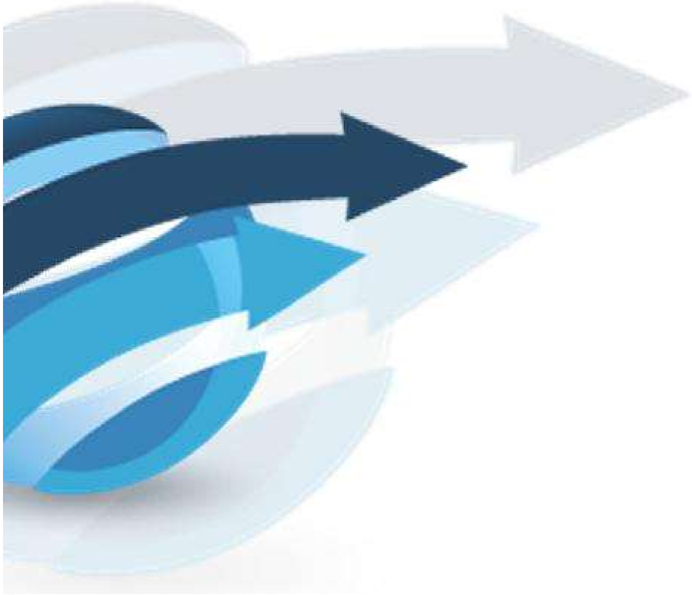


Pacific Knowledge Systems

RippleDown: Knowledge Builder V 8.3

The Knowledge Builder is the component of RippleDown that provides the user with the facilities for creating and maintaining Knowledge Bases. This document outlines how to use the multiple functionalities and features of the Knowledge Builder Application.

Copyright Notice

Intended Use

The RippleDown Software is a decision support system intended to support staff of healthcare organisations in the auditing and/or interpretation of patient data and in the generation of patient reports and/or workflow actions. The criteria it applies is determined by subject matter experts employed by individual clients and is specific to each client site. The software does not contain any pre-programmed clinical information or any inherent diagnostic functionality.

The RippleDown Expert module is intended to be used purely as a decision support system that provides complementary reports for patient data to qualified individuals based upon established rules set up by relevant trained customer domain experts. It is intended that the clinicians receiving reports based on patient results will consider the report in conjunction with all current and previous patient history, and apply their own judgement when determining patient management. It is intended that the clinician does not rely solely on the existence of a report from RippleDown for the management of a patient.

> ℹ It is the responsibility of the licensee to use the product in accordance with its intended use. In support of this, it is recommended domain experts include a statement consistent with the following on all reports and include the basis for any recommendation made.
>
> “This report has been generated using clinical decision support software. This report is intended to provide adjunctive information only and should not replace clinical judgement.”

Disclaimer

Pacific Knowledge Systems Pty Ltd makes no warranties, either express or implied, regarding these computer software packages, or their fitness for any particular purpose other than warranty provisions embodied in any agreement or purchase contract.

Acknowledgments

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Manufactured By

Pacific Knowledge Systems: http://support.pks.com.au/product-register/

RippleDown is distributed by Abbott Laboratories as AlinIQ CDS

RippleDown is distributed by Philips Healthcare as LABOSYS CDS

Table of Contents

Glossary

Attribute
Name representing each element or data item of a patient record
Primary attribute: sent from the Online Information System
Derived attribute: created from primary attributes as intermediate characteristic

Auto-validation
Release of cases from RippleDown without manual review. The degree and level of auto-validation can be set by rules in the Knowledge Base.

Case
Set of data for a patient that will be interpreted by a knowledge base

Cornerstone
A case against which a rule has been built. These are stored in the project as references for future rules

Comments
The output of the project and shown in the Validator screen.

Condition
A logical statement of either true or false used in creating rules e.g. glucose >=5.5 and can refer to current and or previous cases

Episode
The set of results for a patient at a particular time on a particular day

Expert
Medical staff member directing the objectives and outcomes of the project

Feature
Higher level characteristic of a case built using rules

Knowledge Base
This is the project or database containing the rules used for case interpretation for each clinical or operational domain

Knowledge Builder
Application used to create the rules and maintain the Clinical Knowledge Base.

Objective
Used for project documentation

Project Explorer
Documentation of the project showing the history of the knowledge base.

Queues
A list of cases pending review
Special purpose queue: may be set up to hold a particular type of case
Copy queue: created to allow independent review of cases

Report
The output of a RippleDown Knowledge Base i.e. the set of comments given for a case

Rule
A set of conditions to create an action, for example to add, delete or replace comments for a particular type of case

Rule Wizard
The set of steps to guide the rule building process.

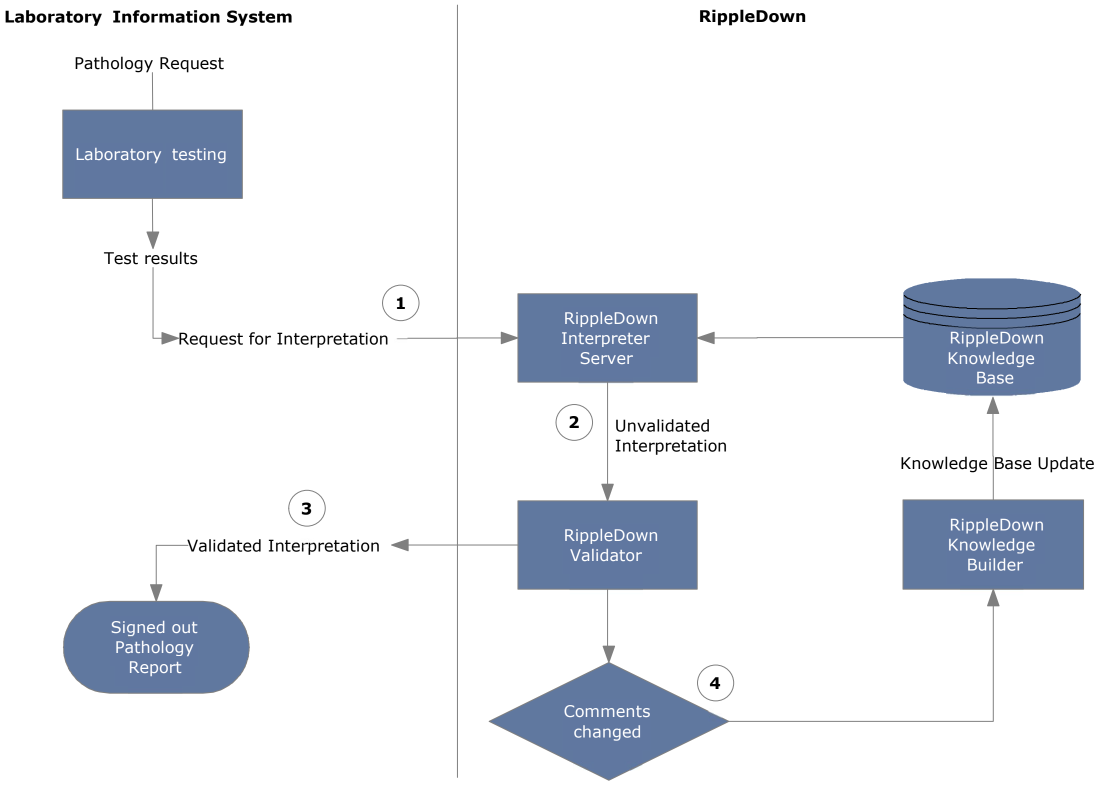
Validator
Application used to review reports prior to release.

Overview

Workflow

The Knowledge Builder is the component of RippleDown that provides the user with the facilities for creating and maintaining the Knowledge Bases required for the various clinical domains.

Once it is deployed on-line by the RippleDown Server, the primary purpose of each Knowledge Base is to generate an interpretation on receipt of a case from the Online Information System.


Laboratory Information System
RippleDown
Pathology Request
Laboratory testing
Test results
1
Request for Interpretation
RippleDown Interpreter Server
RippleDown Knowledge Base
2
Unvalidated Interpretation
Knowledge Base Update
3
Validated Interpretation
RippleDown Validator
RippleDown Knowledge Builder
Signed out Pathology Report
4
Comments changed

Workflow:

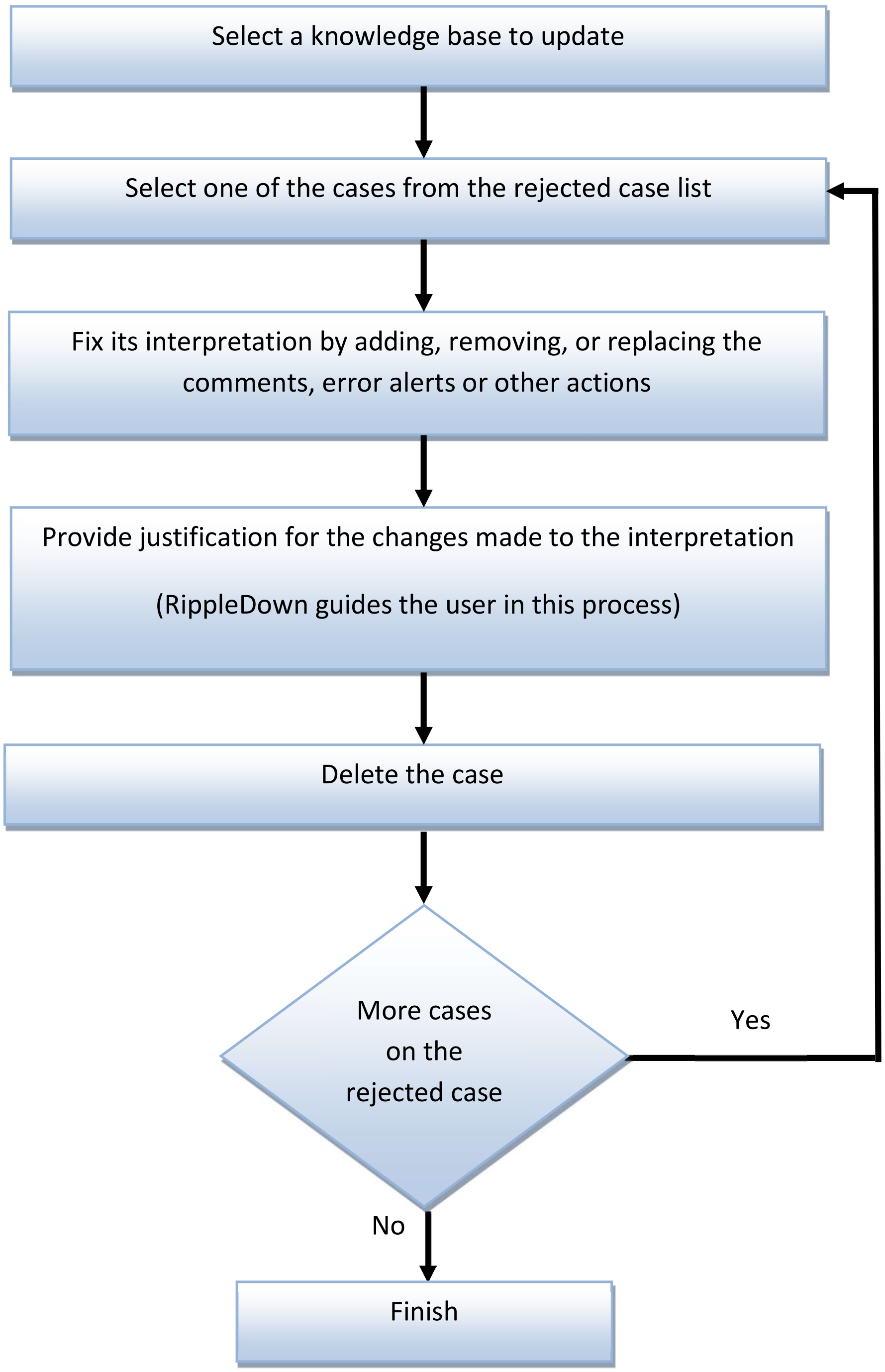
1. Laboratory case and result sent to RippleDown server.
2. Case is interpreted and comments added according to Knowledge Base rules.
3. Case authorised and the validated interpretation sent to the Online Information System.
4. If comments are changed in the Validator, the case is sent to the Knowledge Builder rejected list. Rules can be created to update the RippleDown Knowledge Base.

Knowledge Base Organisation

RippleDown organises the rules into a structure using its RippleDown technology to maintain consistency of the Knowledge Base. The user of the Knowledge Builder can see all the conditions that were used to give that report.

Knowledge Builder Tasks

Once the Knowledge Base is on-line the basic task for the Knowledge Builder is shown below:


Select a knowledge base to update
Select one of the cases from the rejected case list
Fix its interpretation by adding, removing, or replacing the comments, error alerts or other actions
Provide justification for the changes made to the interpretation
(RippleDown guides the user in this process)
Delete the case
More cases on the rejected case
Yes
No
Finish

Projects

What is a Project?

A project contains all the data (test results, episode and patient information, comments and rules) needed to make interpretive diagnostic conclusions for a pathology domain.

A project is also known as a knowledge base.

The purpose of a clinical knowledge base is to create an output in the form of an interpretive report, workflow comment, alert or combination of these. This output is displayed in the comments pane of Case View.

Starting a New Project

To start a new project, 3 user types need to be involved.

The **RippleDown Administrator** to configure the Online Information System to:

- send cases to RippleDown consisting of current and historical patient data
- receive reports, alerts etc. from RippleDown
- to create the project in RippleDown
- to ensure the users have access to the project

The **project expert** to determine which data elements are needed in the project. These should include:

- Patient demographics
- Episode details
- Test codes and results
- History files

The **rule builder** to create the Knowledge Base project in RippleDown.

Planning

Before building your project it is imperative to plan the main objectives, the attributes required from the Online Information System and the outputs expected from the knowledge base. Also required are an appropriate number of report sections to make the validating process as easy as possible.

Project Components

Objectives

Objectives are used for project documentation. These can be created during rule building, but for new projects it might be useful to create them first. Go to Edit | Objectives...

Report Sections

Report sections need to be created before comments are written. These sections may be used to group together similar comment types. Go to Edit | Report Sections... to build your sections.

Queues

A default queue is automatically created for each project. Rules can be built to create special purpose review queues for particular cases or conditions. Go to Edit | Queues... to build your queues.

Attributes

A case in RippleDown consists of a set of **attributes**, and for each attribute a sequence of samples which are the values for those attributes for current and historical samples. There are several different types of attributes:

Primary attributes represent the "raw" data sent to RippleDown by the Online Information System.

Derived attributes can be created from the primary attributes to capture examples of key concepts e.g. a risk ratio

These attributes can be built in the Attribute Editor located in Edit | Attributes.... Examples are:

- **lists** refer to a primary attribute and indicate when the attribute is an element contained in a defined list, for example of a group of doctor codes.
- **double lists** refer to two primary attributes, and match samples from one list depending on the value in the other list, for example to determine whether a doctor code is valid for particular wards.
- **numeric range attributes** – refer to a primary attribute and are used to flag values falling within a defined range.
- **deltas** refer to a primary attribute and use a table to flag cases for which there has been a significantly abnormal result or a significant change in values since the last episode.
- **text attributes** are used for careful matching of samples for a primary attribute, such as clinical notes, against lists of known keywords, taking into account lists of exceptions.
- **merged attributes** combine two or more attributes received from the Online Information System, possibly using different units, into a single attribute

- **calculated values** are derived from combinations of other attributes using the condition functions.
- **group attributes** refer to a set of related primary attributes.
- **episode attributes** are used to make complex assertions about individual episodes in a case.

Features and Derived Features: Features are higher level concepts that are created by rules to make further rule building easier. Derived features are similar to features but these can be constructed using existing features.

Features are used when there may be many definitions of a concept e.g. to identify a diabetic patient

RippleDown Rules

A rule consists of one or more logical expression(s) or condition(s), which, if true for the case means that an output such as a report comment, an alert, or a workflow instruction is generated from the project.

The rule wizard is used in RippleDown Knowledge Builder to guide the user through the rule building steps.

Rule Wizard

The rule wizard appears each time a rule is started and is used to guide the process step-by-step.

This facility allows the user to define a rule and document the reason for adding the rule and define the condition/s under which the rule will be applied. A reference case, called a Cornerstone, is stored for each rule in the knowledge base. All subsequent rules are checked against these cornerstones to maintain integrity of the knowledge base.

The user is able to undo the last rule that was added. This action may be repeated, if required, for all rules in the Knowledge Base.

Any changes made to the Knowledge Base, will take effect immediately and for subsequent interpretations made by the RippleDown Server.

Conditions

The user is able to create conditions using near-natural language syntax. Conditions may refer to:

- The current episode or previous episodes
- A specified time period or date
- Trends in a test value
- Comparisons between values
- Previous or current reference ranges
- Textual values in clinical notes or other fields

The full condition syntax is provided in the RippleDown Help utility.

ALL conditions in an expression must be met before a rule is ‘true’ (or has passed).

Logging In

Three modules of RippleDown are available from the login screen. The Laboratory Systems module is required for the clinical applications of Validator and Knowledge Builder.

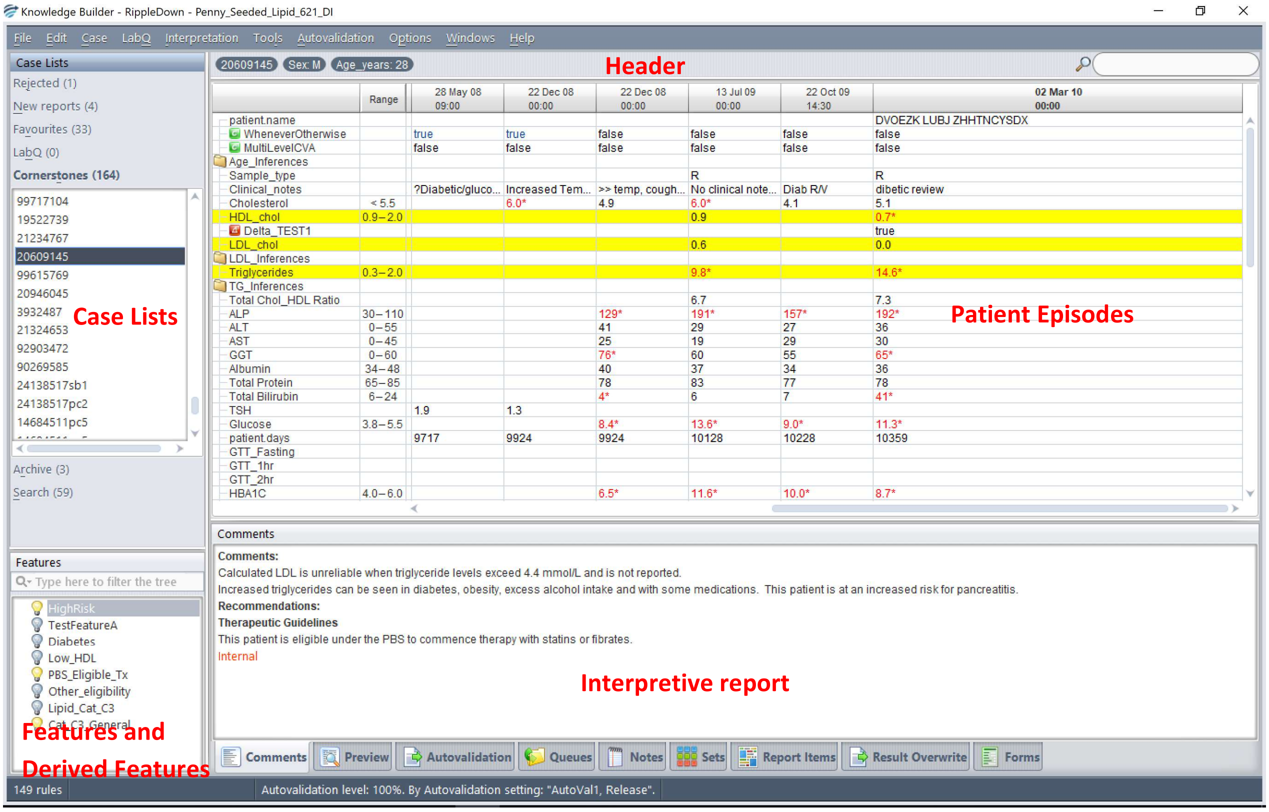
To login to the Knowledge Builder:

1. Open the application.
2. From the log in screen ensure the Knowledge Builder module is selected:
3. Enter your username and password and click OK. (Please contact your RippleDown Administrator to create these)
4. From the File menu use “Open Project” to select the relevant project.
5. The main case view opens.
6. To change your password, select “Reset Password” from the help menu and follow the prompts.

Case View

Case View provides facilities for the user to view all aspects of the currently selected case including:

- The demographic data
- The current and previous episodes
- The current interpretation of the case
- The rules associated with each of the comments given in the interpretation
- Any notes added to the case by the person who reviewed the case in the Validator
- Any derived attributes or features that have been defined for the case.



	Range	28 May 08 09:00	22 Dec 08 00:00	22 Dec 08 00:00	13 Jul 09 00:00	22 Oct 09 14:30	02 Mar 10 00:00
patient.name							DVOEZK LUBJ ZHHTNCYSDX
WheneverOtherwise		true	true	false	false	false	false
MultiLevelCVA		false	false	false	false	false	false
Age_Inferences							
Sample_type					R		R
Clinical_notes		?Diabetic/gluco...	Increased Tem...	>> temp, cough...	No clinical note...	Diab R/V	dibetic review
Cholesterol	< 5.5		6.0*	4.9	6.0*	4.1	5.1
HDL_chol	0.9–2.0				0.9		0.7*
Delta_TEST1							true
LDL_chol					0.6		0.0
LDL_Inferences							
Triglycerides	0.3–2.0				9.8*		14.6*
TG_Inferences							
Total Chol_HDL Ratio					6.7		7.3
ALP	30–110			129*	191*	157*	192*
ALT	0–55			41	29	27	36
AST	0–45			25	19	29	30
GGT	0–60			76*	60	55	65*
Albumin	34–48			40	37	34	36
Total Protein	65–85			78	83	77	78
Total Bilirubin	6–24			4*	6	7	41*
TSH		1.9	1.3				
Glucose	3.8–5.5			8.4*	13.6*	9.0*	11.3*
patient.days		9717	9924	9924	10128	10228	10359
GTT_Fasting							
GTT_1hr							
GTT_2hr							
HBA1C	4.0–6.0			6.5*	11.6*	10.0*	8.7*

Comments:
Calculated LDL is unreliable when triglyceride levels exceed 4.4 mmol/L and is not reported.
Increased triglycerides can be seen in diabetes, obesity, excess alcohol intake and with some medications. This patient is at an increased risk for pancreatitis.
Recommendations:
Therapeutic Guidelines
This patient is eligible under the PBS to commence therapy with statins or fibrates.
Internal

Autovalidation level: 100%. By Autovalidation setting: "AutoVal1, Release".

The Knowledge Builder provides the following facilities to enable case management:

- Finding and selecting a case
- Copying a case to the Favourites list
- Adding a new case
- Adding notes to a case for later reference
- Allowing a report to be auto-validated
- Renaming a case
- Editing the sample values within an existing case (to provide variants of the case that can be used for further rule building – only applicable to the Favourites List)

Case View Toolbar

File	Used to open and close projects
Edit	Provides access to Editors for Constants, Attributes, Objectives, Comments, Queues and Report Sections
Case	Used to add/delete/copy and search for cases
Interpretation	Used for opening the rule wizard to add Comments, Features, Derived Features, Auto-validation and Queues. Also used to undo the last rule.
Tools	Provides access to the Project Explorer, Audit Trail, Scratch pad and Project reports
Auto-validation	Used for managing Auto-validation and viewing Comment statistics
Options	Used to customise font type, size and format, set project settings such as case list properties and show/hide removed comments
Windows	Describes keyboard shortcuts for navigating through the CaseView screen. Can also be used to hide panes on the main screen
Help	Used to access the RippleDown online help facility or change passwords

Case Lists

Several case lists are found in the Knowledge Builder:

Case List	Functionality
Rejected	These are the cases in which a change has been made to the RippleDown interpretations changed during validation in either RippleDown Validator or in the Online Information System. Rejected cases are the ones that you are likely to use when building rules for these interpretations. Only the most recent 1,000 rejected cases will be kept on this list. You can delete a case from the Rejected list once you have dealt with it, e.g. by building rules to fix its report. You cannot edit a case on this list. If you wish to edit a case on this list, copy it to Favourites and then edit it on the Favourites list.
New Reports	Cases which have reports that you have not yet allowed to be auto-validated. You can allow a specific report for a case on this list using the **Allow** button on the Preview window. Only the most recent 1,000 cases will be kept on the New Reports list.
Favourites	This list is manually created by the Knowledge Builder by copying cases from one of the other lists to the Favourites list. Once a case is on the Favourites list it may be kept for demonstration or training purposes and it can be edited for rule building and testing purposes. There is no limit to the number of cases on the Favourites list.
Cornerstones	These are the cases that have been used to build rules, and so will normally provide the best examples of each type of interpretation given by the Knowledge Base. You cannot edit or delete a case from this list. You can however copy a case to the Favourites. There is no limit to the number of cases on the Cornerstone list.
Archive	Stores 10,000 cases following interpretation by the RippleDown server. When a case is deleted from any of the other lists, it is stored in the Archive, where it can still be accessed if required. If a case is deleted from the Archive, it is permanently deleted from the project but may be retrieved from a backed up zip file stored by the server.
Search	When a search by condition is performed on one of the other lists, the cases found are copied into the search list

Case View

The first column displays the attribute / derived attribute / feature / derived feature name.

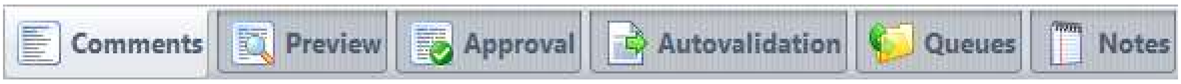
The next column displays the **reference ranges** (if any exist) for the analyte attribute.

These ranges are not set-up in RippleDown, but are sent from the Online Information System with the case, and therefore only relate to the current episode.

Columns from there onwards are for the episodes. Each episode has its own column, with results in the appropriate rows. The current episode column is the one farthest from the reference column.

Information / Action Panel

The panel below the results has 6 tabs referring to screen panes:



Comments	The pane used to add comments
Preview	Displays the report as presented in the Validator and the Online Information System
Approval	Displays the report as approved by the RippleDown Validator and any differences between the original and approved reports.
Auto-validation	Pane used to define auto-validation settings
Queues	Pane used to define a special purpose queue
Notes	Displays any notes added to the case by either the Validator or a Knowledge Builder

Attributes

Attributes are data items / elements for patient, episode, and test information, e.g. Given Name, DOB, Sodium, Chloride, etc.

Derived attributes are user defined manipulations of raw data (primary attribute results).

Primary Attributes

These attributes represent the “raw” data items sent to RippleDown from the Online Information System. They appear in the Attribute Editor with an external and internal name.

The external name refers to the unique Online Information System codes. The internal name is initially identical to the external name but it is able to be customised in the Attribute Editor in RippleDown to make it user-friendly. It is this internal name that will be displayed in RippleDown Knowledge Builder and Validator case views.

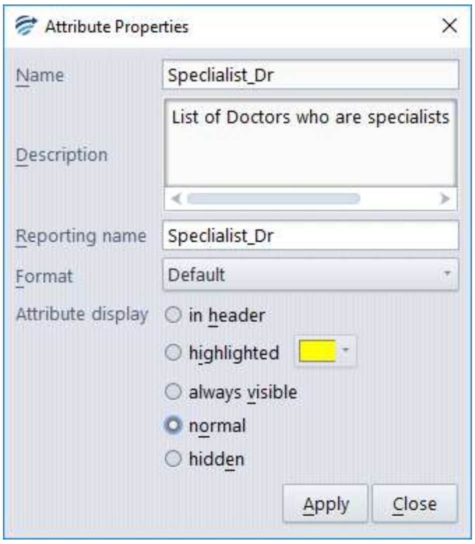
Attribute Editor

Managing and manipulating attributes is done using the Attribute Editor, which can be found in Edit | Attributes....

You can search for attributes within the Attribute Editor using the search bar. This search is not case-sensitive and will filter the attribute list for matches to your search

To move the attributes to a new position in the list click on the attribute and drag it to the preferred location. Multiple attributes can be moved by using CTRL-Click or SHIFT-Click. The user can also select from the right click menu to “move to beginning” or “move to end” of the attribute list.

Other case view properties of existing attributes can be customised. Right click on the attribute and select Attribute Display:



Attribute Properties

Name: Speclialist_Dr

Description: List of Doctors who are specialists

Reporting name: Speclialist_Dr

Format: Default

Attribute display:
○ in header
○ highlighted 🟨
○ always visible
● normal
○ hidden

Function	Description	
Name	Enter the name to be displayed in Case View. Do not use spaces. Words can be separated by an underscore.	
Description	Enter a description if required for the Attribute.	
Attribute display	Select one of the options:	
	in header	To position the attribute and current value in the Case Header
	highlighted	To add a highlight colour to the row containing the attribute. The colour can be customised in the Attribute Editor | Options | Highlight colour
	always visible	The attribute is always shown
	normal	The attribute and corresponding value at each episode are shown unless the sample value is empty. This the default.
	hidden	The attribute and sample values are not shown but may be used in creating rules
Format	Default	The attribute will not show in the body of a case if all of the samples for the attribute for that case are 'uninteresting'. For a primary attribute, a calculated value, delta, a double list or a merged attribute, 'uninteresting' means blank. For ranges, lists, text attributes and episode attributes, 'uninteresting' means having the value 'false'.
	No formatting	The attribute value will appear as presented by the Online Information System.
	0, 1 or 2 Decimal places	Attribute sample can be formatted to either 0, 1 or 2 decimal places. This affects the display. Calculations using the attribute value will be performed on the data as it received from the Online Information System.
	1,2,3 or 4 significant figures	Attribute sample can be formatted to either 1,2,3,4 significant figures. This affects the display. Calculations using the attribute value will be performed on the data as it received from the Online Information System.

Derived Attributes

The Knowledge Builder provides the user with the facilities to define “derived” attributes. These are user-defined attributes developed for each particular project.

These derived attributes and their values can be used for rule building in exactly the same way as primary attributes. RippleDown will evaluate each derived attribute for each episode in the case.

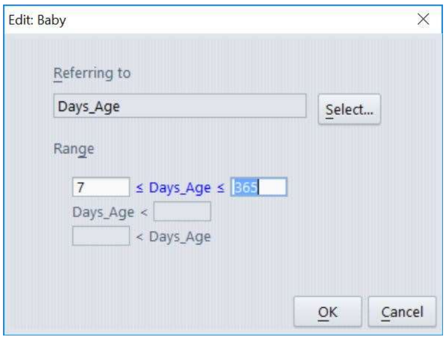
Attributes cannot be deleted. To manage unwanted attributes, right click on the attribute in the Attribute editor, rename the attribute to, for example “DoNotUse”, and move it to the bottom of the list.

The derived attributes available are:

Numeric Range

A numeric range gives a true or false value relative to a sub-range of values for the selected numerical attribute, e.g. a high, low or normal range

For example: The attribute Baby is defined as a patient with an age in days between 7 and 365 days:



Edit: Baby

Referring to: Days_Age

Range

7 ≤ Days_Age ≤ 365

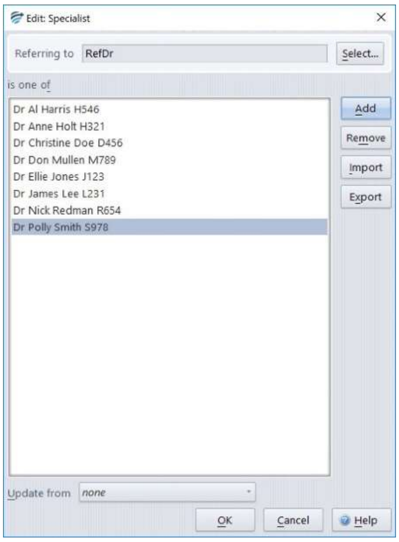
Days_Age <

< Days_Age

List Range

A list range is used to detect episodes in which the value for a primary attribute matches one of a list of values. A list attribute will return a *true* if the primary attribute, on which it is based, contains one of the entries from a discrete list.

An example might be a list of known endocrinologists. The Endocrinologist list attribute will be true if the referring doctor for the episode appears in the list, see below:



Referring to RefDr

is one of

Dr Al Harris H546
Dr Anne Holt H321
Dr Christine Doe D456
Dr Don Mullen M789
Dr Ellie Jones J123
Dr James Lee L231
Dr Nick Redman R654
Dr Polly Smith S978

Update from none

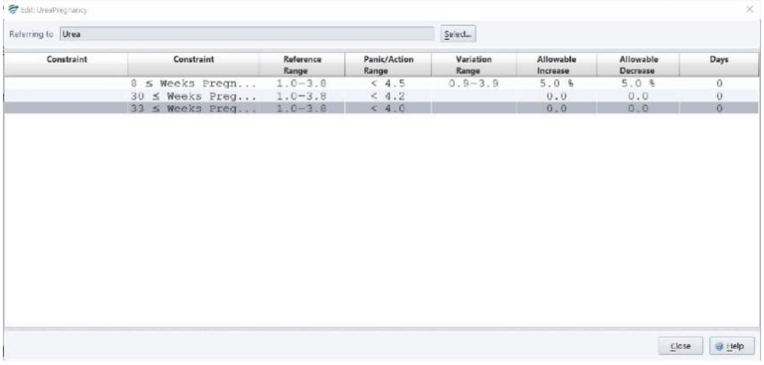
From the Case View it is possible to add elements to the list. To do this right click on the sample, select Add value to ... and select the appropriate list.

Panic / Delta Check

A panic / delta check refers to a detailed set of numeric ranges defined in a tabular fashion for that test. These ranges may be separately defined according patient age, sex or other demographic, and for sub-ranges of the test values.

The alert is generated if the current value is outside some defined range, or alternatively, if the value has increased or decreased from its previous value by an amount greater than what is allowed.

A Panic Delta will give a value of "true" if the alert has been triggered for the **current** case. If the Panic Delta is not triggered for the case, it simply will not show.



Referring to Urea

Constraint	Constraint	Reference Range	Panic/Action Range	Variation Range	Allowable Increase	Allowable Decrease	Days
	8 ≤ Weeks Pregn...	1.0-3.8	< 4.5	0.9-3.9	5.0 %	5.0 %	0
	30 ≤ Weeks Preg...	1.0-3.8	< 4.2		0.0	0.0	0
	33 ≤ Weeks Preg...	1.0-3.8	< 4.0		0.0	0.0	0

Each row of the Panic Delta contains the following:

Constraints

- These are criteria that must be satisfied for the panic delta processing to continue. Examples are, age > 50, sex is "F".
- Up to 2 constraints can be defined per row.
- To edit or add a constraint, select a constraint cell and use the right-click menu "Edit Constraint".
- The constraints can refer to primary or derived attributes of the case, and often refer to demographics such as age or sex, rather than the test attribute itself.
- All other ranges in the Panic Delta refer to the test attribute.
- If you leave a constraint empty, then it is assumed to be satisfied.
- If both constraints are satisfied by the current case, then the panic delta processing will continue, looking next at the Reference Range.

Reference range

- The Panic Delta reference range is something you define for the test attribute - it may or may not correspond to the reference range supplied by the Online Information System as part of the case.

- If there is only 1 episode in the case, then the panic delta will trigger if the attribute is outside the reference range.
- You should not leave the reference range blank, otherwise all single episode cases will trigger the panic delta.
- To edit or add a reference range, select the reference range cell and use the right-click menu "Edit reference range"
- If there are 2 or more episodes in the case, and they are sufficiently close in time, then the reference range is ignored and the panic / action range is used.

Panic / Action range

- Used if there are 2 or more episodes in the case, and the time between the current test value and the previous test value is less than the specified number of days, then the panic delta will trigger if the attribute is outside the panic / action range.
- This enables you to define a wider panic / action range than reference range, for those situations where you are closely monitoring the trend of a test value.
- To edit or add a panic / action range, select the panic / action cell and use the right-click menu "Edit panic/action range"
- If the test value is within the panic / action range (and we have the situation of the 2 episodes close in time), then the variation range and allowable delta processing is done.

Delta Check (variation range, allowable increase, allowable decrease, units, days)

- A variation range defines the range of test values for which the allowable increase or decrease is checked.
- This allows you to define different allowable increases or decreases depending on the absolute value of the test attribute.
- To edit any of the delta check fields, select a delta check cell and use the right-click menu "Edit delta check fields"
- By default, each allowable increase or decrease will be interpreted as being in absolute units. However, you can change this to percentage.
- To add several variation ranges, each with a different allowable increase or decrease, use the right-click menu "Copy this row", then change just the delta check fields for the new row.

Multiple rows can be defined if you require different reference ranges, panic / action ranges, or delta checks for different constraints, e.g. a new set of ranges if the patient is female, or over a certain age. To do this, use the right-click menu "Add a row", then edit the constraints and ranges for the new row.

Rows in the Panic Delta can be re-ordered using the right-click menus "Move up", "Move down". They can also be removed if required.

When you have finished defining the Panic Delta, click the [Close] button. The new Panic Delta will appear as an attribute in all cases where it triggers.

Once the panic / delta ranges have been defined, the logic to trigger the check is as follows:

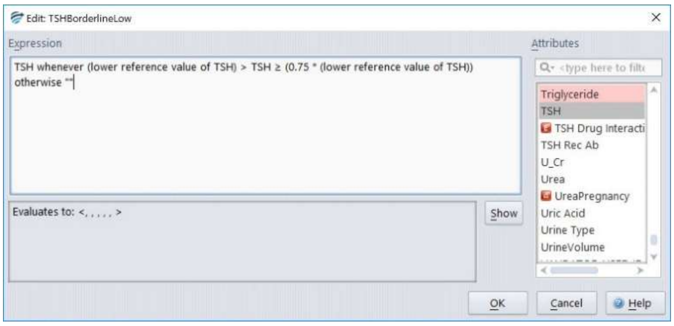
1. If there is only 1 episode for the case (i.e. the first presentation of this patient), the check is triggered if the test value is outside the reference range.
2. If there is more than 1 episode, and the previous episode is within the specified time period, the check is triggered if the test value is outside the panic range.
3. If the test value is inside the panic range, then the check is triggered if the change in the test value from the previous to the current episode is outside the allowable increase or decrease (specified either in absolute units or percentages)

Rules may be defined to take the appropriate action according to the value of a panic / delta check. For example, rules may be added to ensure that if a check is triggered, the case is never auto-validated. When the case is shown to the user in Validation, the reason why the check was triggered is shown.

Calculated Value

A calculated value refers to a formula involving one or more attributes, e.g.

A determination of borderline low TSH might appear as:



Edit: TSHBorderlineLow

Expression

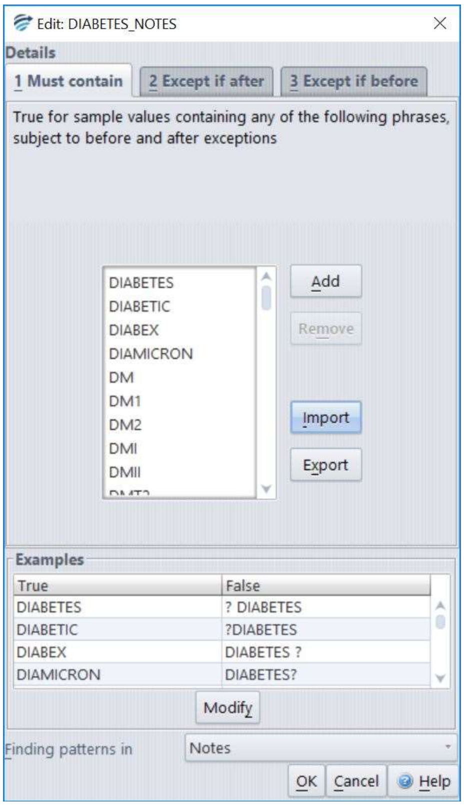
```
TSH whenever (lower reference value of TSH) > TSH ≥ (0.75 * (lower reference value of TSH))
otherwise ""
```

Evaluates to: <, , , , , >

The formula can involve other calculated values as long as circular references are avoided i.e. A calculated attribute A can refer to calculated value attribute B as long as B does not already refer to A.

Text Attribute

A text attribute is used to refer to the presence of a word or phrase in an attribute that has textual values, e.g. looking for “diabetic” in a patient’s clinical notes. Facilities are provided to refine the search so as to exclude similar but unwanted words or phrases, e.g. diabetic.



True for sample values containing any of the following phrases, subject to before and after exceptions

DIABETES
DIABETIC
DIABEX
DIAMICRON
DM
DM1
DM2
DMI
DMII

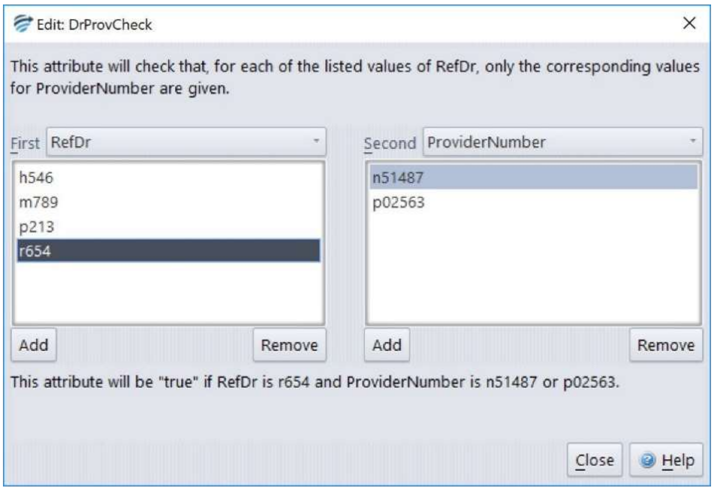
Examples

True	False
DIABETES	? DIABETES
DIABETIC	?DIABETES
DIABEX	DIABETES ?
DIAMICRON	DIABETES?

Finding patterns in: Notes

Double Lists

Double Lists are used to relate values of two primary attributes, where for certain values of one attribute, only a limited list of values of the second attribute are allowed.



Edit: DrProvCheck

This attribute will check that, for each of the listed values of RefDr, only the corresponding values for ProviderNumber are given.

First: RefDr

- h546
- m789
- p213
- r654

Second: ProviderNumber

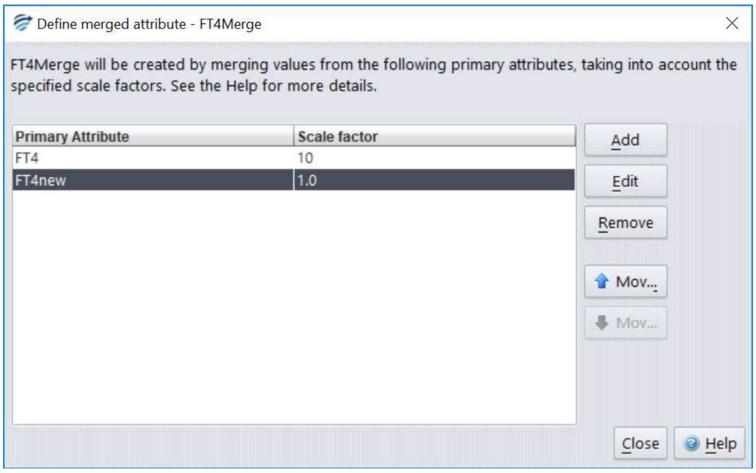
- n51487
- p02563

This attribute will be "true" if RefDr is r654 and ProviderNumber is n51487 or p02563.

In the above example if the referring Dr is R654 and the provider number is N51487 or P02563, the attribute will evaluate as true. If the referring Dr is R654 but the provider number is not N51487 or P02563 then the double list will evaluate to false. If the referring Dr is none of the values listed in the first column than the double list will provide no evaluation.

Merged Attribute

A merged attribute is one that takes its value from two or more primary attributes. For example if a test is superseded in the Online Information System, the old test name may appear in historical episode and the new name in the current episode. A merged attribute combines the two tests into a single attribute for use in the project. When defining a merged attribute, each test may be provided with a scaling factor to take into account changes in units between superseded and new test.



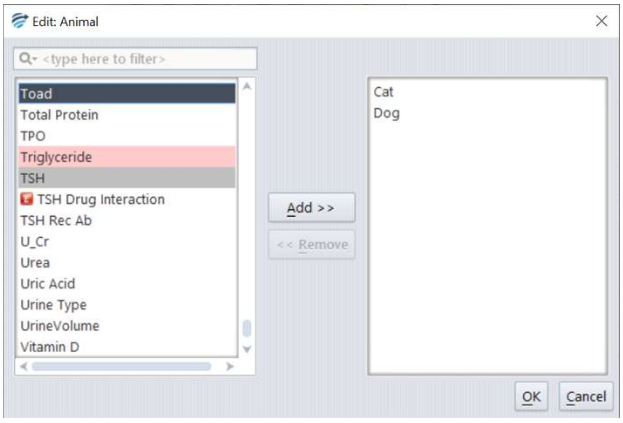
Define merged attribute - FT4Merge

FT4Merge will be created by merging values from the following primary attributes, taking into account the specified scale factors. See the Help for more details.

Primary Attribute	Scale factor
FT4	10
FT4new	1.0

Group Attribute

A group attribute refers to a set of related primary attributes. They can be useful in the situation where the project deals with possibly very many primary attributes, but where the comments and rules can be more simply expressed in terms of groups rather than the individual attributes.



Edit: Animal

<type here to filter>

Toad
Total Protein
TPO
Triglyceride
TSH
TSH Drug Interaction
TSH Rec Ab
U_Cr
Urea
Uric Acid
Urine Type
UrineVolume
Vitamin D

Cat
Dog

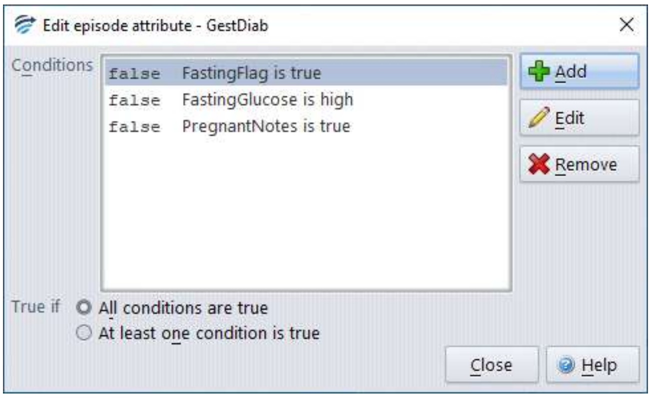
For example, we can define the following groups of allergen attributes:

Group	Attributes in Group
pollen	birch mugwort
food	milk peanut
animal	dog cat

Episode Attribute

Episode attributes are used to identify episodes in a case, at which all of a number of defined conditions apply. Along with primary attributes, the conditions can include other derived attributes. These are useful to identify clinical patterns. For example, suppose that we need to identify patients who have had gestational diabetes. One indication of this will be that there is an episode at which:

- the 'PregnantNotes' has value "true"
- the 'FastingFlag' attribute has value "Fasting"
- the 'FastingGlucose' value is elevated.



Conditions
false FastingFlag is true
false FastingGlucose is high
false PregnantNotes is true

True if ● All conditions are true
○ At least one condition is true

The episode attribute, “GestDiab” will be true for when all three conditions are met, for example:

	05 Feb 12	06 Oct 12	07 Nov 12
Pregnant	false	True	false
Glucose	4.1	7.8*	4.5
CollectionType	R	F	F
E GestDiab	False	True	False

Pattern attribute

Pattern Attribute is a sophisticated tool for the extraction of information from free text. Consider the task of analysing clinical notes to determine whether or not a patient is a diabetic. A typical clinical note might be:

```
family hist diabetes
```

There are really two main phrases in this note: “family hist” and “diabetes”. Together, these mean that a relative of the patient has diabetes, but the patient may or may not have the disease.

Now consider another clinical note:

```
fh diabetes. ? anaemia
```

The fullstop indicates that there are two independent pieces of information in this note. The first is that there is a family history of diabetes (“fh” is short for “family history”) and the second is that the patient might be anaemic.

These two examples demonstrate three complications of text analysis:

- Key terms such as ‘diabetes’ have many synonyms and are often mis-spelled.

- The meaning of a key term in a block of text depends on what other key terms are nearby.
- There are generally several important pieces of information contained within a piece of text.
- Pattern Attributes address these issues with a three-phase analysis of text.
- The text is converted to a list of keywords.
- The converted text is searched for meaningful patterns.
- Attributes are put into the case to represent these patterns.

Pattern Attribute Keywords

Each pattern attribute has a set of keywords, and each keyword has a name and a collection of matching phrases:

Suppose that a pattern attribute has keywords as follows:

Keyword	Matching Phrase
FH	fh, family history, f hist
DB	db, diabetes, dm, t2dm
ANAEMIA	anemia, anaemia

Here are some clinical notes converted to lists of these keywords:

Clinical note	Converted
f hist anemia and t2dm, unwell, tired	FH ANAEMIA DB
overweight, family history diabetes	FH DB

When a pattern attribute appears in a case, its value is that of the associated attribute converted to a list of keywords. This is done for every episode. For example, here is a pattern attribute called “NotesSummary” that analyses values of “ClinicalNotes”:

It is possible to write rule conditions that directly compare a required string with these values. For example, for this case we could use a condition

```
NotesSummary contains "ANAEMIA"
```

To add a comment about the patient having a known disease.

However, this is *not* the recommended approach to using pattern attributes. One reason for not using keywords directly in a condition is that we may need to change the name of a keyword, and if we do, rules that rely on the presence or absence of the original keyword name will fail.

Keywords are regular expressions

A regular expression is a string of text in which certain characters take on a special meaning, so that the expression itself represents a whole class of texts. A familiar example is that "*.jpg" represents any file with the extension ".jpg".

In Pattern Attributes, the matching phrase of keywords are interpreted by RippleDown as regular expressions. This is mostly of use in analysing lengthy documents. For the analysis of clinical notes and other short texts, regular expressions can pretty much be ignored.

Matching punctuation

Because a matching phrase for a keyword is a regular expression, care must be taken when adding a matching phrase for punctuation such as a full-stop (period). These characters need to be 'escaped' with a backslash symbol. Another character that must be escaped is the question mark.

Derived Pattern Attributes

For each pattern attribute you can define one or more derived pattern attributes. Each derived pattern attribute refers to a list of keywords indicating a useful concept for your Knowledge Base. You can define more than one list of keywords for the same derived pattern attribute.

The example above shows a Pattern Attribute with two Derived Pattern Attributes, called FH_DB and DB. Note that the Derived Pattern Attribute DB has the same name as one of the Keywords. This is not a problem; in fact, it is quite normal.

The Derived Pattern Attribute FH_DB represents the concept of 'family history of diabetes'. There are two patterns of keywords that represent this concept, 'FH DB' ("family history … diabetes") and 'DB FH' ("diabetes … family history")

Each derived pattern attribute has three parts: a name, a formula, and a list of keywords.

- The first column is the name of the attribute that will appear in each case, so that it is available to rules.
- The second column is the formula which determines how the value of the attribute is calculated.
- The third column is the pattern of keywords that this attribute is looking for.

Looking for patterns

If any of the patterns of keywords is found in a case, then the corresponding derived pattern attribute will have a value for that case.

In the example above, the derived pattern attribute ANAEMIA has value "true" for both episodes of the case, whereas DB AND FH_DB only have the value "true" for the previous episode.

Negating patterns

In the example above, the clinical note “fh diabetes, anaemia” gives ‘true’ values for the derived pattern FH_DB and also for the derived pattern DB. To deal with inconsistencies such as this, we use Negating Derived Attribute Patterns, or ‘Negations’ for short.

The evaluation of matches of Derived Attribute Patterns occurs in two steps. In the first, each Derived Pattern is matched, and those that match are recorded. In the second pass, each Derived Pattern found in the first pass is checked against its group of negations, and if any of these were found to be matches, the overall evaluation does not include the Derived Pattern.

For example, suppose that we add the Derived Pattern FH_DB to the list of negations for DB, then the final evaluation of the Pattern Attribute for the clinical note “fh diabetes, anaemia” has a match for FH_DB but not for DB:

Formulas

Derived pattern attributes are used to analyse many different kinds of text, from short clinical notes to long paragraphs found in legal documents. Sometimes, it is necessary to extract a numerical value or a date from a block of text. This is achieved with derived pattern formulas.

However, for most applications, the most commonly used formula is “?”, which tells RippleDown to put the value “true” for a derived pattern attribute when a pattern is found, as in the example above.

Patterns of Keywords

When adding a pattern of keywords to a Derived Pattern, we need to specify a string of pattern keywords to be matched. Additionally, there is a place for an example of text that matches the pattern of keywords.

If the example provided does not match with the pattern of keywords, this is pointed out, and the changes to the pattern will not be able to be saved. You will either need to change the pattern or change the example. The conversion of the example text to keywords is shown, so that the problem can be found.

Pattern Attribute Maintenance

It is sometimes necessary to add a new keyword to take into account different information in a newly encountered piece of text. This can create problems because the new keyword can appear in between existing ones, and so a derived pattern attribute may no longer find this new pattern of keywords.

For example, suppose that we decide that “close” is an important concept when applied to a family member, and needs to be a new keyword, e.g “CL”. A note such as anaemia, diabetes in close family, overweight” that previously corresponded to keywords “ANAEMIA DM FH” now becomes “ANAEMIA DM CL FH”. A derived pattern attribute FH_DB looking for “DM FH” would now not find it.

To help find these problems, the example text of each pattern of keywords is checked each time keywords are edited. If the derived pattern attribute no longer finds its pattern in the example text, the pattern is shown in red:

The pattern can then be edited to take into account the changed list of keywords. In the example above, you would add a new pattern “DM CL FH” for the derived pattern attribute FH_DB. As well, to correct the now-red pattern “DM FH”, you would need to change its example from “diabetes in close family” to “diabetes in family”.

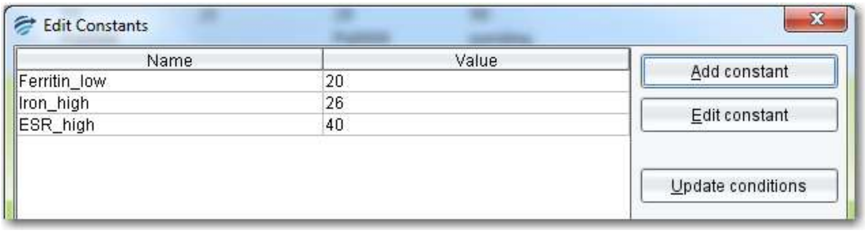
Constants

Numerical or textual values used in conditions can be defined as constants. For example, in a glucose Knowledge Base you might define a constant GLUCOSE_HIGH = 5.5. Then, instead of "hard-coding" numbers into your rules like "glucose > 5.5" you can use the constants, e.g. "glucose > GLUCOSE_HIGH".

The advantages of using constants rather than hard-coded numbers are:

- A constant such as GLUCOSE_HIGH is more understandable than the number "5.5", especially for someone who is exploring the Knowledge Base, and
- Once you have defined a constant, you can change its value at a later stage if required. This means that any conditions that use that constant will not have to be changed, saving you a lot of time.

Constants are defined and modified using the Constants Editor Edit | Constants. Each constant consists of a name and a value. Multiple constants can be defined and will appear listed in a table:



Name	Value
Ferritin_low	20
Iron_high	26
ESR_high	40

The table can be sorted either by name or value simply by clicking on the heading.

To use constants:

Constants can be used by typing the constant name into expressions, for example:

Ferritin < Ferritin_Low

Note: The constant name must be typed and it cannot be selected from the tabs in the rule wizard nor from the attribute selector in the expression editor.

Updating an existing Knowledge Base to use constants

Once you have defined a set of constants, you can update the existing conditions in your knowledge base to use these constants. Use the 'Update conditions' button. The system will then find all the conditions that use one of the constant values you have defined. You will then be asked to confirm each condition before the replacement is made, as shown in the example above in which the value 20 will be replaced with the newly defined constant called Ferritin_Low.

Features and Derived Features

Features

Features are indicated by the grey light globe icon: 💡

Each feature represents a high-level concept that is defined by rules and may in turn be used in other rules. Features, as derived attributes, are not added to the interpretive report.

An example of a feature could be ‘Is_Diabetic’. This feature may be added to a case for many different reasons, e.g. *more than 3 Glucose > 11.0* or *clinical notes contains “diabetic”*. Once this feature has been defined, the interpretive comment “History of diabetes noted” can be added, with condition ‘Is_Diabetic’ is true”.

There are several advantages to using features:

- Conditions using features may be more intelligible than using more complex expressions involving attributes
- A feature is a shorthand way of expressing a pair (or more) of conditions that often are used together
- A feature is easy to find and re-use in different rules
- The same feature may be defined in more than one way, e.g. "IsDiabetic" could also be given if the GTT 2 Hr result was at a certain level

Derived Features

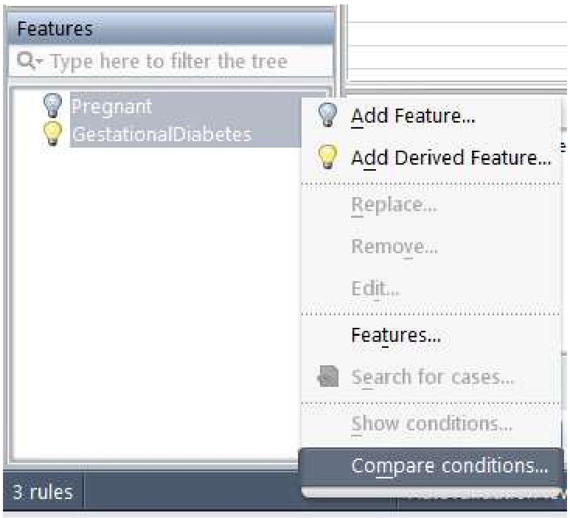
Derived features are indicated by a yellow light globe icon: 💡

Derived features are higher level features. The only difference from features is that derived features can be defined using features.

For example, you might have already defined three features XNormal, YNormal and ZNormal. There may be rules where you want to use these 3 features together. Rather than repeating these 3 features in conditions each time, you can then define a derived feature XYZNormal which is given if the other three are given, thus simplifying these rules.

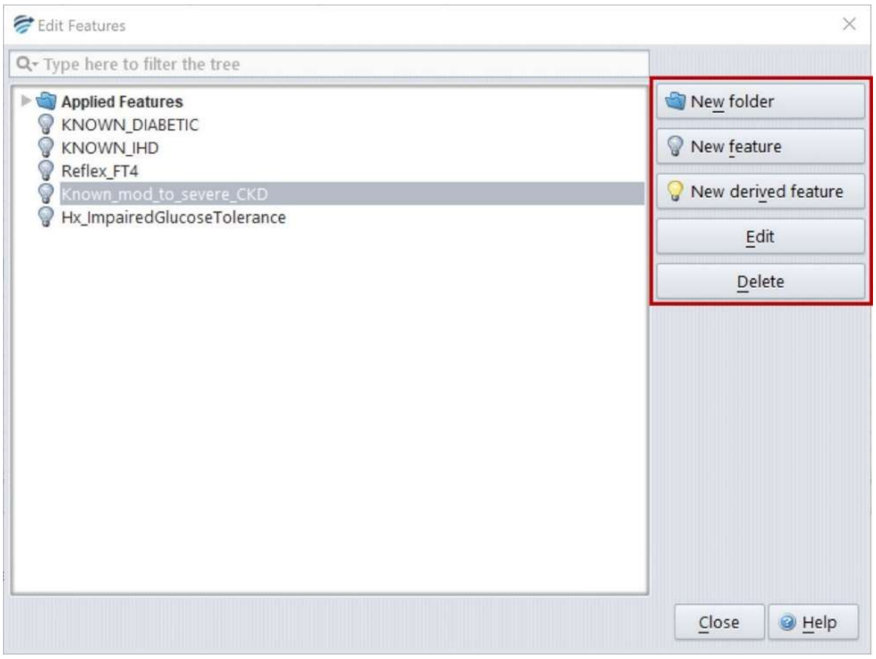
Compare conditions for Features and Derived Features

You can compare the conditions set for Features and Derived Features. Simply highlight the Features or Derived Features to be compared and right click, then select “Compare conditions…”.



Feature and Derived Feature Organisation

Features and Derived Features can be edited and organized into folders. This can be done through the Edit Features window. They can be reorganized into folders or re-ordered by using Drag and Drop.



Rules and the Rule Wizard

A rule is a set of conditions to create an action, for example adding or deleting a comment. RippleDown employs case-based reasoning and a rule can be only be built against the conditions presented in the current case showing in Case View.

RippleDown provides the Rule Wizard facility to guide the process of updating the knowledge base for the currently showing case. The stages of the Rule Wizard are

1. Define the rule action
2. Set an objective
3. Add conditions
4. Review cornerstone cases.

Rule Wizard - Define the Action

A variety of actions can be controlled by rules. These are

Function	Action
Comments	▪ Adding a comment ▪ Replacing the selected comment by another one. ▪ Removing the selected comment ▪ Enabling or disabling a report section
Features and Derived Features	▪ Adding a feature ▪ Replacing the selected feature by another one ▪ Removing the selected feature
Auto-Validation settings	▪ Adding an auto-validation setting ▪ Replacing the selected auto-validation setting by another one ▪ Removing the selected auto-validation setting
Review queues and Copy queues	▪ Adding a review/copy queue ▪ Replacing the selected review/copy queue by another one ▪ Removing the selected review/copy queue
Report items	▪ Enabling the selected report item ▪ Disabling the selected report item

To initiate an action use the right-click function in the required tab.

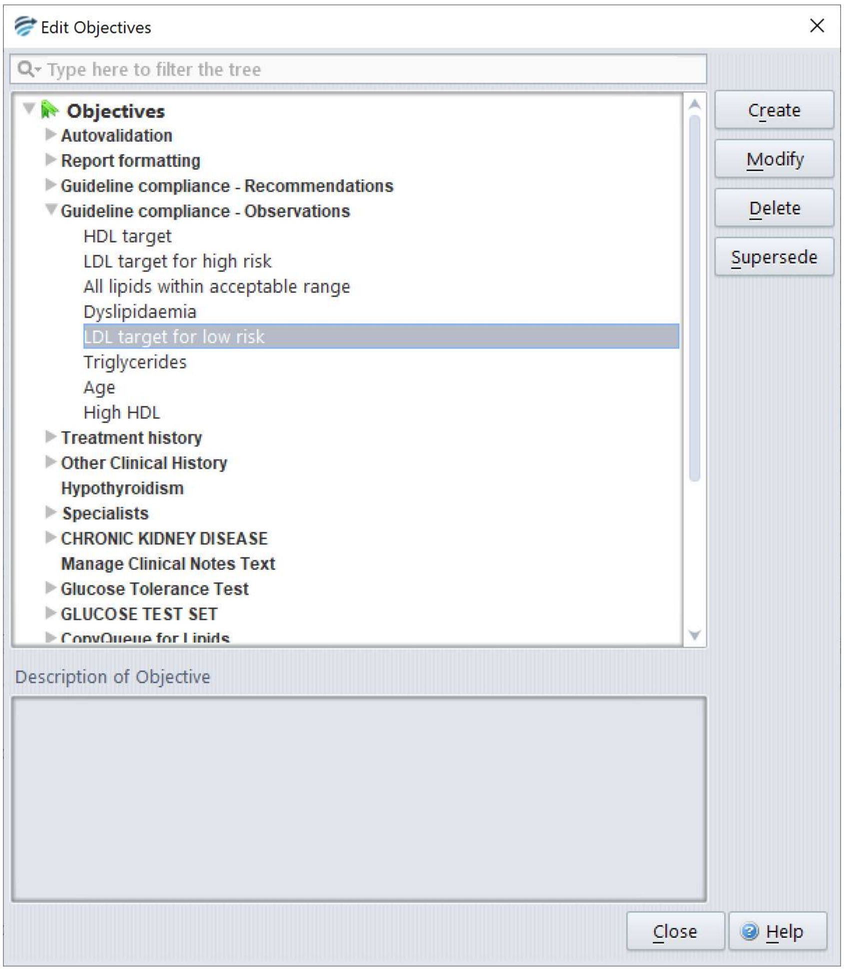
Rule Wizard - Set Objective

Objectives are used to document a Knowledge Base by adding to each rule an explanation of why the rule is being built. An Objective can be further refined into a set of sub-Objectives. For example, an Objective could consist of a local commenting policy within the laboratory, and a sub-Objective could be a specific aspect of that policy. Therefore, it is possible to have several levels of Objectives.

The Objective Editor (Edit|Objectives) is used to view and manage objectives. The available operations are to:

- Create new Objectives
- Modify existing Objectives
- Delete unused Objective
- Supersede one Objective with another
- Re-order or move an Objective under a new parent Objective using drag-and-drop

An extract of an example Objective Editor window is shown below.



Objectives
- Autovalidation
- Report formatting
- Guideline compliance - Recommendations
- Guideline compliance - Observations
  - HDL target
  - LDL target for high risk
  - All lipids within acceptable range
  - Dyslipidaemia
  - LDL target for low risk
  - Triglycerides
  - Age
  - High HDL
- Treatment history
- Other Clinical History
- Hypothyroidism
- Specialists
- CHRONIC KIDNEY DISEASE
- Manage Clinical Notes Text
- Glucose Tolerance Test
- GLUCOSE TEST SET
- CopyQueue for Lipids

Description of Objective

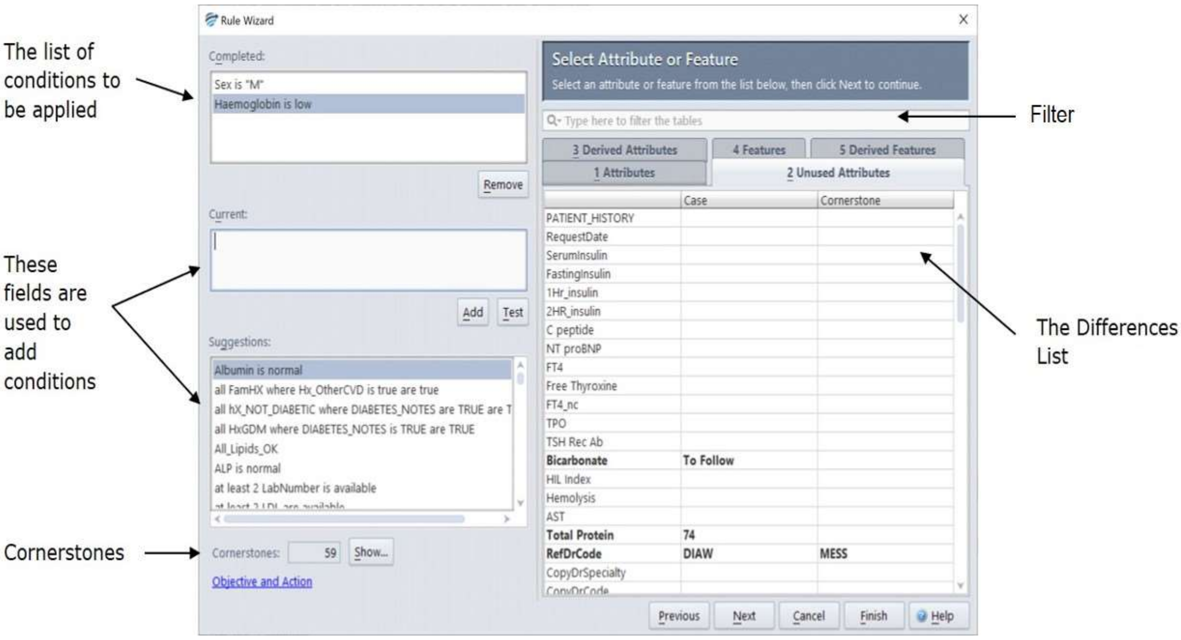
The very first item is the heading "**Objectives**". Underneath this heading are the top-level Objectives shown in **bold** font with lower level Objectives if applicable.

Objectives can be created before rules are built using the Objective Editor from the main screen. Alternatively, an Objective can be created and selected at the time of building a rule from the Select an Objective rule wizard step.

Objectives can be reorganised if required. To move the objective to a new position in the list click on the objective and drag it to the preferred location.

Rule Wizard - Add Conditions

Conditions are specifications that are used to justify the comments added in a rule. Each condition therefore evaluates to true or false for a case. Conditions are written in a near natural language syntax built up from various pre-defined functions. The following screen is used to add conditions:



Select Attribute or Feature

Select an attribute or feature from the list below, then click Next to continue.

Completed:

Sex is "M"
Haemoglobin is low

Suggestions:

Albumin is normal
all FamHX where Hx_OtherCVD is true are true
all hX_NOT_DIABETIC where DIABETES_NOTES are TRUE are T
all HxGDM where DIABETES_NOTES is TRUE are TRUE
All_Lipids_OK
ALP is normal
at least 2 LabNumber is available
at least 2 LDL are available

Cornerstones: 59

	Case	Cornerstone
PATIENT_HISTORY		
RequestDate		
SerumInsulin		
FastingInsulin		
1Hr_insulin		
2HR_insulin		
C peptide		
NT proBNP		
FT4		
Free Thyroxine		
FT4_nc		
TPO		
TSH Rec Ab		
Bicarbonate	**To Follow**	
HIL Index		
Hemolysis		
AST		
Total Protein	**74**	
RefDrCode	**DIAW**	**MESS**
CopyDrSpecialty		
CopyDrCode		

Conditions to be applied will be listed in the Completed pane. Further information on condition syntax is located in the next section.

There are three methods of adding conditions:

Using the Current tab

1. Type the condition directly into the Current field (information on conditions can be found in the next chapter). The autocomplete function (CTRL + space) can help build syntax.
2. Use the Test button to determine:

a. Whether the condition is true for the case;
b. Whether the condition is true for the cornerstone case, if there is one;
c. How each part of the condition has evaluated.

3. If the condition is true, click the Add button.

Using the Difference List

The difference list shows the following tabs:

- Attributes: (Only those that have been used in conditions)
- Unused Attributes: (Only those not yet used in conditions)
- Derived Attributes (numeric range, list, text, or calculated value attributes)
- Features
- Derived Features
- Comments
- Report Sections

Each tab contains a table that shows the names of the respective items (attributes, features etc.), their values for the current case and their values for the current cornerstone case if there is one. If an item's value if different for the case and cornerstone case, it is shown in bold type. If an item has already been used in the current rule session, it is shown as coloured in the difference list.

When an item is selected, it is automatically inserted into the Current field. Pressing the next button will then show the templates that can assist in writing the condition. Only those items in the difference list that are relevant for the current rule session will appear.

Some items (derived attributes, features and derived features) can be double-clicked to quickly add them as conditions. For example, double-clicking a feature called 'Diabetic' will add it as a condition if it is true for the case, otherwise the condition 'not Diabetic' will be added.

When selecting one of the tabs on the difference list, the suggested conditions are correspondingly filtered. For example, when selecting the derived attributes tab, then only conditions that use a derived attribute will be shown as suggestions.

Using Suggestions

Suggestions are conditions that have previously been used in this Knowledge Base and might be relevant. To add one of these suggestions, double click and it will be placed into the Completed field.

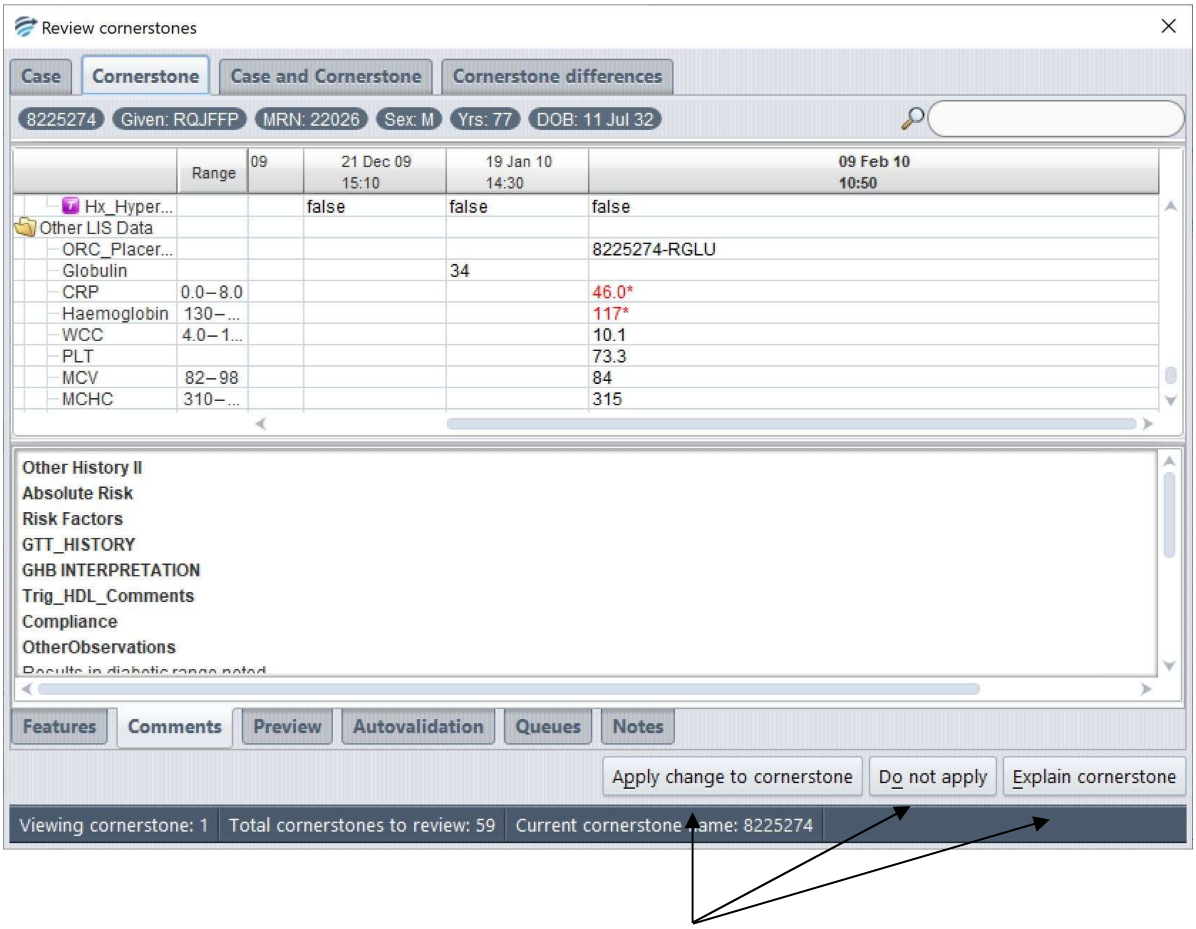
Note: a single click will move the suggestion to the Current field. Click the Add button to move the suggestion to the Completed field

Rule Wizard - Check Cornerstone Cases

As conditions are added to the Completed field, the conditions are compared to Cornerstone cases. A cornerstone case is one against which a rule has already been built. RippleDown keeps, in each project in the Cornerstone case list, a store of cornerstone cases that help prevent rules having unintended side-effects.

Cornerstone cases affected by the conditions are presented as 'Conflicting cornerstones'. The total number is shown at the Total cornerstones dialog box.

Each of these cases must be reviewed to determine whether the rule is appropriate. To do this click on the Show button to reveal the Check Cornerstone screen shown below:



	Range	09	21 Dec 09 15:10	19 Jan 10 14:30	09 Feb 10 10:50
Hx_Hyper...			false	false	false
Other LIS Data					
ORC_Placer...					8225274-RGLU
Globulin				34	
CRP	0.0–8.0				46.0*
Haemoglobin	130–...				117*
WCC	4.0–1...				10.1
PLT					73.3
MCV	82–98				84
MCHC	310–...				315

Other History II
Absolute Risk
Risk Factors
GTT_HISTORY
GHB INTERPRETATION
Trig_HDL_Comments
Compliance
OtherObservations
Results in diabetic range noted

Viewing cornerstone: 1 | Total cornerstones to review: 59 | Current cornerstone name: 8225274

If the rule is to apply to the Cornerstone case, click the ‘Apply change to cornerstone’ button. If not, click Do Not Apply and add a condition that differentiates between the rule case and the cornerstone, thus excluding the cornerstone and preventing the rule from affecting it. An explanation as to how the cornerstone case and the report will be effected if the rule is applied can also be seen by choosing “explain cornerstone”.

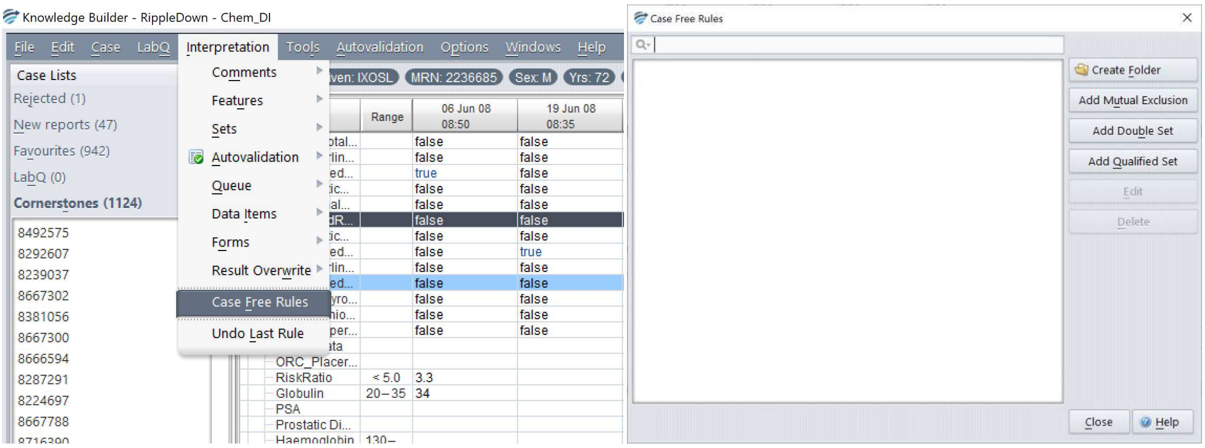
Once all Cornerstone cases have been reviewed and the Cornerstone count is shown as 0 close the dialog using the red X in the top right corner.

The Rule Wizard screen is shown again. Click Finish to apply the rule.

Case Free Rules

There is a different type of rule available within RippleDown called "Case Free Rules". Unlike the traditional rule building process, these rules can be defined in the absence of example cases. These rules can be used to identify situations where one or more elements contained in the case data are in conflict with other elements of the case. Examples of this might be an observation that a White Cell profile is normal when white blood cells are elevated; or a financial class indicates non-billable with a billing code for some tests. See the help topic "Case Free Rules".

Case free rules may be accessed through Interpretation| Case Free Rules.

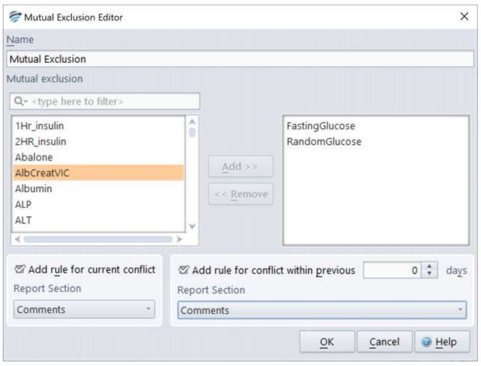


	Range	06 Jun 08 08:50	19 Jun 08 08:35
otal...		false	false
lin...		false	false
ed...		true	false
ic...		false	false
al...		false	false
dR...		false	false
ic...		false	false
ed...		false	true
lin...		false	false
ed...		false	false
yro...		false	false
hio...		false	false
per...		false	false
ata			
ORC_Placer...			
RiskRatio	< 5.0	3.3	
Globulin	20–35	34	
PSA			
Prostatic Di...			
Haemoglobin	130–		

There are three types of Case Free Rules that can be defined:

1. Mutual Exclusion Rules – when defined, consists of a list of primary attributes that should not be found together in the case. This rule will alert the user when 2 or more of the defined attributes are present in a case.
2. Double Set Rules – when defined, consists of two lists of primary attributes; List 1 and List 2. The definition also allows the knowledge builder to set a complex relationship between the members of the two lists. This is based on selecting one of all/some/none from EACH list. In some configurations, this rule will alert the user that certain exclusive attributes are present together in the case OR, conversely attributes which should be present based on co-dependent attributes missing.
3. Qualified Set Rules – This type of rule is similar in functionality to the Double Set Rule however allows the insertion of expressions into List 1.

All three types of Case Free Rules can be defined for current and historical episodes. The generated comment can be allocated to any report section.



Name: Mutual Exclusion

Mutual exclusion

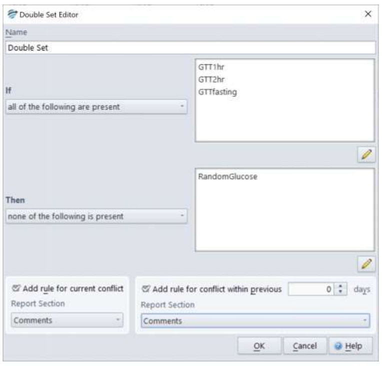
1Hr_insulin
2HR_insulin
Abalone
AlbCreatVIC
Albumin
ALP
ALT

FastingGlucose
RandomGlucose

☑ Add rule for current conflict — Report Section: Comments

☑ Add rule for conflict within previous 0 days — Report Section: Comments

Mutual Exclusion Editor



Name: Double Set

If: all of the following are present

GTT1hr
GTT2hr
GTTfasting

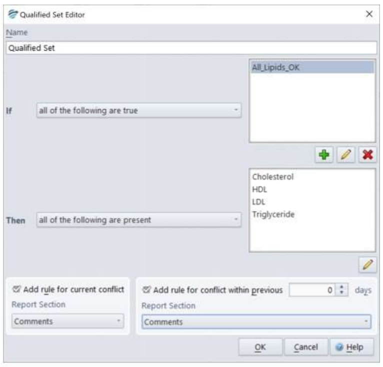
Then: none of the following is present

RandomGlucose

☑ Add rule for current conflict — Report Section: Comments

☑ Add rule for conflict within previous 0 days — Report Section: Comments

Double Set Editor



Name: Qualified Set

If: all of the following are true

All_Lipids_OK

Then: all of the following are present

Cholesterol
HDL
LDL
Triglyceride

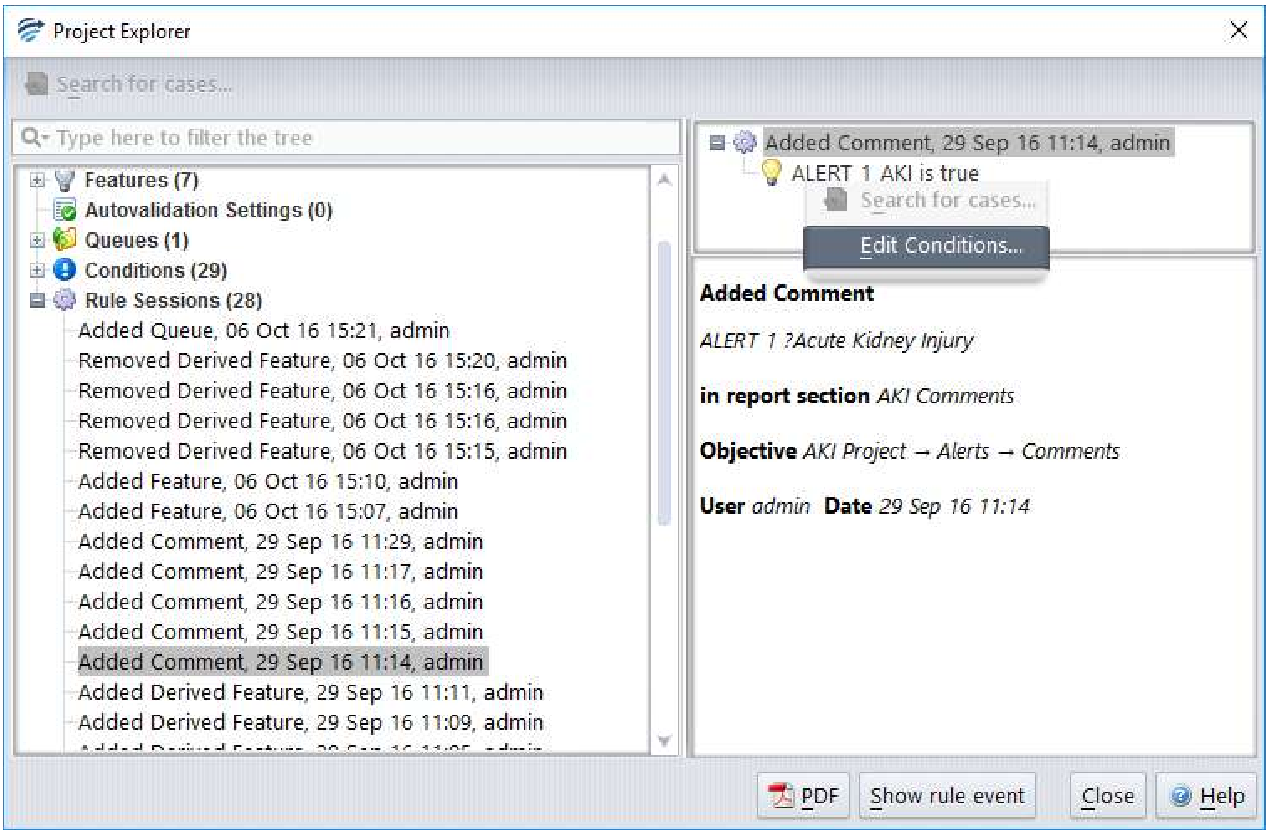
☑ Add rule for current conflict — Report Section: Comments

☑ Add rule for conflict within previous 0 days — Report Section: Comments

Qualified Set Editor

Edit Rules

In the circumstance where a rule needs to be edited, there is the ability to do this through the Project Explorer. The specific rule session concerning the rule that requires its conditions to be edited needs to be selected in the Project Explorer Rule Sessions. By right clicking on the conditions the user can select to edit these conditions.



Features (7)
Autovalidation Settings (0)
Queues (1)
Conditions (29)
Rule Sessions (28)
Added Queue, 06 Oct 16 15:21, admin
Removed Derived Feature, 06 Oct 16 15:20, admin
Removed Derived Feature, 06 Oct 16 15:16, admin
Removed Derived Feature, 06 Oct 16 15:16, admin
Removed Derived Feature, 06 Oct 16 15:15, admin
Added Feature, 06 Oct 16 15:10, admin
Added Feature, 06 Oct 16 15:07, admin
Added Comment, 29 Sep 16 11:29, admin
Added Comment, 29 Sep 16 11:17, admin
Added Comment, 29 Sep 16 11:16, admin
Added Comment, 29 Sep 16 11:15, admin
Added Comment, 29 Sep 16 11:14, admin
Added Derived Feature, 29 Sep 16 11:11, admin
Added Derived Feature, 29 Sep 16 11:09, admin

Added Comment, 29 Sep 16 11:14, admin
ALERT 1 AKI is true

Added Comment

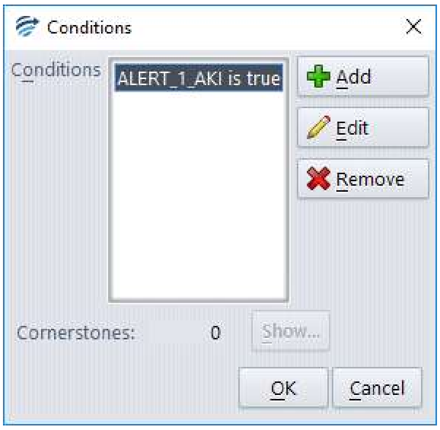
ALERT 1 ?Acute Kidney Injury

in report section *AKI Comments*

Objective *AKI Project → Alerts → Comments*

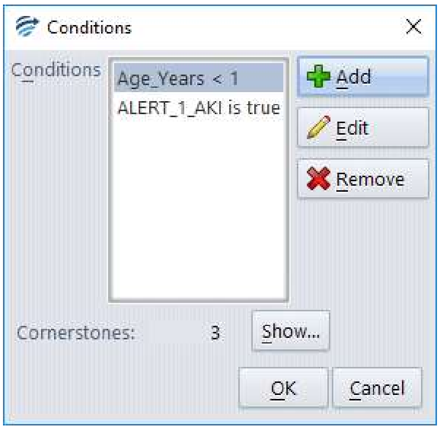
User *admin* **Date** *29 Sep 16 11:14*

The following window will then appear, giving you the option to add, remove or edit the conditions.



Conditions: ALERT_1_AKI is true

Cornerstones: 0



Conditions: Age_Years < 1
ALERT_1_AKI is true

Cornerstones: 3

Similar to when writing the rule in the Rule Wizard, as the conditions are edited, cornerstone cases will be checked against to identify examples that would be impacted by this change. Once the edited conditions are completed then the user can select the 'Show' button and review the cornerstone cases that would have their interpretation altered by this rule change. As it is important to determine if this edit requires further refinement, the cornerstone cases can be viewed before and after the edit and displayed with their differences highlighted.

Conditions

RippleDown has a rich language for writing rule conditions for analysing numeric and text results. The functions range from simple comparisons to complex pattern matching.

When building rules in RippleDown you often want to restrict the conditions to some of the episodes of a case. For example, you might only be interested in glucose values where the patient was fasting

Basic syntax

The first table shows some results for a patient:

Current episode

		10 Jan 11	16 Jul 12	5 Dec 12	3 Mar 13
Status			Non Fasting	Fasting	Fasting
Therapy		Y		N	
ClinicalNotes		?DM			Is diabetic
Glucose	3.45 – 5.45	6.3*	5.9*	5.8*	5.2
HbA1c	3.3 – 5.7	4.1			
Cholesterol	< 5.2	6.8*	5.0	5.3*	

Previous results

Most recent result

The table below shows examples of simple syntax based on the results above. For the full syntax list see http://support.pks.com.au/help/knowledgebuilder/conditiontitlepage/conditions/

Type	Condition	Example
Comparison - numeric	<, >, ≤, ≥,=	Glucose < 5.3 Glucose = 5.2
Comparison - text	is, is not, contains	Status is “Fasting”; ClinicalNotes contains “diabetic”
Existence	is available	Glucose is available
	is not available	Cholesterol is not available
Normal Range comparison	is normal is low/is high	Glucose is normal
The most recent available sample (could be the current)	Most recent	Most recent Cholesterol is high Most recent Glucose is normal
Previous episode/s (i.e. before the current)	previous	Previous HbA1c is not available Previous Cholesterol is 5.3 Some previous Cholesterol is normal
All episodes	Some None/no	Some Glucose ≥ 6.2 No HbA1c is high
	At most, At least	At most 3 Glucose are high At least 12 months since HbA1c is available
	Exactly	Exactly 3 Glucose are high
All episodes (including current) within a time period	Within last	Some Cholesterol within last year is normal No HbA1c within last year is available
All previous episodes within a time period (i.e. not the current)	Within last	Some previous (Cholesterol within last 6 months) is high
Change in result	After/where	some Therapy is "N", after Therapy is "Y"

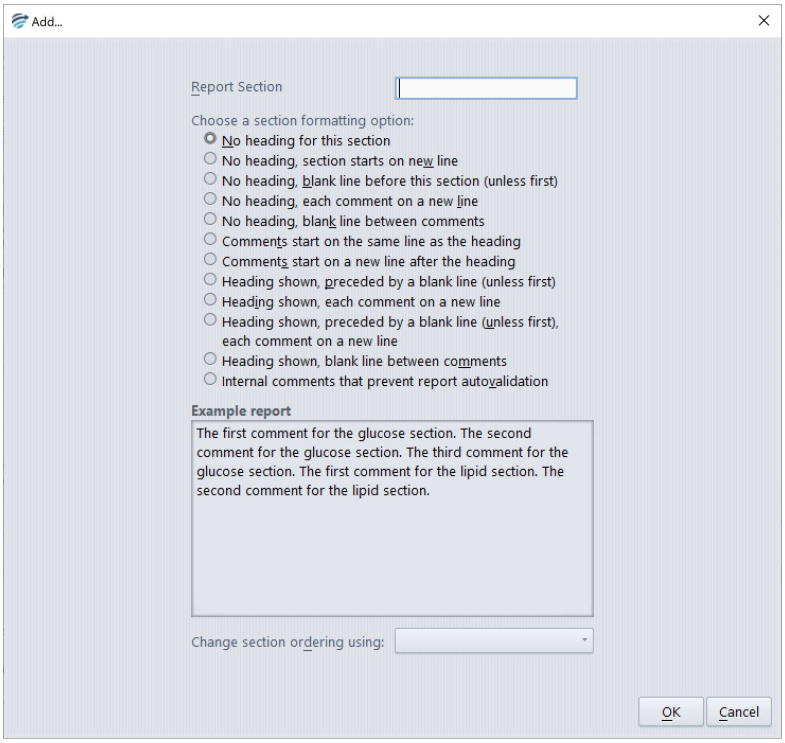
Reports

A "Report" is the ordered set of comments provided by the Knowledge Base for a case and may be divided into report sections containing one or more comments.

When a new project is created, there are two pre-defined Report Sections available:

1. A Report Section called "Comments" that is enabled by default.
2. A Report Section called "Internal", which has the pre-defined Internal format properties:
    a. Neither the heading nor the comments in this section are included in the report.
    b. Any comments for this section will appear in the Validator in a separate window as an "internal" comment for the reviewer.
    c. Comments will not be released to the Online Information System if you are using the Validator to review cases.
    d. The Internal Report Section can never be enabled by rules.

Report sections may be created, modified and ordered using the Report Section Editor: Edit | Report Sections.... and either selecting *add* or *modify* from the edit menu to show the properties of a specific Report Section:



Add...

Report Section

Choose a section formatting option:

- ● No heading for this section
- ○ No heading, section starts on new line
- ○ No heading, blank line before this section (unless first)
- ○ No heading, each comment on a new line
- ○ No heading, blank line between comments
- ○ Comments start on the same line as the heading
- ○ Comments start on a new line after the heading
- ○ Heading shown, preceded by a blank line (unless first)
- ○ Heading shown, each comment on a new line
- ○ Heading shown, preceded by a blank line (unless first), each comment on a new line
- ○ Heading shown, blank line between comments
- ○ Internal comments that prevent report autovalidation

Example report

The first comment for the glucose section. The second comment for the glucose section. The third comment for the glucose section. The first comment for the lipid section. The second comment for the lipid section.

Change section ordering using:

Here it is possible to name the report section and to select one of the predefined format options. An example of the format type is shown lower in the 'Example report' box once an option is selected.

By default, report sections will appear in the order specified in the Report Section Editor. It is possible to specify that Report Sections should be ordered on a case-by-case basis, See the Help topic on "Report Section Ordering"

Interpretive Report

An interpretive report is created by adding comments using the rule syntax. Please see next section on creating and managing comments.

The Knowledge Builder provides facilities for the user to view the current interpretation, either in a "preview" mode, to view the interpretation as it is presented to the Validator or Online Information System, or in a more detailed "comments" view.

Any notes associated with the case can also be seen in the Notes Tab. These generally consist of memos from the Validator user as to what sort of changes need to be made to the interpretation of this case before it is suitable for release.

Interpretive Actions

As well as an interpretive report, the Knowledge Builder can be configured to provide a defined set of actions. The same knowledge building methodology used for comments also applies to actions. The types of actions that can currently be defined are:

- Instructing the RippleDown Server to auto-validate cases with a particular report (see the Auto-validation section),
- Instructing the RippleDown Server to refer cases to project or specific user queues of the Validator as appropriate,
- Generating alerts within the Online Information System.
- Sending emails
- Producing PDF versions of the report
- Instructing the Online Information System to add a new test or test panel for this patient,
- Instructing the Online Information System to set some data value in the patient record, e.g. a diabetic status flag, or billing flag,

To **disable** a report section in the interpretation for a particular type of case (that is, so that any comments in this section will not be reported), select the report section heading in the "Comments" window and use the right-click menu **Disable Report Section**. The Wizard will immediately open the rule wizard to the steps that allow you to give the reasons why you wanted the report section to be disabled for that case. Similarly for enabling a report section that is currently disabled.

Comments

The Knowledge Builder provides facilities to modify the currently given interpretation, so that subsequent interpretations by the Knowledge Base give the appropriate interpretation. These facilities include:

- Creating and adding a new comment to the interpretation
- Adding an existing comment
- Replacing one comment with another one
- Removing a comment
- Changing the position of a comment in the report
- Showing why a comment is part of the interpretation
- Showing why a comment was not removed from the interpretation
- Showing other comments that have been removed from the interpretation
- Adding or removing a report section that groups related comments

Functions	**Steps**
Add	Select the "Comments" window and use the right-click menu **Add comment**. Select the comment you want to add, or to create a new one. Use the Objective to give the reasons for adding the comment
Replace	Select the comment you want to replace, then use the right-click menu **Replace comment**. Choose the comment you want to be given as the replacement. You can also create a new comment as the replacement. Use the Objective to give the reasons for replacing the comment
Remove	Select the comment you wish to remove and use the right-click menu **Remove comment**. The Wizard will jump immediately to the steps that allow you to give the reasons why you wanted the comment removed. .

Comment Editor

In addition to the facilities provided for changing the interpretation of the currently selected case, the Knowledge Builder provides the facilities to manage the list of all comments in the Knowledge Base.

Comments are grouped by “report section”. The facilities available to the user to manage comments include:

- Adding a new comment
- Inserting a variable expression into a comment, e.g. referring to a test value
- Editing a comment
- Superseding one comment by another
- Deleting a comment, provided it has not been used in rules
- Moving a comment from one report section to another
- Reordering comments within a report section
- Filter facility to search for existing comments

It is also possible to import or export comments to and from text files external to RippleDown. To do this use the Edit menu in the Comment Editor and select Import values...   The comments must be in a text (.txt) file.

Variable Comments

The Knowledge Builder provides facilities to create comments that have embedded variables. When a case is interpreted, the specific value for these variables can be inserted into the report.

This feature enhances the ability of the Knowledge Builder to generate reports that are highly specific to the patient, referring doctor and test results, for example,

> “Last ferritin level 118 ug/L noted over 3 months ago on 29/May/13.”

Where the variables have been embedded into the comment as follows:

> “Last ferritin level {last available Ferritin} ug/L noted over 3 months ago
> on {date at last available Ferritin}.”

This facility is not recommended in general for sites that use the standard Ultra interface, as library codes are used to represent reports, and an excessive number of library codes could be generated if the variables take on many different values.

Internal Comments

"Internal" comments are intended to be seen in the process of reviewing reports in the RippleDown Validator. They are not intended to be released. Typically, these comments indicate a workflow instruction, a problem in the case or an alert situation.

To create an internal comment for a case, use the Rule Wizard to add a comment in the Report Section called "Internal".

If reports are being reviewed in the Validator, any internal comments will be shown in a separate window and will not be sent to the External Information System once the report is approved. In particular, these comments do not need to be deleted.

If reports are being reviewed using the facilities of the External Information System, any internal comments will be appended to the report and so will have to be manually removed by the person reviewing the case.

If an Internal comment is added to a case it will never be auto-validated.

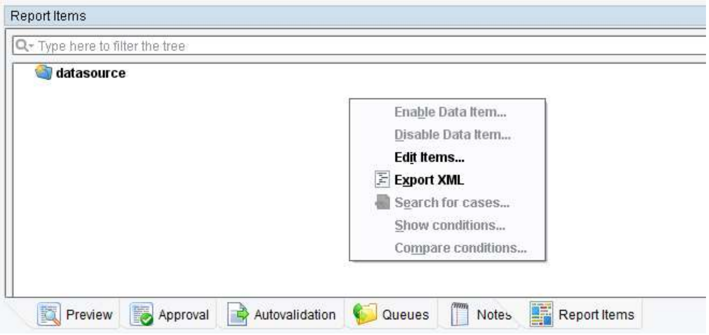
Report Formatter

Report Items

The Report Items Tab is used to generate an XML data source or schema for the reporting engine and report design tool.

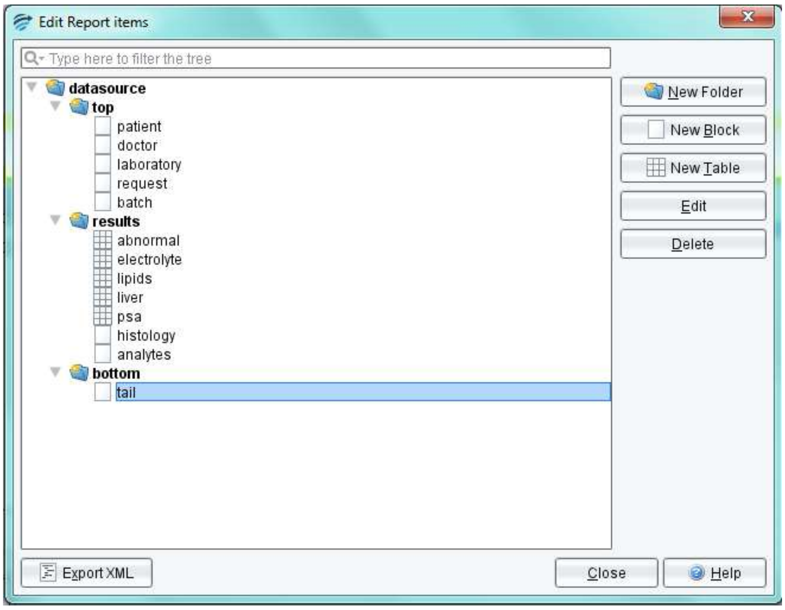
This Report Items docking pane is not shown by default. To display this pane select Windows | Report Items.

Right click in the tab and select Edit items... as shown:



The Edit Report Items dialog will appear to display and/or edit the schema that consists of block and table items. Folders may be used to organise the report items.

The example shows 3 folders: top, results and bottom. The folders contain block items such as patient and/or table items such as electrolyte.

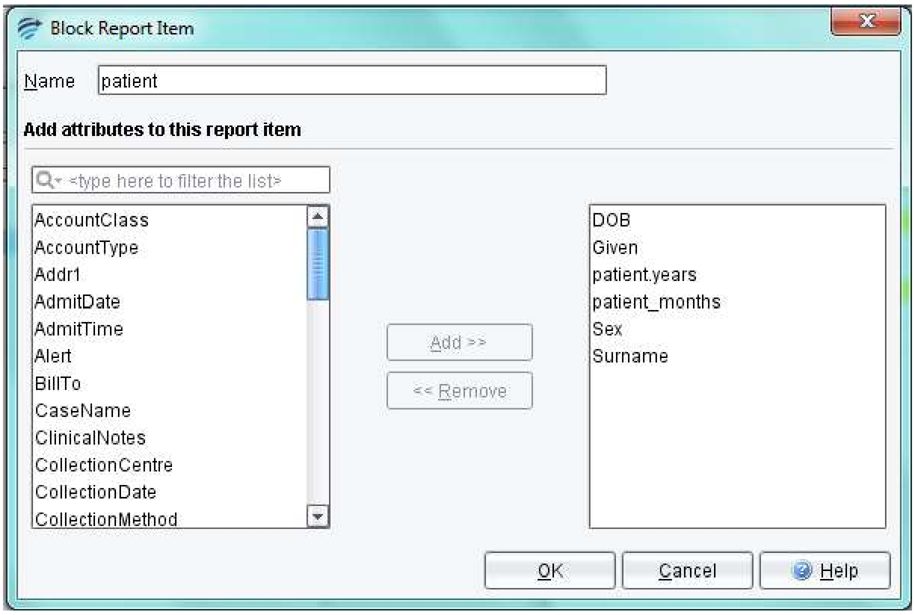


Several functions are available for use:

Functions	Description
New Folder	Allows report items to be grouped. The top level folder is always "datasource". Folders create tags in the exported XML schema. **Note**: Do not use spaces when naming folders or report items.
New Block	To define a new block item (set of attributes for which the current value is to be displayed in the report)
Table item	Define a new table item (set of attributes for which the current and historical values are to be displayed in the report).
Edit	Used to modify existing items
Delete	To permanently delete a report item
Export XML	Once all items are defined click on Export XML and select an appropriate location for the exported files. This file will be used as a data source by the reporting engine

Block Items

A block item is used when the current values of the attributes in the block item are to be printed in the report. The Edit Block Item dialog is used to name the block item and to add appropriate attributes. The example shows a block item for Patient:



Block Report Item

Name: patient

Add attributes to this report item

DOB
Given
patient.years
patient_months
Sex
Surname

Table Items

Table items are used to display both current and historical attribute values in the report. The edit table items dialog is used to name the table item and to allocate table and row data. The example shows a table item for Lipids firstly for the Column headers, secondly for the rows.

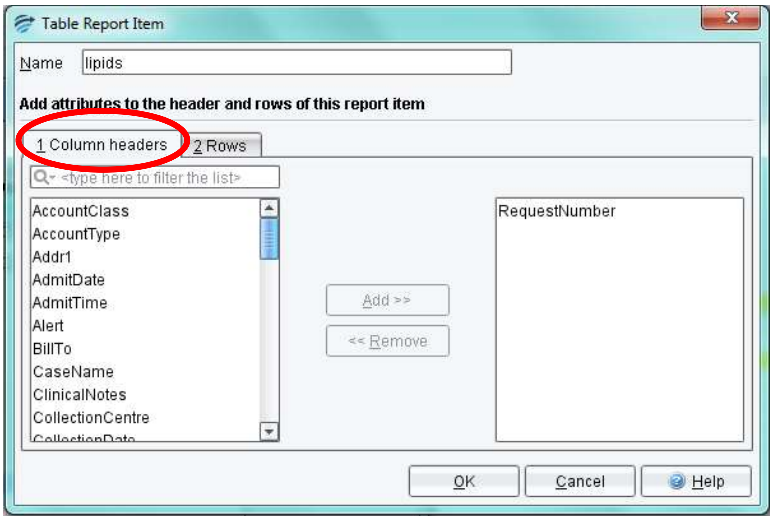


Table Report Item

Name: lipids

Add attributes to the header and rows of this report item

RequestNumber

By default, RippleDown inserts the episode date and time into the table.

The RequestNumber is added as an additional item

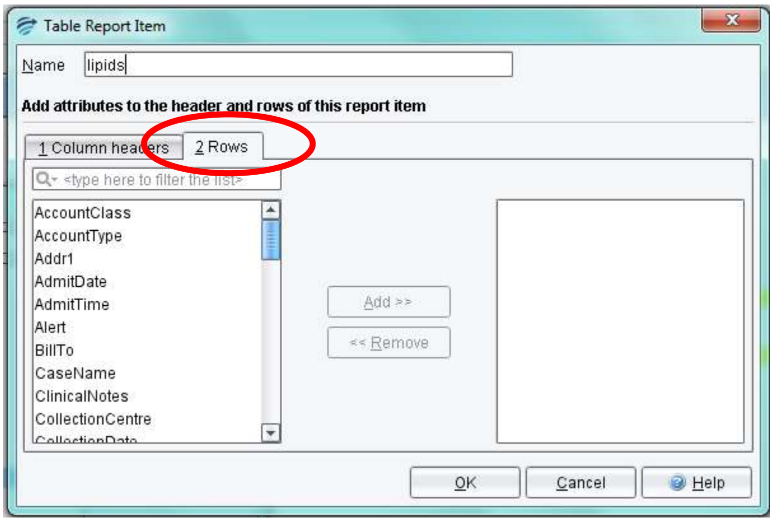


Table Report Item

Name: lipids

Add attributes to the header and rows of this report item

The rows will be populated by the primary attributes included in the group attribute called Lipids

Report Items and Rules

By default a new report item is enabled and appears as a report data source. However, any report item may be disabled using a rule. An example might be in a situation where certain report items are not required by a particular GP practice. To disable a report item, right-click on the item and select Disable Data Item. The Rule Wizard will be presented to guide the process. Once complete the disabled data item will appear in red.

Any disabled item may also be re-enabled, again using the RippleDown Rule Wizard.

XML EXPORT

Once the report items are defined they can be made available for both a report designer

tool and a report engine. Use the Export XML button to export both the XML schema (and XSD file) and an example XML file for use by a report designer tool. The values for the report items will be taken from the case currently on display in the Knowledge Builder and an example is shown on the next page.

Example XML

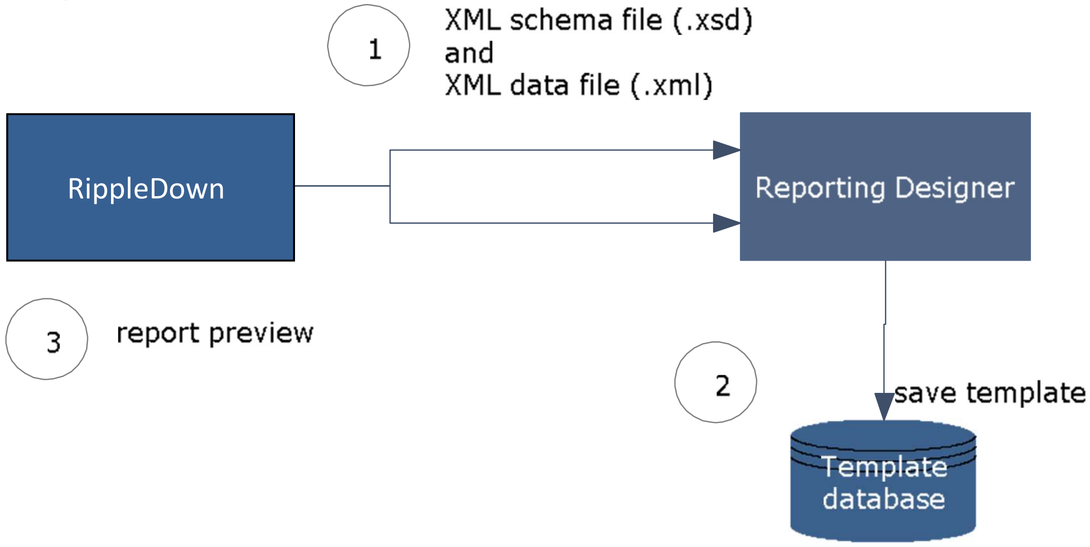
```
<?xml version="1.0" encoding="UTF-8"?>
<datasource>
  <header>
    <patient>
      <PatientFirstName>ALLISON</PatientFirstName>
      <PatientLastName>TEST</PatientLastName>
      <PatientAddressLine1>C/-IT DEPARTMENT,DO NOT POST</PatientAddressLine1>
      <PatientCity>CLAYTON</PatientCity>
      <PatientState>VIC</PatientState>
      <PatientPostcode>3168</PatientPostcode>
      <PatientPhone_work>03 9538 2239</PatientPhone_work>
      <PatientBirthdate>1995-01-01</PatientBirthdate>
      <PatientSex>M</PatientSex>
    </patient>
    <doctor>
      <DrTitle>DR</DrTitle>
      <DrFirstName>.</DrFirstName>
      <DrLastName>EDP TESTING ...</DrLastName>
      <DrCode>EDP</DrCode>
      <DrProviderNumber>2037213L</DrProviderNumber>
      <DrAddressLine1>EDP DEPARTMENT</DrAddressLine1>
      <DrAddressLine2>1868 DANDENONG ROAD</DrAddressLine2>
      <DrCity>CLAYTON</DrCity>
      <DrState>VIC</DrState>
      <DrPostcode>3168</DrPostcode>
      <DrPhone_work>03 9828 7748</DrPhone_work>
      <DrPhone_mobile>0411206657</DrPhone_mobile>
      <DrPhone_pager>94831611</DrPhone_pager>
      <DrPhone_fax>0395386742</DrPhone_fax>
      <DrPhone_home>03 9543 6853</DrPhone_home>
    </doctor>
    <laboratory>
      <LabName>LABORATORY 3427</LabName>
      <Lab_info>DR D DEAM</Lab_info>
      <LabAddressLine1>1868 DANDENONG RD</LabAddressLine1>
      <LabState>VIC</LabState>
      <LabCity>CLAYTON</LabCity>
      <LabPostcode>3168</LabPostcode>
      <LabPhone_work>03 9538 6777</LabPhone_work>
    </laboratory>
    <request>
      <ReferalDate>2007-06-25</ReferalDate>
      <CollectionDateTime>2007-06-25 00:00:00</CollectionDateTime>
    </request>
    <batch>
      <RunNumber>10680</RunNumber>
      <PrintingDateTime>2010-05-11 19:49:00</PrintingDateTime>
    </batch>
  </header>
  <results>
    <table>
      <name>abnormal</name>
      <header>
        <date>25/06/2007 00:00</date>
        <RequestNumber>07-1111661</RequestNumber>
      </header>
      <header>
        <date>07/03/2007 00:00</date>
        <RequestNumber>07-3333331</RequestNumber>
      </header>
  <row>
    <name>PSA</name>
    <sample>
      <value>1000.0</value>
      <low>&lt;</low>
      <high>2.6</high>
      <units>Î¼g/L</units>
      <flag>1</flag>
    </sample>
    <sample>
      <value>0.1</value>
      <units>Î¼g/L</units>
  <flag>0</flag></sample>
  </row>
</table>
<table>
  <name>electrolyte</name>
  <header>
    <date>25/06/2007  00:00</date>
    <RequestNumber>07-1111661</RequestNumber>
  </header>
  <header>
    <date>07/03/2007  00:00</date>
    <RequestNumber>07-3333331</RequestNumber>
  </header>
</table>
<table>
  <name>lipids</name>
  <header>
    <date>25/06/2007  00:00</date>
    <RequestNumber>07-1111661</RequestNumber>
  </header>
  <header>
    <date>07/03/2007  00:00</date>
    <RequestNumber>07-3333331</RequestNumber>
  </header>
</table>
<table>
  <name>liver</name>
  <header>
    <date>25/06/2007  00:00</date>
    <RequestNumber>07-1111661</RequestNumber>
  </header>
  <header>
    <date>07/03/2007  00:00</date>
    <RequestNumber>07-3333331</RequestNumber>
  </header>
</table>
<table>
  <name>psa</name>
  <header>
    <date>25/06/2007  00:00</date>
    <RequestNumber>07-1111661</RequestNumber>
  </header>
  <header>
    <date>07/03/2007  00:00</date>
    <RequestNumber>07-3333331</RequestNumber>
  </header>
  <row>
    <name>PSA</name>
    <sample>
      <value>1000.0</value>
      <low>&lt;</low>
      <high>2.6</high>
```

```
          <units>Î¼g/L</units>
          <flag>1</flag>
        </sample>
        <sample>
          <value>0.1</value>
          <units>Î¼g/L</units>
          <flag>0</flag>
        </sample>
      </row>
    </table>
    <histology/>
  </results>
  <patient_footer>
    <footer>
      <DrLastName>EDP TESTING ...</DrLastName>
      <PrintingDateTime>2010-05-11  19:49:00</PrintingDateTime>
    </footer>
  </patient_footer>
</datasource>
```

Template Design

Templates for pdf reports are created in a Report Design tool such as iReport and Adobe→ LiveCycle→ Workbench Designer using data defined in RippleDown.

Design Workflow



1. RippleDown exports two files to the design tool. The .xsd is a schema file and the .xml contains data from the current case.
2. Template is created and saved in the template database.
3. Report preview in RippleDown. To preview the pdf report in RippleDown open the Preview Tab and select the Pdf Preview button. This action will export an XML file of the current case to the report engine which will select the specific template and return the pdf file to RippleDown.

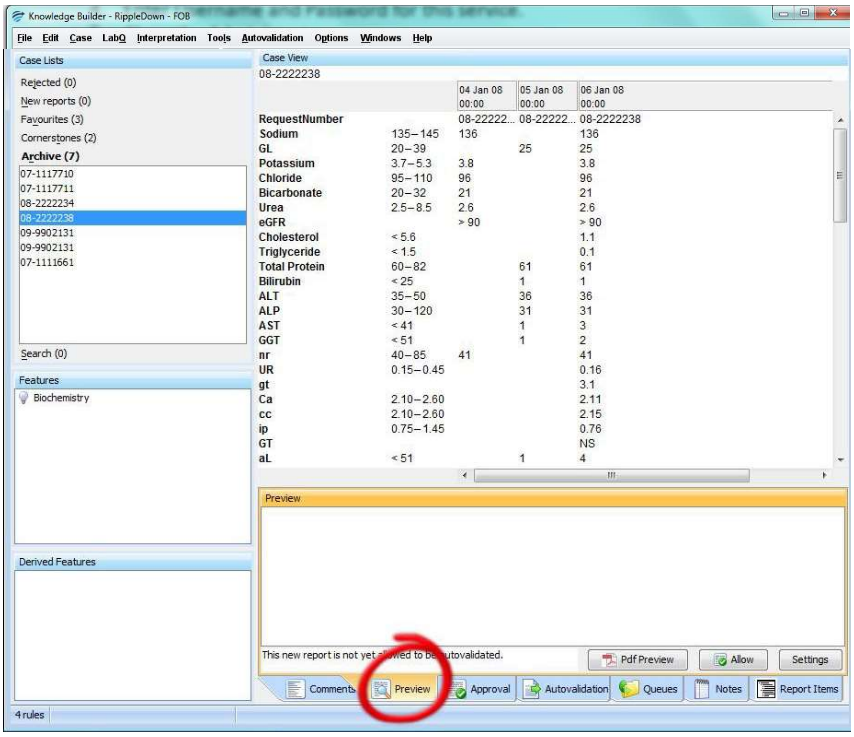
Report Preview

In RippleDown it is possible to preview the PDF output generated by the reporting engine.

This function requires configuration of the RippleDown server. To do this:

1. Open the RippleDown Administrator client.
2. Use the Tools menu and select Edit server settings.
3. Enter the URL of the print service and click Apply.
4. Enter Username and Password for this service.
5. Close the Administrator.

In the Knowledge Builder, select the Preview tab (adjacent to the Comments tab as shown):



08-2222238

		04 Jan 08 00:00	05 Jan 08 00:00	06 Jan 08 00:00
RequestNumber		08-22222...	08-22222...	08-2222238
Sodium	135–145	136		136
GL	20–39		25	25
Potassium	3.7–5.3	3.8		3.8
Chloride	95–110	96		96
Bicarbonate	20–32	21		21
Urea	2.5–8.5	2.6		2.6
eGFR		> 90		> 90
Cholesterol	< 5.6			1.1
Triglyceride	< 1.5			0.1
Total Protein	60–82		61	61
Bilirubin	< 25		1	1
ALT	35–50		36	36
ALP	30–120		31	31
AST	< 41		1	3
GGT	< 51		1	2
nr	40–85	41		41
UR	0.15–0.45			0.16
gt				3.1
Ca	2.10–2.60			2.11
cc	2.10–2.60			2.15
ip	0.75–1.45			0.76
GT				NS
aL	< 51		1	4

This new report is not yet allowed to be autovalidated.

Click the PDF Preview button. This action will cause RippleDown to send an XML export to the reporting engine. A PDF report will be created and returned to RippleDown.

Project Explorer

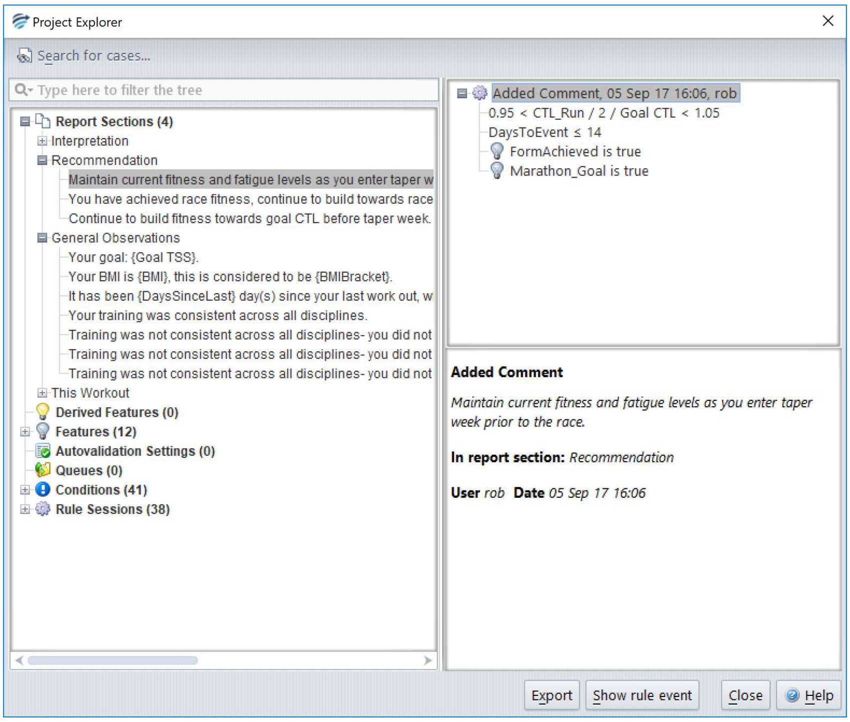
Project Explorer is an invaluable tool showing the history of the project and a display of all major components of the project with a tracking user id, date and time stamp.

Start Project Explorer by going to menu Tools | Project Explorer (this may take a minute or two depending on the size of the project. A progress bar is shown at the bottom left hand corner of the screen).

Each of the sections are listed in the index, along with the number of each in parenthesis.

To see the details of a section double click on the name.

The example shows the Report Section "Recommendation" and the comment: "Maintain current fitness and fatigue levels as you enter taper week prior to the race"



Report Sections (4)
- Interpretation
- Recommendation
  - Maintain current fitness and fatigue levels as you enter taper w
  - You have achieved race fitness, continue to build towards race
  - Continue to build fitness towards goal CTL before taper week.
- General Observations
  - Your goal: {Goal TSS}.
  - Your BMI is {BMI}, this is considered to be {BMIBracket}.
  - It has been {DaysSinceLast} day(s) since your last work out, w
  - Your training was consistent across all disciplines.
  - Training was not consistent across all disciplines- you did not
  - Training was not consistent across all disciplines- you did not
  - Training was not consistent across all disciplines- you did not
- This Workout

Derived Features (0)

Features (12)

Autovalidation Settings (0)

Queues (0)

Conditions (41)

Rule Sessions (38)

Added Comment, 05 Sep 17 16:06, rob
- 0.95 < CTL_Run / 2 / Goal CTL < 1.05
- DaysToEvent ≤ 14
- FormAchieved is true
- Marathon_Goal is true

Added Comment

Maintain current fitness and fatigue levels as you enter taper week prior to the race.

In report section: *Recommendation*

User *rob* **Date** *05 Sep 17 16:06*

In the upper right pane it shows the history of the comment and all the reasons for it being added or removed.

The lower right pane displays location and tracking details of the comment.

To view the specific details associated with the rule event select the **Show Rule Event** button. A window will be displayed, showing the case on which the rule was made (i.e. the cornerstone case). The window will have 2 tabs, the 1st showing the case details before the rule was created, and the 2nd showing the case details after the rule was created.

Export Project Explorer

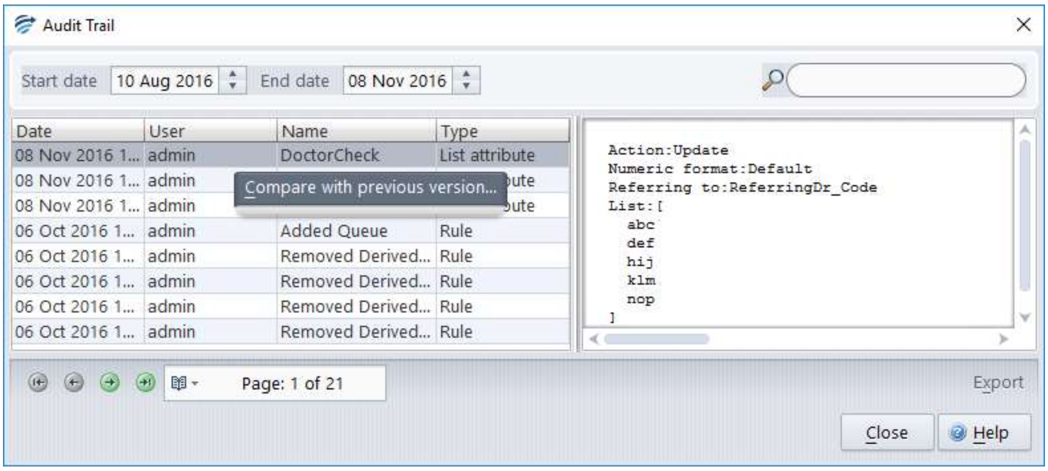
The Project Explorer may be exported in an HTML format:

1. Click the **Export** button
2. Enter a name
3. Select a location and click **Ok.**

Audit Trail

Audit Trail is an additional record keeping function that is found in the tools menu. It enables the user to tracks any changes of any variables (like comments or derived attributes) within the project. The user can to compare the current status of the item with previous versions of the item, allowing full traceability in the project.

The Audit Trail has a multiple search options to locate particular entries, the period can be adjusted, all columns in the table are sortable and in the top right hand corner there is a free text search box.



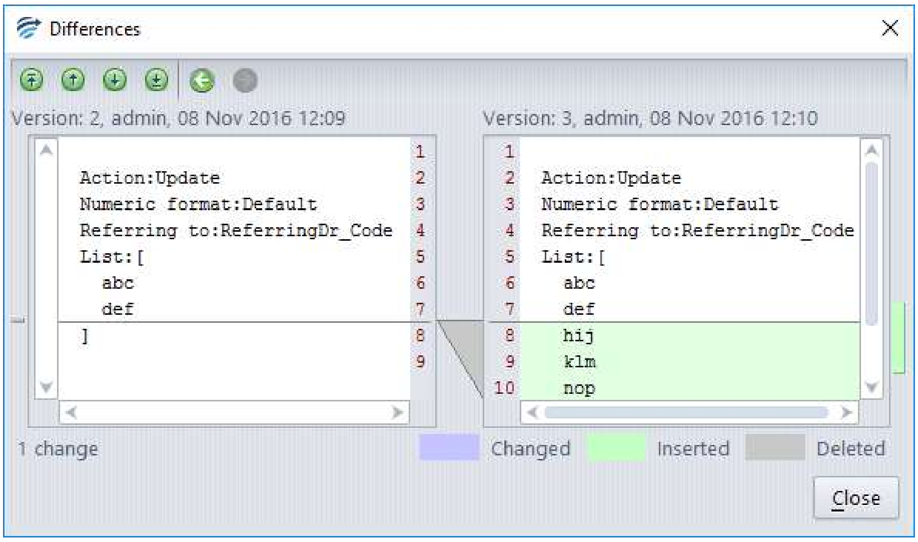
Start date 10 Aug 2016 End date 08 Nov 2016

Date	User	Name	Type
08 Nov 2016 1...	admin	DoctorCheck	List attribute
08 Nov 2016 1...	admin		ute
08 Nov 2016 1...	admin		ute
06 Oct 2016 1...	admin	Added Queue	Rule
06 Oct 2016 1...	admin	Removed Derived...	Rule
06 Oct 2016 1...	admin	Removed Derived...	Rule
06 Oct 2016 1...	admin	Removed Derived...	Rule
06 Oct 2016 1...	admin	Removed Derived...	Rule

```
Action:Update
Numeric format:Default
Referring to:ReferringDr_Code
List:[
  abc
  def
  hij
  klm
  nop
]
```

Page: 1 of 21

Once a desired entry has been located, a list derived attribute in this example, in order to drill down and compare the current version and the previous version of the list simply right click.



Version: 2, admin, 08 Nov 2016 12:09

```

Action:Update
Numeric format:Default
Referring to:ReferringDr_Code
List:[
  abc
  def
]
```

Version: 3, admin, 08 Nov 2016 12:10

```

Action:Update
Numeric format:Default
Referring to:ReferringDr_Code
List:[
  abc
  def
  hij
  klm
  nop
```

1 change

Changes are clearly highlighted so you can see in this example that three new doctor codes were added to the list attribute at 12:10 on November 8th by the user: admin.

Auto-validation

Auto-validation is the facility by which cases receiving particular types of interpretations are released to the Online Information System without the need for human intervention.

Auto-validation may be appropriate when:

- You have reviewed a large number of cases giving a particular interpretation, and that interpretation is now always correct whenever it is given, or
- The rules giving a certain type of interpretation are so general and simple that only minimal review is needed, or
- The project is not being used to provide reports, but to automate processes in the laboratory such as reflexive testing or billing actions.

Definitions

	Description
Report	A Report is the ordered set of comments provided by the Knowledge Base for a case
Report Section	A Report Section is a component of a report. It consists of a series of comments and may or may not contain a heading (depending on settings defined in the report Section Editor)
New Report	A new report is automatically created when a comment is first given for some case, or when a new combination of comments is first given

There are three ways of managing auto-validation, either by reports, or by report sections or by rules

1. "per report" basis, a percentage auto-validation level is set for each unique report, regardless of the number of report sections, that is generated by the knowledge base. This is the most restrictive policy, but the safest.

2. "per report section" basis, a percentage auto-validation level is set for each unique report section that is generated by the knowledge base. This is less restrictive than the "per report" policy. It is used if the Report Sections are relatively independent.

3. Rule-based auto-validation, use the rule wizard to add, modify or remove auto-validation settings that are given by the project, in exactly the same way as you would if you were adding, modifying or removing comments using rules. This is the most flexible approach.

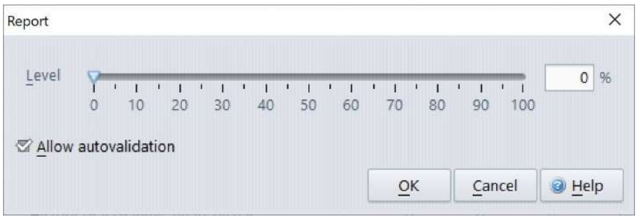
For any particular project, the choices of "per report" or "per report section" are mutually exclusive. However, auto-validation by rules will take precedence over any "per-report" or "per report section" based levels

Allow Autovalidation

Auto-validation cannot occur unless the report or report section has been allowed to be considered for auto-validation. This can occur in one of three ways:

a. Either use the Preview pane on the Knowledge Builder and select the Allow button OR

b. Open Auto-validation/Report Settings and double click the report to open the screen shown below. Click the checkbox to allow auto-validation:



Report

Level: 0 %

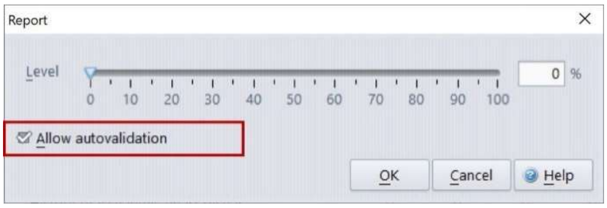
☑ Allow autovalidation

OR

c. Using the Validator to approve the case through normal lab workflow (this will only apply if the user has Knowledge Builder privilege)

Auto-validation setting to release a new report.

There is a check box in the Auto-validation setup screen that will release a report and apply the set Auto-validation rate even if it has not been specifically allowed. This will occur once the report has been reviewed by a Knowledge Builder user or a Validator user with the corresponding permission.

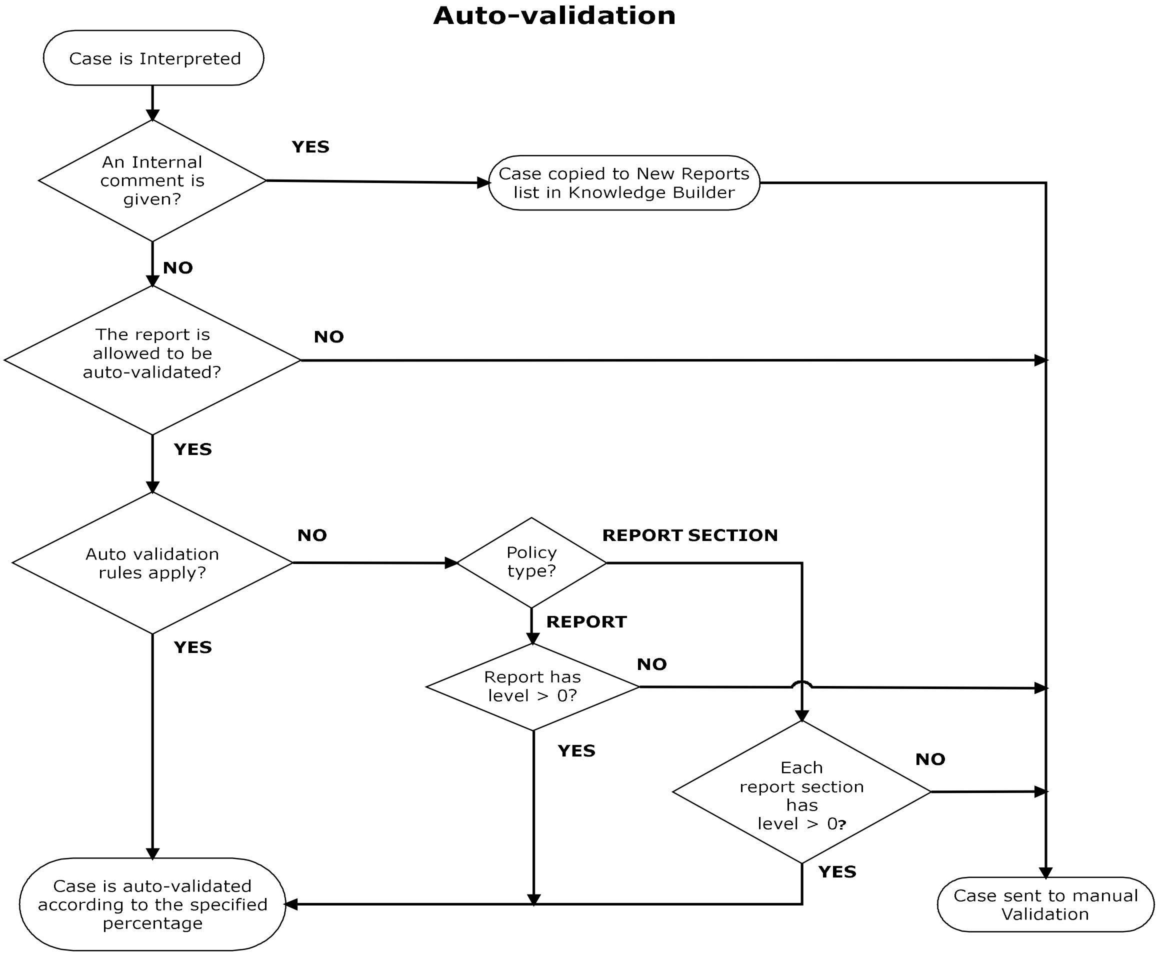


Report

Level: 0 %

☑ Allow autovalidation

Auto-validation Options


Auto-validation
Case is Interpreted
An Internal comment is given?
YES
Case copied to New Reports list in Knowledge Builder
NO
The report is allowed to be auto-validated?
NO
YES
Auto validation rules apply?
NO
Policy type?
REPORT SECTION
REPORT
Report has level > 0?
NO
YES
YES
Each report section has level > 0?
NO
YES
Case is auto-validated according to the specified percentage
Case sent to manual Validation

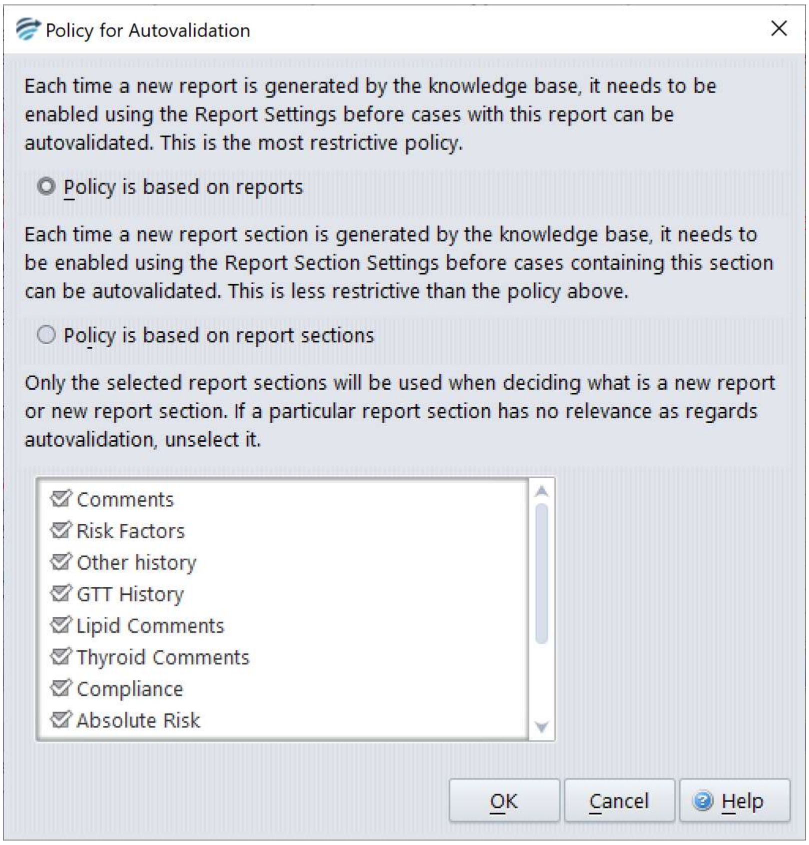
Auto-Validation levels

The knowledge builder is able to define an auto-validation level between 0% and 100% for each report, report section or auto-validation rule setting

Note: In normal situations, it is advisable to not set an auto-validation level above about 95%. This ensures that a small number of cases continue to appear for review in validation and provides continuing quality audit of the knowledge base.

Policy

The choice between the "per report' and "per report section" policy is done using the menu Autovalidation | Policy.



Policy for Autovalidation

Each time a new report is generated by the knowledge base, it needs to be enabled using the Report Settings before cases with this report can be autovalidated. This is the most restrictive policy.

● Policy is based on reports

Each time a new report section is generated by the knowledge base, it needs to be enabled using the Report Section Settings before cases containing this section can be autovalidated. This is less restrictive than the policy above.

○ Policy is based on report sections

Only the selected report sections will be used when deciding what is a new report or new report section. If a particular report section has no relevance as regards autovalidation, unselect it.

- ☑ Comments
- ☑ Risk Factors
- ☑ Other history
- ☑ GTT History
- ☑ Lipid Comments
- ☑ Thyroid Comments
- ☑ Compliance
- ☑ Absolute Risk

The default setting for a project is “Policy based on reports” with all report sections being selected.

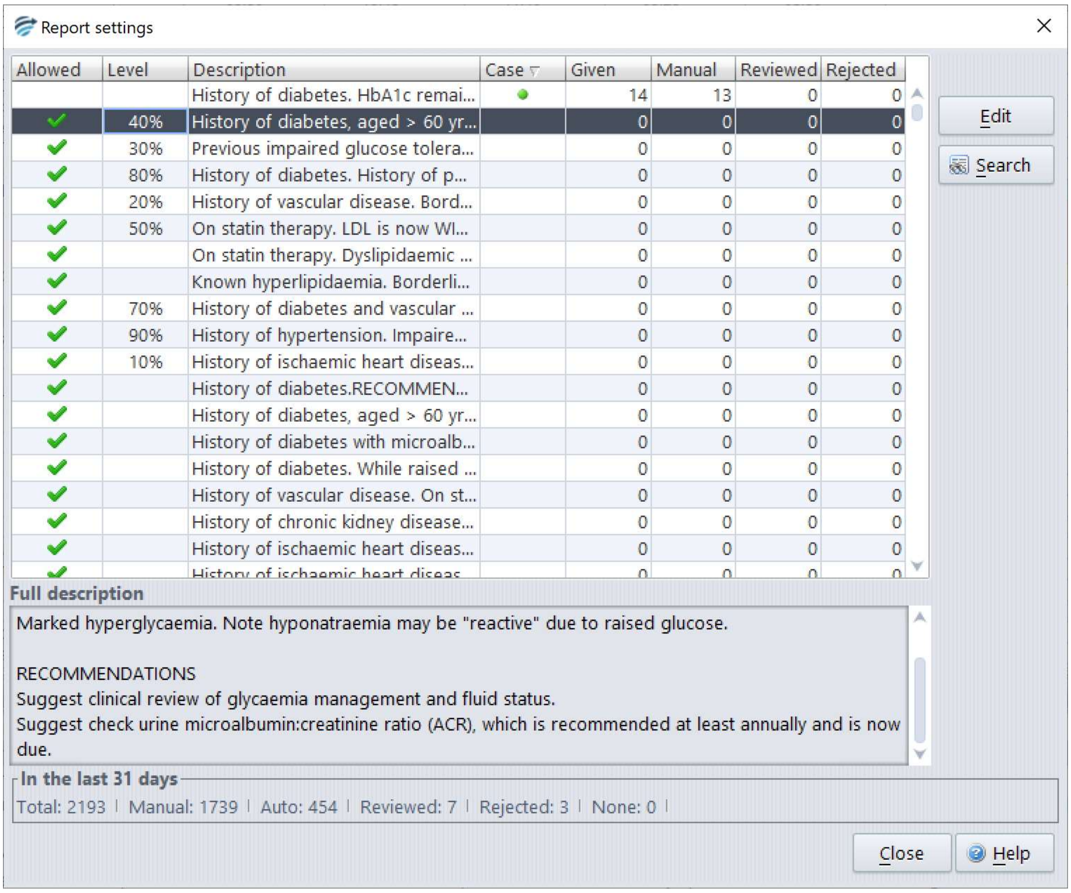
Policy based on reports

This option allows each unique report to be considered for auto-validation once the report has been allowed for auto-validation consideration.

To set auto-validation levels on a report-by-report basis:

1. Set the Policy to Policy based on reports in Autovalidation | Policy.

2. Use the menu Auto-validation | Report Settings to show a table of all reports that have been recently given by the project, that is, given within the last month.



Allowed	Level	Description	Case	Given	Manual	Reviewed	Rejected
		History of diabetes. HbA1c remai...	●	14	13	0	0
✓	40%	History of diabetes, aged > 60 yr...		0	0	0	0
✓	30%	Previous impaired glucose tolera...		0	0	0	0
✓	80%	History of diabetes. History of p...		0	0	0	0
✓	20%	History of vascular disease. Bord...		0	0	0	0
✓	50%	On statin therapy. LDL is now Wl...		0	0	0	0
✓		On statin therapy. Dyslipidaemic ...		0	0	0	0
✓		Known hyperlipidaemia. Borderli...		0	0	0	0
✓	70%	History of diabetes and vascular ...		0	0	0	0
✓	90%	History of hypertension. Impaire...		0	0	0	0
✓	10%	History of ischaemic heart diseas...		0	0	0	0
✓		History of diabetes.RECOMMEN...		0	0	0	0
✓		History of diabetes, aged > 60 yr...		0	0	0	0
✓		History of diabetes with microalb...		0	0	0	0
✓		History of diabetes. While raised ...		0	0	0	0
✓		History of vascular disease. On st...		0	0	0	0
✓		History of chronic kidney disease...		0	0	0	0
✓		History of ischaemic heart diseas...		0	0	0	0
✓		History of ischaemic heart diseas		0	0	0	0

Full description

Marked hyperglycaemia. Note hyponatraemia may be "reactive" due to raised glucose.

RECOMMENDATIONS
Suggest clinical review of glycaemia management and fluid status.
Suggest check urine microalbumin:creatinine ratio (ACR), which is recommended at least annually and is now due.

In the last 31 days

Total: 2193 | Manual: 1739 | Auto: 454 | Reviewed: 7 | Rejected: 3 | None: 0 |

3. Review each report in the table. Reports that already have some level of auto-validation set will have a tick in the **Allowed** column, and possibly a percentage in the **Level** column. Note that clicking on a column heading will sort the table by the values in that column. For example, clicking on the **Given** column will order the reports according to how often each report has been given by the project in the last month.

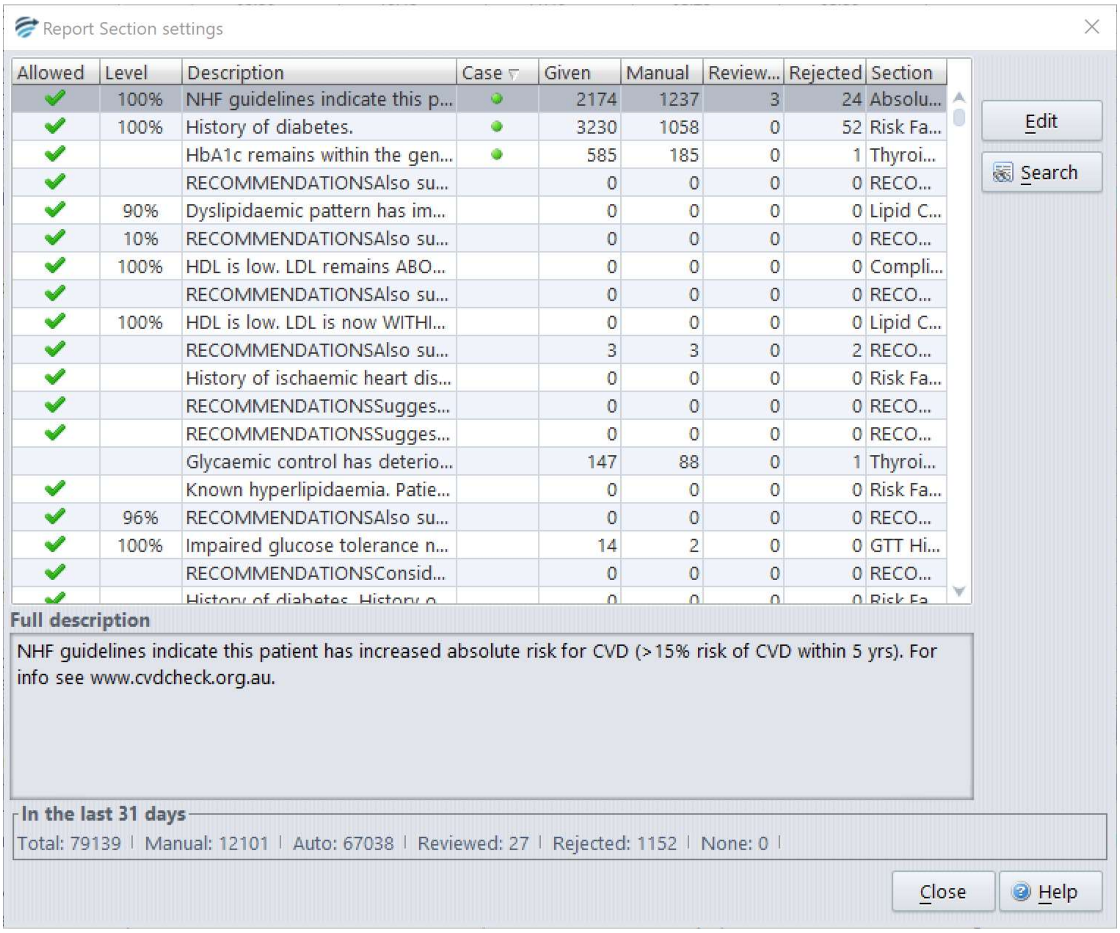
4. The **Rejected** column shows how many times the corresponding report was corrected during validation, that is, after being inappropriately given by the project. Reports that are never, or rarely, rejected are possible candidates for auto-validation.

5. If you decide to set a non-zero auto-validation level for a report, click the **Edit** button and set the desired auto-validation level for that report. The Allow auto-validation check box will be checked automatically if you set a non-zero auto-validation level.

Changes made to auto-validation settings are effective immediately.

Policy based on report sections

This option allows each unique report section to be auto-validated. This is appropriate when comments given on one report section are independent of comments given in other report sections.



Report Section settings

Allowed	Level	Description	Case	Given	Manual	Review...	Rejected	Section
✓	100%	NHF guidelines indicate this p...	●	2174	1237	3	24	Absolu...
✓	100%	History of diabetes.	●	3230	1058	0	52	Risk Fa...
✓		HbA1c remains within the gen...	●	585	185	0	1	Thyroi...
✓		RECOMMENDATIONSAlso su...		0	0	0	0	RECO...
✓	90%	Dyslipidaemic pattern has im...		0	0	0	0	Lipid C...
✓	10%	RECOMMENDATIONSAlso su...		0	0	0	0	RECO...
✓	100%	HDL is low. LDL remains ABO...		0	0	0	0	Compli...
✓		RECOMMENDATIONSAlso su...		0	0	0	0	RECO...
✓	100%	HDL is low. LDL is now WITHI...		0	0	0	0	Lipid C...
✓		RECOMMENDATIONSAlso su...		3	3	0	2	RECO...
✓		History of ischaemic heart dis...		0	0	0	0	Risk Fa...
✓		RECOMMENDATIONSSugges...		0	0	0	0	RECO...
✓		RECOMMENDATIONSSugges...		0	0	0	0	RECO...
		Glycaemic control has deterio...		147	88	0	1	Thyroi...
✓		Known hyperlipidaemia. Patie...		0	0	0	0	Risk Fa...
✓	96%	RECOMMENDATIONSAlso su...		0	0	0	0	RECO...
✓	100%	Impaired glucose tolerance n...		14	2	0	0	GTT Hi...
✓		RECOMMENDATIONSConsid...		0	0	0	0	RECO...
✓		History of diabetes. History o...		0	0	0	0	Risk Fa...

Full description

NHF guidelines indicate this patient has increased absolute risk for CVD (>15% risk of CVD within 5 yrs). For info see www.cvdcheck.org.au.

In the last 31 days

Total: 79139 | Manual: 12101 | Auto: 67038 | Reviewed: 27 | Rejected: 1152 | None: 0 |

To set auto-validation levels on a report section basis:

1. Set the policy to policy based on report sections in Autovalidation | Policy.
2. Use the menu Autovalidation | Report Section settings to show a table of all report sections that have been recently given by the project, that is, given within the last month.
3. Review each report section in the table. Report sections that already have some level of auto-validation set will have a tick in the **Allowed** column, and a percentage in the **Level** column. Note that clicking on a column heading will sort the table by the values in that column. For example, clicking on the **Given** column will order the report sections according to how often each report section has been given by the project.

4. The **Rejected** column shows how many times the corresponding sub-report was corrected during validation, that is, was inappropriately given by the project. Report sections that are never, or rarely, rejected are possible candidates for auto-validation.
5. If you decide to set a non-zero auto-validation level for a report section, click the **Edit** button and set the desired auto-validation level for that report section. The Allow auto-validation check box will be checked automatically if you set a non-zero auto-validation level.

Changes made to auto-validation settings will be effective immediately.

Auto-validation by Rules

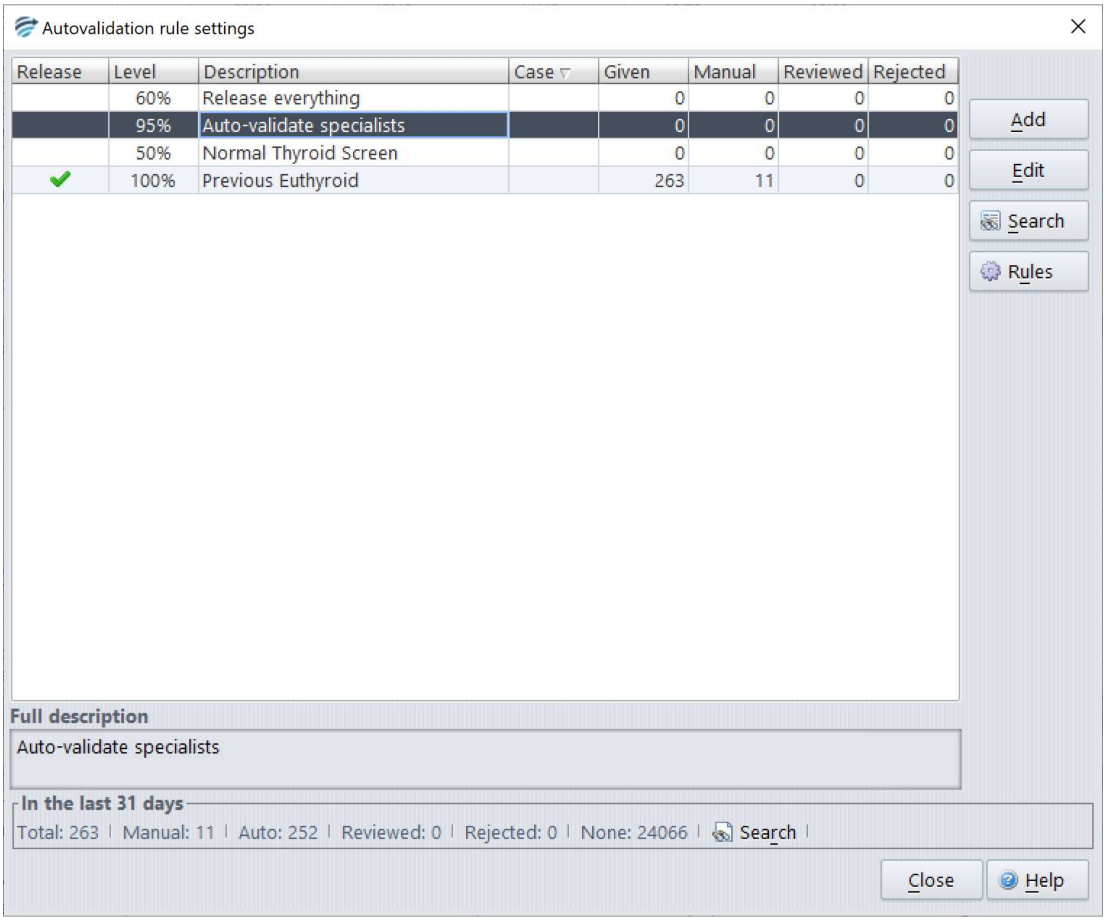
To use rules to manage auto-validation, you add an auto-validation setting to a case using the rule wizard, in exactly the same way as you would add a comment to a case.

1. Find a case (e.g. on the Archive case list or the Rejected case list) which is representative of the type of case that could safely be auto-validated at some level.
2. Click on the Auto-validation Pane (adjacent to the Preview Pane)
3. Right click to start the rule wizard to add an "auto-validation setting", consisting of a percentage level and a textual description. You can select an existing setting, or create a new one.
4. Create or select an objective for this rule, and finally
5. Add conditions for the rule, check cornerstone cases, and finish the rule session.

To replace one auto-validation setting for a case with another using a rule, you select the current setting, then use the rule wizard to replace it with another one, in exactly the same way as you would replace a comment given for a case, giving only the conditions relevant to the change.

Finally, to remove the auto-validation setting for a case, you select the current setting and use the rule wizard to remove it.

You can see the list of all possible auto-validation settings using the menu from the main screen 'Autovalidation | Autovalidation rule settings...'



Autovalidation rule settings

Release	Level	Description	Case	Given	Manual	Reviewed	Rejected
	60%	Release everything		0	0	0	0
	95%	Auto-validate specialists		0	0	0	0
	50%	Normal Thyroid Screen		0	0	0	0
✓	100%	Previous Euthyroid		263	11	0	0

Full description

Auto-validate specialists

In the last 31 days

Total: 263 | Manual: 11 | Auto: 252 | Reviewed: 0 | Rejected: 0 | None: 24066 | Search |

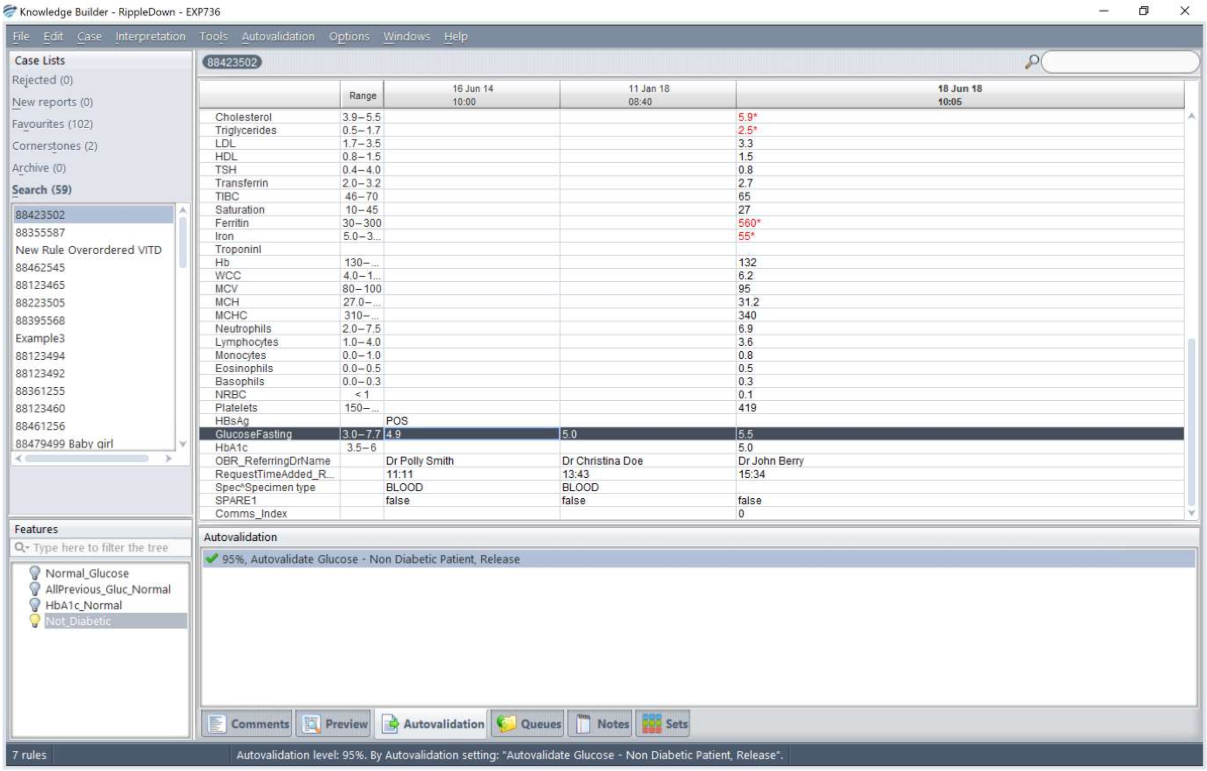
This facility also allows you to edit the level of an auto-validation setting. The new level will apply to all cases with that auto-validation setting.

Notes:

1. If more than one setting is given for a case, the setting with the lowest auto-validation level will apply.
2. Auto-validation settings using rules will take priority over any levels that have been set using reports or report sections.
3. Even if an auto-validation setting is given for a case, it will NOT be auto-validated if the report for the case has yet to be allowed (or report section depending on the policy).
4. Auto-validation will not take effect if an earlier occurrence of the case exists on a Validation queue awaiting manual validation. NB: Interpretive comments WILL be updated.

Suggestion for setting Auto-validation by rules: Decide the characteristics of a report that may be suitable for Auto-validation e.g. all normal results or all negative results and/or relevant locations

1. Build a series of features that apply for the normal/negative situations
2. Build a derived feature that will apply when all the features apply
3. Build a rule to allow auto-validation based on the presence of the derived feature.

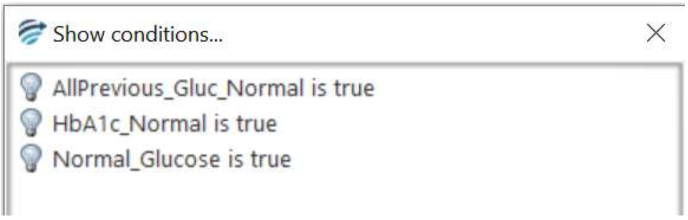


	Range	16 Jun 14 10:00	11 Jan 18 08:40	18 Jun 18 10:05
Cholesterol	3.9–5.5			5.9*
Triglycerides	0.5–1.7			2.5*
LDL	1.7–3.5			3.3
HDL	0.8–1.5			1.5
TSH	0.4–4.0			0.8
Transferrin	2.0–3.2			2.7
TIBC	46–70			65
Saturation	10–45			27
Ferritin	30–300			560*
Iron	5.0–3...			55*
TroponinI				
Hb	130–...			132
WCC	4.0–1...			6.2
MCV	80–100			95
MCH	27.0–...			31.2
MCHC	310–...			340
Neutrophils	2.0–7.5			6.9
Lymphocytes	1.0–4.0			3.6
Monocytes	0.0–1.0			0.8
Eosinophils	0.0–0.5			0.5
Basophils	0.0–0.3			0.3
NRBC	< 1			0.1
Platelets	150–...			419
HBsAg		POS		
GlucoseFasting	3.0–7.7	4.9	5.0	5.5
HbA1c	3.5–6			5.0
OBR_ReferringDrName		Dr Polly Smith	Dr Christina Doe	Dr John Berry
RequestTimeAdded_R...		11:11	13:43	15:34
Spec^Specimen type		BLOOD	BLOOD	
SPARE1		false	false	false
Comms_Index				0

✓ 95%, Autovalidate Glucose - Non Diabetic Patient, Release

Autovalidation level: 95%. By Autovalidation setting: "Autovalidate Glucose - Non Diabetic Patient, Release".

In the simple example above the Derived Feature allowing auto-validation is applied under the following conditions:



AllPrevious_Gluc_Normal is true
HbA1c_Normal is true
Normal_Glucose is true

These are all features describing auto-validation critera for the separate tests. All of the criteria must be satisfied for the auto-validation setting to be given.

Manual Validation and Queues

Whenever a case is processed by the RippleDown server it is assigned one of two possible status flags:

a. Auto-validated (i.e. ready for release), or
b. Manual validation required (i.e. must be approved by a user before release).

If the case has been auto-validated, it is sent to the Online Information System with a status flag indicating that the report has been approved and so is ready for release.

Conversely, if manual validation is required, there are the following possibilities for reviewing for the case, depending on how the Knowledge Base has been setup:

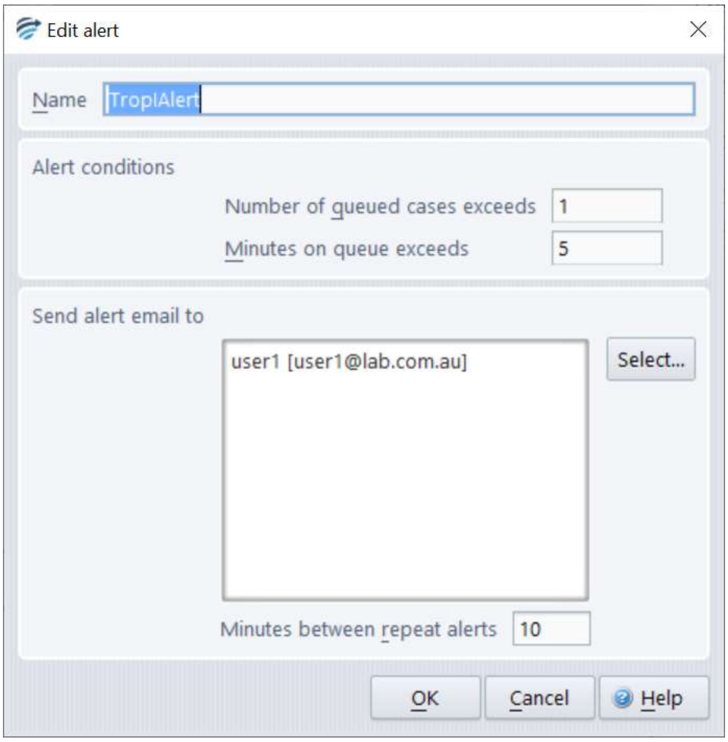
a. If a review queue has been defined by the Knowledge Base for this case, then the case will be queued to that special-purpose queue on the Validator,

b. Otherwise, if the Knowledge Base has been setup by the RippleDown Administrator to use the RippleDown Validator by default the case will be sent to the default project queue on the RippleDown Validator,

c. Otherwise, the case will be sent to the Online Information System with the appropriate status flag indicating that it still needs to be approved. The facilities of the Online Information System can be used to review and possibly correct the report prior to release. This is known as External validation.

Queue Alerts

Knowledge builders are able to define alerts for queues from Edit | Queues - emails will be generated to the assigned alert recipients when a case has been on a queue longer than a specified number of minutes, or if more than a defined number of cases are exceeded on that queue. The intervals at which these alerts are sent can be customised to suit the alert and multiple alerts can be assigned to the one queue allowing escalation.



Name TropIAlert

Number of queued cases exceeds 1

Minutes on queue exceeds 5

user1 [user1@lab.com.au]

Minutes between repeat alerts 10

Searching

Find a Case in a Case List

To find a case by case name click on the Case List title and start typing the case name. The “*” may be used as a wildcard in the search.

Find cases by condition

You can search for cases that satisfy some condition, even if that condition has not been used in a rule.

To do this:

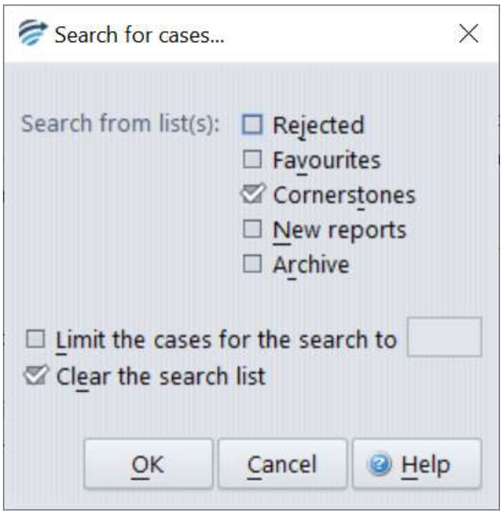
1. Select the case list you want to search (you can select any list, including the Search list),
2. Use the menu "Case | Search by condition" ,
3. Enter the condition you want to use (it does not have to be true for the currently selected case), then press OK.

By default, each new search will clear any cases that were previously in the Search list.

If you do not want to clear cases on the Search list before you do the search, then uncheck the option to "Clear the search list first". Any cases that are found will then be copied to the end of the Search list. In this way you can refine your search by using another search condition

Find cases by comment

You can search for cases that give a specific comment, feature or derived feature. If the comment (or feature) is given by the current case, right click on the comment and select the pop-up menu Search for cases. A dialog will appear allowing you to refine the search:



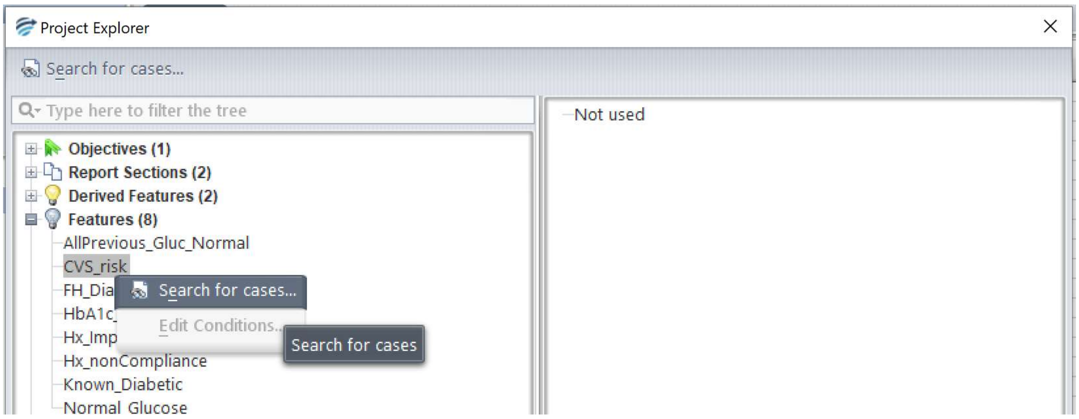
Search for cases...

Search from list(s):
☐ Rejected
☐ Favourites
☑ Cornerstones
☐ New reports
☐ Archive

☐ Limit the cases for the search to
☑ Clear the search list

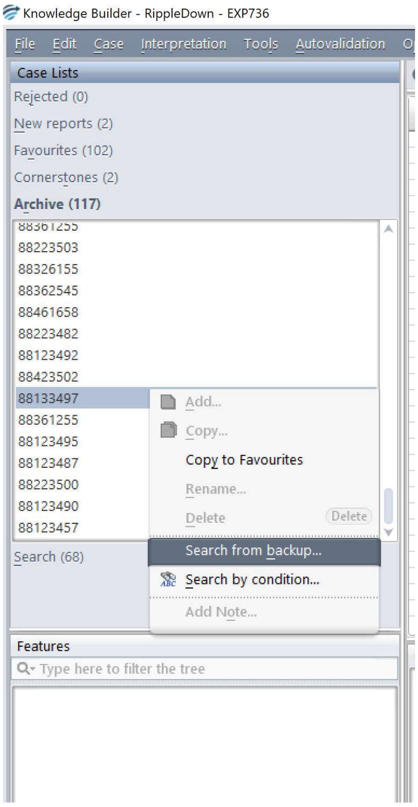
Any cases that are found will be copied to the Search case list.

If the comment (or feature) that you are interested in is not given by the current case, open the Project Explorer and select the comment or feature, and click the Search tool on the tool bar.

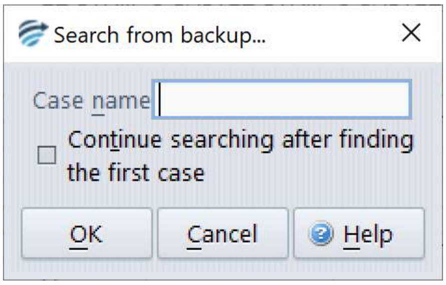


Find cases from backup

It is possible to search for a back-up copy of a case that has been interpreted by a project. To do this, activate the 'Case' menu and select 'Search from backup...'



A dialog will appear, prompting you for the name (lab number) of the case. When you click 'OK', the server begins the search as a background process.



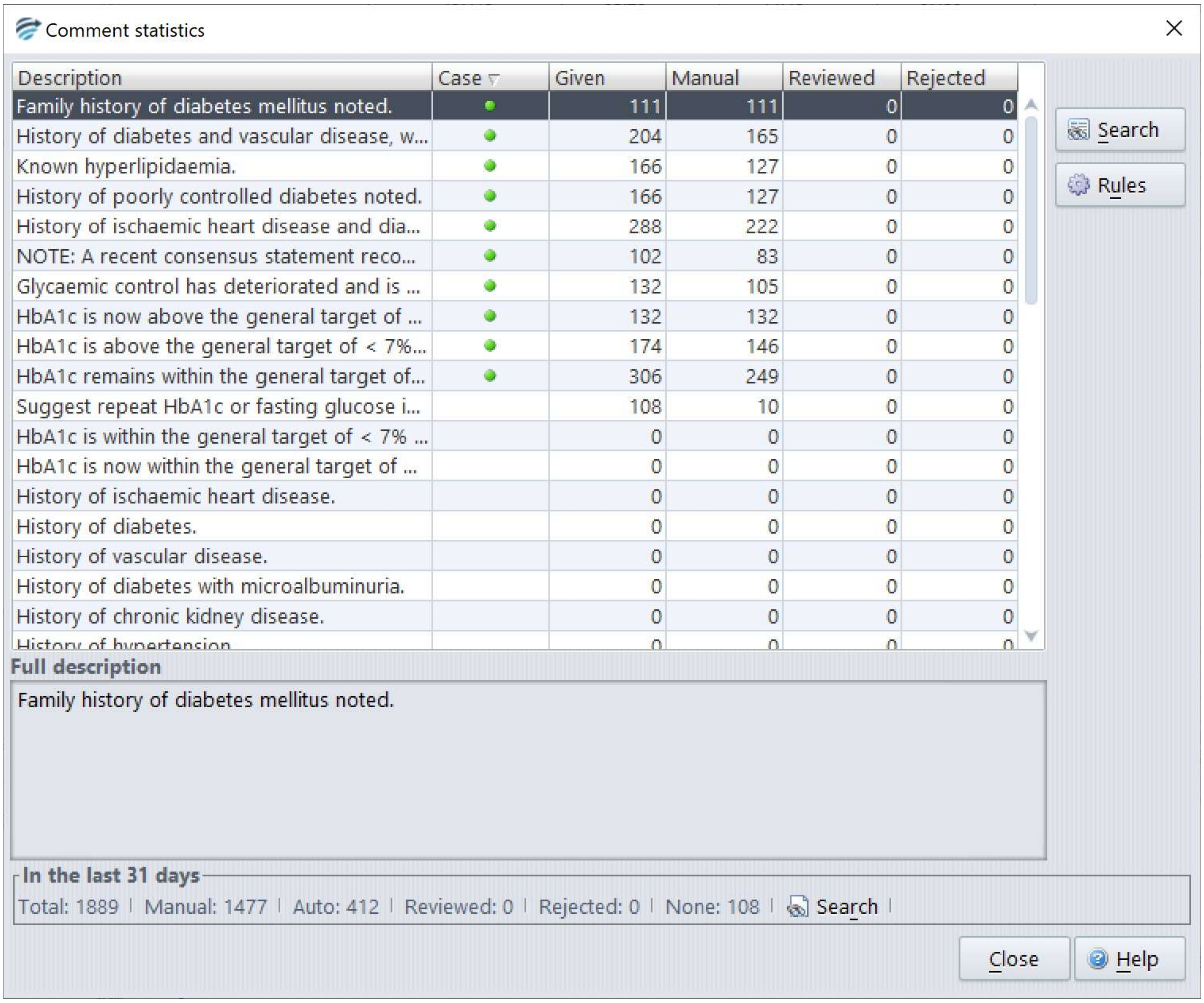
A progress bar is shown at the bottom left hand corner of the screen to indicate how many of the backup files have been searched. As the cases are found, they are put into the Search case list.

Sometimes a case is sent several times by the online information system, over the course of a few days. If you want to see all of these versions of the case, choose the option 'Continue the search after the first case is found'. This will search all backups, and thus take longer, but can be cancelled at any stage.

Performance Reports

Comment Statistics

To see statistics on a particular comment given by the project, use the menu Autovalidation | Comment statistics. This will show a table of all comments that have been recently given by the project, that is, given within the last month.



Description	Case ▿	Given	Manual	Reviewed	Rejected
Family history of diabetes mellitus noted.	●	111	111	0	0
History of diabetes and vascular disease, w...	●	204	165	0	0
Known hyperlipidaemia.	●	166	127	0	0
History of poorly controlled diabetes noted.	●	166	127	0	0
History of ischaemic heart disease and dia...	●	288	222	0	0
NOTE: A recent consensus statement reco...	●	102	83	0	0
Glycaemic control has deteriorated and is ...	●	132	105	0	0
HbA1c is now above the general target of ...	●	132	132	0	0
HbA1c is above the general target of < 7%...	●	174	146	0	0
HbA1c remains within the general target of...	●	306	249	0	0
Suggest repeat HbA1c or fasting glucose i...		108	10	0	0
HbA1c is within the general target of < 7% ...		0	0	0	0
HbA1c is now within the general target of ...		0	0	0	0
History of ischaemic heart disease.		0	0	0	0
History of diabetes.		0	0	0	0
History of vascular disease.		0	0	0	0
History of diabetes with microalbuminuria.		0	0	0	0
History of chronic kidney disease.		0	0	0	0
History of hypertension		0	0	0	0

Full description

Family history of diabetes mellitus noted.

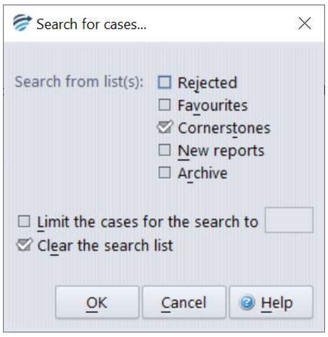
In the last 31 days

Total: 1889 | Manual: 1477 | Auto: 412 | Reviewed: 0 | Rejected: 0 | None: 108 | Search |

The statistics for each comment are:

- the number of cases with this comment that have been given
- the number of cases with this comment that have been manually validated
- the number of cases with this comment that have been rejected

Furthermore, the summary panel also shows the number of cases that did not receive any comment within the last month.



Search from list(s):
☐ Rejected
☐ Favourites
☑ Cornerstones
☐ New reports
☐ Archive

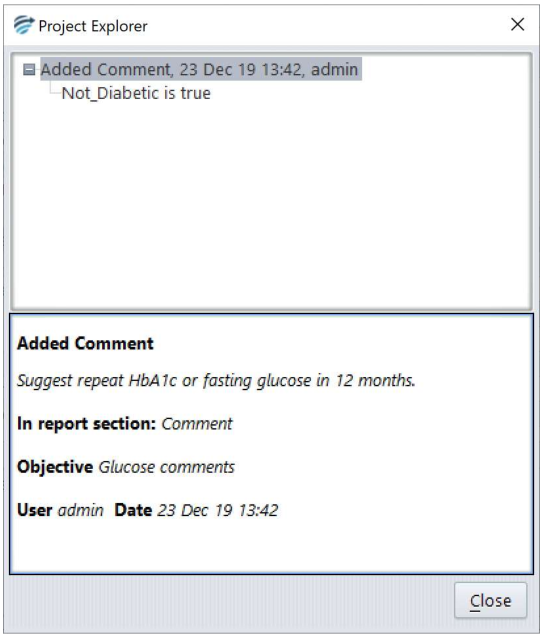
☐ Limit the cases for the search to
☑ Clear the search list

The **Search** button will activate a search facility for the comment that is highlighted.

Select the case list for the search and enter a limit, if required. Click OK to commence the search.

The cases that are found will be added to the Search Case List.

The **Rules** button shows all the rules in the project that add or remove the comments:



Added Comment, 23 Dec 19 13:42, admin
Not_Diabetic is true

Added Comment

Suggest repeat HbA1c or fasting glucose in 12 months.

In report section: *Comment*

Objective *Glucose comments*

User *admin* **Date** *23 Dec 19 13:42*

Simulation reports

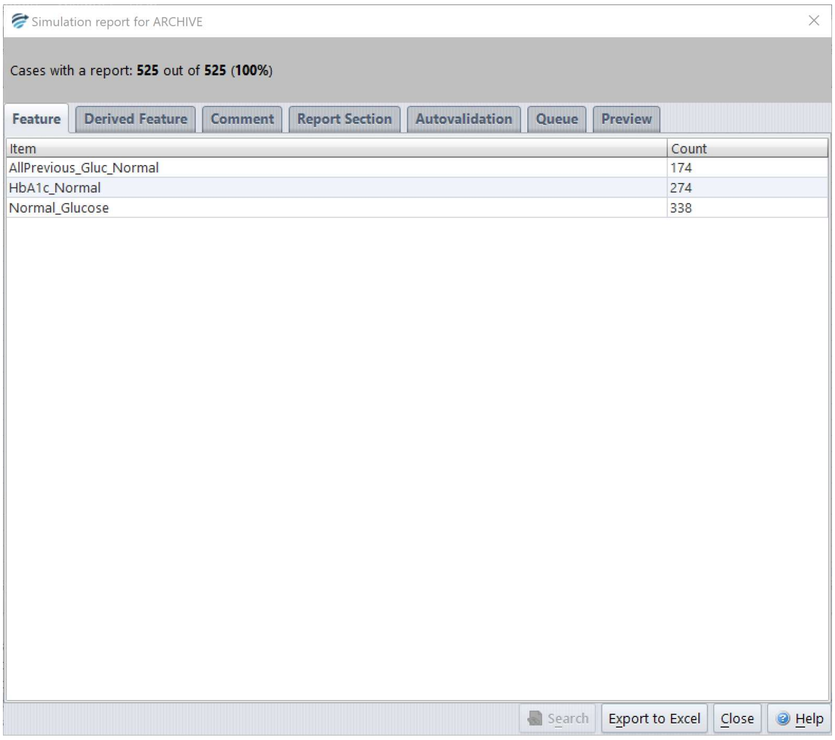
RippleDown has a Simulation tool that enables you to analyse the interpretations of all cases in any one of the case lists, such as the Archive or Cornerstone list.

Select the list you want to analyse, then use the menu Tools | Simulation Report. After a short time, a dialog will appear that contains the following statistics:

- A summary statistic showing the number of cases that have an interpretation (that is, a non-empty report)
- A set of detailed statistics for each:
  - Feature
  - Derived Feature
  - Comment
  - Report Section
  - Auto-validation setting
  - Review Queue
  - Report

The detailed statistics are arranged in a set of tables, with each table containing the particular item (e.g. Feature) and the associated count (i.e. the number of cases giving that Feature). You can sort the information in each table by clicking on the column heading.

An example of the Simulation report run on the Cornerstone case list and sorted by the "Feature" column is shown below:



Simulation report for ARCHIVE

Cases with a report: **525** out of **525** (**100%**)

Item	Count
AllPrevious_Gluc_Normal	174
HbA1c_Normal	274
Normal_Glucose	338

If you select any item from one of the tables (e.g. a Feature), the Search button becomes enabled, allowing you to search for cases that give this particular item. You can search any case list for the specified item, not just the list used for the Simulation Report.

The results of the search will go into the Search list on the main screen. You are able to access the main screen even while the Simulation Report dialog is showing.

The Export button enables you to export the Simulation Report to an Excel spreadsheet for further analysis.

NB: Ensure the name given to the file is suffixed with .xls

LabQ®

LabQ is RippleDown's data mining tool.

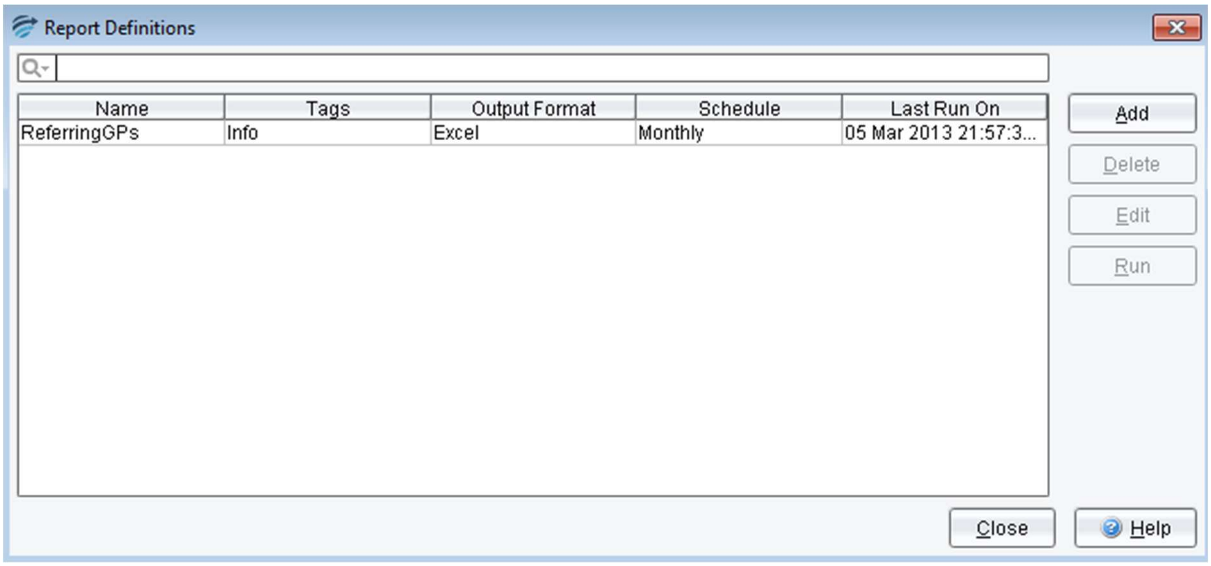
With LabQ, you can build highly customised rules-based data extractions. These extractions can be run against any RippleDown case store.

This facility allows you to define a report with the following properties:

Conditions	Allows uses to filter cases to be included in a generated report.
Tags	Allows uses to filter report definitions in the Report Definition table for quick access.
Schedule Option	Allows uses to specify to generate reports on-demand or periodically. A report will be generated periodically when a schedule is given. Periodically generated reports are stored under %RDR_HOME%/projects/reports with a file name {report-definition-name}-{yyyyMMddHHmmss}.{output-format}, e.g ReferringGPs-201302280730.xslx.
Columns	Allows uses to select columns that to be included in a generated report with a specific order. If Set Conclusions are used, the report will NOT contain predefined columns “Case”, “Id” and “Date”, otherwise the predefined columns will be included.

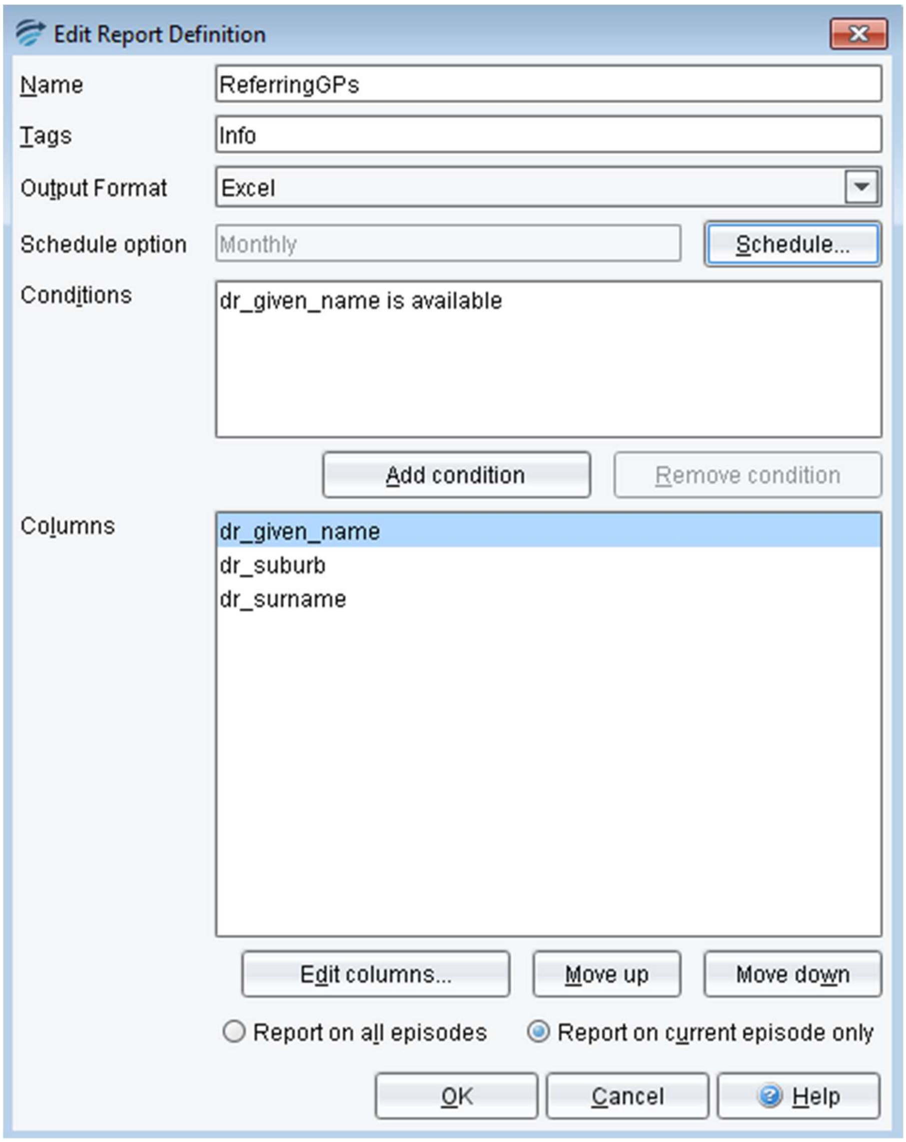
Note: Only users with LabQ access privileges are able to add, edit or delete a Report Definition.

1. Launch the “Report Definitions” dialog from the LabQ menu.



Name	Tags	Output Format	Schedule	Last Run On
ReferringGPs	Info	Excel	Monthly	05 Mar 2013 21:57:3...

2. Select the ‘Add’ button to launch “Edit Data Definition” dialog.



Edit Report Definition

Name: ReferringGPs

Tags: Info

Output Format: Excel

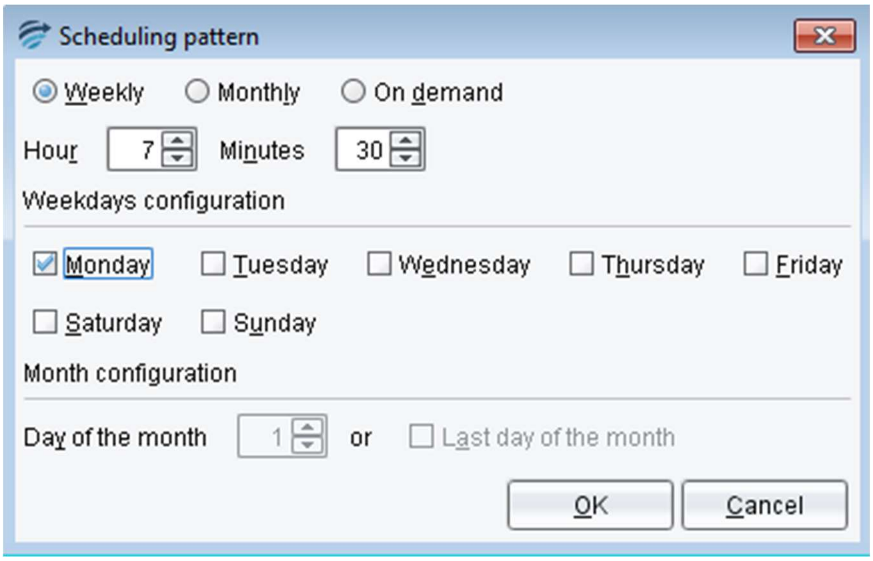
Schedule option: Monthly

Conditions: dr_given_name is available

Columns:
dr_given_name
dr_suburb
dr_surname

○ Report on all episodes ● Report on current episode only

3. Enter the name of the Report Definition (required).
4. Enter the tags (optional), a comma separated values. eg. NY, Public, Patient
5. Select output format (required).
6. Select schedule option (required), “On Demand” by default e.g. Scheduling report generation on every Monday at 07:30



Scheduling pattern

● Weekly ○ Monthly ○ On demand

Hour 7 Minutes 30

Weekdays configuration

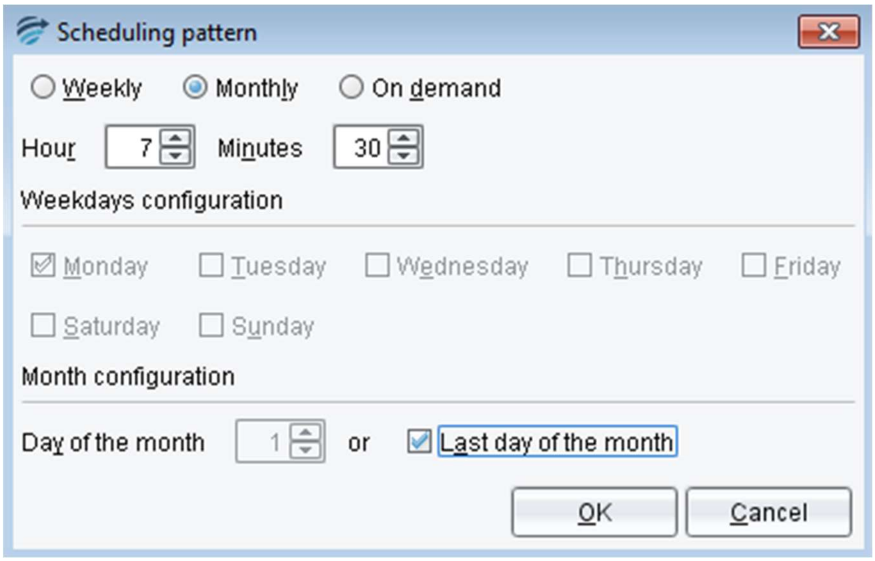
☑ Monday ☐ Tuesday ☐ Wednesday ☐ Thursday ☐ Friday

☐ Saturday ☐ Sunday

Month configuration

Day of the month 1 or ☐ Last day of the month

e.g. Scheduling report generation on every last day of month at 07:30



Scheduling pattern

○ Weekly ● Monthly ○ On demand

Hour 7 Minutes 30

Weekdays configuration

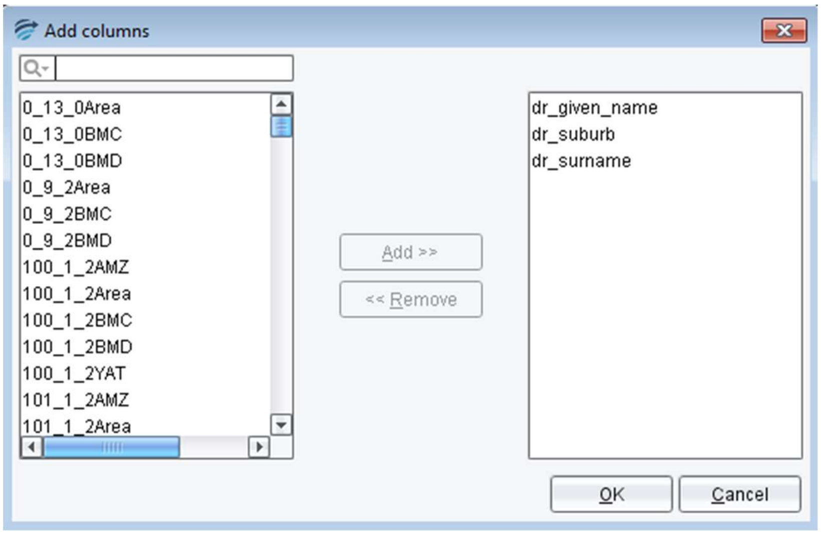
☑ Monday ☐ Tuesday ☐ Wednesday ☐ Thursday ☐ Friday

☐ Saturday ☐ Sunday

Month configuration

Day of the month 1 or ☑ Last day of the month

7. Select the ‘Add Condition’ button to add conditions.
8. Select the ‘Edit Columns’ button to edit columns, and select Attributes, Features and/or Set Conclusions.



9. Order columns using the “Move up” or “Move down” buttons if necessary.
10. Select either “Report on all episodes” or “Report on current episode only”.
11. Select the “OK” button to save the report definition.

Note: The HL7 Output Format function currently does not generate a report. Please use the Excel option for data export.

Housekeeping

Housekeeping of the system is performed automatically once per day and typically takes 15minutes to complete. The default time for housekeeping to begin is 2am local time, however this can be modified if needed. When housekeeping begins, all users will be logged off automatically and will not be able to log back on until housekeeping is complete.

Housekeeping tasks:

- The archive case lists will be purged of the oldest cases and the number of days selected per project to keep archived cases will be retained.
- New reports that have been approved will be removed from the New reports case list.
- Rejected case list will be purged of the oldest cases and the number of days selected per project to keep rejected cases will be retained.
- Backups of all projects including the RippleDownServer and InterpStore projects will be created and replace the oldest backups.
- Sends the log files for evaluation to a specified monitoring address.
- Check of the licence file.
- Update of all statistics.

Note: If a user attempts to log into the application while housekeeping is being performed, when the OK button is selected, the application will not open and the application selector will remain on the screen.

Help

The help website can be launched from the main screen and from within the various application components by selecting the **Help** button.

Appendix 1 – Syntax Summary

The table below shows some common condition expressions, grouped into categories. The complete condition syntax is in the Reference Guide in the RippleDown Help pages.

Category	Expression	Description
arithmetic	**absolute value of**	Takes the absolute value of a number.
	as integer	Rounds a calculation to the nearest whole number.
	+, -, *, /	Sum, difference, product and ratio of two numbers.
	to the power of	Raises one number to the power of another.
	ln	Calculates the natural (base e) logarithm of a number.
	log	Calculates the logarithm (base 10) of a number.
	exp	Calculates e (Euler's number) to the power of a number.
	sum across episodes of	The total value, across all episodes, of an attribute.
comparison	**<, <=, >, >=**	Compares two numerical values.
	__ < x < __	Compares a value (x) with two values at once.
statistical analysis	**max, min**	The maximum and minimum values for an attribute, amongst all episodes.
	mean	Average of the numerical values.
	baseline mean of first … … values beginning … days ago	Average of a “window” of values beginning sometime in the past.
	standard deviation	Standard deviation of the numerical values.
text analysis and manipulation	**is, is not, are, are not**	Compares text values, ignoring case. Also compares numeric values for equality.
	as one line	Converts multi-line text to a single line
	are all the same	Identifies sample sequences that all have the same value.
	has changed since last episode	Identifies sample sequences in which there is a change between the current and second-last values.
	contains, does not contain	Analyses text strings for presence or absence of substrings (case insensitive).
	starts with, ends with	Identifies samples that start or end with a particular text fragment (case insensitive).
	append	Appends the text value of one attribute to another.
	distinct values of	Appends together the distinct text values across episodes of a single attribute to produce a comma separated, alphabetically ordered list.
	list of ... values	Appends together the text values across episodes of a single attribute to produce a list separated by new lines.
	text after	Extracts the phrase after a matching phrase.
	text before	Extracts the phrase before a matching phrase.
	part ... of ... split by	For extracting elements from a list in which elements are separated by a fixed token.
	text between	Extracts the phrase between two phrases.
	text of ... between ... and ...	Extracts the phrase (substring) bounded by two positions in some text.
	character of ... at position	Extracts the symbol at a given position in a string of text.
	first match in... of ...	Extracts a phrase defined by a regular expression from some text.
	first match in... of ... calculated per-episode	Extracts a phrase defined by a regular expression from some text, with the regular expression itself a calculation.
	matches	Detects samples in which text matching a regular expression is found.
	matches ... with case	Same as **matches**, but case sensitive.
	exactly matches	Detects samples that exactly match a regular expression.
	exactly matches ... with case	Same as **exactly matches**, but case sensitive.
	to upper case	Re-writes some text in capitals so that it stands out in conclusions.
	as word	Converts small integers to the equivalent words for use in comments.

Category	Expression	Description
	as set	Converts comma separated values of an attribute set.
	fill from	Sets all episodes with the value from an attribute or constant.
reference range	**high, normal, low**	Compares sample values with their reference ranges from the external information system.
	upper reference value of	Refers to the upper value in a reference range provided by the external information system.
	lower reference value of	Refers to the lower value in a reference range provided by the external information system.
change since last result	**has increased by more than**	Identifies cases in which there has been an increase (absolute or percentage) for an attribute from the previously available value to the current value.
	has decreased by more than	Identifies cases in which there has been a decrease (absolute or percentage) for an attribute from the previously available value to the current value.
	has changed by more than	Identifies cases in which there has been an increase or decrease (absolute or percentage) for an attribute from the previously available value to the current value.
	has not changed by more than	Identifies cases in which the current value for an attribute is within some limit (absolute or percentage) of the previously available value for that attribute.
	percent increase from previous	Relative change between current and previous sample.
	percent decrease from previous	Relative change between current and previous sample.
trend	**has increased by more than x from each result to next**	For detection of cases in which the values of an attribute show a rising trend, overall results or overall results in a given time frame.
	has increased by more than x between each of last n results	For detection of cases in which the values for an attribute show a rising trend over a certain number of results.
	has decreased by more than x from each result to next	For detection of cases in which the values for an attribute show a falling trend, overall results or overall results in a given time period.
	has decreased by more than x between each of last n results	For detection of cases in which the values for an attribute show a falling trend over a certain number of results.
	net change in	Calculates the difference between the first and last samples for an attribute.
	percent increase from min	Relative change between current and minimum value.
	percent decrease from max	Relative change between current and maximum value.
	differences between	Differences between successive numerical episodes.
	differences from first	Differences between the numerical episodes and the first numerical episode.
	differences from average	Differences between the numerical episodes and the average of the numerical episodes.
	differences from max	Differences between the numerical episodes and the maximum value amongst the numerical episodes.
	difference from min	Differences between the numerical episodes and the minimum value amongst the numerical episodes.
	as percentages	Modifies the differences functions to give results as percentages.
	rate of change	Absolute or point-to-point daily or annual rate of change between selected episodes.
	yearly rate of change by linear regression	Annual linear rate of change according to the least-squares line of best fit.
counting	**all, each**	Identifies cases in which every episode satisfies a constraint.
	some	Identifies cases in which at least one episode satisfies a constraint.
	at least, more than	Identifies cases in which the number of episodes that satisfy a constraint is greater than some given number.
	exactly	Identifies cases in which the number of episodes that satisfy a constraint is equal to some given number.
	at most, fewer than	Identifies cases in which the number of episodes that satisfy a constraint is less than or equal to some given number.
	no, none	Identifies cases in which no episodes satisfy a constraint.
identifying a particular episode	**at**	Identifies a particular episode based on the values of an attribute.
	first	Refers to the first non-blank sample.
	most recent	Refers to the most recent non-blank sample.
	second most recent	Refers to the second most recent non-blank sample.
	next	Refers to the sample after the indicated one.
	previous	Refers to the latest non-blank sample before the indicated one.
	nearest	Refers to the sample closest in time to the indicated one.

Category	Expression	Description
	current	Used to refer to the current episode in complex conditions.
time since an incident	**days since (weeks, months, years too)**	Identify cases in which no episodes satisfying a constraint have occurred recently.
	days between (also minutes between, hours between, weeks between, years between)	Compare the times of two significant events.
	days between episodes	Calculates the number of days between successive episodes.
	has value of more recently than	Compare the times at which two attributes have attained a particular value (e.g. "true").
calendar calculations	**date at**	Determines the date of a particular sample and presents it in the form "DD/mmm/YY".
	long date at	Determines the date of a particular sample and presents it in the form "DD mmm YY".
	day of	Determines the day in the week of a particular sample.
	minutes at	Evaluates a sample as the number of minutes in the day after midnight.
	days from now, months from now	Calculates the date some days/months after the date of the most recent episode.
	is after date	Compares an attribute value with a fixed date in one of the two formats "DD/mmm/YY" or "DD mmm YY".
	as days	Used for finding the number of days between two sample values
	minutes after epoch using format	Used for finding the time between any two case values, in any format.
	minutes difference between … and …	Time in minutes between the values of two attributes.
	days between episodes	Calculates the days between each successive episode in the case
	fractional days (years) between episodes	Calculates the time between selected episodes.
	fractional days (years) between first and subsequent episodes	Calculates the time between the first episode and subsequent episodes.
	today	Today's date and time
formatting	**formatted to … places**	Formats a numerical output to a fixed number of decimal places
	formatted as …	Formats a numerical value according to some output pattern
	as date	Display the sample value as a readable date in the format dd mmm yy
	as time	Display the sample value as a readable time in the format hh:mm
	as datetime	Display the sample value as a readable time using a flexible date and time format
	in format	Display the date and time using a flexible date and time format
restriction clauses	**of latest ... values**	Restricts attention to the most recent episodes.
	… episodes ago	For getting a value from some previous episode
	previous	Restricts attention to the episodes before the current episode.
	where	Restricts attention to those episodes satisfying a constraint.
	within last	Restrict attention to recent episodes.
	older than	Restrict attention to older episodes.
	after	Restrict attention to episodes after some specified episode.
	before	Restrict attention to episodes before some specified episode.
logical operators	**and**	For combining conditions. Evaluates to true if both conditions are true
	or	For combining conditions. Evaluates to true if either condition is true
	if...else	For choosing which attribute's values to use, depending on some true or false condition.
	Whenever…Otherwise	For choosing between the samples of two attributes, on a per-episode basis.
sets	**{attributes}**	Returns all the primary attributes in a case.
	that equal	Selects the attributes in a case that have a specified value
	in range	Defines a set as those attributes whose values are within a specified range
	except	Defines a set as those attributes in the first set but not in the second set (i.e. set difference)
	intersect	Defines a set as those attributes that are in both the first set and the second set (i.e. set intersection)
	union	Combines two sets into single set containing all elements from both sets (i.e. set union)
	not in	Same as ‘except’

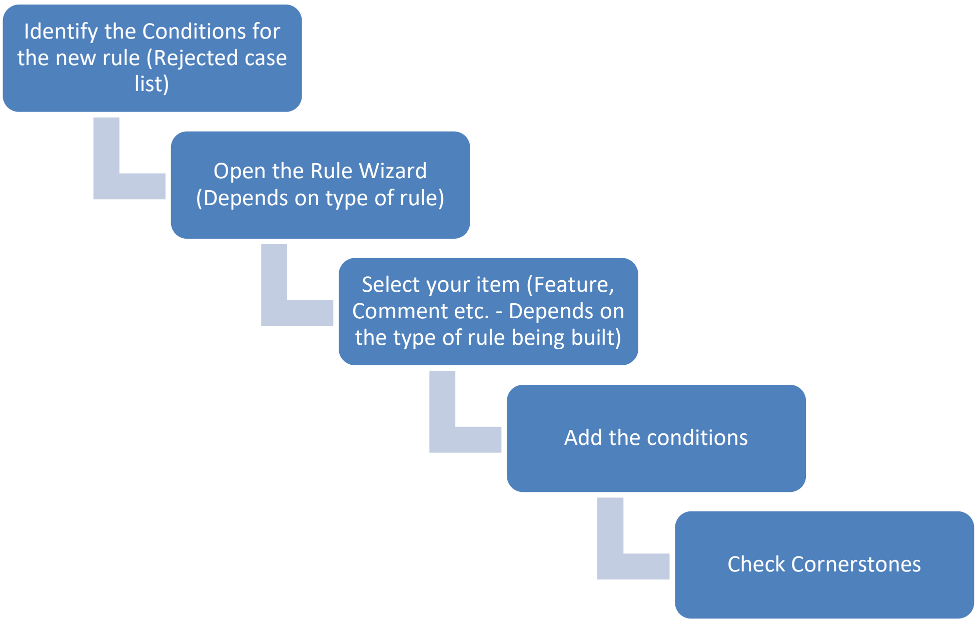
Category	Expression	Description
	for, for current	Selects a set subject to some condition.
	for which, for which currently	Selects the attributes from a set that satisfy a predicate that uses variable attributes.
	in range ... for	Defines a set with the 'in range' syntax, subject to some condition.
	number of	The number of elements in a set.
	sum of	The sum of the values of the elements in a set.
	includes, does not include	Whether or not a set contains a particular attribute.
	is empty, is not empty	Whether or not a set contains any elements.
	with some abnormal value	Collects together those attributes that have had an abnormal value.
	contains only high (low, normal) values	Detects cases in which every attribute in a set has high, low or normal values.
	contains only values within … percent of normal	Detects cases in which every attribute in a set has a normal value, to within some percentage tolerance.
	as groups from	Displays a group name for each attribute in the set, rather than the attribute name.
	as names	Displays just the names of the attributes in a set, rather than the names and values

Appendix 2 – Rule Building Workflow (Reference Guide)

There are 5 simple steps to creating a rule in RippleDown.

1. Identify the Conditions for the new rule
2. Open the rule wizard using an action within the area you wish to modify
3. Select or create the item to be used within that area
4. Add the conditions
5. Check cornerstones

The diagram shows the 5 simple steps to follow to create a rule using RippleDown. The procedure for each step is described below.


Identify the Conditions for the new rule (Rejected case list)
Open the Rule Wizard (Depends on type of rule)
Select your item (Feature, Comment etc. - Depends on the type of rule being built)
Add the conditions
Check Cornerstones

Steps to building a rule in RippleDown

1. **Identify the conditions for the new rule**
   - A rule can only be built if the selected case satisfies all the conditions that are to be included in the rule.
   - Identify the conditions to be used for the new rule. If the rule is being built on a case from the rejected cases list, then the Validator may have already listed the issues. If not, find or create (in the “Favourites” case list) a case that contains all of the conditions for the rule to be build.

2. **Open the Rule Wizard by selecting an action**
   - Open the rule wizard by right clicking on the area that the rule is to be built on. These areas include Features, Derived Features, Report Sections, Queues and Autovalidation.
   - Select the action Add, Replace, etc.
   - The Rule Wizard will then open.

3. Select the applicable Item (Feature, Comment, etc.)

- Depending on where you have selected to create your rule, you will need to select an applicable item. These items include a feature, derived feature, comment, queue, etc. If the required item does not already exist, you must create a new item. Once the correct item has been selected, **Click next.**

4. Add the Conditions

- Add the conditions to be applied to the rule.
- If the syntax for the condition is known:
  - Write the condition in the current window which is located in the middle of the left hand side of the screen.
  - Click on Add.
  - The Wizard will then add the condition to the Completed Window.
- If assistance is required with syntax for the condition:
  - Select the tab which contains the attribute, feature etc. that is to be used for the condition.
  - Select the attribute, feature etc. for the condition.
  - Click on Next.
  - Select an expression from one of the 6 available tabs. Complete the expression if needed. Click Next.
  - The Wizard will then show you the condition that was added and also add the condition to the Completed window. Click Next.
- Repeat the previous steps to add multiple conditions to the rule.
- A list of condition expressions is available in the online help. To find this list, search for the word “Summary”.

5. Check Cornerstones

- The total number of cornerstone cases which will have their interpretations changed by the rule being configured, is shown at the bottom left hand side of the rule wizard.
- To view the cornerstone cases select **Show…**
- The Cornerstones window will open. Each cornerstone case that will be affected by the current rule will be shown.
- Apply the change to all applicable cornerstones by clicking “Apply change to cornerstone”. If there is a cornerstone case that the rule should not to be applied to, click “Do not apply”. The Cornerstones window will close. You must add more conditions to the current rule to distinguish it from the cornerstone case.
- Once all cornerstones have been confirmed or excluded, the Total cornerstones should equal 0.
- Click on Finish to save the rule.
- The rule will now be applied to the current case and all other cases with the same conditions henceforth.

Using rules to perform actions

The steps to rule building can be used to apply many different actions in the knowledge base. A list of the actions is below:

- Add/Replace/Remove a feature
- Add/Replace/Remove a derived feature
- Add/Replace/Remove a queue
- Add/Replace/Remove an autovalidation setting

- Add/Replace/Remove/Re-add a comment to a report section
- Enable/Disable a report section
- Add/Enable/Disable/Replace a result overwrite
- Enable/Disable a report item

Project Explorer

The project explorer keeps a record of all rule building activity. It is separated into 7 sections which are listed below:

- Report Sections
- Derived Features
- Features
- Autovalidation Settings
- Queues
- Conditions
- Rule Sessions

Each section will show the User, time and date, description, and conditions for each rule that has been configured.

Appendix 3 – Defect Summary

Known defects in this release

Reference	Summary	Description
PRD-3937	Unable to see the conditions giving an internal comment	**Description:** In the thin client Validator or Auditor, internal comments do not show in the Conditions panel, so you can't see why an internal comment is given. **Evaluation:** Evaluated as Low. All cases with internal comments are held for manual validation. Internal comment typically describes what action is required indicating the reasons for its addition. No impact to patient reports.
PRD-2637	Report Section format is not displayed in preview pane if comment is edited directly beforehand	**Description:** The report section format does not display correctly if the following occurs: • Within a particular report section: ○ a comment is modified by the knowledge builder and is not in the top position of the report section ○ a comment position is changed by the knowledge builder • The display format for the report section that contains the edited comment is then modified The Report in the Preview pane will not reflect the newly selected report section format. All comments within the modified report. The format will only be displayed correctly once the daily housekeeping is performed. **Evaluation:** Majority of report sections are defined during initial configuration of knowledge base and are very rarely changed. If the issue occurs, report section Heading will display correctly, however the comments will not reflect the change. No impact on patient report as the correct comments and report is generated. Minor impact on user experience.
PRD-3507	HL7 LABQ does not generate report	**Description:** Using HL7 definition does not export any data from LABQ. **Evaluation:**

		No impact on patient reports. Slight impact on user experience. No existing customers currently using function. Issue will be included in labeling.
PRD-3520	Validator queue screen is also shown when show conditions is selected - V8	**Description:** While reviewing cases in a queue, if the comments tab is selected and "show conditions" is selected for a particular comment, then the validator queue window is also shown along with the conditions window. **Evaluation:** The conditions window is displayed to the user giving them the information required. The Validator queues window is just annoying not prohibitive. Reports are unaffected.
PRD-3519	Level issue with episode attribute condition should display an error message.	**Description:** If an episode attribute is being edited with a condition using an attribute at the same level, in normal circumstances, this will increase the level of the episode attribute. If an attribute is edited to use other derived attributes that increase its level, these edits may not be saved if its level change impacts existing other derived attributes. That is expected behaviour to prevent a circular dependency, but no warning is displayed to the user. This is not seen in newly created derived attributes where all levels are managed by the application. **Evaluation:** Even though the condition is allowed to be added and evaluated, once the episode attribute is saved, the offending condition is then correctly removed automatically from the episode attribute. No impact on patient reports as example cases must be used to build new rules. There are also multiple opportunities during the rule building process that will ensure the rule builder checks all conditions applicable for the new rule. Slight impact on knowledge builder experience. Information on level restrictions is given during training.