: Edm.Int64

6.40.13 EnteredQueueTime

Type: EnteredQueueTime

6.40.14 EnteredQueueTimeKey

Reference to an Entered Queue Time identifier in the Entered Queue Time dimension table.

Type: Edm.Int32

6.40.15 EnteredQueueTimestamp

The timestamp for when a Customer entered the Queue.

Type: Edm.DateTimeOffset

6.40.16 EntryPoint

Type: EntryPoint

6.40.17 EntryPointKey

Reference to an Entry Point identifier in the Entry Point dimension table.

Type: Edm.Int64

6.40.18 Id

The Visit transaction identifier.

Type: Edm.Int32

6.40.19 IsAppointment

Shows if the Visit is an Appointment or not.

Type: Edm.Boolean

6.40.20 Outcome

Type: Outcome

6.40.21 OutcomeKey

Reference to an Outcome identifier in the Outcome dimension table.

Type: Edm.Int64

6.40.22 Queue

Type: Queue

6.40.23 QueueKey

Reference to a Queue identifier in the Queue dimension table.

Type: Edm.Int64

6.40.24 QueueType

Type: QueueType

6.40.25 QueueTypeKey

Reference to a Queue **Type** identifier in the Queue **Type** dimension table.

Type: Edm.Int64

6.40.26 RecycleCount

The number of times the transaction was recycled. A recycle does not result in a new transaction.

Type: Edm.Int32

6.40.27 Service

Type: Service

6.40.28 ServiceKey

ServiceKey remains for backwards compatibility and is populated with the id for the first service if it is a multi service Appointment.

Type: Edm.Int64

6.40.29 ServicePoint

Type: ServicePoint

6.40.30 ServicePointKey

Reference to the id of an Service Point in the Service Point dimension table.

Type: Edm.Int64

6.40.31 ServiceTransactionTimeLevel

The Transaction Time Service Level, in seconds, from the Service configuration in the Business Configuration.

Type: Edm.Int64

6.40.32 Staff

Type: Staff

6.40.33 StaffKey

Reference to a Staff identifier in the Staff dimension table.

Type: Edm.Int64

6.40.34 StaffTransactionTime

The transaction time for a staff member is the total time spent for Customer transactions including store next, walk time, face-to-face time and wrap- up time.

Type: Edm.Int64

6.40.35 TimeSeconds

The seconds part of the time stamp.

Type: Edm.Int32

6.40.36 TransactionTime

The time from the Customer call was shown on the main display, until the transaction ends (next Customer is called, Visit ends, Customer is transferred to another Queue or Counter is closed) Note that the last VISIT_CALL is used when recall has been used (recall time is not included in the transaction time).

Type: Edm.Int64

6.40.37 TransactionTimeVsServiceLevel

The relationship between the Transaction Time and the Service Level, i.e. if the Transaction Time is Above or Below the Service Level.

Type: Edm.String

6.40.38 VerticalMessageKey

Reference to a Vertical Message in the Vertical Message dimension table.

This table holds information about Vertical Messages that are configured in the Surface Editor and Context Marketing applications. Every time a new or updated Vertical Message is encountered in an inserted Visit, a new entry is generated in the table. When a Vertical Message is updated, an end date will be set to the previous version.

Type: Edm.Int64

6.40.39 Visit

Type: Visit

6.40.40 VisitCallType

Type: VisitCallType

6.40.41 VisitCallTypeKey

This is how the Customer was called and is only valid for events calling Customers, otherwise it is NULL.

Type: Edm.Int64

6.40.42 VisitKey

The Visit key is unique for the Visit (ticket number).

Type: Edm.Int64

6.40.43 VisitOutcome

Type: VisitTransactionOutcome

6.40.44 VisitOutcomeKey

The outcome of the transaction. There is only a value for events that ended a transaction, otherwise it is NULL.

Type: Edm.Int64

6.40.45 WaitingTime

Single Service: The time from the customer entered the queue by taking a ticket or a transfer and until the call is shown on the main display

Multi Service: This is when multiple Services are delivered in the same transaction. The waiting time for the first transaction is the same as described above. For the following transactions, started with the ADD_SERVICE event, the waiting time is NULL because the Customer was not in any Queue before being served.

Type: Edm.Int64

6.40.46 WaitingTimeServiceLevel

The Waiting Time Service Level, in seconds, from the Queue configuration in the Business Configuration. This is 0 when the Customer is called from a Service Point Pool, or a Staff Pool.

Type: Edm.Int64

6.40.47 WaitingTimeVsServiceLevel

The relationship between the Waiting Time and the Service Level, i.e. if the Waiting Time is Above or Below the Service Level.

Type: Edm.String

6.40.48 WalkingTime

The time from the call was shown on the main display and until the Customer arrived to the Counter.

Type: Edm.Int64

6.40.49 WorkProfile

Type: WorkProfile

6.40.50 WorkProfileKey

Reference to a Work Profile identifier in the Work Profile dimension table.

The Work Profile key has a value for the SET_WORK_PROFILE event and the ADD_USER_TO_STORE_NEXT event, otherwise it is NULL.

Type: Edm.Int64

6.40.51 WrapupTime

The time from wrap-up started and until the transaction ended.

Type: Edm.Int64

6.41 VisitTransactionOutcome

This table holds information about the different **types** of Visit outcomes that are possible for a Service in a Visit (visit transaction).

The following values are defined:

Id	Name	Description
1	NORMAL	The transaction ended when the next Customer was called, or as the end of the Visit (the last transaction in the Visit).
2	NO_SHOW	The Customer never responded to the call and was never served.
3	REMOVE	A Staff member manually removed the Visit from the Queue (the ticket number was deleted from the Queue).
4	RECYCLE	The ticket number was placed back into the same Queue from which it was called.
5	TRANSFER_TO_QUEUE	The Visit was transferred to another Queue and a new transaction started.
6	TRANSFER_TO_SERVICE_POINT	The Visit was transferred to a Service Point Pool and a new transaction started.
7	TRANSFER_TO_STAFF	The Visit was transferred to a Staff Pool and a new transaction started.
8	REMOVE_BY_RESET	The Visit was removed by a Branch reset.
9	REMOVE_BY_CUSTOMER	A Customer removed the ticket number from the Queue, for example used in Mobile solutions.
10	END_BY_FORCE_LOGOUT	A Staff member did not logout before leaving the counter when a transaction was ongoing and a forced logout was performed to end the staff session and the ongoing transaction.
11	REMOVE_BY_PUBLISH	A publish was done and removed the Visit.
12	ENDED_BY_LOGOUT	The Staff member logged out from the counter and ended the Visit.
13	ENDED_BY_SHIRO_TIMEOUT	The session and Visit ended due to a shiro timeout.

6.41.1 FactVisitTransactions

Type: VisitTransaction

6.41.2 Id

The Visit Transaction identifier.

Type: Edm.Int32

6.41.3 VisitTransactionOutcome

The name of the outcome, see table above.

Type: Edm.String

6.42 WorkProfile

6.42.1 BranchOriginId

The unique identifier of the Branch, used to keep track of changes. This value is never changed.

Type: Edm.Int64

6.42.2 ClientKey

The id of an entry in the table dim_client which holds the license.

Type: Edm.Int64

6.42.3 EndDate

Time stamp for when this entity was deleted or updated. If this value is NULL, it means that it is the active instance of this entity.

Type: Edm.DateTimeOffset

6.42.4 FactVisitTransactions

Type: VisitTransaction

6.42.5 Id

The Work Profile identifier.

Type: Edm.Int32

6.42.6 OriginId

The appointment id automatically assigned by the system.

Type: Edm.Int32

6.42.7 StartDate

Time stamp for when this entity was created.

Type: Edm.DateTimeOffset

6.42.8 WorkProfile

The name of the Work Profile.

Type: Edm.String

Q-MATIC

Q-MATIC AB,
Box 198,
SE-431 23 Mölndal,
Sweden.

VAT No SE556212749701

Phone: +46 31 756 46 00
Email: info@qmatic.com
Web: www.qmatic.com