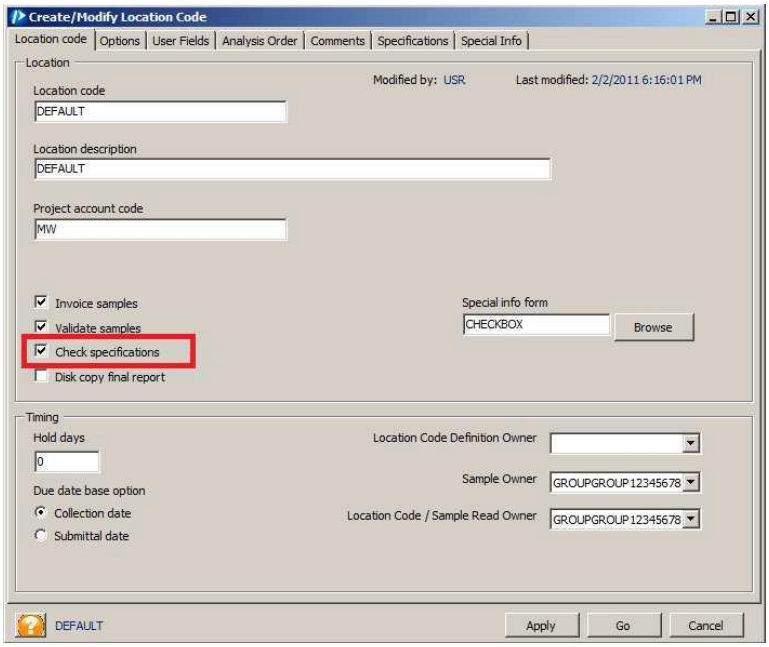
SPECIFICATIONS

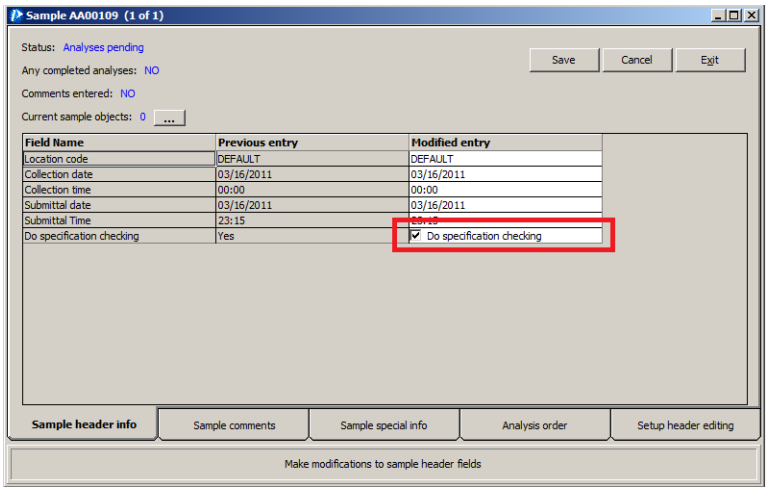
Introduction: LABWORKS enables the user to store and use specifications to monitor sample values against textual specifications and numerical specifications. When a specification is exceeded, LABWORKS will flag that result accordingly. LABWORKS also provides the user with the ability to create custom specifications to be used in reporting of sample values that are NOT evaluated when a specification is exceeded. These custom specifications most often used in custom reports. All specifications can be preserved with the sample data for future review and reporting.

Specification Evaluation

The choice to use/ignore specifications is made at the location code and sample level. When configuring a location code, there is a check box for: Use Specifications (DB: LOCLIST.DOSPEC)



Then as each sample is created, Check Specifications becomes a property of the sample. It can be modified or confirmed using Modify Sample. (DB: SAMPLE.DOSPEC)



Note: When a location code definition is updated, samples previously logged in are not altered. So a change in the location code won't affect existing samples.

Specification Hierarchy

LABWORKS implements a hierarchy to apply specifications at the most specific level possible. The levels are:

Analysis Code – Most Generic

Location Code – Sample point or Product specific

Sample – Location Code/Analysis code specifications persisted with the sample – Configuration Required

When evaluating specifications, LABWORKS first checks for Sample specific specifications. If there are no sample specs, then LABWORKS checks for Location Code specifications. If no location code specifications are present, then Analysis level specifications are used.

Specification Setup

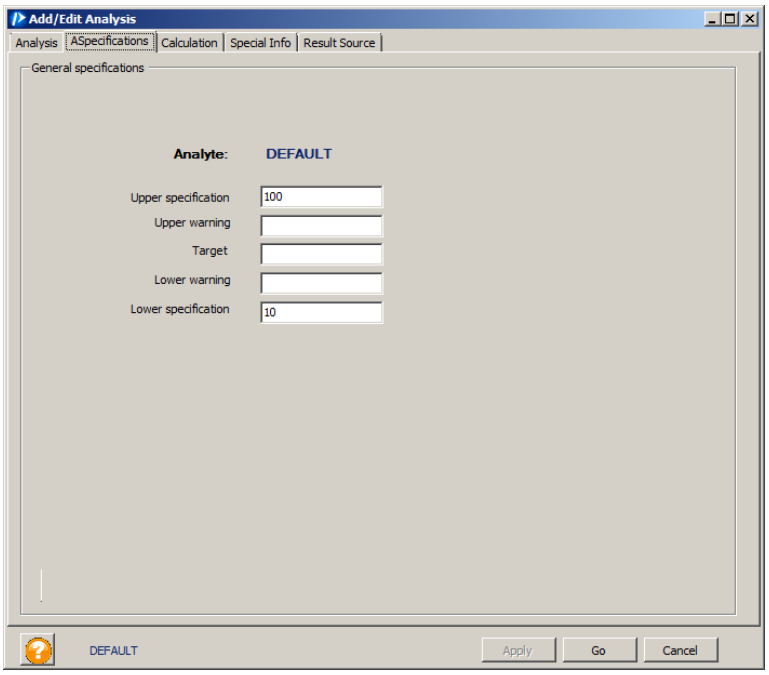
Setting up Analysis Code specifications is done using the analysis list maintenance program. The Specifications table is used to enter the specification values. Upper and Lower warnings and limits are numeric values. Target is a text value.

When comparing warnings and limits, LABWORKS reports an out of spec when the value is:

Greater than for upper limits, or Less than for lower limits.

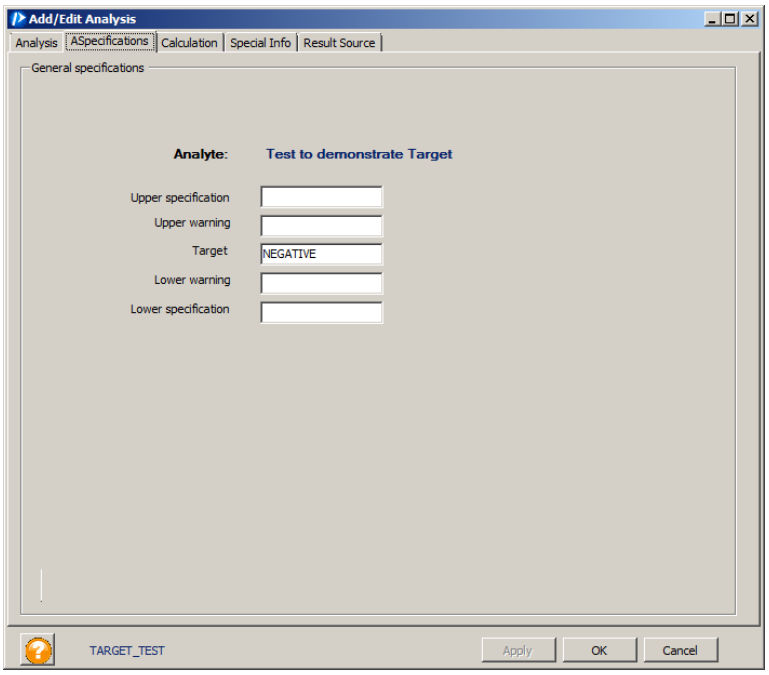
Example:

With Upper specification 100, if the value is 99 then no violation is reported. If the value is 100 then no violation is reported. If the value is > 100 then a violation is reported.



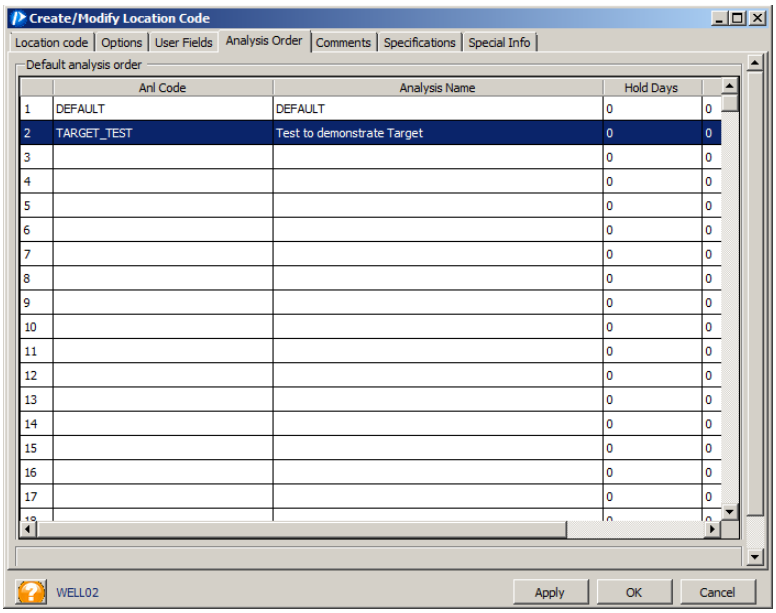
For Target values, it compares the strings, checking the Target specification is found in the value:

If the result is “POSITIVE” then a violation is reported. If the value is “NEGATIVE” no violation is reported. If the value is “NEGATIVE-1” then no violation is reported.

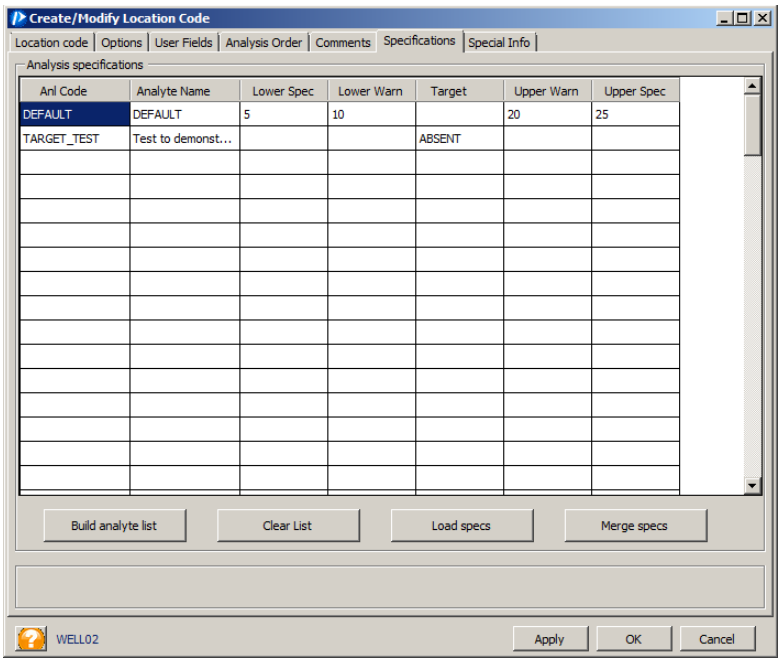

Add/Edit Analysis
Analysis
ASpecifications
Calculation
Special Info
Result Source
General specifications
Analyte:
Test to demonstrate Target
Upper specification
Upper warning
Target
NEGATIVE
Lower warning
Lower specification
TARGET_TEST
Apply
OK
Cancel

Location Code Specifications

Specifications at the location code level are created using the Location List maintenance program. To add specifications at the location code level, first add the analysis to the Analysis Order tab.



Then switch to the Specifications tab and press Build analyte list.



These specifications are then saved at the location code level.

Sample Level Specifications

To use sample level specifications, the login process must be configured to save the specifications with the sample.

Versions 6.2.230+

To configure persisting specifications with the sample add the following entry to LABWORKS.INI

[Windows_Multi_Sample_Login]

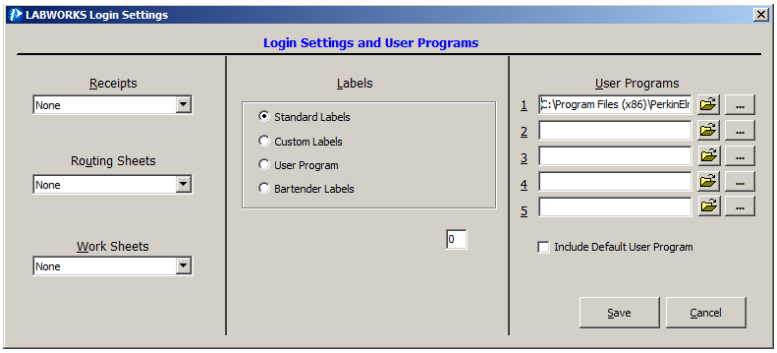
USE_PERSISTENT_SPECS = YES

[Windows_Single_Sample_Login]

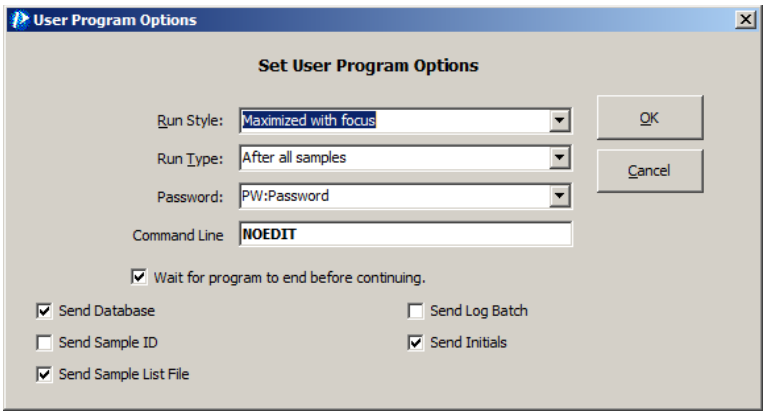
USE_PERSISTENT_SPECS = YES

Versions before 6.2.230

Configure the program SampSpecLogin6.exe to run as a post login program.

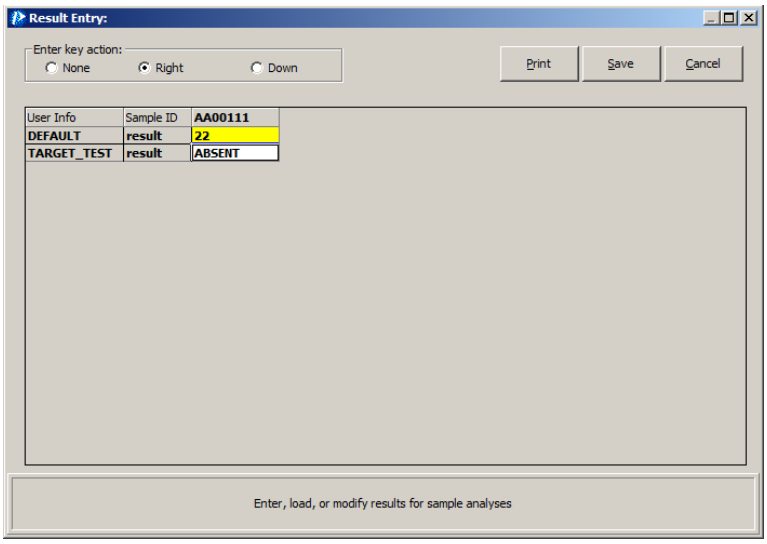


The program is SampSpecLogin6.exe

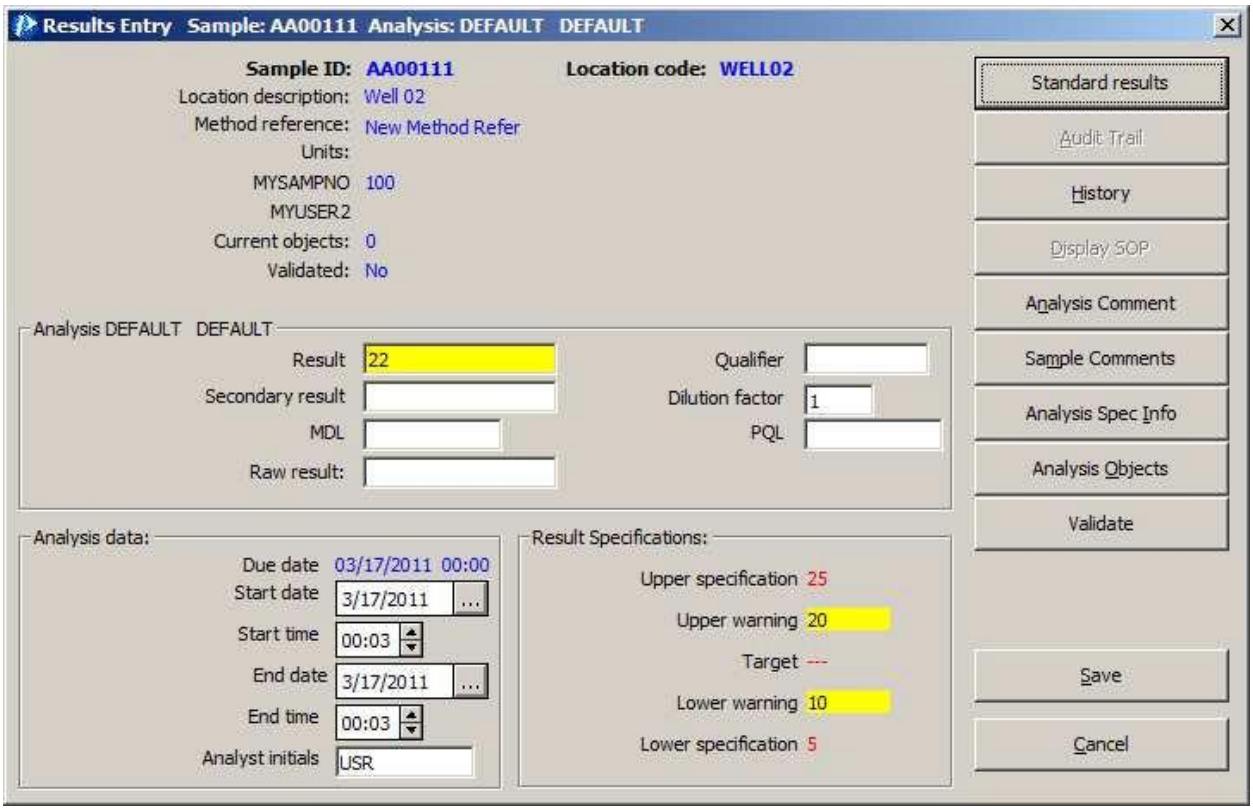


Entering Results and Specification Evaluation.

Specifications are evaluated during results entry. When a user enters a result, the value is checked using the specifications.



Detailed Edit



Note: Detailed edit puts the result into edit mode. If not configured to use Sample Specifications, and the Location Code or Analysis Code specifications have changed, displaying detail edit may evaluate a violation based on the new values that didn't previously exist.

Effective Specifications

Because of the specification hierarchy there is only one level of specifications active. And the complete set of specifications from the level is used. (Data in table may be different from previous screenshots.)

Example: Without Sample Specs

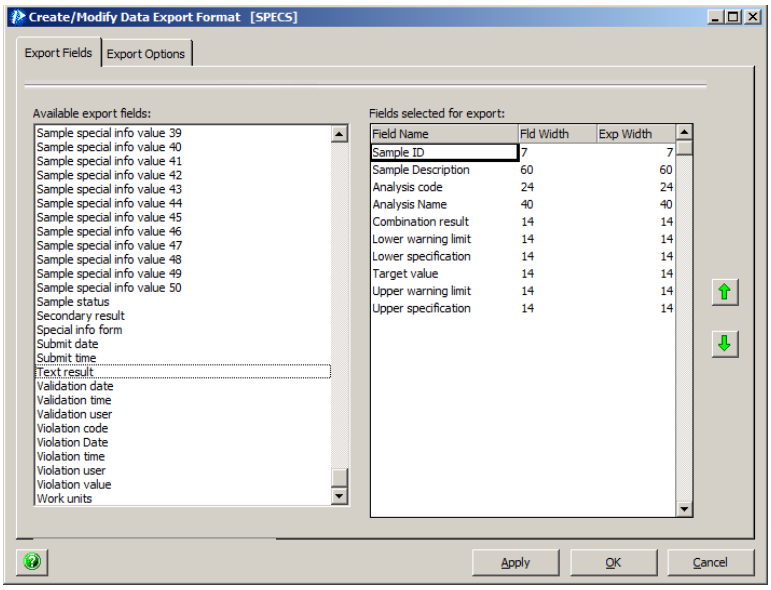
Analysis Code	Specification Type	Analysis Code Spec	Location Code	Effective Spec
DEFAULT	Lower Spec	10	5	5
DEFAULT	Lower Warning		10	10
DEFAULT	Target			
DEFAULT	Upper Warning		20	20
DEFAULT	Upper Spec	100	25	25
TARGET_TEST	Lower Spec			
TARGET_TEST	Lower Warning			
TARGET_TEST	Target	NEGATIVE	ABSENT	ABSENT
TARGET_TEST	Upper Warning			
TARGET_TEST	Upper Spec			

Example: With Sample Specs

Analysis Code	Specification Type	Analysis Code Spec	Location Code	Sample Spec	Effective Spec
DEFAULT	Lower Spec	10	5	6	6
DEFAULT	Lower Warning		10	10	10
DEFAULT	Target				
DEFAULT	Upper Warning		20	21	21
DEFAULT	Upper Spec	100	25	24	24
TARGET_TEST	Lower Spec				
TARGET_TEST	Lower Warning				
TARGET_TEST	Target	NEGATIVE	ABSENT	NEGATIVE	NEGATIVE
TARGET_TEST	Upper Warning				
TARGET_TEST	Upper Spec				

Exporting and Reporting Specifications

Specifications can be reported and exported. Add the export fields to Export/Report format.



When a violation is reported the VIOLTYPE and VIOLVALUE are saved with the result.

RESULT.VIOLVALUE is the value of the specification that was violated.

RESULT.VIOLTYPE is the type of violation.

Violation Type	VIOLTYPE Value
0	No Violation
1	Lower Spec
2	Lower Warning
3	Target
4	Upper Warning
5	Upper Spec

User Defined Specifications:

In addition to standard specifications, user defined entries can be added. As of Version 6.2, they are display only, no evaluation exists.

Version 6.2+

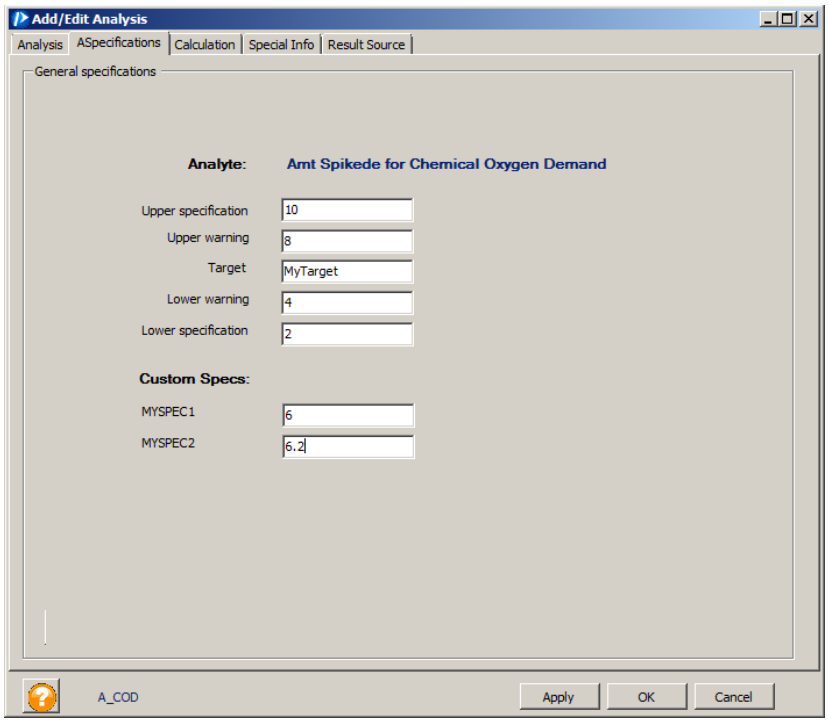
The SpecMaster table defines all of the specification types supported in the system, their UI descriptions, the relationship used for evaluating exceptions, and the severity level of those exceptions.

Example

SpecMaster table rows defining “standard” and “user defined” spec types. The custom specs in this example are *MYSPEC1 and MYSPEC2*

▶	LSPEC	GTE	Lower specification	20	H	1
	LWARN	GTE	Lower warning	90	M	2
	TARGET	ISCONTAINED	Target	30	H	3
	USPEC	LTE	Upper specification	10	H	5
	UWARN	LTE	Upper warning	80	M	4
	MYSPEC1	DISPLAY	User Specification	100	N	6
	MYSPEC2	DISPLAY	User Specificatio…	110	N	7

All custom specifications must be SPECTYPE = DISPLAY.



Version 6.1 and previous

User defined specifications are added by modifying the columns in the RLTSPECS table. A new column is added for each user defined spec. Maximum width is 255 characters.

LASSHLRL1040...dbo.RLTSPECS*

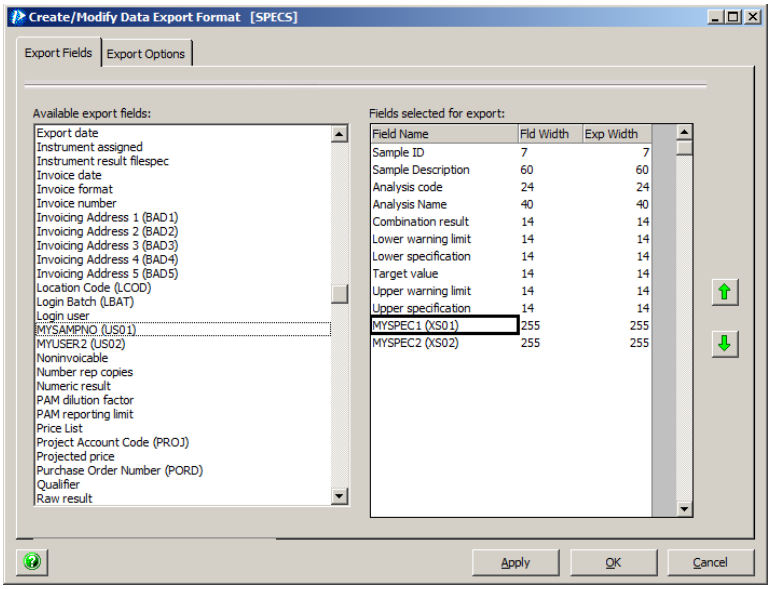
Column Name	Data Type	Allow Nulls
LOCCODE	varchar(24)	☐
ACODE	varchar(24)	☐
ANLNAME	varchar(40)	☐
LSPEC	varchar(14)	☑
LWARN	varchar(14)	☑
TARGET	varchar(14)	☑
UWARN	varchar(14)	☑
USPEC	varchar(14)	☑
MODDATE	datetime	☐
MODUSER	varchar(12)	☐
hSec	int	☐
MYSPEC1	varchar(14)	☑
MYSPEC2	varchar(14)	☑

Note: The Migration Utility converts records from RLTSPEC to SPECMASTER and RESULTSPECS when converting to 6.2

After the update and SysMgr key is turned on, logging into LABWORKS 6.2 and checking the maintenance programs should display the new SpecMaster specifications ready for use. Users shouldn't notice any difference in the User Interface of the application.

Reporting Custom Specs

Defined custom specification fields can be included in the export format in the same method as standard specification fields.



LABWORKS.INI Entries

Store persistent specifications for samples.

[Windows_Multi_Sample_Login]
USE_PERSISTENT_SPECS = YES

[Windows_Single_Sample_Login]
USE_PERSISTENT_SPECS = YES

SYSMGR Entries

SYSMGROPT	SYSMGRVAL	SYSMGRLVAL	SYSMGRBVAL	SYSFIELDTYPE
SAMP_SPEC_SOURCE	<see below>			IT

Valid Entries

NULL/Record doesn't exist	Sample Specification are not evaluated
SAMP_SPEC_TABLE	Sample Specification are evaluated
SAMP_SPEC_TABLE_ONLY	Sample Specification are evaluated – No Analysis or Location Code specs

Database Storage of Specifications in Version 6.2

SpecMaster Table

LABWORKS 6.2 introduces two new database tables for specifications, SpecMaster and ResultSpecs.

In order to use the new specification tables, you need to add the USE_62_PLATFORM SysMgr key and turn it ON (setting the value to TRUE or -1).

	SYSMGROPT	SYSMGRVAL	SYSMGRLVAL	SYSMGRBVAL	SYSFIELDTYPE
▶	USE_62_PLATFORM		0	-1	IB

ResultSpecs Table

Table ResultSpecs will store all entered specifications, including analyte general specifications, location specifications for analytes, and sample specifications (persisted) for analytes.

Example Records:

Analysis Code. Each specification has a corresponding record in the RESULTSPECS table.

	ANALYSIS	ANALYSIS	LSPEC	A_COD	Amt Spikede for ...	INDVALUE	2	NULL	NULL
	ANALYSIS	ANALYSIS	LWARN	A_COD	Amt Spikede for ...	INDVALUE	4	NULL	NULL
	ANALYSIS	ANALYSIS	MYSPEC1	A_COD	Amt Spikede for ...	INDVALUE	NULL	NULL	6
	ANALYSIS	ANALYSIS	MYSPEC2	A_COD	Amt Spikede for ...	INDVALUE	NULL	NULL	6.2
	ANALYSIS	ANALYSIS	TARGET	A_COD	Amt Spikede for ...	INDVALUE	NULL	NULL	MyTarget
	ANALYSIS	ANALYSIS	USPEC	A_COD	Amt Spikede for ...	INDVALUE	10	NULL	NULL
	ANALYSIS	ANALYSIS	UWARN	A_COD	Amt Spikede for ...	INDVALUE	8	NULL	NULL

Location Code. Each specification has a corresponding record in the RESULTSPECS table.

	LOCATION	DEFAULT	LSPEC	DEFAULT	DEFAULT	INDVALUE	1	NULL	NULL
	LOCATION	DEFAULT	LWARN	DEFAULT	DEFAULT	INDVALUE	2	NULL	NULL
	LOCATION	DEFAULT	USPEC	DEFAULT	DEFAULT	INDVALUE	20	NULL	NULL
	LOCATION	DEFAULT	UWARN	DEFAULT	DEFAULT	INDVALUE	10	NULL	NULL
	LOCATION	WELL02	LSPEC	DEFAULT	DEFAULT	INDVALUE	5	NULL	NULL
	LOCATION	WELL02	LWARN	DEFAULT	DEFAULT	INDVALUE	10	NULL	NULL
	LOCATION	WELL02	USPEC	DEFAULT	DEFAULT	INDVALUE	25	NULL	NULL
	LOCATION	WELL02	UWARN	DEFAULT	DEFAULT	INDVALUE	20	NULL	NULL

Sample Specifications. Each sample/specification has a record in RESULTSPECS

	SAMPLE	AA00106	LSPEC	A_COD	Amt Spikede for ...	INDVALUE	2	NULL	NULL
	SAMPLE	AA00106	LWARN	A_COD	Amt Spikede for ...	INDVALUE	4	NULL	NULL
	SAMPLE	AA00106	MYSPEC1	A_COD	Amt Spikede for ...	INDVALUE	NULL	NULL	6
	SAMPLE	AA00106	MYSPEC2	A_COD	Amt Spikede for ...	INDVALUE	NULL	NULL	6.2
	SAMPLE	AA00106	TARGET	A_COD	Amt Spikede for ...	INDVALUE	NULL	NULL	MyTarget
	SAMPLE	AA00106	USPEC	A_COD	Amt Spikede for ...	INDVALUE	10	NULL	NULL
	SAMPLE	AA00106	UWARN	A_COD	Amt Spikede for ...	INDVALUE	8	NULL	NULL

6.0

In LABWORKS version prior to 6.2 specifications were stored in the RLTSPECS Table.

Analysis Code. Each specification has a corresponding record in the RLTSPECS table. Location code = "&GENERAL"

	&GENERAL	$AREA_AN	Bromide	11300000				13900000
	&GENERAL	$AREA_AN	Chloride	27000000				35000000
	&GENERAL	$AREA_AN	Nitrate as N-NO3	70000000				90000000

Location Code. Each specification has a corresponding record in the RLTSPECS table.

	INF_COMP	$AN	Phosphate as P-...				10	
	INF_COMP	$AN	Sulfate				60	
	INF_COMP	COD	COD	100				1500
	INF_COMP	NH3	NH3	1				50

Sample Specifications. Each sample/specification has a record in RLTSPECS

	AE52111	BOD	BOD					5000
	AE52111	COD	COD					8500
	AE52111	EC	EC					5000

Questions/Configuration to review.

Does the sample have specification checking turned on?

Are sample specs being saved?

Are SYSMGR entries correct for sample spec evaluation?

Are user defined specs "DISPLAY" type?