AU/REMISOL STANDARD DICTIONARY

INTRODUCTION

The purpose of this document is to provide an overview of the changes in logic and rules included in the new standard dictionary. As you look at the new dictionary you will notice a number of changes. This dictionary was built specifically to provide a framework recognizing that each site may require specific rule changes and additions. It is not meant to be an install and run dictionary.

The team developing the dictionary was tasked with not only creating a standard Chemistry dictionary but creating one that encompassed Hematology. Because of this requirement, validation rules were moved from the parameter level to the upload level. In addition, a prefix was added to the names of the assays to allow for easy separation. A suffix of '_d' is added to tests that will auto-dilute for automation accounts.

NEW TEST PREFIXES

B_TEST	Hematology (Blood)
BF_TEST	Chemistry (Body Fluid)
C_TEST	COAG (PLASMA) *
CSF_TEST	Chemistry (CSF)
I_TEST	Immunochemistry (Serum)
S_TEST	Chemistry (Serum)
Sc_TEST	Special Calculation
U_TEST	Chemistry (Urine)
UA_TEST	IRIS Results (Urine)
UD_TEST	Drugs of Abuse
X_TEST	Other

*Reserved for future use

VALIDATION THEORY

Validation will occur in two ways. The first is a profile level validation. The second is individual test validation. A profile has been built for each sample type. Every test within a profile must pass the validation criteria to validate. This is possible since the AU analyzer transmits all results at once. When one of the tests fails, a comment is generated and every test in the specimen whether in the profile or not will be stopped. Validation does not require all the tests to be ordered only that those ordered are valid. Individual validation applies to tests that are not part of the GCHEM profile. These tests are validated individually. This allows the automatic validation of the GCHEM profile in situations like enzymes requiring dilution. It also allows the elevated enzymes to validate after dilution.

REMISOL/AU BEHAVIOR

This guide is specifically designed to cover Remisol/AU behaviors.

It is important to note that all settings in this dictionary will require IDENTICAL settings in the AU.

Failure to match settings WILL cause interface failure.

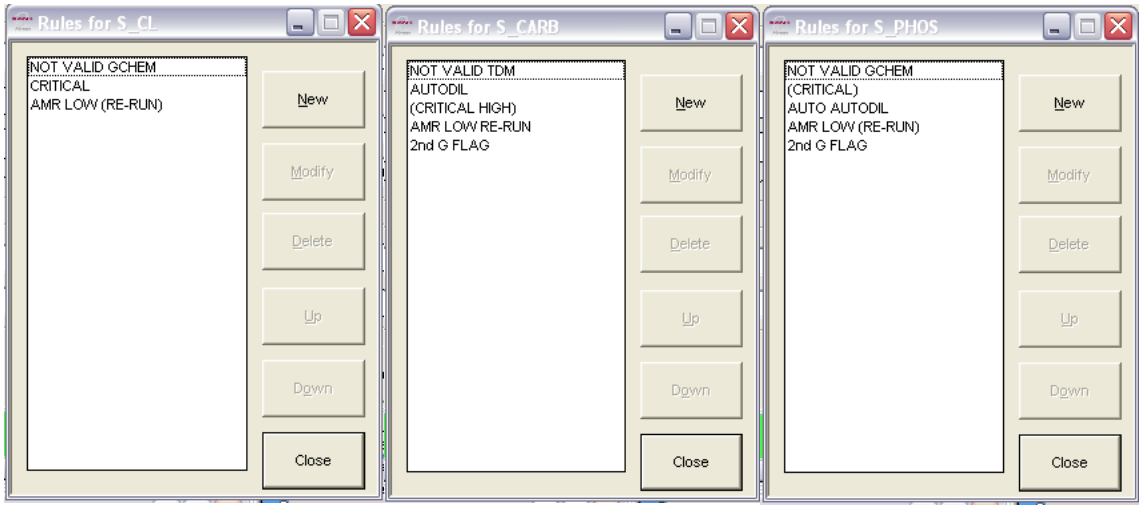
RULES

DOWNLOAD RULES

There are currently no download rules in the standard dictionary. Examples of basic download rules will be included in a supplemental documentation.

PARAMETER RULES

Although the validation occurs at upload, that validation is contingent on parameter rules. Some of these will be familiar, some will be new. Below are examples of various sets of rules for different parameters.



UPLOAD RULES

- Upload rules will perform multiple functions. Validation occurs at this level and it is governed by conditions and actions taken at the parameter level.
- Reruns are generated if required.
- Dilutions are reflexed when managed through Remisol instead of at the instrument level.
- Critical 'Called to' comments are created here and upload to the LIS with names and timestamps.

The following pages will detail these rules.

Standard Parameter Rules Explanations

NOT VALID

This rule adds the GCHEM comment used to stop validation at the upload level. When this comment is included, the upload rules look for the *absence* of this comment and validates if the comment is not present. Any alpha result will be considered invalid. If a less than result should be validated, an exception condition can be added here for those tests.

RULE

IF	(NOT (InValRange(S_TEST))) *AND (NOT (S_TEST Contains '<'))**
THEN	COMMENT2(GCHEM)

*May exist for certain assays

IF Conditions:

(NOT (InValRange(S_TEST)))	If the result is not NUMERICALLY in the validation range
(NOT (S_TEST Contains '<'))	Optional for tests that can validate if less than the lower AMR (i.e. DBIL)

THEN Actions:

COMMENT2(GCHEM)	Add a comment to indicate that the test and group as NOT VALID

Example

AUTODIL or MANUAL DIL

When an AU analyzer gives an F, D, E, @, or Z flag modify result to “AUTODIL” or “MAN DIL” as well as adding a comment with a description of the action and also documenting the actual result.

Autodil Rule

IF	(S_TEST <> 'AUTODIL') AND ((S_TEST Contains 'F') OR (S_TEST Contains 'D') OR (S_TEST Contains '@') OR (S_TEST Contains 'E') OR (S_TEST Contains 'Z'))
THEN	COMMENT2(S_TEST AUTODILUTION RUNNING) ;COMMENT2(ORIGINAL RESULT = {S_TEST}) ; RESULT('AUTODIL')

Manual Rule

IF	(S_TEST <> 'MAN DIL') AND ((S_TEST Contains 'F') OR (S_TEST Contains 'D') OR (S_TEST Contains '@') OR (S_TEST Contains 'E') OR (S_TEST Contains 'Z'))
THEN	COMMENT2(MANUALLY DILUTE) ; COMMENT2(ORIGINAL RESULT = {S_TEST}) ; RESULT('MAN DIL')

IF Conditions:

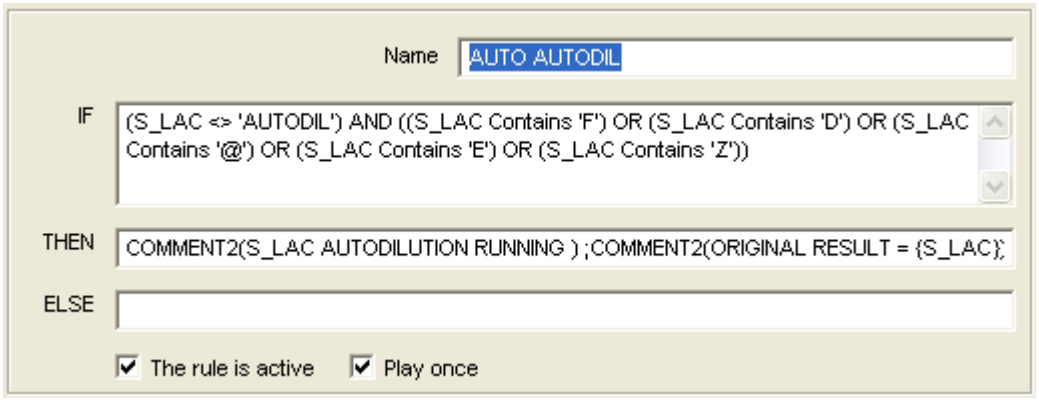
(S_TEST <> 'AUTODIL') AND (S_TEST <> 'MAN DIL') AND	Tests to see if the result equals ‘AUTODIL’, if not it is the first result. Tests to see if the result equals ‘MAN DIL’, if not it is the first result.
((S_TEST Contains 'F')	AU flag indicating a suppressed result
(S_TEST Contains 'D')	AU flag indicating a suppressed result
(S_TEST Contains '@')	AU flag indicating a suppressed result
(S_TEST Contains 'E')	AU flag indicating a suppressed result
(S_TEST Contains 'Z')	AU flag indicating a suppressed result

THEN Actions:

COMMENT2(S_TEST AUTODILUTION RUNNING) COMMENT2(MANUALLY DILUTE	Comment to reflex the S_TEST_d at the upload level
COMMENT2(ORIGINAL RESULT = {S_TEST})	Comment that contains actual result before being changed
RESULT('AUTODIL') * RESULT('MAN DIL')	Changes the result to ‘AUTODIL’ * Changes the result to ‘MAN DIL’ *

* AUTODIL may be replaced by MAN DIL for tests that should not be diluted onboard the AU

Example



Note: THEN-action line exceeds field width and isn’t entirely displayed

CRITICAL LOW/HIGH

Rule to handle results of critical chemistries that fall outside of the critical range. The validation range must reflect the critical range that applies to these chemistries.

RULE

IF	(S_TEST < S_TEST.ValidLow) OR (S_TEST > S_TEST.ValidHigh)
THEN	MSG(CRITICAL RESULT) ; COMMENT2(CRITICAL RESULT S_TEST)

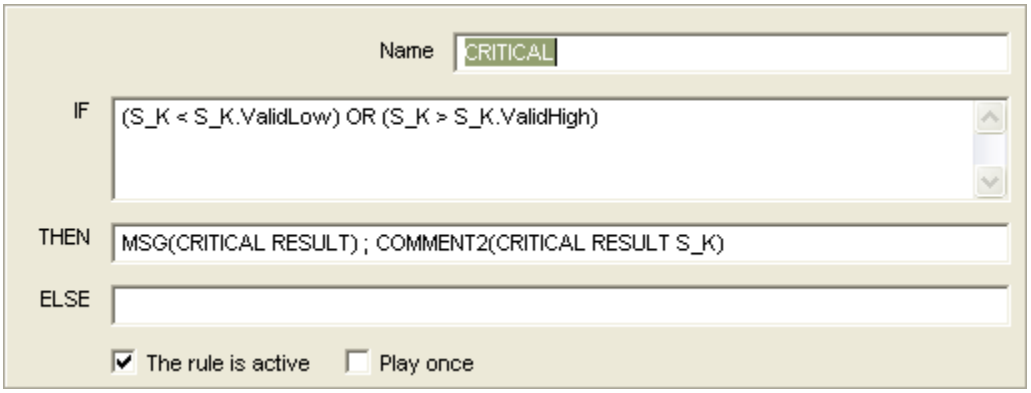
IF Conditions:

(S_TEST < S_TEST.ValidLow) OR (S_TEST > S_TEST.ValidHigh)	Validation range is set to the Critical range to trigger message and comments

THEN Actions:

MSG(CRITICAL RESULT)	Add critical result message in the MESSAGE window
COMMENT2(CRITICAL RESULT S_TEST)	Comment that is used for a rerun order and a CALL TO request

Example



AMR LOW

There are three versions of the LOW AMR rule. They are listed below and reflect how Remisol will manage results that flag below the AMR. These should be managed at a parameter level. The choice of these rules should be made after discussion with each customer. It is recommended to never use a conversion rule that modifies the result to a value this is clinically impossible.

Version 1

AMR LOW RERUN

When the AU generates a G or Gx flag indicating the sample is below the AMR, a comment is generated that triggers a re-run.

Comment Only. No conversion will be made. This rule would be used for assays such as S_NA or S_K that should not be converted.

RULE

IF	((S_TEST.FLAG = 'G') OR (S_TEST.Flag = 'Gx')) AND (NOT (RULES_COMMENT Contains 'S_TEST AUTO'))
THEN	COMMENT2(ORIGINAL RESULT = {S_TEST}) ; COMMENT2(Verify Sample Integrity and Rerun S_TEST)

IF Conditions:

(S_TEST.Flag Contains 'G')	AU result has a “G” flag associated with it
(NOT (RULES_COMMENT Contains 'S_TEST AUTO'))	The Comments do not contain an auto dilution comment

THEN Actions:

COMMENT2(Verify Sample Integrity and Rerun S_TEST)	Comment to trigger the rerun at the upload level

Example

Note: THEN Action line is not entirely displayed

AMR LOW (NO RE-RUN)

When the AU generates a G or Gx flag indicating the sample is below the AMR, the rule converts the flag to the low AMR limit (<x.xx)

Low AMR replacement after first run. No rerun recommended. Applies to assays such as S_ACTM and U_MTP.

RULE

IF	((S_TEST.FLAG = 'G') OR (S_TEST.Flag = 'Gx')) AND (NOT (RULES_COMMENT Contains 'S_TEST AUTO'))
THEN	COMMENT2(ORIGINAL RESULT = {S_TEST}) ; RESULT('<LOW_AMR')

IF Conditions:

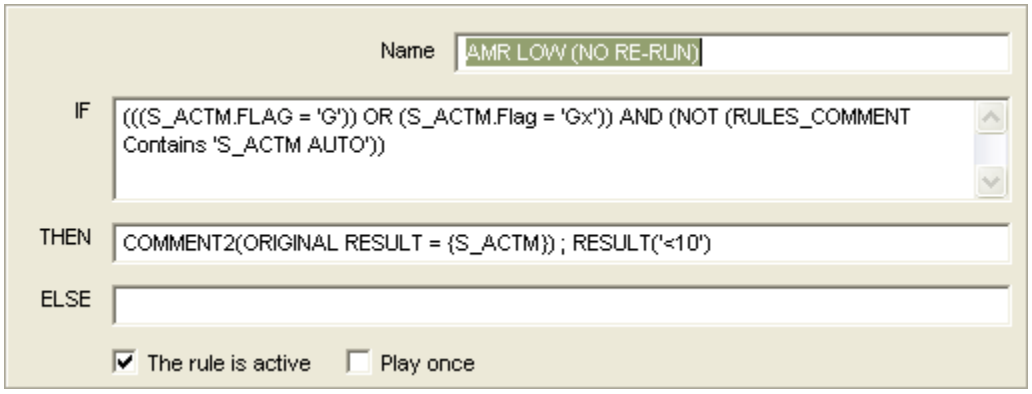
((S_TEST.Flag = 'G')	AU result has a “G” flag associated with it
(S_TEST = 'Gx'))	AU result has a “Gx” flag associated with it
AND (NOT (RULES_COMMENT Contains 'S_TEST AUTO'))	The Comments do not contain an auto dilution comment

THEN Actions:

COMMENT2(ORIGINAL RESULT = {S_TEST})	Inserts the original result into a comment
RESULT('<LOW_AMR') *	Changes the result to the low AMR value

- Change the ‘<LOW_AMR’ to the actual low AMR limit including decimal places. Include the single quotes around the value.

Example



2nd G FLAG

When the AU generates a G or Gx flag on the 2nd result (the rerun), confirming the sample is below the AMR, the rule converts the second result to the low AMR limit (<x.xx)

Low AMR replacement after rerun. Applies to assays such as S_ALT and S_LIP.

RULE

IF	((S_TEST.Flag = 'G') OR (S_TEST.Flag = 'Gx')) AND (S_TEST.Last.Flag Contains 'G')
THEN	COMMENT2(ORIGINAL RESULT = {S_TEST}) ; RESULT('<LOW_AMR') ; COMMENT2(Confirm order before release)

IF Conditions:

((S_TEST.Flag = 'G')	AU result has a "G" flag associated with it
(S_TEST.Flag = 'Gx'))	Or AU result has a "Gx" flag associated with it
((S_TEST.Last.Flag Contains 'G')	And AU result in previous run has a "G" flag associated with it
(NOT (RULES_COMMENT Contains 'S_TEST AUTO'))	The Comments do not contain an auto dilution comment

THEN Actions:

COMMENT2(ORIGINAL RESULT = {S_TEST})	Inserts the original result into a comment
RESULT('<LOW_AMR') *	Changes the result to the low AMR value
COMMENT2(Confirm order before release) **	Encourages confirmation on low TDMs

* Change the '<LOW_AMR' to the actual low AMR limit including decimal places. Include the single quotes around the value.
** Optional comment for TDMs

Example

AMR LOW (Immuno)

Rule that will modify a result that is numerically less than the AMR to a value. It will also play on a result with a '<' character in the result.

RULE

IF	(I_IMMUNO < 0.1) OR (I_IMMUNO Contains '<') OR (I_AFP = 'LOW')
THEN	COMMENT2(ORIGINAL RESULT= {I_IMMUNO}) ; RESULT('<0.1')

IF Conditions:

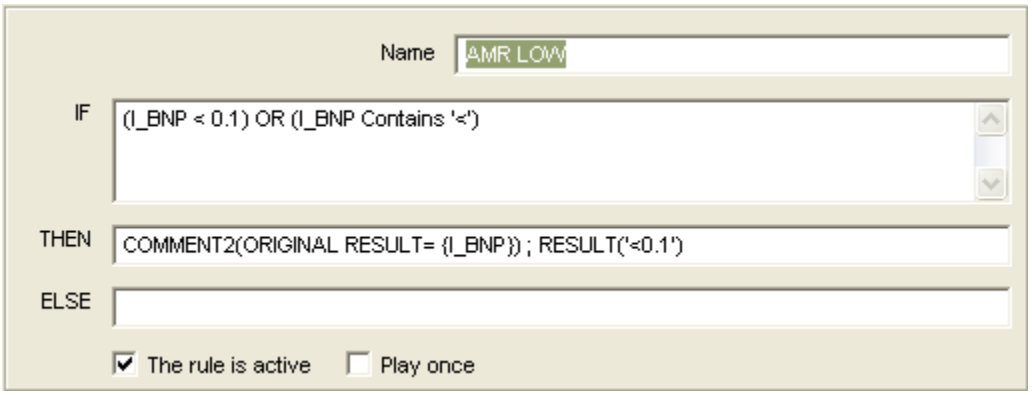
(I_IMMUNO < 0.1)	Checks if value is numerically below the AMR
(I_IMMUNO Contains '<')	Checks if the result contains a less than sign
(I_AFP = 'LOW')*	Checks if the result contains a result of "LOW"

*Applies to I_AFP and I_GI199

THEN Actions:

COMMENT2(ORIGINAL RESULT= {I_IMMUNO})	Places the original result in a comment
RESULT('<0.1')	Changes the result to a less than value

Example



AMR HIGH (Immuno)

Rule to handle results of critical chemistries that fall outside of the critical range. The validation range must reflect the critical range that applies to these chemistries. It is recommended to use the lowest value for the high calibrator associated for the high AMR. A support document will be included with that information.

RULE

IF	(I_IMMUNO > 5000) OR (I_IMMUNO Contains '>')
THEN	COMMENT2(ORIGINAL RESULT= {I_IMMUNO}) ; RESULT('>5000')

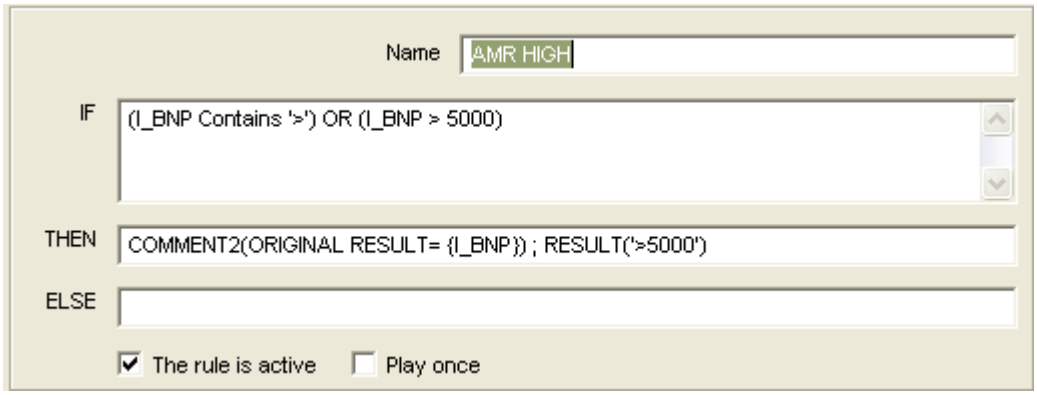
IF Conditions:

(I_IMMUNO > 5000)	Checks if value is numerically above the AMR
(I_IMMUNO Contains '>')	Checks if the result contains a greater than sign

THEN Actions:

COMMENT2(ORIGINAL RESULT= {I_IMMUNO})	Places the original result in a comment
RESULT('>5000')	Changes the result to a greater than value

Example



DAU POSITIVE

Converts flags and values to POSITIVE. The cutoff is entered into the validation range so that numbers will also be converted.

RULE

IF	(UD_TEST.Flag = 'P') OR (UD_TEST >= UD_TEST.ValidHigh)
THEN	COMMENT2(ORIGINAL RESULT = {UD_TEST}) ; RESULT('POSITIVE')

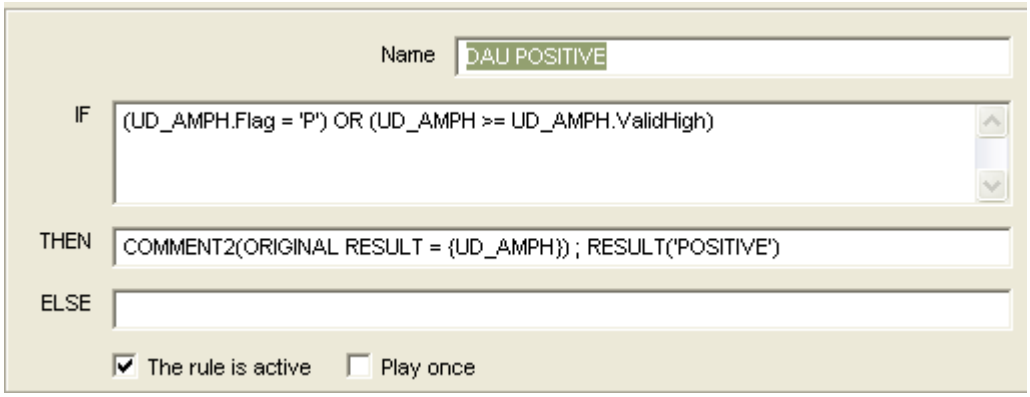
IF Conditions:

(UD_TEST.Flag = 'P')	AU returns a 'P' or an 'N' for drugs of abuse
(UD_TEST >= UD_TEST.ValidHigh)	The validation range is set to the cutoff. (QC sends numbers)

THEN Actions:

COMMENT2(ORIGINAL RESULT = {UD_TEST})	Documents the original result in comments
RESULT('POSITIVE')	Changes the result to match interpretation

Example



DAU NEGATIVE

Converts flags and values to NEGATIVE. The cutoff is entered into the validation range so that numbers will also be converted.

RULE

IF	(UD_AMPH.Flag = 'N') OR ((UD_AMPH Like '-*(N)') AND (NOT (UD_AMPH.Flag Contains 'D'))) OR (UD_AMPH <= UD_AMPH.ValidLow)
THEN	COMMENT2(ORIGINAL RESULT = {UD_TEST}) ; RESULT('NEGATIVE')

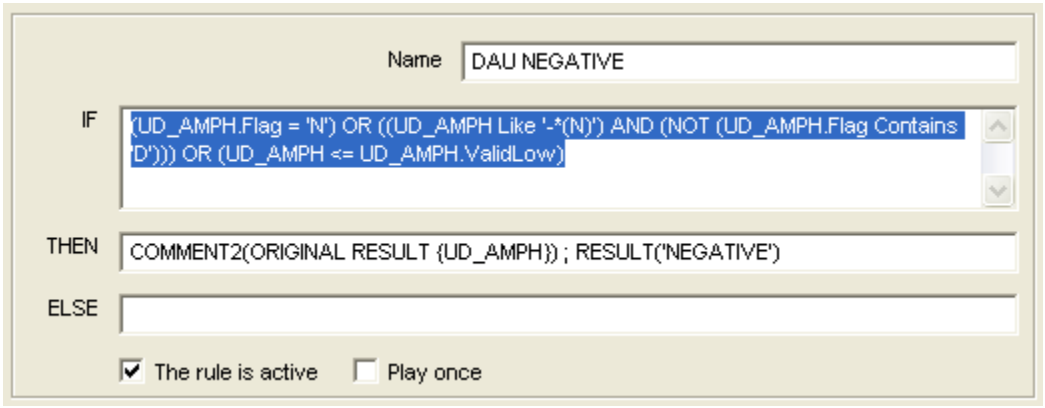
IF Conditions:

(UD_TEST.Flag = 'N')	AU returns a 'P' or an 'N' for drugs of abuse
(UD_TEST >= UD_TEST.ValidHigh)	The validation range is set to the cutoff. (QC sends numbers)

THEN Actions:

COMMENT2(ORIGINAL RESULT = {UD_TEST})	Documents the original result in comments
RESULT('NEGATIVE')	Changes the result to match interpretation

Example



NOT VALID

If flags are present the result will be flagged in comments as being not valid using the DAU comment. This will prevent the DAU profile from validating at the upload level.

RULE

IF	(NOT (UD_TEST Contains 'POS')) AND (NOT (UD_TEST Contains 'NEG'))
THEN	COMMENT2(DAU)

IF Conditions:

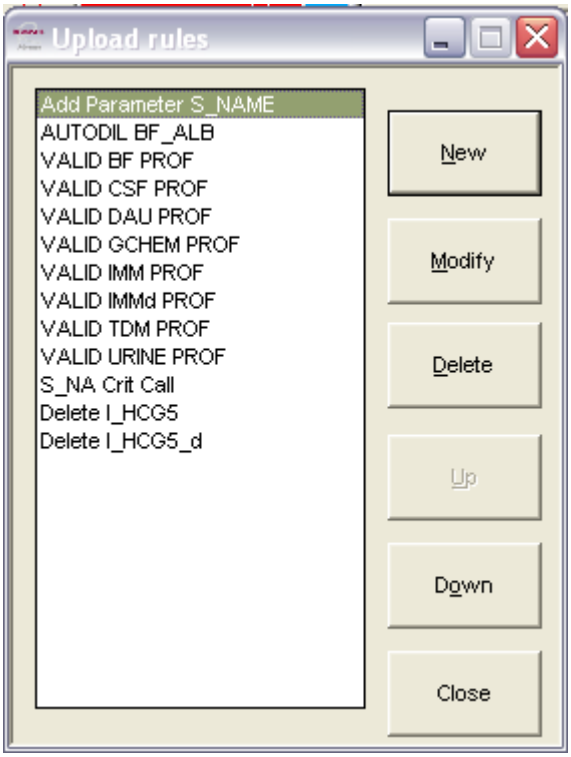
(NOT (UD_TEST Contains 'POS'))	If the result has not been modified to POSITIVE
(NOT (UD_TEST Contains 'NEG'))	And the result has not been modified to NEGATIVE then it is invalid

THEN Actions:

COMMENT2(DAU)	Adds comment to stop validation at the upload level

Example

Standard Chemistry UPLOAD rules explanations



Note (1)-These are example rules and the dictionary will contain many more rules than are displayed here. Several of these rules are duplicated for each parameter and do not appear in this list for the sake of simplicity. The Hematology rules have been deleted for the same reason.

Note (2)-If a listed chemistry is not installed or active at a customer site, and you choose to delete the chemistry parameter from the parameter table, the associated upload rules should also be deleted.

Add Parameter S_NAME

Adds a 'parameter' for entering a first and last name of an individual that is notified of a critical result. A separate upload rule then embeds this name into a comment that identifies the user, date, time, and the individual contacted. This comment can then be uploaded to the LIS

RULE

IF	(NOT (Exist_In_Tube(S_FIRSTNAME))) AND (RULES_COMMENT Contains 'CHEM_CRITICAL')
THEN	Reflex(S_FIRSTNAME) ; Reflex(S_LASTNAME)

IF Conditions:

(NOT (Exist_In_Tube(S_FIRSTNAME)))	Tests if S_FIRSTNAME exists
(RULES_COMMENT Contains 'CHEM_CRITICAL')	A critical result (any test) exists. Comment generated from parameter rule

THEN Actions:

Reflex(S_FIRSTNAME)	Adds the parameter S_FIRSTNAME
Reflex(S_LASTNAME)	Adds the parameter S_LASTNAME

Example

AUTODIL S_TEST

This rule orders a DIL test when dilutions are managed by Remisol instead of at the instrument (AU680 automation). The diluted test is configured as an OFFLINE test in Remisol as it will never receive a result back from the AU. The diluted result is returned to Remisol as a neat result.

RULE

IF	(Exist_In_Tube(S_TEST)) AND (S_TEST = 'AUTODIL')
THEN	Reflex(S_TEST_d)

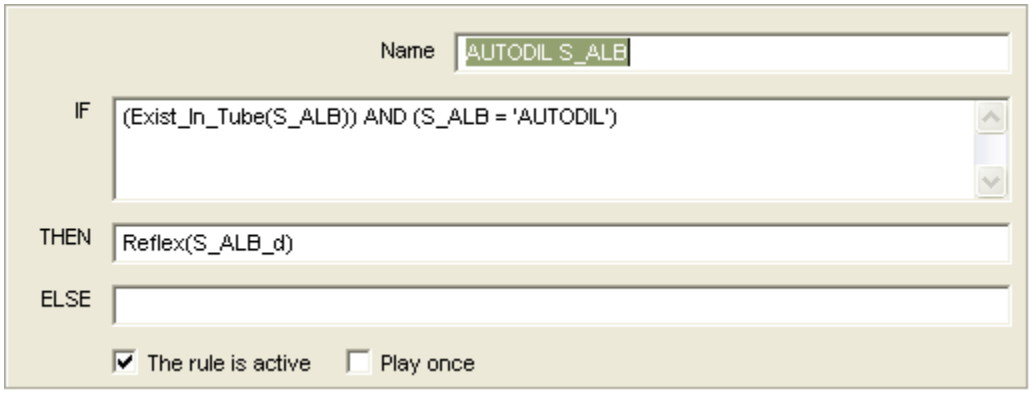
IF Conditions:

(Exist_In_Tube(S_TEST))	S_TEST exists in the Remisol sample
AND (S_TEST = 'AUTODIL')	Result requires dilution. Previous parameter rule changed result to 'AUTODIL'

THEN Actions:

Reflex(S_TEST_d)	Reflexes the diluted version of the test

Example



VALID group PROF

The rule looks to see if the group name or the word 'HOLD' exists in a rules comment. If they do not exist in the comments the profile is validated. Profiles are defined in Remisol under configuration and all the tests that are listed in the profile will be validated when this rule plays.

RULE

IF	(NOT (RULES_COMMENT Contains 'GCHEM')) AND (NOT (RULES_COMMENT Contains 'HOLD'))
THEN	Valid_Profile(GCHEM_VAL)

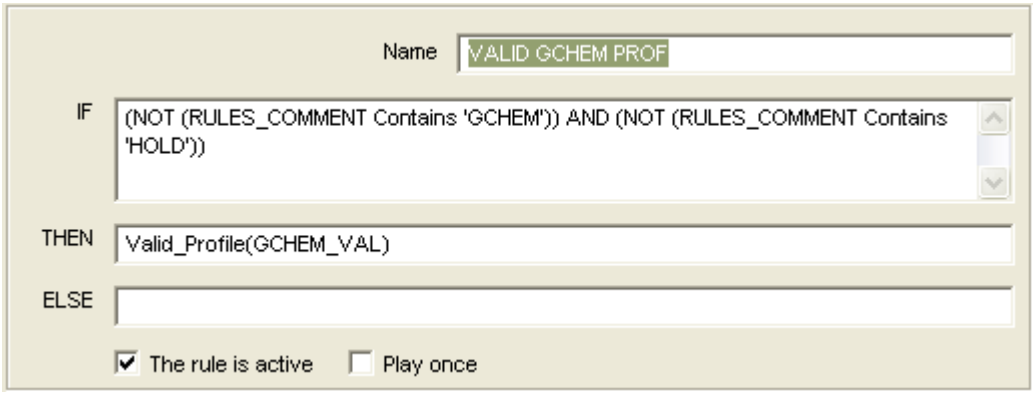
IF Conditions:

(NOT (RULES_COMMENT Contains 'GCHEM'))	Checks that 'GCHEM' is not in any Rules Comment
(NOT (RULES_COMMENT Contains 'HOLD'))	Checks that 'HOLD' is not in any Rules Comment

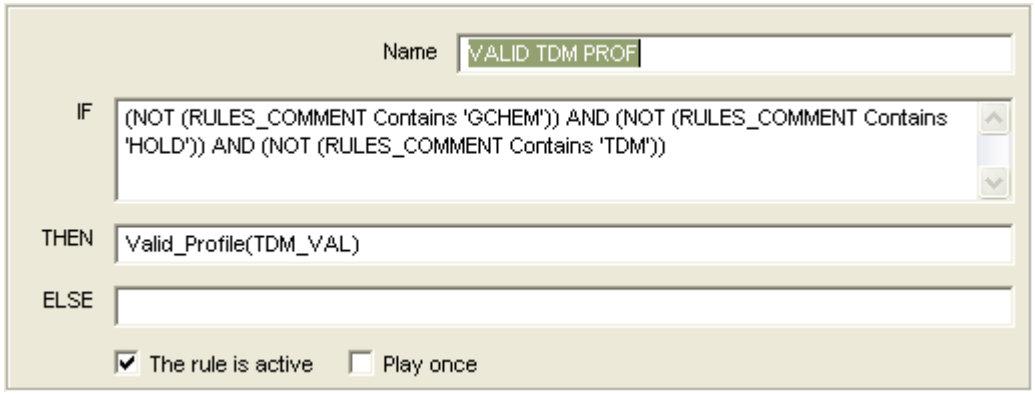
THEN Actions:

Valid_Profile(GCHEM_VAL)	Validates all tests defined in the profile named GCHEM_VAL

Example



Example



NOTE: To add validation dependency to another group, the group name is added to the conditions

S_CHEM Crit Call

Generate a comment to include the First and Last name of an individual notified of a critical result. The comment also adds the current user that is logged in along with the current date and time. The LIS must be able to accept comments from Remisol in order to properly document critical results.

RULE

IF	(Exist_In_Tube(S_FIRSTNAME)) AND (S_FIRSTNAME <> NULL) AND (Exist_In_Tube(S_CHEM)) AND (RULES_COMMENT Contains 'S_CHEM CRITICAL')
THEN	COMMENT(S_CHEM:Critical Result S_CHEM:{S_CHEM} Called to and read back by: {S_FIRSTNAME} {S_LASTNAME} at: {CUR_DATETIME} by:{USER})

IF Conditions:

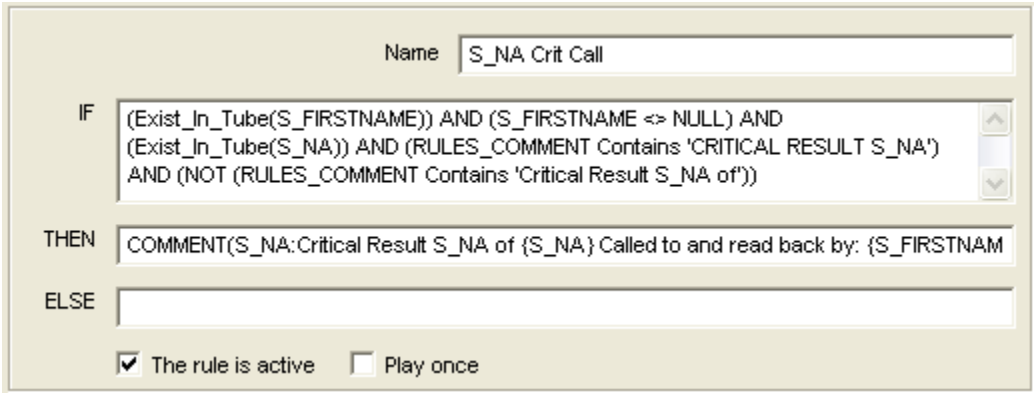
(Exist_In_Tube(S_FIRSTNAME))	Tests to see if S_FIRSTNAME is programmed in the sample
(S_FIRSTNAME <> NULL)	Tests to see if is equal to something (not null).
(Exist_In_Tube(S_CHEM))	Tests to see if S_CHEM is programmed in the sample
(RULES_COMMENT Contains 'S_CHEM CRITICAL')	Tests to see if the comments contain 'S_CHEM CRITICAL'
(NOT (RULES_COMMENT Contains 'Critical Result S_NA of'))	This makes sure that the rule only plays one time by looking for part of the text that the action creates the first time through.

THEN Actions:

COMMENT(S_CHEM:Critical Result S_CHEM of {S_CHEM} Called to and read back by: {S_FIRSTNAME} {S_LASTNAME} at: {CUR_DATETIME} by:{USER})	Add Critical call comment to include: First and Last Name, Time Stamp, and Tech ID assigned in Remisol. The 'S_CHEM:' associates the comment with the result instead of with the sample. {S_CHEM} = Keyword for test result {S_FIRSTNAME} = Keyword for First Name of person called to. {S_LASTNAME} = Keyword for Last Name of person called to. {CUR_DATETIME} = Keyword for Time Stamp {USER} = Keyword for Tech currently logged in to Remisol.

NOTE(1) : Chem names in braces {} are KEYWORDS and returns the value of the object
NOTE(2) : In an upload rule comments are sent as ORDER comments. Prefacing the comment with a CHEMNAME and a colon '(S_CHEM:' will send it as a RESULT comment

Example



DELETE – I_TEST / DELETE – I_TEST_d (Immuno)

When a test is diluted on the DxI or Access2, these rules handle the situation where Remisol receives a NEAT result and a DILUTED result. Generally, only one of the results should be valid while the other is suppressed, or both results could be suppressed (either low or high), and rarely would both results be valid. A previous validation rule validates one or the other or neither. *(The validation ranges can be set not to overlap so as to prevent both from being valid.)* This rule will comment the invalid result and then delete it, leaving the validated result to be released to the LIS.

RULE

IF	(Exist_In_Tube(I_NEAT)) AND (Exist_In_Tube(I_DILUTED)) AND (IsValid(I_DILUTED))
THEN	COMMENT2(I_NEAT - result {I_NEAT} removed) ; DEL_IN_TUBE(I_NEAT)

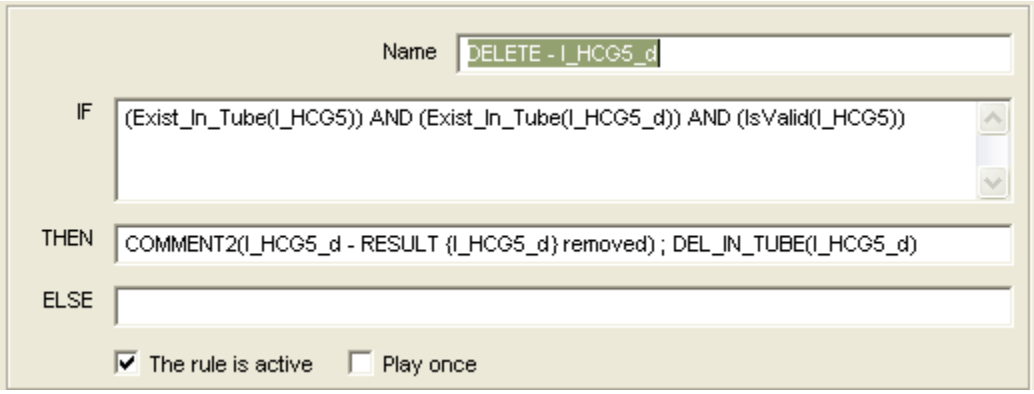
IF Conditions:

(Exist_In_Tube(I_NEAT)) AND	Ensures that the test exists in the sample program
(Exist_In_Tube(I_DILUTED))	Ensures that the test exists in the sample program
AND (IsValid(I_DILUTED))	Confirms that the DILUTED test is already valid

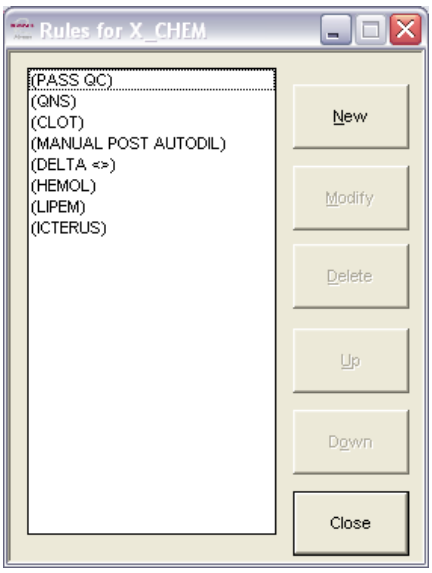
THEN Actions:

COMMENT2(I_NEAT - result {I_NEAT} removed)	Creates a comment with the original NEAT result prior to deletion
DEL_IN_TUBE(I_NEAT)	Deletes the NEAT result since the DILUTED result is validated

Example



OPTIONAL RULES (FOUND IN THE X_CHEM TEST)



These rules are optional. They are included as an example that can be copied and pasted into parameter rules if required.

- PASS QC
  - Used to validate QC results without applying rules. This is not needed on systems with EQC installed as EQC results do not have rules applied to them.
- QNS and CLOT
  - This rule will convert the AU flags for clotted or short sample specimens into an unambiguous result.
- MANUAL POST AUTODIL
  - A rule for performing manual dilutions on samples that cannot or should not be autodiluted.
- DELTA <>
  - A rule to be used for chemistries that will have delta checks performed on the results.
- HEMOL and LIPEM and ICTER
  - These rules can add comments to results and prevent validation based on customer limits and cutoffs.

PASS QC

This rule is used in accounts that do not have EQC and want their QC results to be exempt from any rules. When EQC is used, the results do not have rules applied to them so this rule is not necessary. If EQC is not installed, then most likely the lab will want to allow the QC result to be validated without having any rules play on the results.

This rule depends on the letters 'QC' being in the sample ID. When setting up QC for an instrument, if possible, include these characters as part of the barcode number.

For example: QCLevel1, QCLevel2, QCLevel3 or BIORAD1_QC, BIORAD2_QC
URINE1_QC, URINE2_QC QC_CARDIAC_1, QC_CARDIAC_2

Also, a DxC instrument sends the QC defined in it with a PATIENTID of 'CTRL-xxxxx' Your QC rule could then be constructed to read: IF (PATIENDID Like 'CTRL*') where the asterisk wildcard accepts any text. Some LIS prefix their QC sample IDs with 999xxxxx. You must make sure that you use: (SAMPLEID Like '999*') and not use Contains. A patient sample ID of 3340999 contains 999 and would be incorrectly treated as QC.

You must confirm that no patient sample ID will EVER contain 'QC' in it for this rule to be safe. Also caution the customer about programming manually using that letter sequence (not likely, but…)

RULE

IF	(SAMPLEID Contains 'QC')
THEN	VALID ; STOP

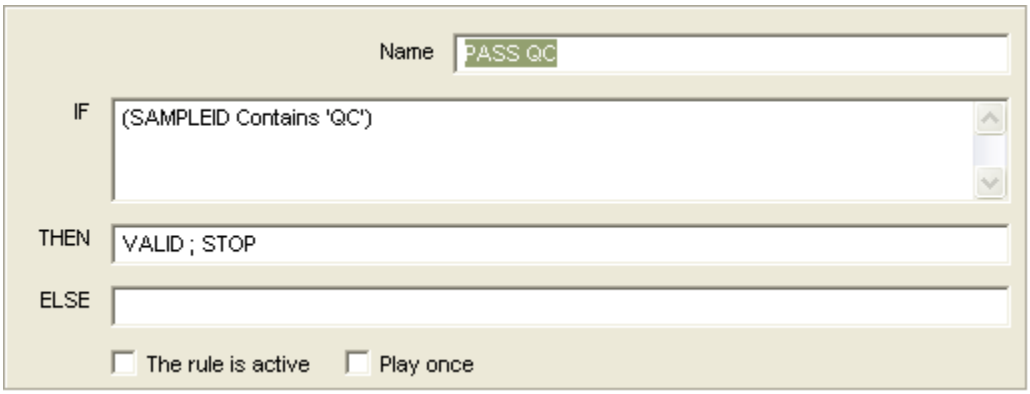
IF Conditions:

(SAMPLEID Contains 'QC')	Tests to see if the sample ID contains the letters 'QC' in that order.
(SAMPLEID Like 'QC??????')	Checks to see if an 8 character sample ID begins with 'QC'

THEN Actions:

VALID	Validates the QC result
STOP	Stops playing anymore rules for this parameter

Example



This should be the FIRST rule in the parameter list.

QNS

This optional rule can convert the # flag generated at the AU into a result of ‘QNS’ to make the result easier to interpret.

RULE

IF	(X_CHEM.Flag Contains '#')
THEN	COMMENT2(result {X_CHEM} changed to QNS) ; Result('QNS')

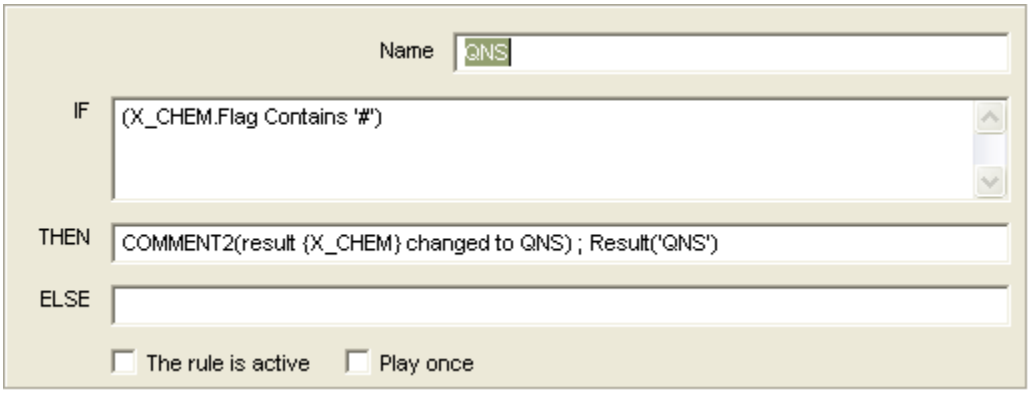
IF Conditions:

(X_CHEM.Flag Contains '#')	Tests to see if the result has a # flag indicating a short sample.

THEN Actions:

COMMENT2(result {X_CHEM} changed to QNS)	Adds a comment to document the flag received from the AU
Result('QNS')	Changes the flag to ‘QNS’

Example



CLOT

This optional rule can convert the % flag generated at the AU into a result of ‘CLOT’ to make the result easier to interpret.

RULE

IF	(X_CHEM.Flag Contains '%')
THEN	COMMENT2(result {X_CHEM} changed to CLOT) ; Result('CLOT')

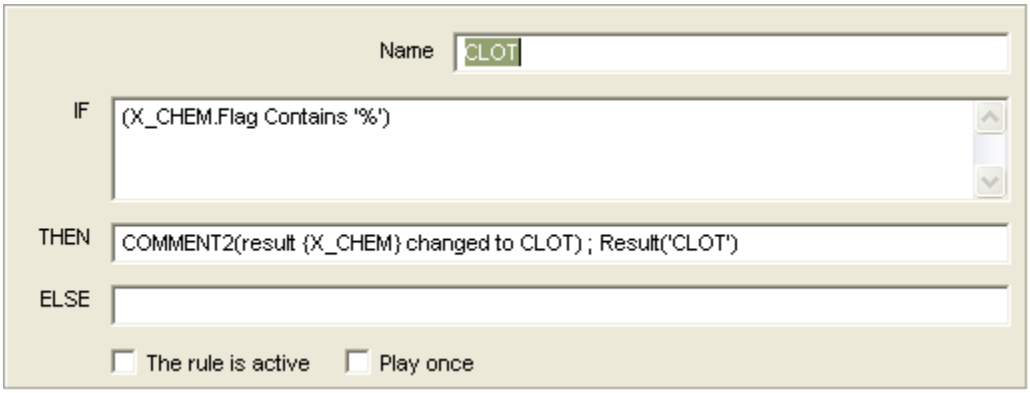
IF Conditions:

(X_CHEM.Flag Contains '%')	Tests to see if the result has a # flag indicating a short sample.

THEN Actions:

COMMENT2(result {X_CHEM} changed to CLOT)	Adds a comment to document the flag received from the AU
Result('CLOT')	Changes the flag to ‘CLOT’

Example



Manual Dilute After Auto Dilute

This optional rule will add a dilution comment to an assay that is still above the AMR after an automatic dilution.

RULE

IF	((X_CHEM Contains 'F') OR (X_CHEM Contains 'D') OR (X_CHEM Contains '@') OR (X_CHEM Contains 'E') OR (X_CHEM Contains 'Z')) AND (RULES_COMMENT Contains 'X_CHEM AUTO')
THEN	COMMENT2(X_CHEM DILUTE WITH SALINE)

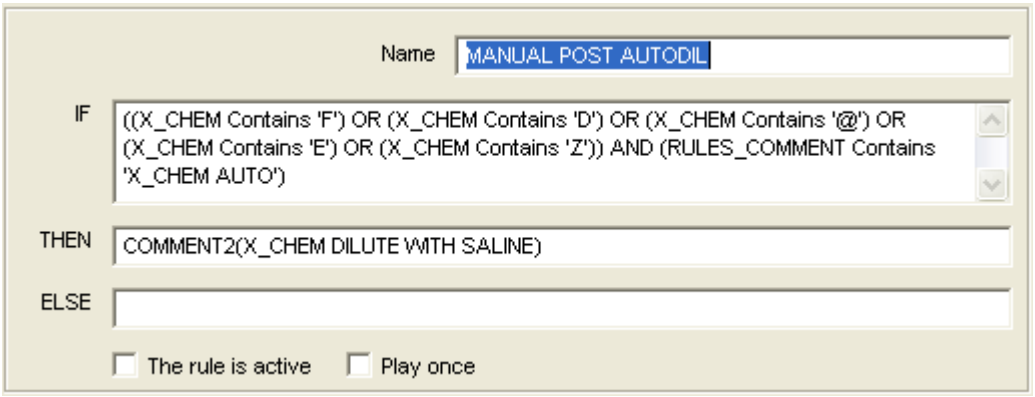
IF Conditions:

((X_CHEM.Flag Contains 'F')	Tests if result contains F
OR (X_CHEM Contains 'D')	Tests if result contains D
OR (X_CHEM Contains '@')	Tests if result contains @
OR (X_CHEM Contains 'E')	Tests if result contains E
OR (X_CHEM Contains 'Z')	Tests if result contains Z
AND (RULES_COMMENT Contains 'X_CHEM AUTO')	Checks if there is a comment 'X_CHEM AUTO' generated from another rule

THEN Actions:

COMMENT2(result {X_CHEM} changed to QNS)	Adds a comment to document the flag received from the AU
COMMENT2(X_CHEM DILUTE WITH SALINE)	Cues the tech to dilute with a specific diluent

Example



The diluent must be appropriate for the analyte. SALINE is an example.
A customer could also opt to include a dilution factor in the comment.
i.e. COMMENT2(X_CHEM DILUTE x5 WITH SALINE)

DELTA <>

Delta checks work with numeric results. If a result is not numeric, then a mathematical evaluation cannot be performed. For example, a previous TBIL result of '>25' is alpha and a current result of '0.3' cannot be compared numerically. In this case, the delta check does not pass nor does it fail. To compensate, this rule checks not only to see if they mathematically are acceptable but also whether or not they are the same.

RULE

IF	(X_CHEM.Prev <> NULL) AND (X_CHEM.Prev.Days < 30) AND (NOT (DeltaCheck(X_CHEM))) AND (X_CHEM <> X_CHEM.Prev)
THEN	COMMENT2(X_CHEM FAILED DELTA HOLD)

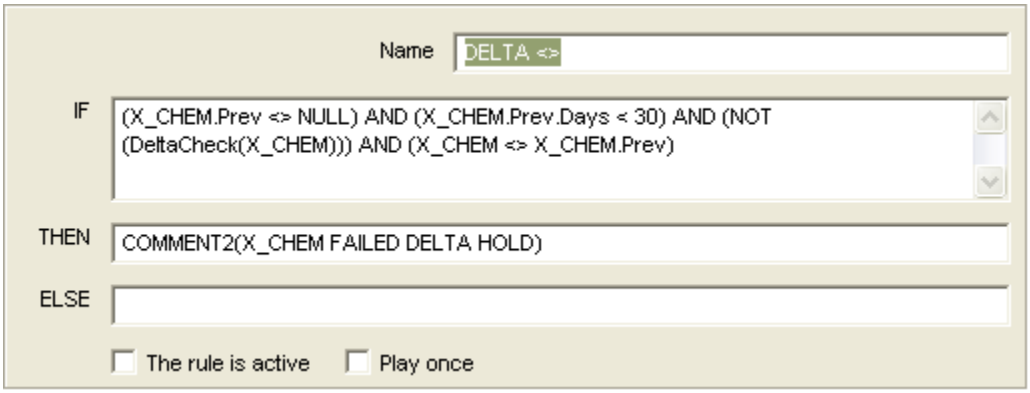
IF Conditions:

(X_CHEM.Prev <> NULL)	Confirms that a previous result does exist
AND (X_CHEM.Prev.Days < 30)	Ensures that the number of days is valid
AND (NOT (DeltaCheck(X_CHEM)))	Checks to see if it mathematically fails
AND (X_CHEM <> X_CHEM.Prev)	Tests to see if the current and previous are not the same

THEN Actions:

COMMENT2(X_CHEM FAILED DELTA HOLD)	Adds a comment with HOLD to prevent validation of results

Example



An upload rule looks to see if a comment contains 'HOLD' and if so prevents validation.

HEMOL

This optional rule can be used to add comments indicating the effect of hemolysis on an analyte. The value is determined by the customer and the comments may need to be site specific according to the needs of the LIS. For tests that may not be affected by slight to moderate hemolysis, the value would be the point at which the lab requests a redraw since no result would be reported.

RULE

IF	(Exist(S_HEMOL)) AND ((S_HEMOL >= 3) OR (S_HEMOL <= 5)
THEN	COMMENT2(X_CHEM affected by HEMOLYSIS HOLD)

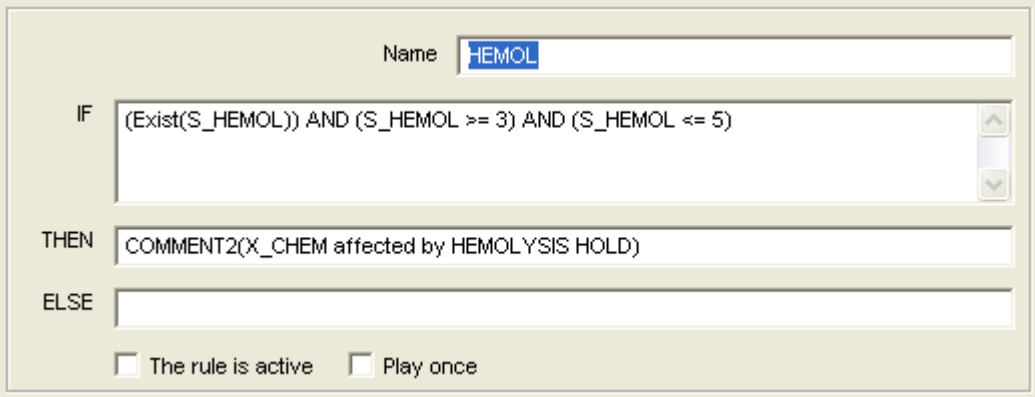
IF Conditions:

(Exist(S_HEMOL)) AND	Tests to see if S_HEMOL exists and that:
((S_HEMOL >= 3)	The S_HEMOL value exceeds a numeric limit
AND (S_HEMOL <= 5))	The S_HEMOL value is less than HEMOL numeric limit

THEN Actions:

COMMENT2(X_CHEM affected by HEMOLYSIS HOLD)	Adds HOLD to the comments to prevent validation

Example



LIPEM

This optional rule can be used to add comments indicating the effect of lipemia on an analyte. The value is determined by the customer and the comments may need to be site specific according to the needs of the LIS. For tests that may not be affected by lipemia, the value would be the point at which the lab may opt to ultra-fuge the sample.

RULE

IF	(Exist(S_LIPEM)) AND (S_LIPEM >= 3) AND (S_LIPEM <= 5)
THEN	COMMENT2(X_CHEM affected by LIPEMIA HOLD)

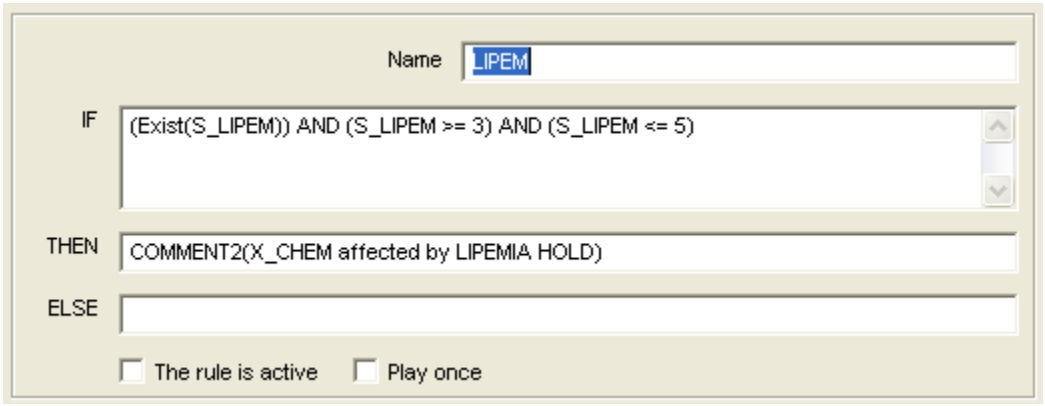
IF Conditions:

(Exist(S_HEMOL)) AND	Tests to see if S_LIPEM exists and that
(S_LIPEM >= 3)	The S_LIPEM value exceeds numeric limit
AND (S_LIPEM <= 5)	The S_LIPEM value is less than LIPEM numeric limit

THEN Actions:

COMMENT2(X_CHEM affected by LIPEMIA HOLD)	Adds HOLD to the comments to prevent validation

Example



ICTER

This optional rule can be used to add comments indicating the effect of icterus on an analyte. The value is determined by the customer and the comments may need to be site specific according to the needs of the LIS.

RULE

IF	(Exist(S_ICTER)) AND (S_ICTER >= 3) AND (S_ICTER <= 5)
THEN	COMMENT2(X_CHEM affected by ICTERUS)

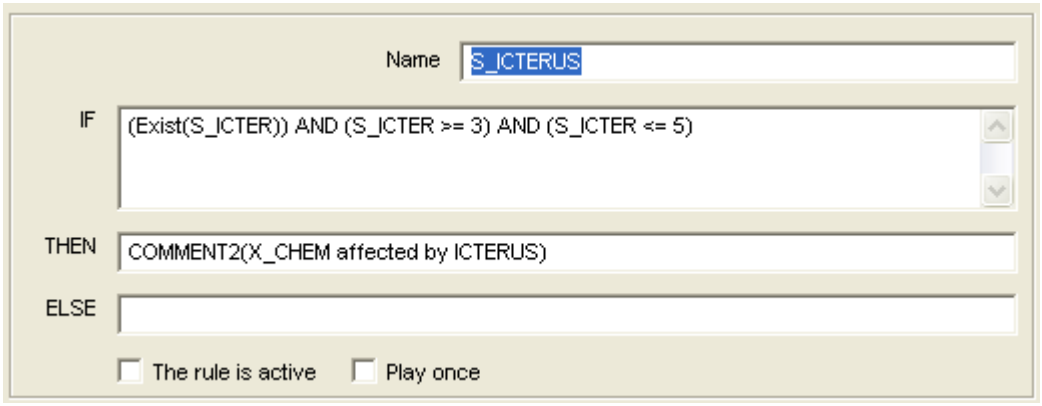
IF Conditions:

(Exist(HS_EMOL)) AND	Tests to see if S_ICTER exists and that
(S_ICTER >= 3)	The S_ICTER value exceeds numeric limit
AND (S_ICTER <= 5)	The S_ICTER value is less than numeric limit

THEN Actions:

COMMENT2(X_CHEM affected by ICTERUS)	Adds a comment. Does not HOLD the validation

Example



This document is a collaborative effort of:
Christine Daniels, Cindy Smith-Derby, Jon White, Michelle Armand, Paul Reischmann, Roy Traveria, Stefanie McFadden, Stephen A. Johnson