



# SampleManager LIMS Web Interface Configuration Guide

**Version 12.2**

© 2019 Thermo Fisher Scientific Inc. All rights reserved.

Thermo Fisher Scientific Inc. provides this document to its customers with a product purchase to use in the product operation. This document is copyright protected and any reproduction of the whole or any part of this document is strictly prohibited, except with the written authorization of Thermo Fisher Scientific Inc.

The contents of this document are subject to change without notice. All technical information in this document is for reference purposes only. System configurations and specifications in this document supersede all previous information received by the purchaser.

**Thermo Fisher Scientific Inc. makes no representations that this document is complete, accurate, or error-free and assumes no responsibility and will not be liable for any errors, omissions, damage, or loss that might result from any use of this document, even if the information in the document is followed properly.**

This document is not part of any sales contract between Thermo Fisher Scientific Inc. and a purchaser. This document shall in no way govern or modify any Terms and Conditions of Sale, which Terms and Conditions of Sale shall govern all conflicting information between the two documents.

All trademarks recognized.

# Contents

|                                                                           |                |
|---------------------------------------------------------------------------|----------------|
| <b>Preface .....</b>                                                      | <b>v</b>       |
| Related Documentation .....                                               | v              |
| Special Notices.....                                                      | vi             |
| Contacting Us.....                                                        | vi             |
| <br><b>Chapter 1    Web Interface Architecture .....</b>                  | <br><b>1</b>   |
| SampleManager Web Application .....                                       | 2              |
| WCF Service.....                                                          | 3              |
| <br><b>Chapter 2    Standard LIMSML Entity Actions .....</b>              | <br><b>9</b>   |
| LIMSML Entity Actions .....                                               | 10             |
| LIMSML Entity Actions for Dynamic Data Entities .....                     | 17             |
| LIMSML Entity Actions for Static Data Entities.....                       | 45             |
| Other LIMSML Entity Actions.....                                          | 64             |
| <br><b>Chapter 3    Result Entry with LIMSML .....</b>                    | <br><b>133</b> |
| Sample Identification and Management .....                                | 133            |
| Test Identification and Management .....                                  | 140            |
| Result Entry.....                                                         | 150            |
| Updating Sample and Test Information .....                                | 160            |
| <br><b>Chapter 4    LIMSML.....</b>                                       | <br><b>167</b> |
| LIMSML Hierarchy.....                                                     | 168            |
| Element and Attribute Definitions .....                                   | 169            |
| <br><b>Chapter 5    Installing the SampleManager Web Application.....</b> | <br><b>197</b> |
| Minimum System Requirements.....                                          | 198            |
| Overall Installation Procedure.....                                       | 198            |
| Upgrading the SampleManager Web Application .....                         | 205            |
| <br><b>Chapter 6    The SampleManager Web Site.....</b>                   | <br><b>207</b> |
| Defining the Configuration Settings .....                                 | 208            |
| Styles .....                                                              | 209            |
| Allowing LDAP Authentication .....                                        | 214            |
| <br><b>Chapter 7    Defining Web Interface Menu Options.....</b>          | <br><b>215</b> |

|                  |                                                       |            |
|------------------|-------------------------------------------------------|------------|
|                  | Adding a Folder to the Web Site Menu.....             | 216        |
|                  | Adding an InfoMaker Report to the Web Site Menu ..... | 223        |
|                  | Adding a Link to the Web Site Menu .....              | 223        |
|                  | LIMSML Entity Action Tester.....                      | 225        |
| <b>Chapter 8</b> | <b>Localization Editor .....</b>                      | <b>229</b> |
|                  | Definitions.....                                      | 230        |
|                  | Using the Localization Editor.....                    | 231        |
|                  | Adding a New Language .....                           | 233        |
|                  | Adding a Message .....                                | 233        |
|                  | Adding a Message Group .....                          | 234        |
|                  | Editing Messages.....                                 | 235        |
| <b>Chapter 9</b> | <b>Electronic Signatures.....</b>                     | <b>237</b> |
|                  | Configuring Electronic Signatures.....                | 238        |
|                  | Esigs for Web Interface Menu Options .....            | 238        |
|                  | Esigs for New Data from the Web Interface.....        | 239        |
|                  | Failed Esigs .....                                    | 239        |
|                  | Viewing ESig Information .....                        | 240        |
|                  | <b>Index.....</b>                                     | <b>241</b> |

## Preface

Thermo Scientific™ SampleManager LIMS™ software can plan, manage, and maintain an accurate record of work performed in a laboratory environment. The SampleManager LIMS software incorporates Laboratory Information Management (LIMS), Data Management (SDMS), and Process Execution and Procedural Electronic Lab Notebook (LES) capabilities in a single solution.

This guide describes the SampleManager WCF service, the LIMSML schema used to transfer data to and from SampleManager, and the SampleManager Web application.

### Contents

- [Related Documentation](#)
- [Special Notices](#)
- [Contacting Us](#)

## Related Documentation

SampleManager includes Help, manuals, and guides as PDF files.

### ❖ To view the SampleManager product manuals

Navigate to the Docs folder on the SampleManager installation media.

### ❖ To open the Help

With the SampleManager client running, you can access Help in the following ways:

- Press the F1 key.
- From the main menu bar at the top of the SampleManager window, choose Help and then click Contents.
- Click a Help button.

## Special Notices

Make sure that you follow the precautionary statements presented in this guide. The special notices appear in boxes.

Special notices include the following:

**CAUTION** Highlights information necessary to prevent damage to software, loss of data, invalid test results, or information that is critical for optimal performance of the system.

**Note** Highlights information of general interest.

**Tip** Highlights helpful information that can make a task easier.

## Contacting Us

There are several ways to contact Thermo Fisher Scientific for the information you need.

### ❖ To contact Technical Support

Email [support.informatics@thermofisher.com](mailto:support.informatics@thermofisher.com)

Web <https://support.thermoinformatics.com/contacts>

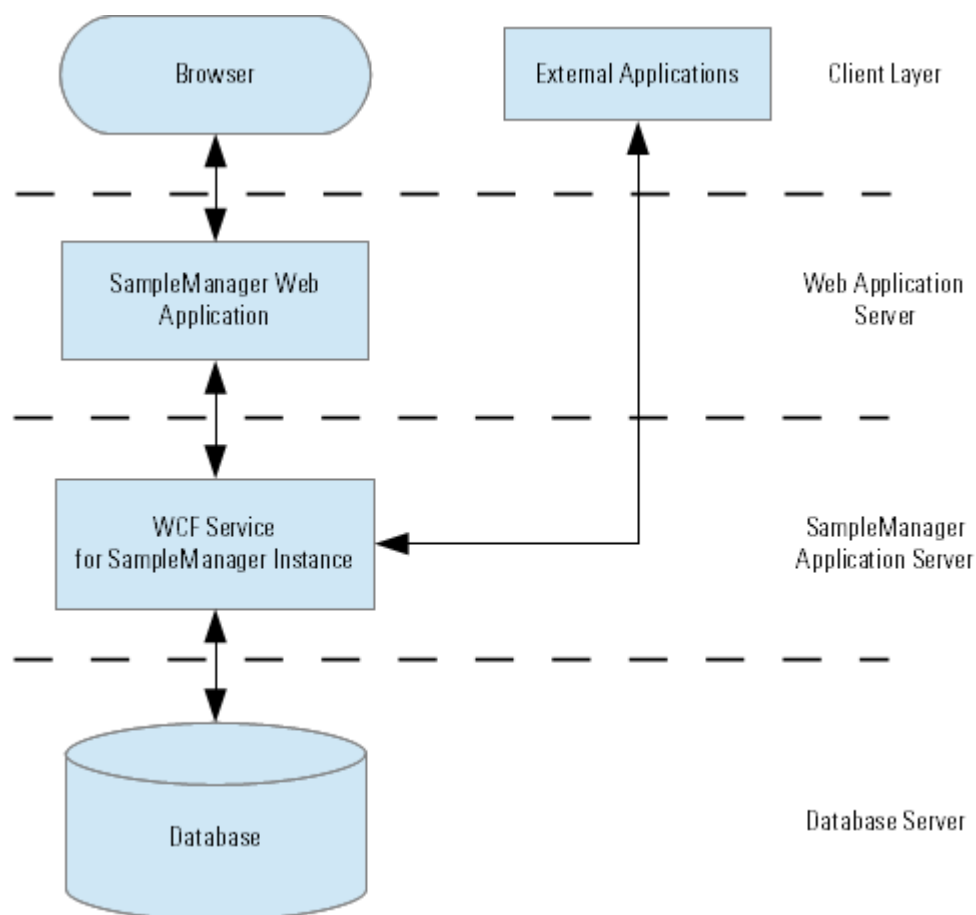
### ❖ To suggest changes or provide feedback on the documentation

Send an email to the Informatics Documentation Manager:  
[sidinf.documentation@thermofisher.com](mailto:sidinf.documentation@thermofisher.com).

## Web Interface Architecture

The SampleManager web interface uses several components, as shown in the following figure:

**Figure 1-1.** SampleManager Web Interface – Major Components



The SampleManager Web Application is a legacy feature that provides access to a limited subset of SampleManager functions through the associated SampleManager web site.

For information about the new SampleManager web client, see the *SampleManager LIMS Web Client Installation and Configuration Guide*.

#### **Contents**

- [SampleManager Web Application](#)
- [WCF Service](#)

## **SampleManager Web Application**

The SampleManager web application is configured through SampleManager, rather than being coded as ASP or ASP.NET pages. The web pages rendered to the client do not require any client-side components, but do use client-side scripting.

The interface uses several third-party controls, such as the navigation panel on the left side of each web page. It also uses the following components, installed on the SampleManager web server:

- LIMSML Helper
- Internationalization Helper
- Error handler

### **LIMSML Helper**

LIMSML Helper is a .NET component, which is used to construct and decode LIMSML documents.

### **Internationalization Helper**

Internationalization Helper obtains the translation, in the required language, for any labels used in the web application. Translations are stored in an internationalization file. All components are written to use message identities, rather than local text, to allow internationalization of the web pages.

### **Error Handler**

Error Handler is used to output all error information in a consistent manner.

## WCF Service

The SampleManager Add Instance Wizard creates a WCF service for each SampleManager instance. Each WCF service has two interfaces:

- LIMSML interface: Handles transactions encoded in [LIMSML](#) (page 167), which is an XML schema used to transfer data between the SampleManager database and external applications, such as the SampleManager website.
- REST interface: Handles transactions using REST (representational state transfer), which is a software architectural style used for simple client-server interfaces. SampleManager Mobile uses the REST interface.

By default, the WCF service uses port 56104 for its LIMSML interface and port 56105 for its REST interface. You can change these ports by editing the SampleManagerWCFWindowsService.exe.config file in the Exe folder of your instance.

If you make significant changes to the configuration of your SampleManager instance, such as modifications to database structure, restart the WCF service.

**Note** SampleManager versions before SampleManager 11.0 used a different architecture, with a LIMSML Web service and COM+ interface in place of the WCF service. The WCF service implements the same ILIMSWebService interface as the LIMSML Web service, so any external applications developed to communicate with the LIMSML Web service will work in the same way with the WCF service.

## Testing the WCF Service

You can use the Test SampleManager Service utility to test the LIMSML interface of the WCF service associated with a SampleManager instance.

### ❖ To run the Test SampleManager Service utility

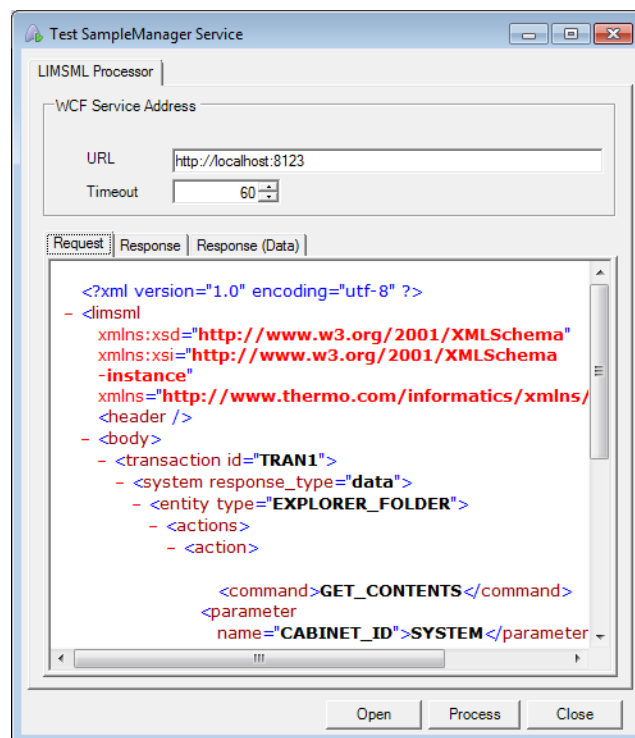
1. In Windows Explorer, navigate to the Exe folder for the SampleManager instance.
2. Double-click the **TestRemote.exe** file.

The Test SampleManager Service window appears.

3. Verify that the URL field shows the correct port number for the LIMSML interface. The default port is 56104.

- Click **Open** and select a LIMSML file to use for the test. A set of example files is supplied with SampleManager. The contents of the chosen file appear in the Request page.

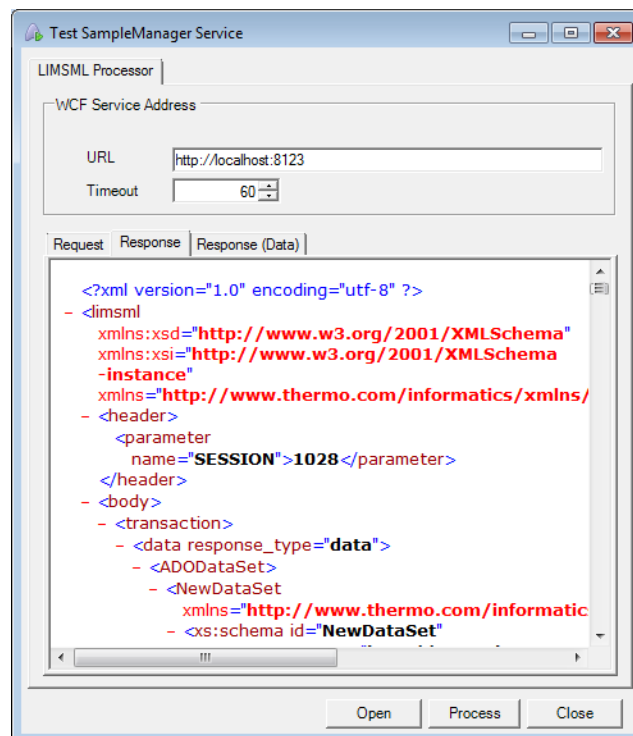
**Figure 1-2.** Test SampleManager Service Window showing Request Page



- Click **Process** to send the request to the WCF service.  
The SampleManager Login dialog box appears.
- Enter a valid SampleManager username and password.

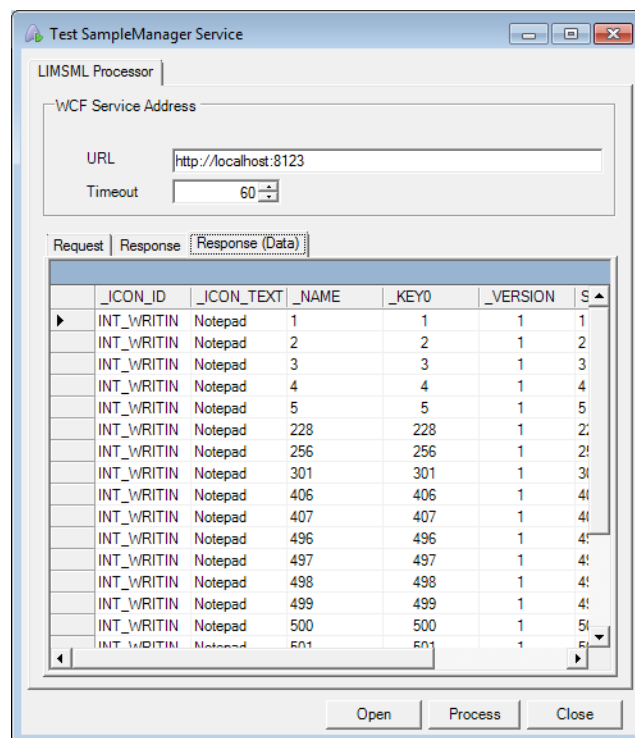
7. Click **OK**. When the request has been processed, a Complete message appears and the LIMSML response is shown in the Response page.

**Figure 1-3.** Test SampleManager Service Window showing Response Page



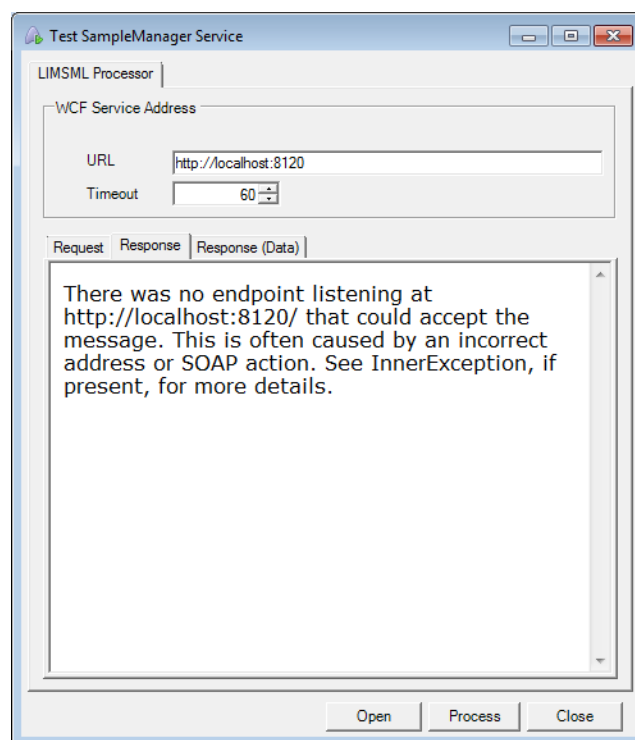
If the LIMSML response <data> element has the response type="data", the dataset returned in the response is shown in the Response (Data) page.

**Figure 1-4.** Test SampleManager Service Window showing Response (Data) Page



If an error occurs when the request is sent to the WCF service, an Errors Occurred message appears and the error text is shown on the Response page.

**Figure 1-5.** Test SampleManager Service Window showing Response Page with Error





## Standard LIMSML Entity Actions

This chapter describes LIMSML entity actions. A standard SampleManager installation is supplied with a set of standard LIMSML entity actions for performing common tasks.

Each LIMSML entity action has a corresponding master menu option. The master menu option calls a VGL routine that performs the required operation within the SampleManager system.

The following LIMSML entity actions are particularly useful for transferring data between SampleManager and another software product, such as instrument control software:

- [Job Create By Template](#) (page 17)
- [Sample Create By Template](#) (page 23)
- [Sample Add Test](#) (page 21)
- [Sample Get Component Limits](#) (page 28)
- [Sample Get Results](#) (page 30)
- [Sample Result Entry](#) (page 39)

**Note** Many of the LIMSML entity actions are used by [the SampleManager Web Site](#) (page 207) to support functionality such as explorer folder access and browsing for data. These LIMSML entity actions are unlikely to be useful in other contexts.

### Contents

- [LIMSML Entity Actions](#)
- [LIMSML Entity Actions for Dynamic Data Entities](#)
- [LIMSML Entity Actions for Static Data Entities](#)
- [Other LIMSML Entity Actions](#)

## LIMSML Entity Actions

A LIMSML entity action describes a SampleManager function that can be called from the web. The LIMSML entity action defines the routine that will be used to handle the functionality.

To create a LIMSML entity action, choose **Setup > Interface > LIMSML Action > Add** from the SampleManager main menu. The LIMSML Entity Action Data Entry screen appears.

### LIMSML Entity Action – General Page

The General page shows basic information about the LIMSML entity action.

**Figure 2-1.** LIMSML Entity Action – General Page

The screenshot shows a Windows-style dialog box titled "LIMSML Entity Action - Modify - SAMPLE CREATE\_BY\_TEMPLATE". It has two tabs: "General" and "Parameters". The "General" tab is selected. The form contains the following fields:

- Entity:** A dropdown menu showing "CREATE\_BY\_TEMPLATE".
- Name:** A text box containing "Create sample with template".
- Entity:** A dropdown menu showing "SAMPLE".
- Description:** A text box containing "Creates a new sample using the specified template".
- Return Type:** A dropdown menu showing "SYSTEM".
- Group Id:** A dropdown menu (empty).
- Modified On:** A text box containing "18/09/2009 13:38".
- Modified By:** A dropdown menu showing "BATCH".
- Modifiable:** A checkbox that is checked.

At the bottom right are three buttons: "OK", "Cancel", and "Apply".

| Option | Description                                                                                                                                                                                                                                                                                                                          |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Name   | The name of the LIMSML entity action. The name is human-readable text that gives operators more useful information than the record identity. The name appears in prompts that link to the record and in browse lists. The name can be changed after the record has been saved, and must be unique in the LIMSML_ENTITY_ACTION table. |

| Option      | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Entity      | <p>The SampleManager entity that the functionality applies to.</p> <p>This corresponds to the value for the type attribute of the entity element in a LIMSML request.</p> <p>The following options are available:</p> <ul style="list-style-type: none"> <li>• The name of the entity type that the functionality applies to, for example SAMPLE or INSTRUMENT.</li> <li>• GENERIC: The GENERIC option is used if the functionality can apply to more than one entity type. For example, the <a href="#">Generic Browse LIMSML entity action</a> (page 80) supplied with SampleManager can browse on any entity type.</li> <li>• SYSTEM: The SYSTEM option is used for functionality that is related to general operation of SampleManager. For example, the <a href="#">System Get File LIMSML entity action</a> (page 121) supplied with SampleManager returns an encoded copy of a file.</li> </ul> |
| Description | Description of the LIMSML entity action.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Return Type | <p>The type of data that will be returned by the action. This can be:</p> <ul style="list-style-type: none"> <li>• DATA: the action will return a data element</li> <li>• SYSTEM: the action will return a system element</li> <li>• BOTH: the action will return both system and data elements</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Action      | <p>The action to be performed on the entity.</p> <p>This corresponds to the value of the command element in a LIMSML request.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

| Option      | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Group Id    | <p>If a group is specified, only operators who are members of that group can access the record.</p> <p>If no group is specified, any operator can access the record.</p> <p>The behavior of this field is controlled by the following configuration items:</p> <ul style="list-style-type: none"> <li>GROUP_MANDATORY – If the value is Yes, the field cannot be blank when you attempt to save the record. Users with privilege 9042 (Allow the user to override the Mandatory group check) can save a record with a blank group field.</li> <li>GROUP_SET_USERS_DEFAULT – If the value is Yes, when you create a record using the Add option, the default value of the group field is the Default Group specified on the Definition page of your Personnel record.</li> </ul> <p>If the value is No, anyone can save a record with a blank group field.</p> <p>If the value is No, the default value is blank.</p> <p>For more information, see the "Security" chapter in the <i>SampleManager LIMS System Administration Guide</i>.</p> |
| Modified On | Date the record was last modified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Modified By | Operator who performed the last modification.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Modifiable  | <p>If selected, any user with the necessary role and group access can modify the record.</p> <p>If cleared, only a user with system privilege 9002 (Allow access to data marked as non-modifiable) can modify the record.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

## LIMSML Entity Action – Parameters Page

The Parameters page lists the parameters required by the LIMSML entity action.

**Figure 2-2.** LIMSML Entity Action – Parameters Page

| Parameter              | Description                    | Mandatory                           |
|------------------------|--------------------------------|-------------------------------------|
| ▶ IGNORE_TEST_SCHEDULE | Should Test schedule tests ... | <input checked="" type="checkbox"/> |
| JOB_NAME               | Job name                       | <input type="checkbox"/>            |
| REPLICATES             | Number of replicates to cre... | <input type="checkbox"/>            |
| TEMPLATE               | Sample Template identity       | <input type="checkbox"/>            |

| Option      | Description                                                                                    |
|-------------|------------------------------------------------------------------------------------------------|
| Parameter   | The name of the parameter                                                                      |
| Description | A description of the parameter                                                                 |
| Mandatory   | If the Mandatory check box is selected, the parameter must be provided for the routine to run. |

## Testing LIMSML Entity Actions

The \$LIMSML\_TEST VGL report can be used to test LIMSML entity actions within SampleManager.

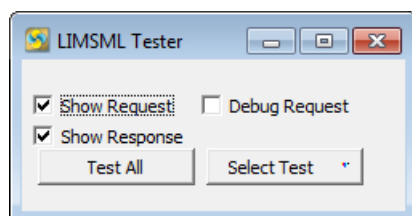
### ❖ To test a LIMSML entity action

1. Create a LIMSML file containing a request that specifies the LIMSML entity action you want to test.

**Note** As an alternative, you can create a VGL routine that constructs the LIMSML, and add it as an option in the menu opened by the Select Test button. See the code in the \$LIMSML\_TEST report for examples.

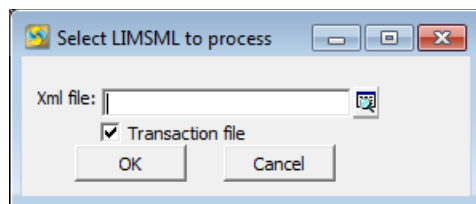
2. In Explorer, right-click the \$LIMSML\_TEST VGL report and choose **Run**. The LIMSML Tester dialog box appears.

**Figure 2-3.** LIMSML Tester Dialog Box



3. Select the **Show Request** check box.
4. Select the **Show Response** check box.
5. Click **Select Test**. A menu appears, containing entries for all the standard LIMSML entity actions.
6. Choose **Browse for xml file**, the first item in the menu. The Select LIMSML to Process dialog box appears.

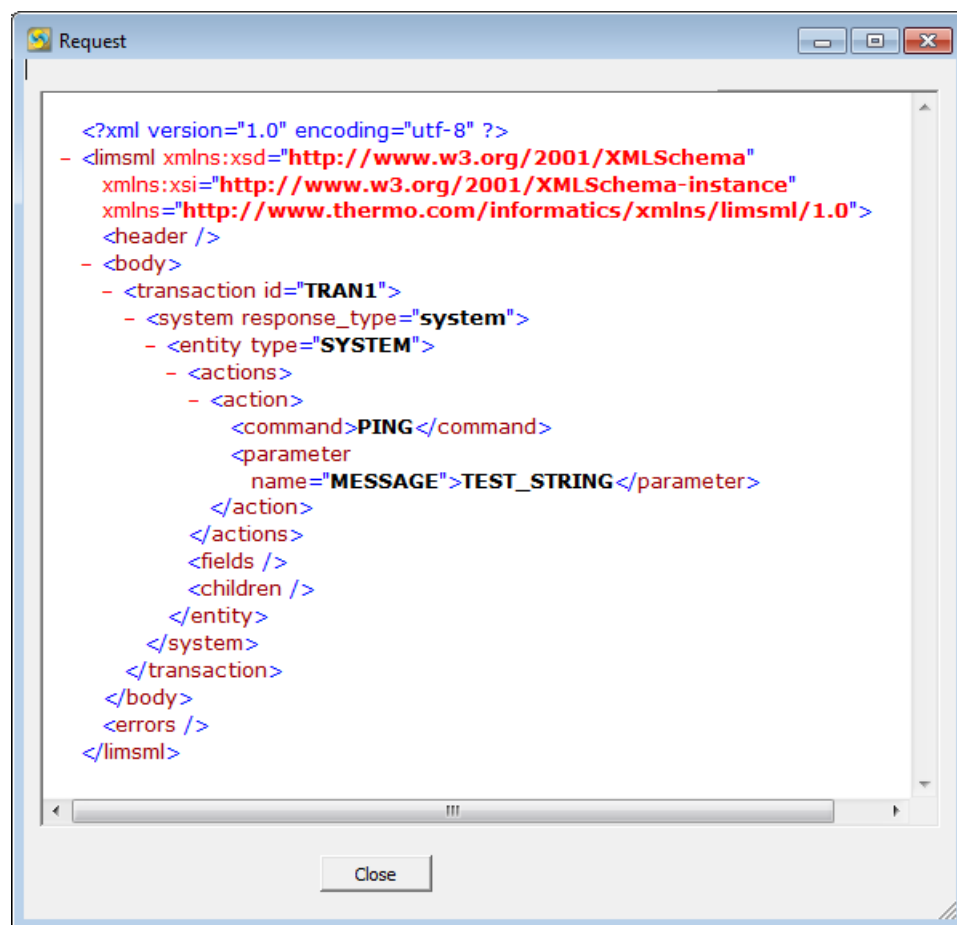
**Figure 2-4.** Select LIMSML to Process Dialog Box



7. Select the LIMSML file.
8. If the LIMSML file contains a transaction element, select the **Transaction file** check box.
9. Click **OK**.

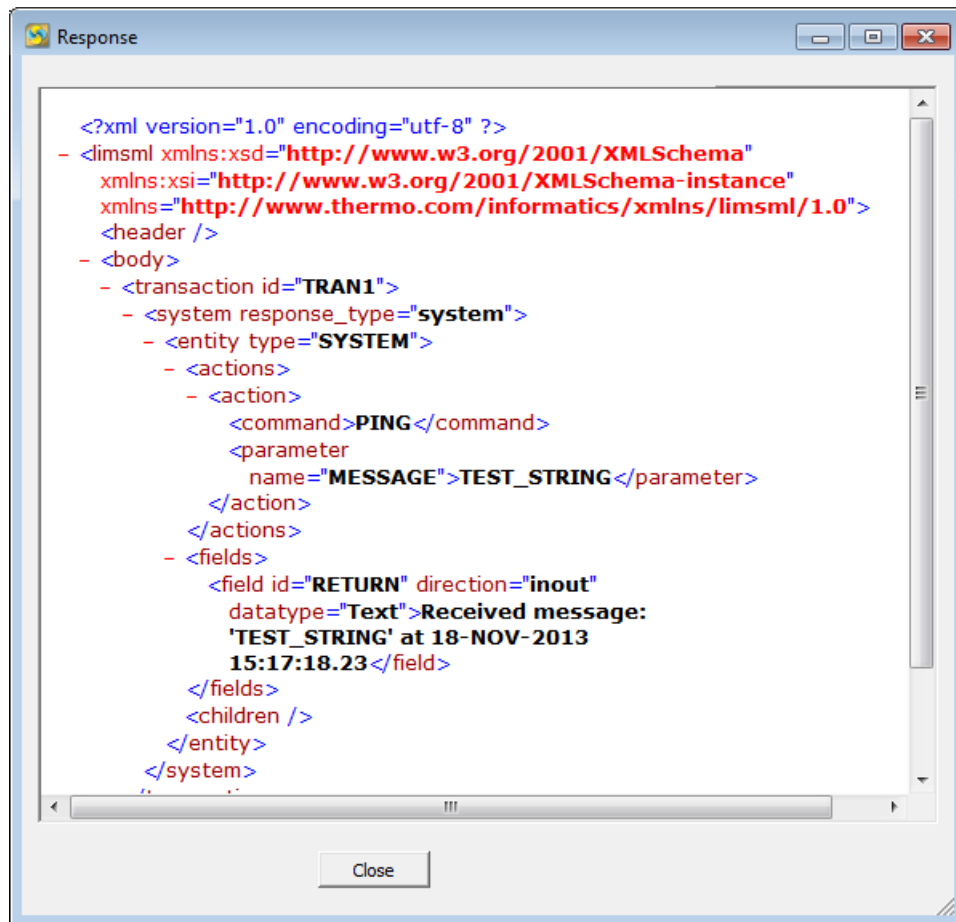
10. The Request screen appears, showing the content of your file.

**Figure 2-5.** Request Screen



11. Click **Close**. The Response screen appears, showing the LIMSML response generated by SampleManager when it processed the request.

**Figure 2-6.** Response Screen



## LIMSML Entity Actions for Dynamic Data Entities

This section describes the standard LIMSML entity actions for creating and managing dynamic data entities.

### Job Create By Template

Entity: JOB

Action: CREATE\_BY\_TEMPLATE

Master Menu Procedure Number: 15594

VGL Library: \$LIMSML\_JOB

Routine: job\_create\_by\_template

Job Create By Template logs in a new job using the specified template.

The following sections describe the parameters for Job Create By Template.

#### JOB\_NAME

The name of the new job.

This parameter is optional.

#### TEMPLATE

The job template to use to log in the new job.

This parameter is optional.

## Example

The following LIMSML request uses the CREATE\_BY\_TEMPLATE command to log in a new job using the DEFAULT job template.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB">
      <actions>
        <action>
          <command>CREATE_BY_TEMPLATE</command>
          <parameter name="TEMPLATE">DEFAULT</parameter>
        </action>
      </actions>
      <fields>
        <field id="JOB_NAME" direction="out" datatype="Text" />
        <field id="BROWSE_DESCRIPTION" direction="inout" datatype="Text">New job create from
LIMSML</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB">
      <actions>
        <action>
          <command>CREATE_BY_TEMPLATE</command>
          <parameter name="TEMPLATE">DEFAULT</parameter>
        </action>
      </actions>
      <fields>
        <field id="JOB_NAME" direction="out" datatype="Text">DEFAULT . 137</field>
        <field id="BROWSE_DESCRIPTION" direction="inout" datatype="Text">New job create from
LIMSML</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Result Get Audits

Entity: RESULT

Action: GET\_AUDITS

Master Menu Procedure Number: 15637

VGL Library: \$LIMSML\_RESULT

Routine: result\_get\_audits

Result Get Audits returns audit data for a specific result.

The following sections describe the parameters for Result Get Audits.

### COMPONENT\_NAME

The component name associated with the result.

This parameter is mandatory.

### TEST\_ID

The test number associated with the result.

This parameter is mandatory.

### Example

The following LIMSML request uses the GET\_AUDITS command to obtain the audits recorded against the Cadmium result for test number 13940.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="RESULT">
      <actions>
        <action>
          <command>GET_AUDITS</command>
          <parameter name="TEST_ID">13940</parameter>
          <parameter name="COMPONENT_NAME">Cadmium</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="AUDITS">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="Date" type="xs:dateTime" minOccurs="0" />
                      <xs:element name="who" type="xs:string" minOccurs="0" />
                      <xs:element name="Previous_x0020_Value" type="xs:string"
minOccurs="0" />
                      <xs:element name="Audit_x0020_Reason" type="xs:string" minOccurs="0"
/>
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
        <AUDITS>
          <who>SYSTEM</who>
          <Previous_x0020_Value>11</Previous_x0020_Value>
          <Audit_x0020_Reason>Typographic error.</Audit_x0020_Reason>
        </AUDITS>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

## Sample Add Test

Entity: SAMPLE

Action: ADD\_TEST

Master Menu Procedure Number: 15641

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_add\_test

Sample Add Test adds a test to a sample.

Sample Add Test has no parameters.

## Example

The following LIMSML request uses the ADD\_TEST command to add a test based on the PH analysis to sample 18199.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>ADD_TEST</command>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="in" datatype="Text">18199</field>
      </fields>
      <children>
        <entity type="TEST">
          <actions />
          <fields>
            <field id="ANALYSIS" direction="inout" datatype="Text">PH</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

## 2 Standard LIMSMML Entity Actions

### LIMSMML Entity Actions for Dynamic Data Entities

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/lmsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>ADD_TEST</command>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="in" datatype="Text">      18199</field>
      </fields>
      <children>
        <entity type="TEST">
          <actions />
          <fields>
            <field id="ANALYSIS" direction="inout" datatype="Text">PH</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

## Sample Create By Template

Entity: SAMPLE

Action: CREATE\_BY\_TEMPLATE

Master Menu Procedure Number: 15567

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_create\_by\_template

Sample Create By Template logs in a new sample using the specified sample template.

The following sections describe the parameters for Sample Create By Template.

### IGNORE\_TEST\_SCHEDULE

If IGNORE\_TEST\_SCHEDULE is TRUE, a test schedule specified in the sample template will not be used to assign tests when the sample is logged in. If IGNORE\_TEST\_SCHEDULE is FALSE, tests included in a test schedule specified in the sample template will be assigned to the sample at login.

If IGNORE\_TEST\_SCHEDULE is not specified, tests included in a test schedule specified in the sample template will be assigned to the sample at login.

This parameter is optional.

### JOB\_NAME

The name of the job to log the sample into. If no JOB\_NAME is specified, the sample is not logged into a parent job.

This parameter is optional.

### REPLICATES

The number of replicate samples to create. If the REPLICATES parameter is not specified, 1 sample is created.

This parameter is optional.

### TEMPLATE

The identity of the sample template. If no template is specified, the BLANK template is used.

This parameter is optional.

## Example

The following LIMSML request uses the CREATE\_BY\_TEMPLATE command to log in a new sample based on the DEFAULT template.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>CREATE_BY_TEMPLATE</command>
          <parameter name="TEMPLATE">BLANK</parameter>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="out" datatype="Text" />
        <field id="DESCRIPTION" direction="inout" datatype="Text">Description text</field>
        <field id="PRIORITY" direction="inout" datatype="Text">3</field>
      </fields>
      <children>
        <entity type="TEST">
          <actions />
          <fields>
            <field id="ANALYSIS" direction="inout" datatype="Text">PH</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>CREATE_BY_TEMPLATE</command>
          <parameter name="TEMPLATE">BLANK</parameter>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="out" datatype="Text">18212</field>
        <field id="DESCRIPTION" direction="inout" datatype="Text">Description text</field>
        <field id="PRIORITY" direction="inout" datatype="Text">3</field>
      </fields>
```

```
<children>
  <entity type="TEST">
    <actions />
    <fields>
      <field id="ANALYSIS" direction="inout" datatype="Text">PH</field>
    </fields>
    <children />
  </entity>
</children>
</entity>
</system>
</transaction>
```

## Sample Create By Workflow

Entity: SAMPLE

Action: CREATE\_BY\_WORKFLOW

Master Menu Procedure Number: 16002

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_create\_by\_workflow

Sample Create By Workflow logs in a new sample using the specified sample login workflow.

The following sections describe the parameters for Sample Create By Workflow.

### WORKFLOW\_NAME

The name of the sample login workflow. If no workflow is specified, the Single Sample workflow is used.

This parameter is optional.

### REPLICATES

The number of times to run the sample login workflow. If the REPLICATES parameter is not specified, the sample login workflow is run once.

This parameter is optional.

## Example

The following LIMSML request uses the CREATE\_BY\_WORKFLOW command to run the Single Sample workflow three times.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>CREATE_BY_WORKFLOW</command>
          <parameter name="WORKFLOW_NAME">Single Sample</parameter>
          <parameter name="REPLICATES">3</parameter>
        </action>
      </actions>
      <fields>
        <field id="ID_TEXT" direction="out" datatype="Text" />
        <field id="ID_NUMERIC" direction="out" datatype="Text" />
        <field id="DESCRIPTION" direction="inout" datatype="Text">Updated after workflow by
VGL.</field>
        <field id="PRIORITY" direction="inout" datatype="Text">3</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>CREATE_BY_WORKFLOW</command>
          <parameter name="WORKFLOW_NAME">Single Sample</parameter>
          <parameter name="REPLICATES">3</parameter>
        </action>
      </actions>
      <fields>
        <field id="ID_TEXT" direction="out" datatype="Text">SM-4-APR-2014-000050</field>
        <field id="ID_NUMERIC" direction="out" datatype="Text">52</field>
        <field id="DESCRIPTION" direction="inout" datatype="Text">Updated after workflow by
VGL.</field>
        <field id="PRIORITY" direction="inout" datatype="Text">3</field>
      </fields>
      <children />
    </entity>
```

```

<entity type="SAMPLE">
  <actions>
    <action>
      <command>CREATE_BY_WORKFLOW</command>
      <parameter name="WORKFLOW_NAME">Single Sample</parameter>
      <parameter name="REPLICATES">3</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="out" datatype="Text">SM-4-APR-2014-000051</field>
    <field id="ID_NUMERIC" direction="out" datatype="Text">53</field>
    <field id="DESCRIPTION" direction="inout" datatype="Text">Updated after workflow by
VGL.</field>
    <field id="PRIORITY" direction="inout" datatype="Text">3</field>
  </fields>
  <children />
</entity>
<entity type="SAMPLE">
  <actions>
    <action>
      <command>CREATE_BY_WORKFLOW</command>
      <parameter name="WORKFLOW_NAME">Single Sample</parameter>
      <parameter name="REPLICATES">3</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="out" datatype="Text">SM-4-APR-2014-000052</field>
    <field id="ID_NUMERIC" direction="out" datatype="Text">54</field>
    <field id="DESCRIPTION" direction="inout" datatype="Text">Updated after workflow by
VGL.</field>
    <field id="PRIORITY" direction="inout" datatype="Text">3</field>
  </fields>
  <children />
</entity>
</system>
</transaction>

```

## Sample Get Component Limits

Entity: SAMPLE

Action: GET\_COMPONENT\_LIMITS

Master Menu Procedure Number: 15639

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_get\_component\_limits

Sample Get Component Limits returns a datatable of MLP limits for the specified result.

The following sections describe the parameters for Sample Get Component Limits.

### COMPONENT\_NAME

The name of the component associated with the result.

This parameter is mandatory.

### SAMPLE\_ID

The identity of the sample.

This parameter is mandatory.

### TEST\_ID

The number of the test associated with the result.

This parameter is mandatory.

## Example

The following LIMSML request uses the GET\_COMPONENT\_LIMITS command to obtain the limits that apply to the Cadmium result in test number 13966 on sample 18213.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>GET_COMPONENT_LIMITS</command>
          <parameter name="SAMPLE_ID">18213</parameter>
          <parameter name="TEST_ID">13966</parameter>
          <parameter name="COMPONENT_NAME">Cadmium</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="LIMITS">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="Level_x0020_ID" type="xs:string" minOccurs="0" />
                      <xs:element name="Minimum" type="xs:string" minOccurs="0" />
                      <xs:element name="Maximum" type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
```

## 2 Standard LIMSML Entity Actions

### LIMSML Entity Actions for Dynamic Data Entities

```
<LIMITS>
  <Level_x0020_ID>INDOOR    </Level_x0020_ID>
  <Minimum>1 </Minimum>
  <Maximum>10 </Maximum>
</LIMITS>
<LIMITS>
  <Level_x0020_ID>AIR_DANGER</Level_x0020_ID>
  <Minimum>>10 </Minimum>
  <Maximum />
</LIMITS>
</NewDataSet>
</ADODDataSet>
</data>
</transaction>
```

## Sample Get Results

Entity: SAMPLE

Action: GET\_RESULTS

Master Menu Procedure Number: 15631

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_get\_results

Sample Get Results returns a datatable of results for a specified sample.

The following section describes the parameter for Sample Get Results.

### SAMPLE\_ID

The identity of the sample.

This parameter is mandatory.

## Example

The following LIMSML request uses the GET\_RESULTS command to obtain the results for sample 18199.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>GET_RESULTS</command>
          <parameter name="SAMPLE_ID">18199</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="RESULT">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="ANALYSIS_ID" msdata:Caption="Analysis id"
type="xs:string" minOccurs="0" />
                      <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
                      <xs:element name="RESULT" msdata:Caption="Result" type="xs:string"
minOccurs="0" />
                      <xs:element name="STATUS" msdata:Caption="Status" type="xs:string"
minOccurs="0" />
                      <xs:element name="RESULT_TYPE" msdata:Caption="Result type"
type="xs:string" minOccurs="0" />
                      <xs:element name="IN_SPEC" msdata:Caption="In spec" type="xs:boolean"
minOccurs="0" />
                      <xs:element name="ORDER_NUMBER" msdata:Caption="Order number"
type="xs:int" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

## 2 Standard LIMSML Entity Actions

### LIMSML Entity Actions for Dynamic Data Entities

```

        <xs:element name="TEST_STATUS" msdata:Caption="Test status"
type="xs:string" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<RESULT>
  <ANALYSIS_ID>          1 ASBESTOS/          1</ANALYSIS_ID>
  <NAME>Asbestos fibers</NAME>
  <RESULT>52 ppb</RESULT>
  <STATUS>E</STATUS>
  <RESULT_TYPE>N</RESULT_TYPE>
  <IN_SPEC>true</IN_SPEC>
  <ORDER_NUMBER>1</ORDER_NUMBER>
  <TEST_STATUS>C</TEST_STATUS>
</RESULT>
<RESULT>
  <ANALYSIS_ID>          2 METALS/          1</ANALYSIS_ID>
  <NAME>Cadmium</NAME>
  <RESULT>13 ppb</RESULT>
  <STATUS>M</STATUS>
  <RESULT_TYPE>N</RESULT_TYPE>
  <IN_SPEC>true</IN_SPEC>
  <ORDER_NUMBER>3</ORDER_NUMBER>
  <TEST_STATUS>C</TEST_STATUS>
</RESULT>
<RESULT>
  <ANALYSIS_ID>          2 METALS/          1</ANALYSIS_ID>
  <NAME>Lead</NAME>
  <RESULT>15 ppb</RESULT>
  <STATUS>M</STATUS>
  <RESULT_TYPE>N</RESULT_TYPE>
  <IN_SPEC>true</IN_SPEC>
  <ORDER_NUMBER>2</ORDER_NUMBER>
  <TEST_STATUS>C</TEST_STATUS>
</RESULT>
<RESULT>
  <ANALYSIS_ID>          2 METALS/          1</ANALYSIS_ID>
  <NAME>Mercury</NAME>
  <RESULT>54 ppb</RESULT>
  <STATUS>E</STATUS>
  <RESULT_TYPE>N</RESULT_TYPE>
  <IN_SPEC>true</IN_SPEC>
  <ORDER_NUMBER>1</ORDER_NUMBER>
  <TEST_STATUS>C</TEST_STATUS>
</RESULT>
</NewDataSet>
</ADODDataSet>
</data>
</transaction>
```

## Sample Get Sample Re Prompt

Entity: SAMPLE

Action: GET\_SAMPLE\_RE\_PROMPT

Master Menu Procedure Number: 15635

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_re\_get\_results\_prompts

Sample Get Sample Re Prompt returns a datatable of results for the specified sample, and a datatable of prompt information for the results.

Sample Get Sample Re Prompt has no parameters.

### Example

The following LIMSML request uses the GET\_SAMPLE\_RE\_RESULTS command to obtain the result and result entry prompt information for sample 18199.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>GET_SAMPLE_RE_PROMPTS</command>
        </action>
      </actions>
      <fields />
      <children>
        <entity type="SAMPLE">
          <actions />
          <fields>
            <field id="_KEY0" direction="in" datatype="Text">18199</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:msns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="RESULT">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="_KEY0" msdata:Caption="key0" type="xs:string"
minOccurs="0" />
                      <xs:element name="ANALYSIS_ID" msdata:Caption="Analysis id"
type="xs:string" minOccurs="0" />
                      <xs:element name="ANALYSIS" msdata:Caption="Analysis"
type="xs:string" minOccurs="0" />
                      <xs:element name="ORDER_NUMBER" msdata:Caption="Order number"
type="xs:string" minOccurs="0" />
                      <xs:element name="ALLOWED_CHARACTERS" msdata:Caption="Allowed
characters" type="xs:string" minOccurs="0" />
                      <xs:element name="CALCULATION" msdata:Caption="Calculation"
type="xs:string" minOccurs="0" />
                      <xs:element name="COLUMN_NAME" msdata:Caption="Column name"
type="xs:string" minOccurs="0" />
                      <xs:element name="ENTERED_ON" msdata:Caption="Entered on"
type="xs:dateTime" minOccurs="0" />
                      <xs:element name="FALSE_WORD" msdata:Caption="False word"
type="xs:string" minOccurs="0" />
                      <xs:element name="MAXIMUM" msdata:Caption="Maximum" type="xs:double"
minOccurs="0" />
                      <xs:element name="MINIMUM" msdata:Caption="Minimum" type="xs:double"
minOccurs="0" />
                      <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
                      <xs:element name="OUT_OF_RANGE" msdata:Caption="Out of range"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="RESULT_TYPE" msdata:Caption="Result type"
type="xs:string" minOccurs="0" />
                      <xs:element name="STATUS" msdata:Caption="Status" type="xs:string"
minOccurs="0" />
                      <xs:element name="TEST_NUMBER" msdata:Caption="Test number"
type="xs:int" minOccurs="0" />
                      <xs:element name="TEXT" msdata:Caption="Text" type="xs:string"
minOccurs="0" />
                      <xs:element name="TRUE_WORD" msdata:Caption="True word"
type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:element>
          </xs:schema>
        </NewDataSet>
      </ADODDataSet>
    </data>
  </transaction>
```

```

        <xs:element name="UNITS" msdata:Caption="Units" type="xs:string"
minOccurs="0" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="COMPONENT_PROMPTS">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="IDENTITY" msdata:Caption="Get identity"
type="xs:string" minOccurs="0" />
            <xs:element name="DESCRIPTION" msdata:Caption="Get name"
type="xs:string" minOccurs="0" />
            <xs:element name="TYPE" msdata:Caption="Type" type="xs:string"
minOccurs="0" />
            <xs:element name="ALLOWED_CHARACTERS" msdata:Caption="Get allowed
characters" type="xs:string" minOccurs="0" />
            <xs:element name="FORMAT_STRING" msdata:Caption="Get format string"
type="xs:string" minOccurs="0" />
            <xs:element name="TRUE_WORD" msdata:Caption="Get true word"
type="xs:string" minOccurs="0" />
            <xs:element name="FALSE_WORD" msdata:Caption="Get false word"
type="xs:string" minOccurs="0" />
            <xs:element name="READ_ONLY" msdata:Caption="Is read only"
type="xs:boolean" minOccurs="0" />
            <xs:element name="MANDATORY" msdata:Caption="Is mandatory"
type="xs:boolean" minOccurs="0" />
            <xs:element name="LENGTH" msdata:Caption="Get length" type="xs:int"
minOccurs="0" />
            <xs:element name="DEFAULT_VALUE" msdata:Caption="Default value"
type="xs:string" minOccurs="0" />
            <xs:element name="LOWER_LIMIT" msdata:Caption="Lower limit"
type="xs:string" minOccurs="0" />
            <xs:element name="UPPER_LIMIT" msdata:Caption="Upper limit"
type="xs:string" minOccurs="0" />
            <xs:element name="BROWSE_TYPE" msdata:Caption="Browse type"
type="xs:string" minOccurs="0" />
            <xs:element name="BROWSE_PROPERTY" msdata:Caption="Browse property"
type="xs:string" minOccurs="0" />
            <xs:element name="SELECT_LIST" msdata:Caption="Select list"
type="xs:string" minOccurs="0" />
            <xs:element name="HIDDEN" msdata:Caption="Hidden" type="xs:boolean"
minOccurs="0" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<RESULT>
    <_KEY0>      13939Asbestos fibers      </_KEY0>
    <ANALYSIS_ID>      1 ASBESTOS/      1</ANALYSIS_ID>
    <ANALYSIS>ASBESTOS</ANALYSIS>
    <ORDER_NUMBER>      1</ORDER_NUMBER>
    <ALLOWED_CHARACTERS />
    <CALCULATION />
    <ENTERED_ON>2012-10-25T13:14:39+01:00</ENTERED_ON>

```

## 2 Standard LIMSML Entity Actions

### LIMSML Entity Actions for Dynamic Data Entities

```
<FALSE_WORD />
<MAXIMUM>0</MAXIMUM>
<MINIMUM>0</MINIMUM>
<NAME>Asbestos fibers</NAME>
<OUT_OF_RANGE>false</OUT_OF_RANGE>
<RESULT_TYPE>N</RESULT_TYPE>
<STATUS>E</STATUS>
<TEST_NUMBER>13939</TEST_NUMBER>
<TEXT>52</TEXT>
<TRUE_WORD />
<UNITS>ppb_wt</UNITS>
</RESULT>
<RESULT>
  <_KEY0>      13940Cadmium                                </_KEY0>
  <ANALYSIS_ID>      2 METALS/                                1</ANALYSIS_ID>
  <ANALYSIS>METALS</ANALYSIS>
  <ORDER_NUMBER>      3</ORDER_NUMBER>
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <COLUMN_NAME />
  <ENTERED_ON>2012-10-25T13:20:14+01:00</ENTERED_ON>
  <FALSE_WORD />
  <MAXIMUM>0</MAXIMUM>
  <MINIMUM>0</MINIMUM>
  <NAME>Cadmium</NAME>
  <OUT_OF_RANGE>false</OUT_OF_RANGE>
  <RESULT_TYPE>N</RESULT_TYPE>
  <STATUS>M</STATUS>
  <TEST_NUMBER>13940</TEST_NUMBER>
  <TEXT>13</TEXT>
  <TRUE_WORD />
  <UNITS>ppb</UNITS>
</RESULT>
<RESULT>
  <_KEY0>      13940Lead                                    </_KEY0>
  <ANALYSIS_ID>      2 METALS/                                1</ANALYSIS_ID>
  <ANALYSIS>METALS</ANALYSIS>
  <ORDER_NUMBER>      2</ORDER_NUMBER>
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <COLUMN_NAME />
  <ENTERED_ON>2012-10-25T13:15:05+01:00</ENTERED_ON>
  <FALSE_WORD />
  <MAXIMUM>0</MAXIMUM>
  <MINIMUM>0</MINIMUM>
  <NAME>Lead</NAME>
  <OUT_OF_RANGE>false</OUT_OF_RANGE>
  <RESULT_TYPE>N</RESULT_TYPE>
  <STATUS>M</STATUS>
  <TEST_NUMBER>13940</TEST_NUMBER>
  <TEXT>15</TEXT>
  <TRUE_WORD />
  <UNITS>ppb</UNITS>
</RESULT>
```

```

<RESULT>
  <_KEY0>      13940Mercury                                </_KEY0>
  <ANALYSIS_ID>      2 METALS/                                1</ANALYSIS_ID>
  <ANALYSIS>METALS</ANALYSIS>
  <ORDER_NUMBER>      1</ORDER_NUMBER>
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <ENTERED_ON>2012-10-25T13:14:39+01:00</ENTERED_ON>
  <FALSE_WORD />
  <MAXIMUM>0</MAXIMUM>
  <MINIMUM>0</MINIMUM>
  <NAME>Mercury</NAME>
  <OUT_OF_RANGE>false</OUT_OF_RANGE>
  <RESULT_TYPE>N</RESULT_TYPE>
  <STATUS>E</STATUS>
  <TEST_NUMBER>13940</TEST_NUMBER>
  <TEXT>54</TEXT>
  <TRUE_WORD />
  <UNITS>ppb</UNITS>
</RESULT>
<COMPONENT_PROMPTS>
  <IDENTITY>Asbestos fibers:ASBESTOS</IDENTITY>
  <DESCRIPTION>Asbestos fibers</DESCRIPTION>
  <TYPE>REAL</TYPE>
  <FORMAT_STRING />
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE />
  <HIDDEN>false</HIDDEN>
</COMPONENT_PROMPTS>
<COMPONENT_PROMPTS>
  <IDENTITY>Cadmium:METALS</IDENTITY>
  <DESCRIPTION>Cadmium</DESCRIPTION>
  <TYPE>REAL</TYPE>
  <FORMAT_STRING />
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE />
  <HIDDEN>false</HIDDEN>
</COMPONENT_PROMPTS>
<COMPONENT_PROMPTS>
  <IDENTITY>Lead:METALS</IDENTITY>
  <DESCRIPTION>Lead</DESCRIPTION>
  <TYPE>REAL</TYPE>
  <FORMAT_STRING />
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE />
  <HIDDEN>false</HIDDEN>
</COMPONENT_PROMPTS>
<COMPONENT_PROMPTS>
  <IDENTITY>Mercury:METALS</IDENTITY>
  <DESCRIPTION>Mercury</DESCRIPTION>
  <TYPE>REAL</TYPE>
  <FORMAT_STRING />
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>

```

```
<DEFAULT_VALUE />
<HIDDEN>false</HIDDEN>
</COMPONENT_PROMPTS>
</NewDataSet>
</ADODDataSet>
</data>
</transaction>
```

## Sample Re Put Results

Entity: SAMPLE

Action: RE\_PUT\_RESULTS

Master Menu Procedure Number: 15636

VGL Library: \$LIMSML\_SAMPLE

Routine: re\_put\_results

Sample re put results creates ad hoc results.

Sample Re Put Results has no parameters.

## Example

The following LIMSML request uses the RE\_PUT\_RESULTS command to add two ad hoc results to test 13968.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>RE_PUT_RESULTS</command>
        </action>
      </actions>
      <fields />
      <children>
        <entity type="RESULT">
          <actions />
          <fields>
            <field id="NAME" direction="in" datatype="Text">AD HOC NUMBER</field>
            <field id="TEST_NUMBER" direction="in" datatype="Text">13968</field>
            <field id="UNITS" direction="in" datatype="Text">ppb</field>
            <field id="RESULT_TYPE" direction="in" datatype="Text">N</field>
            <field id="TEXT" direction="in" datatype="Text">12</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

```

<entity type="RESULT">
  <actions />
  <fields>
    <field id="NAME" direction="in" datatype="Text">AD HOC Text</field>
    <field id="TEST_NUMBER" direction="in" datatype="Text">13968</field>
    <field id="RESULT_TYPE" direction="in" datatype="Text">T</field>
    <field id="TEXT" direction="in" datatype="Text">Text of the ad-hoc text
result</field>
  </fields>
  <children />
</entity>
</children>
</entity>
</system>
</transaction>

```

## Sample Result Entry

Entity: SAMPLE

Action: RESULT\_ENTRY

Master Menu Procedure Number: 15581

VGL Library: \$LIMSML\_SAMPLE

Routine: sample\_result\_entry

Sample Result Entry enters results for a specified sample. See [Result Entry with LIMSML](#) (page 133) for a detailed description of Sample Result Entry and the constants and VGL callbacks that can be used to control result entry using LIMSML.

The following sections describe the parameters for Sample Result Entry.

### COMPLETE\_SAMPLE

If COMPLETE\_SAMPLE is TRUE, the sample\_result\_entry routine updates the sample status to C (Complete) after the results are entered, even if some expected results for the sample have not been entered yet.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_COMPLETE_SAMPLE = "FALSE"
```

This parameter is optional.

## **COMPLETE\_TESTS**

If COMPLETE\_TESTS is TRUE, the sample\_result\_entry routine updates the status of each test to C (Complete) after the results are entered, even if some expected results for the test have not been entered yet.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_COMPLETE_TESTS = "FALSE"
```

This parameter is optional.

## **CREATE\_SAMPLE**

If CREATE\_SAMPLE is TRUE and no sample with the specified identity is found in the database, a new sample is created using the template specified in the CREATE\_SAMPLE\_TEMP parameter.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_SAMPLE = "FALSE"
```

This parameter is optional.

## **CREATE\_SAMPLE\_TEMP**

This parameter defines the sample template to use if a sample is created by the Sample Result Entry LIMSML entity action.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_SAMPLE_TEMPLATE = "BLANK"
```

This parameter is optional.

## **CREATE\_SAMPLE\_WORKFL**

This parameter defines the sample login workflow to use if a sample is created by the Sample Result Entry LIMSML entity action.

This parameter is optional.

## CREATE\_TESTS

If CREATE\_TESTS is TRUE and no appropriate test is found in the database, new tests are created.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_TESTS = "FALSE"
```

This parameter is optional.

## IGNORE\_ADHOC

If IGNORE\_ADHOC is TRUE, ad hoc results included in the LIMSML will not be entered.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_IGNORE_ADHOC = "FALSE"
```

This parameter is optional.

## IGNORE\_ALL\_RESULT\_ER

If IGNORE\_ALL\_RESULT\_ERRORS is TRUE, all mapping and general result entry errors are ignored.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_IGNORE_ALL_RESULT_ERRORS = "FALSE"
```

This parameter is optional.

## IGNORE\_RESULT\_ERRORS

If IGNORE\_RESULT\_ERRORS is TRUE, general result entry errors are ignored.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_IGNORE_RESULT_ERRORS = "TRUE"
```

This parameter is optional.

## **RECALCULATE**

If RECALCULATE is TRUE, calculated components are recalculated for updated tests when results are entered using LIMSML.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_RECALCULATE = "FALSE"
```

This parameter is optional.

## **REQUIRE\_VALID\_SAMPLE**

If REQUIRE\_VALID\_SAMPLE is TRUE, failing to find a sample will result in an error. If REQUIRE\_VALID\_SAMPLE is FALSE, results that are associated with a sample that cannot be found will be ignored.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_REQUIRE_SAMPLE = "TRUE"
```

This parameter is optional.

## **REQUIRE\_VALID\_TEST**

If REQUIRE\_VALID\_TEST is TRUE, failing to find a test will result in an error. If REQUIRE\_VALID\_TEST is FALSE, results that are associated with a test that cannot be found will be ignored.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_REQUIRE_TEST = "TRUE"
```

This parameter is optional.

## **TEST\_REPS\_ENABLED**

If TEST\_REPS\_ENABLED is TRUE, the system will keep track of the test replicates used for each test, so if multiple tests with the same analysis and sample are submitted in the same request, the results of each test will be entered for the next available replicate.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_REPLICATES = "TRUE"
```

This parameter is optional.

## UPDATE\_SAMPLE

If UPDATE\_SAMPLE is TRUE, sample fields in the LIMSML with a direction attribute of "in" or "inout" will update the sample record in the database.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_UPDATE_SAMPLE = "TRUE"
```

This parameter is optional.

## UPDATE\_TESTS

If UPDATE\_TESTS is TRUE, test fields in the LIMSML with a direction attribute of "in" or "inout" will update the test record in the database.

The default value of this parameter is set in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_UPDATE_TESTS = "TRUE"
```

This parameter is optional.

## VALID\_TEST\_STATUS

VALID\_TEST\_STATUS defines the list of acceptable test status values for which result entry is allowed.

The default value of this parameter is defined in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_VALID_TEST_STATUS = "CVP"
```

This parameter is optional.

## Example

The following LIMSML request uses the RESULT\_ENTRY command to enter the pH1 result for the PH test on sample 18215.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>RESULT_ENTRY</command>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="in" datatype="Text">18215</field>
      </fields>
      <children>
        <entity type="TEST">
          <actions />
          <fields>
            <field id="ANALYSIS" direction="in" datatype="Text">PH</field>
          </fields>
          <children>
            <entity type="RESULT">
              <actions />
              <fields>
                <field id="NAME" direction="in" datatype="Text">pH1</field>
                <field id="RESULT_TYPE" direction="in" datatype="Text">N</field>
                <field id="TEXT" direction="in" datatype="Text">12</field>
              </fields>
              <children />
            </entity>
          </children>
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

## LIMSML Entity Actions for Static Data Entities

This section describes the standard LIMSML entity actions for creating and managing static data entities.

### Analysis Get Components

Entity: ANALYSIS

Action: GET\_COMPONENTS

Master Menu Procedure Number: 15569

VGL Library: \$LIMSML\_ANALYSIS

Routine: analysis\_get\_components

Analysis Get Components returns a dataset containing all the components associated with the specified analysis.

The following sections describe the parameters for Analysis Get Components.

#### ANALYSIS\_ID

The identity of the analysis.

This parameter is mandatory.

#### ANALYSIS\_VERSION

The version number of the analysis.

This parameter is optional.

## Example

The following LIMSML request uses the ANALYSIS GET\_COMPONENTS command to return the list of components for an analysis with the identity PH:

```
<?xml version="1.0" encoding="utf-8" ?>
  <transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
    <system response_type="data"
      xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
      <entity type="ANALYSIS">
        <actions>
          <action>
            <command>GET_COMPONENTS</command>
            <parameter name="ANALYSIS_ID">PH</parameter>
          </action>
        </actions>
        <fields />
        <children />
      </entity>
    </system>
  </transaction>
```

The request shown above returns the following type of response if the specified analysis is found. In this example, the PH analysis has four components.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
          targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
          xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
          xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
          xmlns:xs="http://www.w3.org/2001/XMLSchema"
          xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
          elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="ANALYSIS_COMPONENT">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="ANALYSIS" msdata:Caption="Analysis"
                        type="xs:string" minOccurs="0" />
                      <xs:element name="ANALYSIS_VERSION" msdata:Caption="Analysis version"
                        type="xs:int" minOccurs="0" />
                      <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
                        minOccurs="0" />
                      <xs:element name="ORDER_NUMBER" msdata:Caption="Order number"
                        type="xs:int" minOccurs="0" />
                      <xs:element name="RESULT_TYPE" msdata:Caption="Result type"
                        type="xs:string" minOccurs="0" />
                      <xs:element name="UNITS" msdata:Caption="Units" type="xs:string"
                        minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

```

minOccurs="0" />
<xs:element name="MINIMUM" msdata:Caption="Minimum" type="xs:double"
minOccurs="0" />
<xs:element name="MAXIMUM" msdata:Caption="Maximum" type="xs:double"
minOccurs="0" />
<xs:element name="TRUE_WORD" msdata:Caption="True word"
type="xs:string" minOccurs="0" />
<xs:element name="FALSE_WORD" msdata:Caption="False word"
type="xs:string" minOccurs="0" />
<xs:element name="ALLOWED_CHARACTERS" msdata:Caption="Allowed
characters" type="xs:string" minOccurs="0" />
<xs:element name="CALCULATION" msdata:Caption="Calculation"
type="xs:string" minOccurs="0" />
<xs:element name="PLACES" msdata:Caption="Places" type="xs:int"
minOccurs="0" />
<xs:element name="REP_CONTROL" msdata:Caption="Rep control"
type="xs:string" minOccurs="0" />
<xs:element name="REPLICATES" msdata:Caption="Replicates"
type="xs:boolean" minOccurs="0" />
<xs:element name="SIG_FIGS_NUMBER" msdata:Caption="Sig figs number"
type="xs:int" minOccurs="0" />
<xs:element name="SIG_FIGS_ROUNDING" msdata:Caption="Sig figs
rounding" type="xs:int" minOccurs="0" />
<xs:element name="SIG_FIGS_FILTER" msdata:Caption="Sig figs filter"
type="xs:string" minOccurs="0" />
<xs:element name="MINIMUM_PQL" msdata:Caption="Minimum pql"
type="xs:double" minOccurs="0" />
<xs:element name="MAXIMUM_PQL" msdata:Caption="Maximum pql"
type="xs:double" minOccurs="0" />
<xs:element name="PQL_CALCULATION" msdata:Caption="Pql calculation"
type="xs:string" minOccurs="0" />
<xs:element name="FORMULA" msdata:Caption="Formula" type="xs:string"
minOccurs="0" />
<xs:element name="MATRIX_NO" msdata:Caption="Matrix no" type="xs:int"
minOccurs="0" />
<xs:element name="MATRIX_NAME" msdata:Caption="Matrix name"
type="xs:string" minOccurs="0" />
<xs:element name="COLUMN_NO" msdata:Caption="Column no" type="xs:int"
minOccurs="0" />
<xs:element name="COLUMN_NAME" msdata:Caption="Column name"
type="xs:string" minOccurs="0" />
<xs:element name="ROW_NO" msdata:Caption="Row no" type="xs:int"
minOccurs="0" />
<xs:element name="ROW_NAME" msdata:Caption="Row name"
type="xs:string" minOccurs="0" />
<xs:element name="UNCERTAINTY_TEXT" msdata:Caption="Uncertainty text"
type="xs:string" minOccurs="0" />
<xs:element name="ENTITY_CRITERIA" msdata:Caption="Entity criteria"
type="xs:string" minOccurs="0" />
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

## 2 Standard LIMSML Entity Actions

### LIMSML Entity Actions for Static Data Entities

```
<ANALYSIS_COMPONENT>
  <ANALYSIS>PH</ANALYSIS>
  <ANALYSIS_VERSION>1</ANALYSIS_VERSION>
  <NAME>pH1</NAME>
  <ORDER_NUMBER>1</ORDER_NUMBER>
  <RESULT_TYPE>N</RESULT_TYPE>
  <UNITS />
  <MINIMUM>1</MINIMUM>
  <MAXIMUM>14</MAXIMUM>
  <TRUE_WORD />
  <FALSE_WORD />
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <PLACES>-1</PLACES>
  <REP_CONTROL />
  <REPLICATES>false</REPLICATES>
  <SIG_FIGS_NUMBER>-1</SIG_FIGS_NUMBER>
  <SIG_FIGS_ROUNDING>-1</SIG_FIGS_ROUNDING>
  <SIG_FIGS_FILTER />
  <MINIMUM_PQL>0</MINIMUM_PQL>
  <MAXIMUM_PQL>0</MAXIMUM_PQL>
  <MATRIX_NO>0</MATRIX_NO>
  <MATRIX_NAME />
  <COLUMN_NO>0</COLUMN_NO>
  <ROW_NO>0</ROW_NO>
  <UNCERTAINTY_TEXT />
  <ENTITY_CRITERIA />
</ANALYSIS_COMPONENT>
<ANALYSIS_COMPONENT>
  <ANALYSIS>PH</ANALYSIS>
  <ANALYSIS_VERSION>1</ANALYSIS_VERSION>
  <NAME>pH2</NAME>
  <ORDER_NUMBER>2</ORDER_NUMBER>
  <RESULT_TYPE>N</RESULT_TYPE>
  <UNITS />
  <MINIMUM>1</MINIMUM>
  <MAXIMUM>14</MAXIMUM>
  <TRUE_WORD />
  <FALSE_WORD />
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <PLACES>-1</PLACES>
  <REP_CONTROL />
  <REPLICATES>false</REPLICATES>
  <SIG_FIGS_NUMBER>-1</SIG_FIGS_NUMBER>
  <SIG_FIGS_ROUNDING>-1</SIG_FIGS_ROUNDING>
  <SIG_FIGS_FILTER />
  <MINIMUM_PQL>0</MINIMUM_PQL>
  <MAXIMUM_PQL>0</MAXIMUM_PQL>
  <FORMULA />
  <MATRIX_NO>0</MATRIX_NO>
  <MATRIX_NAME />
  <COLUMN_NO>0</COLUMN_NO>
  <COLUMN_NAME />
  <ROW_NO>0</ROW_NO>
  <ROW_NAME />
  <UNCERTAINTY_TEXT />
```

```

    <ENTITY_CRITERIA />
</ANALYSIS_COMPONENT>
<ANALYSIS_COMPONENT>
  <ANALYSIS>PH</ANALYSIS>
  <ANALYSIS_VERSION>1</ANALYSIS_VERSION>
  <NAME>pH3</NAME>
  <ORDER_NUMBER>3</ORDER_NUMBER>
  <RESULT_TYPE>N</RESULT_TYPE>
  <UNITS />
  <MINIMUM>1</MINIMUM>
  <MAXIMUM>14</MAXIMUM>
  <TRUE_WORD />
  <FALSE_WORD />
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <PLACES>-1</PLACES>
  <REP_CONTROL />
  <REPLICATES>false</REPLICATES>
  <SIG_FIGS_NUMBER>-1</SIG_FIGS_NUMBER>
  <SIG_FIGS_ROUNDING>-1</SIG_FIGS_ROUNDING>
  <SIG_FIGS_FILTER />
  <MINIMUM_PQL>0</MINIMUM_PQL>
  <MAXIMUM_PQL>0</MAXIMUM_PQL>
  <FORMULA />
  <MATRIX_NO>0</MATRIX_NO>
  <MATRIX_NAME />
  <COLUMN_NO>0</COLUMN_NO>
  <COLUMN_NAME />
  <ROW_NO>0</ROW_NO>
  <ROW_NAME />
  <UNCERTAINTY_TEXT />
  <ENTITY_CRITERIA />
</ANALYSIS_COMPONENT>
<ANALYSIS_COMPONENT>
  <ANALYSIS>PH</ANALYSIS>
  <ANALYSIS_VERSION>1</ANALYSIS_VERSION>
  <NAME>Average pH</NAME>
  <ORDER_NUMBER>4</ORDER_NUMBER>
  <RESULT_TYPE>K</RESULT_TYPE>
  <UNITS />
  <MINIMUM>0</MINIMUM>
  <MAXIMUM>0</MAXIMUM>
  <TRUE_WORD />
  <FALSE_WORD />
  <ALLOWED_CHARACTERS />
  <CALCULATION />
  <PLACES>-1</PLACES>
  <REP_CONTROL />
  <REPLICATES>false</REPLICATES>
  <SIG_FIGS_NUMBER>-1</SIG_FIGS_NUMBER>
  <SIG_FIGS_ROUNDING>-1</SIG_FIGS_ROUNDING>
  <SIG_FIGS_FILTER />
  <MINIMUM_PQL>0</MINIMUM_PQL>
  <MAXIMUM_PQL>0</MAXIMUM_PQL>
  <FORMULA>("pH1"+"pH2"+"pH3")/3</FORMULA>
  <MATRIX_NO>0</MATRIX_NO>
  <MATRIX_NAME />

```

## 2 Standard LIMSML Entity Actions

### LIMSML Entity Actions for Static Data Entities

```
<COLUMN_NO>0</COLUMN_NO>
<COLUMN_NAME />
<ROW_NO>0</ROW_NO>
<ROW_NAME />
<UNCERTAINTY_TEXT />
<ENTITY_CRITERIA />
</ANALYSIS_COMPONENT>
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## Infomaker Report Generate

Entity: INFOMAKER\_REPORT

Action: GENERATE

Master Menu Procedure Number: 15577

VGL Library: \$LIMSML\_REPORT

Routine: infomaker\_report\_generate

Infomaker Report Generate generates the specified InfoMaker report.

The following sections describe the parameters for Infomaker Report Generate.

### REPORT\_ID

The identity of the InfoMaker report.

This parameter is mandatory.

## Infomaker Report Get Parameters

Entity: INFOMAKER\_REPORT

Action: GET\_PARAMETERS

Master Menu Procedure Number: 15578

VGL Library: \$LIMSML\_REPORT

Routine: infomaker\_report\_get\_parameters

Infomaker Report Get Parameters returns the parameter details of the specified InfoMaker report.

The following sections describe the parameters for Infomaker Report Get Parameters.

### REPORT\_ID

The identity of the InfoMaker report.

This parameter is mandatory.

## Infomaker Report Validate Values

Entity: INFOMAKER\_REPORT

Action: VALIDATE\_VALUES

Master Menu Procedure Number: 15613

VGL Library: \$LIMSML\_REPORT

Routine: infomaker\_report\_validate\_values

InfoMaker Report Validate Values validates the supplied InfoMaker report parameter values.

The following sections describe the parameters for Infomaker Report Validate Values.

### REPORT\_ID

The identity of the InfoMaker report.

This parameter is mandatory.

## Job Template Get Fields

Entity: JOB\_TEMPLATE

Action: GET\_FIELDS

Master Menu Procedure Number: 15579

VGL Library: \$LIMSML\_JOB\_TMPL

Routine: job\_template\_get\_fields

Job Template Get Fields returns the fields associated with the specified job template.

The following sections describe the parameters for Job Template Get Fields.

### TEMPLATE\_ID

The identity of the job template.

This parameter is mandatory.

### Example

The following LIMSML request uses the GET\_FIELDS command to get the list of fields included in the DEFAULT job template.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB_TEMPLATE">
      <actions>
        <action>
          <command>GET_FIELDS</command>
          <parameter name="TEMPLATE_ID">DEFAULT</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODataset>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="JOB_TEMPLATE_FIELD">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="IDENTITY" msdata:Caption="Get identity"
type="xs:string" minOccurs="0" />
                      <xs:element name="DESCRIPTION" msdata:Caption="Get name"
type="xs:string" minOccurs="0" />
                      <xs:element name="TYPE" msdata:Caption="Type" type="xs:string"
minOccurs="0" />
                      <xs:element name="ALLOWED_CHARACTERS" msdata:Caption="Get allowed
characters" type="xs:string" minOccurs="0" />
                      <xs:element name="FORMAT_STRING" msdata:Caption="Get format string"
type="xs:string" minOccurs="0" />
                      <xs:element name="TRUE_WORD" msdata:Caption="Get true word"
type="xs:string" minOccurs="0" />
                      <xs:element name="FALSE_WORD" msdata:Caption="Get false word"
type="xs:string" minOccurs="0" />
                      <xs:element name="READ_ONLY" msdata:Caption="Is read only"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="MANDATORY" msdata:Caption="Is mandatory"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="LENGTH" msdata:Caption="Get length" type="xs:int"
minOccurs="0" />
                      <xs:element name="DEFAULT_VALUE" msdata:Caption="Default value"
type="xs:string" minOccurs="0" />
                      <xs:element name="LOWER_LIMIT" msdata:Caption="Lower limit"
type="xs:string" minOccurs="0" />
                      <xs:element name="UPPER_LIMIT" msdata:Caption="Upper limit"
type="xs:string" minOccurs="0" />
                      <xs:element name="BROWSE_TYPE" msdata:Caption="Browse type"
type="xs:string" minOccurs="0" />
                      <xs:element name="BROWSE_PROPERTY" msdata:Caption="Browse property"
type="xs:string" minOccurs="0" />
                      <xs:element name="SELECT_LIST" msdata:Caption="Select list"
type="xs:string" minOccurs="0" />
                      <xs:element name="HIDDEN" msdata:Caption="Hidden" type="xs:boolean"
minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:element>
          </xs:schema>
        </NewDataSet>
      </ADODataset>
    </data>
  </transaction>
```

```

        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
<JOB_TEMPLATE_FIELD>
  <IDENTITY>BROWSE_DESCRIPTION</IDENTITY>
  <DESCRIPTION>Browse description . .</DESCRIPTION>
  <TYPE>TEXT</TYPE>
  <ALLOWED_CHARACTERS />
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <LENGTH>234</LENGTH>
  <DEFAULT_VALUE />
  <HIDDEN>false</HIDDEN>
</JOB_TEMPLATE_FIELD>
<JOB_TEMPLATE_FIELD>
  <IDENTITY>LOT_ID</IDENTITY>
  <DESCRIPTION>Lot Id</DESCRIPTION>
  <TYPE>TEXT</TYPE>
  <ALLOWED_CHARACTERS>$ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890_-
</ALLOWED_CHARACTERS>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <LENGTH>30</LENGTH>
  <DEFAULT_VALUE />
  <BROWSE_TYPE>LOT</BROWSE_TYPE>
  <BROWSE_PROPERTY>LOT_ID</BROWSE_PROPERTY>
  <HIDDEN>false</HIDDEN>
</JOB_TEMPLATE_FIELD>
<JOB_TEMPLATE_FIELD>
  <IDENTITY>TOTAL_ELEMENTS</IDENTITY>
  <DESCRIPTION>Total Elements</DESCRIPTION>
  <TYPE>INTEGER</TYPE>
  <FORMAT_STRING>*****
*****
*****
*****
</FORMAT_STRING>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE>0</DEFAULT_VALUE>
  <LOWER_LIMIT>1</LOWER_LIMIT>
  <UPPER_LIMIT>2147483647</UPPER_LIMIT>
  <HIDDEN>false</HIDDEN>
</JOB_TEMPLATE_FIELD>
<JOB_TEMPLATE_FIELD>
  <IDENTITY>AUTO_VALIDATE</IDENTITY>
  <DESCRIPTION>Auto validate . . . . .</DESCRIPTION>
  <TYPE>BOOLEAN</TYPE>
  <TRUE_WORD>Yes</TRUE_WORD>
  <FALSE_WORD>No</FALSE_WORD>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE>false</DEFAULT_VALUE>
  <HIDDEN>false</HIDDEN>
</JOB_TEMPLATE_FIELD>

```

```
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## Job Template Validate Values

Entity: JOB\_TEMPLATE

Action: VALIDATE\_VALUES

Master Menu Procedure Number: 15591

VGL Library: \$LIMSML\_JOB\_TMPL

Routine: job\_template\_validate\_values

Job Template Validate Values validates the values supplied to the specified job template.

The following sections describe the parameters for Job Template Validate Values.

### TEMPLATE\_ID

The identity of the job template.

This parameter is mandatory.

### Example

The following LIMSML request uses the VALIDATE\_VALUES command to validate supplied values against fields defined in the DEFAULT job template.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB_TEMPLATE">
      <actions>
        <action>
          <command>VALIDATE_VALUES</command>
          <parameter name="TEMPLATE_ID">DEFAULT</parameter>
        </action>
      </actions>
      <fields>
        <field id="BROWSE_DESCRIPTION" direction="inout" datatype="Text">Description
text</field>
        <field id="LOT_ID" direction="inout" datatype="Text">NOT_EXIST</field>
      </fields>
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response. In this case, the field element for the BROWSE\_DESCRIPTION field contains True because the supplied value passed the validation, while the field element for the LOT\_ID field contains False because the supplied value is not a valid lot identity and the field failed the validation.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB_TEMPLATE">
      <actions />
      <fields>
        <field id="BROWSE_DESCRIPTION" direction="out" datatype="Text">True</field>
        <field id="LOT_ID" direction="out" datatype="Text">False</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Sample Template Get Fields

Entity: SAMPLE\_TEMPLATE

Action: GET\_FIELDS

Master Menu Procedure Number: 15582

VGL Library: \$LIMSML\_SAMPLE\_TMPL

Routine: sample\_template\_get\_fields

Sample Template Get Fields returns a datatable containing the fields associated with the sample template.

The following sections describe the parameters for Sample Template Get Fields.

### JOB\_NAME

The name of the job the sample template will be used to add a sample to.

This parameter is optional.

### TEMPLATE\_ID

The identity of the sample template.

This parameter is mandatory.

## Example

The following LIMSML request uses the GET\_FIELDS command to return a list of the fields associated with the SIMPLE sample template.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions>
        <action>
          <command>GET_FIELDS</command>
          <parameter name="TEMPLATE_ID">SIMPLE</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="SAMPLE_TEMPLATE_FIELD">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="IDENTITY" msdata:Caption="Get identity"
type="xs:string" minOccurs="0" />
                      <xs:element name="DESCRIPTION" msdata:Caption="Get name"
type="xs:string" minOccurs="0" />
                      <xs:element name="TYPE" msdata:Caption="Type" type="xs:string"
minOccurs="0" />
                      <xs:element name="ALLOWED_CHARACTERS" msdata:Caption="Get allowed
characters" type="xs:string" minOccurs="0" />
                      <xs:element name="FORMAT_STRING" msdata:Caption="Get format string"
type="xs:string" minOccurs="0" />
                      <xs:element name="TRUE_WORD" msdata:Caption="Get true word"
type="xs:string" minOccurs="0" />
                      <xs:element name="FALSE_WORD" msdata:Caption="Get false word"
type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

```

        <xs:element name="READ_ONLY" msdata:Caption="Is read only"
type="xs:boolean" minOccurs="0" />
        <xs:element name="MANDATORY" msdata:Caption="Is mandatory"
type="xs:boolean" minOccurs="0" />
        <xs:element name="LENGTH" msdata:Caption="Get length" type="xs:int"
minOccurs="0" />
        <xs:element name="DEFAULT_VALUE" msdata:Caption="Default value"
type="xs:string" minOccurs="0" />
        <xs:element name="LOWER_LIMIT" msdata:Caption="Lower limit"
type="xs:string" minOccurs="0" />
        <xs:element name="UPPER_LIMIT" msdata:Caption="Upper limit"
type="xs:string" minOccurs="0" />
        <xs:element name="BROWSE_TYPE" msdata:Caption="Browse type"
type="xs:string" minOccurs="0" />
        <xs:element name="BROWSE_PROPERTY" msdata:Caption="Browse property"
type="xs:string" minOccurs="0" />
        <xs:element name="SELECT_LIST" msdata:Caption="Select list"
type="xs:string" minOccurs="0" />
        <xs:element name="HIDDEN" msdata:Caption="Hidden" type="xs:boolean"
minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<SAMPLE_TEMPLATE_FIELD>
  <IDENTITY>PREPARATION_ID</IDENTITY>
  <DESCRIPTION>Preparation</DESCRIPTION>
  <TYPE>TEXT</TYPE>
  <ALLOWED_CHARACTERS>$ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890_-
</ALLOWED_CHARACTERS>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <LENGTH>10</LENGTH>
  <DEFAULT_VALUE>SIEVE</DEFAULT_VALUE>
  <BROWSE_TYPE>PREPARATION</BROWSE_TYPE>
  <BROWSE_PROPERTY>IDENTITY</BROWSE_PROPERTY>
  <HIDDEN>false</HIDDEN>
</SAMPLE_TEMPLATE_FIELD>
<SAMPLE_TEMPLATE_FIELD>
  <IDENTITY>SAMPLING_POINT</IDENTITY>
  <DESCRIPTION>Sample Point</DESCRIPTION>
  <TYPE>TEXT</TYPE>
  <ALLOWED_CHARACTERS>$ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890_-
</ALLOWED_CHARACTERS>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <LENGTH>10</LENGTH>
  <DEFAULT_VALUE />
  <BROWSE_TYPE>SAMPLE_POINT</BROWSE_TYPE>
  <BROWSE_PROPERTY>IDENTITY</BROWSE_PROPERTY>
  <HIDDEN>false</HIDDEN>
</SAMPLE_TEMPLATE_FIELD>

```

```
<SAMPLE_TEMPLATE_FIELD>
  <IDENTITY>SAMPLED_DATE</IDENTITY>
  <DESCRIPTION>Collected On</DESCRIPTION>
  <TYPE>DATETIME</TYPE>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE>2012-10-26T11:28:25+01:00</DEFAULT_VALUE>
  <LOWER_LIMIT />
  <UPPER_LIMIT />
  <HIDDEN>false</HIDDEN>
</SAMPLE_TEMPLATE_FIELD>
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## Sample Template Validate Values

Entity: SAMPLE\_TEMPLATE

Action: VALIDATE\_VALUES

Master Menu Procedure Number: 15589

VGL Library: \$LIMSML\_SAMPLE\_TMPL

Routine: sample\_template\_validate\_values

Sample Template Validate Values validates the values for the sample template fields.

The following section describes the parameter for Sample Template Validate Values.

### TEMPLATE\_ID

The identity of the sample template.

This parameter is mandatory.

## Example

The following LIMSML request uses the VALIDATE\_VALUES command to check values supplied for fields in the SIMPLE sample template.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions>
        <action>
          <command>VALIDATE_VALUES</command>
          <parameter name="TEMPLATE_ID">SIMPLE</parameter>
        </action>
      </actions>
      <fields>
        <field id="SAMPLING_POINT" direction="inout" datatype="Text">NOWHERE</field>
        <field id="SAMPLED_DATE" direction="inout" datatype="Text">15-OCT-2012</field>
        <field id="PREPARATION_ID" direction="inout" datatype="Text">SIEVE</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response. If the value supplied for a field is valid, the field element in the response contains True. If the value supplied for a field is not valid, the field element in the response contains the value False.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions />
      <fields>
        <field id="SAMPLING_POINT" direction="out" datatype="Text">False</field>
        <field id="SAMPLED_DATE" direction="out" datatype="Text">True</field>
        <field id="PREPARATION_ID" direction="out" datatype="Text">True</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Test Schedule Get Contents

Entity: TEST\_SCHEDULE

Action: GET\_CONTENTS

Master Menu Procedure Number: 15593

VGL Library: \$LIMSML\_TEST\_SCHEDULE

Routine: test\_schedule\_get\_contents

Test Schedule Get Contents returns a list of the analyses in a specified test schedule, with the settings made for each analysis. If a test schedule contains a child test schedule, the analyses within the child test schedule are returned.

The following section describes the parameter for Test Schedule Get Contents.

### TEST\_SCHEDULE\_ID

The identity of the test schedule.

This parameter is mandatory.

### Example

The following LIMSML request uses the GET\_CONTENTS command to return a list of the analyses associated with the TESTSCHED1 test schedule.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="TEST_SCHEDULE">
      <actions>
        <action>
          <command>GET_CONTENTS</command>
          <parameter name="TEST_SCHEDULE_ID">TESTSCHED1</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODataset>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:msns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="TEST_SCHEDULE_ENTRY">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="STD_TEST" msdata:Caption="Std test"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="ANALYSIS_ID" msdata:Caption="Analysis id"
type="xs:string" minOccurs="0" />
                      <xs:element name="COMPONENT_LIST" msdata:Caption="Component list"
type="xs:string" minOccurs="0" />
                      <xs:element name="INSTRUMENT_ID" msdata:Caption="Instrument id"
type="xs:string" minOccurs="0" />
                      <xs:element name="INSTRUMENT_TYPE" msdata:Caption="Instrument type"
type="xs:string" minOccurs="0" />
                      <xs:element name="REPLICATE_COUNT" msdata:Caption="Replicate count"
type="xs:int" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
        <TEST_SCHEDULE_ENTRY>
          <STD_TEST>true</STD_TEST>
          <ANALYSIS_ID>ASBESTOS</ANALYSIS_ID>
          <COMPONENT_LIST />
          <INSTRUMENT_ID>GCMS6895</INSTRUMENT_ID>
          <INSTRUMENT_TYPE />
          <REPLICATE_COUNT>1</REPLICATE_COUNT>
        </TEST_SCHEDULE_ENTRY>
        <TEST_SCHEDULE_ENTRY>
          <STD_TEST>true</STD_TEST>
          <ANALYSIS_ID>CAL1</ANALYSIS_ID>
          <COMPONENT_LIST />
          <INSTRUMENT_ID />
          <INSTRUMENT_TYPE>GAS_MONIT</INSTRUMENT_TYPE>
          <REPLICATE_COUNT>1</REPLICATE_COUNT>
        </TEST_SCHEDULE_ENTRY>
      </NewDataSet>
    </ADODataset>
  </data>
</transaction>
```

```
<TEST_SCHEDULE_ENTRY>
  <STD_TEST>true</STD_TEST>
  <ANALYSIS_ID>METALS</ANALYSIS_ID>
  <COMPONENT_LIST />
  <INSTRUMENT_ID />
  <INSTRUMENT_TYPE />
  <REPLICATE_COUNT>1</REPLICATE_COUNT>
</TEST_SCHEDULE_ENTRY>
<TEST_SCHEDULE_ENTRY>
  <STD_TEST>true</STD_TEST>
  <ANALYSIS_ID>MERCURY</ANALYSIS_ID>
  <COMPONENT_LIST />
  <INSTRUMENT_ID />
  <INSTRUMENT_TYPE />
  <REPLICATE_COUNT>1</REPLICATE_COUNT>
</TEST_SCHEDULE_ENTRY>
<TEST_SCHEDULE_ENTRY>
  <STD_TEST>true</STD_TEST>
  <ANALYSIS_ID>FORM_QA</ANALYSIS_ID>
  <COMPONENT_LIST />
  <INSTRUMENT_ID />
  <INSTRUMENT_TYPE />
  <REPLICATE_COUNT>1</REPLICATE_COUNT>
</TEST_SCHEDULE_ENTRY>
</NewDataSet>
</ADODDataSet>
</data>
</transaction>
```

## Other LIMSML Entity Actions

The LIMSML entity actions in this section provide functionality that supports the SampleManager web site.

### Criteria Get

Entity: CRITERIA

Action: GET

Master Menu Procedure Number: 15621

VGL Library: \$LIMSML\_CRITERIA

Routine: get\_criteria

Criteria Get returns a dataset containing all the saved criteria for the specified entity type.

The following section describes the parameter for Criteria Get.

#### TABLE\_NAME

The name of the table associated with the required entity type.

This parameter is mandatory.

### Example

The following LIMSML request uses the CRITERIA GET command to return all of the saved criteria for the SAMPLE\_TEMPLATE table.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="CRITERIA">
      <actions>
        <action>
          <command>GET</command>
          <parameter name="TABLE_NAME">SAMP_TMPL_HEADER</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request shown above returns the following type of response if saved criteria exist for the specified table. In this example, there are two saved criteria associated with the SAMP\_TMPL\_HEADER table.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="CRITERIA">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="TABLE_NAME" msdata:Caption="Table name"
type="xs:string" minOccurs="0" />
                      <xs:element name="IDENTITY" msdata:Caption="Identity"
type="xs:string" minOccurs="0" />
                      <xs:element name="MODIFIED_ON" msdata:Caption="Modified on"
type="xs:dateTime" minOccurs="0" />
                      <xs:element name="MODIFIED_BY" msdata:Caption="Modified by"
type="xs:string" minOccurs="0" />
                      <xs:element name="MODIFIABLE" msdata:Caption="Modifiable"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="REMOVEFLAG" msdata:Caption="Removeflag"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="GROUP_ID" msdata:Caption="Group id"
type="xs:string" minOccurs="0" />
                      <xs:element name="DESCRIPTION" msdata:Caption="Description"
type="xs:string" minOccurs="0" />
                      <xs:element name="TASK_NAME" msdata:Caption="Task name"
type="xs:string" minOccurs="0" />
                      <xs:element name="FORM" msdata:Caption="Form" type="xs:string"
minOccurs="0" />
                      <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
                      <xs:element name="PUBLIC_CRITERIA" msdata:Caption="Public criteria"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="HAS_ATTACHMENTS" msdata:Caption="Has attachments"
type="xs:boolean" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

## 2 Standard LIMSML Entity Actions

### Other LIMSML Entity Actions

```
<CRITERIA>
  <TABLE_NAME>SAMP_TMPL_HEADER</TABLE_NAME>
  <IDENTITY>CAL1</IDENTITY>
  <MODIFIED_ON>2012-10-22T14:16:11+01:00</MODIFIED_ON>
  <MODIFIED_BY>SYSTEM</MODIFIED_BY>
  <MODIFIABLE>true</MODIFIABLE>
  <REMOVEFLAG>false</REMOVEFLAG>
  <GROUP_ID />
  <DESCRIPTION>Cal1 Samp Tmpl Header</DESCRIPTION>
  <TASK_NAME />
  <FORM />
  <NAME>Cal1 Samp Tmpl Header</NAME>
  <PUBLIC_CRITERIA>true</PUBLIC_CRITERIA>
  <HAS_ATTACHMENTS>false</HAS_ATTACHMENTS>
</CRITERIA>
<CRITERIA>
  <TABLE_NAME>SAMP_TMPL_HEADER</TABLE_NAME>
  <IDENTITY>DEFAULT_SYSTEM</IDENTITY>
  <MODIFIED_ON>2012-10-22T14:15:14+01:00</MODIFIED_ON>
  <MODIFIED_BY>SYSTEM</MODIFIED_BY>
  <MODIFIABLE>true</MODIFIABLE>
  <REMOVEFLAG>false</REMOVEFLAG>
  <GROUP_ID />
  <DESCRIPTION />
  <TASK_NAME />
  <FORM />
  <NAME>Default System Samp Tmpl Header</NAME>
  <PUBLIC_CRITERIA>true</PUBLIC_CRITERIA>
  <HAS_ATTACHMENTS>false</HAS_ATTACHMENTS>
</CRITERIA>
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## Explorer Aux Execute

Entity: EXPLORER\_AUX

Action: EXECUTE

Master Menu Procedure Number: 15626

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_aux\_execute

Explorer Aux Execute runs the specified explorer auxiliary report.

The following sections describe the parameters for Explorer Aux Execute.

### IDENTITY

The identity of the explorer auxiliary report.

This parameter is mandatory.

### TABLE

The name of the table associated with the explorer auxiliary report.

This parameter is mandatory.

## Explorer Aux Get Parameters

Entity: EXPLORER\_AUX

Action: GET\_PARAMETERS

Master Menu Procedure Number: 15625

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_aux\_get\_parameters

Explorer Aux Get Parameters returns the parameter details of the specified explorer auxiliary report.

The following sections describe the parameters for Explorer Aux Get Parameters.

### IDENTITY

The identity of the explorer auxiliary report.

This parameter is optional.

### TABLE

The name of the table associated with the explorer auxiliary report.

This parameter is optional.

## Explorer Aux Validate

Entity: EXPLORER\_AUX

Action: VALIDATE

Master Menu Procedure Number: 15627

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_aux\_validate

Explorer Aux Validate validates the specified explorer auxiliary report.

The following sections describe the parameters for Explorer Aux Validate.

### IDENTITY

The identity of the explorer auxiliary report.

This parameter is mandatory.

### TABLE

The name of the table associated with the explorer auxiliary report.

This parameter is mandatory.

## Explorer Folder Get Columns

Entity: EXPLORER\_FOLDER

Action: GET\_COLUMNS

Master Menu Procedure Number: 15616

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_folder\_get\_columns

Explorer Folder Get Columns returns a dataset containing the columns for the specified explorer folder.

The following sections describe the parameters for Explorer Folder Get Columns.

### **CABINET\_ID**

The identity of the explorer cabinet that contains the folder.

This parameter is mandatory.

### **FOLDER\_NUMBER**

The explorer folder number.

This parameter is mandatory.

## Explorer Folder Get Content Count

Entity: EXPLORER\_FOLDER

Action: GET\_CONTENT\_COUNT

Master Menu Procedure Number: 15572

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_folder\_get\_content\_count

Explorer Folder Get Content Count returns the total number of items in the specified explorer folder.

The following sections describe the parameters for Explorer Folder Get Content Count.

### CABINET\_ID

The identity of the explorer cabinet that contains the folder.

This parameter is mandatory.

### FOLDER\_NUMBER

The explorer folder number.

This parameter is mandatory.

### Example

The following LIMSML request uses the EXPLORER\_FOLDER GET\_CONTENT\_COUNT command to obtain the number of items in the explorer folder with folder\_number 1 in the DEFAULT cabinet.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="EXPLORER_FOLDER">
      <actions>
        <action>
          <command>GET_CONTENT_COUNT</command>
          <parameter name="CABINET_ID">DEFAULT</parameter>
          <parameter name="FOLDER_NUMBER">1</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
```

</transaction>

The request above produces the following type of response, if the folder is found. In this case, there are two items in the specified folder.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="EXPLORER_FOLDER">
      <actions>
        <action>
          <command>GET_CONTENT_COUNT</command>
          <parameter name="CABINET_ID">DEFAULT</parameter>
          <parameter name="FOLDER_NUMBER">1</parameter>
        </action>
      </actions>
      <fields>
        <field id="RETURN" direction="inout" datatype="Text">2</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Explorer Folder Get Contents

Entity: EXPLORER\_FOLDER

Action: GET\_CONTENTS

Master Menu Procedure Number: 15572

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_folder\_get\_contents

Returns a dataset containing a page of items for the specified explorer folder.

The following sections describe the parameters for Explorer Folder Get Contents.

### CABINET\_ID

The identity of the explorer cabinet that contains the folder.

This parameter is mandatory.

## FOLDER\_NUMBER

The explorer folder number.

This parameter is mandatory.

## PAGE

The number of the page to display.

This parameter is optional.

## PAGESIZE

The number of items that a page can contain.

This parameter is optional.

## SCHEMA

If the SCHEMA parameter is TRUE, the schema is included in the output.

This parameter is optional.

## Example

The following LIMSML request uses the EXPLORER\_FOLDER GET\_CONTENTS command to obtain the first page of 25 items in the explorer folder with folder\_number 7 in the SYSTEM cabinet.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="EXPLORER_FOLDER">
      <actions>
        <action>
          <command>GET_CONTENTS</command>
          <parameter name="CABINET_ID">SYSTEM</parameter>
          <parameter name="FOLDER_NUMBER">7</parameter>
          <parameter name="PAGE">1</parameter>
          <parameter name="PAGESIZE">5</parameter>
          <parameter name="SCHEMA">True</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response, if the folder is found.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODataset>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:msdtns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="MASTER_MENU">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="_ICON_ID" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
                      <xs:element name="_ICON_TEXT" msdata:Caption="Icon Text"
type="xs:string" minOccurs="0" />
                      <xs:element name="_NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
                      <xs:element name="_KEY0" msdata:Caption="key0" type="xs:string"
minOccurs="0" />
                      <xs:element name="_VERSION" msdata:Caption="Version" type="xs:string"
minOccurs="0" />
                      <xs:element name="PROCEDURE_NUM" msdata:Caption="Procedure num"
type="xs:int" minOccurs="0" />
                      <xs:element name="DESCRIPTION" msdata:Caption="Description"
type="xs:string" minOccurs="0" />
                      <xs:element name="DATA_TYPE" msdata:Caption="Data type"
type="xs:string" minOccurs="0" />
                      <xs:element name="ACTION_TYPE" msdata:Caption="Action type"
type="xs:string" minOccurs="0" />
                      <xs:element name="MNEMONIC" msdata:Caption="Mnemonic"
type="xs:string" minOccurs="0" />
                      <xs:element name="TABLE_NAME" msdata:Caption="Table name"
type="xs:string" minOccurs="0" />
                      <xs:element name="IMPLEMENTATION_TYPE"
msdata:Caption="Implementation type" type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
                <xs:element name="EXPLORER_FOLDER">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="CABINET" msdata:Caption="Cabinet" type="xs:string"
minOccurs="0" />
                      <xs:element name="FOLDER_NUMBER" msdata:Caption="Folder number"
type="xs:int" minOccurs="0" />
                      <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
```

```

        <xs:element name="DESCRIPTION" msdata:Caption="Description"
type="xs:string" minOccurs="0" />
        <xs:element name="FOLDER_ICON_1" msdata:Caption="Folder icon 1"
type="xs:string" minOccurs="0" />
        <xs:element name="FOLDER_ICON_2" msdata:Caption="Folder icon 2"
type="xs:string" minOccurs="0" />
        <xs:element name="DATA_ICON" msdata:Caption="Data icon"
type="xs:string" minOccurs="0" />
        <xs:element name="CRITERIA" msdata:Caption="Criteria"
type="xs:string" minOccurs="0" />
        <xs:element name="TABLE_NAME" msdata:Caption="Table name"
type="xs:string" minOccurs="0" />
        <xs:element name="COLUMN_MODE" msdata:Caption="Column mode"
type="xs:string" minOccurs="0" />
        <xs:element name="GROUP_ID" msdata:Caption="Group id"
type="xs:string" minOccurs="0" />
        <xs:element name="ORDER_NUMBER" msdata:Caption="Order number"
type="xs:int" minOccurs="0" />
        <xs:element name="AUTOSIZE" msdata:Caption="Autosize"
type="xs:boolean" minOccurs="0" />
        <xs:element name="AUTO_LABTABLE" msdata:Caption="Auto labtable"
type="xs:boolean" minOccurs="0" />
        <xs:element name="USER_LIBRARY" msdata:Caption="User library"
type="xs:string" minOccurs="0" />
        <xs:element name="DATA_ICON_ROUTINE" msdata:Caption="Data icon
routine" type="xs:string" minOccurs="0" />
        <xs:element name="CUSTOM_ROUTINE" msdata:Caption="Custom routine"
type="xs:string" minOccurs="0" />
        <xs:element name="DATA_ROUTINE" msdata:Caption="Data routine"
type="xs:string" minOccurs="0" />
        <xs:element name="CRITERIA_ROUTINE" msdata:Caption="Criteria routine"
type="xs:string" minOccurs="0" />
        <xs:element name="LEAVE_ROUTINE" msdata:Caption="Leave routine"
type="xs:string" minOccurs="0" />
        <xs:element name="TREE_ICON_ROUTINE" msdata:Caption="Tree icon
routine" type="xs:string" minOccurs="0" />
        <xs:element name="GROUP_EXCLUSIVE" msdata:Caption="Group exclusive"
type="xs:boolean" minOccurs="0" />
        <xs:element name="HIERARCHY" msdata:Caption="Hierarchy"
type="xs:string" minOccurs="0" />
        <xs:element name="CRITERIA_SAVED_IDENTITY" msdata:Caption="Criteria
saved identity" type="xs:string" minOccurs="0" />
        <xs:element name="EXPLORER_FORM" msdata:Caption="Explorer form"
type="xs:string" minOccurs="0" />
        <xs:element name="EXPLORER_FORM_TASK" msdata:Caption="Explorer form
task" type="xs:string" minOccurs="0" />
        <xs:element name="SHOW_VERSIONS" msdata:Caption="Show versions"
type="xs:boolean" minOccurs="0" />
        <xs:element name="SHOW_REMOVED" msdata:Caption="Show removed"
type="xs:boolean" minOccurs="0" />
        <xs:element name="DEFAULT_REPORT" msdata:Caption="Default report"
type="xs:string" minOccurs="0" />
        <xs:element name="INCLUDE_IN_TREE" msdata:Caption="Include in tree"
type="xs:boolean" minOccurs="0" />
        <xs:element name="TREE_DISPLAY_FIELD" msdata:Caption="Tree display
field" type="xs:string" minOccurs="0" />

```

```

        <xs:element name="CHILD_FORM" msdata:Caption="Child form"
type="xs:string" minOccurs="0" />
        <xs:element name="CHILD_FORM_TASK" msdata:Caption="Child form task"
type="xs:string" minOccurs="0" />
        <xs:element name="DEFAULT_ACTION" msdata:Caption="Default action"
type="xs:int" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<MASTER_MENU>
    <_ICON_ID>INT_NEW</_ICON_ID>
    <_ICON_TEXT>New.</_ICON_TEXT>
    <_NAME>1</_NAME>
    <_KEY0>1</_KEY0>
    <_VERSION>1</_VERSION>
    <PROCEDURE_NUM>1</PROCEDURE_NUM>
    <DESCRIPTION>Add Analysis</DESCRIPTION>
    <DATA_TYPE>STATIC</DATA_TYPE>
    <ACTION_TYPE>ADD</ACTION_TYPE>
    <MNEMONIC>AAL</MNEMONIC>
    <TABLE_NAME>VERSIONED_ANALYSIS</TABLE_NAME>
    <IMPLEMENTATION_TYPE>DESIGNER</IMPLEMENTATION_TYPE>
</MASTER_MENU>
<MASTER_MENU>
    <_ICON_ID>INT_NEW</_ICON_ID>
    <_ICON_TEXT>New.</_ICON_TEXT>
    <_NAME>2</_NAME>
    <_KEY0>2</_KEY0>
    <_VERSION>1</_VERSION>
    <PROCEDURE_NUM>2</PROCEDURE_NUM>
    <DESCRIPTION>Add Customer</DESCRIPTION>
    <DATA_TYPE>STATIC</DATA_TYPE>
    <ACTION_TYPE>ADD</ACTION_TYPE>
    <MNEMONIC>ACL</MNEMONIC>
    <TABLE_NAME>CUSTOMER</TABLE_NAME>
    <IMPLEMENTATION_TYPE>DESIGNER</IMPLEMENTATION_TYPE>
</MASTER_MENU>
<MASTER_MENU>
    <_ICON_ID>INT_NEW</_ICON_ID>
    <_ICON_TEXT>New.</_ICON_TEXT>
    <_NAME>3</_NAME>
    <_KEY0>3</_KEY0>
    <_VERSION>1</_VERSION>
    <PROCEDURE_NUM>3</PROCEDURE_NUM>
    <DESCRIPTION>Add Report Format</DESCRIPTION>
    <DATA_TYPE>SYSTEM</DATA_TYPE>
    <ACTION_TYPE>ADD</ACTION_TYPE>
    <MNEMONIC>AFL</MNEMONIC>
    <TABLE_NAME>REPORT</TABLE_NAME>
    <IMPLEMENTATION_TYPE>VGL</IMPLEMENTATION_TYPE>
</MASTER_MENU>

```

```

<MASTER_MENU>
  <_ICON_ID>INT_NEW</_ICON_ID>
  <_ICON_TEXT>New.</_ICON_TEXT>
  <_NAME>4</_NAME>
  <_KEY0>4</_KEY0>
  <_VERSION>1</_VERSION>
  <PROCEDURE_NUM>4</PROCEDURE_NUM>
  <DESCRIPTION>Add Hazard</DESCRIPTION>
  <DATA_TYPE>STATIC</DATA_TYPE>
  <ACTION_TYPE>ADD</ACTION_TYPE>
  <MNEMONIC>AHL</MNEMONIC>
  <TABLE_NAME>HAZARD</TABLE_NAME>
  <IMPLEMENTATION_TYPE>DESIGNER</IMPLEMENTATION_TYPE>
</MASTER_MENU>
<MASTER_MENU>
  <_ICON_ID>INT_NEW</_ICON_ID>
  <_ICON_TEXT>New.</_ICON_TEXT>
  <_NAME>5</_NAME>
  <_KEY0>5</_KEY0>
  <_VERSION>1</_VERSION>
  <PROCEDURE_NUM>5</PROCEDURE_NUM>
  <DESCRIPTION>Add Instrument</DESCRIPTION>
  <DATA_TYPE>STATIC</DATA_TYPE>
  <ACTION_TYPE>ADD</ACTION_TYPE>
  <MNEMONIC>AIL</MNEMONIC>
  <TABLE_NAME>INSTRUMENT</TABLE_NAME>
  <IMPLEMENTATION_TYPE>DESIGNER</IMPLEMENTATION_TYPE>
</MASTER_MENU>
<EXPLORER_FOLDER>
  <CABINET>SYSTEM</CABINET>
  <FOLDER_NUMBER>7</FOLDER_NUMBER>
  <NAME>Master Menu</NAME>
  <DESCRIPTION>Master Menu</DESCRIPTION>
  <FOLDER_ICON_1>INT_CLOSED_FOLDER</FOLDER_ICON_1>
  <FOLDER_ICON_2>INT_OPEN_FOLDER</FOLDER_ICON_2>
  <DATA_ICON />
  <CRITERIA />
  <TABLE_NAME>MASTER_MENU</TABLE_NAME>
  <COLUMN_MODE>CUSTOM</COLUMN_MODE>
  <GROUP_ID />
  <ORDER_NUMBER>7</ORDER_NUMBER>
  <AUTOSIZE>false</AUTOSIZE>
  <AUTO_LABTABLE>false</AUTO_LABTABLE>
  <USER_LIBRARY>$MENU_MASTER_PROMPT</USER_LIBRARY>
  <DATA_ICON_ROUTINE>STD_PROMPTMENU_GETICON</DATA_ICON_ROUTINE>
  <CUSTOM_ROUTINE />
  <DATA_ROUTINE />
  <CRITERIA_ROUTINE />
  <LEAVE_ROUTINE />
  <TREE_ICON_ROUTINE>STD_PROMPTMENU_GETTREEICON</TREE_ICON_ROUTINE>
  <GROUP_EXCLUSIVE>true</GROUP_EXCLUSIVE>
  <HIERARCHY />
  <CRITERIA_SAVED_IDENTITY />
  <EXPLORER_FORM />
  <EXPLORER_FORM_TASK />
  <SHOW_VERSIONS>false</SHOW_VERSIONS>
  <SHOW_REMOVED>false</SHOW_REMOVED>

```

```
<DEFAULT_REPORT />
<INCLUDE_IN_TREE>false</INCLUDE_IN_TREE>
<TREE_DISPLAY_FIELD />
<CHILD_FORM />
<CHILD_FORM_TASK />
<DEFAULT_ACTION>0</DEFAULT_ACTION>
</EXPLORER_FOLDER>
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## Explorer Folder Get Rmb

Entity: EXPLORER\_FOLDER

Action: GET\_RMB

Master Menu Procedure Number: 15624

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_folder\_get\_rmb

Explorer Folder Get Rmb returns a dataset containing the popup menu options for the folder.

The following sections describe the parameters for Explorer Folder Get Rmb.

### CABINET\_ID

The identity of the cabinet that contains the folder.

This parameter is mandatory.

### FOLDER\_NUMBER

The explorer folder number.

This parameter is mandatory.

## Explorer Folder Set Columns

Entity: EXPLORER\_FOLDER

Action: SET\_COLUMNS

Master Menu Procedure Number: 15617

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_folder\_set\_columns

Explorer Folder Set Columns saves changes to the set of displayed columns in an explorer folder.

The following sections describe the parameters for Explorer Folder Set Columns.

### CABINET\_ID

The identity of the explorer cabinet that contains the folder.

This parameter is mandatory.

### FOLDER\_NUMBER

The number of the explorer folder.

This parameter is mandatory.

## Explorer Rmb Check Context

Entity: EXPLORER\_RMB

Action: CHECK\_CONTEXT

Master Menu Procedure Number: 15628

VGL Library: \$LIMSML\_EXPLORER

Routine: explorer\_rmb\_check\_context

Explorer Rmb Check Context returns two datasets for the specified explorer folder and popup menu item. One dataset contains items in the explorer folder that the popup menu item can be executed on. The other dataset contains items in the explorer folder that the popup menu item cannot be executed on.

The following sections describe the parameters for Explorer Rmb Check Context.

## **CABINET\_ID**

The identity of the explorer cabinet.

This parameter is mandatory.

## **FOLDER\_NUMBER**

The number of the explorer folder.

This parameter is mandatory.

## **RMB\_NUMBER**

The number of the popup menu item.

This parameter is mandatory.

## **Generic Browse**

Entity: GENERIC

Action: BROWSE

Master Menu Procedure Number: 15573

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_browse

Returns a dataset of items for a browse.

The following sections describe the parameters for Generic Browse.

## **ALL\_VERSIONS**

If ALL\_VERSIONS is TRUE, all versions of each item will be included in the returned list. If ALL\_VERSIONS is FALSE, only the highest version that meets the other requirements of the browse will be included.

This parameter is optional.

## **CRITERIA**

The identity of the saved criteria to use to select the records.

This parameter is optional.

## **FIELD**

The name of the field to browse on.

This parameter is mandatory.

## **FILTER\_1\_PROPERTY**

Property to use in filtering results.

This parameter is optional.

## **FILTER\_1\_VALUE**

The value that each item's property must be equal to in order to be included in the browse.

This parameter is optional.

## **FILTER\_2\_PROPERTY**

Property to use in filtering results.

This parameter is optional.

## **FILTER\_2\_VALUE**

The value that each item's property must be equal to in order to be included in the browse.

This parameter is optional.

## **ONLY\_APPROVED**

If ONLY\_APPROVED is TRUE, only the approved versions will be included in the dataset. If ONLY\_APPROVED is FALSE, all versions will be included.

This parameter is optional.

## **PAGE**

The page number to display.

This parameter is optional.

## **PAGESIZE**

The number of items to fit on a page.

This parameter is optional.

## **SCHEMA**

If the SCHEMA parameter is TRUE, the schema is included in the output.

This parameter is optional.

## **SORT\_ASCENDING**

If SORT\_ASCENDING is TRUE, the records in the dataset are sorted in ascending order by the value in the column specified in the SORT\_BY parameter. If SORT\_ASCENDING is FALSE, the records in the dataset are sorted in descending order by the value in the column specified in the SORT\_BY parameter.

This parameter is optional.

## **SORT\_BY**

The column to use to sort the records in the dataset.

This parameter is optional.

## **STARTTEXT**

The text to indicate what filtering to apply to the browse initially.

This parameter is optional.

## Example

The following LIMSML request uses the GENERIC BROWSE command to return the second page of the browse list of VGL Libraries, where the Identity field begins with "\$LIB". Each page of the browse contains 5 entries.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="VGL_LIBRARY">
      <actions>
        <action>
          <command>BROWSE</command>
          <parameter name="PAGE">2</parameter>
          <parameter name="PAGESIZE">5</parameter>
          <parameter name="SCHEMA">True</parameter>
          <parameter name="FIELD">IDENTITY</parameter>
          <parameter name="STARTTEXT">$LIB</parameter>
          <parameter name="FILTER_1_PROPERTY" />
          <parameter name="FILTER_1_VALUE" />
          <parameter name="FILTER_2_PROPERTY" />
          <parameter name="FILTER_2_VALUE" />
          <parameter name="ALL_VERSIONS">False</parameter>
          <parameter name="ONLY_APPROVED">True</parameter>
          <parameter name="CRITERIA" />
          <parameter name="SORT_ASCENDING">False</parameter>
          <parameter name="SORT_BY" />
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="VGL_LIBRARY">
                  <xs:complexType>
                    <xs:sequence>
```

```

        <xs:element name="IDENTITY" msdata:Caption="Identity"
type="xs:string" minOccurs="0" />
        <xs:element name="_ICON_ID" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
        <xs:element name="_ICON_TEXT" msdata:Caption="Icon Text"
type="xs:string" minOccurs="0" />
        <xs:element name="_NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
        <xs:element name="_KEY0" msdata:Caption="Key0" type="xs:string"
minOccurs="0" />
        <xs:element name="_VERSION" msdata:Caption="Version" type="xs:string"
minOccurs="0" />
        <xs:element name="NAME" msdata:Caption="Name
" type="xs:string" minOccurs="0" />
        <xs:element name="TITLE" msdata:Caption="Title
" type="xs:string" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="_RECENT_ITEMS">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="IDENTITY" msdata:Caption="Identity"
type="xs:string" minOccurs="0" />
            <xs:element name="_ICON_ID" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
            <xs:element name="_ICON_TEXT" msdata:Caption="Icon Text"
type="xs:string" minOccurs="0" />
            <xs:element name="_NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
            <xs:element name="_KEY0" msdata:Caption="Key0" type="xs:string"
minOccurs="0" />
            <xs:element name="_VERSION" msdata:Caption="Version" type="xs:string"
minOccurs="0" />
            <xs:element name="NAME" msdata:Caption="Name
" type="xs:string" minOccurs="0" />
            <xs:element name="TITLE" msdata:Caption="Title
" type="xs:string" minOccurs="0" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="_FAVOURITE_ITEMS">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="IDENTITY" msdata:Caption="Identity"
type="xs:string" minOccurs="0" />
            <xs:element name="_ICON_ID" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
            <xs:element name="_ICON_TEXT" msdata:Caption="Icon Text"
type="xs:string" minOccurs="0" />
            <xs:element name="_NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
            <xs:element name="_KEY0" msdata:Caption="Key0" type="xs:string"
minOccurs="0" />
            <xs:element name="_VERSION" msdata:Caption="Version" type="xs:string"
minOccurs="0" />

```

```

        <xs:element name="NAME" msdata:Caption="Name"
" type="xs:string" minOccurs="0" />
        <xs:element name="TITLE" msdata:Caption="Title"
" type="xs:string" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<VGL_LIBRARY>
    <IDENTITY>$LIB_CALCULATOR</IDENTITY>
    <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
    <_ICON_TEXT>VGL Report</_ICON_TEXT>
    <_NAME>$lib calculator</_NAME>
    <_KEY0>$LIB_CALCULATOR    </_KEY0>
    <_VERSION>1</_VERSION>
    <NAME>$lib Calculator</NAME>
    <TITLE>Library Routines for Numeric Calculator</TITLE>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$LIB_CALC_WINDOWS</IDENTITY>
    <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
    <_ICON_TEXT>VGL Report</_ICON_TEXT>
    <_NAME>$lib calc windows</_NAME>
    <_KEY0>$LIB_CALC_WINDOWS    </_KEY0>
    <_VERSION>1</_VERSION>
    <NAME>$lib Calc windows</NAME>
    <TITLE>Library Routines to Display the Calculator</TITLE>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$LIB_CALLBACK</IDENTITY>
    <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
    <_ICON_TEXT>VGL Report</_ICON_TEXT>
    <_NAME>$lib callback</_NAME>
    <_KEY0>$LIB_CALLBACK    </_KEY0>
    <_VERSION>1</_VERSION>
    <NAME>$lib Callback</NAME>
    <TITLE>Library to support callbacks</TITLE>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$LIB_CHARG</IDENTITY>
    <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
    <_ICON_TEXT>VGL Report</_ICON_TEXT>
    <_NAME>$lib charg</_NAME>
    <_KEY0>$LIB_CHARG    </_KEY0>
    <_VERSION>1</_VERSION>
    <NAME>$lib Charg</NAME>
    <TITLE>Library Routines to calculate Charges</TITLE>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$LIB_CHOOSE_ARRAY</IDENTITY>
    <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
    <_ICON_TEXT>VGL Report</_ICON_TEXT>
    <_NAME>$lib choose array</_NAME>
    <_KEY0>$LIB_CHOOSE_ARRAY    </_KEY0>

```

## 2 Standard LIMSML Entity Actions

Other LIMSML Entity Actions

```
<_VERSION>1</_VERSION>
<NAME>$lib Choose Array</NAME>
<TITLE />
</VGL_LIBRARY>
<_RECENT_ITEMS>
  <IDENTITY>CAL_LIMSML_TEST</IDENTITY>
  <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
  <_ICON_TEXT>VGL Report</_ICON_TEXT>
  <_NAME>Cal limsm1 test</_NAME>
  <_KEY0>CAL_LIMSML_TEST</_KEY0>
  <_VERSION>1</_VERSION>
  <NAME>Cal Limsm1 Test</NAME>
  <TITLE>Test utility for LIMSML processor</TITLE>
</_RECENT_ITEMS>
<_RECENT_ITEMS>
  <IDENTITY>$LIMSML_TEST</IDENTITY>
  <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
  <_ICON_TEXT>VGL Report</_ICON_TEXT>
  <_NAME>$limsm1 test</_NAME>
  <_KEY0>$LIMSML_TEST</_KEY0>
  <_VERSION>1</_VERSION>
  <NAME>$limsm1 Test</NAME>
  <TITLE>Test utility for LIMSML processor</TITLE>
</_RECENT_ITEMS>
<_RECENT_ITEMS>
  <IDENTITY>$LIMSML_SAMPLE</IDENTITY>
  <_ICON_ID>INT_VGL_REPORT</_ICON_ID>
  <_ICON_TEXT>VGL Report</_ICON_TEXT>
  <_NAME>$limsm1 sample</_NAME>
  <_KEY0>$LIMSML_SAMPLE</_KEY0>
  <_VERSION>1</_VERSION>
  <NAME>$limsm1 Sample</NAME>
  <TITLE>Sample-specific LIMSML actions</TITLE>
</_RECENT_ITEMS>
</NewDataSet>
</ADODDataSet>
</data>
</transaction>
```

## Generic Browse Count

Entity: GENERIC

Action: BROWSE\_COUNT

Master Menu Procedure Number: 15574

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_browse\_count

Returns the total number of items that can be retrieved for a browse.

The following sections describe the parameters for Generic Browse Count.

### ALL\_VERSIONS

If ALL\_VERSIONS is TRUE, all versions of each item will be included in the returned list. If ALL\_VERSIONS is FALSE, only the highest version that meets the other requirements of the browse will be included.

This parameter is optional.

### CRITERIA

The identity of the saved criteria to use to select the records.

This parameter is optional.

### FIELD

The name of the field to browse on.

This parameter is mandatory.

### FILTER\_1\_PROPERTY

Property to use in filtering results.

This parameter is optional.

### FILTER\_1\_VALUE

The value that each item's property must be equal to in order to be included in the browse.

This parameter is optional.

## **FILTER\_2\_PROPERTY**

Property to use in filtering results.

This parameter is optional.

## **FILTER\_2\_VALUE**

The value that each item's property must be equal to in order to be included in the browse.

This parameter is optional.

## **ONLY\_APPROVED**

If ONLY\_APPROVED is TRUE, only the approved versions will be included in the dataset. If ONLY\_APPROVED is FALSE, all versions will be included.

This parameter is optional.

## **STARTTEXT**

The text to indicate what filtering to apply to the browse initially.

This parameter is optional.

## **Example**

The following LIMSML request uses the BROWSE\_COUNT command to return the number of VGL Libraries where the Identity field begins with "\$LIB".

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="VGL_LIBRARY">
      <actions>
        <action>
          <command>BROWSE_COUNT</command>
          <parameter name="FIELD">IDENTITY</parameter>
          <parameter name="STARTTEXT">$LIB</parameter>
          <parameter name="FILTER_1_PROPERTY" />
          <parameter name="FILTER_1_VALUE" />
          <parameter name="FILTER_2_PROPERTY" />
          <parameter name="FILTER_2_VALUE" />
          <parameter name="ALL_VERSIONS">False</parameter>
          <parameter name="ONLY_APPROVED">True</parameter>
          <parameter name="CRITERIA" />
        </action>
      </actions>
    </fields />
  </entity>
</transaction>
```

```

    <children />
  </entity>
</system>
</transaction>

```

The request above produces the following type of response.

```

<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="VGL_LIBRARY">
      <actions>
        <action>
          <command>BROWSE_COUNT</command>
          <parameter name="FIELD">IDENTITY</parameter>
          <parameter name="STARTTEXT">$LIB</parameter>
          <parameter name="FILTER_1_PROPERTY" />
          <parameter name="FILTER_1_VALUE" />
          <parameter name="FILTER_2_PROPERTY" />
          <parameter name="FILTER_2_VALUE" />
          <parameter name="ALL_VERSIONS">False</parameter>
          <parameter name="ONLY_APPROVED">True</parameter>
          <parameter name="CRITERIA" />
        </action>
      </actions>
      <fields>
        <field id="RETURN" direction="inout" datatype="Text">479</field>
      </fields>
    </children />
  </entity>
</system>
</transaction>

```

## Generic Create

Entity: GENERIC

Action: CREATE

Master Menu Procedure Number: 15642

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_create

Creates a new record.

The following section describes the parameter for Generic Create.

## IDENTITY

The identity of the record to create.

This parameter is mandatory.

### Example

The following LIMSML request uses the CREATE command to create a new phrase with the identity LIMSML that has a single phrase entry.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="PHRASE">
      <actions>
        <action>
          <command>CREATE</command>
          <parameter name="IDENTITY">LIMSML</parameter>
        </action>
      </actions>
      <fields>
        <field id="DESCRIPTION" direction="in" datatype="Text">Created - 27-SEP-2013
10:32:58.23</field>
      </fields>
      <children>
        <entity type="PHRASE_ENTRY">
          <actions />
          <fields>
            <field id="PHRASE_ID" direction="in" datatype="Text">MAE-1</field>
            <field id="PHRASE_TEXT" direction="in" datatype="Text">Created - 27-SEP-2013
10:32:58.23</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0" />
</transaction>
```

## Generic Favorites Add

Entity: GENERIC

Action: FAVORITES\_ADD

Master Menu Procedure Number: 15619

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_favorites\_add

Adds the specified record to favorites.

The following section describes the parameter for Generic Favorites Add.

### KEY0

The key0 for the record to be added to favorites.

This parameter is mandatory.

### Example

The following LIMSML request uses the FAVORITES\_ADD command to add the BLANK sample template to the Favorites list.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions>
        <action>
          <command>FAVORITES_ADD</command>
          <parameter name="KEY0">BLANK </parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions>
        <action>
          <command>FAVORITES_ADD</command>
          <parameter name="KEY0">BLANK </parameter>
        </action>
      </actions>
      <fields>
        <field id="RETURN" direction="inout" datatype="Text">True</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Generic Favorites Remove

Entity: GENERIC

Action: FAVORITES\_REMOVE

Master Menu Procedure Number: 15629

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_favorites\_remove

Removes the specified record from favorites.

The following section describes the parameter for Generic Favorites Remove.

### KEY0

The Key0 for the record to be removed.

This parameter is mandatory.

## Example

The following LIMSML request uses the FAVORITES\_REMOVE command to remove the BLANK sample template from the Favorites list.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions>
        <action>
          <command>FAVORITES_REMOVE</command>
          <parameter name="KEY0">BLANK </parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE_TEMPLATE">
      <actions>
        <action>
          <command>FAVORITES_REMOVE</command>
          <parameter name="KEY0">BLANK </parameter>
        </action>
      </actions>
      <fields>
        <field id="RETURN" direction="inout" datatype="Text">True</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Generic Find

Entity: GENERIC

Action: FIND

Master Menu Procedure Number: 15587

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_find

Returns a dataset containing all items and all columns of an entity type.

The following sections describe the parameters for Generic Find.

### PAGE

The page number to display.

This parameter is optional.

### PAGESIZE

The number of items to fit on a page.

This parameter is optional.

### SCHEMA

If the SCHEMA parameter is TRUE, the schema is included in the output.

This parameter is optional.

## Example

The following LIMSML request uses the FIND command to return the first page of the list of VGL Libraries. Each page of the list contains 5 entries.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="VGL_LIBRARY">
      <actions>
        <action>
          <command>FIND</command>
          <parameter name="PAGE">1</parameter>
          <parameter name="PAGESIZE">5</parameter>
          <parameter name="SCHEMA">True</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="VGL_LIBRARY">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="IDENTITY" msdata:Caption="Identity"
type="xs:string" minOccurs="0" />
                      <xs:element name="GROUP_ID" msdata:Caption="Group id"
type="xs:string" minOccurs="0" />
                      <xs:element name="TITLE" msdata:Caption="Title" type="xs:string"
minOccurs="0" />
                      <xs:element name="MODIFIED_ON" msdata:Caption="Modified on"
type="xs:dateTime" minOccurs="0" />
                      <xs:element name="MODIFIED_BY" msdata:Caption="Modified by"
type="xs:string" minOccurs="0" />
                      <xs:element name="MODIFIABLE" msdata:Caption="Modifiable"
type="xs:boolean" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

```

        <xs:element name="REMOVEFLAG" msdata:Caption="Removeflag"
type="xs:boolean" minOccurs="0" />
        <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
        <xs:element name="HAS_ATTACHMENTS" msdata:Caption="Has attachments"
type="xs:boolean" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<VGL_LIBRARY>
    <IDENTITY>$ACTIVE</IDENTITY>
    <GROUP_ID />
    <TITLE>Report on Active Samples</TITLE>
    <MODIFIED_ON>2012-10-24T03:28:30+01:00</MODIFIED_ON>
    <MODIFIED_BY>BATCH</MODIFIED_BY>
    <MODIFIABLE>true</MODIFIABLE>
    <REMOVEFLAG>>false</REMOVEFLAG>
    <NAME>$active</NAME>
    <HAS_ATTACHMENTS>>false</HAS_ATTACHMENTS>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$ADDREPORT</IDENTITY>
    <GROUP_ID />
    <TITLE>Adds VGL reports to database</TITLE>
    <MODIFIED_ON>2012-10-24T03:28:30+01:00</MODIFIED_ON>
    <MODIFIED_BY>BATCH</MODIFIED_BY>
    <MODIFIABLE>true</MODIFIABLE>
    <REMOVEFLAG>>false</REMOVEFLAG>
    <NAME>$addreport</NAME>
    <HAS_ATTACHMENTS>>false</HAS_ATTACHMENTS>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$APPROVAL</IDENTITY>
    <GROUP_ID>SYSTEM</GROUP_ID>
    <TITLE>Operator Approval table editor</TITLE>
    <MODIFIED_ON>2012-10-24T03:28:30+01:00</MODIFIED_ON>
    <MODIFIED_BY>BATCH</MODIFIED_BY>
    <MODIFIABLE>true</MODIFIABLE>
    <REMOVEFLAG>>false</REMOVEFLAG>
    <NAME>$approval</NAME>
    <HAS_ATTACHMENTS>>false</HAS_ATTACHMENTS>
</VGL_LIBRARY>
<VGL_LIBRARY>
    <IDENTITY>$AQL</IDENTITY>
    <GROUP_ID />
    <TITLE>Library of Statistical Sampling Routines</TITLE>
    <MODIFIED_ON>2012-10-24T03:28:30+01:00</MODIFIED_ON>
    <MODIFIED_BY>BATCH</MODIFIED_BY>
    <MODIFIABLE>true</MODIFIABLE>
    <REMOVEFLAG>>false</REMOVEFLAG>
    <NAME>$aql</NAME>
    <HAS_ATTACHMENTS>>false</HAS_ATTACHMENTS>
</VGL_LIBRARY>

```

```

<VGL_LIBRARY>
  <IDENTITY>$ARCHIVE</IDENTITY>
  <GROUP_ID>SYSTEM</GROUP_ID>
  <TITLE>Committed table archiving utility</TITLE>
  <MODIFIED_ON>2012-10-24T03:28:30+01:00</MODIFIED_ON>
  <MODIFIED_BY>BATCH</MODIFIED_BY>
  <MODIFIABLE>true</MODIFIABLE>
  <REMOVEFLAG>>false</REMOVEFLAG>
  <NAME>$archive</NAME>
  <HAS_ATTACHMENTS>>false</HAS_ATTACHMENTS>
</VGL_LIBRARY>
</NewDataSet>
</ADODataset>
</data>
</transaction>

```

## Generic Get By Identity

Entity: GENERIC

Action: GET\_BY\_IDENTITY

Master Menu Procedure Number: 15575

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_get\_by\_identity

Generic Get By Identity returns the item with the supplied key0.

The following sections describe the parameters for Generic Get By Identity.

### IDENTITY

The padded record identity, without a version number.

This parameter is mandatory.

### SCHEMA

If the SCHEMA parameter is TRUE, the schema is included in the output.

This parameter is optional.

### VERSION

The version number of the entity.

This parameter is optional.

## Example

The following LIMSML request uses the GET\_BY\_IDENTITY command to return the \$ACTIVE VGL Library.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="VGL_LIBRARY">
      <actions>
        <action>
          <command>GET_BY_IDENTITY</command>
          <parameter name="IDENTITY">$ACTIVE</parameter>
          <parameter name="VERSION">1</parameter>
          <parameter name="SCHEMA">True</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="VGL_LIBRARY">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="IDENTITY" msdata:Caption="Identity"
type="xs:string" minOccurs="0" />
                      <xs:element name="GROUP_ID" msdata:Caption="Group id"
type="xs:string" minOccurs="0" />
                      <xs:element name="TITLE" msdata:Caption="Title" type="xs:string"
minOccurs="0" />
                      <xs:element name="MODIFIED_ON" msdata:Caption="Modified on"
type="xs:dateTime" minOccurs="0" />
                      <xs:element name="MODIFIED_BY" msdata:Caption="Modified by"
type="xs:string" minOccurs="0" />
                      <xs:element name="MODIFIABLE" msdata:Caption="Modifiable"
type="xs:boolean" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

```

        <xs:element name="REMOVEFLAG" msdata:Caption="Removeflag"
type="xs:boolean" minOccurs="0" />
        <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
        <xs:element name="HAS_ATTACHMENTS" msdata:Caption="Has attachments"
type="xs:boolean" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<VGL_LIBRARY>
  <IDENTITY>$ACTIVE</IDENTITY>
  <GROUP_ID />
  <TITLE>Report on Active Samples</TITLE>
  <MODIFIED_ON>2012-10-24T03:28:30+01:00</MODIFIED_ON>
  <MODIFIED_BY>BATCH</MODIFIED_BY>
  <MODIFIABLE>true</MODIFIABLE>
  <REMOVEFLAG>>false</REMOVEFLAG>
  <NAME>$active</NAME>
  <HAS_ATTACHMENTS>>false</HAS_ATTACHMENTS>
</VGL_LIBRARY>
</NewDataSet>
</ADODataset>
</data>
</transaction>

```

## Generic Get Columns

Entity: GENERIC

Action: GET\_COLUMNS

Master Menu Procedure Number: 15614

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_get\_columns

Generic Get Columns returns a dataset that contains the available columns for the selected type of entity.

The following sections describe the parameters for Generic Get Columns.

### AVAILABLE

Indicates whether all fields or only the user selected fields are to be returned.

This parameter is mandatory.

## Example

The following LIMSML request uses the GET\_COLUMNS command to return all the columns in the JOB\_HEADER table.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB_HEADER">
      <actions>
        <action>
          <command>GET_COLUMNS</command>
          <parameter name="AVAILABLE">True</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="JOB_HEADER">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="JOB_NAME" msdata:Caption="Job name"
type="xs:string" minOccurs="0" />
                      <xs:element name="JOB_STATUS" msdata:Caption="Job status"
type="xs:string" minOccurs="0" />
                      <xs:element name="BROWSE_DESCRIPTION" msdata:Caption="Browse
description" type="xs:string" minOccurs="0" />
                      <xs:element name="SAMPLE_SYNTAX_ID" msdata:Caption="Sample syntax id"
type="xs:string" minOccurs="0" />
                      <xs:element name="DATE_CREATED" msdata:Caption="Date created"
type="xs:dateTime" minOccurs="0" />
                      <xs:element name="DATE_COMPLETED" msdata:Caption="Date completed"
type="xs:dateTime" minOccurs="0" />
                      <xs:element name="DATE_RECEIVED" msdata:Caption="Date received"
type="xs:dateTime" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </NewDataSet>
    </ADODDataSet>
  </data>
</transaction>
```

```

        <xs:element name="DATE_TO_START" msdata:Caption="Date to start"
type="xs:dateTime" minOccurs="0" />
        <xs:element name="SUBMITTER_OPER" msdata:Caption="Submitter oper"
type="xs:string" minOccurs="0" />
        <xs:element name="SAMPLES_TO_DO" msdata:Caption="Samples to do"
type="xs:int" minOccurs="0" />
        <xs:element name="GROUP_ID" msdata:Caption="Group id"
type="xs:string" minOccurs="0" />
        <xs:element name="TEMPLATE_ID" msdata:Caption="Template id"
type="xs:string" minOccurs="0" />
        <xs:element name="SAMPLE_TEMPLATE" msdata:Caption="Sample template"
type="xs:string" minOccurs="0" />
        <xs:element name="DATE_AUTHORISED" msdata:Caption="Date authorised"
type="xs:dateTime" minOccurs="0" />
        <xs:element name="AUTHORISER" msdata:Caption="Authoriser"
type="xs:string" minOccurs="0" />
        <xs:element name="AUTHORISATION_NOTES" msdata:Caption="Authorisation
notes" type="xs:string" minOccurs="0" />
        <xs:element name="OLD_STATUS" msdata:Caption="Old status"
type="xs:string" minOccurs="0" />
        <xs:element name="INSPECTION_HEADER" msdata:Caption="Inspection
header" type="xs:string" minOccurs="0" />
        <xs:element name="NODE_TRIGGER_PATH" msdata:Caption="Node trigger
path" type="xs:string" minOccurs="0" />
        <xs:element name="ORIGINAL_JOB" msdata:Caption="Original job"
type="xs:string" minOccurs="0" />
        <xs:element name="BEEN_COPIED" msdata:Caption="Been copied"
type="xs:boolean" minOccurs="0" />
        <xs:element name="AUTO_VALIDATE" msdata:Caption="Auto validate"
type="xs:boolean" minOccurs="0" />
        <xs:element name="TOTAL_ELEMENTS" msdata:Caption="Total elements"
type="xs:int" minOccurs="0" />
        <xs:element name="LOT_ID" msdata:Caption="Lot id" type="xs:string"
minOccurs="0" />
        <xs:element name="CERTIFICATE" msdata:Caption="Certificate"
type="xs:int" minOccurs="0" />
        <xs:element name="WORKFLOW_NODE" msdata:Caption="Workflow node"
type="xs:string" minOccurs="0" />
        <xs:element name="ENTITY_TEMPLATE_ID" msdata:Caption="Entity template
id" type="xs:string" minOccurs="0" />
        <xs:element name="ENTITY_TEMPLATE_VERSION" msdata:Caption="Entity
template version" type="xs:int" minOccurs="0" />
        <xs:element name="HAS_ATTACHMENTS" msdata:Caption="Has attachments"
type="xs:boolean" minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
</NewDataSet>
</ADODDataSet>
</data>
</transaction>

```

## Generic Get Related Items

Entity: GENERIC

Action: GET\_RELATED\_ITEMS

Master Menu Procedure Number: 15576

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_get\_related\_items

Generic Get Related Items returns items related to the specified item using the specified relationship. Relationships are defined in the BL\_RELATIONSHIP table.

The following sections describe the parameters for Generic Get Related Items.

### IDENTITY

The padded record identity, without a version number.

This parameter is mandatory.

### INVERSE\_RELATIONSHIP

If INVERSE\_RELATIONSHIP is TRUE, the relationship is followed from the entity type specified in the destination\_type\_id field of the BL\_RELATIONSHIP table record to the entity type specified in the source\_type\_id of the record. For example, if the PHRASE\_ENTRY record in the BL\_RELATIONSHIP table is used, when INVERSE\_RELATIONSHIP is TRUE, the relationship is from a supplied phrase entry to its parent phrase, and Generic Get Related Items returns the phrase.

If INVERSE\_RELATIONSHIP is FALSE, the relationship is followed from the entity type specified in the source\_type\_id field of the BL\_RELATIONSHIP table record to the entity type specified in the destination\_type\_id of the record. For example, if the PHRASE\_ENTRY record in the BL\_RELATIONSHIP table is used, when INVERSE\_RELATIONSHIP is FALSE, the relationship is from a supplied phrase to its child phrase entries, and Generic Get Related Items returns the phrase entries.

This parameter is optional.

### RELATIONSHIP\_ID

The identity of the relationship. Relationships are stored in the BL\_RELATIONSHIP table.

This parameter is mandatory.

## VERSION

The version number of the entity.

This parameter is optional.

## Example

The following LIMSML request uses the GET\_RELATED\_ITEMS command to return all the phrase entries in the SAMP\_STAT phrase.

```

<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="PHRASE">
      <actions>
        <action>
          <command>GET_RELATED_ITEMS</command>
          <parameter name="IDENTITY">SAMP_STAT</parameter>
          <parameter name="VERSION">1</parameter>
          <parameter name="RELATIONSHIP_ID">PHRASE_ENTRY</parameter>
          <parameter name="INVERSE_RELATIONSHIP">False</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>

```

The request above produces the following type of response.

```

<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADDDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:msns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="PHRASE_ENTRY">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="PHRASE_TYPE" msdata:Caption="Phrase type"
type="xs:string" minOccurs="0" />
                      <xs:element name="PHRASE_ID" msdata:Caption="Phrase id"
type="xs:string" minOccurs="0" />

```

```

minOccurs="0" />
        <xs:element name="ORDER_NUM" msdata:Caption="Order num" type="xs:int"
type="xs:string" minOccurs="0" />
        <xs:element name="PHRASE_TEXT" msdata:Caption="Phrase text"
minOccurs="0" />
        <xs:element name="ICON" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<PHRASE_ENTRY>
    <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
    <PHRASE_ID>U</PHRASE_ID>
    <ORDER_NUM>1</ORDER_NUM>
    <PHRASE_TEXT>Unavailable</PHRASE_TEXT>
    <ICON>INT_SAMPLE_U</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
    <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
    <PHRASE_ID>W</PHRASE_ID>
    <ORDER_NUM>2</ORDER_NUM>
    <PHRASE_TEXT>Awaiting Preparation</PHRASE_TEXT>
    <ICON>INT_SAMPLE_W</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
    <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
    <PHRASE_ID>V</PHRASE_ID>
    <ORDER_NUM>3</ORDER_NUM>
    <PHRASE_TEXT>Available</PHRASE_TEXT>
    <ICON>INT_SAMPLE_V</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
    <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
    <PHRASE_ID>H</PHRASE_ID>
    <ORDER_NUM>4</ORDER_NUM>
    <PHRASE_TEXT>On Hold</PHRASE_TEXT>
    <ICON>INT_SAMPLE_H</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
    <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
    <PHRASE_ID>S</PHRASE_ID>
    <ORDER_NUM>5</ORDER_NUM>
    <PHRASE_TEXT>Suspended</PHRASE_TEXT>
    <ICON>INT_SAMPLE_S</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
    <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
    <PHRASE_ID>C</PHRASE_ID>
    <ORDER_NUM>6</ORDER_NUM>
    <PHRASE_TEXT>Completed</PHRASE_TEXT>
    <ICON>INT_SAMPLE_C</ICON>
</PHRASE_ENTRY>

```

```

<PHRASE_ENTRY>
  <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
  <PHRASE_ID>A</PHRASE_ID>
  <ORDER_NUM>7</ORDER_NUM>
  <PHRASE_TEXT>Authorised</PHRASE_TEXT>
  <ICON>INT_SAMPLE_A</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
  <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
  <PHRASE_ID>R</PHRASE_ID>
  <ORDER_NUM>8</ORDER_NUM>
  <PHRASE_TEXT>Rejected</PHRASE_TEXT>
  <ICON>INT_SAMPLE_R</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
  <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
  <PHRASE_ID>X</PHRASE_ID>
  <ORDER_NUM>9</ORDER_NUM>
  <PHRASE_TEXT>Cancelled</PHRASE_TEXT>
  <ICON>INT_SAMPLE_X</ICON>
</PHRASE_ENTRY>
<PHRASE_ENTRY>
  <PHRASE_TYPE>SAMP_STAT</PHRASE_TYPE>
  <PHRASE_ID>I</PHRASE_ID>
  <ORDER_NUM>10</ORDER_NUM>
  <PHRASE_TEXT>Inspection</PHRASE_TEXT>
  <ICON>INT_SAMPLE_I</ICON>
</PHRASE_ENTRY>
</NewDataSet>
</ADODataset>
</data>
</transaction>

```

## Generic Run Workflow

Entity: GENERIC

Action: RUN\_WORKFLOW

Master Menu Procedure Number: 16003

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_workflow\_run

Generic Run Workflow runs a workflow and returns the data that the workflow creates.

The following sections describe the parameters for Generic Get Run Workflow.

### WORKFLOW\_NAME

The name of the workflow to run.

This parameter is mandatory.

### REPLICATES

The number of times to run the workflow. If the REPLICATES parameter is not specified, the workflow is run once.

This parameter is optional.

### Example

The following LIMSML request uses the RUN\_WORKFLOW command to run the Single Job workflow three times.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="JOB_HEADER">
      <actions>
        <action>
          <command>RUN_WORKFLOW</command>
          <parameter name="WORKFLOW_NAME">Single Job</parameter>
          <parameter name="REPLICATES">3</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODataset>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:msns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="JOB_HEADER">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="_ICON_ID" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
                      <xs:element name="_ICON_TEXT" msdata:Caption="Icon Text"
type="xs:string" minOccurs="0" />
                      <xs:element name="_NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
                      <xs:element name="_KEY0" msdata:Caption="key0" type="xs:string"
minOccurs="0" />
                      <xs:element name="_VERSION" msdata:Caption="Version" type="xs:string"
minOccurs="0" />
                      <xs:element name="JOB_NAME" msdata:Caption="Job name"
type="xs:string" minOccurs="0" />
                      <xs:element name="BROWSE_DESCRIPTION" msdata:Caption="Browse
description" type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
        <JOB_HEADER>
          <_ICON_ID>INT_JOB_HEADER_C</_ICON_ID>
          <_ICON_TEXT>Job Completed</_ICON_TEXT>
          <_NAME>Job 0000000008</_NAME>
          <_KEY0>JOB_0000000008 </_KEY0>
          <_VERSION>1</_VERSION>
          <JOB_NAME>JOB_0000000008</JOB_NAME>
          <BROWSE_DESCRIPTION>Job created on 4-APR-2014 14:14 by SYSTEM using DEFAULT_JOB
entity template</BROWSE_DESCRIPTION>
        </JOB_HEADER>
        <JOB_HEADER>
          <_ICON_ID>INT_JOB_HEADER_C</_ICON_ID>
          <_ICON_TEXT>Job Completed</_ICON_TEXT>
          <_NAME>Job 0000000009</_NAME>
          <_KEY0>JOB_0000000009 </_KEY0>
```

```
<_VERSION>1</_VERSION>
<JOB_NAME>JOB_0000000009</JOB_NAME>
<BROWSE_DESCRIPTION>Job created on 4-APR-2014 14:14 by SYSTEM using DEFAULT_JOB
entity template</BROWSE_DESCRIPTION>
</JOB_HEADER>
<JOB_HEADER>
  <_ICON_ID>INT_JOB_HEADER_C</_ICON_ID>
  <_ICON_TEXT>Job Completed</_ICON_TEXT>
  <_NAME>Job 0000000010</_NAME>
  <_KEY0>JOB_0000000010</_KEY0>
  <_VERSION>1</_VERSION>
  <JOB_NAME>JOB_0000000010</JOB_NAME>
  <BROWSE_DESCRIPTION>Job created on 4-APR-2014 14:14 by SYSTEM using DEFAULT_JOB
entity template</BROWSE_DESCRIPTION>
</JOB_HEADER>
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## Generic Search

Entity: GENERIC

Action: SEARCH

Master Menu Procedure Number: 15644

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_search

Generic Search returns a dataset of items for a search. The optional search criteria can have up to three filter conditions, linked by AND or OR.

The following sections describe the parameters for Generic Search.

### BROWSE\_ONLY

Indicates whether all fields or only the browse fields defined in structure.txt are returned.

This parameter is optional.

### FILTER\_1\_OPERATOR

The operator to use for the first filter statement.

This parameter is optional.

### **FILTER\_1\_PROPERTY**

The property to use in the first filter statement.

This parameter is optional.

### **FILTER\_1\_VALUE**

The value to use in the first filter statement.

This parameter is optional.

### **FILTER\_2\_OPERATOR**

The operator to use for the second filter statement.

This parameter is optional.

### **FILTER\_2\_PREDICATE**

The operator that defines the logical relationship between the first and second filter statements. The value can be AND or OR.

This parameter is optional.

### **FILTER\_2\_PROPERTY**

The property to use in the second filter statement.

This parameter is optional.

### **FILTER\_2\_VALUE**

The value to use in the second filter statement.

This parameter is optional.

### **FILTER\_3\_OPERATOR**

The operator to use for the third filter statement.

This parameter is optional.

### **FILTER\_3\_PREDICATE**

The operator that defines the logical relationship between the second and third filter statements. The value can be AND or OR.

This parameter is optional.

## **FILTER\_3\_PROPERTY**

The property to use in the third filter statement.

This parameter is optional.

## **FILTER\_3\_VALUE**

The value to use in the third filter statement.

This parameter is optional.

## **PAGE**

The page number to display.

This parameter is optional.

## **PAGESIZE**

The number of items to fit on a page.

This parameter is optional.

## **Example**

The following LIMSML request selects records from the UNIT\_HEADER table that match the following selection criteria:

SYSTEM = "SI"

AND

QUANTITY = "M"

AND

DESCRIPTION LIKE "%gram%"

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="UNIT_HEADER">
      <actions>
        <action>
          <command>SEARCH</command>
          <parameter name="BROWSE_ONLY">True</parameter>
          <parameter name="FILTER_1_PROPERTY">SYSTEM</parameter>
          <parameter name="FILTER_1_OPERATOR">=</parameter>
          <parameter name="FILTER_1_VALUE">SI</parameter>
          <parameter name="FILTER_2_PREDICATE">AND</parameter>
```

```

    <parameter name="FILTER_2_PROPERTY">QUANTITY</parameter>
    <parameter name="FILTER_2_OPERATOR">=</parameter>
    <parameter name="FILTER_2_VALUE">M</parameter>
    <parameter name="FILTER_3_PREDICATE">AND</parameter>
    <parameter name="FILTER_3_PROPERTY">DESCRIPTION</parameter>
    <parameter name="FILTER_3_OPERATOR">LIKE</parameter>
    <parameter name="FILTER_3_VALUE">%gram%</parameter>
  </action>
</actions>
<fields />
<children />
</entity>
</system>
</transaction>

```

The request above produces the following type of response. The response only includes the browse fields from the UNIT\_HEADER table, because the BROWSE\_ONLY parameter was set to True in the request.

```

<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODDataSet>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
          targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
          xmlns:msns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
          xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
          xmlns:xs="http://www.w3.org/2001/XMLSchema"
          xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
          elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="UNIT_HEADER">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="_ICON_ID" msdata:Caption="Icon" type="xs:string"
minOccurs="0" />
                      <xs:element name="_ICON_TEXT" msdata:Caption="Icon Text"
type="xs:string" minOccurs="0" />
                      <xs:element name="_NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
                      <xs:element name="_KEY0" msdata:Caption="key0" type="xs:string"
minOccurs="0" />
                      <xs:element name="_VERSION" msdata:Caption="Version" type="xs:string"
minOccurs="0" />
                      <xs:element name="IDENTITY" msdata:Caption="Unit Symbol"
type="xs:string" minOccurs="0" />
                      <xs:element name="QUANTITY" msdata:Caption="Quantity"
type="xs:string" minOccurs="0" />
                      <xs:element name="DESCRIPTION" msdata:Caption="Common Name"
type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>

```

```

        </xs:choice>
      </xs:complexType>
    </xs:element>
  </xs:schema>
  <UNIT_HEADER>
    <_ICON_ID>INT_THERMOMETER</_ICON_ID>
    <_ICON_TEXT>Thermometer</_ICON_TEXT>
    <_NAME>G</_NAME>
    <_KEY0>g                                     </_KEY0>
    <_VERSION>1</_VERSION>
    <IDENTITY>g</IDENTITY>
    <QUANTITY>M</QUANTITY>
    <DESCRIPTION>gram</DESCRIPTION>
  </UNIT_HEADER>
  <UNIT_HEADER>
    <_ICON_ID>INT_THERMOMETER</_ICON_ID>
    <_ICON_TEXT>Thermometer</_ICON_TEXT>
    <_NAME>Kg</_NAME>
    <_KEY0>kg                                     </_KEY0>
    <_VERSION>1</_VERSION>
    <IDENTITY>kg</IDENTITY>
    <QUANTITY>M</QUANTITY>
    <DESCRIPTION>kilogram</DESCRIPTION>
  </UNIT_HEADER>
  <UNIT_HEADER>
    <_ICON_ID>INT_THERMOMETER</_ICON_ID>
    <_ICON_TEXT>Thermometer</_ICON_TEXT>
    <_NAME>Mg</_NAME>
    <_KEY0>mg                                     </_KEY0>
    <_VERSION>1</_VERSION>
    <IDENTITY>mg</IDENTITY>
    <QUANTITY>M</QUANTITY>
    <DESCRIPTION>milligram</DESCRIPTION>
  </UNIT_HEADER>
  <UNIT_HEADER>
    <_ICON_ID>INT_THERMOMETER</_ICON_ID>
    <_ICON_TEXT>Thermometer</_ICON_TEXT>
    <_NAME>Ng</_NAME>
    <_KEY0>ng                                     </_KEY0>
    <_VERSION>1</_VERSION>
    <IDENTITY>ng</IDENTITY>
    <QUANTITY>M</QUANTITY>
    <DESCRIPTION>nanogram</DESCRIPTION>
  </UNIT_HEADER>
  <UNIT_HEADER>
    <_ICON_ID>INT_THERMOMETER</_ICON_ID>
    <_ICON_TEXT>Thermometer</_ICON_TEXT>
    <_NAME>Pg</_NAME>
    <_KEY0>pg                                     </_KEY0>
    <_VERSION>1</_VERSION>
    <IDENTITY>pg</IDENTITY>
    <QUANTITY>M</QUANTITY>
    <DESCRIPTION>picogram</DESCRIPTION>
  </UNIT_HEADER>

```

```

<UNIT_HEADER>
  <_ICON_ID>INT_THERMOMETER</_ICON_ID>
  <_ICON_TEXT>Thermometer</_ICON_TEXT>
  <_NAME>Ug</_NAME>
  <_KEY0>ug</_KEY0>
  <_VERSION>1</_VERSION>
  <IDENTITY>ug</IDENTITY>
  <QUANTITY>M</QUANTITY>
  <DESCRIPTION>microgram</DESCRIPTION>
</UNIT_HEADER>
</NewDataSet>
</ADODataset>
</data>
</transaction>

```

## Generic Search Count

Entity: GENERIC

Action: SEARCH\_COUNT

Master Menu Procedure Number: 15645

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_search\_count

Generic Search returns a value corresponding to the number of records in the database that match the search criteria. The optional search criteria can have up to three filter conditions, linked by AND or OR.

The following sections describe the parameters for Generic Search Count.

### FILTER\_1\_OPERATOR

The operator to use for the first filter statement.

This parameter is optional.

### FILTER\_1\_PROPERTY

The property to use in the first filter statement.

This parameter is optional.

### FILTER\_1\_VALUE

The value to use in the first filter statement.

This parameter is optional.

## **FILTER\_2\_OPERATOR**

The operator to use for the second filter statement.

This parameter is optional.

## **FILTER\_2\_PREDICATE**

The operator that defines the logical relationship between the first and second filter statements. The value can be AND or OR.

This parameter is optional.

## **FILTER\_2\_PROPERTY**

The property to use in the second filter statement.

This parameter is optional.

## **FILTER\_2\_VALUE**

The value to use in the second filter statement.

This parameter is optional.

## **FILTER\_3\_OPERATOR**

The operator to use for the third filter statement.

This parameter is optional.

## **FILTER\_3\_PREDICATE**

The operator that defines the logical relationship between the second and third filter statements. The value can be AND or OR.

This parameter is optional.

## **FILTER\_3\_PROPERTY**

The property to use in the third filter statement.

This parameter is optional.

## **FILTER\_3\_VALUE**

The value to use in the third filter statement.

This parameter is optional.

## Example

The following LIMSML request returns the number of records from the UNIT\_HEADER table that match the following selection criteria:

SYSTEM = "SI"

AND

QUANTITY = "M"

AND

DESCRIPTION LIKE "%gram%"

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="UNIT_HEADER">
      <actions>
        <action>
          <command>SEARCH_COUNT</command>
          <parameter name="FILTER_1_PROPERTY">SYSTEM</parameter>
          <parameter name="FILTER_1_OPERATOR">=</parameter>
          <parameter name="FILTER_1_VALUE">SI</parameter>
          <parameter name="FILTER_2_PREDICATE">AND</parameter>
          <parameter name="FILTER_2_PROPERTY">QUANTITY</parameter>
          <parameter name="FILTER_2_OPERATOR">=</parameter>
          <parameter name="FILTER_2_VALUE">M</parameter>
          <parameter name="FILTER_3_PREDICATE">AND</parameter>
          <parameter name="FILTER_3_PROPERTY">DESCRIPTION</parameter>
          <parameter name="FILTER_3_OPERATOR">LIKE</parameter>
          <parameter name="FILTER_3_VALUE">%gram%</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="UNIT_HEADER">
      <actions>
        <action>
          <command>SEARCH_COUNT</command>
          <parameter name="FILTER_1_PROPERTY">SYSTEM</parameter>
          <parameter name="FILTER_1_OPERATOR">=</parameter>
          <parameter name="FILTER_1_VALUE">SI</parameter>
          <parameter name="FILTER_2_PREDICATE">AND</parameter>
          <parameter name="FILTER_2_PROPERTY">QUANTITY</parameter>
          <parameter name="FILTER_2_OPERATOR">=</parameter>
          <parameter name="FILTER_2_VALUE">M</parameter>
          <parameter name="FILTER_3_PREDICATE">AND</parameter>
          <parameter name="FILTER_3_PROPERTY">DESCRIPTION</parameter>
          <parameter name="FILTER_3_OPERATOR">LIKE</parameter>
          <parameter name="FILTER_3_VALUE">%gram%</parameter>
        </action>
      </actions>
      <fields>
        <field id="RETURN" direction="inout" datatype="Text">6</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## Generic Set Columns

Entity: GENERIC

Action: SET\_COLUMNS

Master Menu Procedure Number: 15615

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_set\_columns

Generic Set Columns saves a new list of columns for an entity type to the TABLE\_DETAILS\$<USER> cabinet, where <USER> is the user for the current session.

Generic Set Columns has no parameters.

### Example

The following LIMSML request uses the SET\_COLUMNS command to set the default columns for the SAMPLE table for the current user.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>SET_COLUMNS</command>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="in" datatype="Text">ID Numeric</field>
        <field id="ID_TEXT" direction="in" datatype="Text">ID Text</field>
        <field id="STATUS" direction="in" datatype="Text">Current Status</field>
        <field id="LOGIN_BY" direction="in" datatype="Text">Logged in by</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>SET_COLUMNS</command>
        </action>
      </actions>
      <fields>
        <field id="ID_NUMERIC" direction="in" datatype="Text">ID Numeric</field>
        <field id="ID_TEXT" direction="in" datatype="Text">ID Text</field>
        <field id="STATUS" direction="in" datatype="Text">Current Status</field>
        <field id="LOGIN_BY" direction="in" datatype="Text">Logged in by</field>
        <field id="RETURN" direction="inout" datatype="Text">Updated</field>
      </fields>
    </entity>
  </system>
</transaction>
```

## Generic Update

Entity: GENERIC

Action: UPDATE

Master Menu Procedure Number: 15643

VGL Library: \$LIMSML\_GENERIC

Routine: generic\_update

Updates an existing record.

The following section describes the parameter for Generic Update.

### IDENTITY

The identity of the record to update.

This parameter is mandatory.

## Example

The following LIMSML request uses the UPDATE command to:

- Modify the Description of the LIMSML phrase
- Add two new phrase entries to the LIMSML phrase

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="PHRASE">
      <actions>
        <action>
          <command>UPDATE</command>
          <parameter name="IDENTITY">LIMSML</parameter>
        </action>
      </actions>
      <fields>
        <field id="DESCRIPTION" direction="in" datatype="Text">Updated Description text -
27-SEP-2013 15:53:23.37</field>
      </fields>
      <children>
        <entity type="PHRASE_ENTRY">
          <actions />
          <fields>
            <field id="PHRASE_ID" direction="in" datatype="Text">MAE-1</field>
            <field id="PHRASE_TEXT" direction="in" datatype="Text">Updated Entry -
27-SEP-2013 15:53:23.37</field>
          </fields>
          <children />
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0" />
</transaction>
```

## **Limsml Entity Actionget By Identity**

Entity: LIMSML\_ENTITY\_ACTION

Action: GET\_BY\_IDENTITY

Master Menu Procedure Number: 15632

VGL Library: \$LIMSML\_ENTITYACTION

Routine: entity\_action\_get\_by\_identity

Limsml Entity Actionget By Identity returns a dataset containing the LIMSML entity action with the supplied identity. An empty dataset is returned if a record with the specified identity does not exist.

The following sections describe the parameters for Limsml Entity Actionget By Identity.

### **IDENTITY**

The padded identity of the LIMSML entity action.

This parameter is mandatory.

### **SCHEMA**

If the SCHEMA parameter is TRUE, the schema is included in the output.

This parameter is optional.

## **Limsml Entity Actionget By Relationship**

Entity: LIMSML\_ENTITY\_ACTION

Action: GET\_BY\_RELATIONSHIP

Master Menu Procedure Number: 15633

VGL Library: \$LIMSML\_ENTITYACTION

Routine: entity\_action\_get\_by\_relationship

LIMSML Entity Actionget By Relationship returns the list of parameters for the specified LIMSML entity action.

The following section describes the parameter for Limsml Entity Actionget By Relationship.

## **IDENTITY**

The padded identity of the LIMSML entity action.

This parameter is mandatory.

## **System Get File**

Entity: SYSTEM

Action: GET\_FILE

Master Menu Procedure Number: 15583

VGL Library: \$LIMSML\_SYSTEM

Routine: system\_get\_file

System Get File returns a base-64 encoded copy of the requested file.

The following section describes the parameter for System Get File.

## **FILENAME**

The absolute path to the required file.

This parameter is mandatory.

## **System Ping**

Entity: SYSTEM

Action: PING

Master Menu Procedure Number: 15584

VGL Library: \$LIMSML\_SYSTEM

Routine: system\_ping

System Ping returns a response containing the message sent in the request.

The following section describes the parameter for System Ping.

## **MESSAGE**

The message to include in the response from the LIMSML Processor.

This parameter is optional.

## Example

The following LIMSML request uses the PING command to return a message from the SampleManager server.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SYSTEM">
      <actions>
        <action>
          <command>PING</command>
          <parameter name="MESSAGE">TEST_STRING</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SYSTEM">
      <actions>
        <action>
          <command>PING</command>
          <parameter name="MESSAGE">TEST_STRING</parameter>
        </action>
      </actions>
      <fields>
        <field id="RETURN" direction="inout" datatype="Text">Received message: 'TEST_STRING'
at 26-OCT-2012 14:44:36.81</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## System Sign Failure

Entity: SYSTEM

Action: SIGN\_FAILURE

Master Menu Procedure Number: 15634

VGL Library: \$LIMSML\_SYSTEM

Routine: system\_sign\_failure

System Sign Failure creates an electronic signature failure for a specified menu option.

The following section describes the parameter for System Sign Failure.

### PROCEDURE\_NUMBER

The procedure number of the master menu option.

This parameter is mandatory.

## System Sign Menu Item

Entity: SYSTEM

Action: SIGN\_MENU\_ITEM

Master Menu Procedure Number: 15623

VGL Library: \$LIMSML\_SYSTEM

Routine: system\_sign\_menu\_item

System Sign Menu Item creates an electronic signature for a specific menu item.

The following section describes the parameter for System Sign Menu Item.

### PROCEDURE\_NUMBER

The procedure number of the master menu option.

This parameter is mandatory.

## User Interface Get Fields

Entity: USER\_INTERFACE

Action: GET\_FIELDS

Master Menu Procedure Number: 15588

VGL Library: \$LIMSML\_UI

Routine: user\_interface\_get\_fields

User Interface Get Fields returns the prompt info for the fields in the specified user interface.

The following sections describe the parameters for User Interface Get Fields.

### INTERFACE

The identity of the user interface.

This parameter is mandatory.

### TYPE\_ID

Type the interface is based on.

This parameter is mandatory.

### Example

The following LIMSML request uses the GET\_FIELDS command to return a list of the fields associated with the WEB\_MENU\_ITEM DEFAULT user interface.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="USER_INTERFACE">
      <actions>
        <action>
          <command>GET_FIELDS</command>
          <parameter name="TYPE_ID">WEB_MENU_ITEM</parameter>
          <parameter name="INTERFACE">DEFAULT</parameter>
        </action>
      </actions>
      <fields />
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <data response_type="data" xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <ADODataset>
      <NewDataSet xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
        <xs:schema id="NewDataSet"
targetNamespace="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:mstns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified"
elementFormDefault="qualified">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="USER_INTERFACE_FIELD">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="IDENTITY" msdata:Caption="Get identity"
type="xs:string" minOccurs="0" />
                      <xs:element name="DESCRIPTION" msdata:Caption="Get name"
type="xs:string" minOccurs="0" />
                      <xs:element name="TYPE" msdata:Caption="Type" type="xs:string"
minOccurs="0" />
                      <xs:element name="ALLOWED_CHARACTERS" msdata:Caption="Get allowed
characters" type="xs:string" minOccurs="0" />
                      <xs:element name="FORMAT_STRING" msdata:Caption="Get format string"
type="xs:string" minOccurs="0" />
                      <xs:element name="TRUE_WORD" msdata:Caption="Get true word"
type="xs:string" minOccurs="0" />
                      <xs:element name="FALSE_WORD" msdata:Caption="Get false word"
type="xs:string" minOccurs="0" />
                      <xs:element name="READ_ONLY" msdata:Caption="Is read only"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="MANDATORY" msdata:Caption="Is mandatory"
type="xs:boolean" minOccurs="0" />
                      <xs:element name="LENGTH" msdata:Caption="Get length" type="xs:int"
minOccurs="0" />
                      <xs:element name="DEFAULT_VALUE" msdata:Caption="Default value"
type="xs:string" minOccurs="0" />
                      <xs:element name="LOWER_LIMIT" msdata:Caption="Lower limit"
type="xs:string" minOccurs="0" />
                      <xs:element name="UPPER_LIMIT" msdata:Caption="Upper limit"
type="xs:string" minOccurs="0" />
                      <xs:element name="BROWSE_TYPE" msdata:Caption="Browse type"
type="xs:string" minOccurs="0" />
                      <xs:element name="BROWSE_PROPERTY" msdata:Caption="Browse property"
type="xs:string" minOccurs="0" />
                      <xs:element name="SELECT_LIST" msdata:Caption="Select list"
type="xs:string" minOccurs="0" />
                      <xs:element name="HIDDEN" msdata:Caption="Hidden" type="xs:boolean"
minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:element>
          </xs:schema>
        </NewDataSet>
      </ADODataset>
    </data>
  </transaction>
```

```

</xs:element>
<xs:element name="USER_INTERFACE">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="TABLE_NAME" msdata:Caption="Table name"
type="xs:string" minOccurs="0" />
      <xs:element name="INTERFACE" msdata:Caption="Interface"
type="xs:string" minOccurs="0" />
      <xs:element name="GROUP_ID" msdata:Caption="Group id"
type="xs:string" minOccurs="0" />
      <xs:element name="MODIFIABLE" msdata:Caption="Modifiable"
type="xs:boolean" minOccurs="0" />
      <xs:element name="MODIFIED_ON" msdata:Caption="Modified on"
type="xs:dateTime" minOccurs="0" />
      <xs:element name="MODIFIED_BY" msdata:Caption="Modified by"
type="xs:string" minOccurs="0" />
      <xs:element name="REMOVEFLAG" msdata:Caption="Removeflag"
type="xs:boolean" minOccurs="0" />
      <xs:element name="TYPE" msdata:Caption="Type" type="xs:string"
minOccurs="0" />
      <xs:element name="DISPLAY_TABLE_NAME" msdata:Caption="Display table
name" type="xs:string" minOccurs="0" />
      <xs:element name="HEADER_MESSAGE" msdata:Caption="Header message"
type="xs:string" minOccurs="0" />
      <xs:element name="FILE_DIRECTORY" msdata:Caption="File directory"
type="xs:string" minOccurs="0" />
      <xs:element name="FILE_EXTENSION" msdata:Caption="File extension"
type="xs:string" minOccurs="0" />
      <xs:element name="INPUT_MESSAGE" msdata:Caption="Input message"
type="xs:string" minOccurs="0" />
      <xs:element name="IDENTITY_MESSAGE" msdata:Caption="Identity message"
type="xs:string" minOccurs="0" />
      <xs:element name="IDENTITY2_MESSAGE" msdata:Caption="Identity2
message" type="xs:string" minOccurs="0" />
      <xs:element name="IDENTITY3_MESSAGE" msdata:Caption="Identity3
message" type="xs:string" minOccurs="0" />
      <xs:element name="IDENTITY4_MESSAGE" msdata:Caption="Identity4
message" type="xs:string" minOccurs="0" />
      <xs:element name="MODIFY_ON_MESSAGE" msdata:Caption="Modify on
message" type="xs:string" minOccurs="0" />
      <xs:element name="MODIFY_BY_MESSAGE" msdata:Caption="Modify by
message" type="xs:string" minOccurs="0" />
      <xs:element name="READ_MESSAGE" msdata:Caption="Read message"
type="xs:string" minOccurs="0" />
      <xs:element name="WRITE_MESSAGE" msdata:Caption="Write message"
type="xs:string" minOccurs="0" />
      <xs:element name="APPROVAL_MESSAGE" msdata:Caption="Approval message"
type="xs:string" minOccurs="0" />
      <xs:element name="DEFAULT_VALUE" msdata:Caption="Default value"
type="xs:string" minOccurs="0" />
      <xs:element name="HELP_CONTEXT" msdata:Caption="Help context"
type="xs:string" minOccurs="0" />
      <xs:element name="REPORT_NAME" msdata:Caption="Report name"
type="xs:string" minOccurs="0" />
      <xs:element name="LIST_PRE_INSERT" msdata:Caption="List pre insert"
type="xs:string" minOccurs="0" />
    
```

```

        <xs:element name="LIST_POST_INSERT" msdata:Caption="List post insert"
type="xs:string" minOccurs="0" />
        <xs:element name="LIST_VALIDATION" msdata:Caption="List validation"
type="xs:string" minOccurs="0" />
        <xs:element name="LIST_SELECT" msdata:Caption="List select"
type="xs:string" minOccurs="0" />
        <xs:element name="LIST_DELETE" msdata:Caption="List delete"
type="xs:string" minOccurs="0" />
        <xs:element name="LIST_CALLBACK" msdata:Caption="List callback"
type="xs:string" minOccurs="0" />
        <xs:element name="LIST_TOOLBAR" msdata:Caption="List toolbar"
type="xs:string" minOccurs="0" />
        <xs:element name="NAME" msdata:Caption="Name" type="xs:string"
minOccurs="0" />
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<USER_INTERFACE_FIELD>
    <IDENTITY>OPTION_NUMBER</IDENTITY>
    <DESCRIPTION>Option number</DESCRIPTION>
    <TYPE>TEXT</TYPE>
    <ALLOWED_CHARACTERS />
    <READ_ONLY>false</READ_ONLY>
    <MANDATORY>false</MANDATORY>
    <LENGTH>234</LENGTH>
    <DEFAULT_VALUE />
    <HIDDEN>false</HIDDEN>
</USER_INTERFACE_FIELD>
<USER_INTERFACE_FIELD>
    <IDENTITY>NAME</IDENTITY>
    <DESCRIPTION>Name</DESCRIPTION>
    <TYPE>TEXT</TYPE>
    <ALLOWED_CHARACTERS />
    <READ_ONLY>false</READ_ONLY>
    <MANDATORY>false</MANDATORY>
    <LENGTH>50</LENGTH>
    <DEFAULT_VALUE xml:space="preserve"> </DEFAULT_VALUE>
    <HIDDEN>false</HIDDEN>
</USER_INTERFACE_FIELD>
<USER_INTERFACE_FIELD>
    <IDENTITY>ICON</IDENTITY>
    <DESCRIPTION>Icon</DESCRIPTION>
    <TYPE>TEXT</TYPE>
    <ALLOWED_CHARACTERS />
    <READ_ONLY>false</READ_ONLY>
    <MANDATORY>false</MANDATORY>
    <LENGTH>234</LENGTH>
    <DEFAULT_VALUE xml:space="preserve"> </DEFAULT_VALUE>
    <BROWSE_TYPE>ICON</BROWSE_TYPE>
    <BROWSE_PROPERTY>IDENTITY</BROWSE_PROPERTY>
    <HIDDEN>false</HIDDEN>
</USER_INTERFACE_FIELD>

```

```

<USER_INTERFACE_FIELD>
  <IDENTITY>SEPARATE_WINDOW</IDENTITY>
  <DESCRIPTION>Separate window</DESCRIPTION>
  <TYPE>BOOLEAN</TYPE>
  <TRUE_WORD>True</TRUE_WORD>
  <FALSE_WORD>False</FALSE_WORD>
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <DEFAULT_VALUE>false</DEFAULT_VALUE>
  <HIDDEN>false</HIDDEN>
</USER_INTERFACE_FIELD>
<USER_INTERFACE_FIELD>
  <IDENTITY>TARGET</IDENTITY>
  <DESCRIPTION>Target</DESCRIPTION>
  <TYPE>TEXT</TYPE>
  <ALLOWED_CHARACTERS />
  <READ_ONLY>false</READ_ONLY>
  <MANDATORY>false</MANDATORY>
  <LENGTH>50</LENGTH>
  <DEFAULT_VALUE xml:space="preserve"> </DEFAULT_VALUE>
  <HIDDEN>false</HIDDEN>
</USER_INTERFACE_FIELD>
<USER_INTERFACE>
  <TABLE_NAME>WEB_MENU_ITEM</TABLE_NAME>
  <INTERFACE>DEFAULT</INTERFACE>
  <GROUP_ID />
  <MODIFIABLE>true</MODIFIABLE>
  <MODIFIED_ON>2012-10-26T03:28:26+01:00</MODIFIED_ON>
  <MODIFIED_BY>BATCH</MODIFIED_BY>
  <REMOVEFLAG>false</REMOVEFLAG>
  <TYPE>LISTEDITOR</TYPE>
  <DISPLAY_TABLE_NAME>Web Menu Item</DISPLAY_TABLE_NAME>
  <HEADER_MESSAGE>Web Menu Item Data Entry Screen</HEADER_MESSAGE>
  <FILE_DIRECTORY />
  <FILE_EXTENSION />
  <INPUT_MESSAGE />
  <IDENTITY_MESSAGE />
  <IDENTITY2_MESSAGE />
  <IDENTITY3_MESSAGE />
  <IDENTITY4_MESSAGE />
  <MODIFY_ON_MESSAGE />
  <MODIFY_BY_MESSAGE />
  <READ_MESSAGE />
  <WRITE_MESSAGE />
  <APPROVAL_MESSAGE />
  <DEFAULT_VALUE />
  <HELP_CONTEXT>$WEB_MENU</HELP_CONTEXT>
  <REPORT_NAME>$WEB_MENU</REPORT_NAME>
  <LIST_PRE_INSERT>ITEMPREINSERTROUTINE</LIST_PRE_INSERT>
  <LIST_POST_INSERT />
  <LIST_VALIDATION>ITEMVALIDATIONROUTINE</LIST_VALIDATION>
  <LIST_SELECT />
  <LIST_DELETE />
  <LIST_CALLBACK />
  <LIST_TOOLBAR />
  <NAME>Web Menu Item Default</NAME>
</USER_INTERFACE>

```

```
</NewDataSet>
</ADODataset>
</data>
</transaction>
```

## User Interface Validate Values

Entity: USER\_INTERFACE

Action: VALIDATE\_VALUES

Master Menu Procedure Number: 15592

VGL Library: \$LIMSML\_UI

Routine: user\_interface\_validate\_values

User Interface Validate Values validates the fields in the specified user interface.

The following sections describe the parameters for User Interface Validate Values.

### INTERFACE

The identity of the user interface.

This parameter is mandatory.

### TYPE\_ID

Type the interface is based on.

This parameter is mandatory.

## Example

The following LIMSML request uses the VALIDATE\_VALUES command to check values supplied for fields in the WEB\_MENU\_SECTION DEFAULT user interface.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="USER_INTERFACE">
      <actions>
        <action>
          <command>VALIDATE_VALUES</command>
          <parameter name="TYPE_ID">WEB_MENU_SECTION</parameter>
          <parameter name="INTERFACE">DEFAULT</parameter>
        </action>
      </actions>
      <fields>
        <field id="DESCRIPTION" direction="inout" datatype="Text">description</field>
        <field id="GROUP_ID" direction="inout" datatype="Text">SYSTEM</field>
        <field id="NAME" direction="inout" datatype="Text">name</field>
        <field id="ORDER_NUMBER" direction="inout" datatype="Text">not a number</field>
        <field id="ICON" direction="inout" datatype="Text">NOT_EXIST</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

The request above produces the following type of response. If the value supplied for a field is valid, the field element in the response contains True. If the value supplied for a field is not valid, the field element in the response contains the value False.

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="USER_INTERFACE">
      <actions />
      <fields>
        <field id="DESCRIPTION" direction="out" datatype="Text">True</field>
        <field id="GROUP_ID" direction="out" datatype="Text">True</field>
        <field id="NAME" direction="out" datatype="Text">True</field>
        <field id="ORDER_NUMBER" direction="out" datatype="Text">False</field>
        <field id="ICON" direction="out" datatype="Text">False</field>
      </fields>
      <children />
    </entity>
  </system>
</transaction>
```

## User Login

Entity: USER

Action: LOGIN

Master Menu Procedure Number: 15585

VGL Library: \$LIMSML\_USER

Routine: user\_login

User Login is called when a user first logs in.

User Login has no parameters.

## User Logout

Entity: USER

Action: LOGOUT

Master Menu Procedure Number: 15586

VGL Library: \$LIMSML\_USER

Routine: user\_logout

User Logout is called when a user logs out or the session expires.

User Logout has no parameters.



## Result Entry with LIMSML

This chapter describes the Sample Result Entry LIMSML entity action in detail.

### Contents

- [Sample Identification and Management](#)
- [Test Identification and Management](#)
- [Result Entry](#)
- [Updating Sample and Test Information](#)

## Sample Identification and Management

Before results can be entered, the sample that they are associated with in the SampleManager database must be identified. If a record for the sample already exists, it can be identified using one of the following methods:

- [Obtaining the Id Numeric from the LIMSML](#) (page 134).
- [Obtaining the Id Text from the LIMSML](#) (page 134).
- [Obtaining the sample identity using the Test Number of the test](#) (page 135).
- [Using a custom VGL routine to identify the sample](#) (page 136).

If a record for the sample cannot be identified, one of the following events occurs:

- [The missing sample is logged in to SampleManager using information in the LIMSML](#) (described on page 137).
- [An error is returned](#) (page 139).
- [No error is returned, and the results associated with the missing sample are ignored](#) (page 139).

## Identifying Samples using ID Numeric

The sample can be identified using the Id Numeric if a field element containing the Id Numeric value is included in the section of the LIMSML request that describes the sample. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
```

The example above specifies result entry for a sample with the Id Numeric 1522.

## Identifying Samples using ID Text

The sample can be identified using the Id Text if a field element containing the Id Text value is included in the section of the LIMSML request that describes the sample. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
```

The example above specifies result entry for a sample with the Id Text "SM-12-NOV-2010-001522".

## Identifying Samples using Test Number

The sample can be identified by locating the test record for its first test in the SampleManager database. The identity of the parent sample is included in the test record. The test can be identified if a field element containing the Test Number value is included in the section of the LIMSML request that describes the test. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields />
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="TEST_NUMBER" direction="inout" datatype="Text">18028</field>
      </fields>
      <children>
        ...result information
      </children>
    </entity>
  </children>
</entity>
```

In this case the sample identity is obtained by locating the test record with Test Number "18028", and getting the Id Numeric of the parent sample from the test record.

**Note** Only the first test entity associated with the sample entity in the LIMSML is used to obtain a test number. Any other test entities associated with the sample are not examined.

## Identifying Samples using VGL Code

The `limsml_result_entry_get_sample` callback routine in `$LIMSMML_SAMPLE_USER` is always called at the end of the sample identification process. This routine allows custom VGL code to determine the Id Numeric of the sample record for result entry. Two parameters are passed to the routine:

- `sample_entity` is a VGL object that represents the sample defined in the LIMSMML. The `sample_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSMML_HELPER` report.
- `sample_id` is the current value of the Id Numeric for the sample.

The `limsml_result_entry_get_sample` routine returns the value of the `sample_id` parameter.

The default `limsml_result_entry_get_sample` routine is as follows:

```
GLOBAL ROUTINE limsml_result_entry_get_sample ( sample_entity, VALUE sample_id )  
  
{  
* Get hold of the ID Numeric for a LIMSMML Sample Result Entry Command  
*  
* Parameters      : Sample Entity, [HELPER_ENTITY] Sample Entity Helper Entity  
*                  Sample Id, [PACKED_DECIMAL] Sample ID  
*  
* Return Value : [PACKED_DECIMAL], The sample id to use. Return EMPTY to abandon  
*  
*****}  
  
    RETURN ( sample_id )  
  
ENDROUTINE
```

## Creating Samples using a Sample Template

The RESULT\_ENTRY action can create a sample using a sample template, if the required sample record does not exist in the SampleManager database. This behavior is controlled by the following constant in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_SAMPLE = "FALSE"
```

If LIMSML\_SAMPLE\_CREATE\_SAMPLE is TRUE, SampleManager creates a new sample if an existing sample is not identified for the result entry operation. If LIMSML\_SAMPLE\_CREATE\_SAMPLE is FALSE, SampleManager treats the sample as [missing](#) (page 139).

The default setting of LIMSML\_SAMPLE\_CREATE\_SAMPLE can be overridden using the [CREATE\\_SAMPLE](#) parameter (page 40) in the LIMSML request.

The LIMSML\_SAMPLE\_CREATE\_SAMPLE\_TEMPLATE constant in \$LIMSML\_SAMPLE\_USER defines the sample template used to create samples for result entry from LIMSML.

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_SAMPLE_TEMPLATE = "BLANK"
```

The default setting of LIMSML\_SAMPLE\_CREATE\_SAMPLE can be overridden using the [CREATE\\_SAMPLE\\_TEMP](#) parameter (page 40) in the LIMSML request.

The following example uses the MY\_TEMP sample template to log in a new sample:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="CREATE_SAMPLE">TRUE</parameter>
      <parameter name="CREATE_SAMPLE_TEMP">MY_TEMP</parameter>
    </action>
  </actions>
  <children>
    ...tests and results
  </children>
</entity>
```

## Creating Samples using a Sample Login Workflow

The RESULT\_ENTRY action can create a sample using a sample login workflow if the required sample record does not exist in the SampleManager database. This behavior is controlled by the following constant in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_SAMPLE = "FALSE"
```

If LIMSML\_SAMPLE\_CREATE\_SAMPLE is TRUE, SampleManager creates a new sample if an existing sample is not identified for the result entry operation. If LIMSML\_SAMPLE\_CREATE\_SAMPLE is FALSE, SampleManager treats the sample as [missing](#) (page 139).

The default setting of LIMSML\_SAMPLE\_CREATE\_SAMPLE can be overridden using the [CREATE\\_SAMPLE parameter](#) (page 40) in the LIMSML request.

The sample login workflow to use is specified by the [CREATE\\_SAMPLE\\_WORKFL parameter](#) (page 40) in the LIMSML request.

The following example uses the Single Sample sample login workflow to log in a new sample:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="CREATE_SAMPLE_WORKFL">Single Sample</parameter>
      <parameter name="CREATE_SAMPLE">True</parameter>
      <parameter name="CREATE_TESTS">True</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="out" datatype="Text" />
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
</children>
</entity>
```

## Handling Missing Samples

The following constant in \$LIMSML\_SAMPLE\_USER determines the default behavior when no suitable sample can be identified for the result entry process:

```
GLOBAL CONSTANT LIMSML_SAMPLE_REQUIRE_SAMPLE = "TRUE"
```

If LIMSML\_SAMPLE\_REQUIRE\_SAMPLE is TRUE, and no sample is found, the following type of error is returned in the LIMSML response:

```
<error>
  <summary>No Sample</summary>
  <description>Unable to Identify a Sample, please specify a valid ID_TEXT or ID_NUMERIC.
Sample Entity [SAMPLE{ID_TEXT='INVALID' ID_NUMERIC='123456789'}]</description>
  <code>NOSAMPLE</code>
  <severity>ERROR</severity>
  <source>SMP Business</source>
  <errors />
</error>
```

If LIMSML\_SAMPLE\_REQUIRE\_SAMPLE is FALSE, and no sample is found, the sample and its child tests and results are ignored.

The default setting of LIMSML\_SAMPLE\_REQUIRE\_SAMPLE can be overridden using the [REQUIRE\\_VALID\\_SAMPLE parameter](#) (page 42) in the LIMSML request.

The following example uses the REQUIRE\_VALID\_SAMPLE parameter to skip the WASH1 sample in the LIMSML request:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="REQUIRE_VALID_SAMPLE">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">WASH1</field>
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="REQUIRE_VALID_SAMPLE">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
```

```
</children>  
</entity>
```

## Test Identification and Management

Before results can be entered, the test that they are associated with in the SampleManager database must be identified. If a record for the test already exists, it can be identified using one of the following methods:

- [Specifying the test number in the LIMSML](#) (page 140).
- [Identifying the analysis](#) (page 141) associated with the test, then [selecting an appropriate replicate](#) (page 142).
- [Using a custom VGL routine](#) (page 147).

If a record for the test cannot be identified, one of the following events occurs:

- [The missing test is logged in to SampleManager using information in the LIMSML](#) (page 148).
- [An error is returned](#) (page 149).
- [No error is returned, and the results associated with the missing test are ignored](#) (page 149).

## Specifying the Test Number

The test can be identified using the Test Number if a field element containing the Test Number value is included in the section of the LIMSML request that describes the test. For example:

```
<entity type="SAMPLE">  
  <actions>  
    <action>  
      <command>RESULT_ENTRY</command>  
    </action>  
  </actions>  
  <fields />  
  <children>  
    <entity type="TEST">  
      <actions />  
      <fields>  
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>  
        <field id="TEST_NUMBER" direction="inout" datatype="Text">18028</field>  
      </fields>  
      <children>  
        ...results  
      </children>  
    </entity>  
  </children>
```

```
</entity>
```

The example above specifies that the results belong to the test with test number 18028.

## Identifying Tests using the Analysis

Each test record in the LIMSML should have a field element indicating which analysis was used to obtain the results.

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
  </children>
</entity>
```

The example above specifies that the test is associated with the MAE-IM analysis.

When the correct analysis has been identified, the [replicate number of the test must be determined](#) (page 142).

## Using VGL Code to determine the Analysis

The `limsml_result_entry_get_analysis` callback in `$LIMSML_SAMPLE_USER` is always called after the analysis associated with a test has been identified. This callback allows custom VGL code to determine the analysis associated with the test that the results are entered against. Two parameters are passed to the routine:

- `test_entity` is a VGL object that represents the test defined in the LIMSML. The `test_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSML_HELPER` report.
- `analysis` is the value of the Test Number for the currently identified test.

The `limsml_result_entry_get_test` routine returns the value of the `test_number` parameter.

The default limsml\_result\_entry\_get\_test routine is:

```
{*****}

GLOBAL ROUTINE limsml_result_entry_get_analysis (      test_entity ,
                                                    VALUE analysis  )

{
* Get hold of the Analysis for a LIMSMML Sample Result Entry Command
*
* Parameters    : Test Entity, [HELPER_ENTITY] Test Entity Helper Entity
*                Analysis, [IDENTITY] Analysis Identity
*
* Return value : [IDENTITY], The Analysis to use. Return EMPTY to abandon
*
*****}

    RETURN ( analysis )

ENDROUTINE
```

## Determining the Test Replicate

The default method for determining the test replicate that results are entered against is to select the first test replicate that matches the following criteria:

- The Id Numeric identified for the test's parent sample matches the value in the sample field of the test record.
- AND
- The Analysis identified for the test matches the value in the analysis field of the test record.
- AND
- The current status of the test record is valid for the result entry operation.

The list of test statuses that allow result entry is defined by the following constant in \$LIMSMML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSMML_SAMPLE_VALID_TEST_STATUS      = "CVP"
```

The default value of LIMSML\_SAMPLE\_VALID\_TEST\_STATUS can be overridden by the [VALID\\_TEST\\_STATUS parameter](#) (page 43) of the RESULT\_ENTRY action. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="VALID_TEST_STATUS">V</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
  </children>
</entity>
```

In the example above, the results will be entered against the first test replicate that meets the following criteria:

- The test record is associated with the sample that has the Id Text SM-12-NOV-2010-001522.

AND

- The test record is associated with the MAE-IM analysis.

AND

- The test status is V.

## Specifying the Test Replicate in LIMSML

The test replicate can be identified from the LIMSML request if a field element containing the Test Count value is included in the section of the LIMSML request that describes the test. The Test Count field contains the replicate number of the test. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
        <field id="TEST_COUNT" direction="in" datatype="Text">3</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
  </children>
</entity>
```

This example will enter results against the test record that meets the following criteria:

- The test record is associated with the sample that has the Id Numeric 1522.
- AND
- The test record is associated with the MAE-IM analysis.
- AND
- The test record Test Count field value is 3.

## Specifying the Test Replicate Using VGL

The `limsml_result_entry_get_test_rep` callback routine can be used to determine the required test replicate. This routine is always called after the test replicate has been identified using the information in the LIMSML request.

The `limsml_result_entry_get_test_rep` routine has two parameters:

- `test_entity` is a VGL object that represents the test defined in the LIMSML. The `test_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSML_HELPER` report.
- `test_rep` is the value of the Test Count for the currently identified test.

The `limsml_result_entry_get_test_rep` routine returns the value of the `test_rep` parameter.

The default `limsml_result_entry_get_test_rep` callback routine is:

```
GLOBAL ROUTINE limsml_result_entry_get_test_rep ( test_entity, VALUE test_rep )
{
* Get hold of the Analysis for a LIMSML Sample Result Entry Command
*
* Parameters    : Test Entity, [HELPER_ENTITY] Test Entity Helper Entity
*                Replicate Number, [INTEGER] Test Replicate Number
*
* Return Value : [INTEGER], The replicate to use. Return EMPTY to use the first
*
*****}

    RETURN ( test_rep )

ENDROUTINE
```

## Incrementing the Test Replicate

SampleManager keeps track of the test replicate used for result entry, so if a LIMSML request contains two tests associated with the same sample and analysis, the first test's results will go into the first appropriate test replicate and the second test's results will go into the second appropriate test replicate.

The automatic replicate tracking is controlled by the following constant in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_REPLICATES = "TRUE"
```

The default value, TRUE, enables replicate tracking.

Automatic replicate tracking can be overridden using the [TEST\\_REPS\\_ENABLED parameter](#) (page 42) in the LIMSML request. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="TEST_REPS_ENABLED">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
  </children>
</entity>
```

In the example above, the TEST\_REPS\_ENABLED parameter is used to switch off the replicate tracking, so the results from the second test will overwrite the results from the first test.

## Identifying the Test Number Using VGL

The `limsml_result_entry_get_test` callback in `$LIMSML_SAMPLE_USER` is always called after the test has been identified. This callback allows custom VGL code to determine the test number of the test that the results are entered against. Four parameters are passed to the routine:

- `sample_entity` is a VGL object that represents the sample defined in the LIMSML. The `sample_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSML_HELPER` report.
- `test_entity` is a VGL object that represents the test defined in the LIMSML. The `test_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSML_HELPER` report.
- `sample_id` is the value of the Id Numeric for the sample associated with the currently identified test.
- `test_number` is the value of the Test Number for the currently identified test.

The `limsml_result_entry_get_test` routine returns the value of the `test_number` parameter.

```
{*****}

GLOBAL ROUTINE limsml_result_entry_get_test (      sample_entity ,
                                                  test_entity   ,
                                                  VALUE sample_id   ,
                                                  VALUE test_number )

{
* Get hold of the Test Number for a LIMSML Sample Result Entry Command
*
* Parameters    : Sample Entity, [HELPER_ENTITY] Sample Entity Helper Entity
*                Test Entity, [HELPER_ENTITY] Test Entity Helper Entity
*                Sample Id, [PACKED_DECIMAL] Sample ID
*                Test Number, [PACKED_DECIMAL] Test Number
*
* Return value : [PACKED_DECIMAL], The test num to use. Return EMPTY to abandon
*
*****}

    RETURN ( test_number )

ENDROUTINE
```

## Creating Tests

The RESULT\_ENTRY action can create a test, if the required test record does not exist in the SampleManager database. This behavior is controlled by the following constant in \$LIMSML\_SAMPLE\_USER:

```
GLOBAL CONSTANT LIMSML_SAMPLE_CREATE_TESTS = "FALSE"
```

If LIMSML\_SAMPLE\_CREATE\_TESTS is TRUE, SampleManager creates a new test if an existing test is not identified for the result entry operation. If LIMSML\_SAMPLE\_CREATE\_TESTS is FALSE, SampleManager treats the test as [missing](#) (page 149).

The default setting of LIMSML\_SAMPLE\_CREATE\_TESTS can be overridden using the [CREATE\\_TESTS parameter](#) (page 41) in the LIMSML request. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="CREATE_TESTS">TRUE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
  </children>
</entity>
```

The example above adds a new test based on the MAE-IM analysis to the sample with Id Text SM-12-NOV-2010-001522.

## Handling Missing Tests

The following constant in \$LIMSML\_SAMPLE\_USER determines the default behavior when no suitable test can be identified for the result entry process:

```
GLOBAL CONSTANT LIMSML_SAMPLE_REQUIRE_TEST = "TRUE"
```

If LIMSML\_SAMPLE\_REQUIRE\_TEST is TRUE and no test is found, the following type of error is returned in the LIMSML response:

```
<error>
  <summary>No Test</summary>
  <description>An appropriate test could not be found on Sample '      1522'. Test Entity
[TEST{ANALYSIS='INVALID'}]</description>
  <code>NOTEST</code>
  <severity>ERROR</severity>
  <source>SMP Business</source>
  <errors />
</error>
```

If LIMSML\_SAMPLE\_REQUIRE\_TEST is FALSE and no test is found, the test and its child results are ignored.

The default setting of LIMSML\_SAMPLE\_REQUIRE\_TEST can be overridden using the [REQUIRE\\_VALID\\_TEST](#) parameter (page 42) in the LIMSML request.

The following example uses the REQUIRE\_VALID\_TEST parameter to skip the PURGE test in the LIMSML request:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="REQUIRE_VALID_TEST">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_TEXT" direction="in" datatype="Text">SM-12-NOV-2010-001522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">PURGE</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
```

```

        ...results
      </children>
    </entity>
  </children>
</entity>

```

## Result Entry

The result entry mechanism collects all the results for a given test, builds an array of field names and result values, and then uses the PUT TEST RESULTS command in VGL to store those results in the database.

## Result Fields

The first stage of processing the results is to determine the fields that will be updated. For each result, the elements are processed and an array of unique field names is constructed. As each field is identified, the `limsml_result_entry_valid_result_field` callback routine in `$LIMSMML_SAMPLE_USER` is called. This routine allows custom VGL code to control whether the field is included in the result set.

The `limsml_result_entry_valid_result_field` routine has the following parameters:

- `test_entity` is a VGL object that represents the test defined in the LIMSMML. The `test_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSMML_HELPER` report.
- `field_name` is the name of the field.
- `valid` is a Boolean value that defines whether the field is included in the results array. The initial value of `valid` is `TRUE`.

The `limsml_result_entry_valid_result_field` routine returns the value of the `valid` parameter.

The default `limsml_result_entry_valid_result_field` routine is:

```

GLOBAL ROUTINE limsml_result_entry_valid_result_field (      test_entity      ,
                                                             VALUE field_name  ,
                                                             VALUE valid       )

{
* Determine if the result field is valid
*
* Parameters   : Test Entity, [HELPER_ENTITY] Test Entity Helper Entity
*               Field Name, [STRING] Field Name
*               Valid, [BOOLEAN] Current Validity of the Field
*
* Return value : [BOOLEAN], True if the result field should be included
*
*****}

```

```
RETURN ( valid )
```

```
ENDROUTINE
```

## Validating Results

During the second stage in preparing the call to PUT\_TEST\_RESULTS the array of result values is constructed. Each result is processed and the values in its field elements are added to the results array in the appropriate columns.

Before a result is added to the results array, the limsml\_result\_entry\_valid\_result callback routine in \$LIMSML\_SAMPLE\_USER is called. If this callback routine returns TRUE the result is included in the results array. The limsml\_result\_entry\_valid\_result routine can be used to filter the results that are entered from LIMSML requests.

The limsml\_result\_entry\_valid\_result routine has the following parameters:

- test\_number is the value of the test\_number field for the parent test associated with the result.
- result\_entity is a VGL object that represents the result defined in the LIMSML. The result\_entity object is based on the HELPER\_ENTITY class, defined in the \$LIMSML\_HELPER report.
- valid is a Boolean value that defines whether the result is included in the results array. The initial value of valid is TRUE.

The default limsml\_result\_entry\_valid\_result routine is:

```
GLOBAL ROUTINE limsml_result_entry_valid_result ( VALUE test_number  ,
                                                  result_entity ,
                                                  VALUE valid      )

{
* Determine if the result is valid
*
* Parameters   : Test Number, [PACKED_DECIMAL] Test Number
*               Result Entity, [HELPER_ENTITY] Result Entity Helper Entity
*               Valid, [BOOLEAN] Current validity of the result
*
* Return Value : [BOOLEAN], True if the result should be included
*
*****}

RETURN ( valid )

ENDROUTINE
```

## Processing Errors

If there is a problem, "Put Test Results" returns an array of errors. The array contains either one value indicating that the whole test was problematic, or set of errors, each one specific to a result that caused the problem. An example error return can be seen below:

```
<error>
  <summary>Result Entry Error</summary>
  <description>Result entry failed with error 'Mandatory field result_type is required for
ad hoc results'. Test Entity [TEST{ANALYSIS='MAE-IM'}], Result Entity
[RESULT{COMPONENT_NAME='17-NOV-2010 11:01:37.89Invalid' TEXT='2'}]</description>
  <code>PTRERROR</code>
  <severity>ERROR</severity>
  <source>SMP Business</source>
  <errors />
</error>
```

The following example shows an error reported for a test level in the result entry process:

```
<error>
  <summary>Result Entry Error</summary>
  <description>Result entry failed with error 'Mandatory field component_name is missing'.
Test Entity [TEST{ANALYSIS='MAE-IM'}].</description>
  <code>PTERROR</code>
  <severity>ERROR</severity>
  <source>SMP Business</source>
  <errors />
</error>
```

Each of the error codes has been categorized as either a mapping error or a general error.

## Mapping Errors

Mapping errors are normally caused by a problem with the mapping, parsing, or LIMSML, rather than the result itself. They usually need to be addressed in the calling code rather than in SampleManager. The following error codes are classified as mapping errors:

| Code               | Description                               |
|--------------------|-------------------------------------------|
| RPC_K_BADFIELD     | The specified result field does not exist |
| RPC_K_BADFLDSTATUS | The status field cannot be updated        |
| RPC_K_BADFLDENTON  | The entered_on field cannot be updated    |
| RPC_K_BADFLDENTBY  | The entered_by field cannot be updated    |
| RPC_K_BADFLDTEST   | The test_number field cannot be updated   |

| Code              | Description                                                |
|-------------------|------------------------------------------------------------|
| RPC_K_BADFLDVALUE | The value field cannot be updated                          |
| RPC_K_COMPISMAND  | Mandatory field component_name is missing                  |
| RPC_K_NORESTYPE   | Mandatory field result type is required for ad hoc results |

## General Errors

All other errors thrown by PUT TEST RESULTS relate to issues with the data or context of the result entry rather than the mapping. By default, general errors are ignored.

## Ignoring Result Errors

If an error occurs during the result entry process, it can be reported in the LIMSML response or ignored. If an error is ignored, the LIMSML response does not contain an error element and the result entry process continues as if no problem occurred.

The IGNORE\_RESULT\_ERRORS constant in \$LIMSML\_SAMPLE\_USER determines whether general errors are ignored. If IGNORE\_RESULT\_ERRORS is TRUE, general errors are ignored. The default value is TRUE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_IGNORE_RESULT_ERRORS = "TRUE"
```

The default setting of IGNORE\_RESULT\_ERRORS can be overridden by the [IGNORE\\_RESULT\\_ERRORS parameter](#) (page 41) in the LIMSML request.

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="IGNORE_RESULT_ERRORS">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
```

The IGNORE\_ALL\_RESULT\_ERRORS constant in \$LIMSML\_SAMPLE\_USER determines whether mapping and general errors are ignored. If IGNORE\_ALL\_RESULT\_ERRORS is TRUE, mapping and general errors are ignored. The default value is FALSE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_IGNORE_ALL_RESULT_ERRORS = "FALSE"
```

The default setting of IGNORE\_ALL\_RESULT\_ERRORS can be overridden by the [IGNORE\\_ALL\\_RESULT\\_ER parameter](#) (page 41) in the LIMSML request.

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="IGNORE_ALL_RESULT_ER">TRUE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    ...tests and results
  </children>
</entity>
```

## Handling Result Errors Using VGL

The limsml\_result\_entry\_valid\_result\_error callback routine in \$LIMSML\_SAMPLE\_USER is called when a result error occurs. This routine can use custom VGL code to determine whether a result error is ignored or reported in the LIMSML response.

The limsml\_result\_entry\_valid\_result\_error routine has the following parameters:

- test\_entity is a VGL object that represents the test defined in the LIMSML. The test\_entity object is based on the HELPER\_ENTITY class, defined in the \$LIMSML\_HELPER report.
- result\_entity is a VGL object that represents the result defined in the LIMSML. The result\_entity object is based on the HELPER\_ENTITY class, defined in the \$LIMSML\_HELPER report.
- error\_id contains the identity of the error.
- valid is a Boolean value that defines whether the error should be ignored. If valid is TRUE, the error is ignored. If valid is FALSE, the error is reported in the LIMSML response.

The default limsmml\_result\_entry\_valid\_result\_error routine is:

```
{*****}

GLOBAL ROUTINE limsmml_result_entry_valid_result_error (      test_entity  ,
                                                             result_entity ,
                                                             VALUE error_id  ,
                                                             VALUE valid     )

{
* Determine if the result error is ignorable
*
* Parameters   : Test Entity, [HELPER_ENTITY] Test Entity Helper Entity
*               Result Entity, [HELPER_ENTITY] Result Entity Helper Entity
*               Field Name, [STRING] Field Name
*               Valid, [BOOLEAN] Current Validity of the Error
*
* Return Value : [BOOLEAN], True if the result error should be ignored
*
*****}

    RETURN ( valid )

ENDROUTINE
```

## Entering Ad Hoc Results

Ad hoc results can be entered if the result type is specified in the LIMSMML. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
      <children>
        <entity type="RESULT">
          <actions />
          <fields>
            <field id="COMPONENT_NAME" direction="in" datatype="Text">17-NOV-2010
11:31:32.83Adhoc 1</field>
            <field id="TEXT" direction="in" datatype="Text">1</field>
            <field id="RESULT_TYPE" direction="in" datatype="Text">N</field>
          </fields>
        </entity>
      </children>
    </entity>
  </children>
</entity>
```

```
</children>
</entity>
```

The LIMSML\_SAMPLE\_IGNORE\_ADHOC constant in \$LIMSML\_SAMPLE\_USER determines whether ad hoc results are added to SampleManager or ignored. If LIMSML\_SAMPLE\_IGNORE\_ADHOC is TRUE, ad-hoc results are ignored when the LIMSML is processed. The default value is FALSE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_IGNORE_ADHOC = "FALSE"
```

The LIMSML\_SAMPLE\_IGNORE\_ADHOC value can be overridden using the [IGNORE\\_ADHOC parameter](#) (page 41) in a LIMSML request. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="IGNORE_ADHOC">TRUE</parameter>
    </action>
  </actions>
```

## Updating the Entered On and Entered By Fields

The LIMSML\_SAMPLE\_UPDATE\_INTERNAL\_FIELDS constant in \$LIMSML\_SAMPLE\_USER determines the mechanism used to update the Entered On and Entered By fields for each result uploaded to SampleManager. This is a system wide setting and not changeable on a per command basis.

```
GLOBAL CONSTANT LIMSML_SAMPLE_UPDATE_INTERNAL_FIELDS = TRUE
```

If LIMSML\_SAMPLE\_UPDATE\_INTERNAL\_FIELDS is TRUE and values for the Entered On and Entered By fields are included in the LIMSML request, these values will be used to populate the corresponding fields in the result record.

If LIMSML\_SAMPLE\_UPDATE\_INTERNAL\_FIELDS is FALSE, or values for the fields are not included in the LIMSML, the Entered On and Entered By fields in the result record are populated with the username of the account that logged in to SampleManager through the web service, and the time when the result was entered in SampleManager.

## Forcing Recalculation of Calculated Results

The Result Entry action can force the recalculation of all calculated results in a test after results are entered using LIMSML.

The LIMSML\_SAMPLE\_RECALCULATE constant in \$LIMSML\_SAMPLE\_USER determines whether the Result Entry action forces the recalculation of calculated results. If LIMSML\_SAMPLE\_RECALCULATE is TRUE, SampleManager recalculates calculated results for a test when results are entered using LIMSML. The default value is FALSE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_RECALCULATE = "FALSE"
```

The LIMSML\_SAMPLE\_RECALCULATE value can be overridden by using the [RECALCULATE](#) parameter (page 42) in the LIMSML request. For example:

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>RESULT_ENTRY</command>
          <parameter name="CREATE_SAMPLE">TRUE</parameter>
          <parameter name="CREATE_SAMPLE_TEMP">BLANK</parameter>
          <parameter name="CREATE_TESTS">TRUE</parameter>
          <parameter name="RECALCULATE">TRUE</parameter>
        </action>
      </actions>
    </fields />
    <children>
      <entity type="TEST">
        <actions />
        <fields>
          <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
        </fields>
        <children>
          <entity type="RESULT">
            <actions />
            <fields>
              <field id="COMPONENT_NAME" direction="in"
datatype="Text">One</field>
              <field id="TEXT" direction="in" datatype="Text">1</field>
            </fields>
          </entity>
          <entity type="RESULT">
            <actions />
            <fields>
              <field id="COMPONENT_NAME" direction="in"
datatype="Text">Two</field>
              <field id="TEXT" direction="in" datatype="Text">2</field>
            </fields>
          </entity>
        </children>
      </entity>
    </children>
  </entity>
</transaction>
```

```

        <children />
      </entity>
    </children>
  </entity>
  <entity type="TEST">
    <actions />
    <fields>
      <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
    </fields>
    <children>
      <entity type="RESULT">
        <actions />
        <fields>
          <field id="COMPONENT_NAME" direction="in"
datatype="Text">One</field>
          <field id="TEXT" direction="in" datatype="Text">11</field>
        </fields>
        <children />
      </entity>
      <entity type="RESULT">
        <actions />
        <fields>
          <field id="COMPONENT_NAME" direction="in"
datatype="Text">Two</field>
          <field id="TEXT" direction="in" datatype="Text">12</field>
        </fields>
        <children />
      </entity>
    </children>
  </entity>
</children>
</entity>
</system>
</transaction>

```

## Specifying the Instrument that Obtained a Result

To specify the instrument that was used to obtain a result, include the `instrument_used` field in the list of fields for the RESULT record. The value of the `instrument_used` field must be the identity of an instrument record in SampleManager. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in">18752</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in">METALS</field>
      </fields>
      <children>
        <entity type="RESULT">
          <actions />
          <fields>
            <field id="COMPONENT_NAME" direction="in">Copper</field>
            <field id="TEXT" direction="in">15.3</field>
            <field id="INSTRUMENT_USED" direction="in">ICP_45259</field>
          </fields>
        </entity>
      </children>
    </entity>
  </children>
</entity>
```

If the `instrument_used` field of a result contains a value and the `RESULT_ALLOW_INST_RESULT_MODIFY` configuration item is set to No, that result cannot be modified by a user in the Result Entry screen.

## Updating Sample and Test Information

Sample and test fields can be updated during the processing of the Sample Result Entry action.

### Updating Samples

Sample field elements where the direction attribute value is "in" or "inout" update the value of the corresponding fields in the SampleManager database. This behavior does not apply to key0 fields. The following example updates the value of the Hazard field to \$MAE1:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
    <field id="HAZARD" direction="inout" datatype="Text">$MAE1</field>
  </fields>
</entity>
```

The LIMSML\_SAMPLE\_UPDATE\_SAMPLE constant in \$LIMSML\_SAMPLE\_USER determines whether the Result Entry action updates fields in sample records. If LIMSML\_SAMPLE\_UPDATE\_SAMPLE is TRUE, SampleManager updates sample fields. The default value is TRUE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_UPDATE_SAMPLE = "TRUE"
```

The LIMSML\_SAMPLE\_UPDATE\_SAMPLE value can be overridden using the [UPDATE\\_SAMPLE parameter](#) (page 43) in the LIMSML request. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="UPDATE_SAMPLE">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
    <field id="HAZARD" direction="inout" datatype="Text">$MAE1</field>
  </fields>
</entity>
```

## Updating Tests

Test field elements where the direction attribute value is "in" or "inout" update the value of the corresponding fields in the SampleManager database. This behavior does not apply to key0 fields. The following example updates the value of the Instrument field to GM\_01:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
        <field id="INSTRUMENT" direction="inout" datatype="Text">GM_01</field>
      </fields>
      <children>
        ...results
      </children>
    </entity>
  </children>
</entity>
```

The LIMSML\_SAMPLE\_UPDATE\_TESTS constant in \$LIMSML\_SAMPLE\_USER determines whether the Result Entry action updates fields in test records. If LIMSML\_SAMPLE\_UPDATE\_TESTS is TRUE, SampleManager updates test fields. The default value is TRUE.

GLOBAL CONSTANT LIMSML\_SAMPLE\_UPDATE\_TESTS = "TRUE"

The LIMSML\_SAMPLE\_UPDATE\_TESTS value can be overridden using the [UPDATE\\_TESTS parameter](#) (page 43) in the LIMSML request. For example:

```
<entity type="SAMPLE">
  <actions>
    <action>
      <command>RESULT_ENTRY</command>
      <parameter name="UPDATE_TESTS">FALSE</parameter>
    </action>
  </actions>
  <fields>
    <field id="ID_NUMERIC" direction="in" datatype="Text">1522</field>
  </fields>
  <children>
    <entity type="TEST">
      <actions />
      <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
      </fields>
    </entity>
  </children>
</entity>
```

```

        <field id="INSTRUMENT" direction="inout" datatype="Text">GM_01</field>
    </fields>
    <children>
        ...results
    </children>
</entity>
</children>
</entity>

```

## Forcing Test Completion

The Result Entry action can force the status of a test to Complete (C) after results are entered using LIMSML.

The LIMSML\_SAMPLE\_COMPLETE\_TESTS constant in \$LIMSML\_SAMPLE\_USER determines whether the Result Entry action forces the status of tests to Complete. If LIMSML\_SAMPLE\_COMPLETE\_TESTS is TRUE, SampleManager completes tests when results are entered using LIMSML. The default value is FALSE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_COMPLETE_TESTS = "FALSE"
```

The LIMSML\_SAMPLE\_COMPLETE\_TESTS value can be overridden by using the [COMPLETE\\_TESTS](#) parameter (page 40) in the LIMSML request. For example:

```

<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>RESULT_ENTRY</command>
          <parameter name="CREATE_SAMPLE">TRUE</parameter>
          <parameter name="CREATE_SAMPLE_TEMP">BLANK</parameter>
          <parameter name="CREATE_TESTS">TRUE</parameter>
          <parameter name="COMPLETE_TESTS">TRUE</parameter>
        </action>
      </actions>
      <fields />
      <children>
        <entity type="TEST">
          <actions />
          <fields>
            <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
          </fields>
          <children>
            <entity type="RESULT">
              <actions />
              <fields>
                <field id="COMPONENT_NAME" direction="in"
datatype="Text">One</field>
                <field id="TEXT" direction="in" datatype="Text">1</field>

```

```

        </fields>
        <children />
    </entity>
    <entity type="RESULT">
        <actions />
        <fields>
            <field id="COMPONENT_NAME" direction="in"
datatype="Text">Two</field>
            <field id="TEXT" direction="in" datatype="Text">2</field>
        </fields>
        <children />
    </entity>
</children>
</entity>
<entity type="TEST">
    <actions />
    <fields>
        <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
    </fields>
    <children>
        <entity type="RESULT">
            <actions />
            <fields>
                <field id="COMPONENT_NAME" direction="in"
datatype="Text">One</field>
                <field id="TEXT" direction="in" datatype="Text">11</field>
            </fields>
            <children />
        </entity>
        <entity type="RESULT">
            <actions />
            <fields>
                <field id="COMPONENT_NAME" direction="in"
datatype="Text">Two</field>
                <field id="TEXT" direction="in" datatype="Text">12</field>
            </fields>
            <children />
        </entity>
    </children>
</entity>
</children>
</entity>
</system>
</transaction>

```

## Forcing Sample Completion

The Result Entry action can force the status of a sample to Complete (C) after results are entered using LIMSML.

The LIMSML\_SAMPLE\_COMPLETE\_SAMPLE constant in \$LIMSML\_SAMPLE\_USER determines whether the Result Entry action forces the status of samples to Complete. If LIMSML\_SAMPLE\_COMPLETE\_TESTS is TRUE, SampleManager completes samples when results are entered using LIMSML. The default value is FALSE.

```
GLOBAL CONSTANT LIMSML_SAMPLE_COMPLETE_SAMPLE = "FALSE"
```

The LIMSML\_SAMPLE\_COMPLETE\_SAMPLE value can be overridden by using the COMPLETE\_SAMPLE parameter in the LIMSML request. For example:

```
<?xml version="1.0" encoding="utf-8"?>
<transaction xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" id="TRAN1">
  <system response_type="system"
xmlns="http://www.thermo.com/informatics/xmlns/limsml/1.0">
    <entity type="SAMPLE">
      <actions>
        <action>
          <command>RESULT_ENTRY</command>
          <parameter name="CREATE_SAMPLE">TRUE</parameter>
          <parameter name="CREATE_SAMPLE_TEMP">BLANK</parameter>
          <parameter name="CREATE_TESTS">TRUE</parameter>
          <parameter name="COMPLETE_SAMPLE">TRUE</parameter>
        </action>
      </actions>
      <fields />
      <children>
        <entity type="TEST">
          <actions />
          <fields>
            <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
          </fields>
          <children>
            <entity type="RESULT">
              <actions />
              <fields>
                <field id="COMPONENT_NAME" direction="in"
datatype="Text">One</field>
                <field id="TEXT" direction="in" datatype="Text">1</field>
              </fields>
            </entity>
            <entity type="RESULT">
              <actions />
              <fields>
                <field id="COMPONENT_NAME" direction="in"
datatype="Text">Two</field>
                <field id="TEXT" direction="in" datatype="Text">2</field>
              </fields>
            </entity>
          </children>
        </entity>
      </children>
    </entity>
  </system>
</transaction>
```

```

        <children />
      </entity>
    </children>
  </entity>
  <entity type="TEST">
    <actions />
    <fields>
      <field id="ANALYSIS" direction="in" datatype="Text">MAE-IM</field>
    </fields>
    <children>
      <entity type="RESULT">
        <actions />
        <fields>
          <field id="COMPONENT_NAME" direction="in"
datatype="Text">One</field>
          <field id="TEXT" direction="in" datatype="Text">11</field>
        </fields>
        <children />
      </entity>
      <entity type="RESULT">
        <actions />
        <fields>
          <field id="COMPONENT_NAME" direction="in"
datatype="Text">Two</field>
          <field id="TEXT" direction="in" datatype="Text">12</field>
        </fields>
        <children />
      </entity>
    </children>
  </entity>
</children>
</system>
</transaction>

```

## Post Test Update Callback Routine

The `limsml_result_entry_post_test_update` routine in `$LIMSML_SAMPLE_USER` is called after the test fields and test status change has been completed for a test. This routine is executed for each test in a transaction.

Two parameters are passed to the routine:

- `test_entity` is a VGL object that represents the test defined in the LIMSML. The `test_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSML_HELPER` report.
- `test_number` is the value of the Test Number for the test.

```
GLOBAL ROUTINE limsml_result_entry_post_test_update (      test_entity      ,
                                                         VALUE test_number  )

{
* Called after the results are entered for a test
*
* Parameters   : Test Entity, [HELPER_ENTITY] Test Entity Helper Entity
*               Test Number, [PACKED_DECIMAL] Test Number
*
*****}

ENDROUTINE
```

## Post Sample Update Callback Routine

The `limsml_result_entry_post_sample_update` routine in `$LIMSML_SAMPLE_USER` is called after all of the updates to a sample have been completed, including field updates, status changes, and updates to all of its associated tests and results. Two parameters are passed to the routine:

- `sample_entity` is a VGL object that represents the sample defined in the LIMSML. The `sample_entity` object is based on the `HELPER_ENTITY` class, defined in the `$LIMSML_HELPER` report.
- `id_numeric` is the value of the Id Numeric for the sample.

```
GLOBAL ROUTINE limsml_result_entry_post_sample_update (      sample_entity  ,
                                                           VALUE id_numeric   )

{
* Called after all the results are entered for a sample
*
* Parameters   : Sample Entity, [HELPER_ENTITY] Sample Entity Helper Entity
*               ID Numeric, [IDENTITY] Sample Identity
*
*****}

ENDROUTINE
```

# LIMSML

LIMSML is an XML schema used to transfer data between the SampleManager database and the SampleManager web site, through a web service. LIMSML is a simple schema that encodes the SampleManager data; the commands that can be executed within SampleManager are stored in the database as [LIMSML entity actions](#) (page 9).

## Contents

- [LIMSML Hierarchy](#)
- [Element and Attribute Definitions](#)

## LIMSML Hierarchy

The following is an overview of the structure of a LIMSML document. This is not an exact definition, but a representation of the order and level of the various elements within the XML hierarchy.

- <lmsml> root element
  - <header> header information for document
    - <parameter> parameter associated with the whole document
  - <body> the body of the document
    - <transaction> a single transaction
      - <system> LIMS system logic
        - <entity> a system entity
          - <actions> actions to perform on the entity
            - <action> a single action to perform
              - <command> the command to perform
              - <parameter> parameter associated with the command
          - <fields> fields for the entity
            - <field> a field
          - <children> collection of child entities
            - <entity> a system entity
      - <data> pure data
      - <signature> signature associated with the transaction
        - <username> LIMS username associated with the signature
        - <password> LIMS password associated with the signature
        - <reason> prompted reason for the signature
        - <comments> comments associated with the signature
        - <menuproc> LIMS menu procedure number generating the signature
        - <clientname> name of the client that generated the signature
        - <clientaddress> IP address of the client that generated the signature
        - <clientdate> date and time of the signature on the client machine
    - <errors> errors associated with the response
      - <error> an error
      - <summary> summary of the error
      - <description> full description of the error
      - <code> error code
      - <severity> severity of the error
      - <source> source of the error
      - <errors> collection of child errors

## Element and Attribute Definitions

Elements in this section are arranged in hierarchical order. See [LIMSML Hierarchy](#) (page 168) for a representation of this order.

### <limsml>

The following table describes the details of the <limsml> element:

| Content | Type | Parent Element      |
|---------|------|---------------------|
| eltOnly |      | None – root element |

### Child Elements

The following table describes the child elements of the <limsml> element:

| Name                                      | Minimum Occurrences | Maximum Occurrences | Required |
|-------------------------------------------|---------------------|---------------------|----------|
| <a href="#">&lt;header&gt;</a> (page 170) | 1                   | 1                   | Yes      |
| <a href="#">&lt;body&gt;</a> (page 172)   | 1                   | 1                   | Yes      |
| <a href="#">&lt;errors&gt;</a> (page 189) | 0                   | 1                   | No       |

### Attributes

None

### Description

This is the root element for a limsml file. The <header>, <body>, and <errors> elements must occur in the order shown in the Child Elements table.

## Notes

The basic structure for a limsml file is as follows:

```
<limsml>
  <header>
    ...header information for the document
  </header>
  <body>
    ...the data
  </body>
  <errors>
    ...any errors produced by the document - this section is optional
  </errors>
</limsml>
```

## <header>

The following table describes the details of the <header> element:

| Content | Type | Parent Element                            |
|---------|------|-------------------------------------------|
| eltOnly |      | <a href="#">&lt;limsml&gt;</a> (page 169) |

## Child Elements

The following table describes the child elements of the <header> element:

| Name                                         | Minimum Occurrences | Maximum Occurrences | Required |
|----------------------------------------------|---------------------|---------------------|----------|
| <a href="#">&lt;parameter&gt;</a> (page 171) | 0                   | unbounded           | No       |

## Attributes

None

## Description

This is the document header for the limsml file. Any number of parameter elements can be added to the header section to define information which applies to the entire document.

## Notes

The header element must be present in a limsml document, but it may be left empty.

## <parameter>

The following table describes the details of the <parameter> element:

| Content  | Type   | Parent Element                              |
|----------|--------|---------------------------------------------|
| textOnly | string | <header> (page 170),<br><action> (page 178) |

### Child Elements

None

### Attributes

The following table describes the attributes of the <parameter> element:

| Name | Type | Default | Required |
|------|------|---------|----------|
| name | name |         | Yes      |

### Description

This element holds a generic parameter or property. The name of the parameter is contained in the name attribute, while the content of the element is used for the parameter value.

### Notes

In the following example, a parameter element supplies the name of the sample template to use for sample login:

```
...
<system response_type="system">
  <entity type="sample">
    <actions>
      <action>
        <command>create_by_template</command>
        <parameter name="template">tank_1_effluent</parameter>
      </action>
    </actions>
  </entity>
</system>
...
```

## <body>

The following table describes the details of the <body> element:

| Content | Type | Parent Element                            |
|---------|------|-------------------------------------------|
| eltOnly |      | <a href="#">&lt;limsml&gt;</a> (page 169) |

## Child Elements

The following table describes the child elements of the <body> element:

| Name                                           | Minimum Occurrences | Maximum Occurrences | Required |
|------------------------------------------------|---------------------|---------------------|----------|
| <a href="#">&lt;transaction&gt;</a> (page 173) | 0                   | unbounded           | No       |

## Attributes

None

## Description

This element contains the document data. The data is contained in one or more transaction elements – each corresponding to a single database transaction.

## Notes

None

## <transaction>

The following table describes the details of the <transaction> element:

| Content | Type            | Parent Element    |
|---------|-----------------|-------------------|
| eltOnly | transactionType | <body> (page 172) |

### Child Elements

The transaction element must contain at least one system or data element:

| Name                   | Minimum Occurrences | Maximum Occurrences | Required |
|------------------------|---------------------|---------------------|----------|
| <system> (page 174)    | 1                   | unbounded           | No       |
| AND / OR               |                     |                     |          |
| <data> (page 183)      | 1                   | unbounded           | No       |
| <signature> (page 184) | 0                   | 1                   | No       |

### Attributes

The following table describes the attributes of the <transaction> element:

| Name | Type   | Default | Required |
|------|--------|---------|----------|
| id   | string |         | No       |

### Description

This element defines a single transaction. It must contain at least one <system> or <data> element, but it can contain a combination of any number of the two types of elements.

The system element contains a collection of LIMS entities, while the data element contains an embedded section of XML data.

### Notes

None

## <system>

The following table describes the details of the <system> element:

| Content | Type | Parent Element                                 |
|---------|------|------------------------------------------------|
| eltOnly |      | <a href="#">&lt;transaction&gt;</a> (page 173) |

## Child Elements

The following table describes the child elements of the <system> element:

| Name                                      | Minimum Occurrences | Maximum Occurrences | Required |
|-------------------------------------------|---------------------|---------------------|----------|
| <a href="#">&lt;entity&gt;</a> (page 175) | 0                   | unbounded           | No       |

## Attributes

The following table describes the attributes of the <system> element:

| Name          | Type   | Default | Required |
|---------------|--------|---------|----------|
| response_type | NCName | system  | Yes      |
| id            | string |         | No       |

## Description

This element contains LIMS system logic.

The response\_type attribute defines the type of response element that is required. There are two possible values:

- system – returns the response in a <system> element
- data – returns the response in a <data> element

## Notes

The following example shows a system element that contains 3 sample entities:

```
...
<system response_type="system">
  <entity type="sample">
    ...sample 1 details
  </entity>
  <entity type="sample">
    ...sample 2 details
  </entity>
  <entity type="sample">
    ...sample 3 details
  </entity>
</system>
...
```

## <entity>

The following table describes the details of the <entity> element:

| Content | Type | Parent Element                                |
|---------|------|-----------------------------------------------|
| eltOnly |      | <system> (page 174),<br><children> (page 182) |

## Child Elements

The following table describes the child elements of the <entity> element:

| Name                  | Minimum Occurrences | Maximum Occurrences | Required |
|-----------------------|---------------------|---------------------|----------|
| <actions> (page 177)  | 0                   | 1                   | No       |
| <fields> (page 180)   | 0                   | 1                   | No       |
| <children> (page 182) | 0                   | 1                   | No       |

## Attributes

The following table describes the attributes of the <entity> element:

| Name | Type   | Default | Required |
|------|--------|---------|----------|
| type | NCName |         | Yes      |

## Description

This element represents a LIMS entity.

The type attribute defines the type of entity, such as sample, test, or hazard.

The child elements must appear in the order shown in the Child Elements table above.

## Notes

The following is the general structure for an entity element:

```
...
  <entity type="entity_type">
    <actions>
      ...actions to perform on the entity
    </actions>
    <fields>
      ...relevant fields belonging to the entity
    </fields>
    <children>
      ...relevant child elements of the entity
    </children>
  </entity>
...
```

## <actions>

The following table describes the details of the <actions> element:

| Content | Type | Parent Element      |
|---------|------|---------------------|
| eltOnly |      | <entity> (page 175) |

## Child Elements

The following table describes the child elements of the <actions> element:

| Name                | Minimum Occurrences | Maximum Occurrences | Required |
|---------------------|---------------------|---------------------|----------|
| <action> (page 178) | 0                   | unbounded           | No       |

## Attributes

None

## Description

Contains the action to be performed.

## Notes

SampleManager only allows one action to be performed per entity. Hence when limsml files are used to pass data to and from a SampleManager system, an <actions> element can only contain one <action> element.

## <action>

The following table describes the details of the <action> element:

| Content | Type | Parent Element       |
|---------|------|----------------------|
| eltOnly |      | <actions> (page 177) |

## Child Elements

The following table describes the child elements of the <action> element:

| Name                   | Minimum Occurrences | Maximum Occurrences | Required |
|------------------------|---------------------|---------------------|----------|
| <command> (page 179)   | 1                   | 1                   | Yes      |
| <parameter> (page 171) | 0                   | unbounded           | No       |

## Attributes

The following table describes the attributes of the <action> element:

| Name        | Type               | Default | Required |
|-------------|--------------------|---------|----------|
| repeatcount | nonNegativeInteger |         | No       |

## Description

Defines an action to perform. The optional repeatcount attribute can be used to specify how many times the action should be repeated.

The child elements must appear in the order shown in the Child Elements table above.

Notes

The following example demonstrates the use of the action element and its repeatcount attribute. The section of limsml below would log in three samples using the tank\_1\_effluent template:

```
...
  <entity type="sample">
    <actions>
      <action repeatcount="3">
        <command>create_by_template</command>
        <parameter
name="template">tank_1_effluent</parameter>
      </action>
    </actions>
  </entity>
...
```

<command>

The following table describes the details of the <command> element:

| Content  | Type | Parent Element       |
|----------|------|----------------------|
| textOnly | Name | <actions> (page 177) |

Child Elements

None

Attributes

None

Description

A command to perform. Any parameters required by the command can be supplied in sibling [parameter](#) (page 171) elements.

Notes

None

## <fields>

The following table describes the details of the <fields> element:

| Content | Type | Parent Element      |
|---------|------|---------------------|
| eltOnly |      | <entity> (page 175) |

## Child Elements

The following table describes the child elements of the <fields> element:

| Name               | Minimum Occurrences | Maximum Occurrences | Required |
|--------------------|---------------------|---------------------|----------|
| <field> (page 181) | 0                   | unbounded           | No       |

## Attributes

None

## Description

Contains a set of fields for a single entity.

## Notes

None

## <field>

The following table describes the details of the <field> element:

| Content  | Type   | Parent Element      |
|----------|--------|---------------------|
| textOnly | string | <fields> (page 180) |

### Child Elements

None

### Attributes

The following table describes the attributes of the <field> element:

| Name      | Type    | Default | Required |
|-----------|---------|---------|----------|
| id        | Name    |         | Yes      |
| direction | NMToken | in      | No       |

### Description

Defines the value of a field. The id attribute contains the field name.

The optional direction attribute defines whether the field value is an input to the database, a returned value from the database, or both. There are three possible values for this attribute:

- in – input value
- out – output value
- inout – input and output

### Notes

None

## <children>

The following table describes the details of the <children> element:

| Content | Type | Parent Element      |
|---------|------|---------------------|
| eltOnly |      | <entity> (page 175) |

## Child Elements

The following table describes the child elements of the <children> element:

| Name                | Minimum Occurrences | Maximum Occurrences | Required |
|---------------------|---------------------|---------------------|----------|
| <entity> (page 175) | 1                   | unbounded           | Yes      |

## Attributes

None

## Description

This element contains the child entities of the entity contained in its parent. For example a sample entity might contain a <children> element, which in turn contains the <entity> elements that define its tests.

## Notes

The following limsmml section shows a sample with an associated test and result:

```
...
<entity type="sample">
  <fields>
    <field id="id_numeric">42</field>
  </fields>
  <children>
    <entity type="test">
      <fields>
        <field id="analysis">PHOSPHATE</field>
      </fields>
      <children>
        <entity type="result">
          <fields>
            <field id="name">phosphate</field>
            <field id="text">1.234</field>
            <field id="units">ppm</field>
          </fields>
        </entity>
      </children>
    </entity>
  </children>
</entity>
```

```
    </entity>  
  </children>  
</entity>  
...
```

## <data>

The following table describes the details of the <data> element:

| Content | Type | Parent Element                                 |
|---------|------|------------------------------------------------|
| eltOnly |      | <a href="#">&lt;transaction&gt;</a> (page 173) |

## Child Elements

The data element has no specific child elements defined in the limsml schema. When this element is used in a limsml document, its child elements are the top-level elements of the embedded xml sections that it encloses.

| Name               | Minimum Occurrences | Maximum Occurrences | Required |
|--------------------|---------------------|---------------------|----------|
| <i>any element</i> | 0                   | unbounded           | No       |

## Attributes

The following table describes the attributes of the <data> element:

| Name | Type   | Default | Required |
|------|--------|---------|----------|
| id   | string |         | No       |

## Description

This element contains embedded xml data.

## Notes

The following attributes apply to the top-level element of the embedded XML section:

| Name          | Type   | Default | Required |
|---------------|--------|---------|----------|
| format        | string |         | No       |
| response_type | NCName | system  | Yes      |

| Name | Type   | Default | Required |
|------|--------|---------|----------|
| id   | string |         | No       |

The response\_type attribute defines the type of response element that is required. There are two possible values:

- system – returns the response in a <system> element
- data – returns the response in a <data> element

## <signature>

The following table describes the details of the <signature> element:

| Content | Type          | Parent Element           |
|---------|---------------|--------------------------|
| eltOnly | signatureType | <transaction> (page 173) |

## Child Elements

The following table describes the child elements of the <signature> element:

| Name                       | Minimum Occurrences | Maximum Occurrences | Required |
|----------------------------|---------------------|---------------------|----------|
| <username> (page 185)      | 1                   | 1                   | Yes      |
| <password> (page 186)      | 1                   | 1                   | Yes      |
| <reason> (page 186)        | 1                   | 1                   | Yes      |
| <comments> (page 187)      | 0                   | 1                   | Yes      |
| <menuproc> (page 187)      | 1                   | 1                   | Yes      |
| <clientname> (page 188)    | 1                   | 1                   | Yes      |
| <clientaddress> (page 188) | 1                   | 1                   | Yes      |

### Attributes

None

### Description

This element contains the signature associated with the transaction.

### Notes

None

### <username>

The following table describes the details of the <username> element:

| Content  | Type | Parent Element                               |
|----------|------|----------------------------------------------|
| textOnly |      | <a href="#">&lt;signature&gt;</a> (page 184) |

### Attributes

None

### Description

This element contains the LIMS username associated with the signature.

### Notes

None

## <password>

The following table describes the details of the <password> element:

| Content  | Type | Parent Element                               |
|----------|------|----------------------------------------------|
| textOnly |      | <a href="#">&lt;signature&gt;</a> (page 184) |

### Attributes

None

### Description

This element contains the LIMS password associated with the signature.

### Notes

The LIMS password can be encrypted.

## <reason>

The following table describes the details of the <reason> element:

| Content  | Type | Parent Element                               |
|----------|------|----------------------------------------------|
| textOnly |      | <a href="#">&lt;signature&gt;</a> (page 184) |

### Attributes

None

### Description

This element contains the prompted reason for the signature.

### Notes

None

## <comments>

The following table describes the details of the <comments> element:

| Content  | Type | Parent Element         |
|----------|------|------------------------|
| textOnly |      | <signature> (page 184) |

### Attributes

None

### Description

This element contains the comments associated with the <signature> element.

### Notes

None

## <menuproc>

The following table describes the details of the <menuproc> element:

| Content  | Type | Parent Element         |
|----------|------|------------------------|
| textOnly |      | <signature> (page 184) |

### Attributes

None

### Description

This element contains the LIMS menu procedure number that generated the signature.

### Notes

None

## <clientname>

The following table describes the details of the <clientname> element:

| Content  | Type | Parent Element         |
|----------|------|------------------------|
| textOnly |      | <signature> (page 184) |

### Attributes

None

### Description

This element contains the name of the client that generated the signature.

### Notes

None

## <clientaddress>

The following table describes the details of the <clientaddress> element:

| Content  | Type | Parent Element         |
|----------|------|------------------------|
| textOnly |      | <signature> (page 184) |

### Attributes

None

### Description

This element contains the IP address of the client that generated the signature.

### Notes

None

## <errors>

The following table describes the details of the <errors> element:

| Content | Type | Parent Element                          |
|---------|------|-----------------------------------------|
| eltOnly |      | <limsml> (page 169), <error> (page 190) |

## Child Elements

The following table describes the child elements of the <errors> element:

| Name               | Minimum Occurrences | Maximum Occurrences | Required |
|--------------------|---------------------|---------------------|----------|
| <error> (page 190) | 0                   | unbounded           | No       |

## Attributes

None

## Description

This element contains a collection of errors for use in a response.

## Notes

None

## <error>

The following table describes the details of the <error> element:

| Content | Type | Parent Element      |
|---------|------|---------------------|
| eltOnly |      | <errors> (page 189) |

## Child Elements

The following table describes the child elements of the <error> element:

| Name                     | Minimum Occurrences | Maximum Occurrences | Required |
|--------------------------|---------------------|---------------------|----------|
| <summary> (page 191)     | 1                   | 1                   | Yes      |
| <description> (page 192) | 0                   | 1                   | No       |
| <code> (page 193)        | 0                   | 1                   | No       |
| <severity> (page 194)    | 0                   | 1                   | No       |
| <source> (page 195)      | 0                   | 1                   | No       |
| <errors> (page 189)      | 0                   | 1                   | No       |

## Attributes

None

## Description

This element describes an error. The child elements must appear in the order shown in the Child Elements table above.

## Notes

None

**<summary>**

The following table describes the details of the <summary> element:

| Content  | Type   | Parent Element |
|----------|--------|----------------|
| textOnly | string |                |

**Child Elements**

None

**Attributes**

None

**Description**

This element contains a summary of the error.

**Notes**

None

## <description>

The following table describes the details of the <description> element:

| Content  | Type   | Parent Element |
|----------|--------|----------------|
| textOnly | string |                |

### Child Elements

None

### Attributes

None

### Description

A full description of the error.

### Notes

None

**<code>**

The following table describes the details of the <code> element:

| Content  | Type   | Parent Element                           |
|----------|--------|------------------------------------------|
| textOnly | string | <a href="#">&lt;error&gt;</a> (page 190) |

**Child Elements**

None

**Attributes**

None

**Description**

The error code.

**Notes**

None

## <severity>

The following table describes the details of the <severity> element:

| Content  | Type          | Parent Element     |
|----------|---------------|--------------------|
| textOnly | anySimpleType | <error> (page 190) |

### Child Elements

None

### Attributes

None

### Description

Severity of the error.

### Notes

None

**<source>**

The following table describes the details of the <source> element:

| Content  | Type   | Parent Element     |
|----------|--------|--------------------|
| textOnly | string | <error> (page 190) |

**Child Elements**

None

**Attributes**

None

**Description**

Source of the error (the error's xpath).

**Notes**

None



## Installing the SampleManager Web Application

The SampleManager Web Application is a legacy feature that provides access to a limited subset of SampleManager functions through the associated SampleManager web site.

For information about the new SampleManager web client, see the *SampleManager LIMS Web Client Installation and Configuration Guide*.

Before installing the SampleManager Web application, you must carefully consider the security of your system. For example, if your web site is to be exposed to the Internet, it will not be protected by your firewall. In this situation, your SampleManager system and its wcf service must be installed on a separate machine, within a firewall.

Before you begin the installation, take the following steps:

- Make sure that you have a full backup of your system and that any work in progress is saved before attempting installation.
- If there are other web sites or services running on the server, make appropriate plans for possible disruption to their operations. The World Wide Web publishing service on the target system might need to be restarted to complete installation. Restarting the publishing service will temporarily stop any existing web sites or services on the system.
- Exit all currently running applications as they might affect the operation of the setup program.

### Contents

- [Minimum System Requirements](#)
- [Overall Installation Procedure](#)
- [Upgrading the SampleManager Web Application](#)

## Minimum System Requirements

The following table identifies the minimum system requirements for the SampleManager Web application:

| Minimum System Requirements |                            |
|-----------------------------|----------------------------|
| Hardware                    | Pentium 4 1.6 GHz          |
|                             | 512 MB RAM                 |
|                             | 50 MB hard disk space      |
| Operating System            | Windows Server 2008 R2 SP1 |
|                             | IIS v5.0 or higher         |

The SampleManager Web Application requires Microsoft .NET Framework version 1.1. You can download all editions of the .NET Framework from:

<http://www.microsoft.com>

Install the Microsoft .NET Framework version 1.1 before you install any later version of the .NET Framework. It is not possible to install the .NET Framework version 1.1 after a later version of the .NET Framework.

Windows Server operating systems later than Windows Server 2008 R2 SP1 do not support Microsoft .NET Framework version 1.1.

## Overall Installation Procedure

The installation of the SampleManager Web application requires the following steps:

- [Install the Web Server \(IIS\) role](#) (page 199), and the Web Server (IIS) role services required by the SampleManager Web application.
- [Install the SampleManager Web application](#) (page 200).
- [Configure the SampleManager Web site](#) (page 202) in Internet Information Services (IIS) Manager.
- [Edit the web.config file](#) (page 202) to supply the Web application with the URL of the wcf service for the target SampleManager instance.
- [Enable the Localization Editor](#) (page 204). This step is required if you want to localize the SampleManager web site.

## Installing Web Server (IIS)

Before you install the SampleManager Web application, you must ensure that the Web Server (IIS) role is installed on your server, and that it has the role services required by the SampleManager Web application.

See the documentation for your operating system for more information about installing Web Server (IIS).

The SampleManager Web application requires the following Web Server (IIS) role services:

- Web Server
  - Common HTTP Features
    - Static Content
    - Default Document
    - Directory Browsing
    - HTTP Errors
  - Application Development
    - ASP.NET
    - .NET Extensibility
    - ASP
    - CGI
    - ISAPI Extensions
    - ISAPI Filters
    - Server Side Includes
  - Health and Diagnostics
    - HTTP Logging
    - Request Monitor
  - Security
    - Request Filtering

- Performance
  - Static Content Compression
- Management Tools
  - IIS Management Console
  - IIS Management Scripts and Tools
  - Management Service
  - IIS 6 Management Compatibility
    - IIS 6 Metabase Compatibility
    - IIS 6 WMI Compatibility
    - IIS 6 Scripting Tools
    - IIS 6 Management Console

## Installing the SampleManager Web Application

The SampleManager Web application is installed by an InstallShield Wizard.

### ❖ To install the SampleManager Web application

1. In Windows Explorer, open the folder that contains your SampleManager installation kit.
2. In the SampleManager installation kit, open the following folder:  
    `..\Modules\SampleManager Web 11.0\Web`
3. Double-click **setup.exe**.

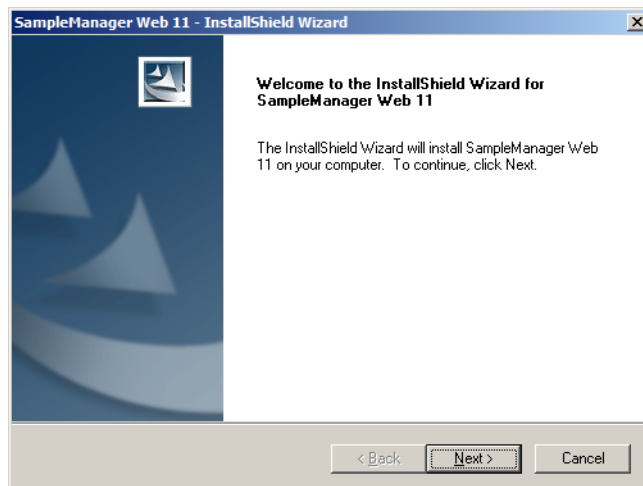
The SampleManager Web InstallShield Wizard appears, and checks for the Microsoft .NET Framework version 1.1.

4. If the Microsoft .NET Framework version 1.1 is not present, a dialog box appears asking if you want to install the .NET Framework version 1.1. Click **Yes** to install the .NET Framework version 1.1.

During the .NET Framework installation, dialog boxes might appear asking you to confirm that you want to proceed. Confirm that you want to continue in all the dialog boxes.

5. When the .NET Framework version 1.1 is installed, and configuration of your system is complete, the SampleManager Web InstallShield Wizard appears.

**Figure 5-1.** SampleManager Web Installation – Welcome Page



6. Click **Next** to continue.  
The License Agreement page appears.
7. Click **Yes** to accept the license agreement.  
The Customer Information page appears.
8. Enter the requested information in the User Name and Company Name fields.  
Leave the installation option set to **Anyone who uses this computer (all users)** and click **Next** to continue.  
The Choose Destination Location page appears.
9. Either accept the default location for the SampleManager Web application files, or browse for an alternative. Click **Next** to continue.  
The SampleManager Virtual Directory page appears.
10. Enter the name of the SampleManager Web application virtual directory then click **Next** to continue.  
The Start Copying Files page appears.  
This page shows information about the installation, such as the target folder for the SampleManager Web application files.
11. Review the settings and, if they are correct, click **Next** to start installing the software.  
The Setup Status screen page appears. This page shows the progress of the installation.  
When the installation is complete, the InstallShield Wizard Complete page appears.

12. Click **Finish** to close the InstallShield Wizard.

## Configuring the SampleManager Web Site

### ❖ To configure your SampleManager Web site in Internet Information Services (IIS) Manager

1. Add the SampleManager Web site to the ASP.NET 1.1 application pool.
2. Set the ISAPI and CGI restrictions for your server to **Allowed** for ASP.NET v1.1.4322.

Refer to your operating system documentation for detailed instructions.

## Editing the Web.Config File

Edit the web.config file for the SampleManager Web application so that it contains the URL of the wcf file for the target SampleManager instance.

### ❖ To edit the web.config file

1. Open the folder where the SampleManager Web application is installed. The default location is:  
  
C:\Program Files (x86)\Thermo\SampleManager\current\_release\Web  
  
Where *current\_release* is the current SampleManager version number.
2. Create a backup copy of the web.config file.
3. Open the web.config file in a text editor.
4. Locate the following section of the file:

```
<!-- Thermo web Services
Servers section display name and URL for each LIMSML Web Service. To add another server
duplicate the "LOCAL" section, rename LOCAL to something more applicable and then change
the URL to point at the LIMS web Service.
-->

<Thermo.SM.Web.Servers>

  <LOCAL>
    <URL>http://localhost/limswebService/default.asmx</URL>
    <Timeout>60000</Timeout>
    <Description>The SampleManager system on the same machine as the web
Server</Description>
  </LOCAL>
```

5. Edit the <Thermo.SM.Web.Servers> section so that it contains the details of the SampleManager server instance that you want the web site to connect to. The general format required is:

```
<SERVER_NAME>
  <URL>http://hostname:port_number</URL>
  <Timeout>60000</Timeout>
  <Description>Description of product instance.</Description>
</SERVER_NAME>
```

Where:

- *SERVER\_NAME* is the name that you want to use for the SampleManager instance.
- *hostname* is the name of the server where the SampleManager instance resides.
- *port\_number* is the port number for the wcf service associated with the SampleManager instance.
- *Description of product instance* is a brief description of the SampleManager instance.

For example:

```
<SMPROD>
  <URL>http://Production_SM:56104</URL>
  <Timeout>60000</Timeout>
  <Description>SampleManager production server.</Description>
</SMPROD>
```

**Note** You can obtain the port number for the wcf service from the Test SampleManager Service tool for the target instance.

6. Save the changes to the file.

You should now be able to connect to the SampleManager web site using a browser on the server where the SampleManager Web application is installed. The default URL for the SampleManager web site on the current machine is:

http://localhost/SampleManager

Configure the firewall settings on the server appropriately to allow authorized users on other machines to access the web site. Refer to your operating system documentation for more information about firewall settings.

## Enabling the Localization Editor

The SampleManager web interface contains a Localization Editor, which can be used by suitably authorized users to configure and edit the on-screen messages that appear. See the [Localization Editor](#) (page 229) chapter of this manual for more information.

SampleManager stores localization resources for the web interface in the Web.i18 file, which is installed in the following folder by default:

```
..\Program Files\Thermo\SampleManager\current_release\web\
```

Where *current\_release* is the current SampleManager version number.

Before the Localization Editor can be used, the appropriate network service account permissions must be assigned to the Web.i18 file.

### ❖ To assign network service account permissions to the Web.i18 file

1. Navigate to the folder where the Web.i18 file is stored.
2. In the Explorer pane, right-click the file and select **Properties**.

The property sheet for the Web.i18 file appears.

3. Click the **Security** tab.

The Security page shows a list of network service accounts in the upper pane.

4. Select an account in the upper pane.

The lower pane shows the permissions that are assigned to the account that is selected in the upper pane.

**Note** If a user requires access to all the functions of the Localization Editor, their account must have Read and Write access to the Web.i18 file.

5. Assign the appropriate account permissions and click **OK** to save the changes.

## Updating the Encryption DLL

From SampleManager LIMS version 12.2, SampleManager uses Unicode to handle text within the application. To support Unicode, the web application server requires an updated DLL for the encryption functionality.

### ❖ To update the encryption DLL

1. On the SampleManager web application server, in Windows Explorer, navigate to the following folder in your SampleManager installation kit:

`\Modules\SampleManager Web 12.2 update\Web\bin`

2. Copy the Thermo.SM.LIMSML.Helper.dll file into the bin folder of your SampleManager web application. The default path for the bin folder is:

`C:\Program Files (x86)\Thermo\SampleManager\11.0\Web\bin`

3. Restart the web application server.

## Upgrading the SampleManager Web Application

From SampleManager LIMS version 12.2, SampleManager uses Unicode to handle text within the application. To support Unicode, the web application server requires an updated DLL for the encryption functionality.

If your SampleManager web application was installed for a SampleManager version older than SampleManager LIMS 12.2, follow the instructions in [Updating the Encryption DLL](#) (page 205) to update the encryption functionality.



## The SampleManager Web Site

The SampleManager web site enables a user to access a selected set of SampleManager functionality through a web browser.

To access the site, users must log in with a current SampleManager username and password. They can then access any data or functionality made available on the site, subject to the standard SampleManager group and role security for their account.

Information that is displayed in the SampleManager web interface can be divided into two distinct categories:

- Data driven: Dynamic data items retrieved from the LIMS.
- Static: On-screen descriptions and textual information.

The appearance of text and data in the SampleManager web interface is controlled by a 'Style' which is specified in the web interface's configuration files.

### Contents

- [Defining the Configuration Settings](#)
- [Styles](#)
- [Allowing LDAP Authentication](#)

## Defining the Configuration Settings

The configuration file used to define the settings for the website is called web.config and can be found in the following folder by default:

Program Files (x86)\Thermo\SampleManager\*current\_release*\web

Where *current\_release* is the current version of SampleManager.

The settings that can be configured are specified in the <Thermo.SM.Web> section of the file:

```
<Thermo.SM.Web>
  <add key="BrowsePageSize" value = "25"/>
  <add key="ExplorerPageSize" value = "25"/>
  <add key="PrettyErrors" value = "true"/>
  <add key="SampleLoginRemember" value = "true"/>
  <add key="JobLoginRemember" value = "true"/>
  <add key="CacheTemplates" value = "false"/>
  <add key="CacheReports" value = "false"/>
  <add key="CacheInterface" value = "true"/>
  <add key="TranslationMode" value = "debug"/>
  <add key="CacheExplorerAux" value = "false"/>
  <add key="DefaultStyle" value = "Thermo"/>
  <add key="LDAPUser" value = "false"/>
</Thermo.SM.Web>
```

| Setting             | Description                                                                                                                                                   |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BrowsePageSize      | Specifies the number of records to obtain from the LIMS, which will then be displayed on a single Browse page.                                                |
| ExplorerPageSize    | Specifies the number of records to obtain from the LIMS, which will then be displayed on a single Explorer page.                                              |
| PrettyErrors        | Set this item to "true" in order to handle errors using a specific handler rather than the default ASP.NET way. Errors handled in this way are more detailed. |
| SecurePages         | Set this item to "true" for the user to only be able to access pages in your menu from the web site.                                                          |
| SampleLoginRemember | Set this item to "true" to remember the last information used when logging in a sample                                                                        |
| JobLoginRemember    | Set this item to "true" to remember the last information used when logging in a job.                                                                          |
| CacheTemplates      | Set this item to "true" to cache details of previously used templates.                                                                                        |
| CacheReports        | Set this item to "true" to cache details of InfoMaker Report prompt details                                                                                   |

| Setting         | Description                                                                                                                                                                                                                                                            |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CacheInterface  | Set this item to "true" to cache details of interface prompts.                                                                                                                                                                                                         |
| TranslationMode | This setting determines the way localized messages are displayed. The three options are: <ul style="list-style-type: none"><li>• Debug: Displays the message ID's</li><li>• Blank: Does not display anything.</li><li>• Normal: Displays translated messages</li></ul> |
| DefaultStyle    | This item is used to determine the style used by the site. Styles are discussed in detail in the next section of this chapter.                                                                                                                                         |
| LDAPUser        | Set this item to "true" to enable LDAP authentication for login and electronic signatures.                                                                                                                                                                             |

## Styles

Styles are used to control the 'look' and 'feel' of the SampleManager web site. A style can be seen as a set of formatting characteristics, which can be applied to items in web pages.

Styles can be created and adapted to change the following characteristics of the SampleManager web interface:

- Color: the colors used in the web interface.
- Fonts: the fonts used to display text in the web interface.
- Graphics: the graphics and icons displayed in the web interface.

An individual style consists of a named folder, which in turn contains a cascading style sheet, a panelbar configuration file, and an images folder. The cascading style sheet (.css) and the panelbar (.pbx) configuration files must have the same name as the folder they are in. This can be any user specified name. Any new styles created should follow this structure.

The images folder should contain any images pertaining to that style. Whilst this folder does not have to be held within this hierarchy, it is strongly recommended that it is for ease of style maintenance.

The Styles folder is stored in the root Web folder:

Program Files (x86)\Thermo\SampleManager\current\_release\Web\Styles

Where *current\_release* is the current version of SampleManager.

## Customizing Styles

When first installed, the SampleManager web interface is displayed in a default "Thermo" style. The individual characteristics of this style are defined in the Thermo.css cascading style sheet.

Cascading style sheets are used to apply a collection of styles consistently across the entire web interface. They are similar to document templates. If a style is changed in a cascading style sheet, all the web interface pages will reflect the change. This makes it easier to control changes to the appearance of the web interface.

The Thermo.css cascading style sheet is stored in the following folder:

```
Program Files (x86)\Thermo\SampleManager\current_release\web\Styles\Thermo
```

Where *current\_release* is the current version of SampleManager.

The following additional styles are supplied with the SampleManager web interface:

- Blue
- Brown
- Green

Examples of these styles follow later in this section.

The style for the SampleManager web interface is controlled by the "DefaultStyle" setting in the web.config file. This file is located within the following folder:

```
Program Files (x86)\Thermo\SampleManager\current_release\web
```

Where *current\_release* is the current version of SampleManager.

**Note** The web.config file is installed with read-only properties. You will need to remove the read-only attribute before making any changes to this file.

### ❖ To apply a different style to the SampleManager web interface

1. Create a new folder within the Styles folder.
2. Create a new cascading style sheet and save it in the new folder. Alternatively, you can make a copy of an existing .css file, rename it, and modify the file to suit your requirements.
3. Open the web.config file, using Notepad or a similar text editing program.

4. Navigate to the following section of the file:

```
<Thermo.SM.web>
  <add key="DefaultStyle" value = "Thermo"/>
</Thermo.SM.web>
```

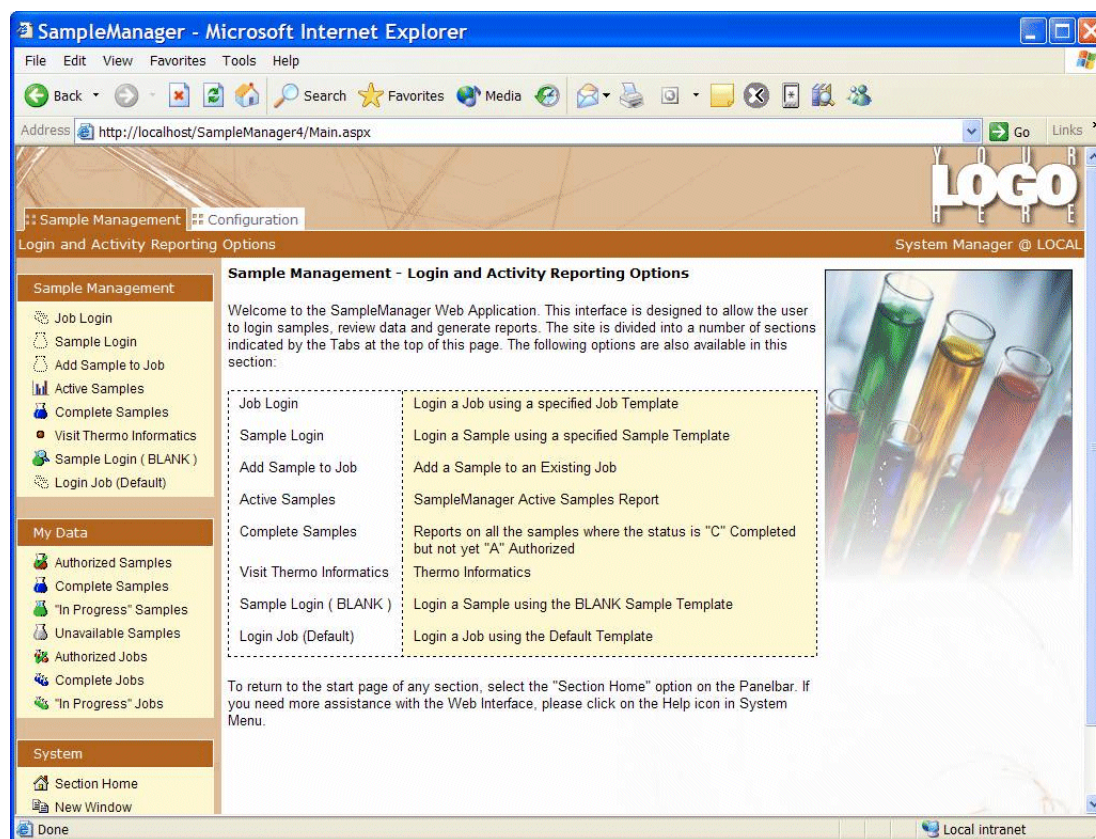
5. Edit the "DefaultStyle" entry, replacing the "Thermo" value with the name of the required folder.

6. Save the changes to the web.config file.

## Default Style – Brown

The following image shows an example of the default "Brown" style:

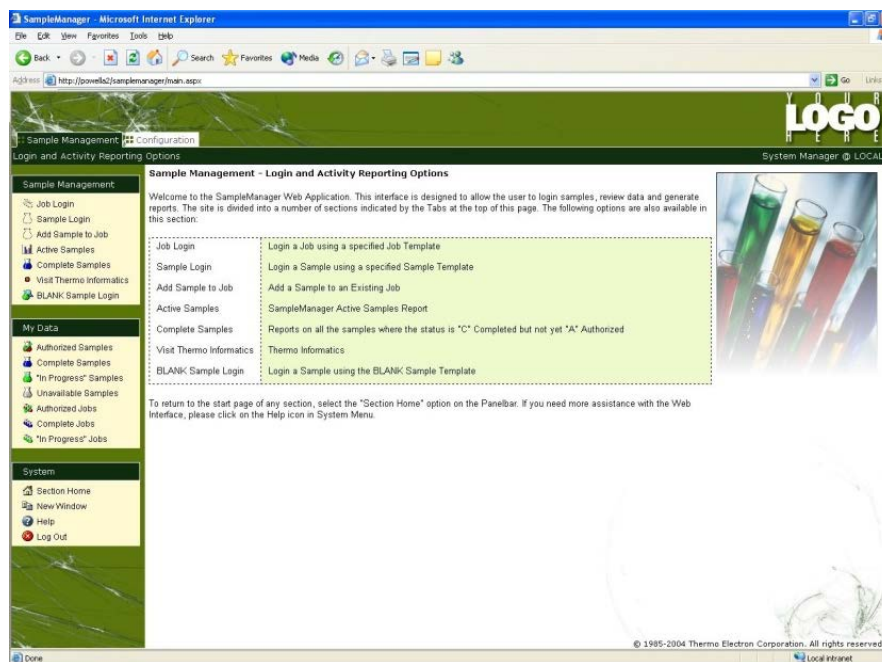
**Figure 6-1.** Example "Brown" Style



## Default Style – Green

The following image shows an example of the default "Green" style:

**Figure 6-2.** Example "Green" Style



## Default Style – Blue

The following image shows an example of the default "Blue" style:

**Figure 6-3.** Example "Blue" Style



## Panelbar Configuration

The appearance of the Panelbar, which is the menu bar displayed to the left of the SampleManager web interface, is controlled by the .pbx configuration file within the currently selected Styles folder. This file contains the following configurable details:

- Background color
- Button size and color
- Font size and color
- Panelbar icons and images

Additional Panelbar configuration files are supplied as further examples.

To change the appearance of the Panelbar, make a copy of an existing .pbx file, rename it, and amend the entries in the file to suit your requirements.

The .pbx file is an XML file and can be edited in any text editor or XML editing tool.

## Allowing LDAP Authentication

If your SampleManager server is configured to use LDAP authentication, the SampleManager website can be set up to allow users to enter their domain username and password when prompted to log in or to create an electronic signature.

To allow LDAP authentication, set the LDAPUser key in the configuration file to "true". The configuration file is called web.config and can be found in the following folder by default:

Program Files (x86)\Thermo\SampleManager\*current\_release*\web

Where *current\_release* is the current version of SampleManager.

The LDAPUser key is located in the <Thermo.SM.Web> section of the file:

```
<Thermo.SM.Web>
  <add key="BrowsePageSize" value = "25"/>
  <add key="ExplorerPageSize" value = "25"/>
  <add key="PrettyErrors" value = "true"/>
  <add key="SampleLoginRemember" value = "true"/>
  <add key="JobLoginRemember" value = "true"/>
  <add key="CacheTemplates" value = "false"/>
  <add key="CacheReports" value = "false"/>
  <add key="CacheInterface" value = "true"/>
  <add key="TranslationMode" value = "debug"/>
  <add key="CacheExplorerAux" value = "false"/>
  <add key="DefaultStyle" value = "Thermo"/>
  <add key="LDAPUser" value = "true"/>
</Thermo.SM.Web>
```

When the LDAPUser key is "true" there are no restrictions on the length of the username or password that can be entered, or on the characters that the username or password can contain.

## Defining Web Interface Menu Options

The menu options in the SampleManager web site are generated from data stored in the SampleManager database.

There are three types of menu option that can be added to the web site:

- Explorer folders
- InfoMaker reports
- Links to other pages

### Contents

- [Adding a Folder to the Web Site Menu](#)
- [Adding an InfoMaker Report to the Web Site Menu](#)
- [Adding a Link to the Web Site Menu](#)
- [LIMSML Entity Action Tester](#)

## Adding a Folder to the Web Site Menu

To add an existing SampleManager folder to the web site menu, you will need to perform the following steps:

1. Create a [master menu item](#) (page 216) for the folder.
2. Modify the [web menu](#) (page 218) that you wish to add the folder to.

## Creating a Master Menu Item for a Folder

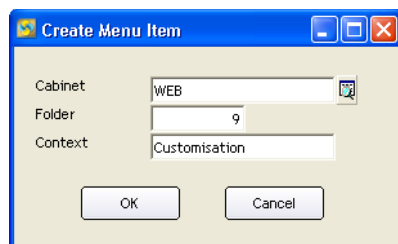
To add a SampleManager folder to the web interface, you must first add the folder to the Master Menu.

### ❖ To create a master menu item for a folder

1. Choose **Setup > Interface > Web Menu > Master Menu > Add Explorer Folder** from the SampleManager main menu.

The Create Menu Item dialog box appears.

**Figure 7-1.** Create Menu Item Dialog Box



2. In the Cabinet field, enter the Explorer cabinet where the folder is located.
3. In the Folder field, enter the numeric identifier for the folder.
4. Select the appropriate context for the item you are adding. The context is used to assign a numeric identity from a defined range to each type of menu item.

5. Click **OK**. The Master Menu property sheet appears.

**Figure 7-2.** Master Menu Property Sheet

The screenshot shows a Windows-style dialog box titled "Master Menu - Modify - 100001". It has several tabs: "General", "Display", "Implementation", "Advanced", "Parameters", and "Roles". The "General" tab is selected. Inside the dialog, there is a folder icon next to a text field containing "100001" and a "Generate" button. Below this is a "Description" field with the text "Effluent Samples.". Underneath is a "Type" dropdown menu set to "MENU". Further down are fields for "Modified On" (31/03/2010 16:27) and "Modified By" (System Manager). At the bottom is a "Modifiable" checkbox which is checked. At the very bottom of the dialog are "OK", "Cancel", and "Apply" buttons.

6. Complete any additional details as required then click **OK** to save the details and close the Master Menu property sheet.

The new master menu item must now be added to the appropriate section of the web menu. This procedure is described in the following section.

## Web Menu Section

To add the new master menu item to the appropriate section of the SampleManager web menu, choose **Setup > Interface > Web Menu > Modify** from the SampleManager main menu. The Web Menu Section property sheet appears.

Each web menu section represents a group of menu items that are displayed on the SampleManager web site.

### Web Menu Section – General Page

The General page contains basic information about the web menu section.

**Figure 7-3.** Web Menu Section – General Page

The screenshot shows a Windows-style dialog box titled "Web Menu Section - Modify - SYSTEM". It has four tabs: "General", "Display", "Content", and "Menu Items", with "General" selected. The "General" tab contains the following fields and controls:

- A tree view icon on the left and a text field containing "MAIN".
- A "Web Menu" label followed by a text field containing "SYSTEM".
- A "Name" label followed by a text field containing "Sample Management".
- A "Description" label followed by a text area containing "Login and Activity Reporting Options".
- A "Group Id" label followed by a dropdown menu.
- A "Modified On" label followed by a date/time field showing "18/09/2009 13:39".
- A "Modified By" label followed by a text field showing "BATCH".
- A "Modifiable" label followed by a checked checkbox.
- At the bottom right are three buttons: "OK", "Cancel", and "Apply".

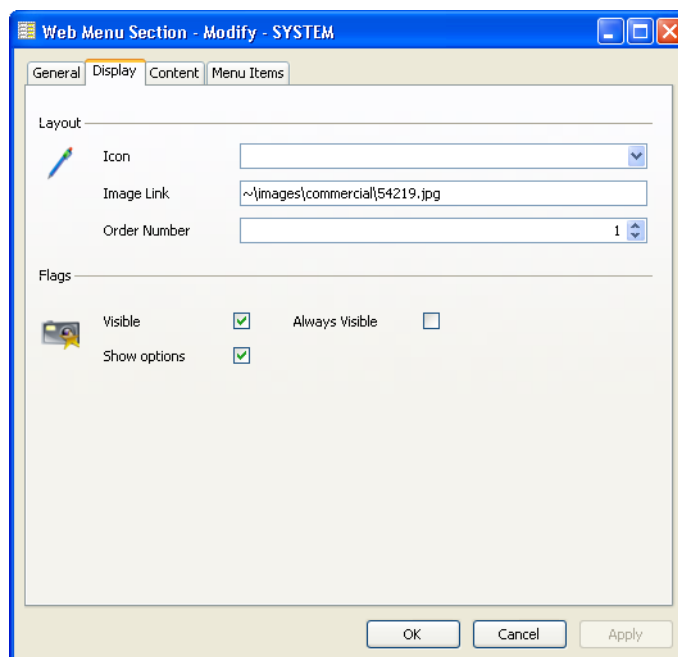
| Option      | Description                                                                                     |
|-------------|-------------------------------------------------------------------------------------------------|
| Web Menu    | The identity of the menu. This is used to group a related set of menu sections.                 |
| Name        | A name for the menu. This will appear on the tab for the menu.                                  |
| Description | A description of the web menu section. This text will appear on the header bar of the menu tab. |

| Option      | Description                                                                                                                                                               |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Group Id    | Select a group to limit access to the record to operators who are members of that group.<br><br>If no group is specified, any operator will be able to access the record. |
| Modified On | Date the record was last modified.                                                                                                                                        |
| Modified By | Operator who performed the last modification.                                                                                                                             |
| Modifiable  | If selected, any user with the necessary role and group access can modify the record.<br><br>If cleared, only a user with system-level authority can modify the record.   |

## Web Menu Section – Display Page

The Display page contains display settings for the web menu section.

**Figure 7-4.** Web Menu Section – Display Page

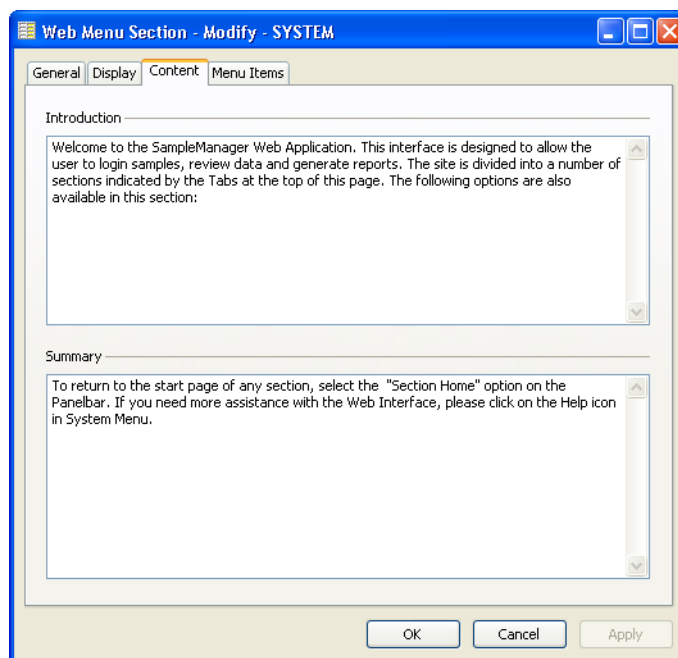


| Option         | Description                                                                                                                                                                                           |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Icon           | The icon is not used on the web site in this release.                                                                                                                                                 |
| Image Link     | <p>Specify the file name and path of the image that should be displayed on the right side of the tab.</p> <p>This field only applies when the default section page is used.</p>                       |
| Order Number   | Enter an integer to define the position of the menu in relation to other menus.                                                                                                                       |
| Visible        | If the Visible check box is selected, the web menu section is visible on the web site.                                                                                                                |
| Show Options   | <p>If the Show Options check box is selected, a list of the menu options will be displayed in the main section of the page.</p> <p>This field only applies when the default section page is used.</p> |
| Always Visible | If the Always Visible check box is selected, the menu will appear on every tab of the web site, but will not have a tab of its own.                                                                   |

## Web Menu Section – Content Page

The Content page defines the text that will appear in the Introduction and Summary sections of the web menu tab on the web site.

**Figure 7-5.** Web Menu Section – Content Page

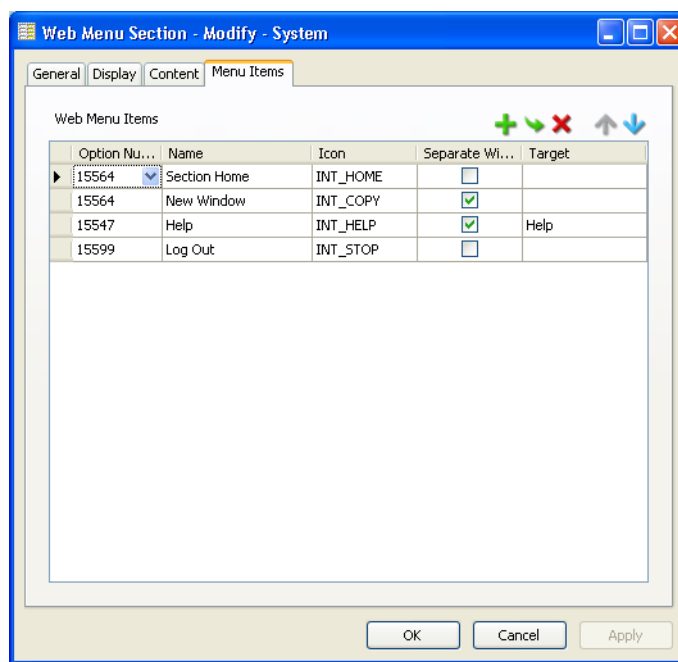


| Option       | Description                                                                                                                                                  |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Introduction | <p>Enter the text that should be displayed in the introduction section of the tab.</p> <p>This field only applies when the default section page is used.</p> |
| Summary      | <p>Enter the text that should be displayed in the summary section of the tab.</p> <p>This field only applies when the default section page is used.</p>      |

## Web Menu Section – Menu Items Page

The Menu Items page defines the items that appear on the menu.

**Figure 7-6.** Web Menu Section – Menu Items Page



| Option          | Description                                                                                                                                         |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| Option Number   | The option number of the item, from the Master Menu table.                                                                                          |
| Name            | The name of the option.                                                                                                                             |
| Icon            | Specify an icon to be displayed beside the name in the web interface.                                                                               |
| Separate Window | If the Separate Window check box is selected, the target page for the option will be displayed in a new browser window when the option is selected. |
| Target          | Enter the name of a frame where the page for the option should be displayed. This field is optional.                                                |

## Adding an InfoMaker Report to the Web Site Menu

### ❖ To add an existing InfoMaker Report to the web site menu

1. Create a master menu option for the report.
2. Open the [Web Menu Section](#) (page 218) for the section of the web site menu that you wish to add the report to.
3. Click the **Menu Items** tab and add the master menu option to the list of items.

## Adding a Link to the Web Site Menu

To add a link to the web site menu you must create a web link record, then create a master menu item for the web link and add the menu item to the required web menu section.

To create a web link choose **Setup > Interface > Web Link > Add** from the SampleManager main menu.

**Figure 7-7.** Web Link Property Sheet

Web Link - Modify - SAMPLE\_LOGIN\_BLANK

General

 SAMPLE\_LOGIN\_BLANK

**Name** Login sample with BLANK template

**Description** Login a Sample using the BLANK Sample Template

**URL** <main.aspx?wizard=samplelogin/page2&template=blank>

**Group Id**

**Modified On** 18/09/2009 13:39

**Modified By** BATCH

**Modifiable** ☒ **Separate Window** ☐

OK Cancel Apply

A web link record defines a link to a URL, which can then be added to the web site menu.

| Option          | Description                                                                                                                                                               |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Name            | A name for the web link.                                                                                                                                                  |
| Description     | A description of the link.                                                                                                                                                |
| URL             | The URL to link to.                                                                                                                                                       |
| Group Id        | Select a group to limit access to the record to operators who are members of that group.<br><br>If no group is specified, any operator will be able to access the record. |
| Modified On     | Date the record was last modified.                                                                                                                                        |
| Modified By     | Operator who performed the last modification.                                                                                                                             |
| Modifiable      | If selected, any user with the necessary role and group access can modify the record.<br><br>If cleared, only a user with system-level authority can modify the record.   |
| Separate Window | If the Separate Window check box is selected, the linked page will be displayed in a new browser window.                                                                  |

❖ **To add the web link to the web site menu**

1. Create a master menu option for the web link.
2. Open the [Web Menu Section](#) (page 218) for the section of the web site menu that you wish to add the link to.
3. Click the **Menu Items** tab and add the master menu option to the list of items.

## LIMSML Entity Action Tester

The LIMSML Entity Action Tester is used to test the building and executing of [LIMSML entity actions](#) (page 9). It will ensure that requests and responses are built correctly and that SampleManager returns the correct data.

The Entity Action Tester is accessed from the Configuration tab of the SampleManager web interface. When the LIMSML hyperlink is selected, the user is directed to an Explorer folder that contains a listing of all currently available Entity Actions. These Actions are taken directly from the LIMS\_ENTITY\_ACTION table in SampleManager.

**Figure 7-8.** LIMSML Entity Action Screen

Test currently available LIMSML Entity Actions.

| Entity           | Action             | Description                                                                                  |                            |
|------------------|--------------------|----------------------------------------------------------------------------------------------|----------------------------|
| ANALYSIS         | GET_COMPONENTS     | Returns a dataset containing all components for analysis                                     | <a href="#">Details...</a> |
| CRITERIA         | GET                | Returns a dataset containing all pre-defined criteria for the specified entity               | <a href="#">Details...</a> |
| EXPLORER_AUX     | EXECUTE            | Executes the Specified Explorer Aux Report                                                   | <a href="#">Details...</a> |
| EXPLORER_AUX     | GET_PARAMETERS     | Returns the Parameter Details of the Specified Explorer Aux Report                           | <a href="#">Details...</a> |
| EXPLORER_AUX     | VALIDATE           | Validates the Specified Explorer Aux Report                                                  | <a href="#">Details...</a> |
| EXPLORER_FOLDER  | GET_COLUMNS        | Returns a dataset containing a page of items for the specified explorer folder               | <a href="#">Details...</a> |
| EXPLORER_FOLDER  | GET_CONTENTS       | Returns a dataset containing a page of items for the specified explorer folder               | <a href="#">Details...</a> |
| EXPLORER_FOLDER  | GET_CONTENT_COUNT  | Returns the total number of items in the specified explorer folder                           | <a href="#">Details...</a> |
| EXPLORER_FOLDER  | GET_RMB            | Returns a dataset containing the RMB options for the Folder                                  | <a href="#">Details...</a> |
| EXPLORER_FOLDER  | SET_COLUMNS        | Allows the new list of columns to be saved                                                   | <a href="#">Details...</a> |
| EXPLORER_RMB     | CHECK_CONTEXT      | Returns two datasets based on the passed in keys, one for in context, one for out of context | <a href="#">Details...</a> |
| GENERIC          | BROWSE             | Returns a dataset of items for a browse                                                      | <a href="#">Details...</a> |
| GENERIC          | BROWSE_COUNT       | Returns the total number of items that can be retrieved for a browse                         | <a href="#">Details...</a> |
| GENERIC          | FAVORITES_ADD      | Adds the specified record to favorites                                                       | <a href="#">Details...</a> |
| GENERIC          | FAVORITES_REMOVE   | Removes the specified record from favorites                                                  | <a href="#">Details...</a> |
| GENERIC          | FIND               | Returns a dataset containing all items and all columns of a type                             | <a href="#">Details...</a> |
| GENERIC          | GET_BY_IDENTITY    | Returns an item given its key0                                                               | <a href="#">Details...</a> |
| GENERIC          | GET_COLUMNS        | Allows all available columns to be selected                                                  | <a href="#">Details...</a> |
| GENERIC          | GET_RELATED_ITEMS  | Returns items related to the specified item using the specified relationship                 | <a href="#">Details...</a> |
| GENERIC          | SET_COLUMNS        | Allows the new list of columns to be saved                                                   | <a href="#">Details...</a> |
| INFOMAKER_REPORT | GENERATE           | Generates the report                                                                         | <a href="#">Details...</a> |
| INFOMAKER_REPORT | GET_PARAMETERS     | Returns the parameter details of the specified report                                        | <a href="#">Details...</a> |
| INFOMAKER_REPORT | VALIDATE_VALUES    | Validates infomaker report parameter values                                                  | <a href="#">Details...</a> |
| JOB              | CREATE_BY_TEMPLATE | Create a job using the provided template                                                     | <a href="#">Details...</a> |
| JOB_TEMPLATE     | GET_FIELDS         | Returns the job template fields                                                              | <a href="#">Details...</a> |

1 2 >

| Column      | Description                                                              |
|-------------|--------------------------------------------------------------------------|
| Entity      | This is the object used to perform an operation.                         |
| Action      | This is the operation to be performed.                                   |
| Description | This field explains the Action.                                          |
| Details     | This link is used to select an item and will display its property sheet. |

By selecting the 'Details' hyperlink of an individual Entity Action, the following property sheet page will be displayed:

**Figure 7-9.** LIMSML Entity Action Property Sheet

**Test INFOMAKER\_REPORT GENERATE command**  
Entity Action information can be seen below

| Entity Action Details |                      |
|-----------------------|----------------------|
| Entity                | INFOMAKER_REPORT     |
| Action                | GENERATE             |
| Description           | Generates the report |
| Modifiable            | Yes                  |
| Modified on           | 10/06/2004 09:03:28  |
| Removeflag            | No                   |
| Return type           | DATA                 |

| Enter Test Parameters                                                                                                                                                                    |                           |                                     |                  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------|-------------------------------------|------------------|
| Test this Entity Action by entering values for all Mandatory and any additional parameters. Use the Test Action button to continue and view the generated LIMSML Requests and Responses. |                           |                                     |                  |
| Parameter                                                                                                                                                                                | Description               | Mandatory                           | Enter Parameter  |
| REPORT_ID                                                                                                                                                                                | Infomaker report identity | <input checked="" type="checkbox"/> | R_ACTIVE_SAMPLES |

Test Action

LIMSML Response   Data Response   LIMSML Request

```

<?xml version="1.0" encoding="utf-8" ?>
<lmsml xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.thermo.com/informatics/xmlns/lmsml/1.0">
  <header />
  <body>
    <transaction>
      <data response_type="data">
        <DataFile>
          <file xmlns="">
            <binary>JVBERi0xLjMKJcfsj6IKNSAwIG9iag08PC9MZWSndGggNIawIF1vRmlsdGVyIC9GbGF0ZURlY29kZT4+CnN0cmVhbQp4nOWdW5Pbth3F3/Up
            <filename>C:\Program Files\LabSystems\SampleManager Server\sm810b2\textreport\lmsml_report76484.pdf</filename>
            <complete>False</complete>
            <delete>True</delete>
          </file>
        </DataFile>
      </data>
    </transaction>
  </body>
</lmsml>

```

| Element | Name            | Description                                                                                                                                                                                                                                                |
|---------|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1       | Details         | This section contains details of the selected Entity Action.                                                                                                                                                                                               |
| 2       | Test Parameters | This section specifies the parameters required to build the LIMSML request. An Action cannot be tested without values for all the mandatory parameters. A mandatory parameter is defined by a <input checked="" type="checkbox"/> in the Mandatory column. |
| 3       | Test Actions    | This section contains the Test Action button which, when pressed will generate a response.                                                                                                                                                                 |

This enables the user to test the selected Action.

In order to test the Action, you must first fill in the parameters that are specified in section 2. When the required fields have been completed, the Action can be processed by clicking the Test Action button. When the Entity Action has been processed, the Display page generates a response, which is displayed in the empty Test Action pane.

The default view will be the LIMSML Response.

The user can choose from 3 view types:

- LIMSML Response displays the LIMSML response received from SampleManager.
- Data Response displays the data response. This option is only available if the Entity Action being tested has a returned a data set, such as the contents of an explorer folder.
- LIMSML Request displays the request sent to the LIMSML.



## Localization Editor

The SampleManager web interface is equipped with a Localization Editor, which allows the user to alter the text that is displayed in the web interface. This alteration can be to the actual text of the message displayed according to the local language settings, or to a language-specific translation.

The Localization Editor enables the user to:

- Customize existing language resources to suit the locale or company policies.
- Add new languages to the system.
- Modify languages that exist on the system.

When the SampleManager web interface is started on a workstation, the language of the text displayed on the login screen will be determined by the browser's language settings.

When the SampleManager web interface is started on a workstation, the i18n localization file is checked for the default language to use. After a user has logged into SampleManager the language is set to be that defined by the WEB\_LANGUAGE configuration item.

If the language support has been modified and SampleManager is upgraded, changes will not be overwritten by the upgrade. The SampleManager installation process detects changes to the language resources and leaves them intact.

### Contents

- [Definitions](#)
- [Using the Localization Editor](#)
- [Adding a New Language](#)
- [Adding a Message](#)
- [Adding a Message Group](#)
- [Editing Messages](#)

## Definitions

The following definitions apply within the Localization Editor:

| Term          | Description                                                                                                                                                                       |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Message       | A specific text item that appears on the screen. Each message has an identity and one translated value for each language currently configured in the SampleManager web interface. |
| Language      | A specific language into which messages are translated, for example French or German. Each message in the interface can be translated into multiple languages.                    |
| Message Group | A category used to classify messages within the Localization Editor. For example, the Login message group is used to group the messages that appear on the Login screen.          |

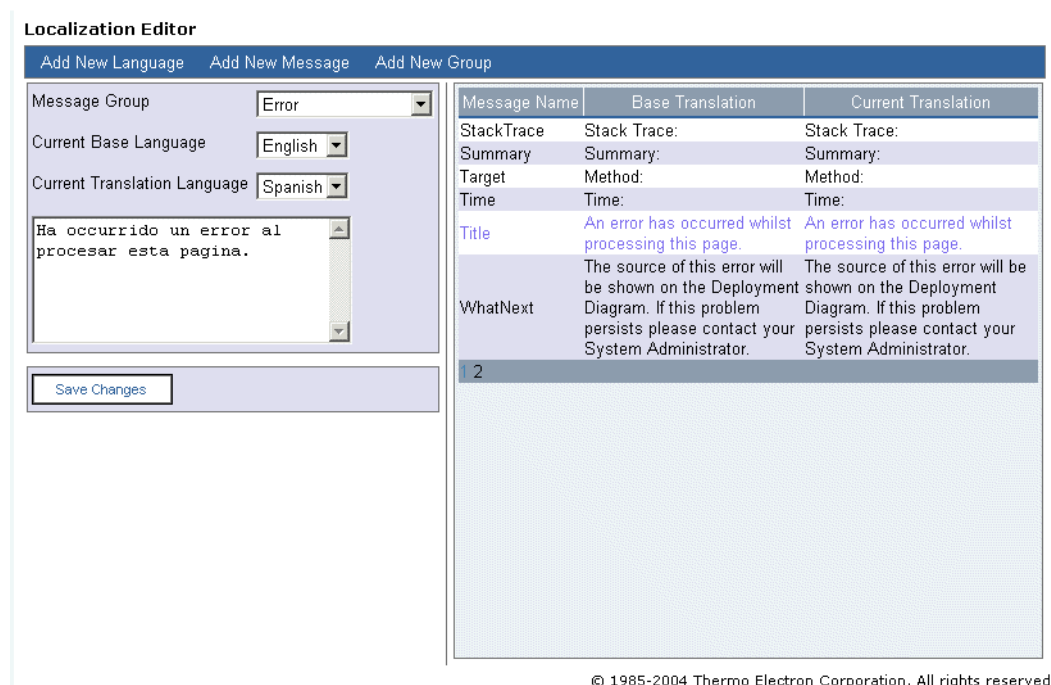
## Using the Localization Editor

The Localization Editor allows the user to:

- Add a new language
- Add a new message
- Add a new message group
- Edit existing messages

To use the Localization Editor, it must be opened by clicking the Configuration tab and then selecting the Localization Editor link.

**Figure 8-1.** Localization Editor



| Message Name | Base Translation                                                                                                                     | Current Translation                                                                                                                  |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| StackTrace   | Stack Trace:                                                                                                                         | Stack Trace:                                                                                                                         |
| Summary      | Summary:                                                                                                                             | Summary:                                                                                                                             |
| Target       | Method:                                                                                                                              | Method:                                                                                                                              |
| Time         | Time:                                                                                                                                | Time:                                                                                                                                |
| Title        | An error has occurred whilst processing this page.                                                                                   | An error has occurred whilst processing this page.                                                                                   |
| WhatNext     | The source of this error will be shown on the Deployment Diagram. If this problem persists please contact your System Administrator. | The source of this error will be shown on the Deployment Diagram. If this problem persists please contact your System Administrator. |

The Localization Editor has a menu bar, immediately below the title. This contains links to screens that allow you to:

- Add a new language
- Add a new message
- Add a new message group

The main Localization Editor screen consists of the following areas:

| Option                       | Description                                                                                |
|------------------------------|--------------------------------------------------------------------------------------------|
| Message Group                | A list of all message groups held on the system.                                           |
| Current Base Language        | A list of all languages held on the system. The default is English.                        |
| Current Translation Language | A list of all languages held on the system                                                 |
| Message Name                 | The name that has been given to any particular message.                                    |
| Base Translation             | This column contains the full message in the default language, in this instance, English.  |
| Current Translation          | The full message text in the language selected from the Current Translation Language list. |
| Save Changes                 | Click this button to save any changes made during the session.                             |

The number of pages is displayed below the last message on the page. To navigate to a specific page, click on the appropriate page number.

## Adding a New Language

### ❖ To add a new language

1. Choose **Add New Language** from the menu bar on the Localization Editor main page.

The Add Language page opens.

**Figure 8-2.** Add Language Page

Localization Editor - Add Language

Return to Localization Editor

New Language

Add Language

Current Languages

- English
- French
- German
- Spanish

2. In the New Language field, type in a name for the new language.
3. Click the **Add Language** button.

The language appears in the Current Languages pane.

## Adding a Message

### ❖ To add a new message

1. Choose **Add New Message** from the menu bar on the Localization Editor main page. The Add Message page opens.

**Figure 8-3.** Add Message Page

Localization Editor - Add Message

Return to Localization Editor

Message Group

New Message

Add Message

Current Messages

- AddAnalysis
- AddSchedule
- Analysis
- Back
- BadJobName
- BadTemplate
- ChooseTemplateDesc
- ChooseTemplateTitle
- Finish
- Forward

2. In the Message Group list, select the message group that the new message will belong to.

3. In the New Message field, type a name for the new message.
4. Click the **Add Message** button.

When a new message is created it has a name, but contains no base or translated text. For further information on how to edit a message, please see the [Editing Messages](#) (page 235) section of this chapter.

## Adding a Message Group

### ❖ To add a new message group

1. Choose **Add New Group** from the menu bar on the Localization Editor main page. The Add Group page is displayed.

**Figure 8-4.** Add Group Page

Localization Editor - Add Group

[Return to Localization Editor](#)

New Group Name

Add a new Message to this Group

**Current Message Groups**

- Browse
- ElectronicSignature
- EntityAction
- Error
- Explorer
- General
- Job
- JobLogin
- Localization
- Login

2. Type in a name for the new group.
3. A new message to be held within the new group must now be added. At this point, only the message name is created. The base and translation text must be entered using the main page of the Localization Editor.
4. Click the **Add Group** button.

## Editing Messages

The Localization Editor allows the customization of languages within the Message Editor. To customize a language, the messages must be individually edited.

### ❖ To edit a message

1. From the main page of the Localization Editor, select the Message Group that contains the message that is to be edited.
2. Select the Current Base Language of the message from the drop down list.
3. From the Current Translation Language drop down list, select the language into which the message is to be translated.
4. In the right pane, click anywhere on the text of the message to be edited. The message is displayed in the left pane.

**Figure 8-5.** Message Editor

**Localization Editor**

Add New Language   Add New Message   Add New Group

Message Group: Error

Current Base Language: English

Current Translation Language: Spanish

Ha ocurrido un error al procesar esta pagina.

[Save Changes](#)

| Message Name | Base Translation                                                                                                                     | Current Translation                                                                                                                  |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| StackTrace   | Stack Trace:                                                                                                                         | Stack Trace:                                                                                                                         |
| Summary      | Summary:                                                                                                                             | Summary:                                                                                                                             |
| Target       | Method:                                                                                                                              | Method:                                                                                                                              |
| Time         | Time:                                                                                                                                | Time:                                                                                                                                |
| Title        | An error has occurred whilst processing this page.                                                                                   | An error has occurred whilst processing this page.                                                                                   |
| WhatNext     | The source of this error will be shown on the Deployment Diagram. If this problem persists please contact your System Administrator. | The source of this error will be shown on the Deployment Diagram. If this problem persists please contact your System Administrator. |

1 2

5. Replace the text in the text box with the translated text and click the **Save Changes** button.



## Electronic Signatures

This chapter describes the electronic signature (ESig) functionality implemented in the SampleManager web interface. This has been designed to meet the requirements of the FDA 21 CFR Part 11: Electronic Records and Signatures rule.

Esigs can be configured for the following functions in the SampleManager web interface:

- Accessing menu options: The user will be required to enter an electronic signature when accessing a particular menu option in the SampleManager web interface.
- Saving data: The user will be required to enter an electronic signature when saving data to the LIMS from the SampleManager web interface.

This chapter describes how electronic signatures affect the SampleManager web interface.

### Contents

- [Configuring Electronic Signatures](#)
- [Esigs for Web Interface Menu Options](#)
- [Esigs for New Data from the Web Interface](#)
- [Failed Esigs](#)
- [Viewing ESig Information](#)

## Configuring Electronic Signatures

Electronic signatures are configured within the standard SampleManager interface. For full details of how to setup electronic signatures, refer to the *Electronic Signatures* chapter of the *SampleManager System Administration Guide*. The signature configuration controls:

- The trigger for the signature, for example attempting to access a specific menu option.
- The fields that will be used to calculate the signature checksum.

The web interface can support the following types of signature configuration:

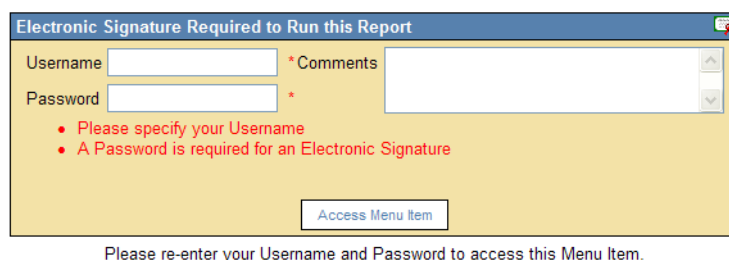
- "Before" option for accessing menu items: The user is prompted for their signature before the option is executed.
- "On Commit" option for saving data: The user is prompted for their signature before data is saved to the database.

**Note** The web interface does not support the "After" or "Before and After" options for Esigs on menu items.

## Esigs for Web Interface Menu Options

If an Esig level of 'Before' is assigned to a menu option, users will be prompted for an electronic signature before the menu item is executed, as shown in the following example:

**Figure 9-1.** Example Esig Prompt for Menu Action



Electronic Signature Required to Run this Report

Username  \* Comments

Password  \*

- Please specify your Username
- A Password is required for an Electronic Signature

Access Menu Item

Please re-enter your Username and Password to access this Menu Item.

In this example, SampleManager has been configured to prompt for an electronic signature when the user runs the Job Header report. If the same report is requested from within SampleManager (SMW), the following prompt is displayed:

**Figure 9-2.** Example Esig Prompt in SampleManager Client

In both examples, when entered by the user, the Username and Password are validated by the server. If the supplied information is valid, the user is permitted to perform the selected function.

## Esigs for New Data from the Web Interface

If an Esig level of 'On Commit' is assigned to an entity, users will be prompted for an electronic signature before the data is saved to the database, as shown in the following example:

**Figure 9-3.** Example Esig Prompt for New Data

## Failed Esigs

An Esig failure will be recorded if a user is prompted for an electronic signature but fails to enter a valid username and password combination after three attempts. When an Esig failure occurs, the user is logged out of the web interface. A failure record for the Esig attempt is created in the SampleManager database.

## Viewing ESig Information

Information relating to all successful and failed Esig attempts can be viewed in SampleManager from the Signature Viewer. For more information about the Signature Viewer, see the *Electronic Signatures* chapter in the *SampleManager LIMS System Administration Guide*.

# Index

## <

- <action> 178
- <actions> 177
- <body> 172
- <children> 182
- <code> 193
- <command> 179
- <comments> 189
- <data> 183
- <description> 192
- <entity> 175
- <errors> 189
- <field> 181
- <fields> 180
- <header> 170
- <limsml> 169
- <parameter> 171
- <severity> 194
- <source> 195
- <summary> 191
- <system> 174
- <transaction> 173

## A

- adding a link to the Web Site menu 223
- adding a message 233
- adding a message group 234
- adding an Explorer folder 216
- adding an InfoMaker report 223
- analysis get components LIMSML entity action 45
- architecture 1
- ASP.NET 202

## C

- clientaddress 184, 188
- clientname 188
- comments 184, 187
- configuration settings 208
- criteria get LIMSML entity action 64
- customizing styles
  - default style – blue 213
  - Default Style – Brown 211
  - default style – green 212
  - panelbar (pbx) 213

## E

- editing messages 235
- electronic signatures 237
  - configuring 238
  - failed esigs 239
  - for menu options 238
  - for new data 239
- element and attribute definitions 169
- enabling ASP.NET 202
- error 189, 190, 191, 192, 193, 194, 195
- Error Handler component 2
- explorer aux execute LIMSML entity action 67
- explorer aux get parameters LIMSML entity action 68
- explorer aux validate LIMSML entity action 69
- Explorer folder 216
- explorer folder get columns LIMSML entity action 70
- explorer folder get content count LIMSML entity action 71
- explorer folder get rmb LIMSML entity action 78
- explorer folder set columns LIMSML entity action 79
- explorer get contents LIMSML entity action 72
- explorer rmb check context LIMSML entity action 79

**G**

generic browse count LIMSML entity action 87  
 generic browse LIMSML entity action 80  
 generic create LIMSML entity action 89  
 generic favorites add LIMSML entity action 91  
 generic favorites remove LIMSML entity action 92  
 generic find LIMSML entity action 94  
 generic get by identity LIMSML entity action 97  
 generic get columns LIMSML entity action 99  
 generic get related items LIMSML entity action 102  
 generic search count LIMSML entity action 113  
 generic search LIMSML entity action 108  
 generic set columns LIMSML entity action 117  
 generic update LIMSML entity action 118

**I**

InfoMaker report 223  
 infomaker report generate LIMSML entity action 50  
 infomaker report get parameters LIMSML entity action 51  
 infomaker report validate values LIMSML entity action 51  
 installation  
   ASP.NET 200  
   localization editor 204  
   web application 200  
   web site 204  
 installation steps 198  
 internationalization editor 231  
 Internationalization Helper component 2

**J**

job create by template LIMSML entity action 17  
 job template get fields LIMSML entity action 52  
 job template validate values LIMSML entity action 55

**L**

languages 233  
 LIMSML 167  
   hierarchy 168  
 LIMSML Entity Action 10  
   analysis get components 45  
   criteria get 64  
   explorer aux execute 67  
   explorer aux get parameters 68  
   explorer aux validate 69  
   explorer folder get columns 70

explorer folder get content count 71  
 explorer folder get rmb 78  
 explorer folder set columns 79  
 explorer get contents 72  
 explorer rmb check context 79  
 generic browse 80  
 generic browse count 87  
 generic create 89  
 generic favorites add 91  
 generic favorites remove 92  
 generic find 94  
 generic get by identity 97  
 generic get columns 99  
 generic get related items 102  
 generic search 108  
 generic search count 113  
 generic set columns 117  
 generic update 118  
 infomaker report generate 50  
 infomaker report get parameters 51  
 infomaker report validate values 51  
 job create by template 17  
 job template get fields 52  
 job template validate values 55  
 limsml entity actionget by identity 120  
 limsml entity actionget by relationship 120  
 result get audits 19  
 sample add test 21  
 sample create by template 23  
 sample get component limits 28  
 sample get results 30  
 sample get sample re prompt 33  
 sample re put results 38  
 sample result entry 39, 133, See result entry  
 sample template get fields 56  
 sample template validate values 59  
 system get file 121  
 system ping 121  
 system sign failure 123  
 system sign menu item 123  
 test schedule get contents 61  
 user interface get fields 124  
 user interface validate values 129  
 user login 131  
 user logout 131  
 LIMSML Entity Action Tester 225  
 limsml entity actionget by identity LIMSML entity action 120  
 limsml entity actionget by relationship LIMSML entity action 120  
 LIMSML Helper component 2  
 link to the Web Site menu 223

localization 233  
 localization editor 229  
   definitions 230  
   using 231

## M

menuproc 187  
 message group 234  
 messages  
   adding 233  
   editing 235  
 minimum system requirements 198

## P

panelbar (pbx) 213

## R

reason 184, 186  
 Remote Services  
   testing 3  
 result entry 133, 150  
   ad-hoc results 155  
   callback routines  
     limsml\_result\_entry\_get\_analysis 141  
     limsml\_result\_entry\_get\_sample 136  
     limsml\_result\_entry\_get\_test 147  
     limsml\_result\_entry\_get\_test\_rep 145  
     limsml\_result\_entry\_post\_sample\_update 166  
     limsml\_result\_entry\_post\_test\_update 166  
     limsml\_result\_entry\_valid\_result 151  
     limsml\_result\_entry\_valid\_result\_error 154  
     limsml\_result\_entry\_valid\_result\_field 150  
   completing samples 164  
   completing tests 162  
   creating samples 137  
   creating tests 148  
   errors 152  
     general errors 153  
     handling with VGL 154  
     ignoring 153  
     mapping errors 152  
   missing samples 139  
   missing tests 149  
   recalculation 157  
   result fields 150  
     entered by field 156  
     entered on field 156  
   sample identification  
     using id numeric 134

    using id text 134  
     using test number 135  
     using VGL 136  
 sample login 137  
 test identification  
   using analysis 141  
   using test number 140  
   using VGL 141, 145, 147  
 test replicate 142  
   incrementing 146  
   using test count 144  
   using VGL 145  
 updating sample fields 160  
 updating test fields 161  
 validating results 151  
 result get audits LIMSML entity action 19

## S

sample add test LIMSML entity action 21  
 sample create by template LIMSML entity action 23  
 sample get component limits LIMSML entity action 28  
 sample get results LIMSML entity action 30  
 sample get sample re prompt LIMSML entity action 33  
 sample re put results LIMSML entity action 38  
 sample result entry LIMSML entity action 39  
 sample template get fields LIMSML entity action 56  
 sample template validate values LIMSML entity action 59  
 SampleManager web site 207  
 signature 184  
 SMP Business component 167  
 styles 209  
   customizing 210  
 system get file LIMSML entity action 121  
 system ping LIMSML entity action 121  
 system requirements  
   minimum 198  
 system sign failure LIMSML entity action 123  
 system sign menu item LIMSML entity action 123

## T

test schedule get contents LIMSML entity action 61  
 testing the Remote Services 3

## U

user interface get fields LIMSML entity action 124  
 user interface validate values LIMSML entity action 129  
 user login LIMSML entity action 131  
 user logout LIMSML entity action 131

## Index

username [184](#), [185](#)

## W

web application component [2](#)

web interface architecture [1](#)

web interface menu options [215](#)

web link [223](#)

web menu [218](#)