

SQL*LIMS 4.0 VALIDATION



Vahe Karamian

This Cooperative Education Report has been submitted to Nazeli Dertsakian and has been approved for release and submission to Dr. Fred Bet-Pera, Cooperative Education Director, Cal Poly, Pomona during the 9th week of the quarter.

Signed: _____ Date: _____

Table of Contents

Preface	4
Acronyms	5
Definitions	6
Objectives	7
Execution Summary.....	8
Expanded Description of the Co-op position and activities, detailed summary and conclusion.....	10
Sample Logging	10
Work Assignment.....	10
Results Entry	10
Calculations Results Forms	10
I Detection Limits	12
II New Spec and Specification	13
Multiple Specification Limits.....	13
Edit Instance.....	14
Adding Attributes and Parameters to Instance Data.....	14
Approve Data.....	14
Edit Condition	14
Worklists.....	14
Result Formats	15
Significant Figures	15
Events	16
Sample Location	16
General Sample Receipt	16
Userstamp and Timestamp	17
Refire Status Actions	17
Auditing and Versioning	17
Attachments	18
Electronic Signature	18
Lot Plans	18
Lot Create	20
Lot Control (Edit Lot).....	21
Lot Release.....	21
Lot Difficulties	22
Lot Monitor	22
Experimental Work and Data	24
Sample Logging	24
Result Entry	25
Detection Limits	26
Approving Data.....	26
Worklists.....	27
Auditing and Versioning	29
Events	29
Suggestion for further work, new objectives, and action required.....	31
Personal Comments, Opinions, and Observations	32

Index.....	33
Figures	34
Tables	34

Preface

The SQL*LIMS application is a powerful laboratory information management system designed for use in the laboratory environment. The data storage and retrieval system enables the user to:

- ? Store laboratory data about samples, the tests performed on them, and the results
- ? Retrieve these data quickly and easily
- ? Track the progress of samples from the time they are logged into the laboratory until testing is complete

The functions in the SQL*LIMS application have to do with the lifecycles of samples in the lab:

- ? Logging (including assigning tests)
- ? Assigning work
- ? Entering results
- ? Approving results
- ? Tracking (the state of the sample)

Key features:

- ? Quickly identifies bottlenecks and problems by using SQL*LIMS® software to monitor and track samples and results throughout the laboratory
- ? Manages quality and process control through seamlessly integrated, specialized modules for Stability Management and QA Manufacturing
- ? Supports the laboratory's compliance with the FDA's 21 CFR 11 Regulations with dynamic security and regulatory compliance features
- ? Maps the laboratory's workflow and custom business rules with simple method definition screens
- ? Saves time by quickly retrieving critical information and generating reports from a single program
- ? Uses a proven enterprise laboratory information management system installed and supported in over 1,000 laboratories worldwide

Acronyms

SOP	Standard Operating Procedure
LIMS	Laboratory Information Management System
TPL	Template Procedural Language
SQL	Structured Query Language
GDP	Good Documentation Practice
FDA	Food and Drug Administration

Definitions

- Datagroups:** SQL*LIMS software Datagroups control and individual's access to data. A user may belong to one or more datagroup and can only see data associated with their datagroup(s). In addition, a special customer datagroup is used to control access to data by outside sources. A customer datagroup requires you to define what condition the data is in before access is given, i.e., approved. Users can be moved in and out of datagroups at any time.
- Jobtype:** Jobtypes control what a user can do with their data. Subset of SQL*LIMS software functionality can be created under a jobtype. Users are then assigned a jobtype. Functions that are not in the subset do not show up on the user's menu. Users can be moved in and out of jobtypes at any time.
- Audit Groups and Audit Trail:** Audit trail is the process of recording what, when, and why changes were made, and who made those changes. SQL*LIMS software can track the time when changes were made. Audit trail is configurable through systems rules and audit groups. The users audit group determines the level of auditing.
- Client Timeout:** To ensure that unattended SQL*LIMS software clients do not remain accessible, a client timeout is available. When a client has been inactive for the defined time period, the client is automatically logged out.
- Electronic Signature:** The ability to conform to the FDA's Electronic Records Rule, 21 CFR 11, requires the implementation of all of the security listed above plus electronic signature. Electronic signature requires that the user is authenticates and that authentication is recorded when certain events occur in the system. SQL*LIMS software provides a modular electronic signature option that works with any form of signatures, including biometrics. Currently SQL*LIMS provides a password verification. When users encounter a signature event they must supply their username and password for the data to be saved. A record of the users full name, the time (client and server), and links to the data saved is stored in the electronics records table. Electronic signatures can be viewed through forms.

Objectives

The objective of SQL*LIMS 4.0 validation is to validate a computer system which will be used in a highly sophisticated setup at BAXTER BioScience. Such a system is necessary today in the pharmaceutical industry because of the vast volume of data generated on a day-to-day basis in the laboratories.

The system needs a way to keep track of this data, and it needs a way to make sure the data stored is accurate, error free, and complete. Overall the system needs to make sure that the data that is being entered into the system is secure and accurate. The data entered in LIMS authenticates the quality of the products (plasma derived medication) made at Baxter.

Test procedures have been designed to validate the most utilized SQL*LIMS forms. In the validation process, the user is responsible for verifying whether each section passed or failed and the outcome of each procedure at the end of each section. A section in the validation process has been provided for the validator to record remarks as necessary, as well as a comments section for additional remarks.

Execution Summary

SQL*LIMS 4.0 validation consisted of two test cases. First test case was the SQL*LIMS 4.0 Core module, and the second test case was SQL*LIMS 4.0 QA module. Because of the nature of the test case design, SQL*LIMS 4.0 Core module had to be tested prior to the QA package.

A testing account, TESTLIMS, was created and used to execute the procedures. The account was created during the general setup. The sections have been executed in the order specified, as there are some interdependencies between certain sections.

Validation started with the Core module and progressed into the QA. During the Core validation several sections have not been executed due to technical difficulties, they have been addressed and if the difficulty was minor they have been resolved and execution proceeded. However, if the problem was major, a fix request was ordered, and when the problem was addressed a fix patch was installed to overcome the problem.

During testing, we not only looked for errors specific to the execution of the test procedures, but also for any anomalies encountered during the course of testing. Any bug found was treated per internal SOPs and reported appropriately.

The following table shows independent and dependant test cases:

Independent Test Cases	Dependant Test Cases
<ul style="list-style-type: none"> ? Calculations In Result Forms ? Limits—Detection, New Spec, Specification ? Multiple Specification Comparison ? Adding Attributes and Parameters to Instance Records ? Edit Condition ? Significant Figures ? Events ? Locations ? Sample receipt ? Userstamp/Timestamp ? Refire Status Action ? Client Timeout ? Auditing and Versioning ? Attachments 	<ul style="list-style-type: none"> ? General Setup ? Sample Logging ? Work Assignment ? Results Entry ? Edit instance ? Approve Data ? Worklists ? Result Formats

The table indicates that most procedures are independent of each other. However, one must be careful before starting test execution by starting with this information and

reviewing the test sections to determine the optimum execution order based on minimizing the impact on other tests when a particular test section is executed.

The test cases were performed in the following order:

(1) General Setup, (2) Sample Logging, (3) Work Assignment, (4) Results Entry,(5) Calculations In result Forms, (6) Limits—Detection, New Spec, Specification, (7) Multiple Specification Comparison, (8) Edit Instance, (9) Adding Attributes and Parameters to Instance Records, (10) Approve Data, (11) Edit Condition, (12) Worklists, (13) Result Formats, (14) Significant Figures, (15) Events, (16) Location, (17) Sample Receipt, (18) Userstamp/Timestamp, (19) Refire Status Actions, (20) Client Timeout, (21) Auditing and Versioning, and (22) Attachments.

For testing the SQL*LIMS 4.0 QA module, a general setup was performed using the TPL files to populate the database with information required for tests included in the validation.

The SQL*LIMS 4.0 QA module was tested in the following order:

(1) Lot Plant, logging Mode 1, Mode 2, Mode 3, and Mixed Logging Mode, (2) Lot Creation, (3) Lot Control (Edit Lot), (4) Lot Release, (5) Lot Difficulties, and (6) Lot Monitor.

Expanded Description of the Co-op position and activities, detailed summary and conclusion

Sample Logging

The forms that are used for creating instance records have been validated, including logging by: Log Plan, Sample Plan, Material, Study and Grid.

In addition, the forms used to add or modify existing records have been validated, including: Add by Sample Plan, Add by Material, Add by Profile, Edit Attributes and Parameters, and Create Child Samples.

Work Assignment

The procedures validated the process used to designate tasks to a specific analyst via the Assign Work form. The tasks are assigned in a variety of ways, including:

1. Tasks that have not been assigned are assigned to an analyst.
2. A task that has been assigned is reassigned to another analyst.
3. Tasks that have not been assigned are assigned to a valid workgroup. Also validated is the View Work form, which can be used to view the tasks, assigned to an analyst.

Results Entry

The test procedures validate the results entry forms, including entry by: Sample, Task, Method, and Grid.

The individual results entered have been verified versus the assigned measure limits, when applicable. Results with modifiers have been entered as well.

Calculations Results Forms

The operators available for use in defining calculations were used to create a series of templates to test each of these functions. The components are either associated with a result plan or an operation. The operators include:

Table 1

+	Addition
-	Subtraction
*	Multiplication
/	Division
^	Exponentiation
~	Negation

ABS	Absolute Value
SQRT	Square Root
LN	Natural Log
LOG	Common Log
EXP	Inverse Natural Log
POW	Inverse Common Log ($POW(x) = 10^x$)
RND	Rounding
TRC	Truncation
CEIL	Ceiling
FLOOR	Floor
SIN	Sine Function
COS	Cosine Function
TAN	Tangent Function
ASIN	Inverse Sine Function
ACOS	Inverse Cosine Function
ATAN	Inverse Tangent Function
CNT, COUNT	Count
SUM	Sum Function
AVG, MEAN	Average
MEANDEV	Mean Deviation
STD, STDN, STDDEV	Population Standard Deviation
STD—1, STDNM	Sample Standard Deviation
VAR, VARIANCE	Variance
MEDIAN	Median
MIN	Min
MAX	Max
Range	Range
DIV	Integer Division
MOD	Integer Modulus
RNDA	ASTM Rounding
FETCH, Get_PARAM	Retrieval of parameters at the submission, sample and task levels
GET_ATTR	Retrieval of attribute values at the submission, sample, task and result levels
%	Percent
Constants	Pi, e and Avagador's Number
MODE	Mode
RAND	Random Number

Each operation or result plan is made up of many components, of which the first several require manual result entry. The results for the remaining components are defined as calculations, which will be completed automatically upon committing the manually entered results. The only exceptions to this rule are the ones defined for testing the FETCH and GET_ATTR functions. In these cases, the first 10 components are calculations that fetch the attribute or parameter values from the correct instance level. The next five components are calculations, which reference the 10 previous calculated results. The Recalculate option is used to perform the calculations.

Additional test sections cover multi-step calculations, where a single calculated result is performed using multiple calculation attributes, complex nested functions, FETCH and GET_ATTR from a task hierarchy, and user defined calculation functions (PL/SQL functions). Also, a test of the verify function that is available from template definition was performed.

I Detection Limits

Each of four detection limits are assigned to various template objects used to log the samples. The assignment of the DETECTION LIMIT keyword were made of two studies, one material and one method plan to verify the enforcement of the limits at differing levels within the instance hierarchy. Also, precedence of attributes over parameters were covered.

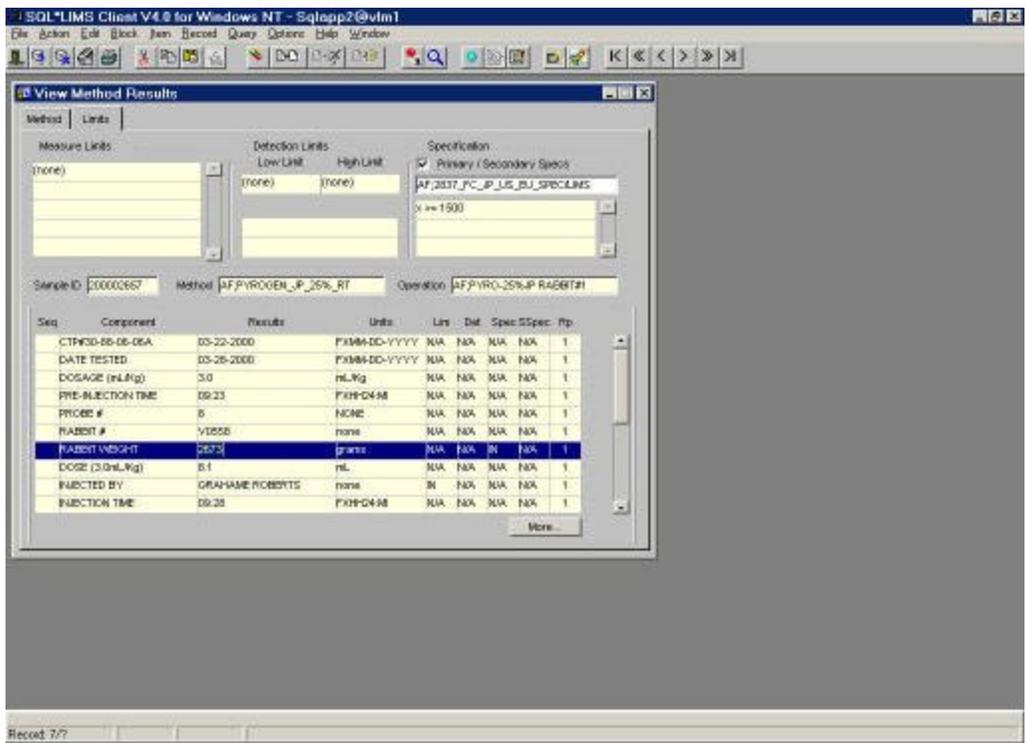


Figure 1 Detection Limits

II New Spec and Specification

A similar approach is taken for these reserved keywords. In this case, the samples were logged with the appropriate keyword attached to the studies and materials. Two samples are used to test each keyword, as follows:

Sample 1 :

Keyword is attached to the study both as an attribute and a parameter. The specification attached as an attribute will supersede that which is attached as a parameter.

Sample 2:

Keyword attached to both the study and the material. The specification attached to the material will supersede that which is attached to the study.

Each sample was assigned three tasks. The predefined result values were entered for each of these tasks as follows:

1. Results entered within the limits.
2. Results entered outside of the limits.
3. Results entered within and outside of the limits.

The specification field was updated as results were entered and the status was then compared to expected values. The conditions remain ONLINE for these samples, as they were used in later sections.

Multiple Specification Limits

The ability to add primary and multiple secondary specifications to template objects and to have limit comparisons applied to results were addressed at many levels.

Multiple specifications were assigned to the material and the sample plan. Limit comparisons are verified for samples logged under a variety of logging methods and results entered using a variety of forms. Limit comparison for changed results was also verified.

User determinations of which limits are displayed were validated as well as the related In Limit value.

This multiple specification testing was also applied to calculated results, replicates and multiple versions of results.

Edit Instance

A variety of forms were accessed in order to modify fields at the submission, sample and task levels. After modification, these records were queried again to verify the modifications.

Another aspect of instance editing involves the creation of task replicates and task versions. Each of these is examined by first creating either a new version or a new replicate of the task, followed by results entry. After the results have been entered, View forms were used to verify either the version and or replicate number value assigned to the result records. The status and condition of the individual task version are also verified.

Adding Attributes and Parameters to Instance Data

The ability to add attributes and parameters to existing instance records was tested here using the option Add Attribute from various forms. Parameters were added to submissions, samples and tasks. Attributes can be added to these levels as well plus the result level.

Approve Data

The approval process was performed in three modes:

1. Manual approval.
2. Automatic approval.
3. Automatic approval of the submission, sample and task, after manual approval of results.

Edit Condition

Unrelated instance records were selected to illustrate how changes in condition are propagated downward through the instance hierarchy. The condition value at the submission, sample and task level was modified. The changes in condition are verified via the View State History option as well as via the use of View forms. The assignment of the level at which the change was effected is also confirmed.

Worklists

The primary functions of Worklists have been examined, including: Worklist Creation, Worklist Editing, Worklist Viewing, Worklist Results Entry, and Worklist Approval.

A number of QC types have been defined either with an expiration date, with no expiration date or with a date denoting that the QC type has expired.

Worklist plans have been created to utilize all of the QC types defined. These include a plan that fails to generate a Worklist since it includes the expired QC type. In addition to

using the worklist plans, a Worklist will be created without the aid of any plan. A quantity of unknowns were specified and the resulting worklist reflects this value.

Result Formats

The ability for results to be constrained or presented using an Oracle number of date format mask was tested. All result entry forms were tested along with the change result forms. The following masks have been tested:

Number Formats

Table 2

Mask	Function
999	Constrain result entry to three digits maximum without decimal.
099	As in 999, except must always be 3 digits.
S999	As in 999, except presented with the sign of the number before the entered value.
9,999	As in 999, except when 4 digits are used a comma must be entered.
99.99	Constraint result entry to two digits maximum before and after decimal point. Presents result with trailing zeros after decimal point.
90.99	As in 99.99, except a leading zero must be entered before the decimal point.

Date Formats

Table 3

Mask	Function
FM-DD-MM-FM-FXYYYY	Display 4 digit year (FX) and single digit (if applicable) day and month (FM). The second FM toggles off its behavior.
DD/MM/FXYYYY	Day and month with leading zeros. Use slash for delimiter.
HH24:MI	Hours and minutes only (24 hour clock).
DD-FMRMFM-YYYY	Use roman numeral for month. FM is used to strip extra spaces around roman numeral.
HH12:MI:SS p.m., SSSSS	Hours (12 hour clock), minutes, seconds followed by a.m. or p.m. and seconds past midnight.

In addition, two long date formats are used to present the date spelled out like, “Friday, June 9, 2000 a.d. 12:12:12: p.m.”.

Significant Figures

The significant figures feature is implemented in two places in SQL*LIMS, at result entry and during evaluation of specification limits. The rounding that occurs during result entry affects the presentation of the results. The rounding that occurs during specification evaluation is internal and is not visible during result entry, except as it affects whether a limit is IN or OUT. There are two rounding methods available USP and ASTM.

The testing was designed to cover the following aspects that are related to result entry and specification evaluation as it relates to rounding to significant figures:

USP vs. ASTM

- ? 4, 2 and no significant figures at result entry paired against 4 significant figures on specifications.
- ? Positive and negative numbers
- ? Large, intermediate and small magnitude numbers
- ? Result modifiers
- ? Calculations
- ? Primary and secondary specifications

The testing has been completed using all result entry forms and change result forms.

Events

The primary functions related to events have been examined including installation, installation as an option, running from the menu, and running as an option. Events have been executed either interactively on the client or remotely on the server. The types of events tested are Oracle Reports, Programs, and PL/SQL.

Sample Location

Samples created with the VALID 01 template objects were assigned a default location at the point of sample logging. The initial location was assigned via the Material used to log the samples. The samples' initial location was verified. One sample's location was then be modified and the location history verified. The sample utility was ultimately disposed of and the history was verified again. The creation of locations via the proper form was also exercised during the setup.

General Sample Receipt

The general sample receipt form was tested using a set of samples that have attributes and parameters as follows: Date Received as an attribute, Date Received as a parameter, no Date Received attributes or parameter, Date Received as an attribute and a parameter.

The ability to create the attribute during receipt was tested. Also, the processing of required attributes and parameters to take an INCOMPLETE sample to ONLINE as a result of sample receipt was tested. Receiving all queried samples versus receiving only selected samples was covered as well.

Operation have been tested by adding sample attributes along with a create attribute switch both on and off and placing samples into a sample location.

Userstamp and Timestamp

The additions of userstamp and timestamp to result records upon creation and/or modification of those records have been examined via the Result Detail event. Result records have been created, have a value entered, and have a value changed, approved and changed again. New task version will be created as well as replicated added. At each step the Result Detail report will be run to document the userstamp and timestamp associated with the record.

Refire Status Actions

The ability to specify that a status action was re-fired when an instance record again reached the appropriate state was verified. A sample will be logged; results will be entered, approved, changed and re-approved to verify the status action fire. The ability to add status actions to an instance object was also verified.

Auditing and Versioning

Template auditing and version control were setup using an audit configuration and an audit map for various template objects. The audit map was used to associate auditing and versioning properties with a specific template object field. For example, if you wanted to audit a material when its datagroup changed and forced a new generation, when this occurred, map settings will enable this behavior. The various map settings are:

Table 4

<u>Setting</u>	<u>Function</u>
Create Generation	Increment the template object's generation column. Without this set, the template object's version column will increment.
Free Entry	Control whether audit reasons are constrained to be from a predetermined list of reasons.
Silent Audit	Causes an audit record to be recorded when changes are made, but without user intervention
Audit = YES	Enables auditing. Without this or silent audit no auditing occurs, but version control is still functional.
Create Master Generation	For detail records, force the master records generation to change when a detail record is added or deleted. This doesn't apply to modification of detail records.

Versioning of attributes on templates and auditing and versioning of attributes on instance records have been verified also.

Attachments

Attachments were added to most template objects and all of the instance records. Validation of attachments have been done by adding attachments to each of the appropriate records and then viewing the attachments with the Browse Attachments form. Most often, this was accessed through clicking on a icon placed on the form next to the appropriate record.

Attachments can be either plain text or binary files. Binary files can be stored in the database or in the server's file system under control of the database.

For study, material, sample plan, method and operation, attachments can be template, copy or reference types. Template attachments are associated with the template record only. Copy attachments are physically copied to instance records by the logging process. Reference attachments are only stored with the template objects, but are accessible from the instance records. Verification will include copying the template, versioning of the attachments and deleting of attachments.

Attachments created during logging have also been tested including versioning of edited attachments. Verification will include confirmation that reference attachments cannot be edited on instance records.

Electronic Signature

Electronic signature testing has been integrated within other test procedure, rather than being confirmed to a specific test section.

It has been tested in result entry tests, approval tests and in edit instance tests. These are portions of the test were these activities occur without signature enabled as well. The main features of this functionality are tested as follows: Activate on and off, Required User ID on and off, Checksums on.

During places where signatures are required, testing may include, entering incorrect passwords, entering incorrect user ids, review of checksums and transaction histories.

Lot Plans

The functionality of the Lot Plans form and the various types of lot plans has been validated. The following types of lot plans were created and lots were created from the lot plans using:

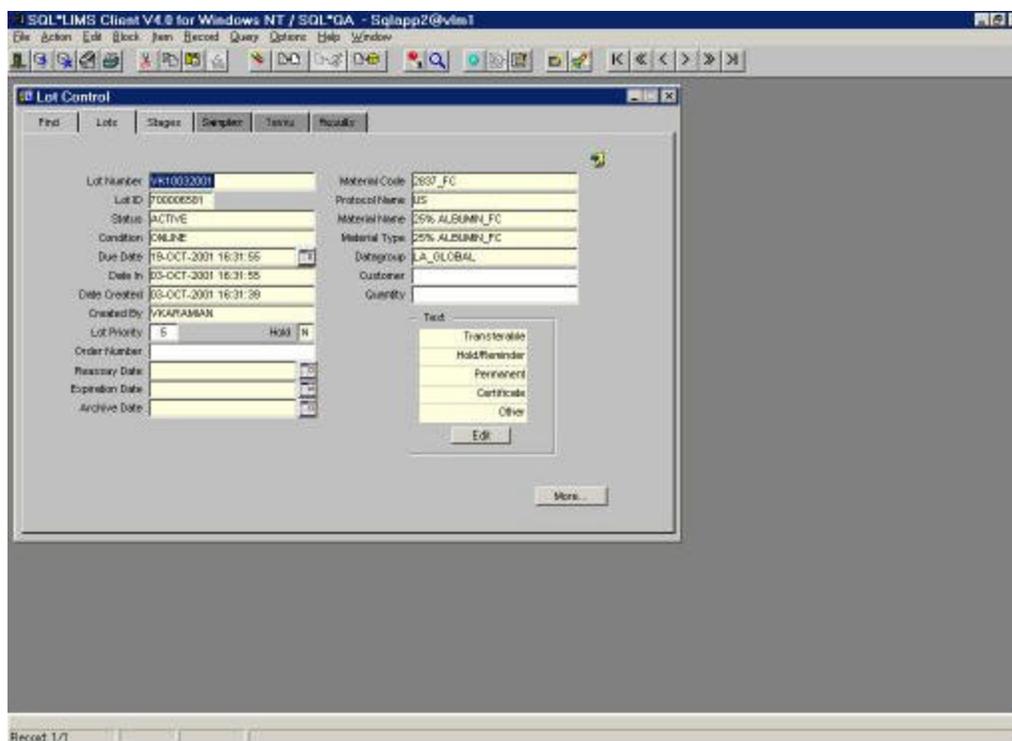


Figure 2 Lot Plan

Table 5

Logging Mode 1:	Sample plans are assigned at the Manufacturing Steps tab; and methods are not assigned.
Logging Mode 2:	Sample plans are assigned at the Manufacturing Steps tab; methods are assigned at the Step Task Plan tab.
Logging Mode 3:	Sample plans are not assigned; methods are assigned at the Step Task Plan tab.
Mixed Logging Mode:	Both Logging Mode 1 and Logging Mode 2 are used for the lot plan.

The following items relating to the Lot Plans form have been addressed:

- ? Ability to add parameters to the lot plan
- ? Default time units can be used
- ? Edit and Browse Attachments functionality
- ? Hold/Unhold functionality
- ? Lot Plan modification/edit ability
- ? Multiple specification assignments
- ? Skip lot scheduling
- ? Short Cut Menu: Copy Lot Plan, View Audit log
- ? Hiding Lot Plan fields

The functionality of the options accessible from each tab of the Lot Plan form has been validated as well.

Lot Create

The functionality of the Lot Create form and the ability to create lots, then create and receive sample have been validated.

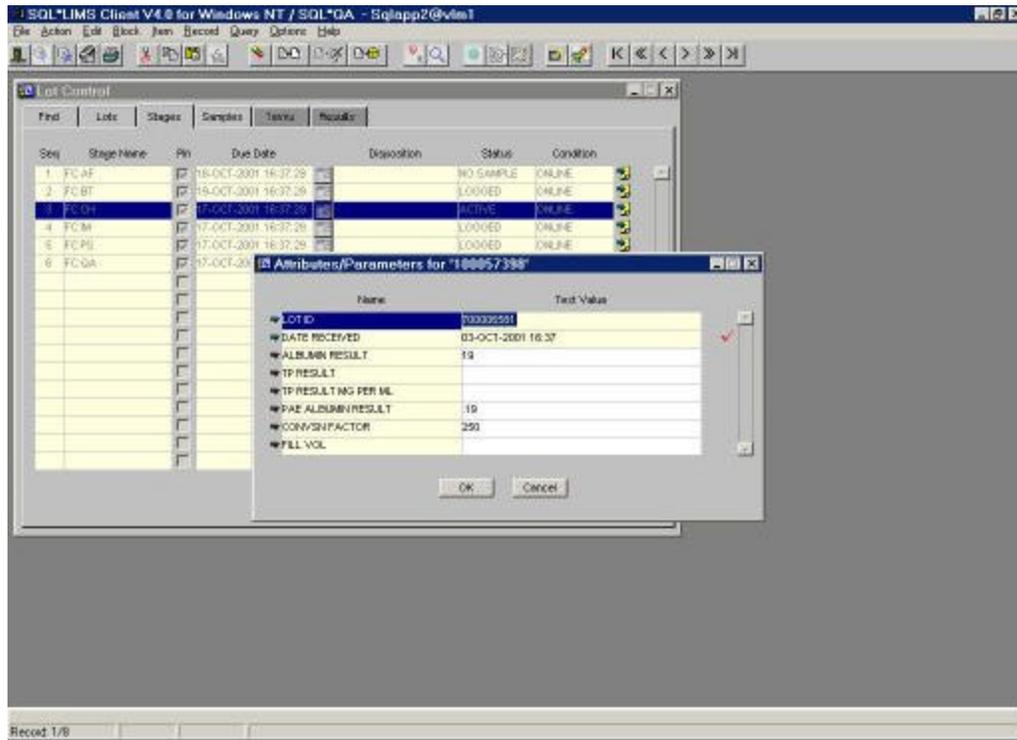


Figure 3 Lot Create

The functionality of the following items relating to the Lot Create form have been addresses:

- ? Ability to associate lots with other lots
- ? Ability to query from various fields in the form
- ? Edit and Browse Attachments Functionality
- ? Functionality of Edit Sublot, Create All, and Receive All buttons
- ? Hold/Unhold functionality
- ? Parameter/Attribute response entry
- ? Zoom form accessibility for SQL*QA Instance objects
- ? Skip Lot functionality
- ? Ability to copy attachments from the lot plan template to the lot instance
- ? Ability to generate top-most Parent Task Ids for each method logged
- ? Sublot creation
- ? Audit and Version Control functionality

The functionality of the options accessible from each region of the lot Create form has been validated as well.

Lot Control (Edit Lot)

The ability to edit information pertaining to a lot in the Lot Control form has been validated. The following functionalities have been validated:

- ? Ability to add or remove samples, task versions, methods, and replicates
- ? Ability to Approve data at various levels (role/rule dependant)
- ? Ability to Edit Condition at various levels
- ? Ability to Edit Condition of one or more lots at the same time (multi-record functionality)
- ? Ability to query from various fields in the form
- ? Edit and Browse Attachments functionality
- ? Hold/Unhold functionality
- ? Parameter/Attribute response entry
- ? Zoom form accessibility for SQL*QA Instance objects
- ? Result entry and update
- ? Stage Disposition functionality (role/rule dependant)
- ? Status Action and State history functionality
- ? Use of multiple specifications and secondary specifications
- ? Audit and Version Control

Lot Release

The ability to disposition lots that have a condition of Ready for release has been validated. The functionality of the lot Disposition form has been tested extensively. The following have been addressed:

- ? Ability to query from various fields
- ? Assign Second reviewer functionality
- ? Edit and Browse Attachments functionality
- ? Hold/Unhold functionality
- ? Zoom form accessibility for SQL*QA instance objects
- ? Verify that the correct roles and rule settings are required to disposition lots in various states.

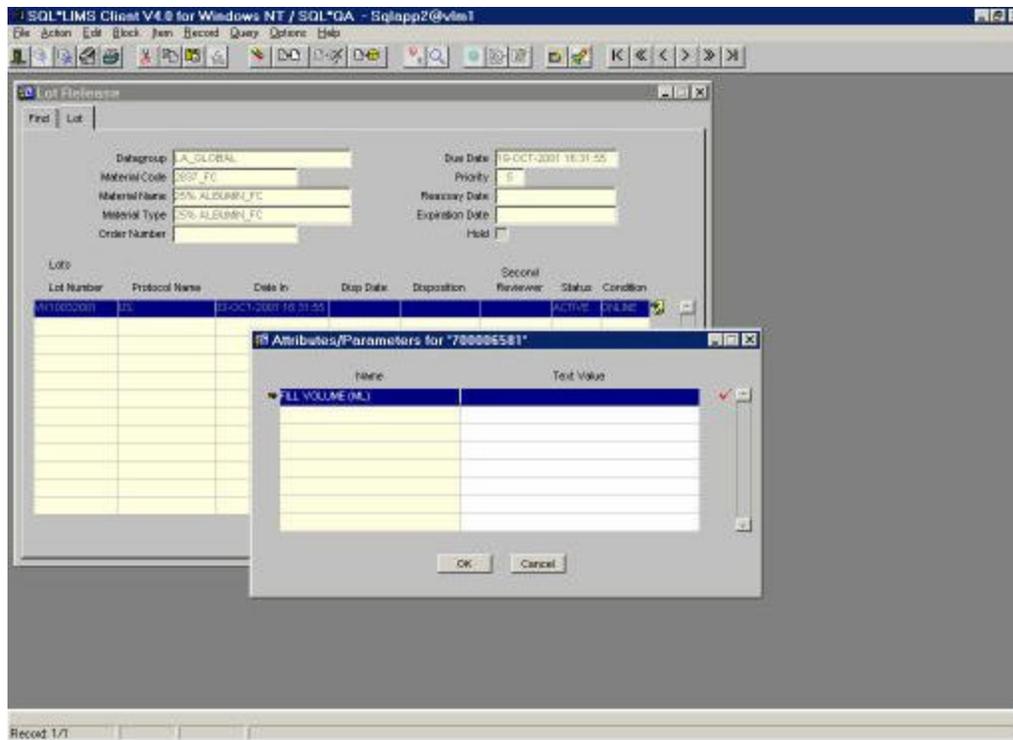


Figure 4 Lot Release

Lot Difficulties

The ability to create, modify, and delete Lot Difficulties records have been validated. The functionality of the following items have been addresses:

- ? Ability to query from various fields in the form
- ? Edit and Browse Attachments functionality
- ? Hold/Unhold functionality
- ? Lot Control form accessibility
- ? Zoom form accessibility for SQL*QA Instance objects

Lot Monitor

The ability of the lot monitor to correctly maintain and update condition changes have been checked throughout the procedures. The functionality of the following items have been addressed:

- ? Ability to execute status actions of type NOTIFY, PROGRA, ORACLE REPORTS, PL/SQL
- ? Ability to send SQL*LIMS memos when NOTIFY status actions execute
- ? Ability to interface with the event monitor when processing event requests for status actions of type PL/SQL, ORACLE REPORTS, and PROGRAM

- ? Ability to create a new disposition record when a lot is changed from condition DISPOSITIONED back to ONLINE or READY FOR RELEASE
- ? Ability to copy disposition attachments from the previous disposition record to the new one

Experimental Work and Data

Sample Logging

The logging process is verified for each of the forms through a number of steps. Each log request must reach a status of COMPLETE. From the log request, the creation of the appropriate instance record(s) is verified against information within the procedure, e.g. a submission, sample, etc. The assignment of instance attributes and parameters and reserved keywords were confirmed along with the absence of template attributes and parameters that are assigned to the various template objects used to log the samples. Similarly, instance attachments were confirmed to be present along with the absence of template attachments.

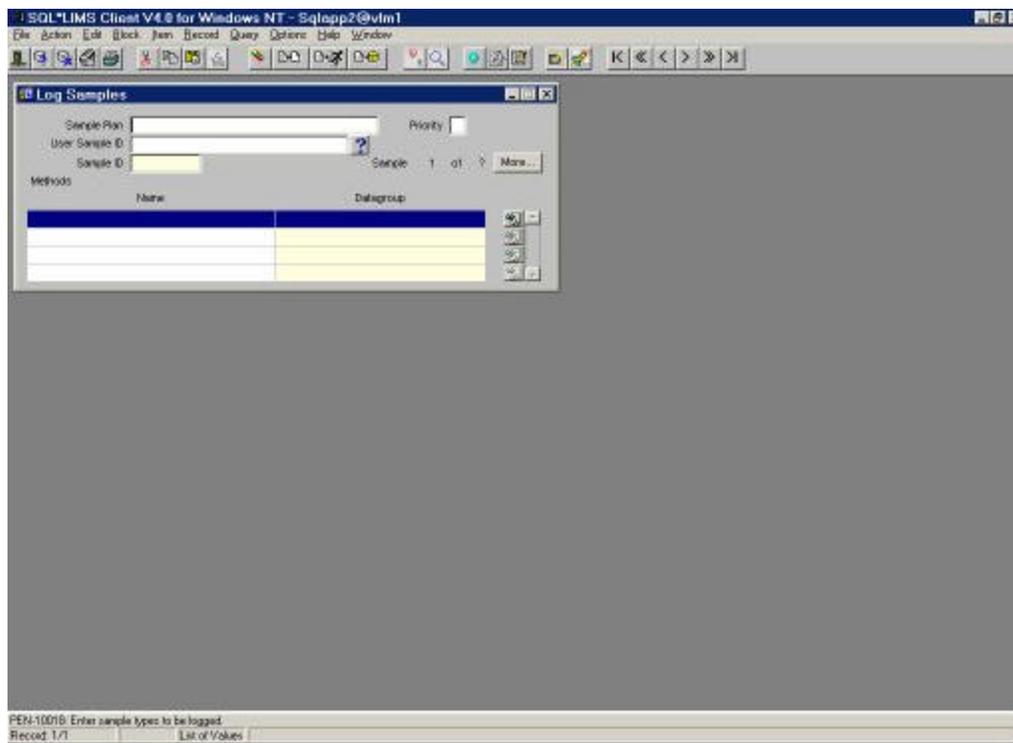


Figure 5 Log By Sample Plan

The assignment of condition value was verified against responses entered for any required attribute and parameters at the point of logging. Unanswered required parameters cause the record to be assigned the condition of INCOMPLETE. The ruleset requires entry of these parameters at the submission, sample and task levels, and several samples were logged to verify the proper assignment at each level of the instance hierarchy, both with and without responses.

The firing of the appropriate status actions was verified versus expected values for status and condition for each sample logged. These are verified for each level of the instance hierarchy.

The Edit Parameters section validates the form used to enter responses to the unanswered required attributes and parameters, but also validates the proper assignment of the ONLINE condition to the records.

Other items tested include the Copy Sample option, logging with a method group, the use of required and optional methods in a sample plan, logging using nested methods and the use of log request attributes.

Result Entry

The ruleset enforces automatic approval of instance records. Upon completion of results entry for each step of the procedure, most will be assigned the status APPROVED. The exception to this is a procedure that verifies that a null result will be forced to SUSPECT. This result will be given the status NOT APPROVED, as will its related instance records. Also, the expected condition COMPLETE will be verified.

The firing of Status Actions for the above changes in status and condition will also be verified, along with the state history, at each level of the instance hierarchy.

The screenshot shows the 'Enter Sample Results' window in the SQL*LIMS Client V4.0. The window is divided into several sections: 'Measure Limits', 'Detection Limits', 'Specification', and a table of results. The 'Sample ID' is 00007507 and the 'Method' is CH204-F. The 'User Sample ID' is YK10042001AH#M_FC CH-01. The 'Operation' is CH204-F. The table below shows the results for various components.

Seq	Component	Result	Units	Lims	Det	Spec	Spec	Pp
1	READING-1	10	NONE	N/A	N/A	N/A	N/A	1
2	READING-2	13	NONE	N/A	N/A	N/A	N/A	1
3	AVERAGE READING	11.5	NONE	N/A	N/A	N/A	N/A	1
4	CA REP DIFFERENCE	3	NONE	N/A	N/A	OUT	N/A	1
5	DILUTION FACTOR	3	NONE	N/A	N/A	N/A	N/A	1
6	CA STD READING		NONE					1
7	CALCULM RESULT		mg/100mL					1
8	COMMENTS	vk		N/A	N/A	N/A	N/A	1
1	IC DATA FILE	12.5	NONE	N/A	N/A	N/A	N/A	1
2	CHROM VALIDITY (%)	N	N/A	N	N/A	OUT	N/A	1

Figure 6 Results By Sample

The process of modifying a result will also be validated. Since auditing is being enforced, a predefined response indicating the reason for the change must be entered.

Miscellaneous items and options used to execute the procedures include adding text and setting the analyst.

Detection Limits

The series of detection limits have been created to verify the status assignment for results that are under or over the assigned limits. There are four possible combinations, as illustrated:

< LOR LIMITS	> HIGH LIMIT
SUSPECT	SUSPECT
SUSPECT	COMPLETE
COMPLETE	SUSPECT
COMPLETE	COMPLETE

Each of the four samples has been assigned one of the detection limits and three tasks. The predefined result values were entered for each of the three tasks, as follows:

1. Results entered within the limits.
2. Results entered below the lower limit.
3. Results entered above the upper limit.

The detection field was updated upon result entry confirmed versus expected values, as are the status and condition. The condition and condition level values are confirmed for these samples since Automatic Approval is being enforced.

Approving Data

Specific samples have been used to verify each of the modes (APPROVED, NOT APPROVED), since they were created with the appropriate rule settings to enforce said approval.

The procedures, records are set to either APPROVED or NOT APPROVED, and the resulting change in condition and the assignment of the condition level were confirmed. The view forms were used for this purpose. The changes are effected via the Approve forms.

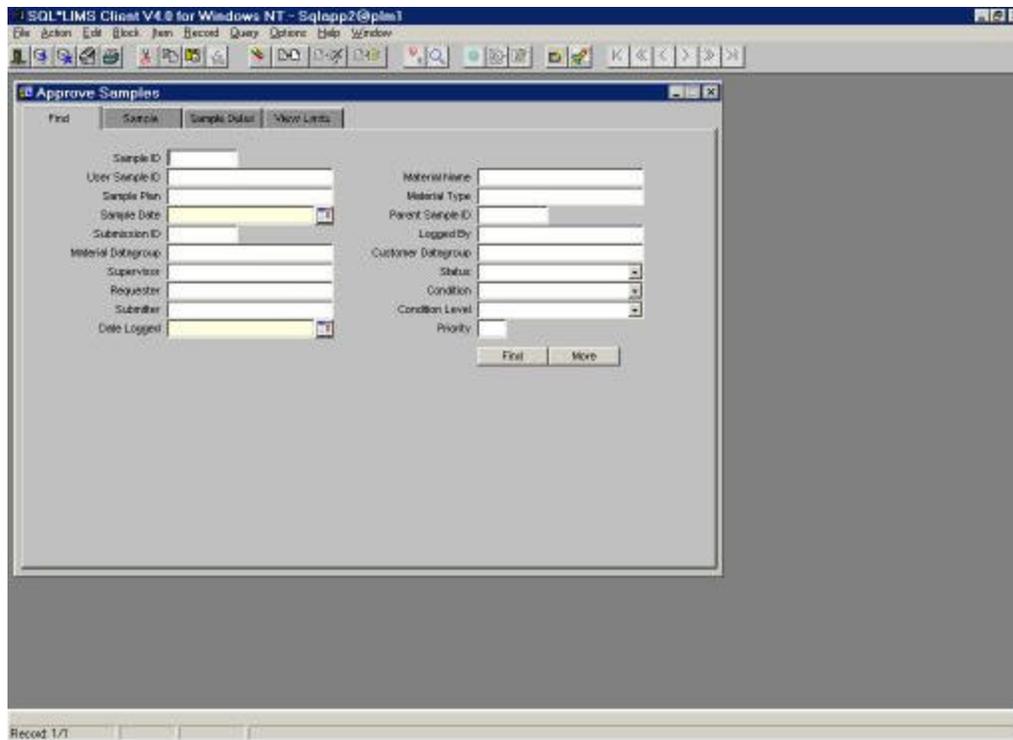


Figure 7 Approve By Sample

The modification of approved results is also examined, including the verification of the result version value for the changed results. Since auditing is enforced on these samples, a predefined reason for making the change must be entered.

Worklists

Worklists have been created by the TESTLIMS account, but will also be created with a valid workgroup to verify the settings of the worklist analyst. In addition to verifying the worklist analyst, the worklist due date, the number of unknowns and the total number of tasks assigned to the worklist will be confirmed.

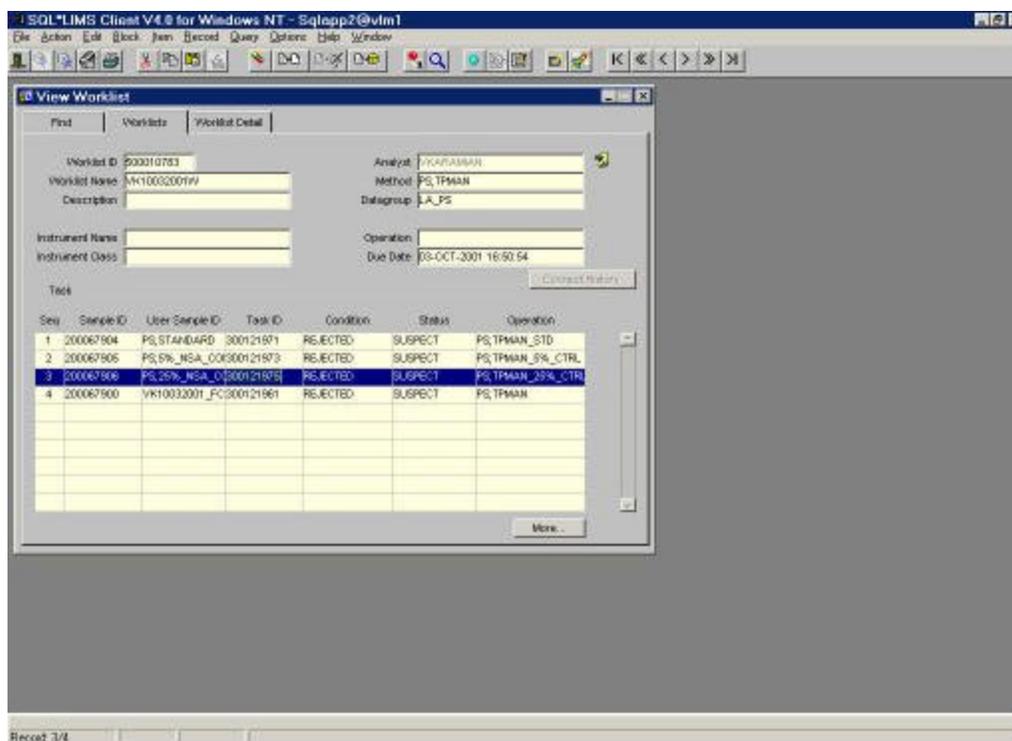


Figure 8 View Worklist

The deletion of worklists has been examined during the creation stage, as well as via the Edit Worklist form.

Other functions associated with editing worklists have been examined, including the addition of unknowns and the deletion of either QC types or unknowns. The Re-sequence Worklist option has been verified in conjunction with these functions.

The resulting worklist has been examined in detail via the View Worklist form to verify that they were created as specified, including parameter assignment and the firing of status actions.

The Worklist Condition options have been exercised and any change in the condition has been verified.

Results both within and outside of specification limits were entered via the Worklist Results Entry form. The reserved keyword SPECIFICATION has been assigned to the VALID 01 WORKLIST OPERATION for this purpose. Changes in status and the firing of the appropriate status actions have been verified.

The worklists was created using the three methods of task selection: Automatic, Choose and Barcode (manual).

Two worklists were used to exercise the approval process. One was set to APPROVED and the other to NOT APPROVED. The worklists were viewed to verify the propagation of the new condition value to the assigned tasks and the firing of approval status actions have been verified. Automatic Approval were enforced on these instance records.

Auditing and Versioning

Samplings of the mentioned settings were employed for that template object. Changes were made to the template objects and the function of each feature was tested. Template status have been tested by executing the various options that place the template in all of the allowed states, which are: NEW, LOCKED, CURRENT, SUSPENDED, MODIFIED, OBSOLETE.

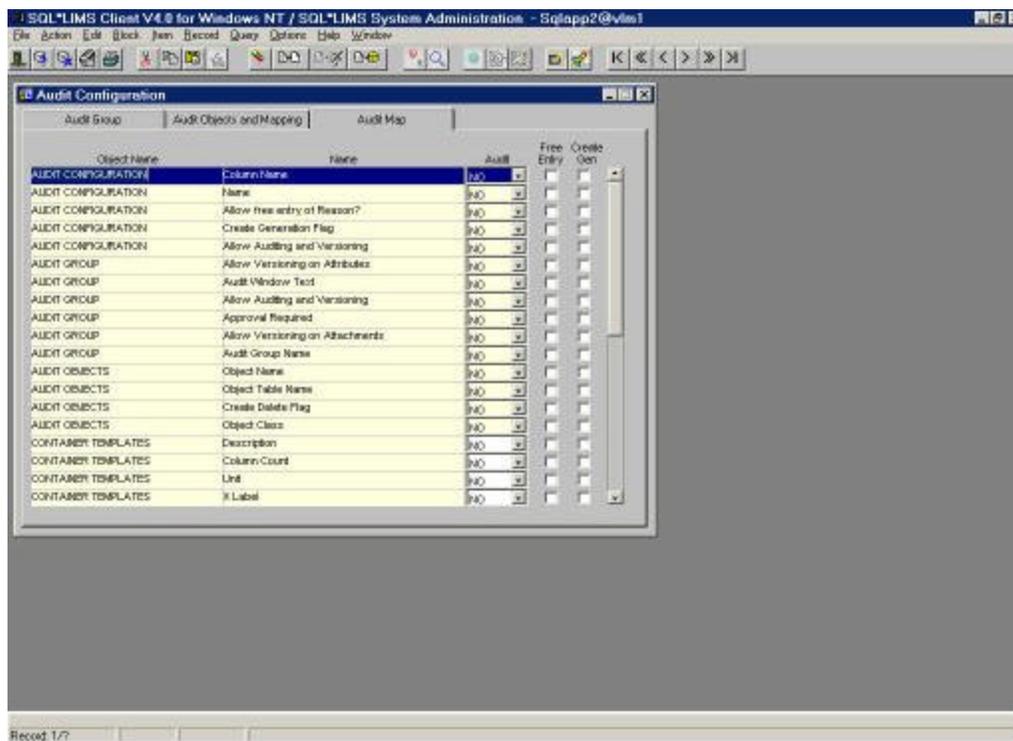


Figure 9 Audit Configuration

Deleting of template objects has also been tested to verify the following. Template objects that are used by other template objects cannot be deleted, template objects are the top-most template hierarchy can be deleted and certain template objects cannot be physically deleted and are set to OBSOLETE instead.

Events

An Oracle Report was installed as an event with distribution assigned to Preview, Printer and File. The installed event was run with multiple outputs specified. Each destination

type was selected in turn with parameter responses entered at run time. An Oracle Report was also installed and then run as an option from within a Results form.

A Program commonly available to Windows users, NOTEPAD.EXE, was installed and run as a client event. A Program supplied with the Validation Suite was installed and run as a server event.

A PL/SQL package was installed and run as an event on the client and the server.

An Oracle Report was also run as a status action and by placing the event on a schedule with the Install Event form. Template status actions are tested with an Oracle Report, with a PL/SQL event and by sending a memo, notification.

Suggestion for further work, new objectives, and action required

Our objective is to develop a Global LIMS strategy to facilitate controlled data centers, which will be located in North America and Europe. The purpose of this project is to establish a controlled, compliant database for quality control testing completed within the business area. These databases will serve as repositories for final test results, as well as a tool through which test results can be quickly gathered, collated and reported.

The goal of the program is to maximize opportunities to share information and leverage documentation and other intellectual property across sites and projects, and reduce the amount of repetitive work related to the implementation of these projects. This will ensure that individual projects are completed on time, on budget, within scope, and as efficiently as possible.

Personal Comments, Opinions, and Observations

Having the opportunity to be a part of a great team at Baxter Healthcare and working on such an important project as the validation of SQL*LIMS has given me valuable knowledge.

I have learned a new system that is used to manage critical information, and had the opportunity to be a part of the validation process. I have had the opportunity to help scientists and analysts with their questions and problems regarding the system, and based on their input I have been able to better understand the system, as well as have a better idea on how to improve the system for the users.

During validation some technical problems caused a drag in the project. However, we have overcome such technical difficulties and are well on our way towards our goal.

The most valuable experience for me was the documentation process of the work that I have completed. At first I had a little difficulty because I didn't know GDP (Good Documentation Practice.) I have had documentation practice prior, however, it was for a different industry, and therefore I had to learn a whole new set of rules and regulations because of FDA (Food and Drug Administration).

Index

A

Adding Attributes and Parameters to Instance Data	14
Approve Data	14
APPROVED	26
Attachments	18
Auditing and Versioning	17

C

Calculations Results Forms	10
----------------------------	----

E

Edit Condition	14
Edit Instance	14
Electronic Signature	18
Events	16

G

General Sample Receipt	16
------------------------	----

I

I Detection Limits	12
II New Spec and Specification	13

L

Lot Control (Edit Lot)	21
Lot Create	20
Lot Difficulties	22
Lot Monitor	22
Lot Plans	18
Lot Release	21

M

Multiple Specification Limits	13
-------------------------------	----

N

NOT APPROVED	26
--------------	----

O

Oracle Report	29
---------------	----

P

PL/SQL	16, 30
Program	30

R

Refire Status Actions	17
Result Formats	15
Results Entry	10

S

Sample Location	16
Sample Logging	10
Significant Figures	15

U

Userstamp and Timestamp	17
-------------------------	----

W

Work Assignment	10
Worklists	14

Figures

Figure 3 Detection Limits 12
Figure 7 Lot Plan..... 19
Figure 8 Lot Create 20
Figure 9 Lot Release 22
Figure 1 Log By Sample Plan..... 24
Figure 2 Results By Sample..... 25
Figure 4 Approve By Sample 27
Figure 5 View Worklist..... 28
Figure 6 Audit Configuration..... 29

Tables

Table 1..... 10
Table 2..... 15
Table 3..... 15
Table 4..... 17
Table 5..... 19