Discoverant SimApi User Guide

Contents

1 Introduction

This document is the user guide for the **Discoverant SimApi** from Sartorius Stedim.

The SimApi uses ODBC to connect to an BIOVIA Discoverant server. For an important list of requirements for this SimApi, see 1.2 below. For more information about Discoverant, see the <u>Accelrys web site</u>.

For a detailed list of changes in different versions of this SimApi, see the **Version Info.txt** file that comes with the installation.

This SimApi can be used by SIMCA-online, SIMCA-Batch On-Line, SIMCA-4000, or SIMCA.

For more information on available SimApis, see <u>https://www.sartorius.com/umetrics-simapi</u>.

1.1 Features

This SimApi implements:

- Configurable set of AG_PATHs to expose through the SimApi (optional – default is to expose all AG_PATHs on the server).
- Read current and historical continuous / process data (AG_C_DATA).
- Read batch data (used for batch conditions and local centering in SIMCA-online) (AG_D_DATA).
- Read discrete data, i.e. IPC or lab data (AG_R_DATA).
- Optional support for reading batch data and discrete data for batches matching the batch IDs in another SimApi's batch node
- Read batch information / batch node.
- Aggregated batch nodes combining batches in different analysis groups / AG paths

The following two items have not been implemented:

- Write back of process or batch data through the SimApi.
- Phase information from batch data are not available from the batch node. Phase tags needs to exist in the AG_C_DATA or in another data source with process data. This means the batch node cannot be used in phase conditions in SIMCA-online to determine when phases execute.

Limitation

- In order for the SimApi to be able to read data correctly there can only be one analysis group per unit operation. The reason for this is because of how the SimApi requests data compared to how data is structured in Discoverant.

A note on nomenclature;

- Continuous data, AG_C_DATA corresponds to what Sartorius Stedim calls continuous process data.
- Discrete data, AG_D_DATA corresponds to what Sartorius Stedim calls batch data.
- Replicate data, AG_R_DATA corresponds to what Sartorius Stedim calls discrete data.

The nomenclature for Discoverant will be used throughout this document.

1.1.1 AG_D_DATA

There are two ways to configure how discrete data will be read. One simple way where no configuration is needed and one way where each AG path is configured with a specific tag names the start time- and the stop time columns.

1. Simplest way, no configuration is required. All AG paths are loaded and default column name is used for start and stop time, see section 2.2 for further details.

2. Each AG path is configured against a specific start and stop time column in the XML file, see The XML Configuration File. This should be used when the default values for the start and stop time columns don't apply for the specific AG path. Note that when AG paths are configured in this section only the configured AG paths will be exposed through the SimApi.

1.1.2 AG_R_DATA

AG_R_Data in Discoverant corresponds to what Sartorius Stedim calls Discrete data. Discrete data is infrequently measured data which have no logical values in between measurements. Usually a sample is taken on each batch at semi-regular intervals (such as once every day). This sample is then sent to a lab which performs analysis on the

sample and at a later stage returned with a report on the sample for the required variables. This is then entered in the database in the discrete data table.

For more information on discrete data, see **SIMCA-online discrete data retrieval.pdf** found on the SIMCA-online download page on www.umetrics.com.

There are two ways to configure how replicate data is read. One simple way where no configuration is needed and one way where each discrete tag is configured with a "group by" tag so that an average of multiple samples for the same age/maturity/time-point can be returned as read value.

1. Simplest way, no configuration is needed. However this assumes that there is only one sample per age/maturity/time-point. It also assumes that all replicate tags that will be used from one AG path have the same number of values. The reason for this is because there are no time points for the replicate values in Discoverant to align data against.

    The following example from Discoverant shows two discrete tags, **Replicate 1** and **Replicate 3** for batch

PS_01 in the AG path **MixTest**.

2. Each replicate tag is configured in the enclosed XML file, see 2.2. This should be used when there are more than one data sample per age/maturity/time-point. When this is used an average value of all samples for a certain age/maturity/time-point is used. In order for this to work a replicate id number needs to be configured for the concerned replicate tag so that the SimApi can group all samples belonging to a certain age/maturity/time-point. See example below.

    The following example from Discoverant shows one discrete tag, **pH** for batch **103078** in the AG path **Stability Summary**. Here we can see there are ten samples for each age/maturity/time-point. In this case we would configure the discrete tag **pH** to be grouped by **AGE_**, i.e. replicate id number **2**. This would mean that all samples with **AGE_** 0 would be grouped together so an average value could be calculated. The same for **AGE_** 3, 6 and so on.

1.1.3 Batch data and discrete data for batches in another SimApis batch node

The optional setting BatchIDField can be used when you want to read batch data or discrete data from Discoverant for batches defined in another SimApi's batch node (that batch node provides the batch identifiers and start and stop times).

To use this feature, you:

*   create a 'derived paramameter' in Discoverant that provides the same batch ID as used in the other SimApi. This derived parameter must be of text/string type (cannot be numerical). If a batch is called Batch1234 in the other SimApi the derived parameter should also have the value Batch1234 for the Discoverant SimApi to be able to return the right batch- or discrete data.

*   provide the name of the derived parameter in the setting BatchIDField in the configuration file. This setting exists at the global level but can also be specified for individual AG paths if those are specified in the configuration file.

This setting is not used for the batch node functionality in Discoverant, only for batch- and discrete data.

1.1.4 Aggregated batch node

There can only be one analysis group per unit operation due to how the SimApi requests data compared to how data is structured in Discoverant. This limitation forces SIMCA-online to have only one unit per configuration since only one batch node can be selected.

If you want to monitor several units in a single project configuration in SIMCA-online, you can use an **Aggregated batch node** in the SimApi. This batch node groups batches from several AG paths into a single aggregated view.

The aggregated batch node does not contain any data tags and can only be used as a batch node to get information on batches such as batch id, start time and stop time over a time period. You specify this node in the Batch node page of a project configuration in SIMCA-online to use it.

When this node is used, the SimApi will loop over all configured AG paths and return batches from them. Note that the batch id must be unique over the AG paths, otherwise the first found batch id with belonging start and stop times will be returned.

For example: If one AG path has batches A, B, and C, and another AG path has D and E, the aggregated batch node of both would return A, B, C, D and E.

See 2.2 for details on how to configure this in the configuration file.

1.1.5 About using Extract for Discrete data from ODBC in SIMCA-online

The Extract functionality in SIMCA-online (found on the File tab) can be used look at discrete data from the data source. To get this to work you must select the discrete tags from a node, but also at least one variable from the batch node. This is required for SIMCA-online to know for which batch to look for discrete data.

See **SIMCA-online Discrete Data Retrieval.pdf** for more information about using Extract with multiple SimApis.

Tip: A way of testing discrete data queries is to enable debug level logging of the SimApi and look at its log files which contain the SQL queries used. These queries can then be run by hand in a query tool from the database manufacturer.

1.2 Prerequisites

In order for the Discoverant SimApi to work, you must first install ODBC drivers for Discoverant on the machine where that application resides. That is done with the standard Discoverant installer (Obtain the Discoverant setup program from Accelrys). This actually does two things: it installs the correct drivers, and it also sets up a ODBC data source in Windows.

After launching the installer, and navigating through the first couple of screens, you would indicate that you want to install only the ODBC client, as seen here (a 4.2 installer is used, but it's the same for all versions):
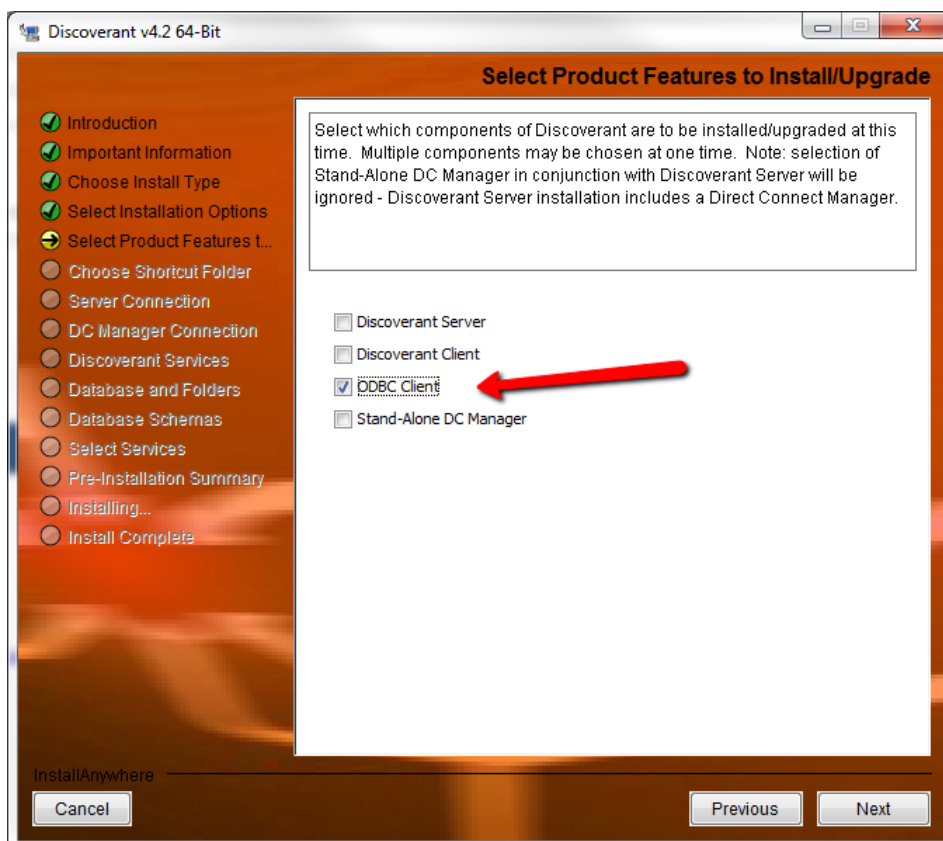
A few screens later, you indicate the name of the Discoverant server you'll be pulling data from. In this example, machine is BOL-JOHNT, so that's what's specified. You would specify your own Discoverant server name.
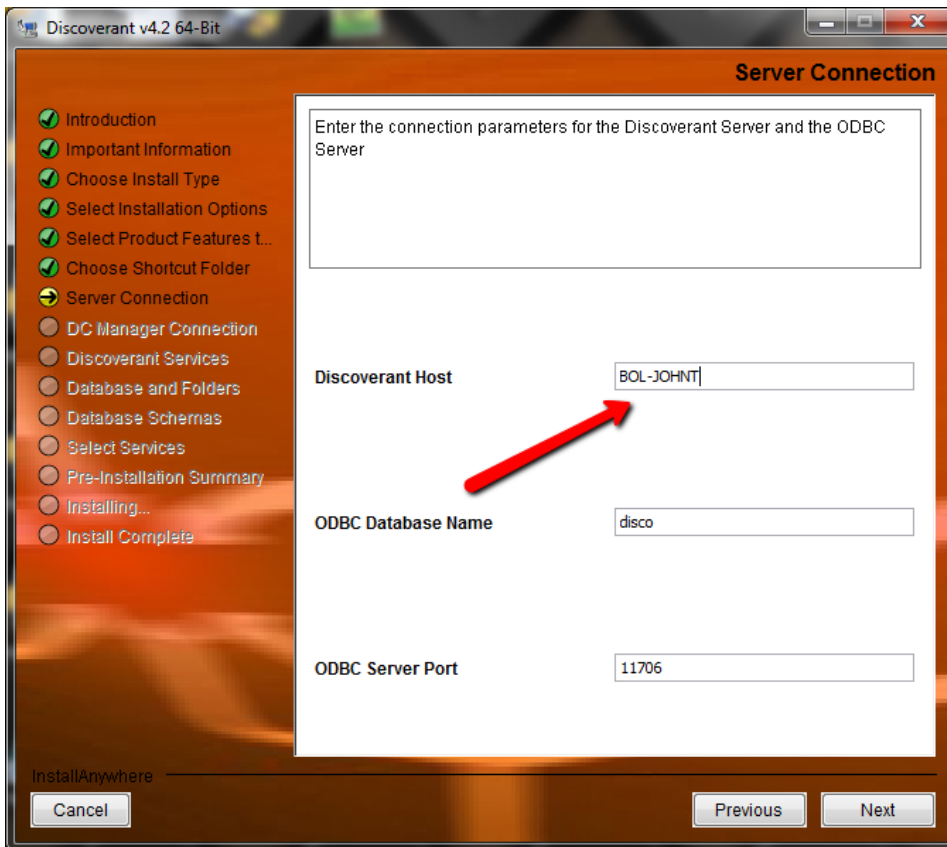
Once the install is completed, in Windows, you can navigate to Administrative Tools -> Data Sources (ODBC), and select the System DNS tab. You should see the newly installed ODBC driver for Discoverant, as seen here.
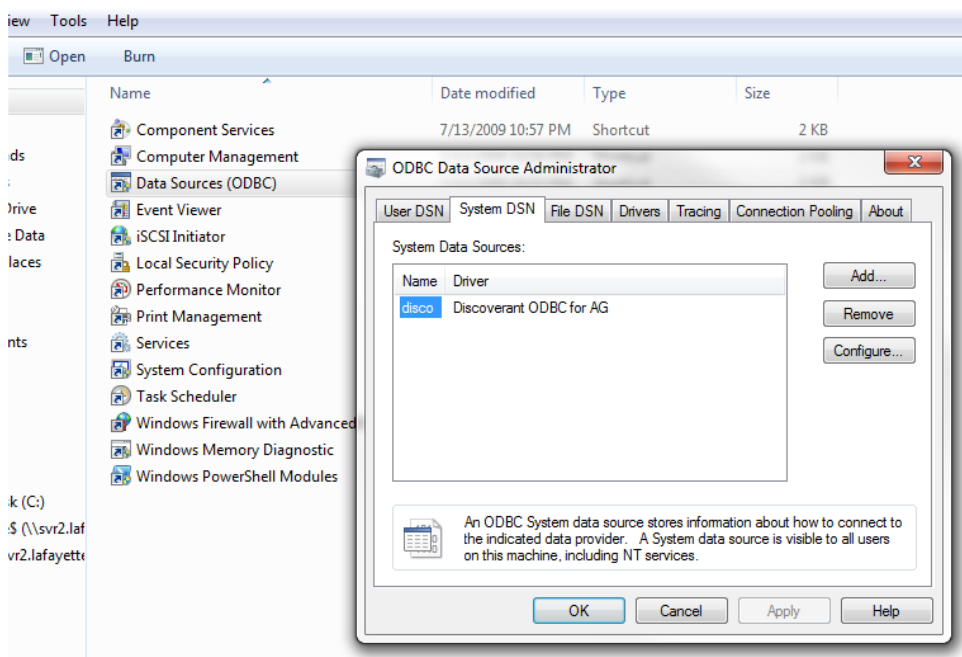
And if you press the Configure button, you will see the values that you indicated during the installation.

This DSN should be specified in the configuration (XML) file of the Discoverant SimApi (see below).

1.2.1   The Discoverant Server

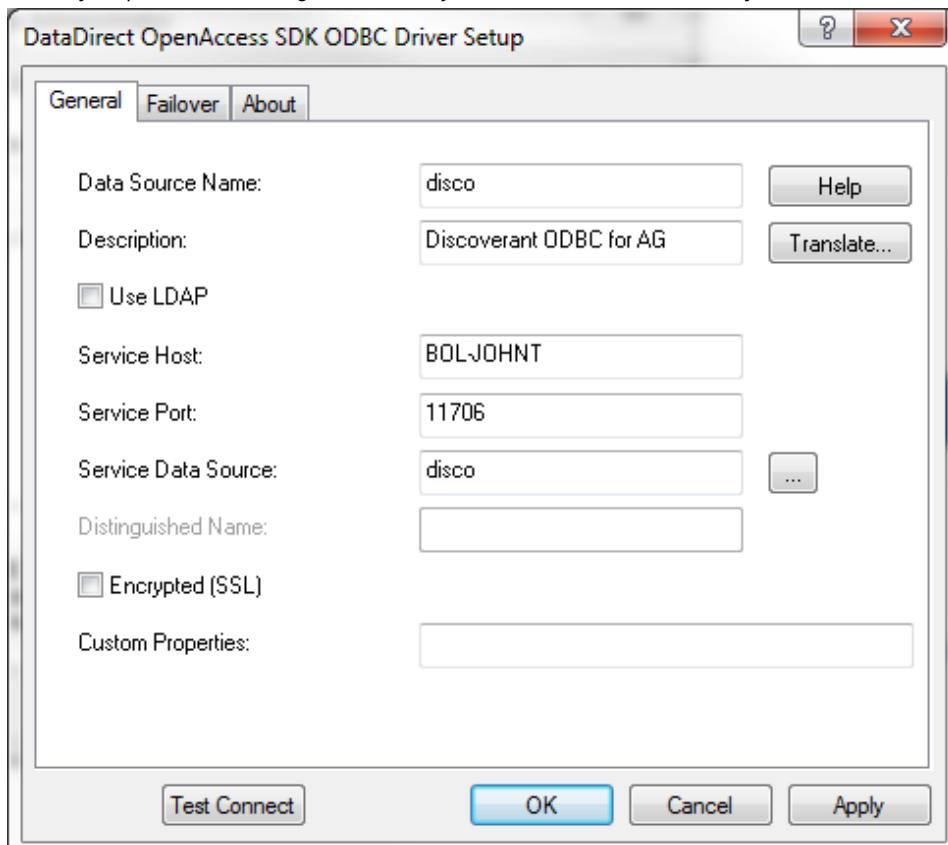The Discoverant server version 4.3 must have the hot fix from December 2013. Otherwise the SimApi will not be able to read data from different AG paths.

1.3   Troubleshooting

Make sure you have installed and configure the Discoverant ODBC client. Test it with another Discoverant client on the same computer that you want to use the SimApi on.
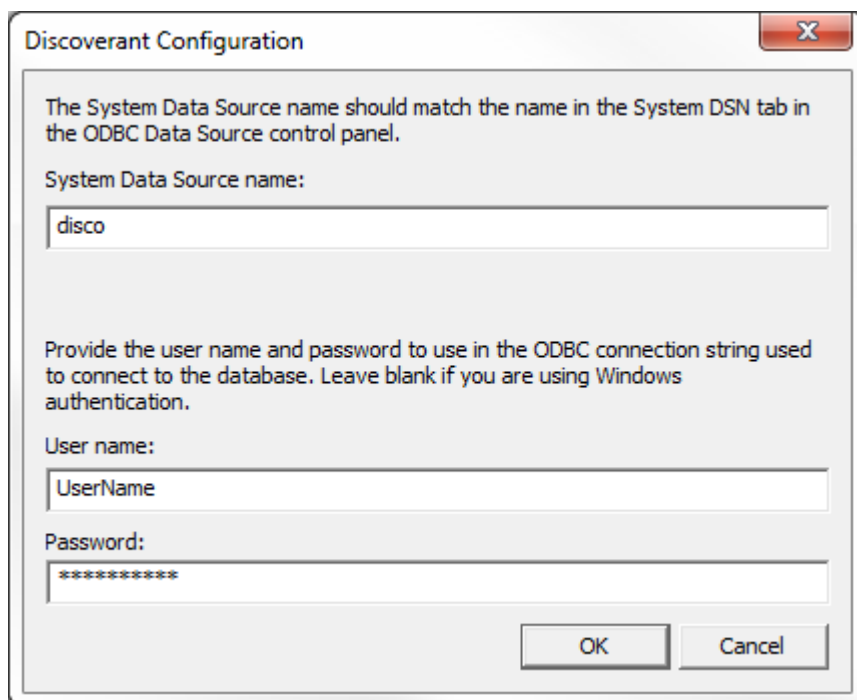
See the SIMCA-online Technical Guide for more information on SimApi troubleshooting, including issues that can occur when trying to use a 32-bit SimApi on a 64-bit windows computer.

2 Installation and Setup

Refer to the **SimApi Overview User Guide** located at umetrics.com/simapi for how to install and setup this SimApi.

2.1   Discoverant SimApi Configuration

In the configuration dialog you can provide the data source name (from the system tab of the ODBC Data Sources Windows control panel as defined in 1.2 above, the user name and password that will be used to connect to Discoverant. The information is stored in encrypted form. If you leave user name and password empty, the SimApi will connect as the process that calls the SimApi (meaning offline SIMCA or in the case of SIMCA-online the account configured to run the SIMCA-online server in the services control panel).

Optionally there are some settings that can be manually set in the XML-file which is described next.

2.2   The XML Configuration File

Below the available settings in the XML file are shown. Some of the settings are made using the graphical interface above, but others have to be made by hand in the file. These settings are marked bold below, and described in the sub sections.

Log file specific settings	Explanation
LogFileSize	The maximum allowed size of the log file before the file is truncated
LogLevel	An integer from 0 to 4. The higher the value the more information is printed to the log file (log level 3 = information).
Connection specific settings	
DSN	The Data Source Name that has been setup by Discoverant in the Windows *ODBC Data Sources* control panel program
Credentials	Encrypted information about user name and password as entered in the graphical configuration utility
QueryTimeout	The time, in seconds, before a query or connection to the database will timeout and fail. The default is 20
StartTimeField	The name of the field/column in the AG path that contains the start time for batch. This setting is used for all AG paths that contains batch information, if different AG paths have different start time fields use the **agpaths** setting below to configure this for each AG path.
StopTimeField	The name of the field/column in the AG path that contains the stop time for batch. This setting is used for all AG paths that contains batch information, if different AG paths

	have different stop time fields use the **agpaths** setting below to configure this for each AG path.
BatchIDField	Optional feature. The name of the field/column in the AG path that contains the batch identifiers used for batch data and discrete data. This setting is used for all AG paths that contains batch information, if different AG paths have different stop time fields use the **agpaths** setting below to configure this for each AG path.
agpaths	One or more AG_PATHs to expose through the SimApi. One <agpath> element (row in the example below) for each AG path. The name of the AG path, the start and stop time fields, and the optional BatchIDField are configured for each element. The following is an example with two configured AG paths:

```
<agpaths>
  <agpath Name="Imports:\Test" StartTimeField="Parameter Set Date"
StopTimeField="Stop" BatchIDField="PS_NAME" />
  <agpath Name="Imports:\umetrics\MixTest" StartTimeField="Start"
StopTimeField="Stop" BatchIDField="PS_NAME />
</agpaths>
```

	Note that when AG paths are configured in this section no AG paths are read from the AG_MANIFEST, only the configured AG paths will be exposed through the SimApi.
discretetags	All discrete tags that are grouped by a replicate id needs to be configured here. One row (<discretetag> element) for each discrete tag. The name of the discrete tag and the AG path it belongs to and by which replicate id it should be grouped by are configured for each row. The following is an example with two configured discrete tags:

```
<discretetags>
  <discretetag Name="pH" AGPath="Imports:\Test" GroupBy="1" />
  <discretetag Name="Dissolution" AGPath="Imports:\Test" GroupBy="2" />
</ discretetags >
```

AggregatedBatchNode	You can create one or more AggregatedBatchNode elements in the XML file. Give each node a unique name.
	Define one or more AG_PATHs that should be used by this virtual batch node. One <agpath> element (row in the example below) for each AG path. The name of the AG path and the start and stop time fields are configured for each element.
	The following is an example with two configured AG paths:

```
<AggregatedBatchNode Name="MyAggregatedNode">
  <agpath Name="Imports:\Test" StartTimeField="Parameter Set Date"
StopTimeField="Stop" />
  <agpath Name="Imports:\umetrics\MixTest" StartTimeField="Start"
StopTimeField="Stop" />
</agpaths>
```

3 Support

umetrics_support-SIMCA-online@sartorius.com

Discoverant SimApi User Guide