Connect Web User's Guide

LABVANTAGE Connect version 8.0.0

Document version 2.6

Connect

Solution for instrument and system interfaces

by Software Point Oy

Connect is a solution for interfacing instruments and systems with LIMS. Instrument and system interfaces developed with Connect can be reused and thus save time and costs in implementation projects.

This page intentionally starts on an odd page, so that it is on the right half of an open book from the readers point of view. This is the reason why the previous page was blank (the previous page is the back side of the cover).

Connect Web User's Guide

Printed: 2016 in Espoo, Finland

Publisher

Software Point Oy

Technical Editors

Raimo Oinas

Table of Contents

1 Welcome

Welcome to the LABVANTAGE Connect Web User Guide.

This document contains the instructions of how to use the LABVANTAGE Connect software with the web user interface.

Program and document info:

Connect version: 8.0.0
Document version: 2.6
Published: 2016-02-23

2 Contents

Overview of Connect admin interface is shown in chapter Getting started.

Connect application configuration instructions are given in the chapter Connect Application configuration.

Connect interface configuration instructions are presented in chapter Connect LABVANTAGE interface configuration.

2.1 Related documents

Information about the installation of the Connect is presented in the document "*Connect Installation Guide*".

3 Getting started

To use the Connect web user interface, the Connect Service has to be running (either as a service / daemon or in the console mode).

To open the Connect web user interface, start up a browser and enter the URL of the Connect web GUI. By default the web GUI listens port 8580, so you would enter URL http://<your_connect_server_name>:8580 on your browser.

When using the Connect web GUI from LABVANTAGE LIMS, you only need to go to the Lab Admin Sitemap and click the "Connect Admin" tram stop as shown below.

3.1 Instrument categories

Connect handles instrument and system connections differently based on their category. The categories as described in the table below:

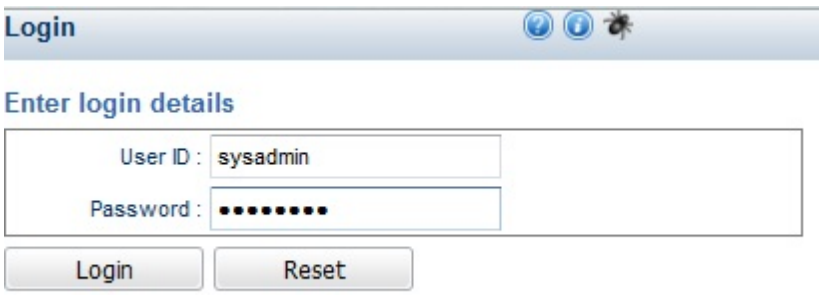
Category:	Description:
Cat I	Simple instruments producing one or more results for unidentified samples and parameters , e.g. balances and pH-meters.
Cat II	Instruments with one-way communication producing data containing information on sample ID, test ID and results.
Cat III	Instruments with two-way communication excluding CDS and special instruments.
Cat IV	Complex and unique instruments not included in the library and system integration.

These categories will be referred to in other parts of the document. The term "simple instruments" in any part of the document refers to Category I instruments and system, and the term "complex instruments" refers to all other categories, i.e. Category II, III and IV instruments and systems.

3.2 Using the Connect Application web GUI

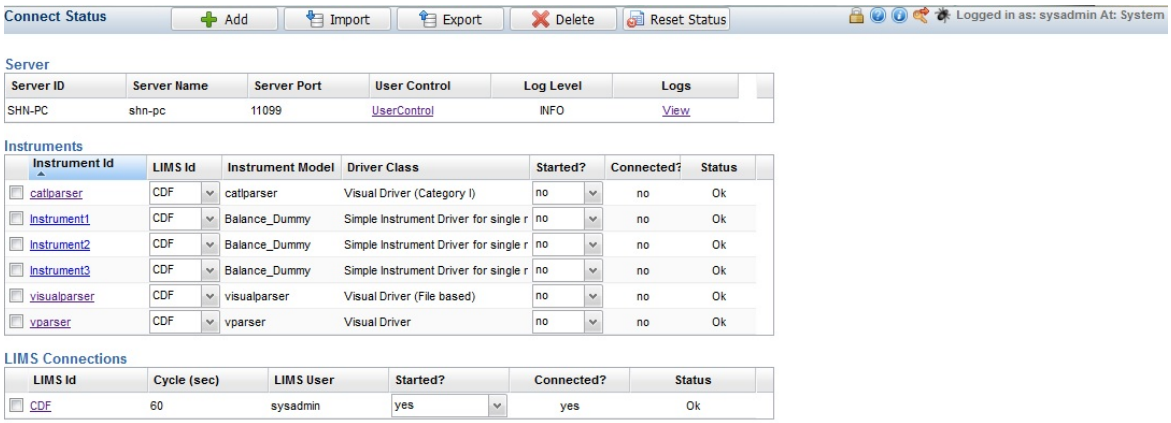
Using the Connect web GUI requires login to the Connect server. LABVANTAGE LIMS can be configured so that it has autologin enabled for certain users and therefore opening the Connect web GUI from the LABVANTAGE Sitemap will directly go to the Connect web GUI main page.

The login page works simply by entering your login details (user id and password) and by clicking the Login button. You can reset the User ID and password fields by pressing the reset button.



After successfully logged in you will be redirected to the main view, which show the status of the Connect server as well as all the instruments and LIMS connections of the logged-in user's location.

The default Connect user that comes with the installation is sysadmin.



4 Connect Application

Connect configuration means controlling of instances with their properties. The instances are of the following types:

- Instruments
- LIMS connections

Configured instances are shown on the list pages of the Connect web GUI. You can add new instances or modify existing ones by defining instance properties on the maintenance forms.

Property values can be texts or numbers. A special "*NA*" value should be given to a property that is not needed for the instance. For properties that contain values, which

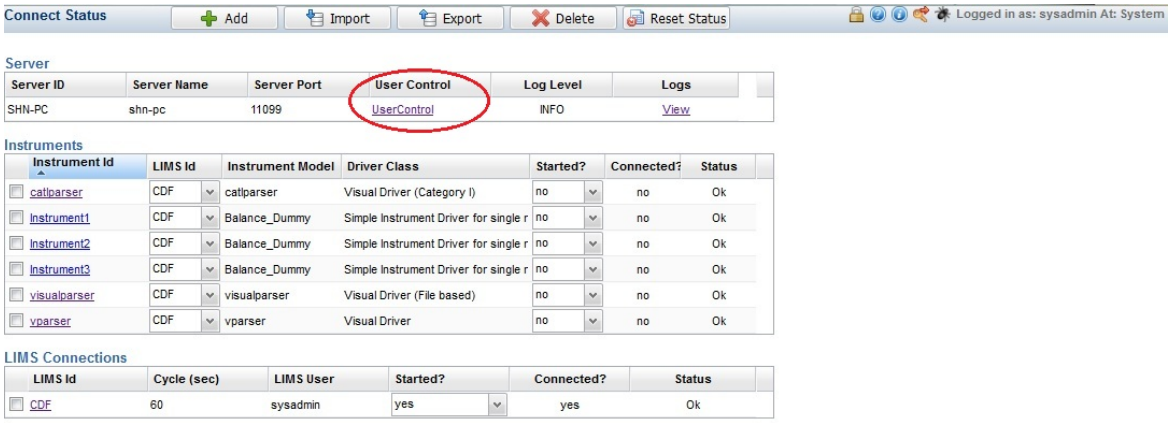
cannot be typed in as characters, you can enter them by their ASCII code with #(XXX), where XXX is character's ASCII code in decimal.

The web GUI has tools for exporting and importing instances. This gives you the possibility to e.g. transfer Connect configuration from one system to another (e.g. from test system to a production system).

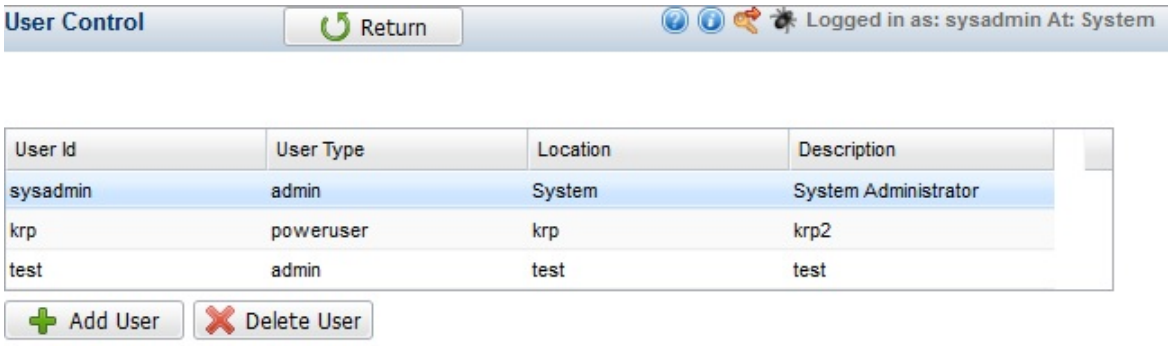
4.1 User Control

LABVANTAGE Connect also offers the possibility to manager users. Editing the list of users is restricted to sysadmin at the System location only.

The User Control can be accessed from the Server information:



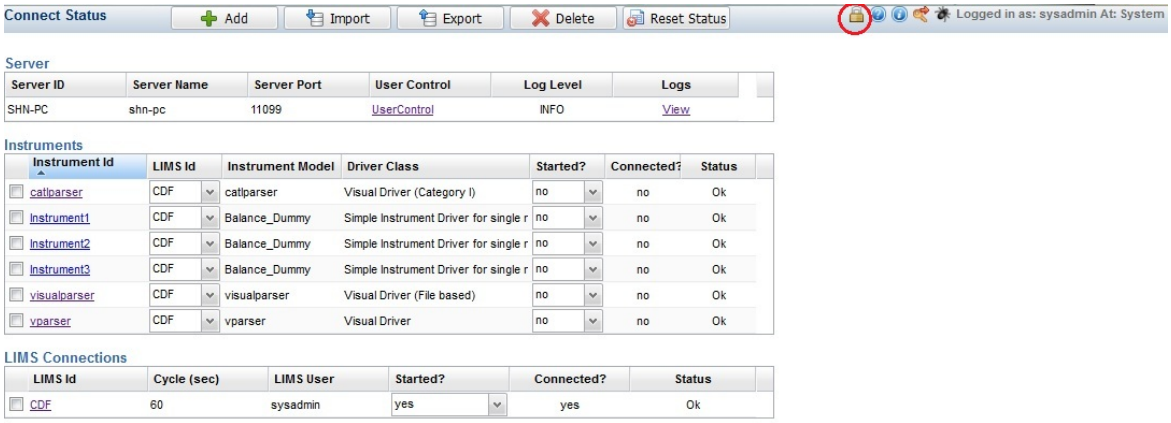
After clicking the link the following page is opened:



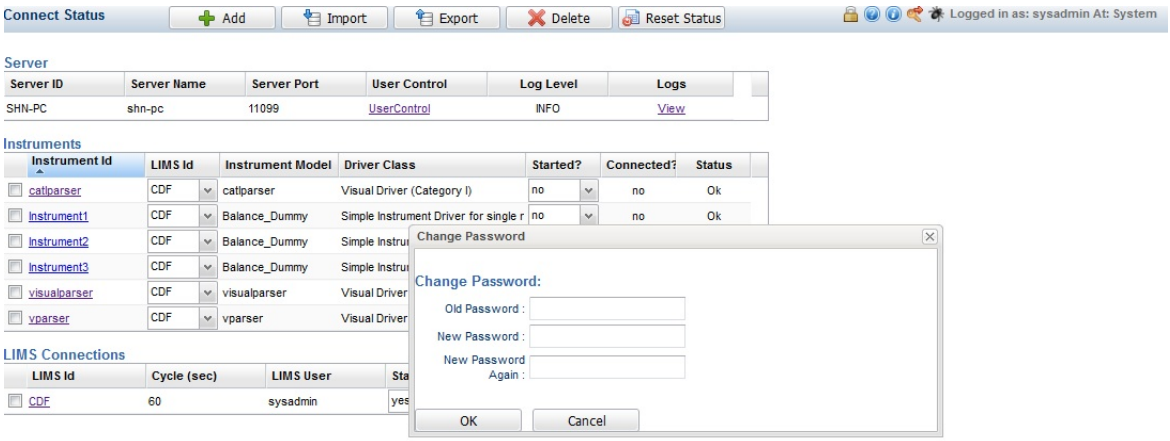
The User Control actions that can be performed are:
Changing Password
Adding Users
Deleting Users
Editing Users

4.1.1 Changing Password

All users can change their password. The password is changed from the Status page:



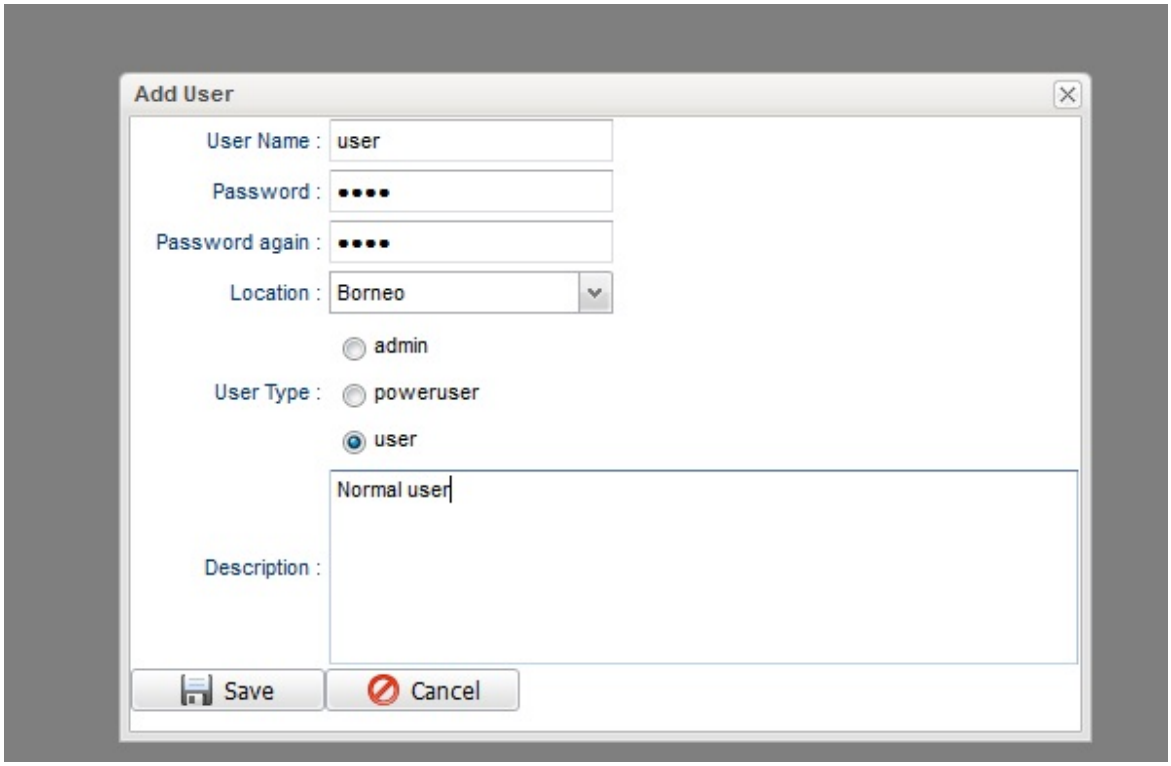
Once the change password button has pressed, a dialogue to change the password appears. The new password has to be entered twice:



4.1.2 Adding Users

You can add a new user by clicking the Add User button on the User Control page.

After clicking on the button a window will open where you can enter the user credentials (description is optional):



Click save to save the new user.

4.1.3 Deleting Users

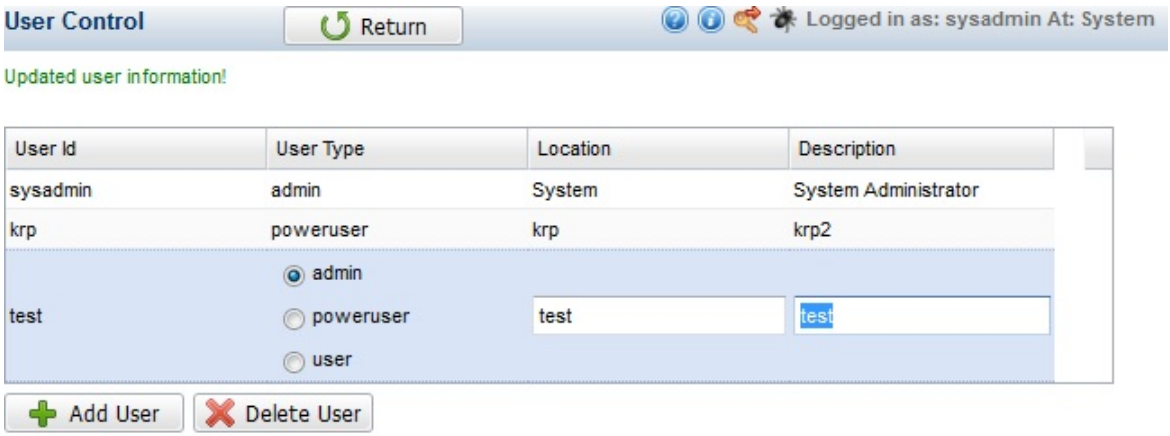
To delete a user, simply select the user you wish to delete from the grid and click delete.

4.1.4 Editing Users

Existing users can also be edited. The editing is performed directly from the grid. Double click on the user you wish to edit and then enter the new values.
Editable fields are:

- User Type
- Location
- Description

The fields are automatically updated to the server once the editing is finished:



4.2 Import and export

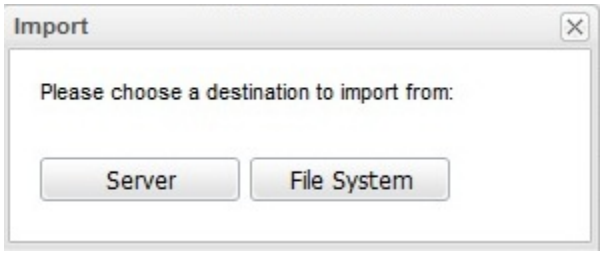
Instruments can be imported and exported from / to Connect either using the local file system or the instrument export directory on the server-side. LIMS connections cannot be exported and can only be imported from the server.

The installation of the client includes default LIMS connections under the "*LIMSConnections*" folder, as well as exported instrument instances under the "*Instruments*" folder. These imports (XML files) are given to you for convenience so that you can easily take them into use by importing them and then modifying their properties according to your choice.

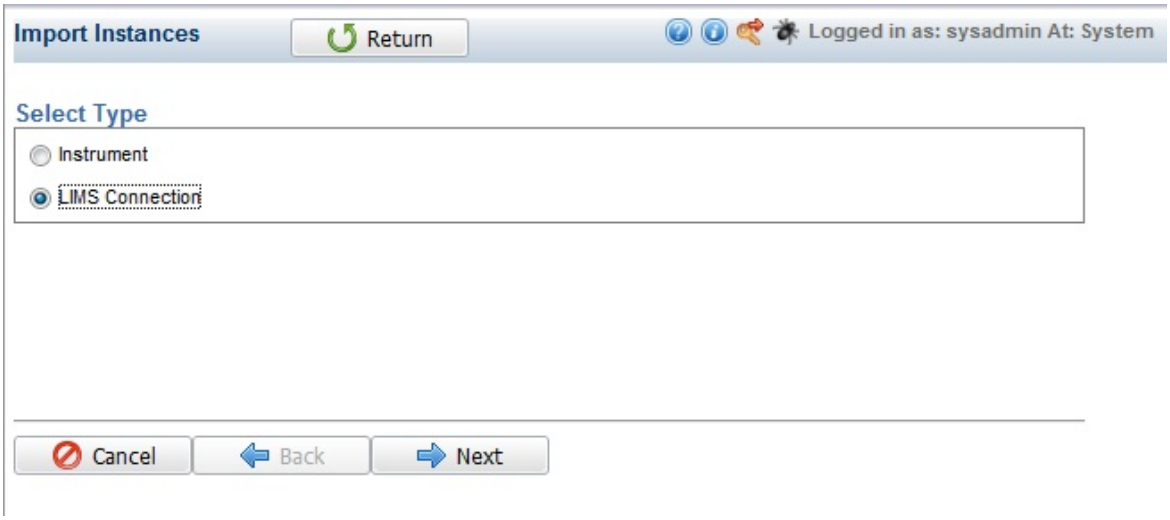
Each imported instrument has a default LIMS connection defined. Therefore, you need to import (or create) a LIMS connection before you can import or create any instrument instances.

4.2.1 Importing a LIMS connection

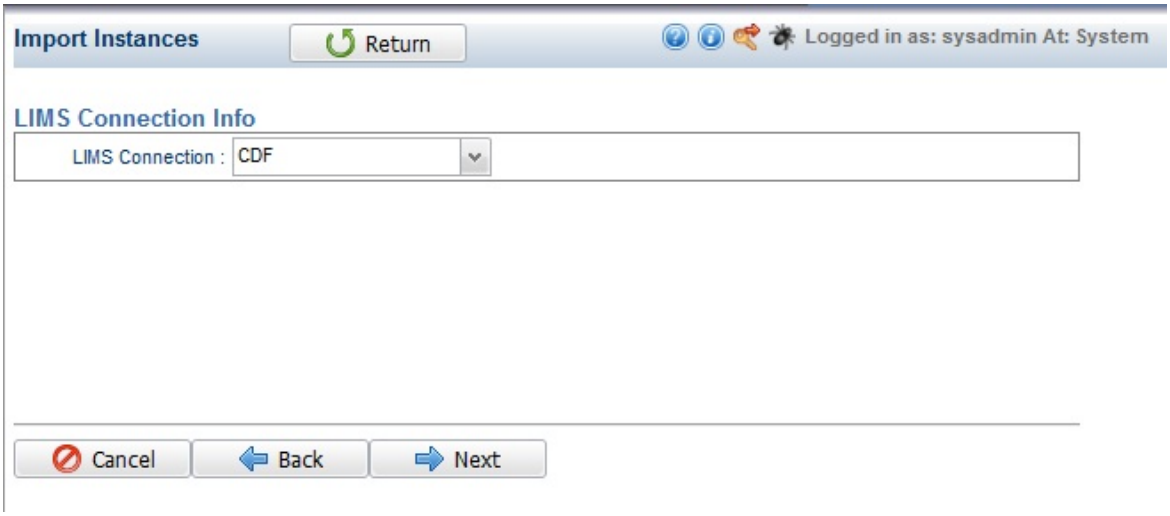
To import a LIMS connection, click the “Import” button. A dialog will open where you can select to import either from the Connect server or from a file system of your workstation.



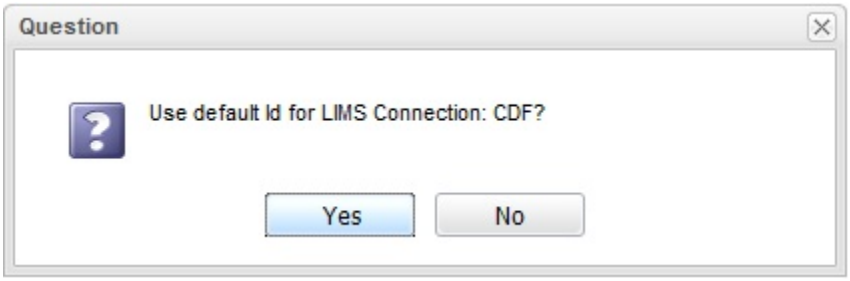
You can import LIMS connections only from the server, so click the Server button. A new page will open up where you can choose to import either an instrument or a LIMS Connection. Choose LIMS Connection and click Next.



This will lead to the next step where you can choose the type of LIMS Connection:

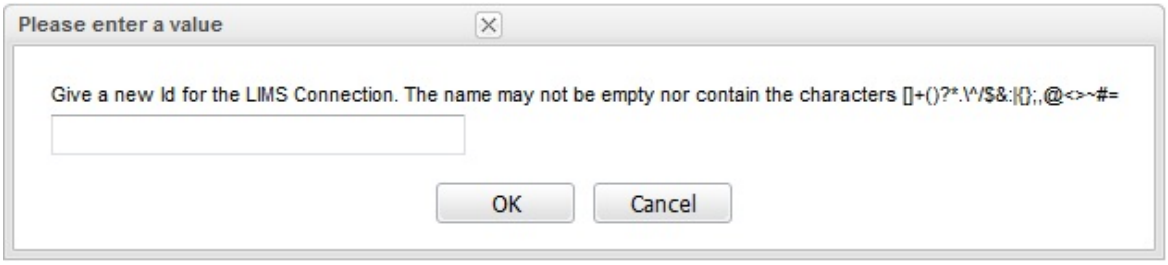


After you have chosen the LIMS Connection, press next again. A prompt will open up asking if you want to use the default name for the instrument:



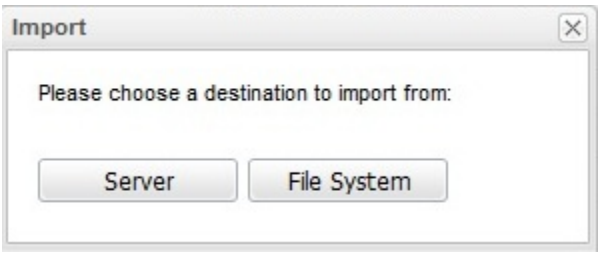
After you have chosen the name, the LIMS Connection will be created and you will be redirected to the front page.

If the name you chose already exists, you will be asked to enter a new name:



4.2.2 Importing instrument instances

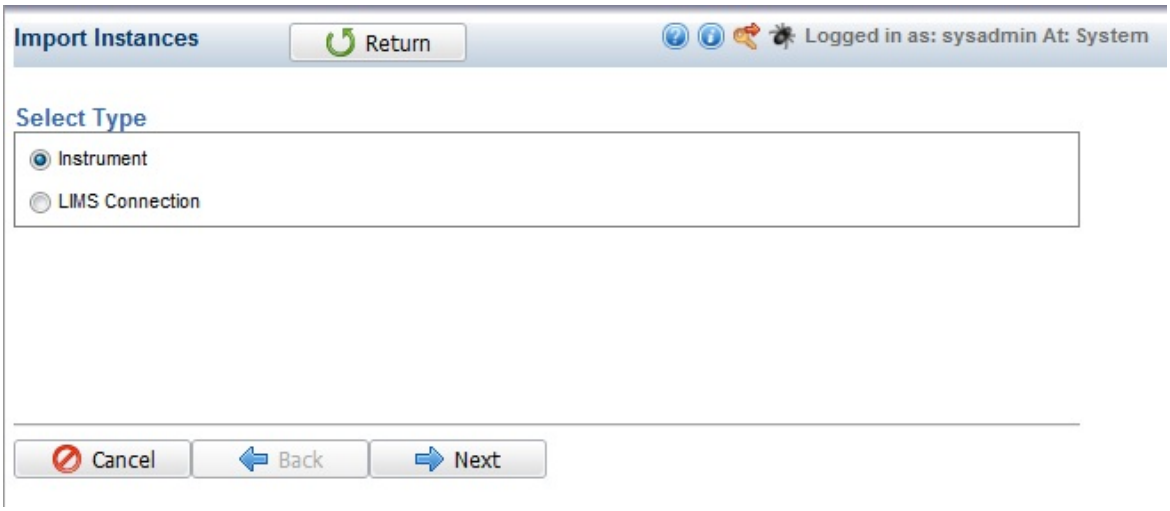
To import a instrument instances, click the “Import” button. A dialog will open where you can select to import either from the Connect server or from a file system of your workstation.



Choosing the Server option will lead you to Connect Instrument library. File system is usually selected in cases where driver was made for project at hand and so the driver not yet exist in connect standard driver library. In that case the one who has developed the driver should have provided the <drivername>.xml file that contains driver configuration.

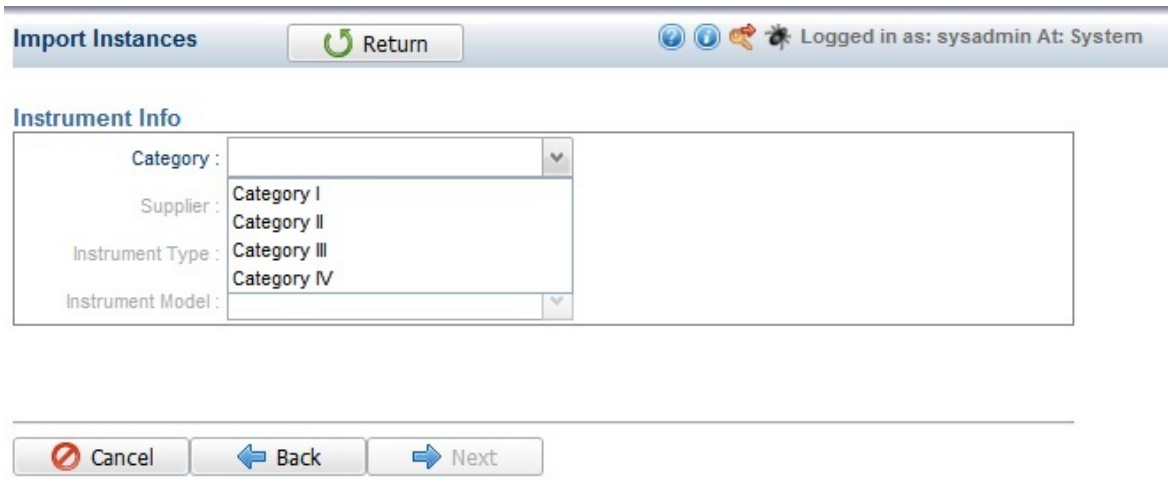
4.2.2.1 Importing from server

If you want to import from the server, click the Server button. This will open up a new page where you can choose to import an instrument or a LIMS connection. Select Instrument and click Next.

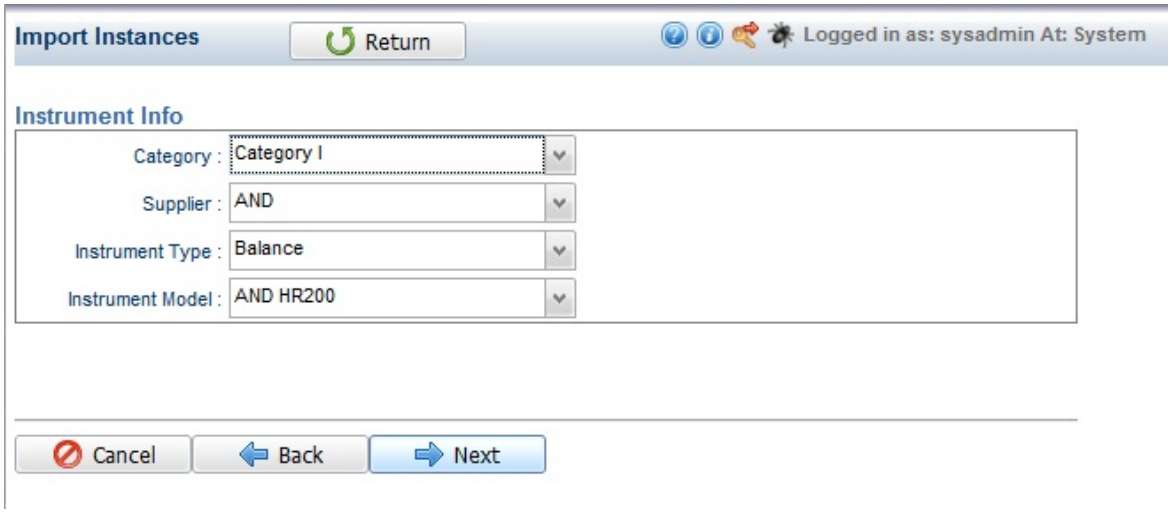


This will lead to the next step where you can choose the category, supplier, instrument type

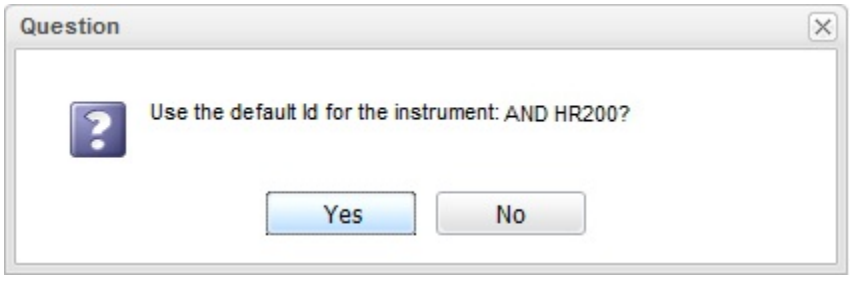
and instrument model. Notice that the supplier, type and model fields will be disabled until you have chosen the category.



Choosing the category will populate the other fields:



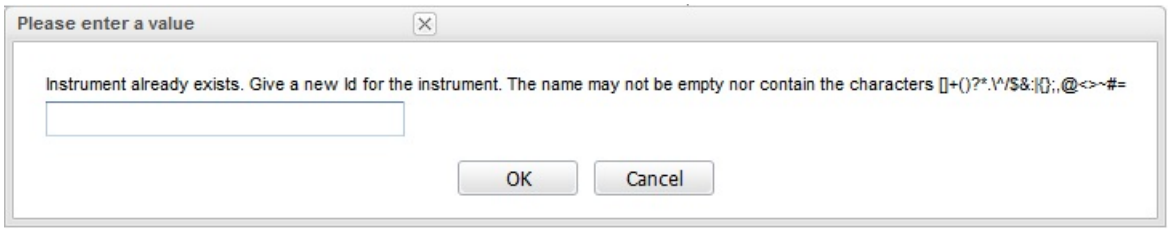
Once you have chosen everything, click Next. A prompt will open up asking if you want to use the default id for the instrument:



If you want to use the default id, click the Yes button. In case you want to define a new Id,

click the No button and a dialog will open whrer you can give the new Id.

If the given Id already exists, you will be asked to enter a new Id:

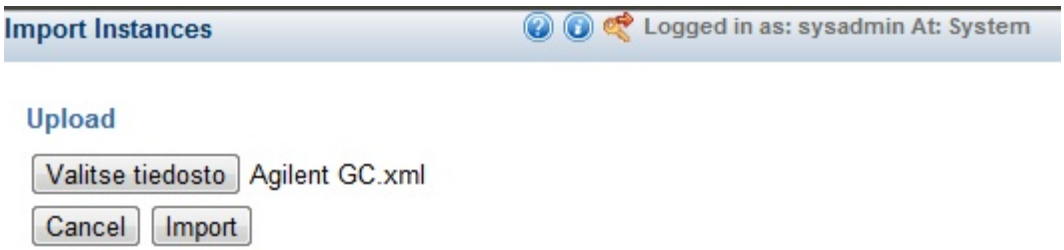


Fill in the new Id and click OK.

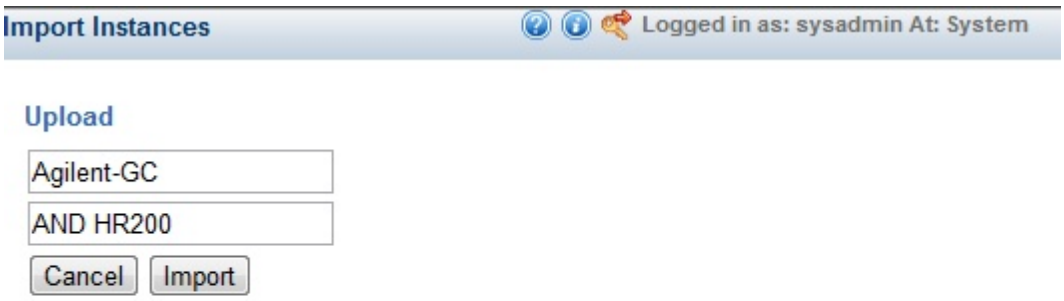
After you have chosen a unique id for the imported instrument, the instrument instance will be created and you will be redirected to the front page, where you should see the new instrument.

4.2.2.2 Importing from file system

After clicking the "File System" -button, a page will be opened where you can define the Instance ID of the imported instrument, and browse the local file system for the instrument definition xml-file. The dialog looks slightly different in different browsers.

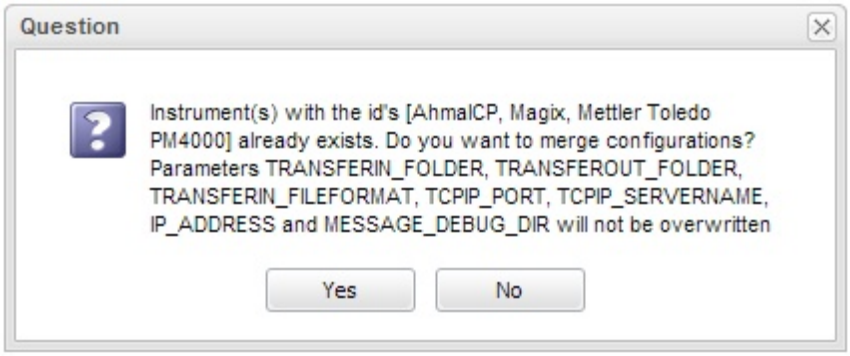


Press "Import" to import the instrument. Then you will be taken to a page where you can rename all instances contained in the XML-file.



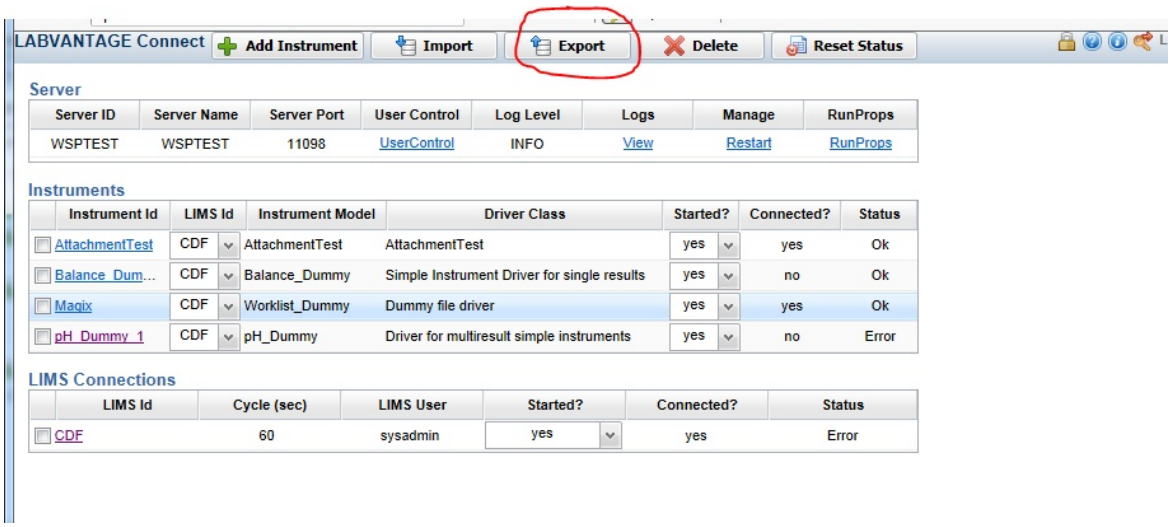
When you press Import, the instrument instances will be created and you will be redirected to the front page, where you should see the new instruments.

If instrument with same name (id) already exists, there is a possibility to merge. This means that exported configuration will not overwrite certain parameters. These parameters are: TRANSFERIN_FOLDER, TRANSFEROUT_FOLDER, TRANSFERIN_FILEFORMAT, TCPIP_PORT, TCPIP_SERVERNAME, IP_ADDRESS and MSG_DEBUG_DIR.
This functionality is useful for example when importing configurations from test environment to production.

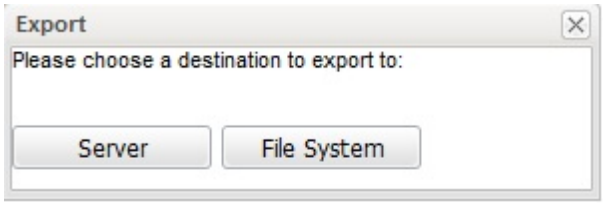


4.2.3 Exporting instrument configurations

To export instances, simply select the instances you wish to export, and click the "*Export*" button.



You will be asked if you wish to export the instances to the local file system, or to the server:



Notice that when exporting to the file system, all selected instruments will be exported to the same file.

For the export to server to function properly, the instance has to have it's instrument note defined, as this will decide the path where the file will be saved on the server. The instrument name decides the name of the exported configuration. The client will warn if the attempted export would overwrite an existing export.

After clicking the export to file system button, the configuration is sent to the browser as a file.

It is not possible to export Lims connections.

4.3 LIMS connections

LIMS connection defines a connection between the Connect and LABVANTAGE.

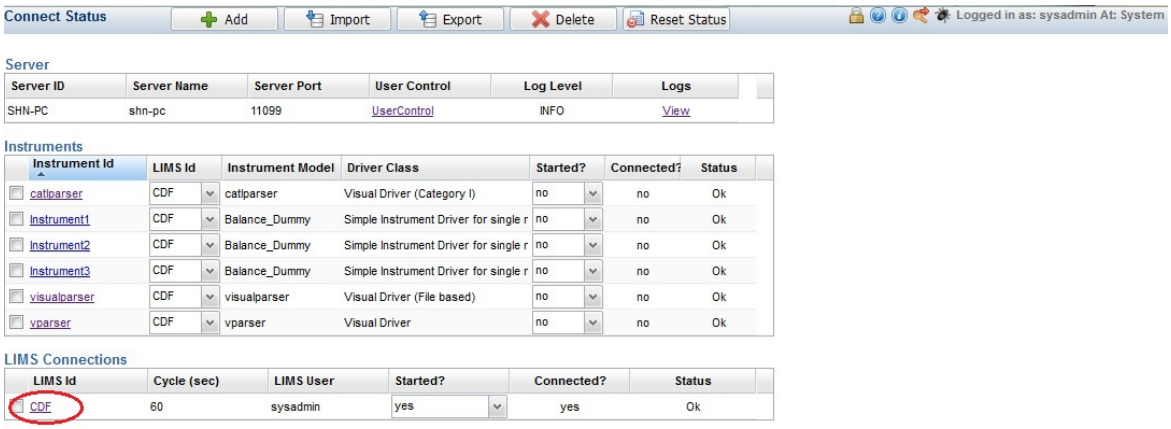
4.3.1 Creating a new LIMS connection

You cannot create new LIMS connection manually, you need to import them from the example LIMS connections provided with the Connect installation.

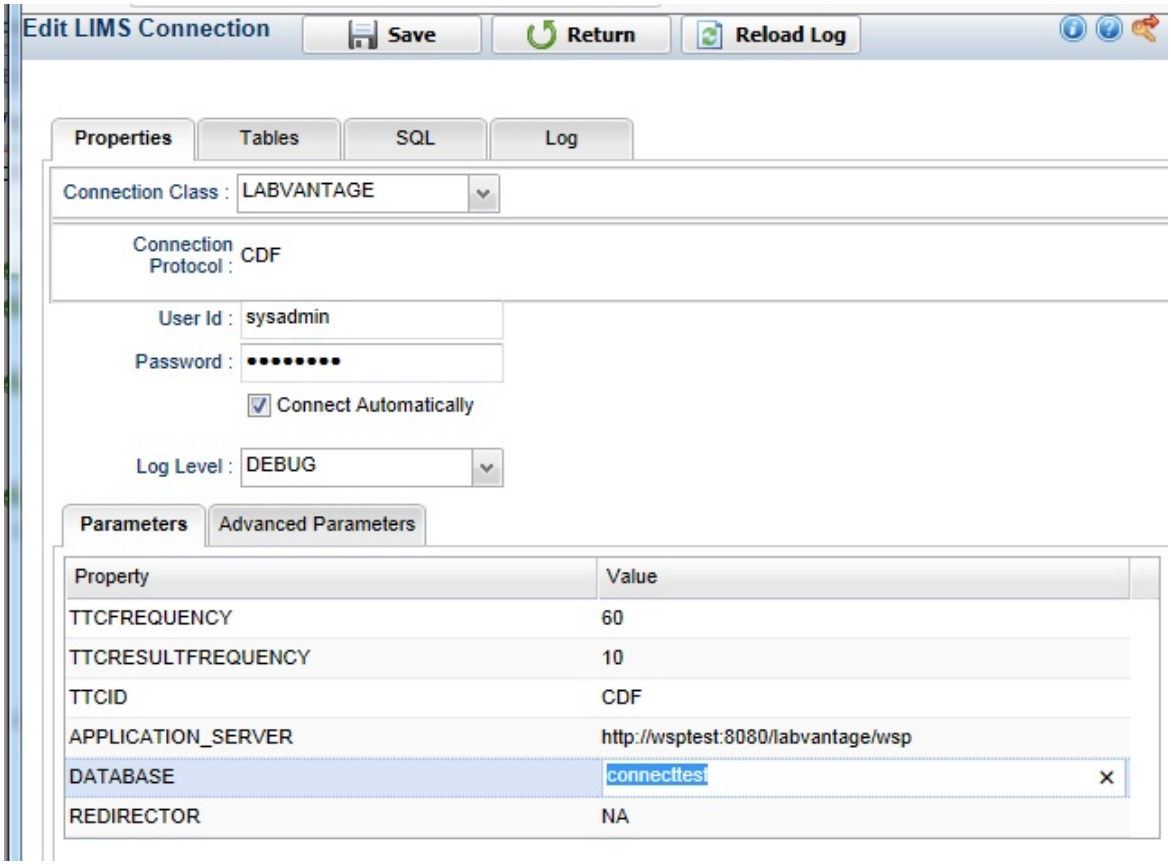
See the chapter Importing a LIMS connection.

4.3.2 Modifying LIMS connections

To modify an existing LIMS connection, click on its id on the main page.



This will open up the edit page for the selected LIMS connection, allowing you to change its properties.



While editing the properties, there are two important things to remember:
1) You cannot leave a field empty while saving
2) You need to fill in the User ID and Password to successfully save the changes.

Once you have done the changes you wanted to do, click Save.

4.3.3 Updating LIMS connection

When updating a LIMS Connection into an existing Connect installation, it is not enough to just import the new instance. The LIMS Connection instance also holds the definitions for the Connect internal database and the existing database structure cannot necessarily be automatically upgraded to support the table structure in the imported LIMS Connection.

Basically the upgrade requires deleting the old temporary database, which enables Connect to create a new empty database and then recreate the appropriate table structure in the new temporary database. This will naturally mean losing all data in the temporary database.

To upgrade the LIMS Connection, follow these step-by-step instructions:
1) Import the new LIMS Connection, overwriting the old one.
2) Set the appropriate properties for the imported LIMS Connection, such as the server name and database. (See Modifying LIMS connections.)
3) Stop the Connect server.
4) Remove the old database by deleting the database/wspconnectdb folder under the main

installation folder.
5) Start the Connect server.

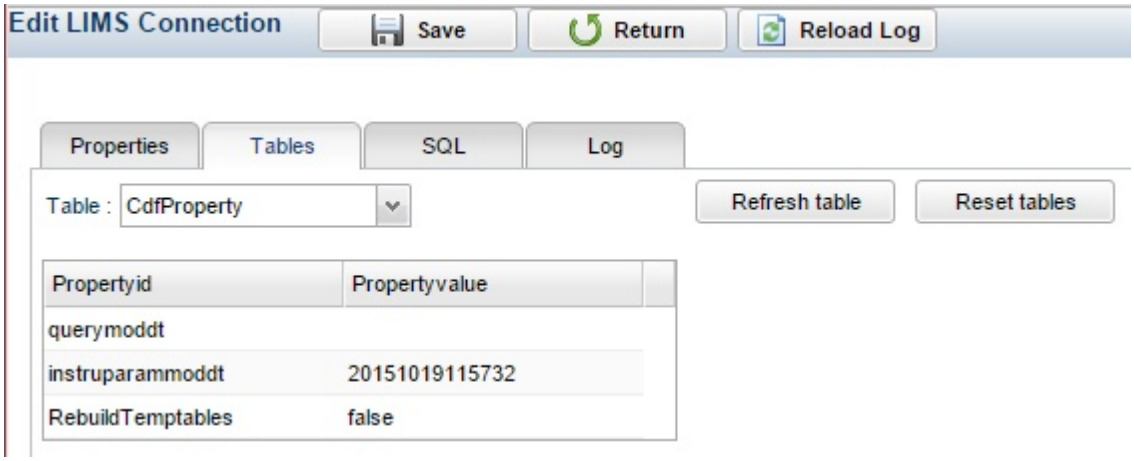
The new LIMS Connection is now functional.

4.3.4 Temporary tables

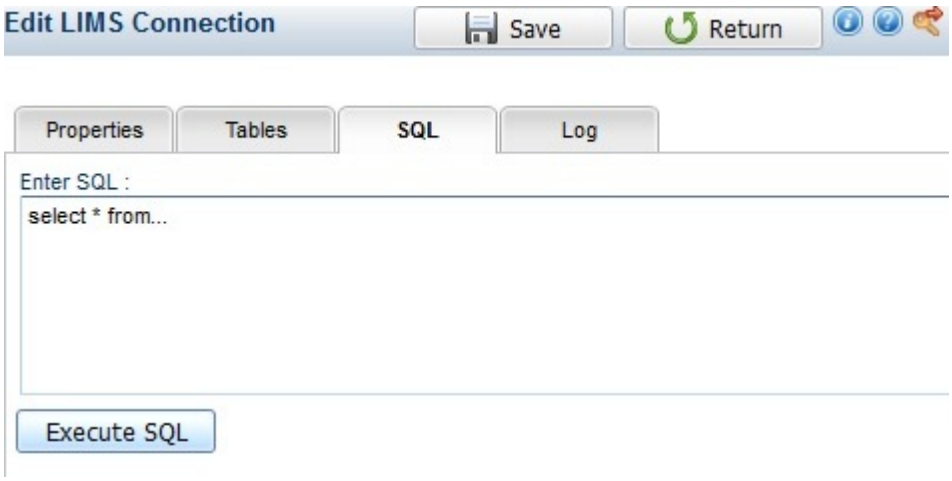
Connect configurations are stored in a lightweight temporary database. In addition of the configuration data, also request and result data are temporarily stored here, to be later processed by the CDFUpdater instance that transfers the data between Connect and LABVANTAGE LIMS.

Temporary tables for the CDF are automatically created upon importing the LIMS connection for the CDF and normally you should not modify them. However, in special cases it might be necessary to add new tables or new columns for the tables to create instances for very special instruments or systems.

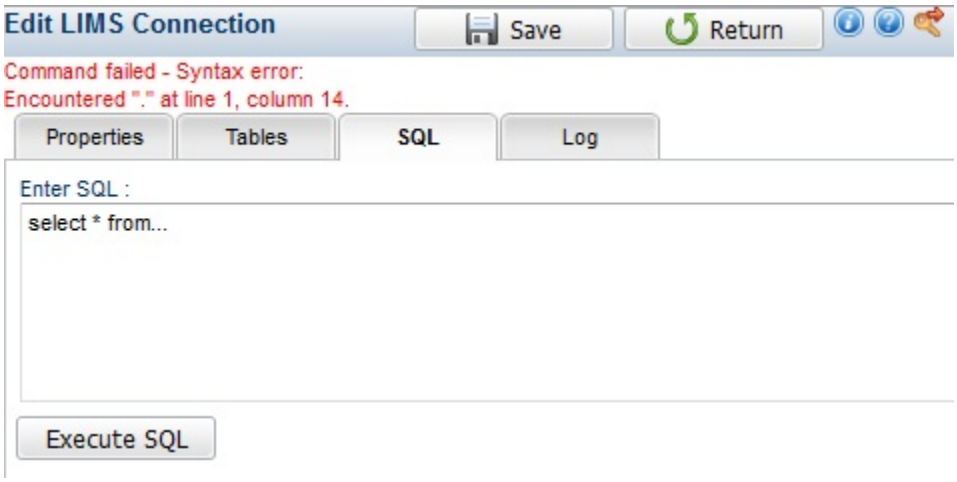
To view these tables, click on the Tables tab on the LIMS connection maintenance form.



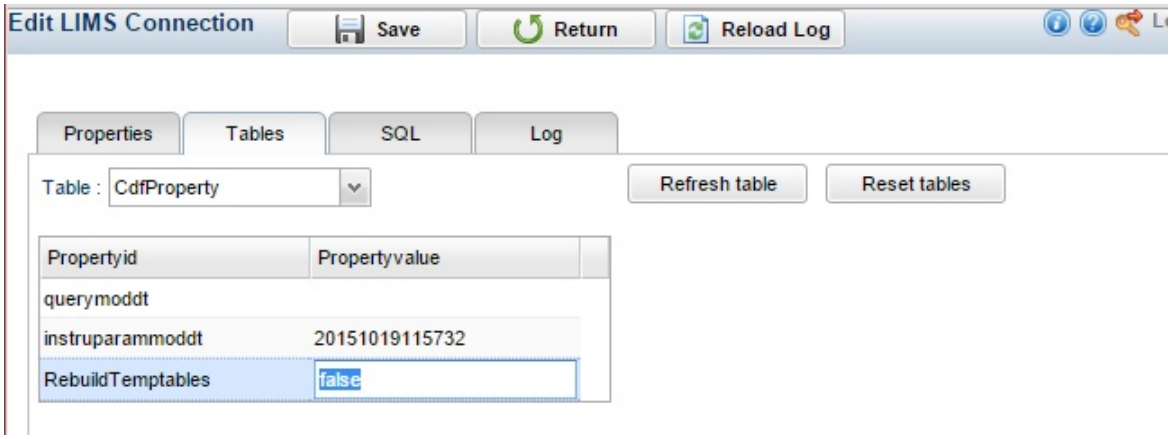
To modify these tables or to do custom queries, you can do SQL queries directly from the web-interface by choosing the SQL tab:



A notification will inform you of the success of the query. The query above would have resulted in the following error:

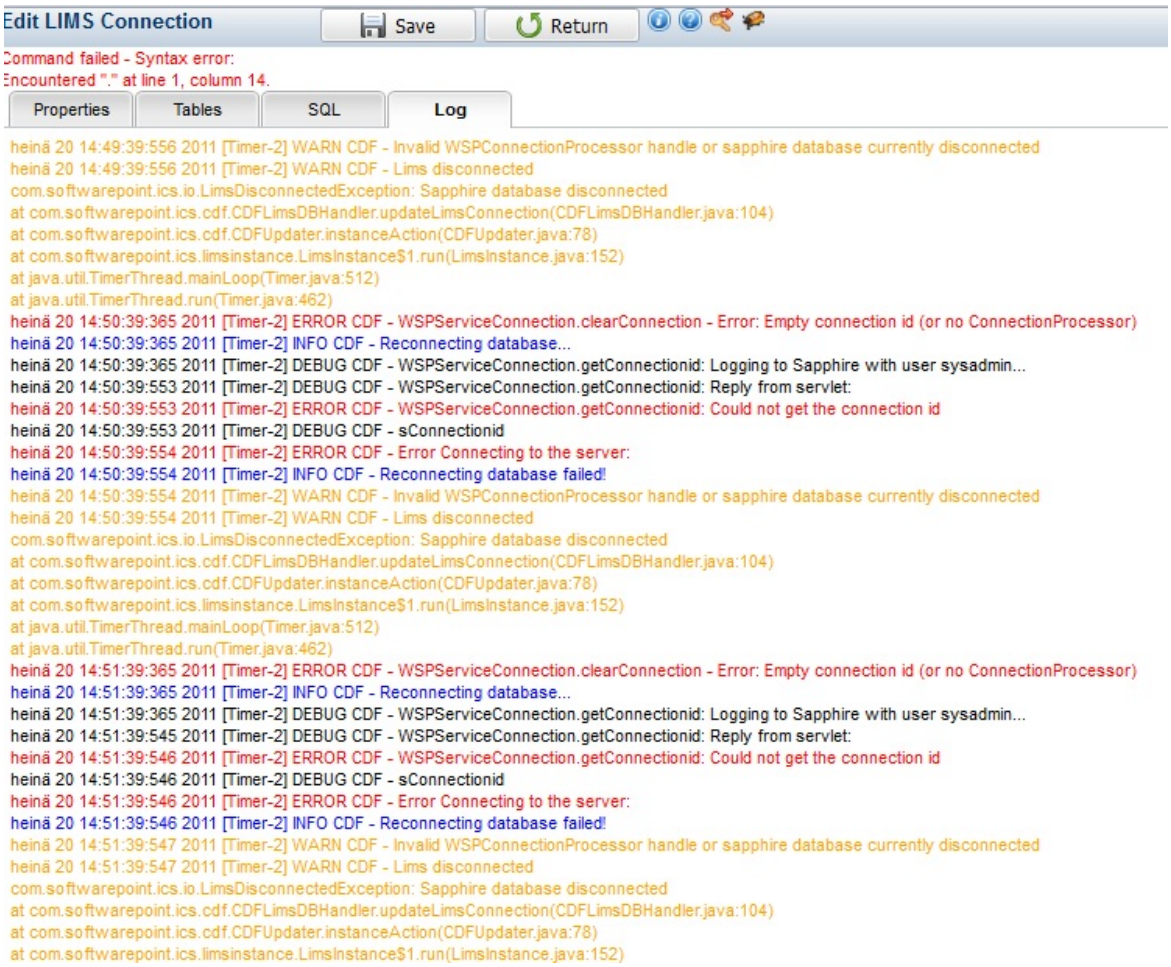


You can also edit the values in the tables directly from the interface. Once you've finished updating a field it is automatically updated in the database:



4.3.5 LIMS log

The LIMS log can be accessed via the LIMS connection maintenance form by clicking the Log tab. The log provides useful information in case of errors.

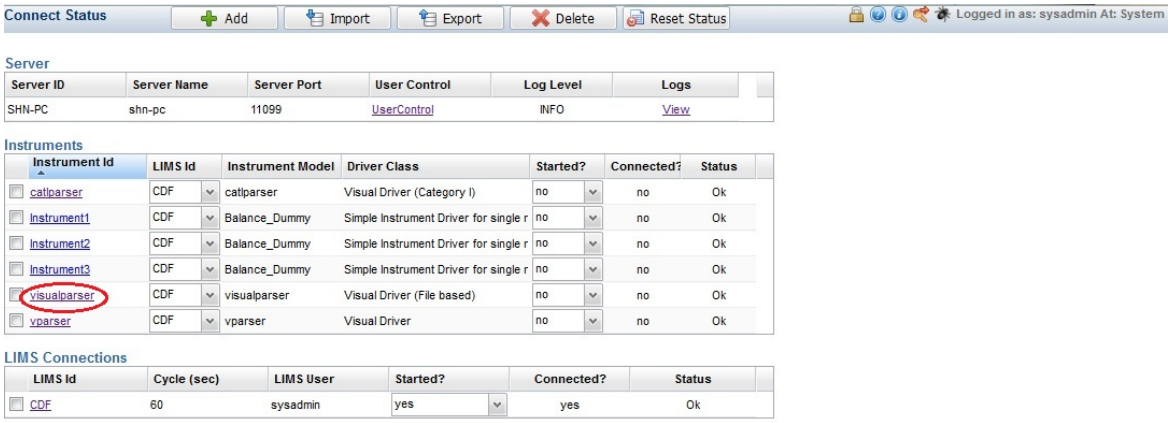


4.4 Instruments

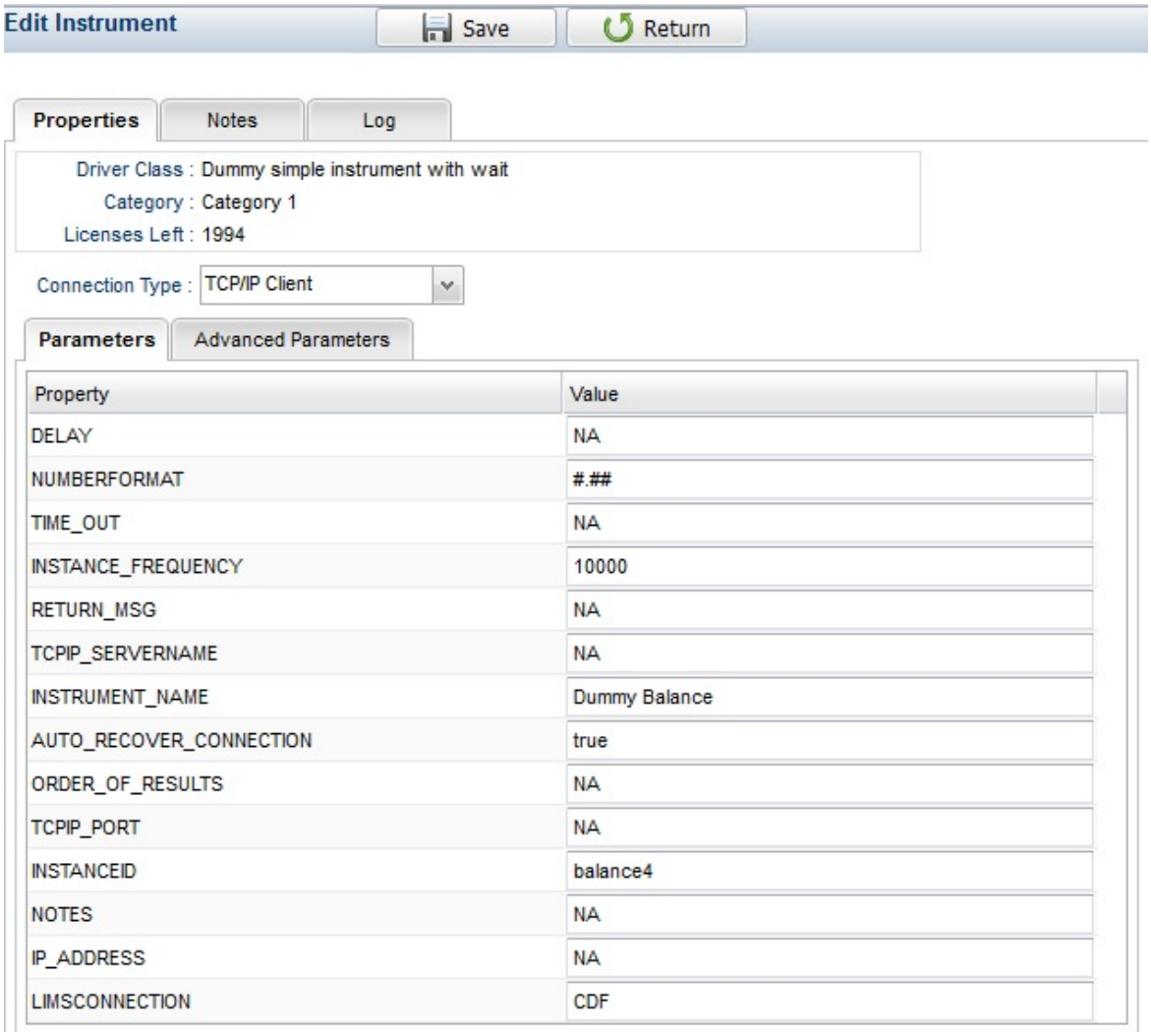
Instruments contains the definitions needed to connect to actual instruments. Created or imported instruments will be saved to the Connect's temporary database.

4.4.1 Modifying instruments

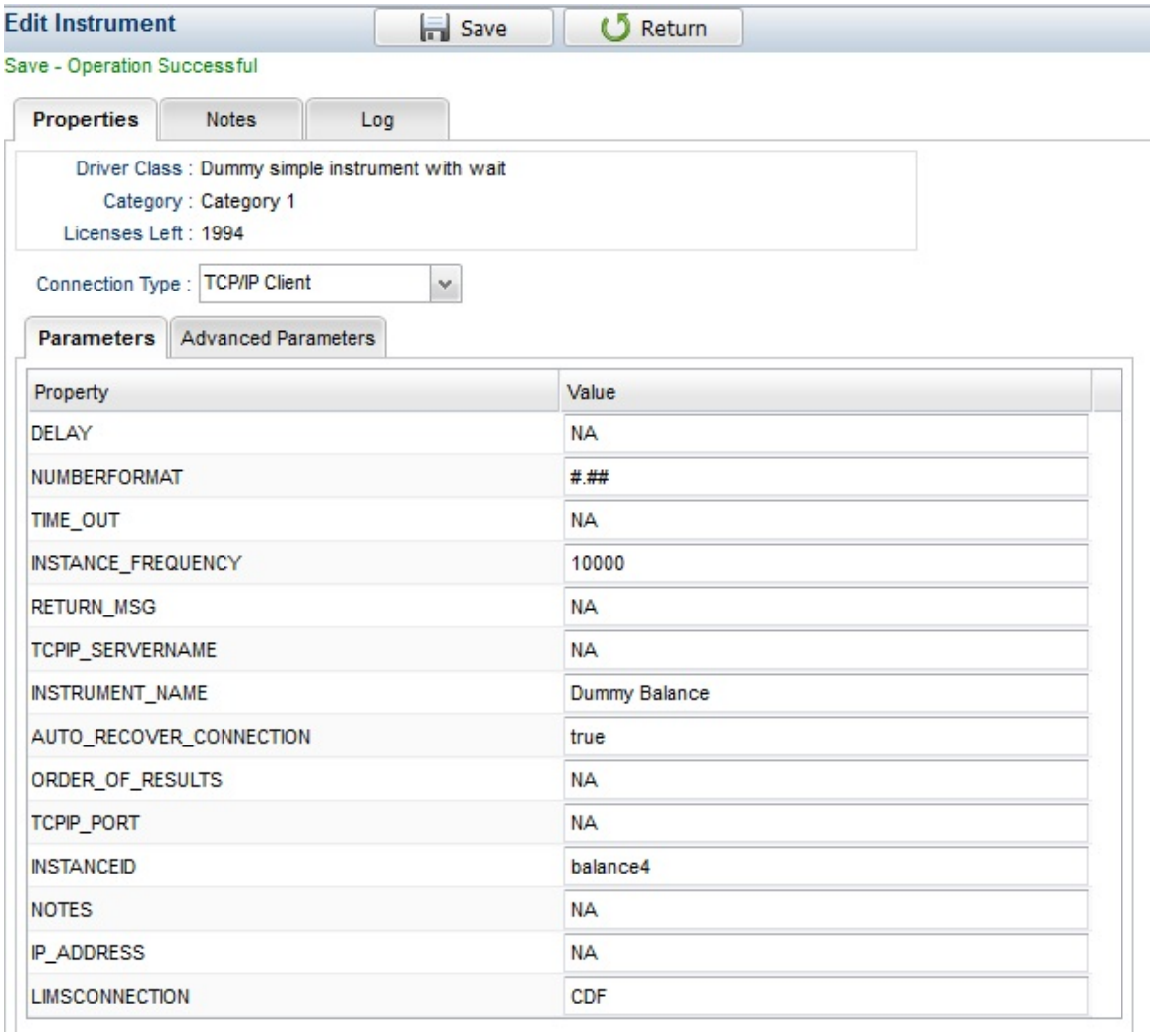
To modify the instruments, first click on the instrument ID to get to the edit page:



You can select the Connection type of the instrument, as well as edit the general and connection properties of it:



Once you have made the changes you wanted to, press save to save the changes:



4.4.2 Drivers

Several drivers for Category I, II, III and IV instruments exists in the Connect instrument library.

To import an instrument from the library, see the chapter Importing_instrument_instances. For imported instruments, you don't need to define the instrument driver - it is already defined in the imported configuration.

A driver can handle several different types of instruments. By modifying the parameters of the instrument instance, different behaviour can be achieved with the same driver.

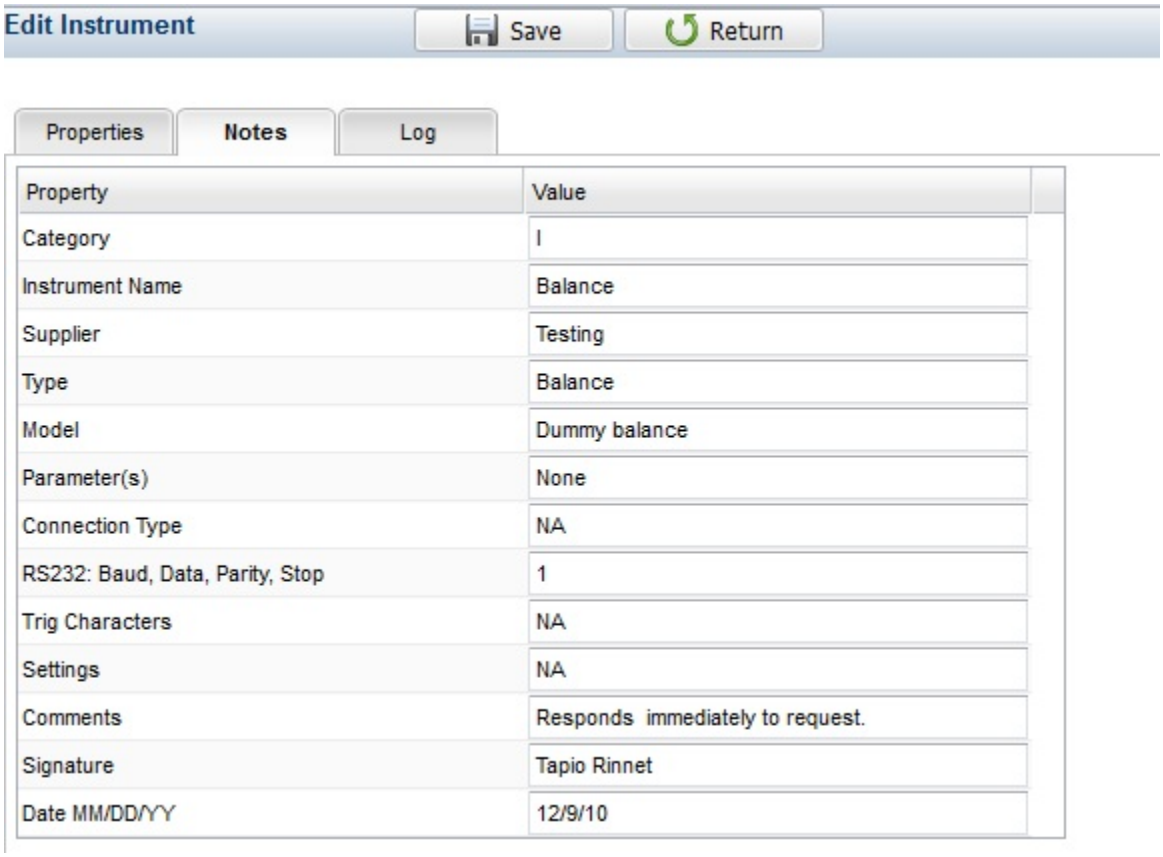
Drivers exists both for batch related instruments and query mode instruments.

4.4.3 Instrument note

Instrument note defines the settings that should be used for an instrument. It can be created, viewed and edited from the Driver Configuration dialog, but it is saved as a plain text file, and can also be created and edited manually.

The INSTRUMENT_NAME -property of the driver configuration specifies the name of the instrument note-file. The path of the saved file is deduced from the contents of the instrument note, and is *Category N/Supplier/Type*. If modifying any of these, the save location of the instrument note will change.

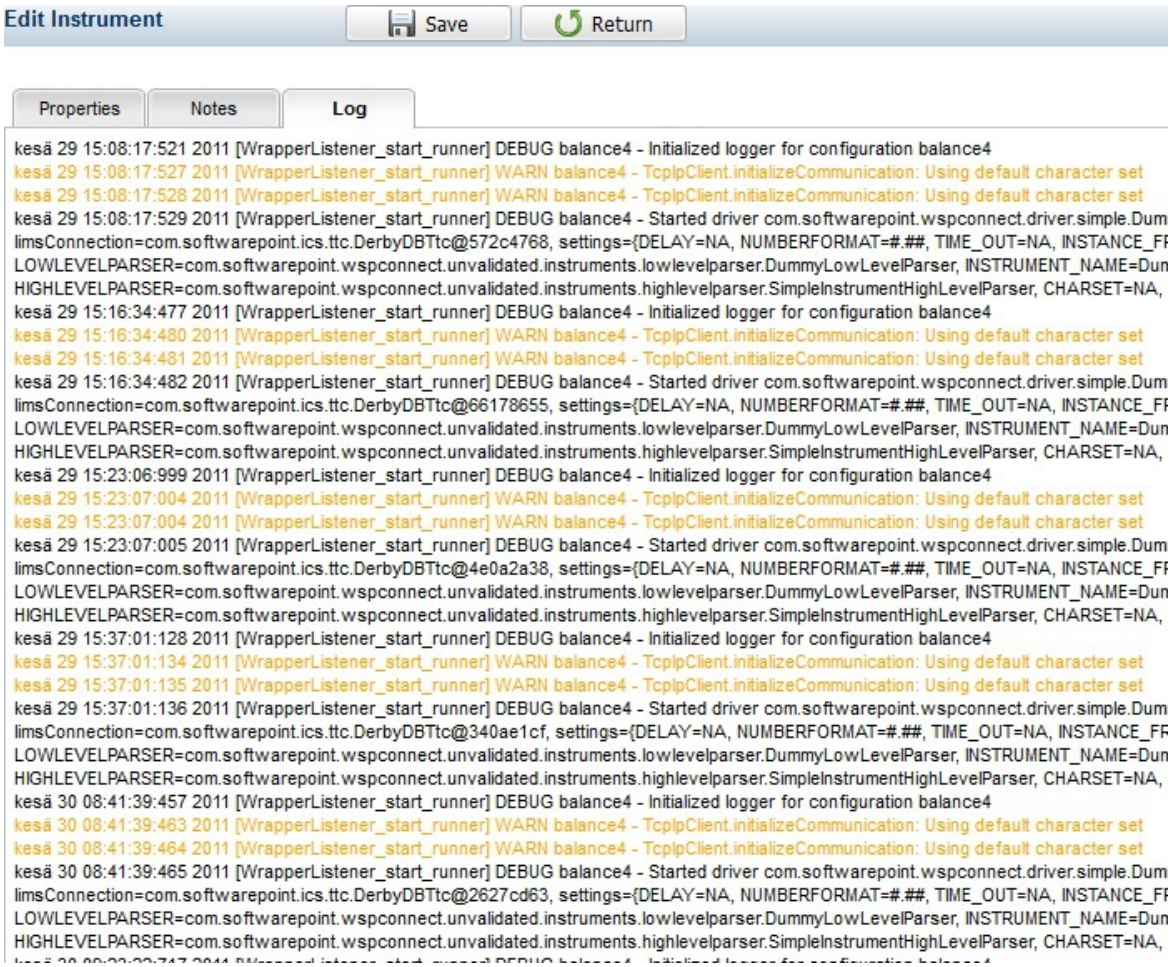
Example of an instrument note:



Notice!
The directory where the instrument notes will be saved can be modified by editing the property *wspconnect.export.directory* in the *run.properties* -file in the Connect server -installation. The property is commented out by default, and points to the Instruments-directory.

4.4.4 Instrument log

The Instrument log can be accessed via the instrument maintenance form by clicking the Log tab. The log provides useful information in case of errors.

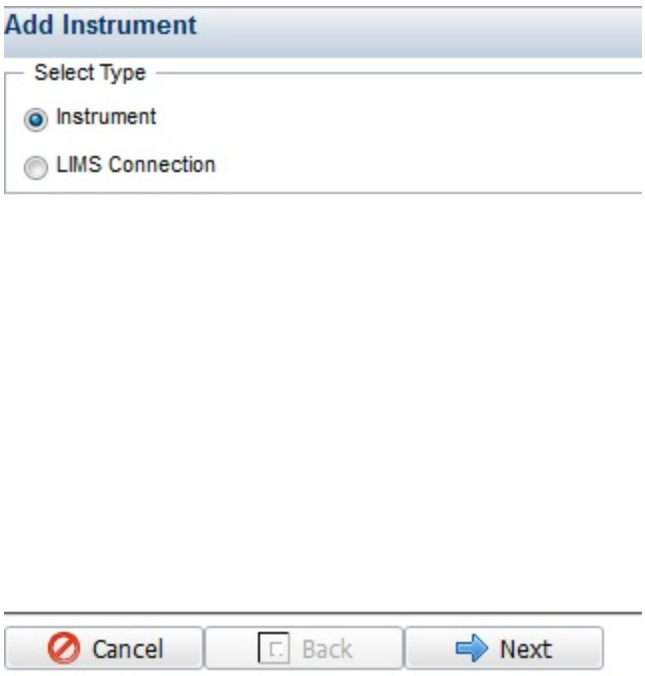


4.4.5 Adding instruments

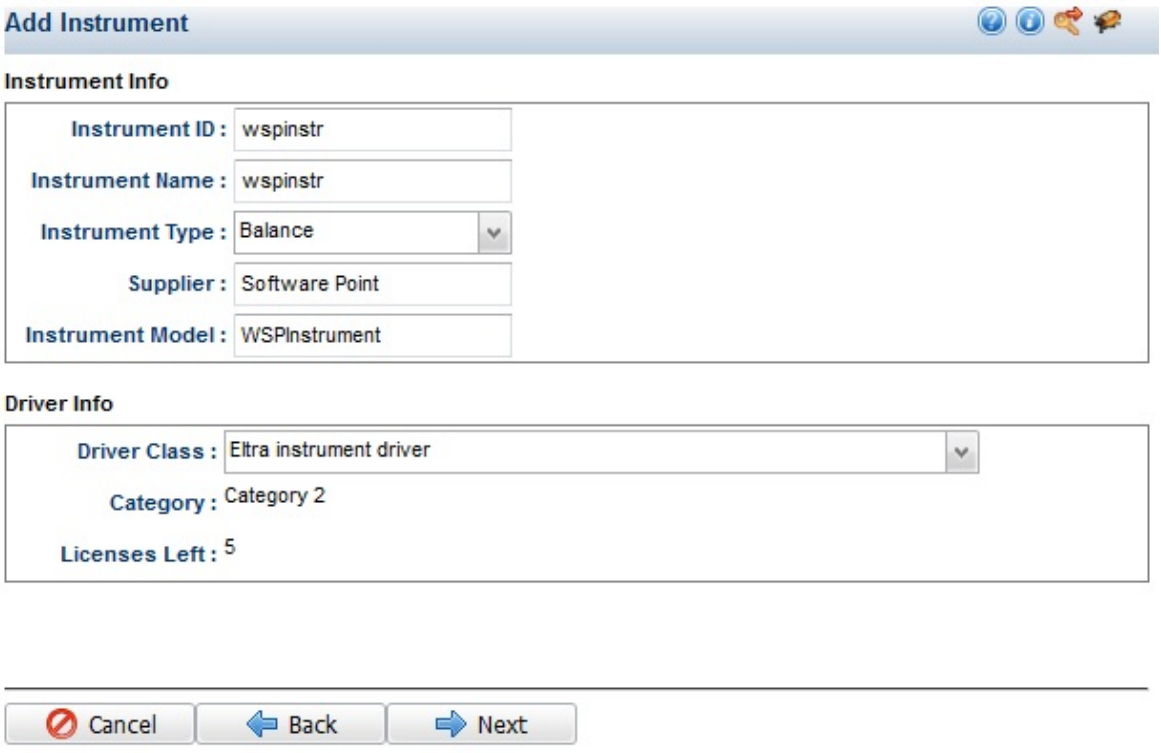
Generally when adding a new instrument it is most convenient to use the Import functionality (see the chapter Importing instrument instances).

Some instruments may not have a default configuration available. In that case the you need to create a new instrument manually. To do this, press "Add" on the front page.

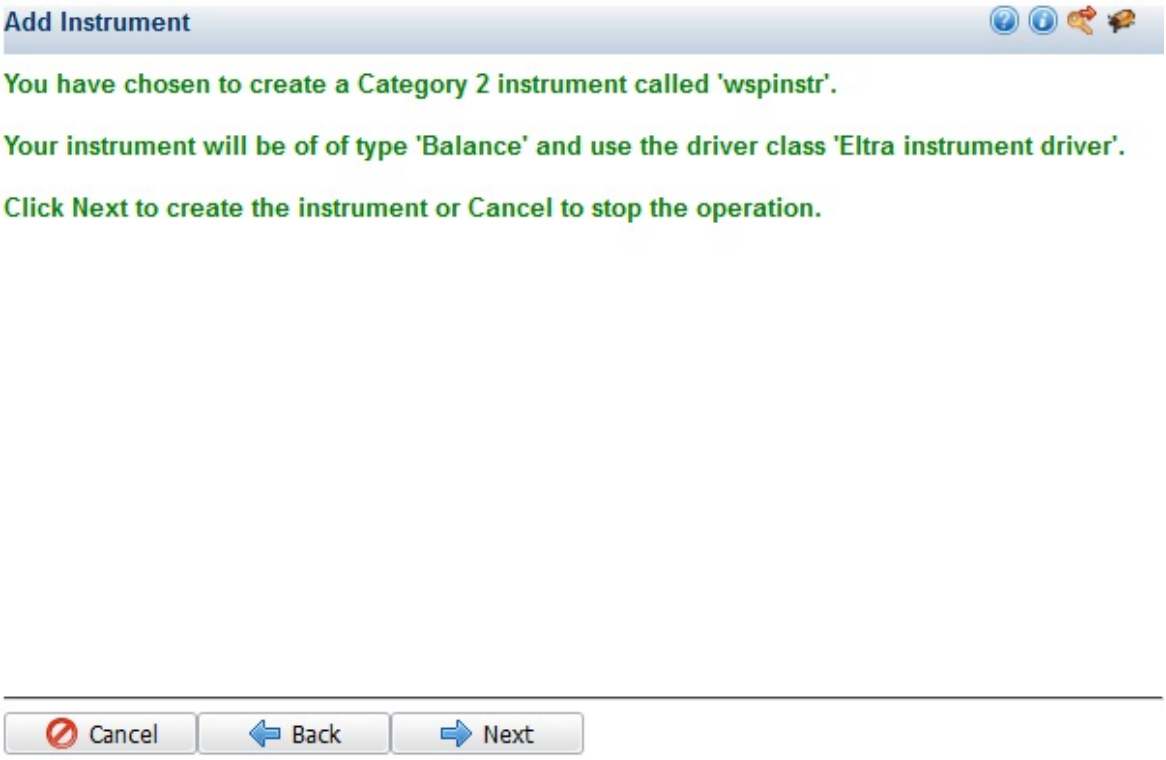
This will open up a new window for adding instruments. Note that it is not possible to add LIMS connections at this time:



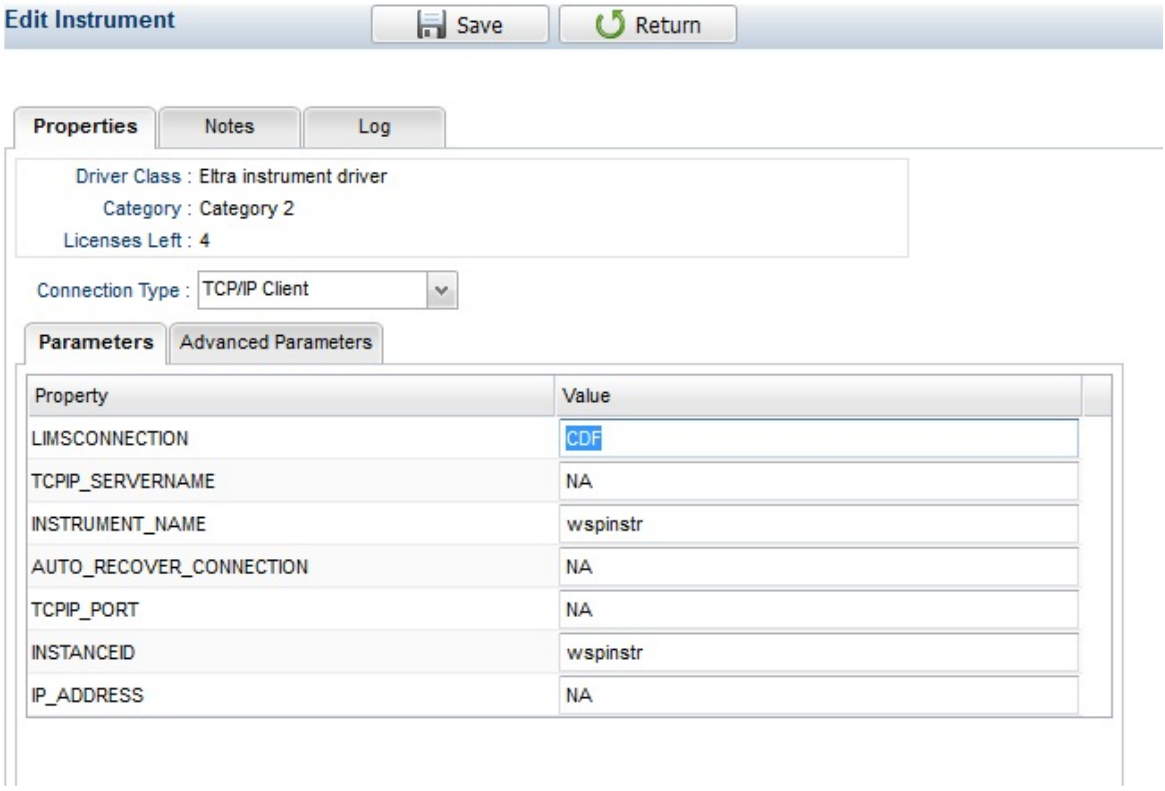
Clicking next will open up a window where you can enter the details for your instruments. Note that all the fields need to be filled in order to proceed:



Pressing next will take you to a confirmation window with a summary of the instrument you are about to create:



Pressing next will create the instrument and redirect you to the Edit page of the instrument. In this page you can modify properties of the instrument:



4.4.6 Instrument status

When viewing the front-page you can see the current Status of the different instruments. There are 3 different statuses:

Status
Ok

The Ok status indicates that everything is alright.

Status
Error

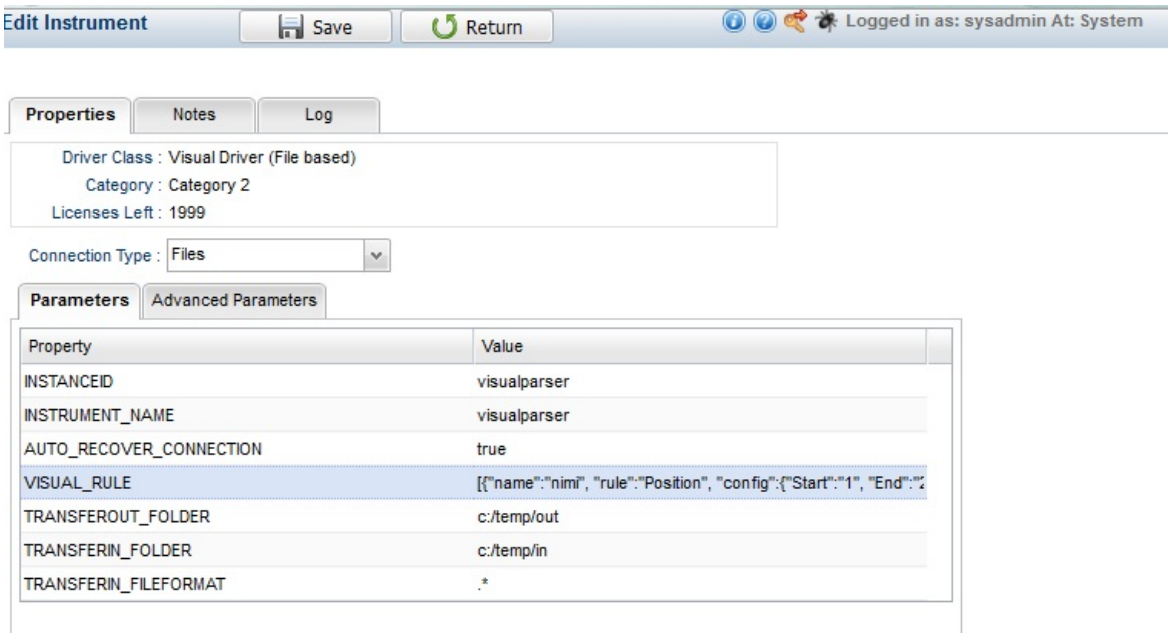
The Error status indicates that there is a problem with the configuration. Please refer to the log of the instrument and adjust the settings appropriately.

Status
Invalid

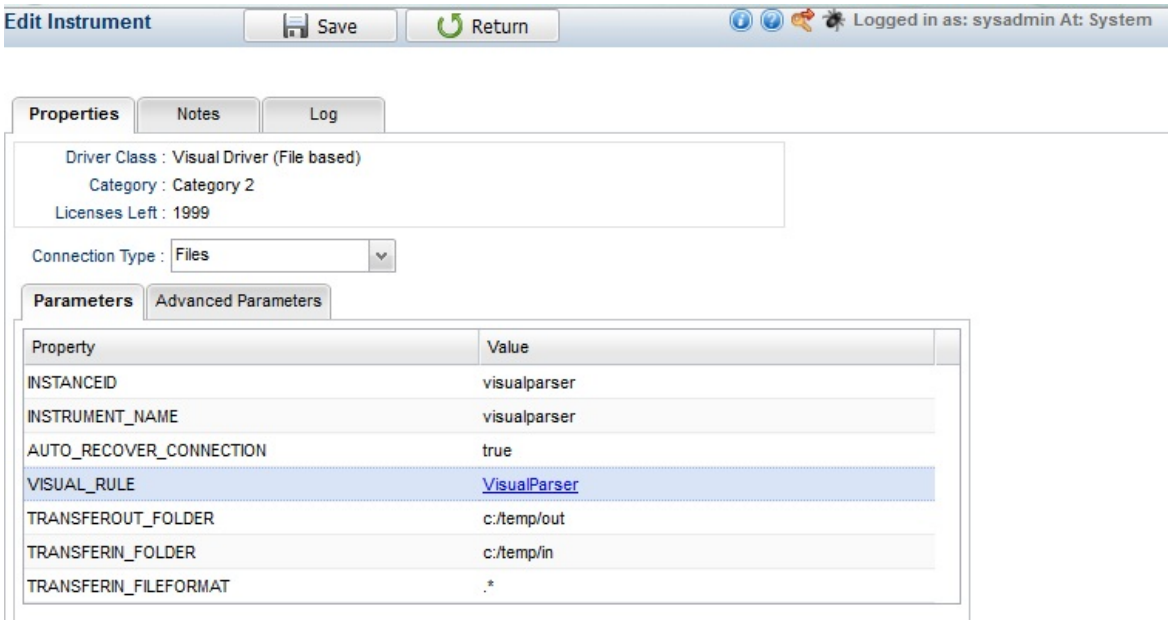
The Invalid status indicates that there is a problem with the driver of the instrument. The driver might be missing from the installation.

4.4.7 Visual Parser

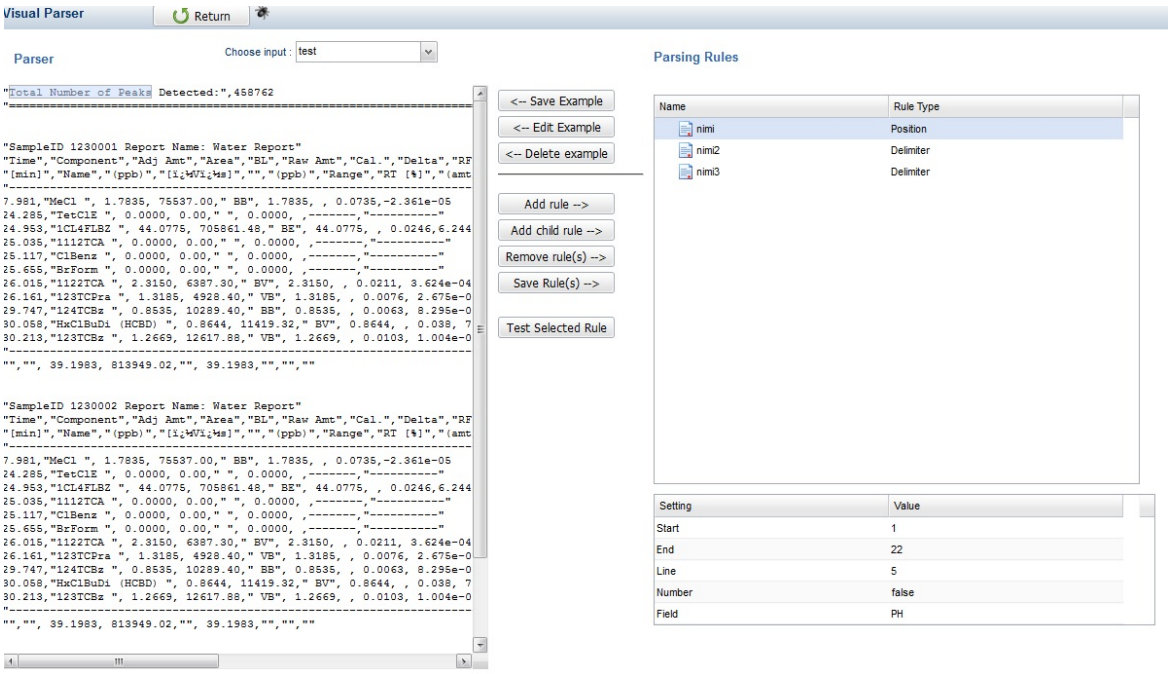
Certain instruments allow the users to directly define the parsing rules for the instrument output. If an instrument has a Driver Class containing Visual Driver, the user can enter a Visual Parser interface from the instrument editing view:



Clicking the Visual Rule field for editing shows a link to the visual parser:



Clicking the link will open up a new page with the Visual Parser:



Currently Visual Parser is available for Category I -instruments using the driver "Visual Driver (Category I)", and File-based Category II -instruments using the driver "Visual Driver (File based)".

Please refer to the following guides for learning How to use visual parser is explained in following chapters.

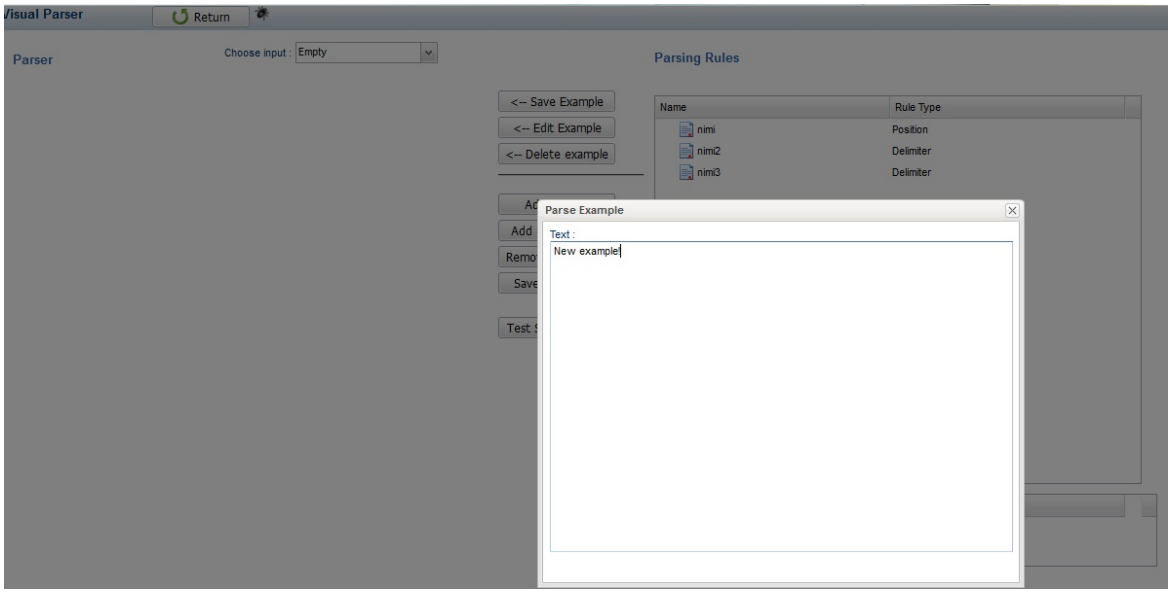
4.4.7.1 Managing Examples

Creating rules without an actual sample of the output that should be parsed is difficult, therefore the Visual Parser allows the user to enter his/her own example. This simplifies the creation of rules and allows the visualization of the rules on the example to further simplify the process.

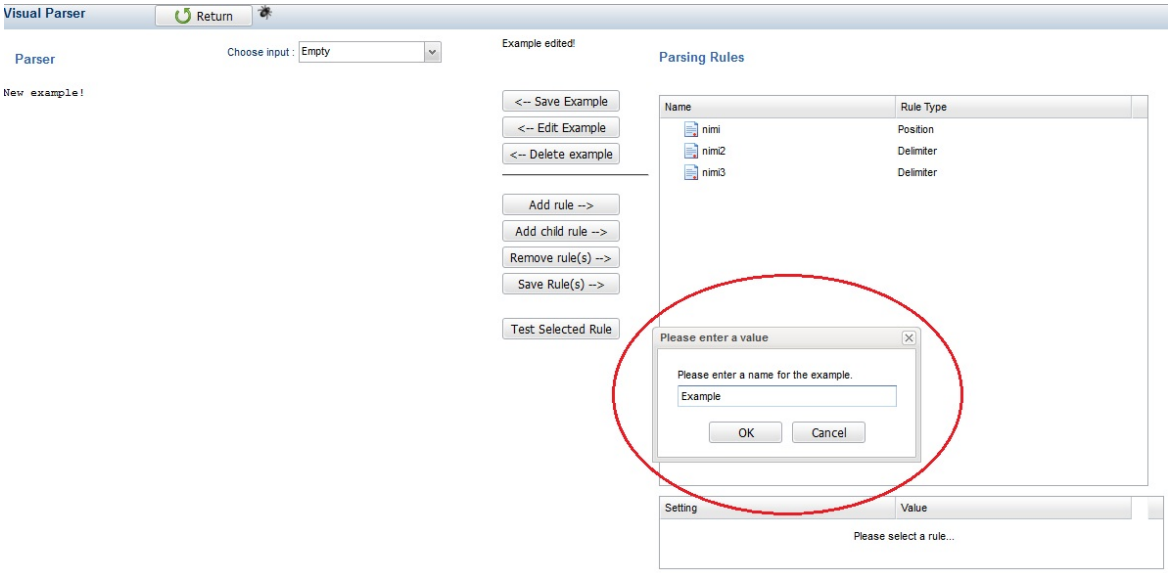
The possible actions for the examples are:

- Edit example - Allows the user to edit an example or create a new one if "Empty" is selected.
- Save example - Saves the current example with an unique identifier.
- Delete example - Deletes the currently selected example.

Example of the Editing view:



Example of the saving dialogue:



4.4.7.2 Managing Rules

The parsing rules are the rules that are used to parse the instrument output.

There are four different rule types available:
- Delimiter - A rule that uses specified strings to limit what is read.
- Positional - A rule that uses positions in the output to determine what to read.
- Tokenize - A rule that uses a token to separate the output (can be regex or e.g. a comma).
- Groovy - A rule that uses Groovy script to parse the instrument output.

Rules can also have child rules. This requires, however, that the parent rule has the field value "sectioner".

Delimiter rules have the following properties:
- Start_token - Starts parsing when this token is encountered
- Include_start - Whether the start token should be included in the result
- Stop_token - Stops parsing when this token is encountered.
- Include_stop - Whether the stop token should be included in the result.
- Accept_N - Can set the rule to only accept some occurrences of the matches (e.g. 3 for the third match). The values are separated with a semicolon.
- Number - Force the rule to only parse numbers out of the result.
- Value - Can add variable references to the results.
- Field - Defines the type of the extracted parameter (e.g. sectioner, keyid, result...).

Positional rules have the following properties:
- Start - The start position
- End - The end position
- Line - The line-number to be parsed (note, this is the line number within the section, if this rule is a child-rule).
- Number - Force the rule to only parse numbers out of the result.
- Field - Defines the type of the extracted parameter (e.g. section, keyid, result...).

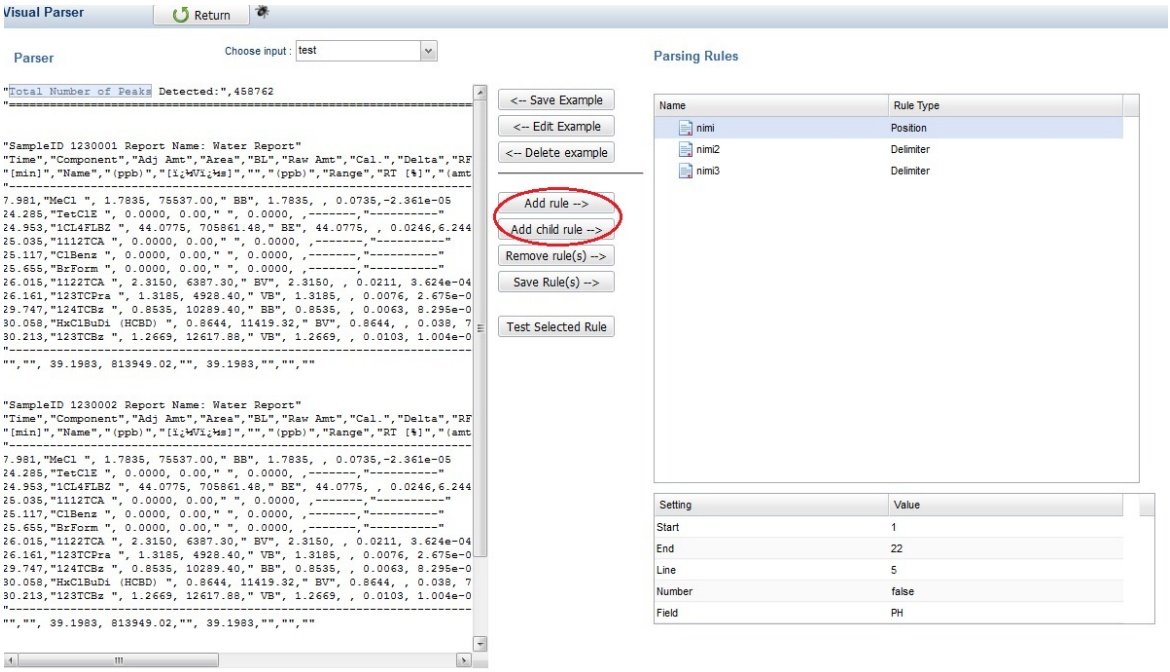
Tokenize rules have the following properties:

- Separator - Token that separates the different tokens
- Accept_N - Can set the rule to only accept some occurrences of the matches (e.g. 3 for the third match). The values are separated with a semicolon.
- Pattern - Check if the separator is a regex pattern.

Groovy rules have the following properties:

- Groovy - The Groovy script used for parsing the output.

You can add rules by pressing the Add rule button. You need to edit the name (default as "(name)") and rule type (default as "Groovy") after the new rule is created:



Please note that the rules need to have unique names on the same level.

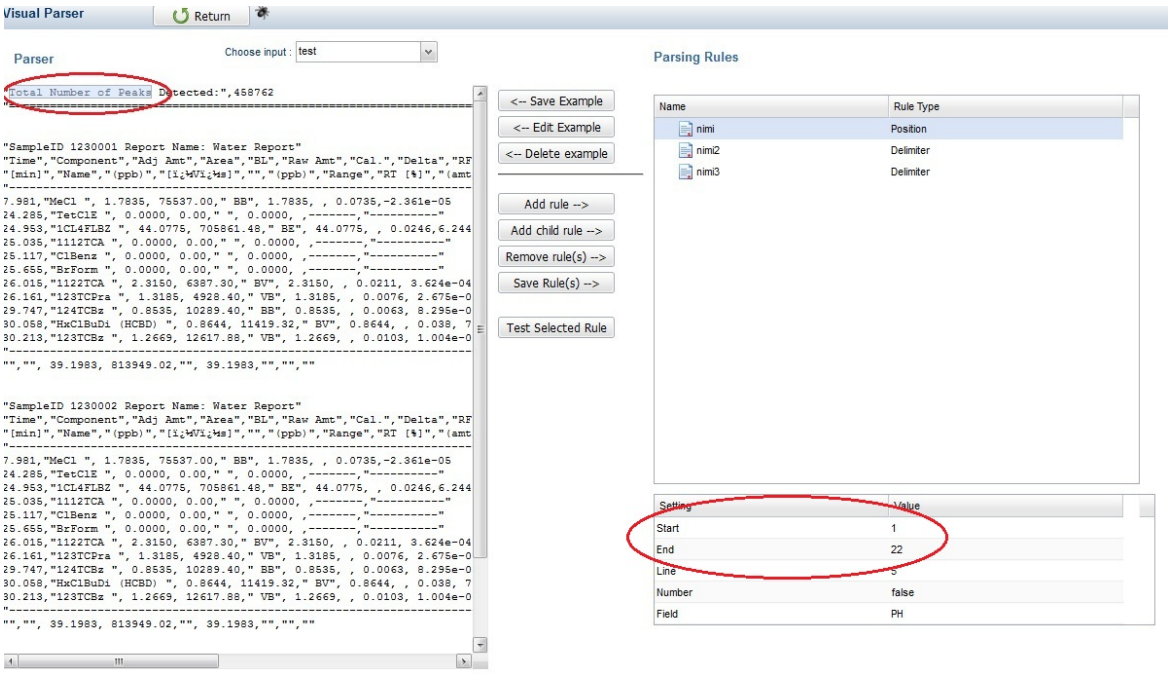
You can remove the selected rule (and its children) by clicking "Remove Rule".

You can save the rules by clicking "Save Rules".

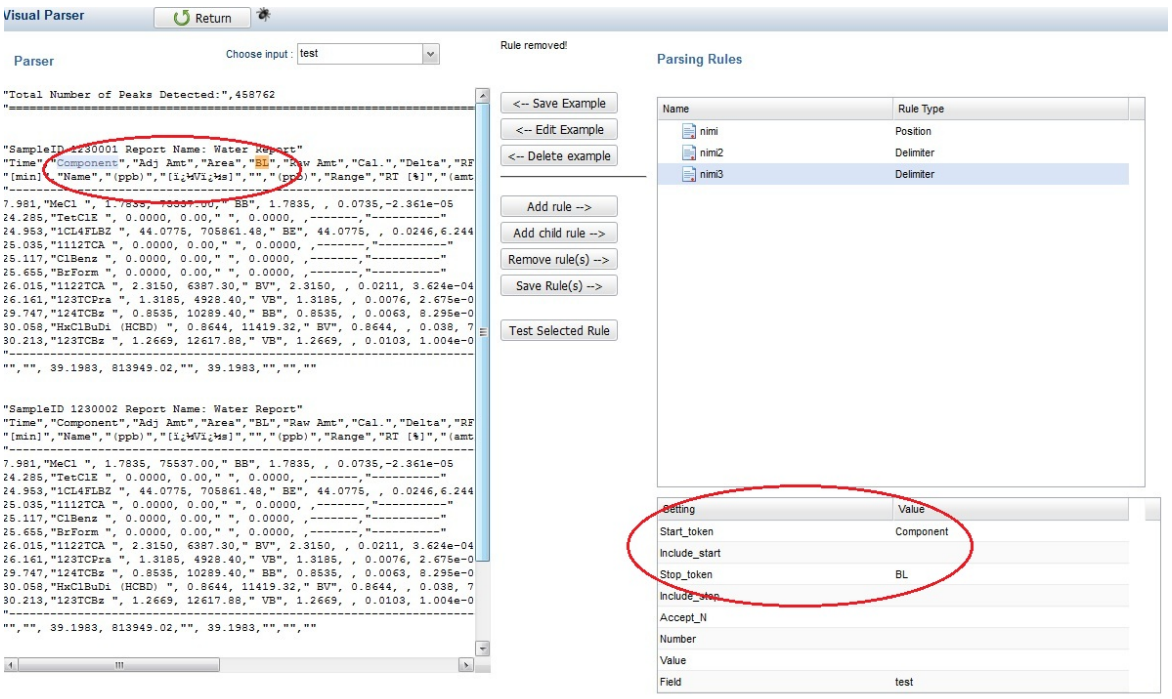
If a Delimiter or Positional rule have the appropriate settings set, the part of the example that would be parsed is highlighted with a box (or boxes). Please note that the Line parameter is not taken into account in the positional rule and that the sections are not taken into account in both Rule Types when visualizing the parsing.

These boxes may also be dragged and resized to determine the start and stop positions or tokens.

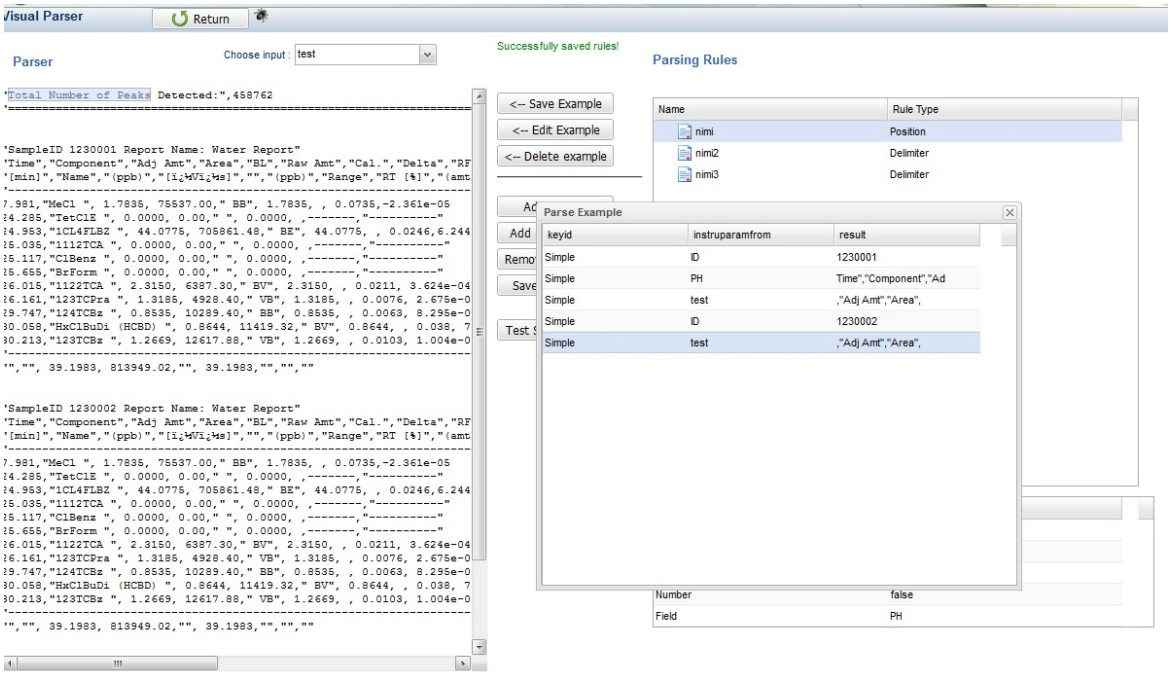
Here is an example of the Positional rule's visualization:



Here is an example of the Delimiter rule's visualization:



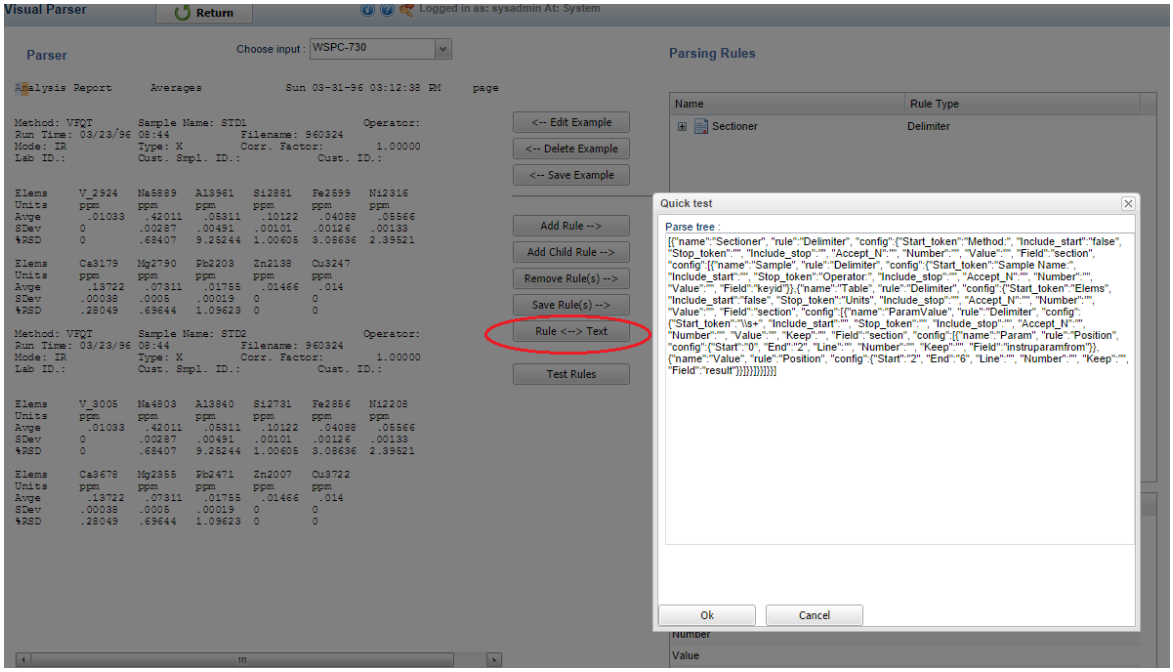
In order to determine the validity of the Rules you have created it is useful to test them against the example. This happens by pressing the "Test Selected Rule" button. The output after pressing the button looks like the following:



4.4.7.3 Copying Rule to Text Editor

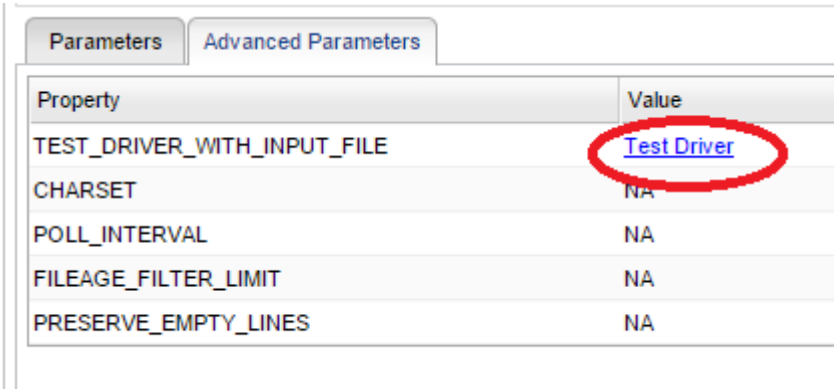
It is possible to copy rule to clipboard using "Rule <--> Text" button. Clicking "Rule <--> Text" opens text window were parse rule is presented as JSON object structure. Parse rule can be copied from and pasted to the window. After accepting and saving new parse rule it is possible to test it using "Test rules". This feature allows quickly to test different parse rules.

Parse tree JSON presentation below:

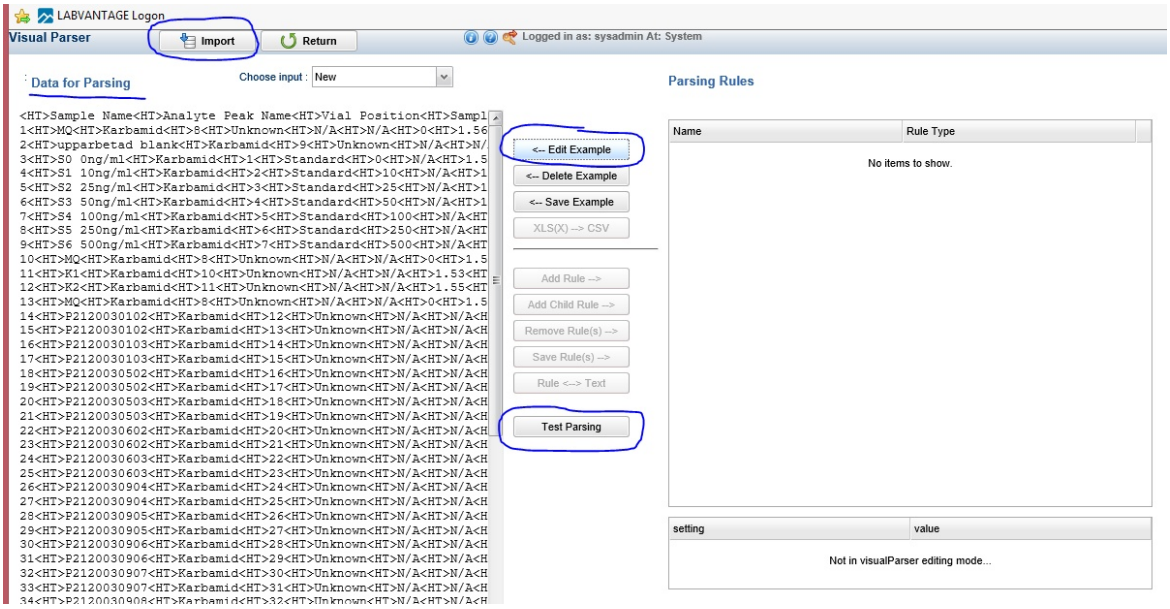


4.4.8 Test result file parsing

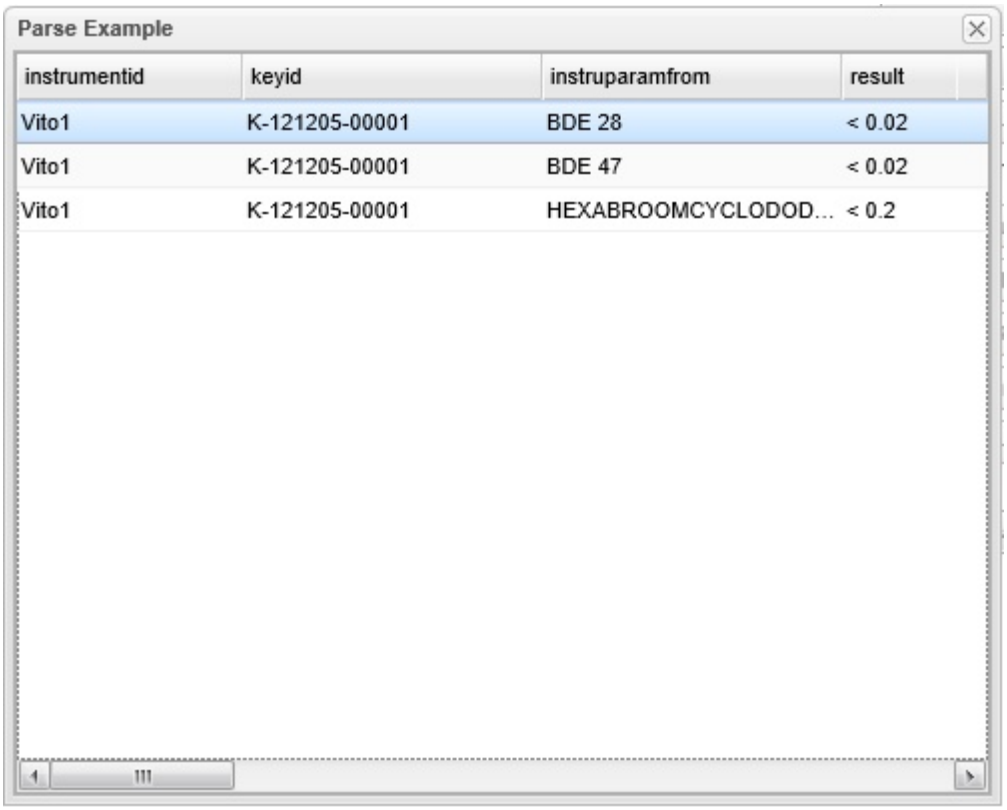
Connect offers a possibility to test outcome of the parsing of the instrument result files. This functionality can come in handy for example when you want to see how changing driver properties affects the outcome of the parsing. Or if the result file format has slightly changed and you want to see if driver still parses right things. Generally this functionality is available for category 2 and 3 drivers, with some exceptions. If driver has a TEST_DRIVER_WITH_INPUT_FILE property, test functionality can be used for driver in question.



Double clicking the 'Test Driver' link will open the parsing test page. You can choose the file you want to test by pressing Import button and choosing the file or by pressing the Edit example button and copying the the content of the result file to the opened pop-up. Text data files (.txt, .dat, etc.) can be edited and saved. Excel files (.xls and .xlsx) are marked blue and saving and editing is disabled. Data to be parsed is shown on 'Data for Parsing' panel.



Test parsing button will run the data through the driver code and show the outcome of the parsing. In the 'Parse Example' pop-up you can see what information is parse from result file (Keyid = Sample id in Lims, instruparamfrom = mapped to parameter in Lims, result, etc.)



In some rare cases testing parsing with this test page don't produce parsing result due to a special handling in driver code. It is good practice to first test with the file and driver configuration that is known to work, before trying with new result file or with new driver configuration.

4.5 System interfaces

Connect does not contain any general solution for system interfaces. Creating a system instance is in principle similar to creating an instrument instance, but generally the temporary table structure of the LIMS Connection has to be created according to the specific needs of the system interface.

5 Connect LABVANTAGE interface

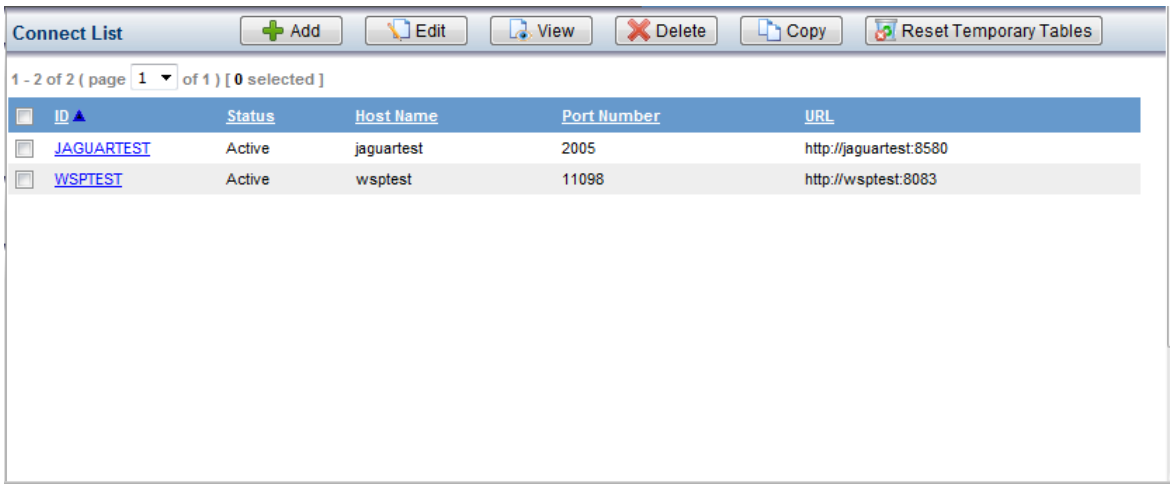
You need to make some definitions in LABVANTAGE before it can properly communicate with Connect server. All the definitions can be done through LABVANTAGE web pages, which are imported to LABVANTAGE upon installation (see the Connect Installation Guide).

All the Connect pages can be accessed from the Lab Admin sitemap as seen below:

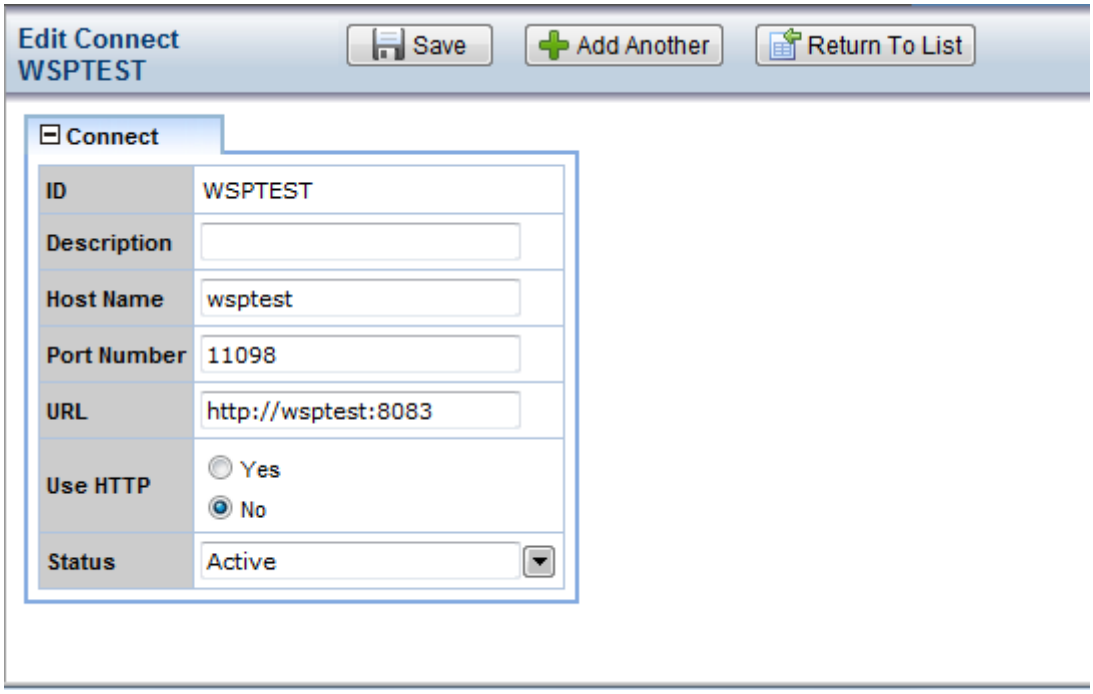
5.1 Connect definitions

Connect definitions are maintained in the Connect SDC. This SDC should contain all the individual Connect instances installed in your network that you want to communicate from LABVANTAGE.

One LABVANTAGE system can be connected to several Connect servers and one Connect can serve several LABVANTAGE systems.



You will define the Connect connection properties on the maintenance page as below:



For Category II and III instruments, the ID must match with the Connect ID of the Connect server. By default, Connect uses the host name in uppercase as its ID. The ID can also be defined in the *run.properties* file located on the Connect server installation folder.

The host name and port number are used for connection **from** LABVANTAGE to Connect server. The host name can be either the IP-address or the fully qualified DNS name of the server. The port must match the property wspconnect.cajo.port in the *run.properties* file, which is 11098 by default. If these values are left empty, LABVANTAGE will use 'localhost' and '11098' as the default values. These settings are not used for regular Category II or Category III instruments, as with them the connection is always from Connect **to** LABVANTAGE.

The URL is the address to the Connect web GUI. The default port is 8083, but you can modify this in the in the *jetty.xml* file located on the Connect server installation folder.

5.2 Instrument definitions

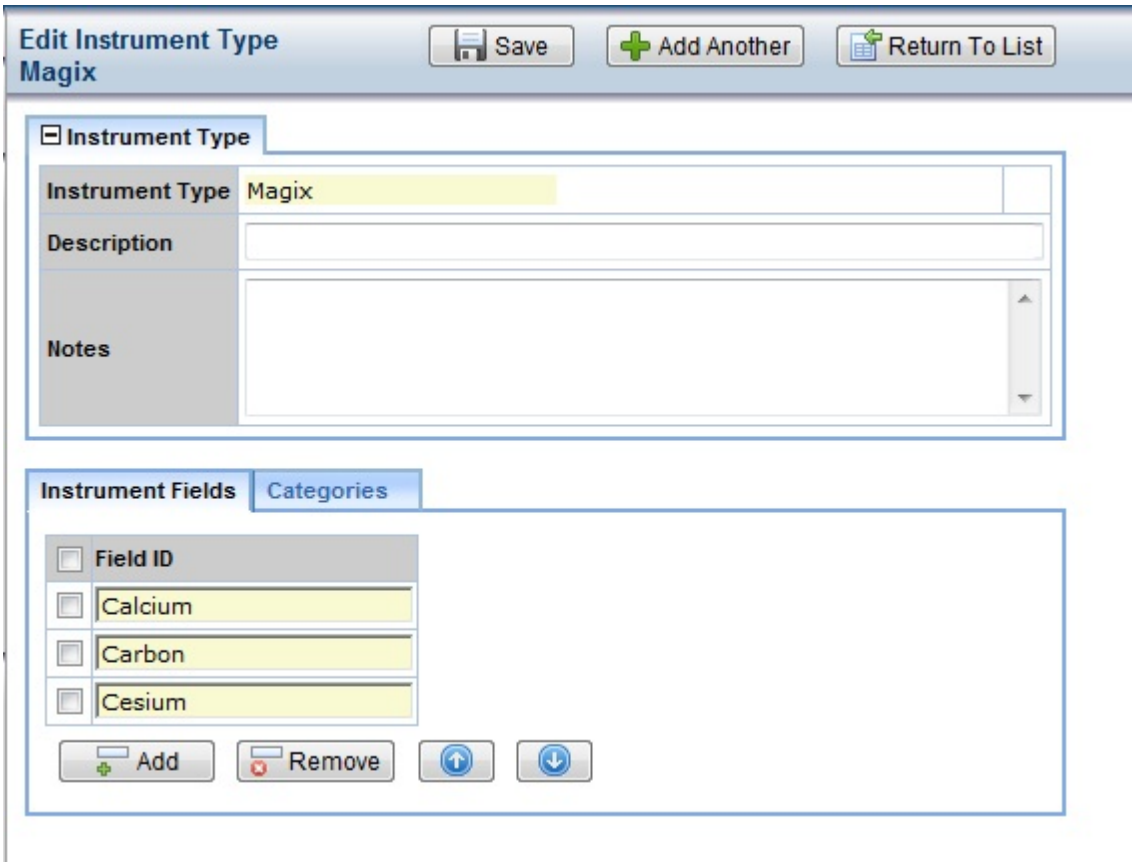
To be able to transfer result data from Connect to LABVANTAGE, you must create the appropriate definitions for the Instrument Type, Instrument Protocol Provider, Instrument Model and Instrument definitions.

This manual only shows one example, for details on the LABVANTAGE instrument model system, please refer to the LABVANTAGE LIMS documentation.

In addition to the instrument related definitions, you need to define the Parameter Lists (Parameter List) and Parameters that can use your Instrument Types or Instrument Models.

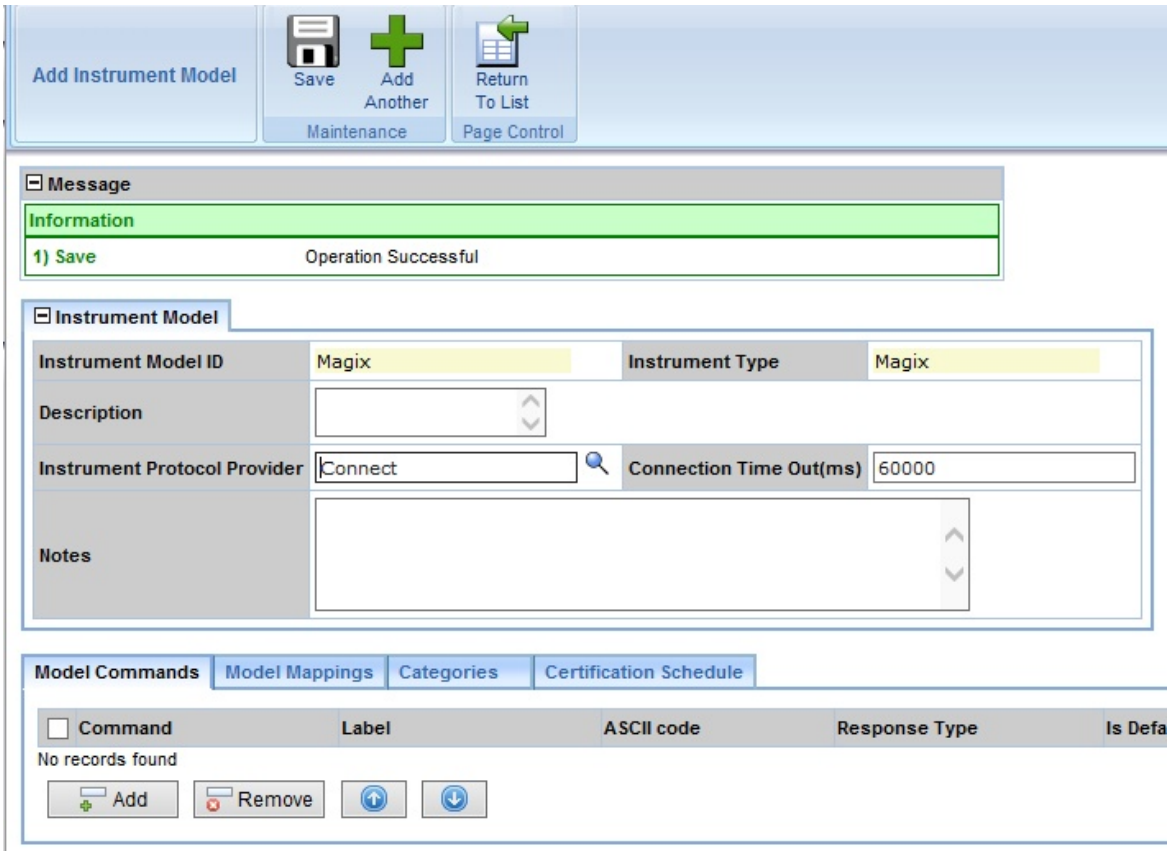
5.2.1 Instrument type

The instrument parameters are defined in LABVANTAGE in the Instrument Types. An instrument type has a number of instrument fields, which by default correspond to the parameter id coming back from instrument.

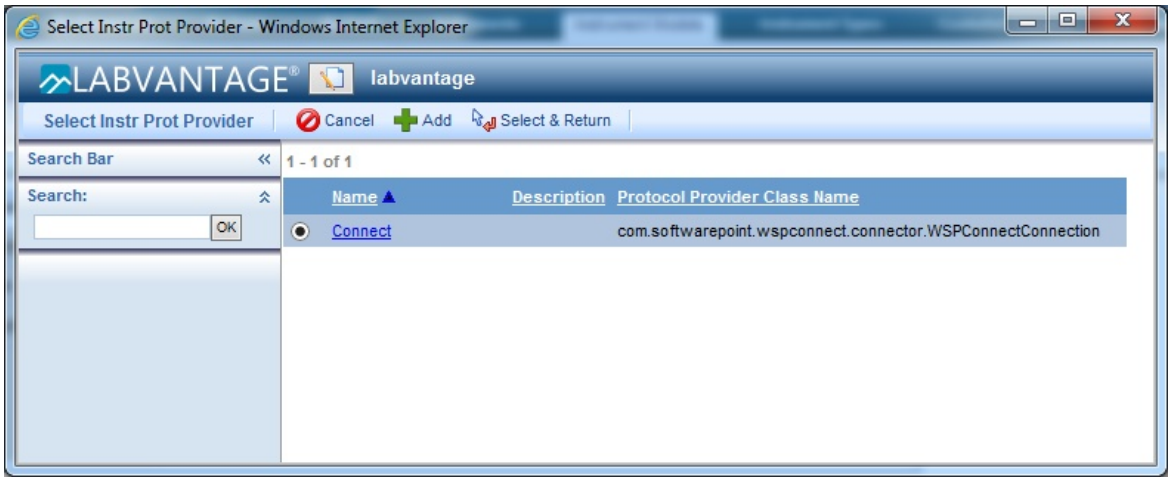


5.2.2 Instrument model

For category II and III instruments you will only need to link the Instrument Model to a specific Instrument Type.



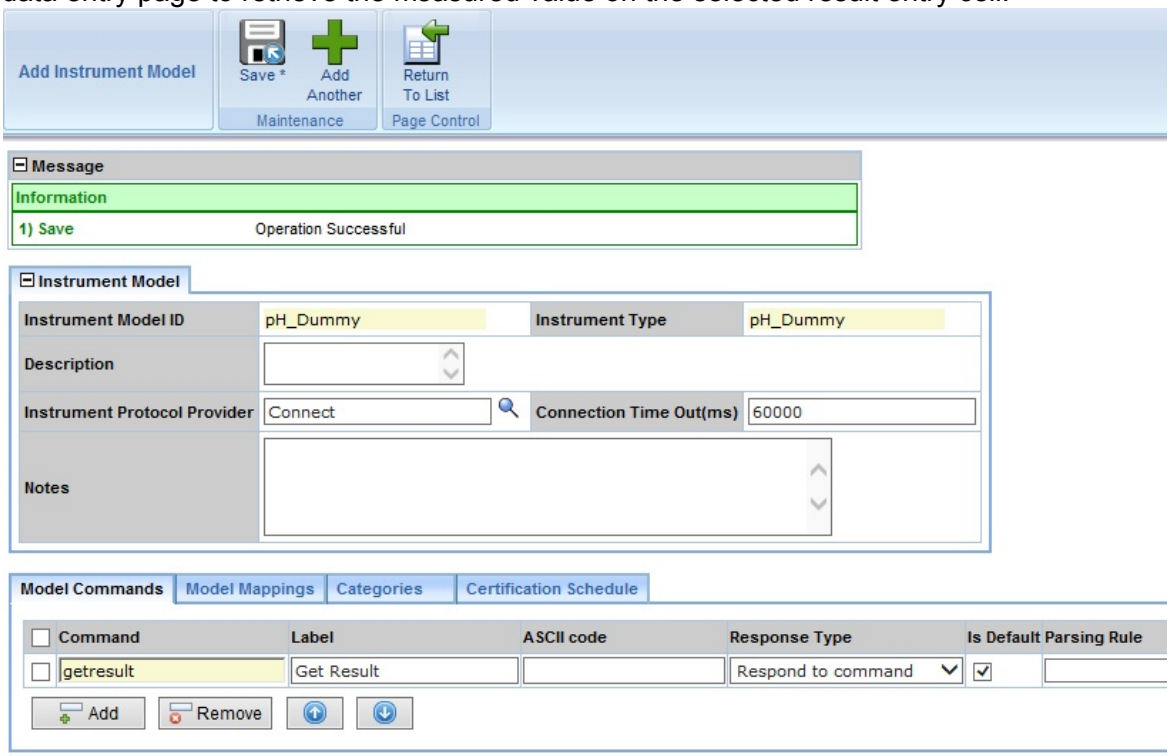
For category I instruments you need to define Instrument Protocol Provider. Select the Instrument Protocol Provider by clicking the search button shown above to open the look up page.



Connect comes with predefined Instrument Protocol Provider 'Connect', select that and hit

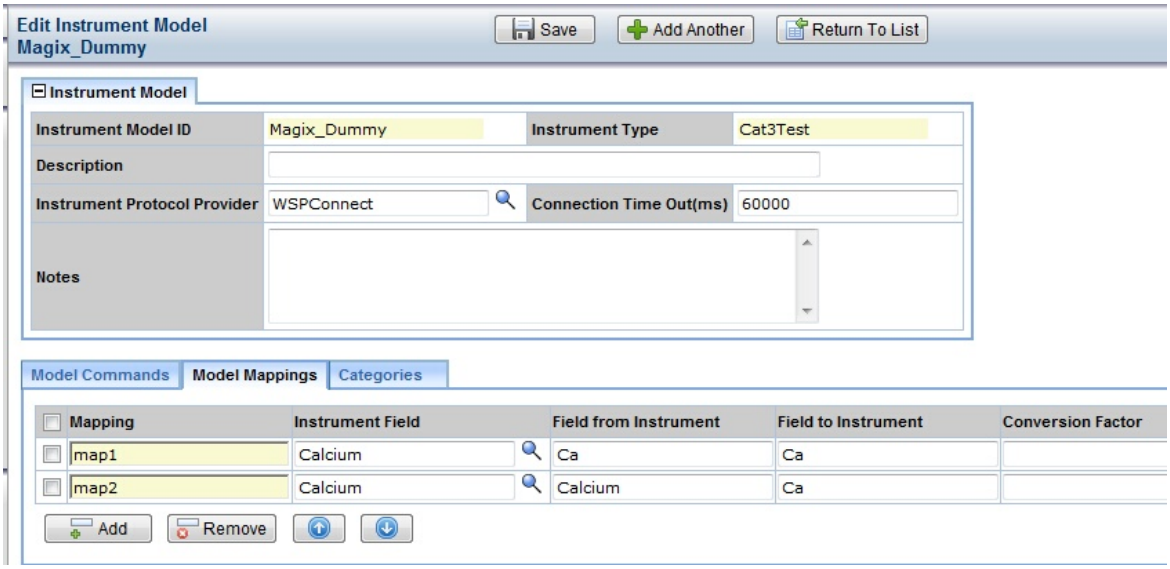
Select & Return button. In this page there is also possibility to add custom Instrument Protocol Provider.

For category I instruments you also need to define the command that will be used on the data entry page to retrieve the measured value on the selected result entry cell.



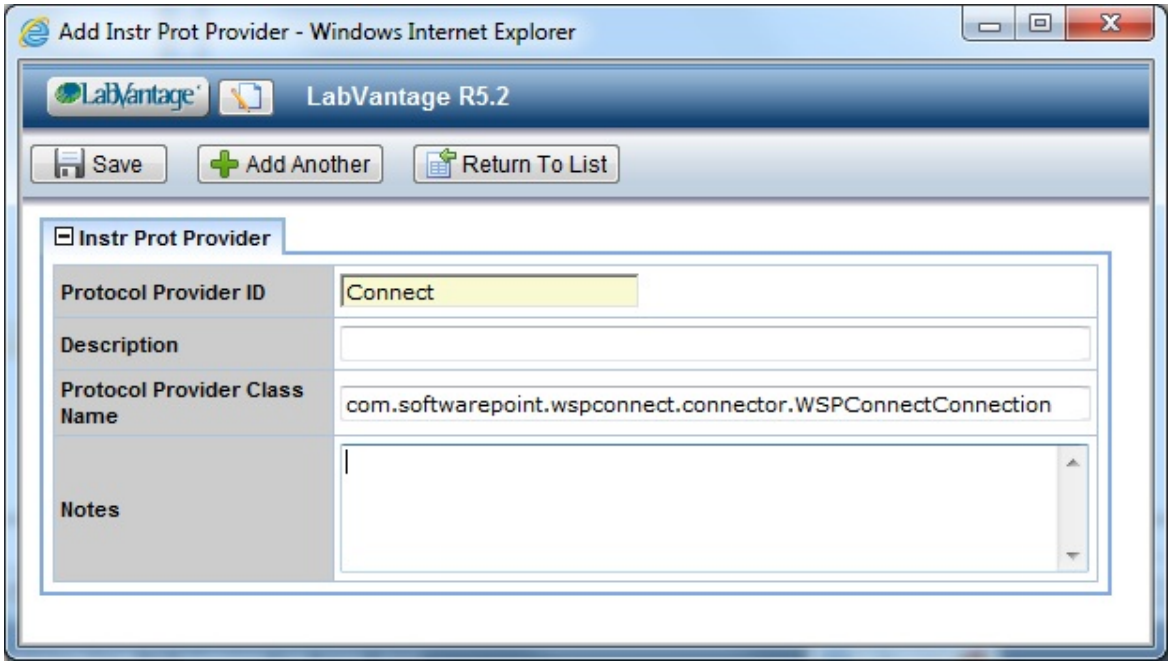
If the name that the instrument uses for tests (parameter name *to* instrument) differs from the name that the instrument uses for results (parameter name *from* instrument), this can be defined in the InstrumentModel. InstrumentModel is also used to configure the possibility that the instrument uses multiple names for the same parameter, but they should still be mapped to the same parameter in LIMS.

The following screenshot shows the case where the instrument uses the name "Ca" for tests, and responds the results as "Ca" or "Calcium". Both results should be mapped to the instrument field Calcium.



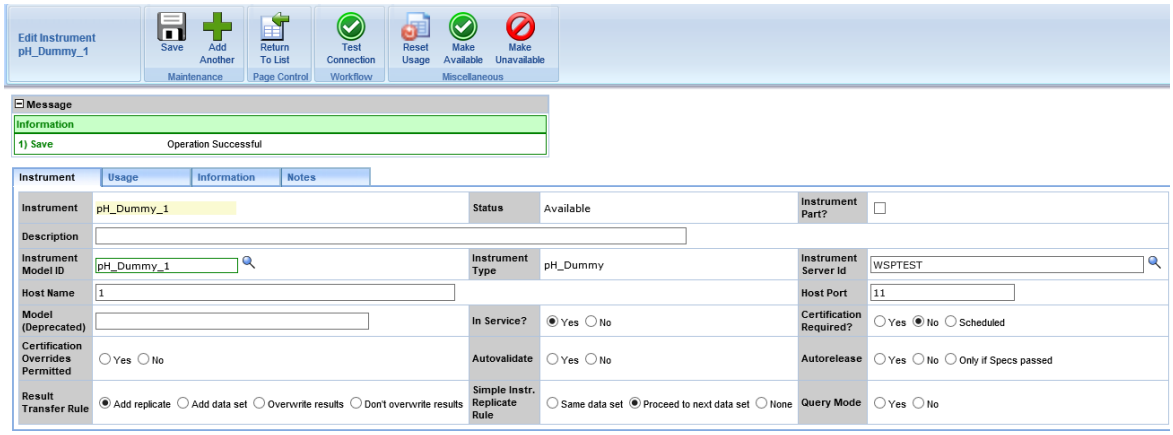
5.2.3 Instrument protocol provider

For the Instrument Protocol provider, define the ID as "Connect" and the provide class as *com.softwarepoint.wspconnect.connector.WSPConnectConnection*



5.2.4 Instrument

You need to link to the instrument to instrument model and to the Connect server (Instrument Server Id).



The instrument unique id must be the same as the instrument instance id defined in Connect. Select the Instrument Server Id to be the Connect instance this instrument is connected to.

Select Yes for the "In Service" if you want this instrument be on-line and connected to the Connect server.

Select Yes for the "Autovalidate", if you want that results form the instrument will be automatically validated and transferred to the LABVANTAGE data items and will be readily visible on the Data Entry pages. If you select No for the "Autovalidate", the results form the instrument will only be transferred to the Instrument results page, where you must manually validate (on invalidate) the results, before they will be move to LABVANTAGE data items.

Select Yes for the "Autorelease", if you want that results transferred to LABVANTAGE data items will be automatically released. Select Only if Specs passed, if you wish to release only those samples that have passed specifications and limits. Select No to manually release the results in the same way as you had entered the result manually on the LABVANTAGE data entry pages.

Result Transfer Rule applies to category 2-4 instruments. It controls what to do if there are more results coming from the instrument than will fit in the data items. 'Add replicate' will add a required amount of new replicates to data set so that all results will have a place in data set. 'Add data set' will add a required amount of new data sets to sample. If there exists partly filled data set, it will be filled first and rest of the result will be filled to newly added data set(s). 'Overwrite results' will overwrite unreleased result in Data Entry page.' Don't overwrite' will leave results unhandled if instrument sends results to sample/parameter that already have result.

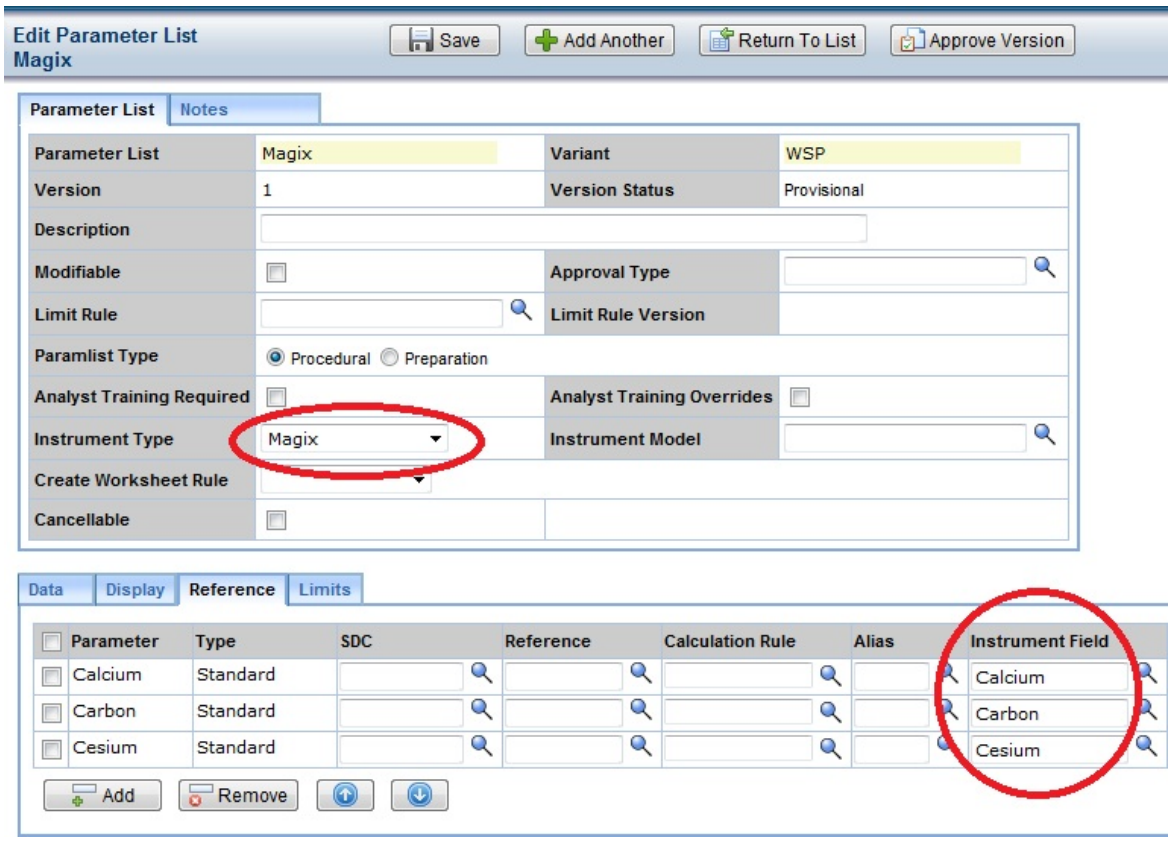
Simple Instrument Replicate Rule applies to category 1 instruments. If 'Same data set' or 'Proceed to next data set' is chosen, Simple Instrument button in Data Entry page only needs to be pressed once and results for the replicates will be filled in the Data Entry page cells as new measurements are available. 'Proceed to next data set' option continues to fill

in the values to next data set if there are no more replicates left on the first one. Option 'None' will not handle replicates.

Query Mode radio button controls sending new tests automatically to Connect. If Instrument in not using 'Query mode', No should be selected to cut down on the data downloaded to Connect.

5.3 Parameter List

To be able to get results for a parameter from a certain instrument type, you need to define the connection on the LABVANTAGE Parameter List as follows:



Define either the Instrument Type and the Instrument Model that can use this parameter list. On the Reference tab, select the mapping of the parameter name to the instrument field you have defined for this instrument type (see chapter Instrument type).

5.4 Instrument results

Results from instruments (Category II and III) will always end up in Instrument Result List page which can be found from Connect tram line.

If the instrument has the auto validation option (see Instruments) selected, the results from the instrument will be accepted without extra validation and they will automatically be transferred to LABVANTAGE data items and will appear on the Data Entry pages. Otherwise, results can be validated on this page by selecting individual results and clicking the Validate button. From the validated result rows user will see where the result will go (parameter list, version, variant, data set and replicate number) when transferred. New DS -column value 'Y' means that new data set will be added to sample. New Rep -column value 'Y' means that new replicate will be added to data item. Pressing Transfer -button will transfer selected results to LABVANTAGE data items. Validate+Transfer button executes both functions with one click.

You can navigate directly to sample's data entry and parameter list page from links in the table.

Debug Transfer button can be used to find out why some results are not transferring to sample's data entry page. Select one result row and hit the button and you will get an info message about what is missing/wrong in configuration.

(Add) Instrument Fields button is used to add Instrument Fields automatically based on selected result rows. Instrument Fields can be added manually on Instrument Type page but if there are hundreds of parameters coming from instrument, this is a time consuming and can lead to typos. Adding Instrument Fields automatically is done on configuration stage when other configuration has been done so far that result can be read in and viewed on Instrument Result List page. So you have to have Instrument, Instrument Model and Instrument Type already defined. Functionality works so that you select rows which Instrument Fields you want to add to Instrument's Instrument Type as Instrument Fields and press (Add) Instrument Field button on tool bar. If there is only one possible Instrument Type for selected rows, Instrument Type fields will be created and user will be notified how many fields were added. If no or many possible Instrument Types are found you get an

error message and need to change your row selection.

Note! In cases where replicates will be added, replicates needs to be handled in order. For example, if Validation shows that replicates 2, 3 and 4 will be added to parameter x, you can't transfer replicate 4 first. You need to transfer replicates 2, 3 and 4 in the same time or in order.

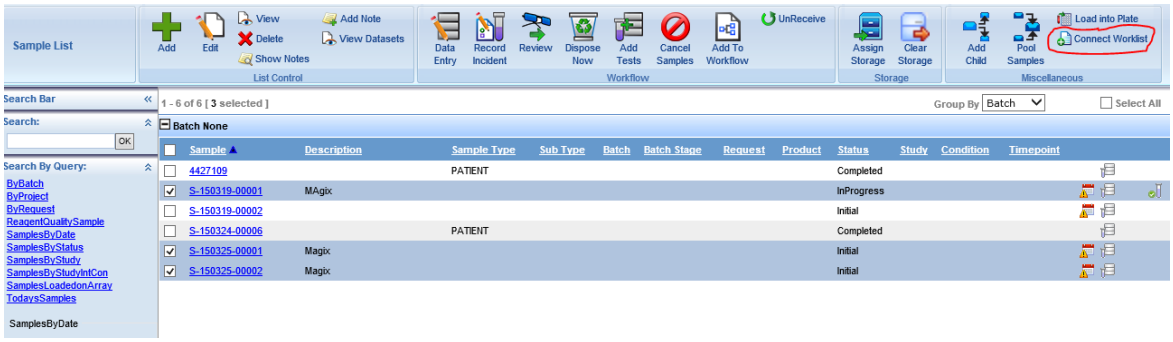
5.5 Instrument worklists

Some of the Category III instruments need to have a worklist (also called runlist) downloaded to be able to work. The Instrument worklist function allows you create such list of samples / parameters that is supposed to be run on a specific instrument. Data to be send to instrument is defined in Connect queries. Sometimes if you for example have different types of Category III instruments, you need to define specific data gathering queries for each instrument. You can do that by using Connect policies.

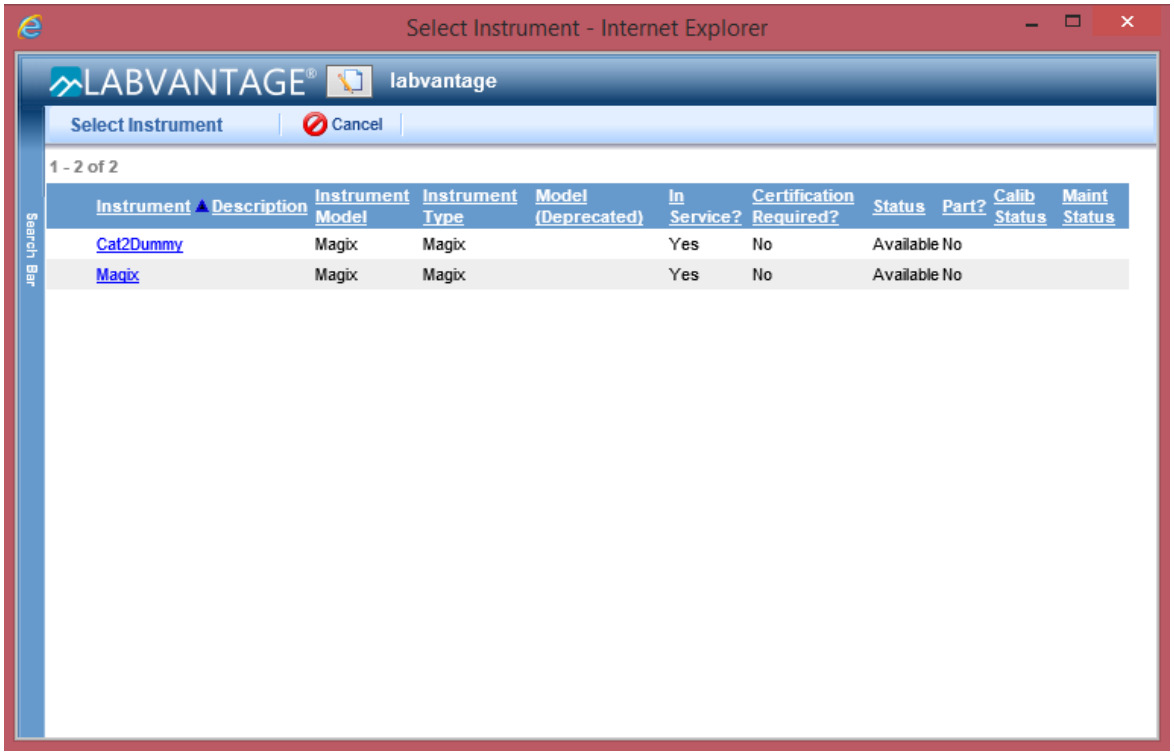
Connect supports creating worklists for instruments in three different ways: Creating worklist from Sample List page, using Connect Worklist wizard and using the LABVANTAGE QCBatch. First two uses WorkList SDC and last LABVANTAGE QC Batch SDC. Connect's instrument driver's responsibility is to form a instrument specific worklist format from the sample ids and other data and pass it to instrument in question.

5.5.1 Worklist from Sample List page

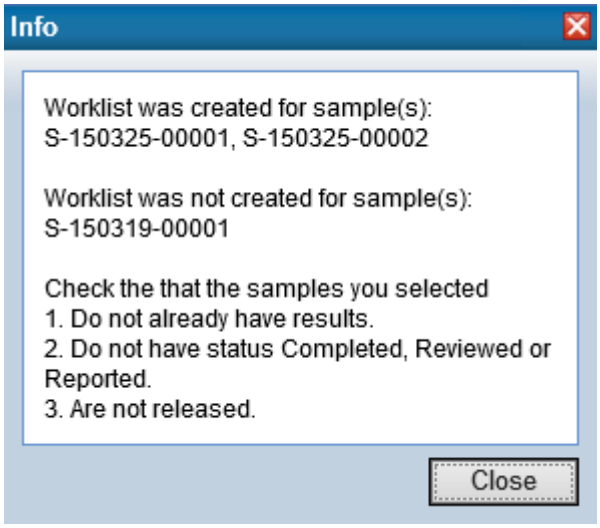
To create worklist for instrument from LABVANTAGE Sample List page, select the samples and press Connect Worklist button.



After pressing the Connect Worklist button, the Instrument Selection window will open. Select the instrument by clicking the name of the instrument.

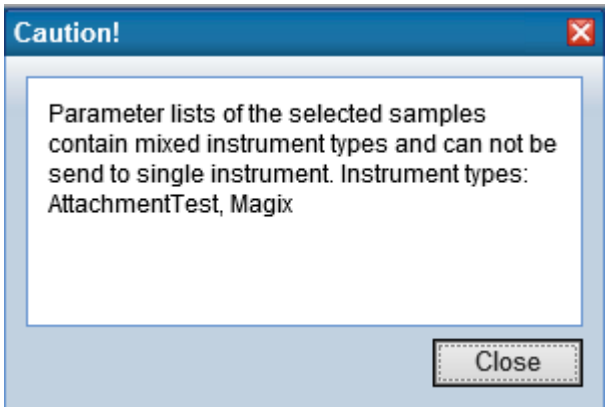


Selecting the instrument will trigger the worklist creation and it is immediately send to the Connect. User will get an info about which samples where send to instrument and also if there were samples that could not be send.



Worklist items created with this functionality are put to WorkList SDC and can be viewed on Connect's Instrument Worklist page.

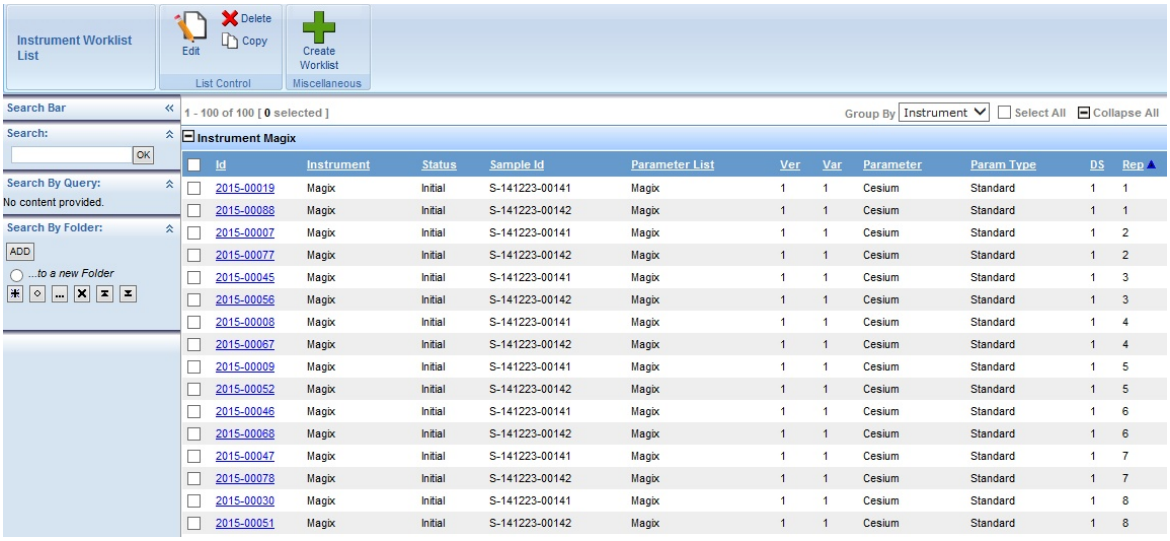
Note! Because worklists are created for one instrument at time, selected samples parameter lists needs to have same instrument type, otherwise you get the notification:



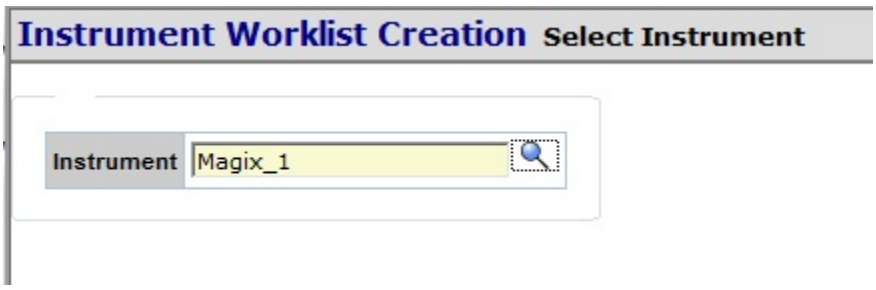
5.5.2 Using the Connect Worklist wizard

The following screens shows how you can create worklists using the Connect Worklist wizard.

Click the 'Instrument Work Lists' menu item on the LabAdmin Sitemap. This will open the list page of the created worklists.

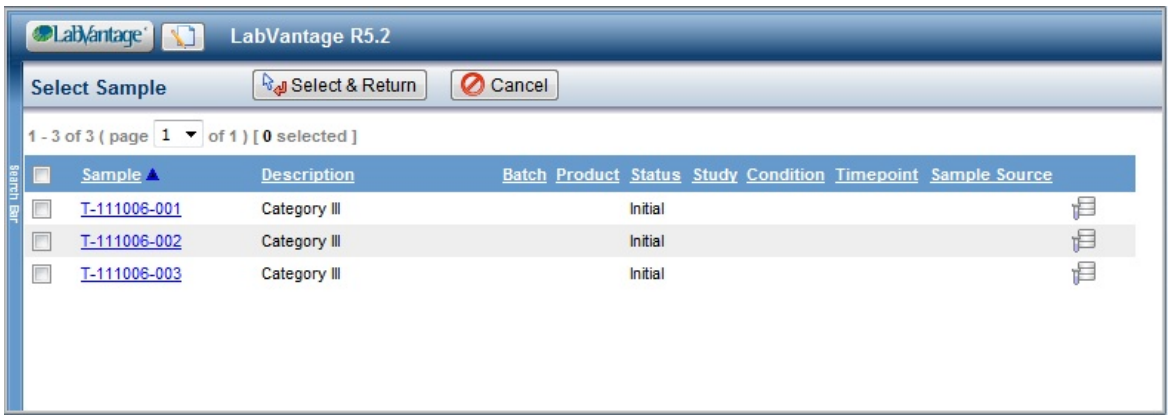


Click the 'Create Worklist' button, the WorkList Creation wizard opens.

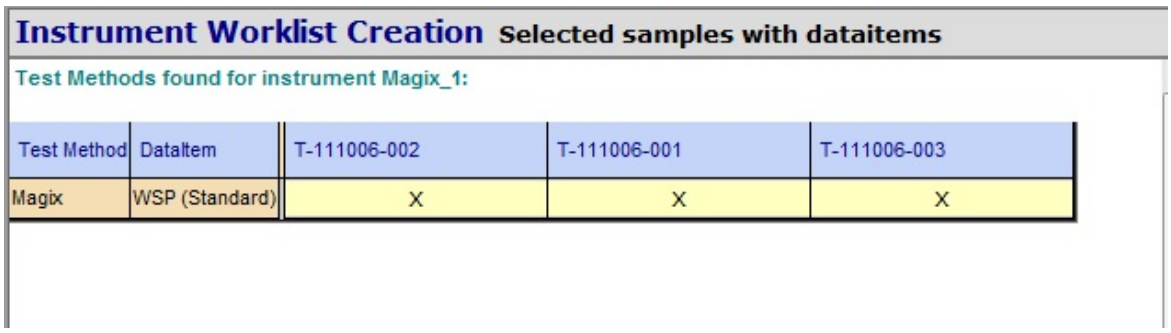


Select the instrument you want to work with, then click Next.

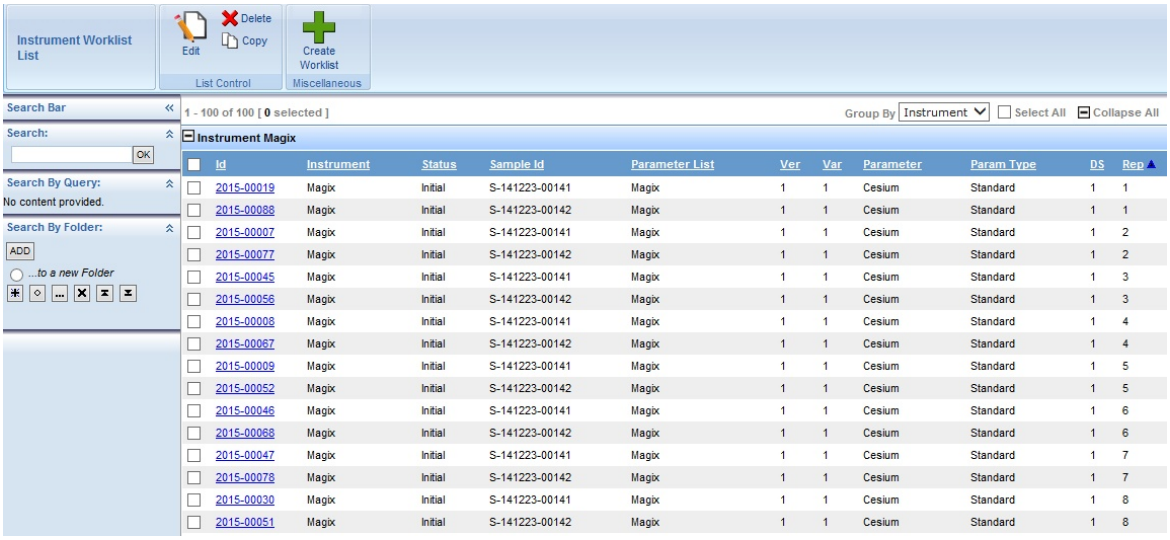
Next, add and select the samples you want to use by clicking the 'Add Samples' button.



Select the samples you want to include in the worklist, then click the 'Select & Return' button to close the selection dialog. When the dialog closes, click Next on the wizard page. If the selected samples have parameter lists that this instrument can run, those parameter lists will be marked with an X like this:



Click Next. The list page of the created worklists will open.



Id	Instrument	Status	Sample Id	Parameter List	Ver	Var	Parameter	Param Type	DS	Rep
2015-00019	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	1
2015-00088	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	1
2015-00007	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	2
2015-00077	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	2
2015-00045	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	3
2015-00056	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	3
2015-00008	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	4
2015-00067	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	4
2015-00009	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	5
2015-00052	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	5
2015-00046	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	6
2015-00068	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	6
2015-00047	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	7
2015-00078	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	7
2015-00030	Magix	Initial	S-141223-00141	Magix	1	1	Cesium	Standard	1	8
2015-00051	Magix	Initial	S-141223-00142	Magix	1	1	Cesium	Standard	1	8

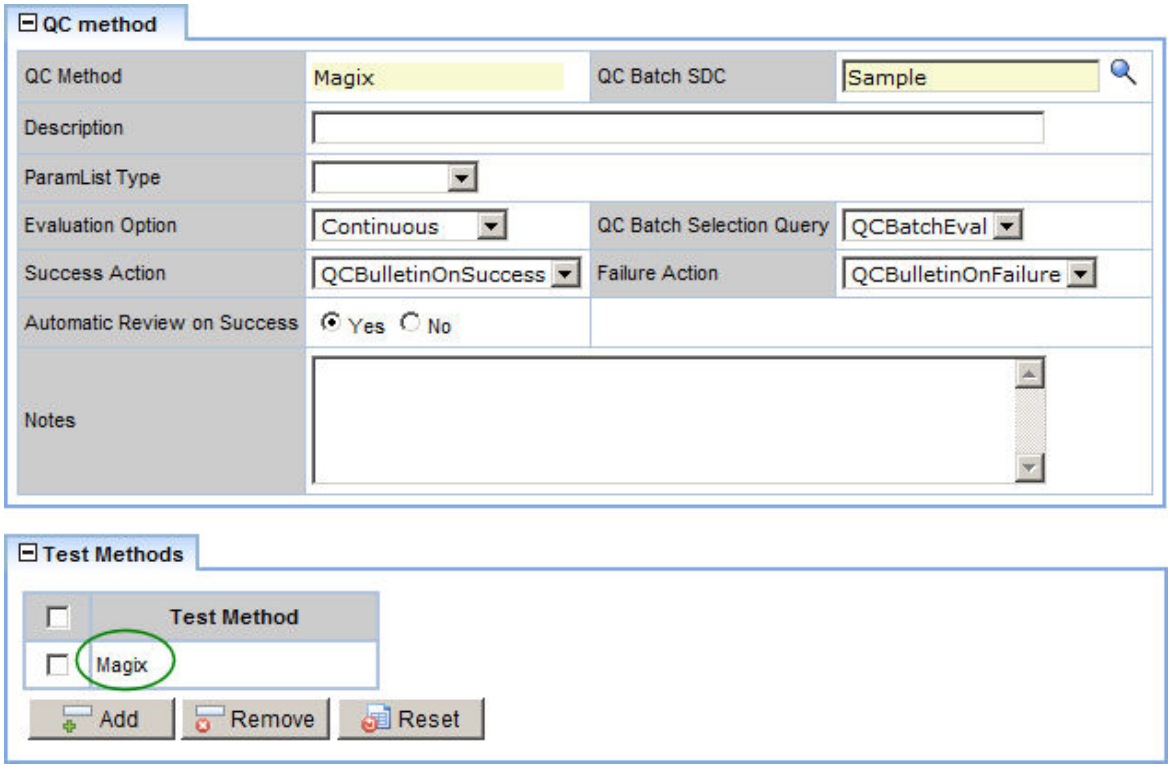
Your worklist is now created and it will be immediately send to Connect.

Note! Worklist is created only for analyses which don't yet have any results. If you wish to create a new worklist for samples which already have results, first clear the existing results from data entry.

5.5.3 Using the LABVANTAGE QC Batch

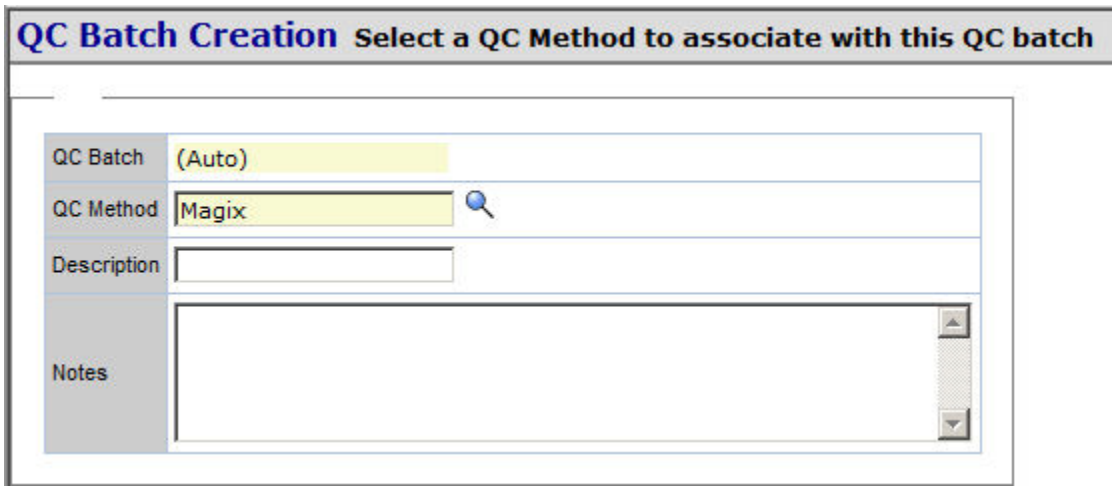
The following screens shows how you can create worklists using the LABVANTAGE QCBatch SDC. This manual only shows one simple example on how to use the LABVANTAGE QC Batch functionality. For more details, please see the LABVANTAGE LIMS documentation, chapter AQC.

To be able to create QC Batches, you need to have a QC method, which includes the test method(s) you want to run in the instrument. Click the 'QC Method' menu item on the LABVANTAGE Sitemap, then click the New button.

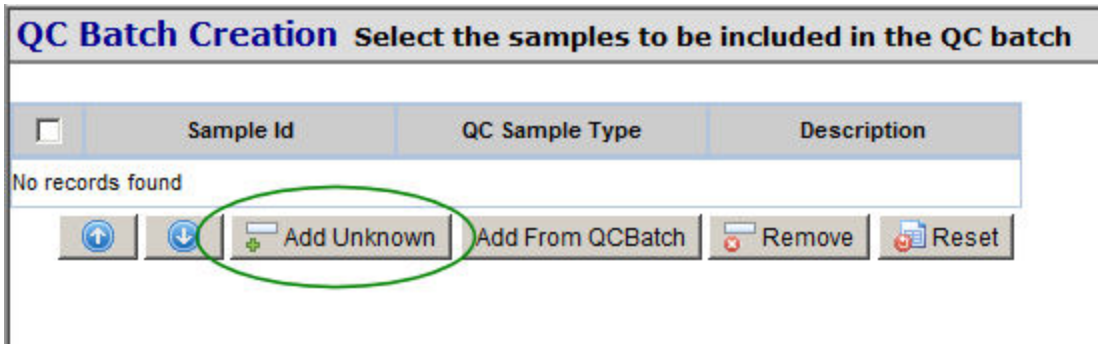


Add the Test method(s) you want to use with the instrument, then save the QC Method.

Next you are going to create a new QC batch. Click the 'New QC Batch' menu item on the LABVANTAGE Sitemap, which will start the QC Batch Creation wizard.

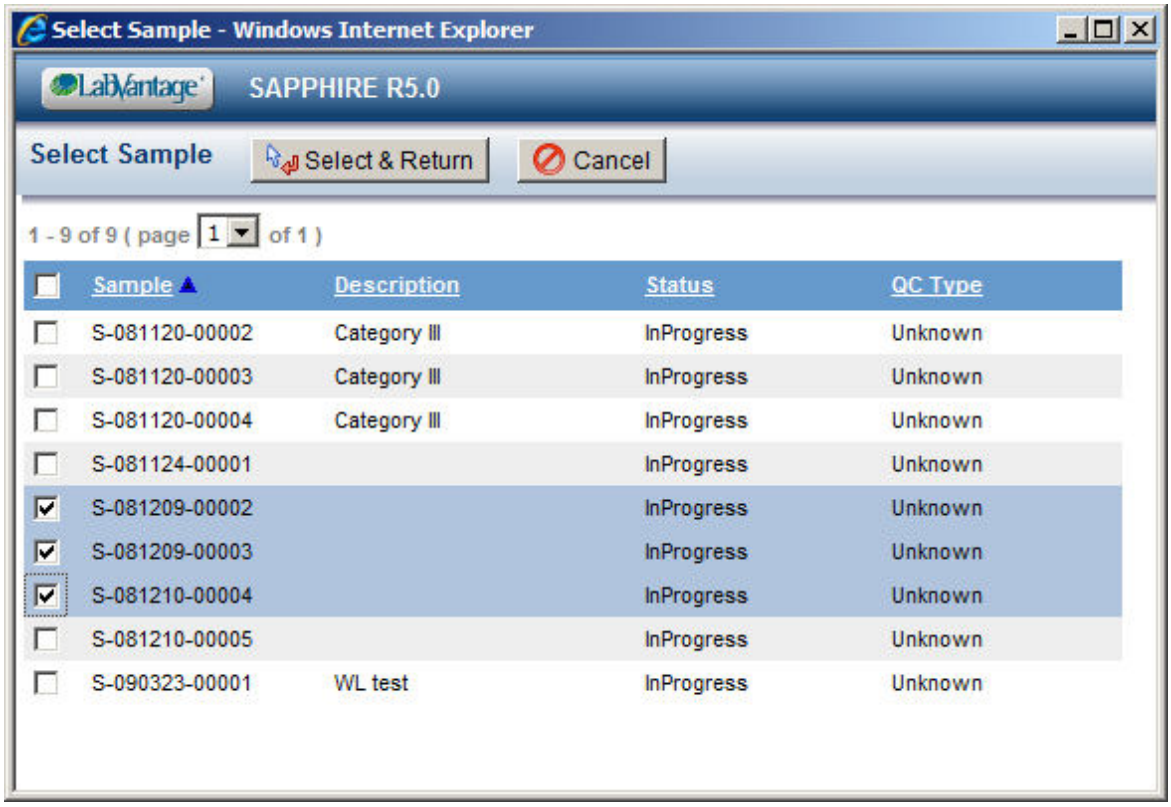


Select the QC Method you are going to use, then click Next.

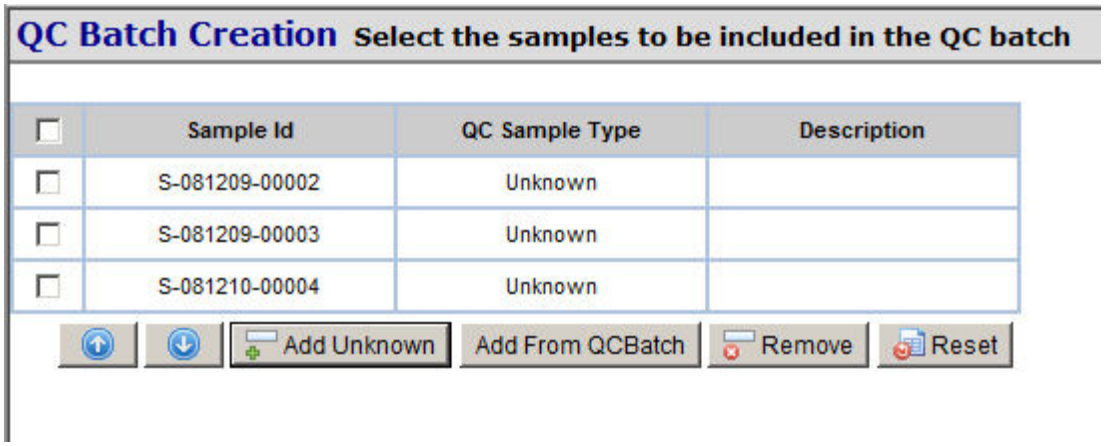


Next, add and select the samples you want to use by clicking the 'Add Unknown' button. Notice that the sample selection dialog will only show samples that match the following criteria (according to LABVANTAGE documentation):

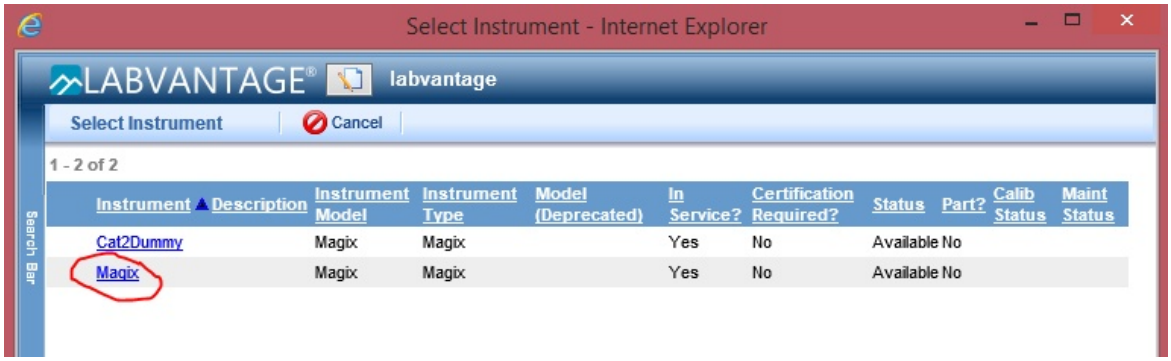
- Data Sets are in the "Test Methods" defined by the QC Method.
- Data Sets are not linked to and not present in any other QC Batch.
- Data have not been entered for any Data Sets. Data Set status is "Initial".
- Sample status is "Received" or "InProgress".



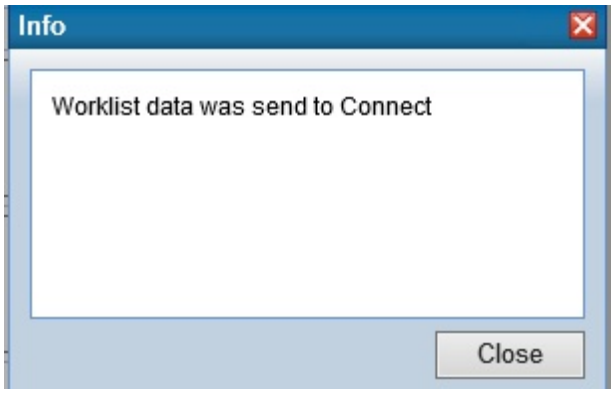
Select the samples you want to include in the worklist, then click the 'Select & Return' button.



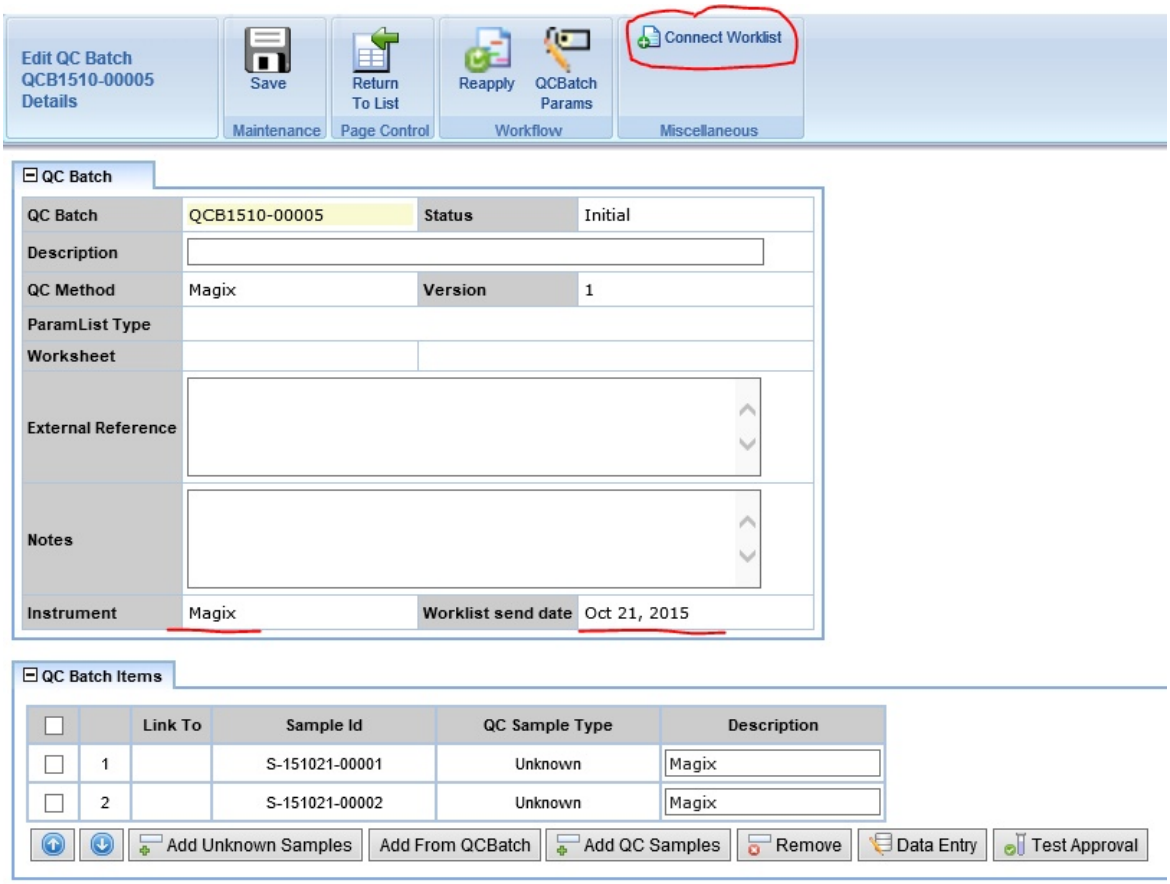
After you have added all the samples click Next. From the Edit QC Batch page press "Connect Worklist" button that is located in Advanced toolbar.



Pressing "Connect worklist" button will open the "Instrument picker" window. Select the instrument you want to send worklist to.



After data is gathered and send to Connect you will get a notification pop-up. Note that if there is hundreds of test to be send to Connect, this might take a few seconds.

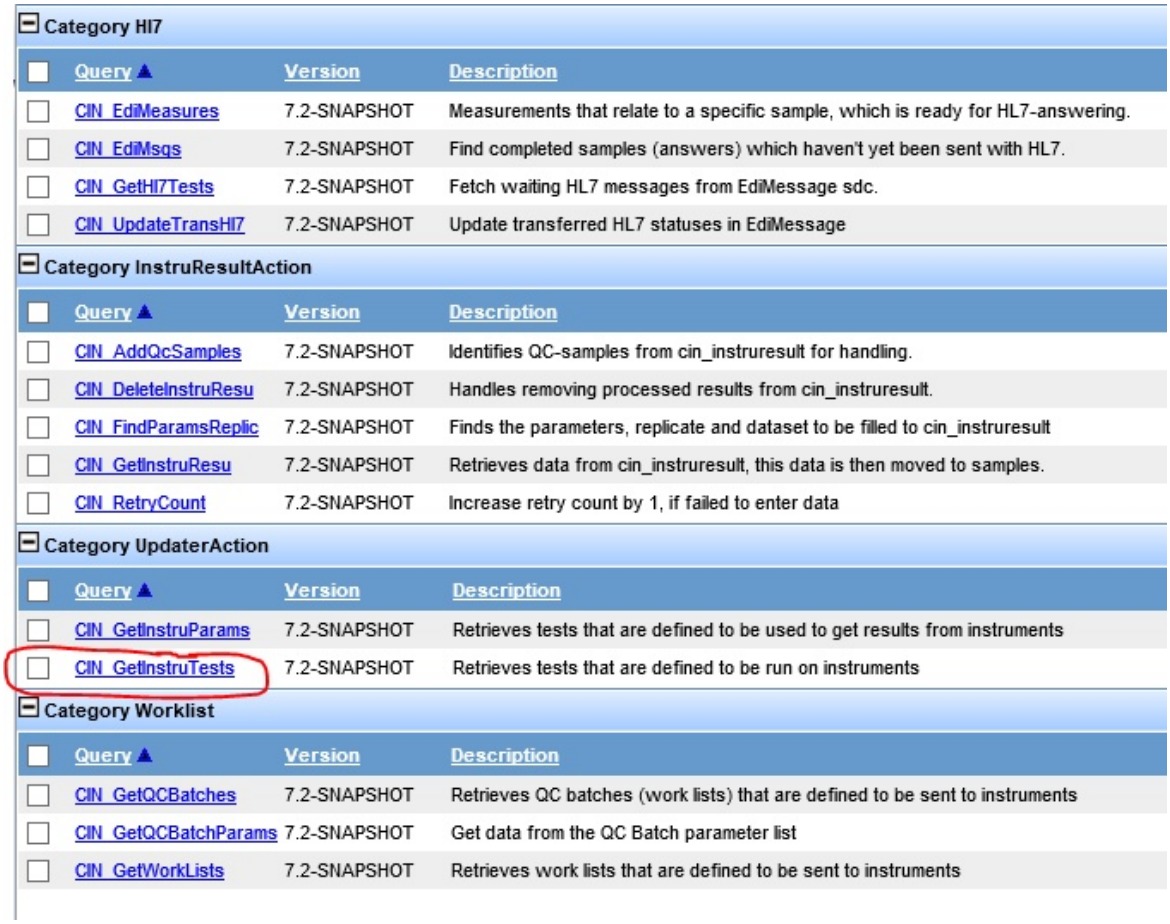


After worklist has been successfully send to Connect, "Instrument" and "worklist send date" fields in Edit QC Batch page are filled accordingly. From these fields you check if worklist has been already send at least once. Sending worklist again will update these fields.

Your QC Batch worklist creation steps are now done and Connect instrument driver will use the data send to Connect to create instrument specific worklists.

5.6 Query-mode Instruments

For query-mode -instruments, Connect will download new tests from LIMS to its internal database. Tests are downloaded using the query CIN_GetInstruTests, which can be found from LVConnect Queries -page.



Category HI7

Query	Version	Description
CIN_EdiMeasures	7.2-SNAPSHOT	Measurements that relate to a specific sample, which is ready for HL7-answering.
CIN_EdiMsgs	7.2-SNAPSHOT	Find completed samples (answers) which haven't yet been sent with HL7.
CIN_GetHl7Tests	7.2-SNAPSHOT	Fetch waiting HL7 messages from EdiMessage sdc.
CIN_UpdateTransHl7	7.2-SNAPSHOT	Update transferred HL7 statuses in EdiMessage

Category InstruResultAction

Query	Version	Description
CIN_AddQcSamples	7.2-SNAPSHOT	Identifies QC-samples from cin_instruresult for handling.
CIN_DeleteInstruResu	7.2-SNAPSHOT	Handles removing processed results from cin_instruresult.
CIN_FindParamsReplic	7.2-SNAPSHOT	Finds the parameters, replicate and dataset to be filled to cin_instruresult
CIN_GetInstruResu	7.2-SNAPSHOT	Retrieves data from cin_instruresult, this data is then moved to samples.
CIN_RetryCount	7.2-SNAPSHOT	Increase retry count by 1, if failed to enter data

Category UpdaterAction

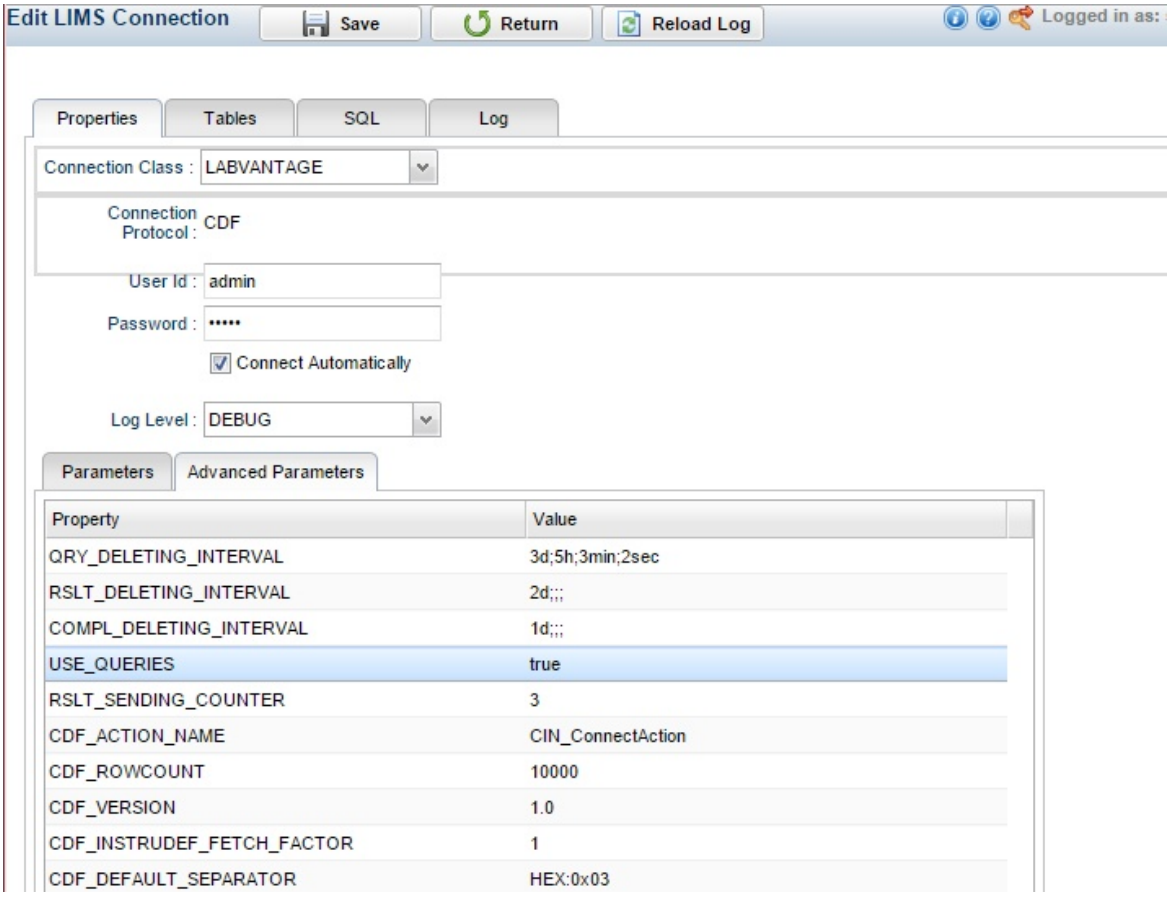
Query	Version	Description
CIN_GetInstruParams	7.2-SNAPSHOT	Retrieves tests that are defined to be used to get results from instruments
CIN_GetInstruTests	7.2-SNAPSHOT	Retrieves tests that are defined to be run on instruments

Category Worklist

Query	Version	Description
CIN_GetQCBatches	7.2-SNAPSHOT	Retrieves QC batches (work lists) that are defined to be sent to instruments
CIN_GetQCBatchParams	7.2-SNAPSHOT	Get data from the QC Batch parameter list
CIN_GetWorkLists	7.2-SNAPSHOT	Retrieves work lists that are defined to be sent to instruments

For large databases, it is recommended to create an index to column moddt of table sdidataitem. Otherwise, the query can be very inefficient.

Downloading of the tests is OFF by default, and have to be enabled from Connect, by editing the properties of the LIMS Connection. USE_QUERIES should be set to true to enable downloading the tests.



Query mode also need to be turned on for instrument in question. That can be done in LABVANTAGE Instrument page with Query Mode -radio button.

5.7 Getting results from Data Entry page

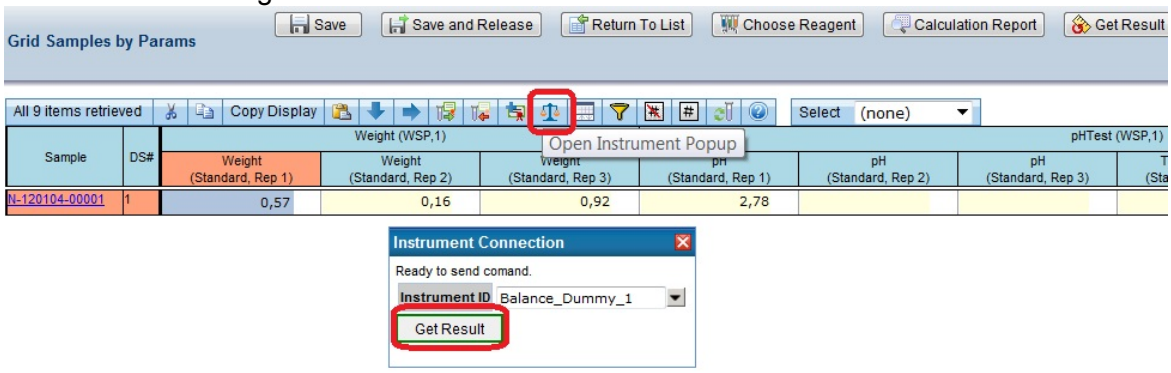
There is a possibility to import results to Data Entry page with buttons in Data Entry page's Get Result group in advanced toolbar. Following chapters will explain how to use this functionality with different category instruments.

5.7.1 Simple Instruments

Category I -instruments can be used from Data Entry -pages in two different ways. Either through the Instrument Popup, which is LABVANTAGE OOB functionality, or through Simple Instrument -button, which comes with Connect-installation.

5.7.1.1 Using Instrument Popup

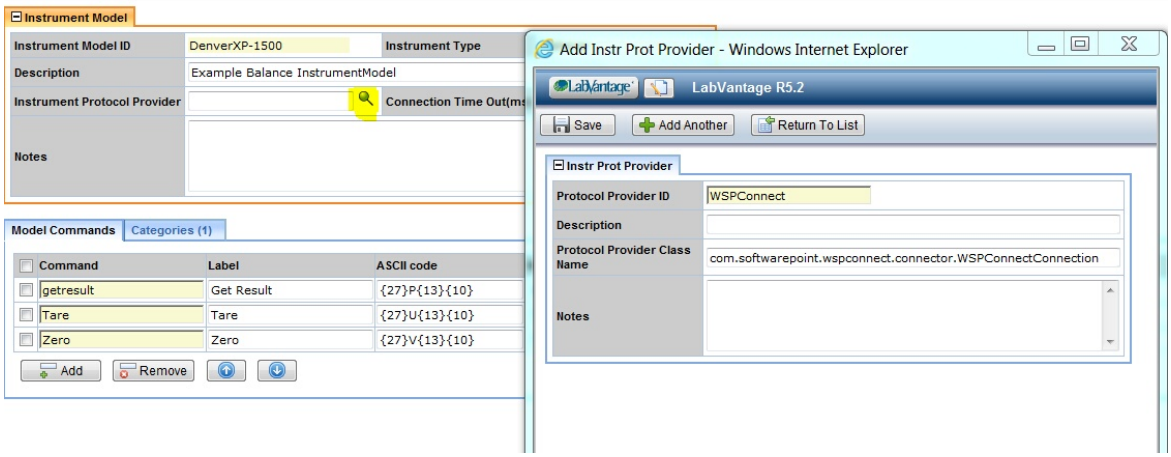
Instrument popup can be opened by clicking an icon in the data entry -page. Instrument Popup has a dropdown menu, where you can select which instrument you want to use. Pressing the Get Result -button in the popup will fetch a result from the instrument. This is illustrated in the image below.



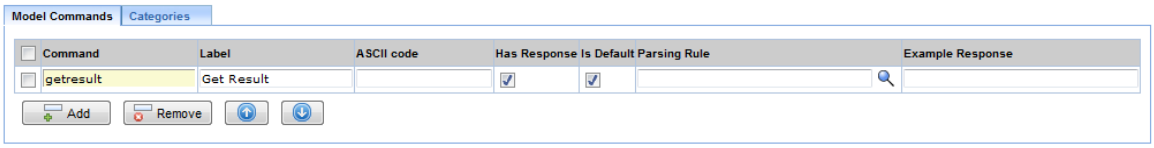
This feature is available in all GWT dataentry -views, but is not available in Classic -views.

To use this framework, first register an Instrument Protocol Provider to handle communication with Connect. The protocol provider can be selected (or added) using the Instrument Protocol Provider lookup in the Instrument Model maintenance page. The protocol provider for LV Connect uses the the full class name:

```
com.softwarepoint.wspconnect.connector.WSPConnectConnection
```



Then, for every instrument, define that this is used for the instrument protocol, and register one command, and set it as default:



The instrument name is saved in sdidataitem in field "instrumentid" for all populated results.

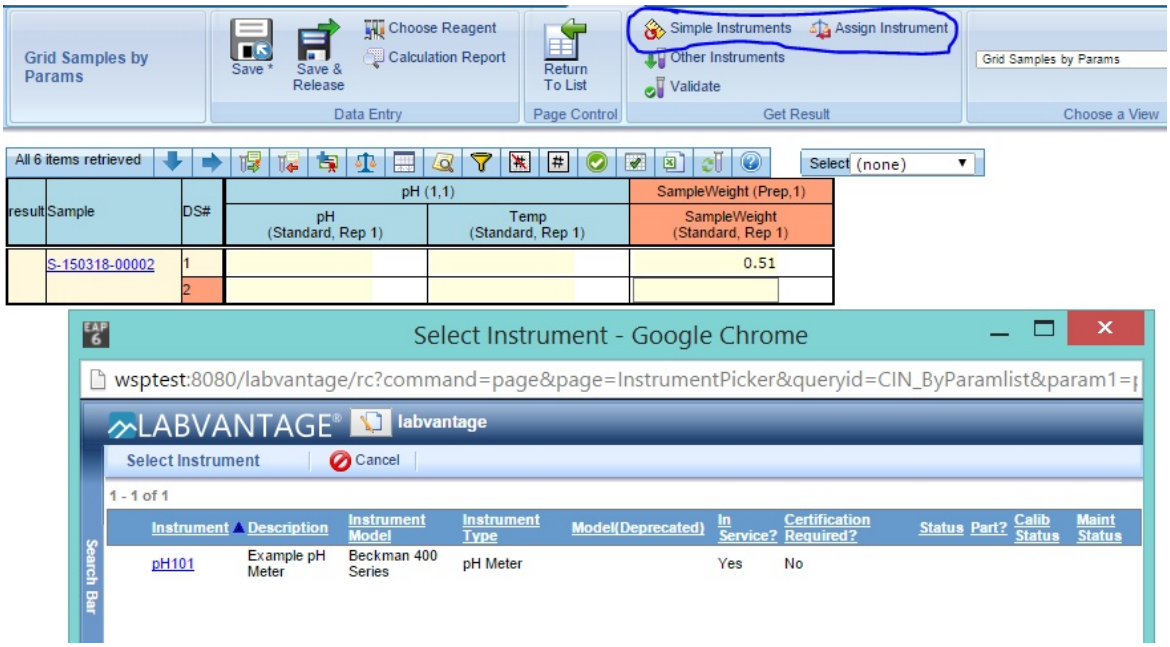
5.7.1.2 Using the Simple Instrument -button

Other way to get results from category I instruments is to use Connect's Simple Instrument -button which is in advanced toolbar's Get Result group. In this case, the instrument to use must be defined in the dataset.

The instrument can be saved to the dataset by using the "Assign instrument" -button in the Get Result group. This will open up a page which shows all instruments in the system which have the same instrument type as the selected dataset. When you select the instrument from the "Select Instrument" -page, this will be immediately saved to the dataset, and you can next use the Simple Instruments -button to fetch the result.

Instrument can also be defined for the current dataset e.g. using the detail -tab. In this case, the modification must be saved before invoking the Simple Instruments -button.

After instrument has been defined, a result can be fetched by pressing the Simple Instrument -button in the advanced toolbar. This feature can be used in all GWT dataentry views, and also in the Classic dataentry views.



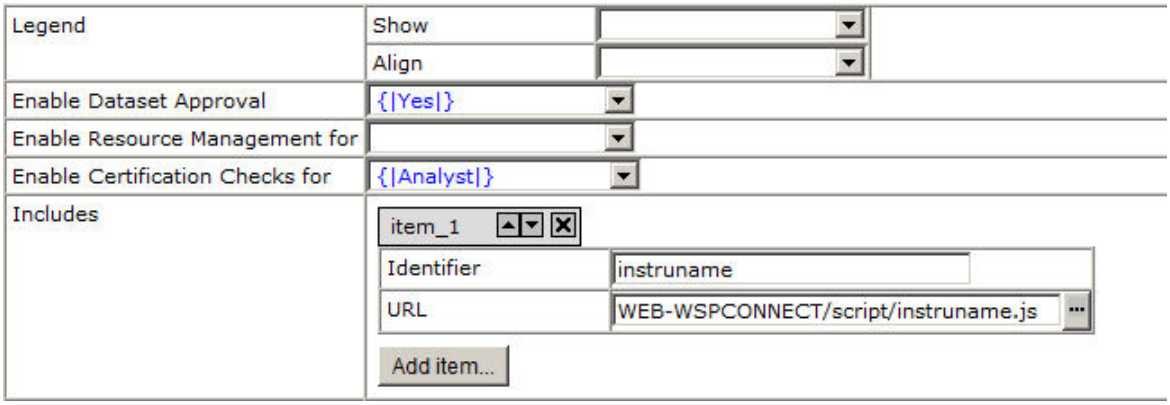
In the GWT -dataentry-views, the Simple Instrument -button supports fetching all replicates from the instrument with one click of a button. This must be configured on the Instrument-page by setting "Simple Instrument Replicate Rule" -functionality on. In this case, pressing the Simple Instrument -button will fetch a result for the currently selected replicate, and after the result arrives, tries automatically to fetch a result for the next replicate.

In the GWT -dataentry-views, Simple Instrument -button also supports more advanced data coming from the instrument, the data might e.g. define a sample id. This makes it possible to interface instruments that analyze several samples simultaneously, directly to the data entry page.

5.7.1.3 Saving the instrument id in Classic dataentry views

Instrument name is saved automatically when using GWT data entry grids. To enable saving the instrument name along with results in classic data entry views, you must include the Java Script *WEB-CIN/script/instruname.js* to the following pages (at the Page Type level):

- iDataEntryGrid
- iDataEntryList
- iDataEntryGridDS

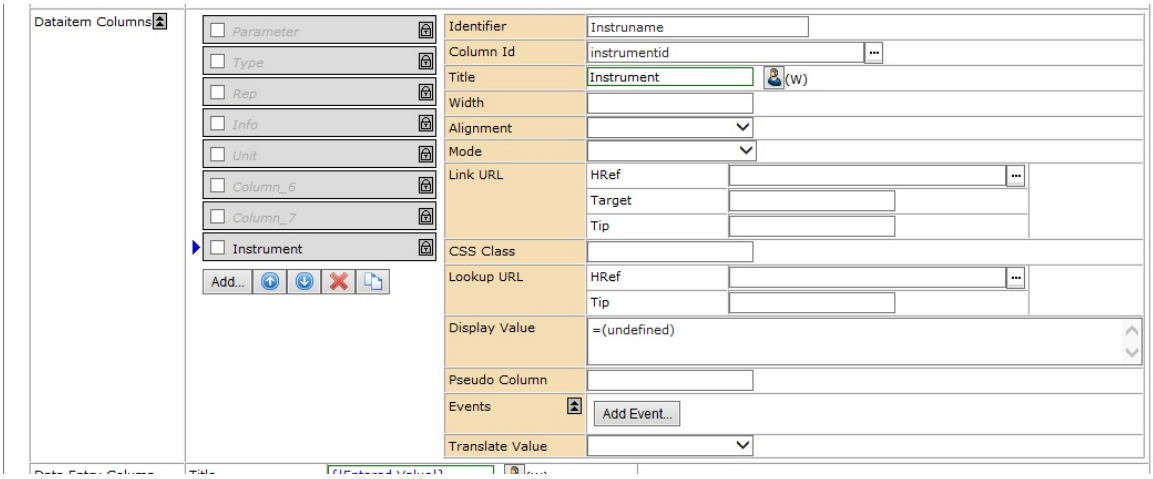


The Java Script *instruname.js* implements the following behavior:

i. Dynamically creates the *instrumentid* field to the web page (unless it already exists). This functionality is in use in the Grid Views.
ii. Sets the *instrumentid* field as read only.
iii. Implements behavior, which clears the instrument name in the *instrumentid* field, if the corresponding result is afterwards manually modified

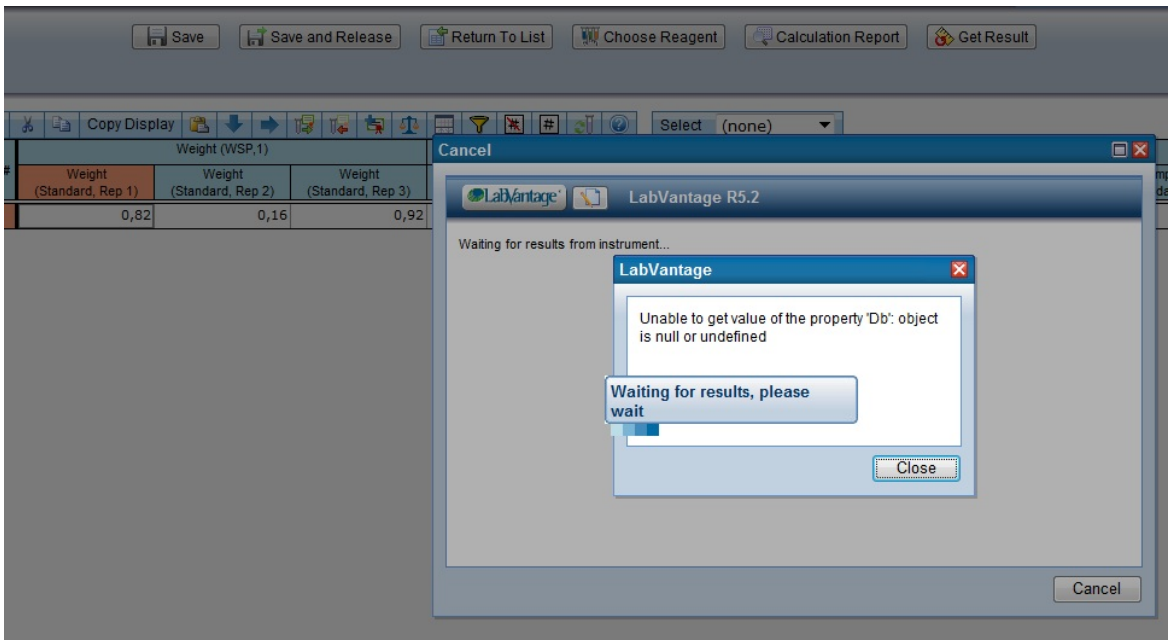
Instrument ID can be made visible on the classic list page with the following configuration:

Add the *instrumentid* column to the *iDataEntryList* page (Dataitem Columns) with the Sapphire Web Page Designer. Set the column title to "*Instrument*", mode "*Readonly*" and the display value to "*=(undefined)*", as shown below.



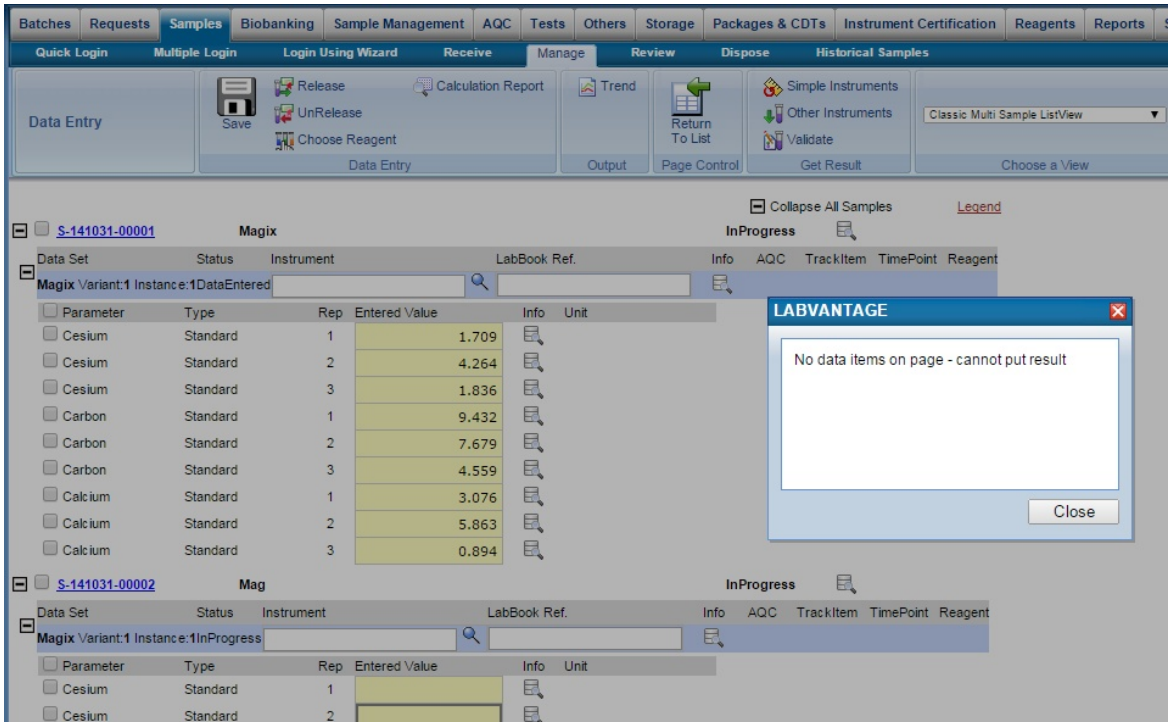
5.7.1.4 Simple Instrument error in Fast Grid dataentry view

The following error might be shown in the data entry view "Fast Grid". Workaround is to add the instrumentid -column to the view as explained in chapter Support for saving the instrument name.



5.7.1.5 Simple Instrument error in Classic dataentry views with Chrome-browser

The following error will be shown when using the Simple Instrument -button in the Classic DataEntry Views when using some other browser than Internet Explorer. This is a known issue. Simple Instrument -button works on the Classic DataEntry Views only when using Internet Explorer.



5.7.1.6 Simple Instrument error codes

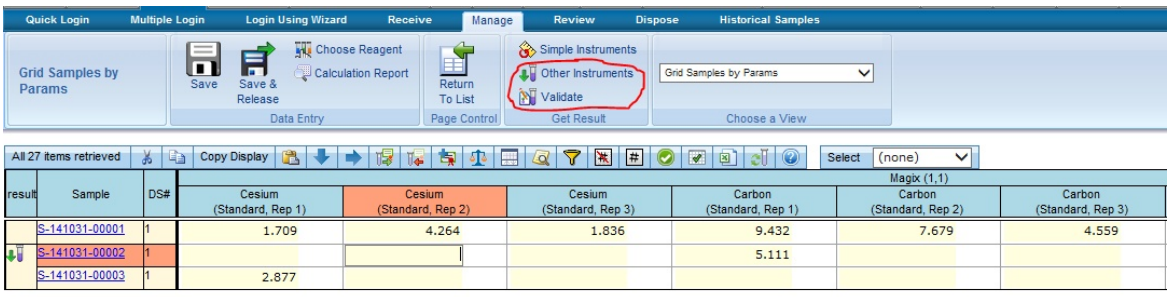
Connect-drivers can return an error code when called from LIMS. The error code follows the notation $$$N, where N is a number from 0-16. The error is normally translated to plain text before showing it to user, but may not be translated if Connect is called from some customer specific functionality.

Error Code:	Error text:	Reason:
0	Undefined error on page.	Usually a compilation error on JSP, indicates incorrectly built labvantage.ear.
1	Missing keys.	The keys passed from DataEntry (paramlistid/paramlistversionid/variantid/ paramid/keyid) are not complete. Probably called from non-standard page.
2	Imcomplete keys.	The keys passed from DataEntry (paramlistid/paramlistversionid/variantid/ paramid/keyid) are not complete. Probably called from non-standard page.
3	Database error.	Invalid SQL query, indicates that LABVANTAGE Connect datamodel is not installed.
4	Instrument not defined.	Instrument is not defined in sdidata.s_instrumentid
5	LABVANTAGE Connect host not defined.	Hostname is not defined for this Connect.
6	LABVANTAGE Connect ID not	Connect ID (Instrument Server ID) is not

	defined.	defined for this instrument
7	No response from instrument.	Complete message was not received from instrument (timeout).
8	Cannot connect to LABVANTAGE Connect server.	Cannot make the connection from LIMS to LABVANTAGE Connect. Maybe Connect is not running, or it's a firewall problem, or incorrect host/port or URL defined in the Instrument Server SDC.
9	Cannot find parameter mapping	No mapping found for the parameters in the parameter list, results not populated to the page.
10	Cannot connect to instrument	Could not make the connection from LABVANTAGE Connect to the instrument.
11	Could not get a stable value from instrument	Instrument returned results, but they were discarded as invalid / nonstable by the Connect driver.
12	Wrong number of results received	Expected a certain number of results, the interface returned too many or too few.
13	Instrument not defined in LABVANTAGE Connect	Instrument is not defined in LABVANTAGE Connect. Maybe the name of the instrument is slightly different in LIMS and in Connect.
14	Failed to process the request in LABVANTAGE Connect	Processing the request caused an exception in Connect, indicates a bug in driver.
15	Driver is not started in LABVANTAGE Connect	Driver is not started, must start the driver from the Connect user interface.
16	Instrument driver does not support remote calls	Instrument driver does not support direct calls from LIMS.

5.7.2 Other Instruments

By other instruments we mean all other instrument that simple instruments, for ex. file based instruments, ASTM instruments ect. If instrument is not configured to auto validate in Instrument maint page, there is two ways to get result from category 2-4 instruments to Data Entry page. One way is to open Connect's Instrument Result List -page, and select what results to import from there. Other way is to use buttons in Data Entry page's Advanced Toolbar's Get Result group.



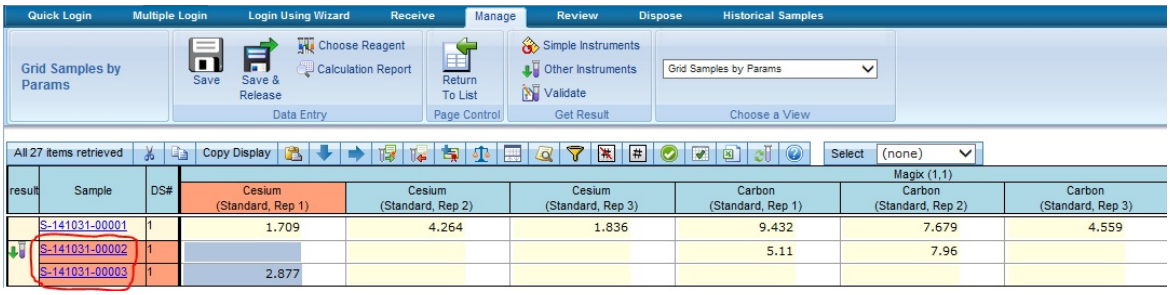
5.7.2.1 Get results without validation

Getting results without validation means that if there are suitable results for samples waiting in Instrument Result List -page, pressing the Other Instruments -button will import all results for all samples visible in on Data Entry page. Results will be saved.

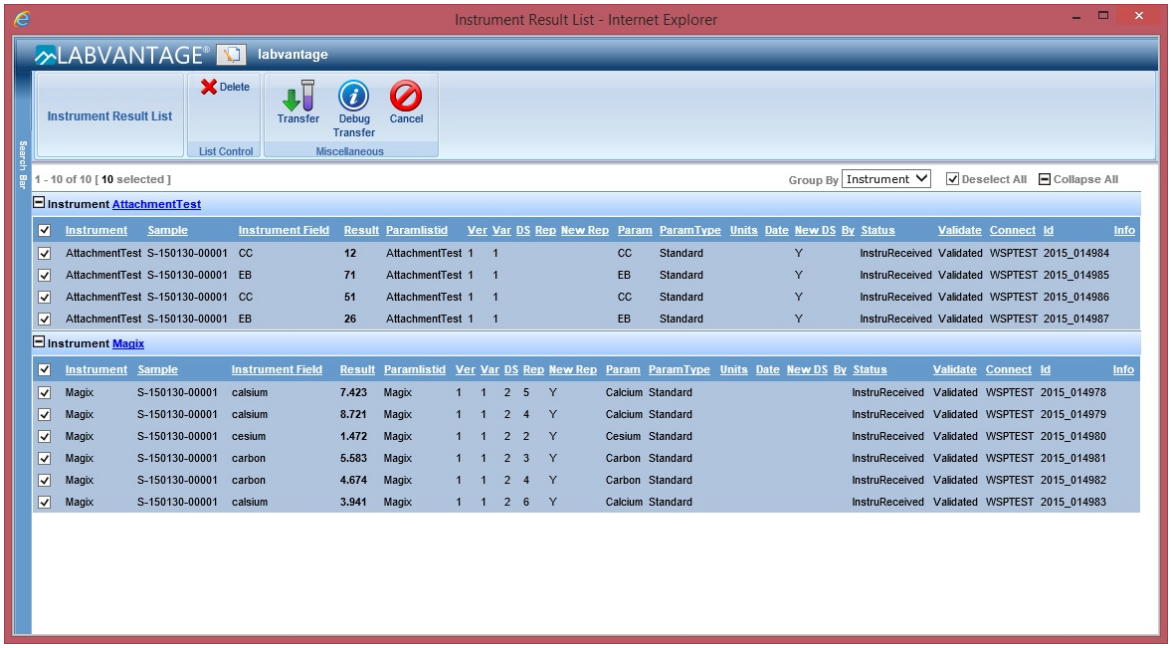
Note! Button will not work when using Classic data entry pages with Chrome browser.

5.7.2.2 Validate results

Pressing Validate -button will open a Validation pop-up page, where one can select what results to import. Which incoming result are shown on the validation page is based on cell selection on Data Entry page. If single cell is selected, validation page will open with results for sample related to selected cell. You can also select multiple cells (samples) with Ctrl key.



Validation -pop up shows user where the result would go (parameter list, version, variant, data set and replicate number). New DS -column value 'Y' means that new data set will be added to sample. New Rep -column value 'Y' means that new replicate will be added to data item. Pressing Transfer -button will import selected results to Data Entry page and close the pop-up. Results will be saved.



5.8 Connect policies in LABVANTAGE

Some properties that define how the Connect Interface operates can be set in the CIN_LVConnect policies.

Connect Interface uses Sapphire logging, mostly on the debug level. Debug-level generates a lot of Connect Interface related logging, and this can be silenced by setting the CIN_LVConnect policy 'Log level'. LogLevel can be set to anything that represents Log4J logging level, e.g. trace, debug, info, warn, error, fatal.

The LABVANTAGE actions included in Connect installation use direct SQL clauses by default, for performance reasons. If the project requires that AddSDI and similar actions would be used instead, to e.g. activate BusinessRules, this can be configured by setting the property 'Use Direct Sql' in CIN_LVConnect policy false.

Connect -actions will try to group items in actions to groups, default is 1000 items at a time. The size of the group can be changed by modifying the CIN_LVConnect Policy 'Result group size'. Max value is 1000, values bigger that that will not have an effect.

Result Group Transaction policy controls transaction handling. When set to Yes, Connect result transfer action will call LABVANTAGE actions (EnterData, EditData, AddReplicate, EditSDI and AddSDI) with the request of new transaction, which is recommended. If set to No, there is no way to for example to send bulletin/email, if LABVANTAGE actions fail (for ex. there is bad calculation rule). Also state of instrument result rows can't be updated if Result Group Transaction policy is set to No and LABVANTAGE action fail. If there is requirement that all incoming result must be committed to db at the same time, preferred way is to increase the result group size. However, if there is need to handle more that a few thousand result at one time, setting Result Group Transaction policy to 'No' will enhance performance.

When new results are inserted to LABVANTAGE database cin_instruresult -table, InstruRresultAction can be added to ToDo list immediately. This can be controlled by CIN_LVConnect policy 'Run Result Transfer automatically'.

Attachment policies are used to control where and how attachments from the Connect drivers are stored. Whether or not there is attachment file transferred to LABVANTAGE with the instrument results, is decided in specific driver's code. More about how to do it, see ConnectDeveloper -guide. Connect attachment can't be used if policy option "Use Direct Sql" is false.

If there is a performance issues on LABVANTAGE side while transferring results, Connect performance log can be turned on by giving an existing folder to policy Performance log. Log gives detail information about different result transfer stages. When the logging is no longer need it is advised to turn off performance logging by emptying policy value.

There is possibility to send bulletin or email, if action exception happens in Connect result transfer. This can happen for example when there is a bad calculation or business rule. Bulletins can be send by users or role. Functionality can be switch on and off by modifying Send bulletin and Send email policies.

You can define a custom action in Connect policy. Action will be run as a last action in Connect's result transfer functionality. More information about how to code and use custom action is in the Connect's Developer Guide.

Custom worklist and QCBatch query policies are used to define a Connect query that fetches data for specific instrument instead of standard (CIN_GetWorkLists and CIN_GetQCBatches) queries. This approach is useful in situations where there is many instruments that require different type data from lims. Instead of writing a whole new query, instrument specific queries should extend the standard queries. This is because there are certain fields and bind variables that needs to be in place for data flow to work. Connect query id defined here needs to exits in Connect query table in LVConnect Query List page. For example, take a copy of existing CIN_GetWorkLists query, rename it and do the instrument specific modifications to that query. Then fill in the instrument id and name of the custom query to policy.

Namespace policy is used to define name space of the LIMS system used for example in HL7-interfaces.

Worklist block DataEntry policy is used to prevent manual result entry when Connect's worklist functionality is used. Blog flag is set to samples DataSets when Connect worklist is created and send to Connect. When Connect enters the results to these samples, block flag is removed. Flag is removed when DataSet involved gets it's first result.

Units rule policy is used to define how the unit from instrument should be handled. Possible values are 1. Use instrument unit 2. Display instrument unit (no unit conversion) 3. Ignore instrument unit. More detailed explanation in the Policy table below.

When making changes to policy values after saving you need also to inform Connect about

changes. For now this is done by refreshing Connect queries. Go to Connect Queries page, select something and press 'Refresh' -button.

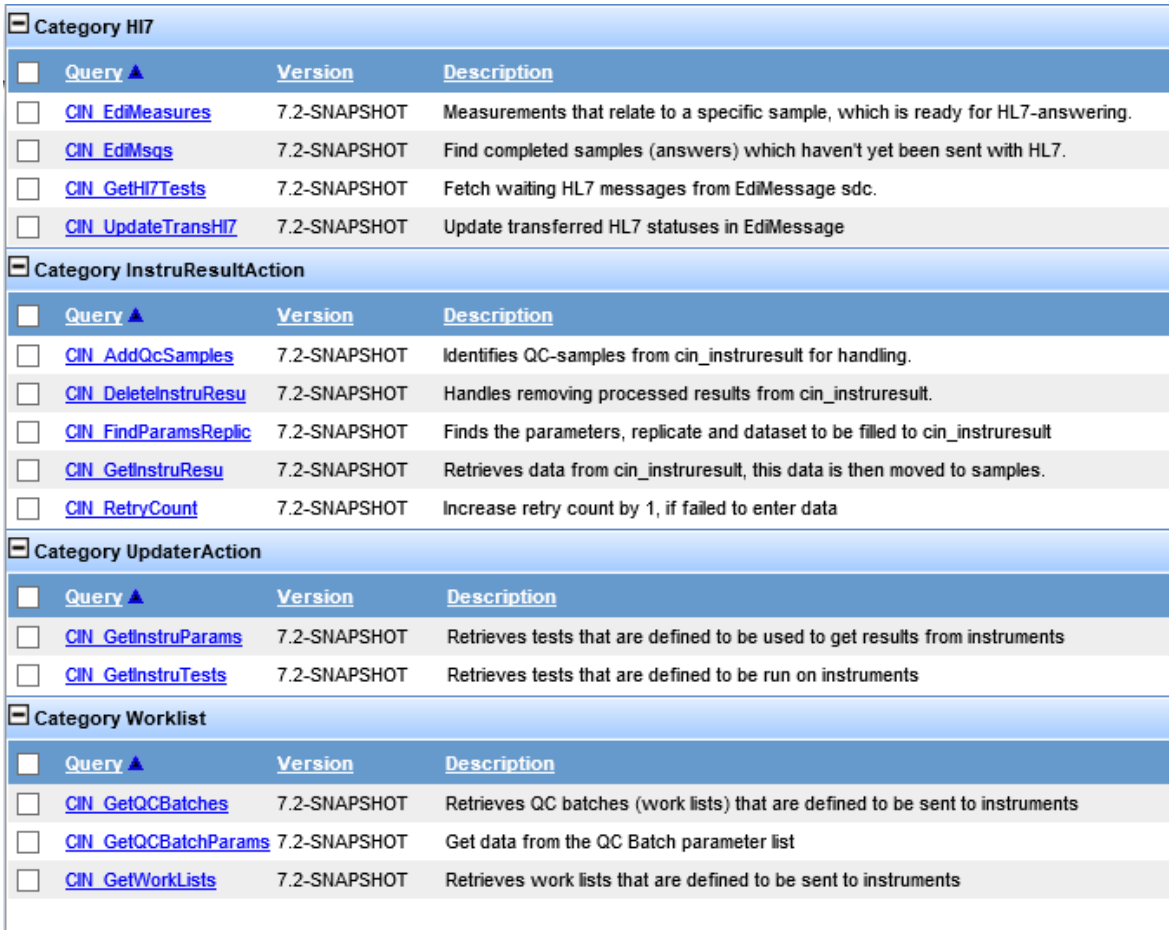
Policy	Example value	Editor type	Description
Log level	Debug	String	LV Connect log level.
Use Direct Sql	Yes	YesNo	Should LV Connect related actions use direct sql or LV actions.
Result group size	1000	String	Size of group that will be handled in one action call.
Run Result Transfer automatically	Yes	YesNo	Runs Result Transfer automatically when new result come from Connect.
Use dilution	No	YesNo	Should Connect related actions use converion factor defined in Instrument Model Mapping.
Result Group Transaction	Yes	YesNo	Transfer each result group in a new transaction.
Enter Audit Reason	No	YesNo	Enter reason to the tracelog-table for every action creating an audit-row.
Specs Passed Values	Pass	String	Values that are interpreted as pass value. Multiple values can be given, use semicolon (;) as a separator. Default value is "Pass". Used when Instrument -> Autorelease -> Only if Specs passed is selected.
Attachment flag	Yes	YesNo	Is Attachment functionality in use.
Attachment subfolder rule	1	String	Controls the creation of attachment subfolders. 0 = No subfolders, 1 = Monthly, 2 = Daily. Ex. C:\Temp\Magix\201306)
Attachment storage method	S	String	Parameter that goes to LV's AddSDIAttachment function. R=Reference, S=Store.
Attachment folder	C:\Temp\ConnectAttachments	String	Folder in local file system to which attachment is written (ex. C:\Temp\ConnectAttachments).
Performance log	C:\Temp\PerformanceLog\	String	Folder in local file system to which performance log is written (ex. C:\Temp\PerformanceLog\). If left empty, no performance log is written. Folder must exist in local file system.
Bulletin users	sysadmin;admin	String	Semicolon separated list of users that will get a bulletin if Connect result transfer fail.
Bulletin roles	Analyst	String	Semicolon separated list of roles that will get a bulletin if Connect result transfer fail.
Send bulletin	Yes	YesNo	Send bulletin if Connect result transfer fail.
Send email	No	YesNo	Send email if Connect result transfer fail.
Email addresses	email@aa.com;address@bb.com	String	Semicolon separated list of email addresses that will get a message if Connect result transfer fail.

Use statistics	Yes	YesNo	Should statistics information for Connect Dashboard be written into cin_statistics table.
Custom action	ExampleCustomAction	String	Name of the custom action that will be called after result transfer.
Custom worklist query	Instrument id: Magix Connect query id: MagixWorklistQuery	PropertyList	Connect query that is used to fetch data for specific instrument instead of CIN_GetWorkLists query.
Custom QCBatch query	Instrument id: Magix Connect query id: MagixQCQuery	PropertyList	Connect query that is used to fetch data for specific instrument instead of CIN_GetQCBatches query.
Namespace	Labvantage	String	Name space of the LIMS system used for example in HL7-interfaces.
Worklist block DataEntry	No	YesNo	Is block flag set to DataSets involved when worklist is created. If set, data can not be manually entered from DataEntry page. Applies also to Connect worklists created from QC Batches.
Units rule	Use instrument unit	List	Use instrument unit means that unit from instrument is added to end of the result making unit conversion possible. Display instrument unit (no unit conversion) means that unit is placed to sdidataitem.displayunits. Empty unit from instrument don't overwrite unit in displayunit. Ignore instrument unit means that unit from instrument is ignored.

5.9 Connect queries

Queries that Connect uses to retrieve data from LABVANTAGE database are defined in the Connect CIN_Query SDC. The default queries will be imported to LABVANTAGE during the Connect interface installation.

The default queries are shown below:



Category HI7

Query	Version	Description
CIN_EdiMeasures	7.2-SNAPSHOT	Measurements that relate to a specific sample, which is ready for HL7-answering.
CIN_EdiMsgs	7.2-SNAPSHOT	Find completed samples (answers) which haven't yet been sent with HL7.
CIN_GetHl7Tests	7.2-SNAPSHOT	Fetch waiting HL7 messages from EdiMessage sdc.
CIN_UpdateTransHl7	7.2-SNAPSHOT	Update transferred HL7 statuses in EdiMessage

Category InstruResultAction

Query	Version	Description
CIN_AddQcSamples	7.2-SNAPSHOT	Identifies QC-samples from cin_instruresult for handling.
CIN_DeleteInstruResu	7.2-SNAPSHOT	Handles removing processed results from cin_instruresult.
CIN_FindParamsReplic	7.2-SNAPSHOT	Finds the parameters, replicate and dataset to be filled to cin_instruresult
CIN_GetInstruResu	7.2-SNAPSHOT	Retrieves data from cin_instruresult, this data is then moved to samples.
CIN_RetryCount	7.2-SNAPSHOT	Increase retry count by 1, if failed to enter data

Category UpdaterAction

Query	Version	Description
CIN_GetInstruParams	7.2-SNAPSHOT	Retrieves tests that are defined to be used to get results from instruments
CIN_GetInstruTests	7.2-SNAPSHOT	Retrieves tests that are defined to be run on instruments

Category Worklist

Query	Version	Description
CIN_GetQCBatches	7.2-SNAPSHOT	Retrieves QC batches (work lists) that are defined to be sent to instruments
CIN_GetQCBatchParams	7.2-SNAPSHOT	Get data from the QC Batch parameter list
CIN_GetWorkLists	7.2-SNAPSHOT	Retrieves work lists that are defined to be sent to instruments

These default queries can be modified as needed for the project. The only restriction is that the SQL statements must return data with the same field names that are used in the default SQL statements.

Queries in UpdaterAction and Worklist categories take one to two input parameters. When modifying these queries, the query parameters must remain the same and in the same order than in the default statements. The moddt returned by the queries in UpdaterAction category will be stored in the Connect side and will be used to limit the rows returned in the subsequent queries.

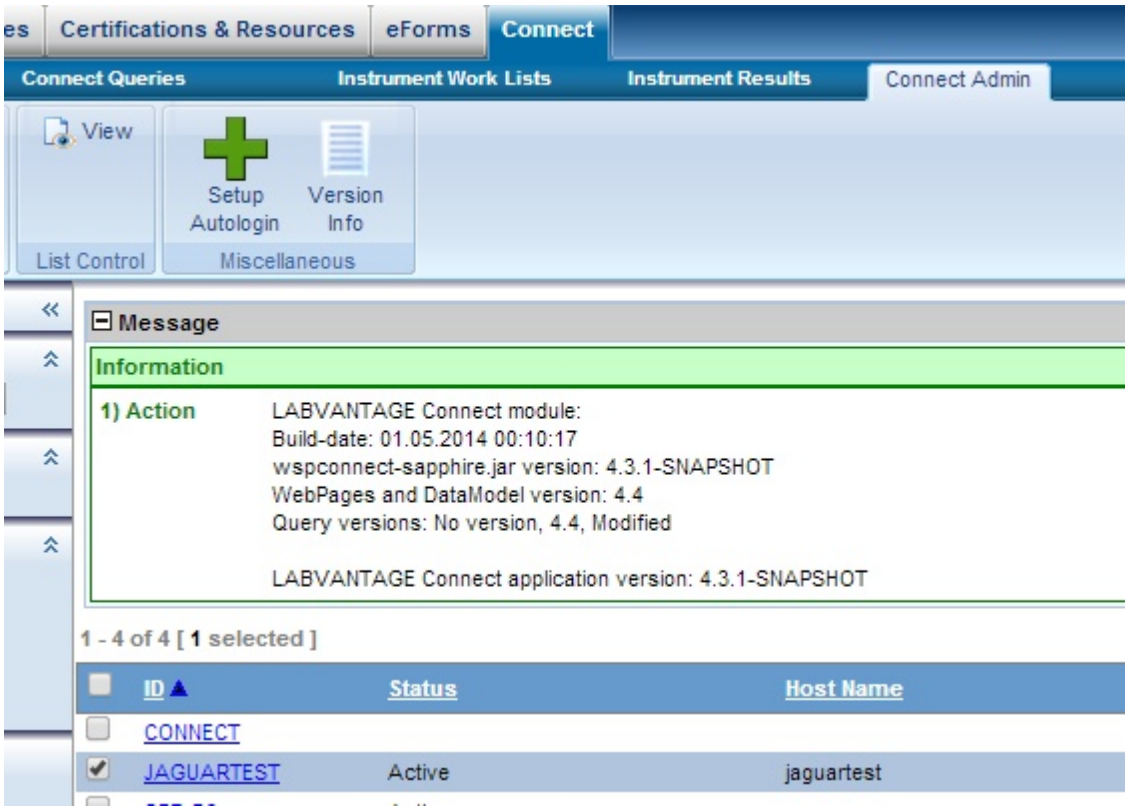
Some of the queries are normally executed only once (like the CIN_GetInstruParams), when the Connect download action (CIN_ConnectAction) is run for the first time (upon starting up the CDF LIMS Connection). To force re-executiong the queries, click the "Refresh"-button. Standard list page operations require that you select at least one row from the list, but the selection doesn't matter as all the queries will be reloaded upon processing the Refresh request.

5.10 Connect version information

You can find out the version of all the Connect-related components from the Connect Admin -page in LABVANTAGE. Select a Connect-installation, and press "Version Info".

This shows up the version information of
- wspconnect-sapphire
- web-pages and datamodel of the Connect module
- Connect queries
- Connect application

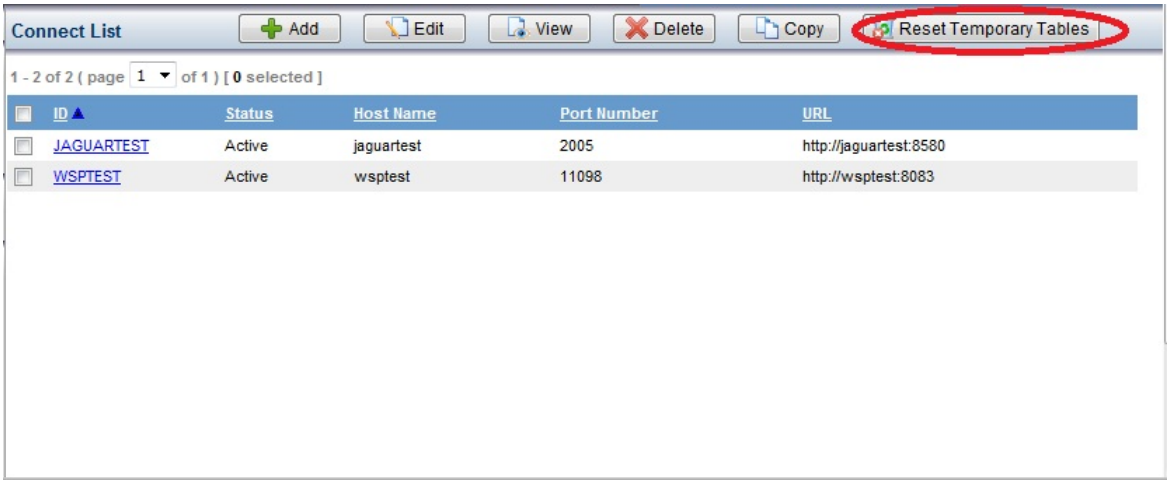
This information can be very useful when solving Connect related problems, especially if Connect has been updated.



5.11 Resetting temporary tables

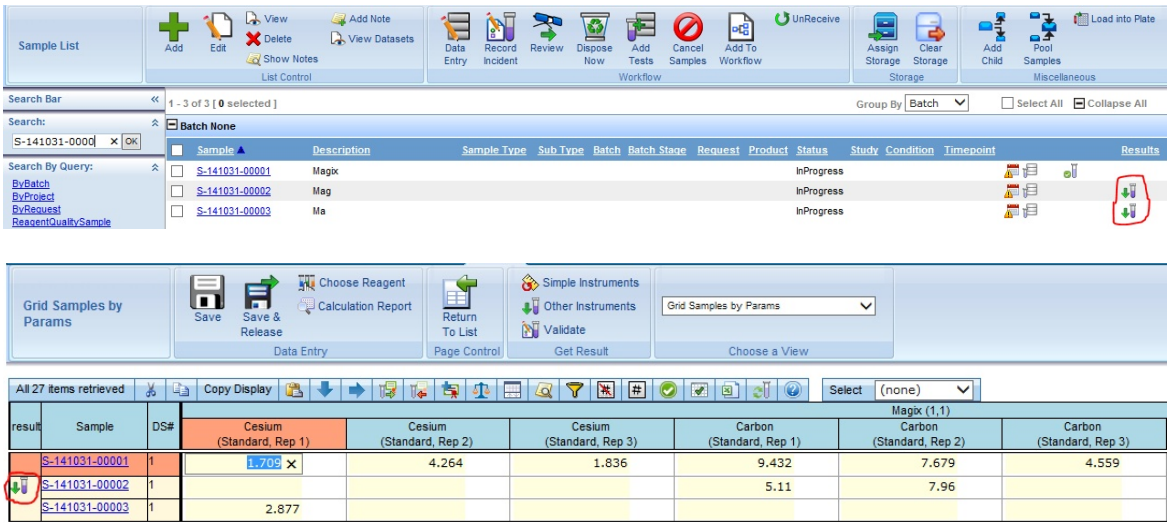
Some definition data (pending tests, instrument parameter definitions) are downloaded from the LABVANTAGE system to Connect and saved into the temporary database.

On some rare error situations the solution is to reset the temporary database. You can execute this function directly from the Connect definitions list page by clicking the "Reset Temporary Tables" button. This is a safe operation and will only cause that Connect will re-download all necessary definitions from LABVANTAGE and populate the temporary database accordingly.



5.12 Unvalidated Results -icon

When running instruments with autovalidation off, the results will stay in the Instrument Result List page (InstruResult -table) until they have been validated. To make it easier to notice when Instrument Result List page contains unvalidated results for a sample, Connect comes with unvalidated result icon on Sample List and Data Entry page. Icon means that there are unhandled results for sample. In other words, there are rows related to this sample in the Instrument Result List page, which state is other than "Saved".



5.13 Connect Dashboard

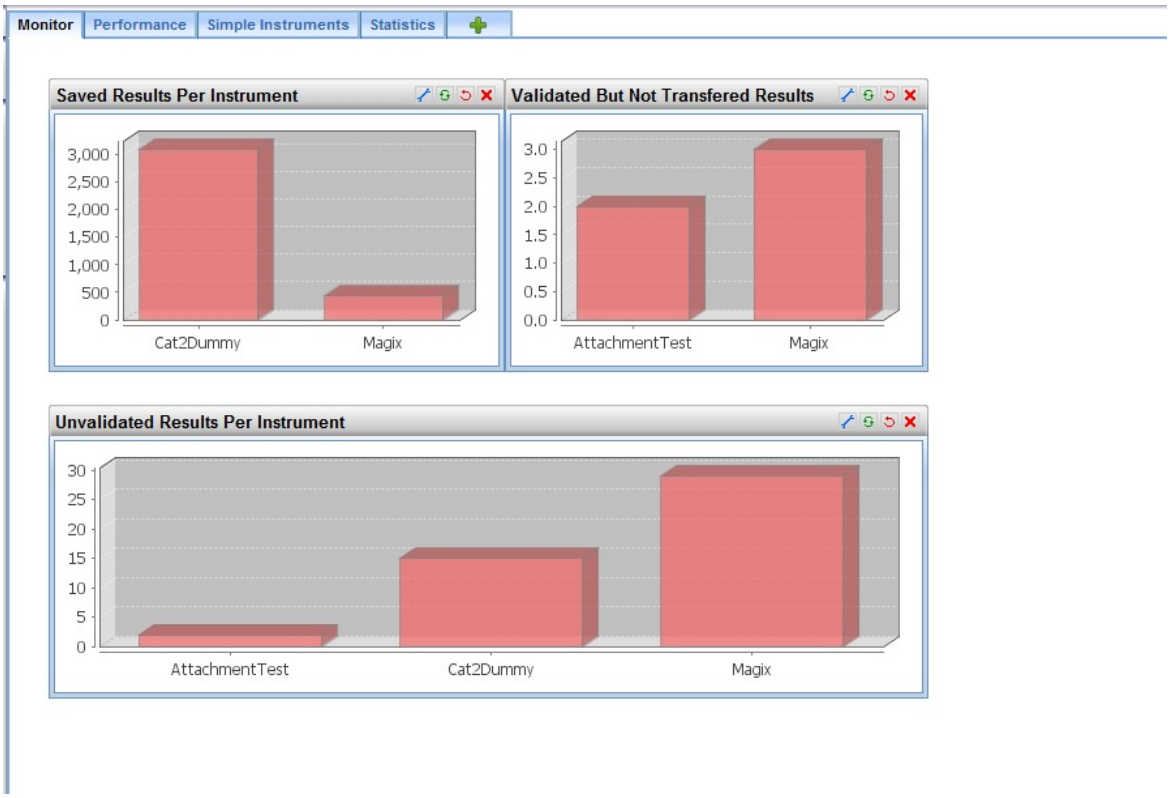
Connect dashboard gives you an overview of amount of results transferred from instruments, performance of result transfer and status of result transfer. Category 1 instrument (also called "simple instrument") have their own tab called Simple Instruments. Data in Simple Instruments tab is stored in cin_statistics table so it is permanent. Data in other tabs is data from category 2-4 instruments. Data in Performance tab is calculated

when result are transferred and stored in cin_statistics table so it is permanent. Data in Monitor and Statistics tab is from cin_instruresult table, which is emptied after sometime (defined in in Connect Queries, CIN_DeleteInstruResu query). Other tabs but Monitor tab has parameter fields to restrict data fetched to charts.

Statistics in Connect Dashboard is based on basic set of charts, new statistics based on customer needs can be done.

5.13.1 Monitor Dashboard

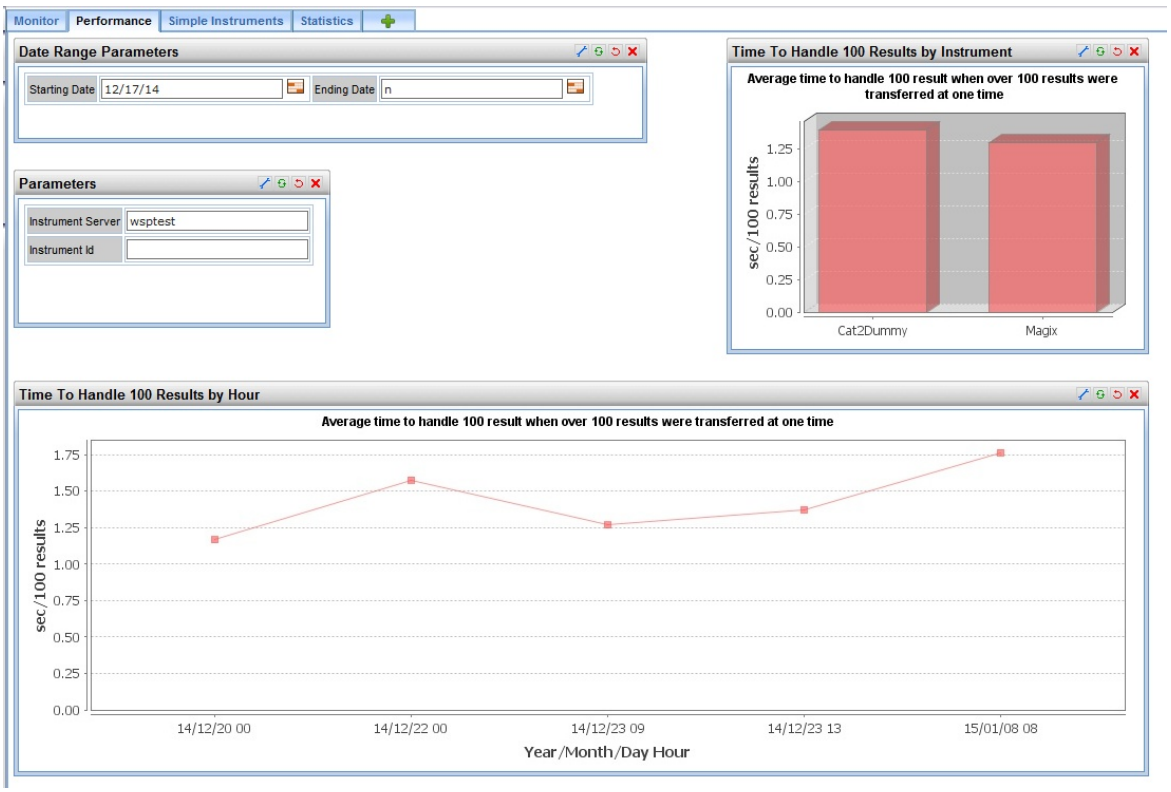
Monitor Dashboard gives user information about status of result transfer. Unvalidated Results Per Instrument chart tells a user that there is unhandled results in Instrument Result List page that should be taken care of. Validated But Not Transferred Results indicates that there has been an attempt to transfer results, but results have not been saved to sample. In most cases something is wrong with configuration and Instrument Result List pages Debug Transfer button will tell where the problem is. Saved Results Per Instrument chart tell an amount of successfully transferred result per instrument.



5.13.2 Performance Dashboard

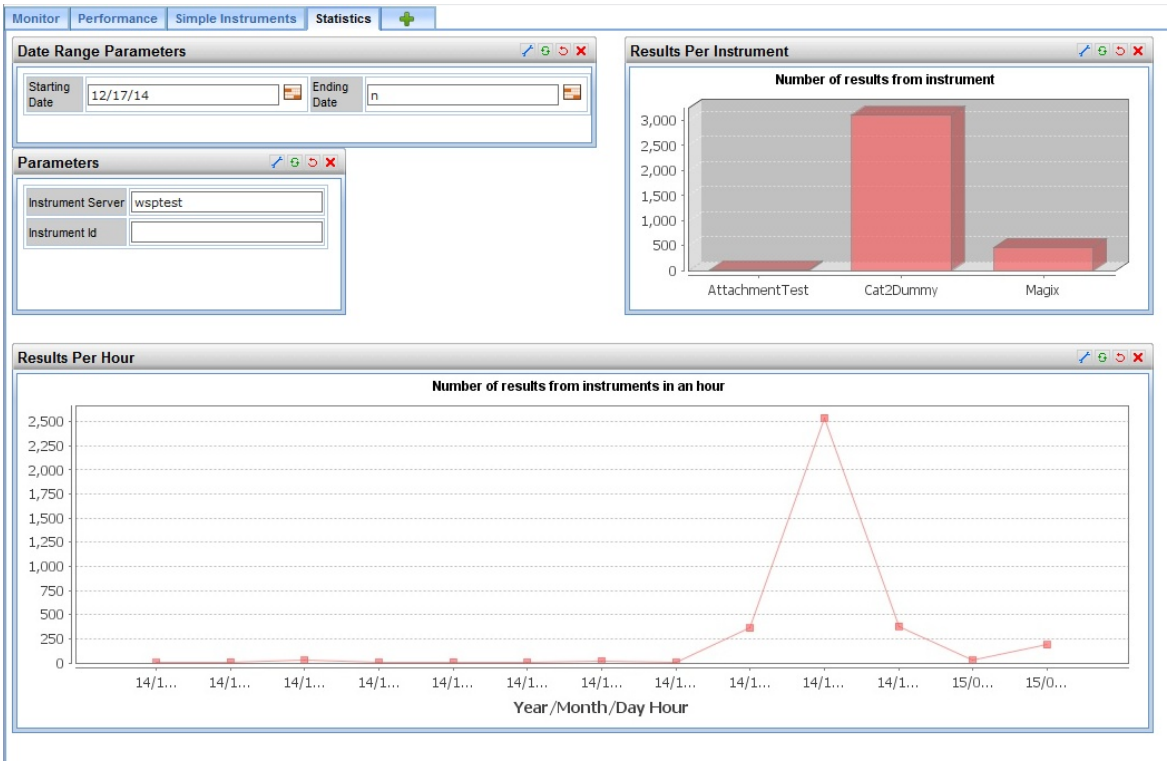
Performance Dashboard gives information about how fast Connect result transfer handles saving results. Time is measured from point that result has been transferred from Connect to LABVANTAGE (result are in cin_instruresult table) to point where result have been saved to sample. Charts tell handling time per 100 result by instrument and by hour of the day. Performance is measured only in cases where more that 100 result where transferred at one time. This is because handling less that 100 result will only take a maximum of few

seconds. If time to handle 100 results grows to tens of seconds, there usually is custom project code that is triggered for example when saving the result that causes the process to slow down. As a rule of thumb, it takes 1-4 seconds to handle of 100 results in basic set up.



5.13.3 Statistic Dashboard

Statistic Dashboard tells how many results has been transferred from Connect to LABVANTAGE by instrument and by hour of the day.



5.13.4 Simple Instrument Dashboard

Simple Instrument Dashboard shows category 1 instrument statistics. Charts give an overview of how many result have been fetched from instruments and how long it has taken.