Pacific Knowledge Systems

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

The information provided in this User's Guide is subject to change without notice and is not a commitment by Pacific Knowledge Systems Pty Ltd. The software described in this User's Guide is provided under a license or non-disclosure agreement. It is unlawful to copy this software except as allowed in the agreement.

No part of this User's Guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information retrieval systems, for any purpose other than for the purchaser's personal use, without the written consent of Pacific Knowledge Systems Pty Ltd.

Reproduction or disassembly of embodied programs or databases that make up the software is prohibited.

© Copyright Pacific Knowledge Systems Pty Ltd, 2019

All Rights Reserved.

Intended Use

The PKS Software, when used as a medical device, 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 (hereafter referred to as "domain experts").

The PKS Software generates and collates comments into a patient-centric report or workflow action based on rules created and maintained by domain experts. The PKS software presents all first-of-a-kind reports for review, modification (if applicable) and approval by the domain expert prior to release to clinicians or other individuals. Beyond this, the domain expert may automate the release of none, some or all identical reports. However, since a report may be generated and automatically approved for a case for which the domain expert has not previously considered, it is intended that the domain expert regularly review a representative sample of all output types.

It is intended that the clinicians receiving reports against 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 on the existence of a report for the management of a patient.

The PKS software in itself does not provide automated diagnosis or treatment-making functions or have the capability to control, in any way, the performance of a device or to treat or diagnose any disease.

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.

"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.

Acknowledgements

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

Action	The recording of an error against a case when auditing
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 an intermediate characteristic
Auto-validation	Release of reports 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.
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 guiding the rule building process.
Validator	Application used to review reports prior to release.

Overview

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.

Clinical Workflow
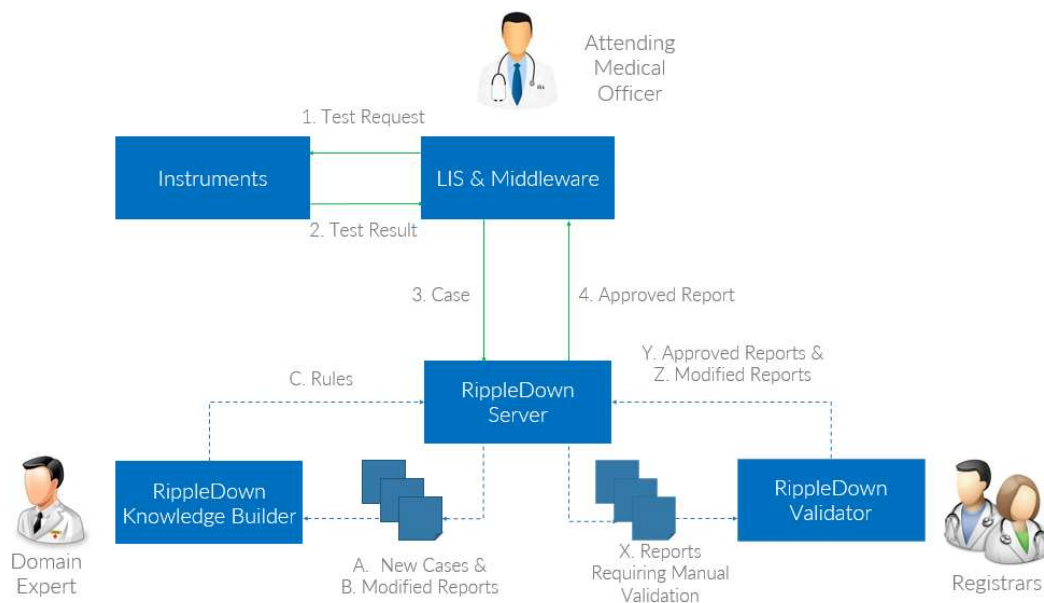
Each step in the above figure is described below:

1. An update to the patient record is made in the Primary Data Source (e.g. patient admitted, Physician orders test, Instrument results test).

2. The Primary Data Source requests an interpretation of the patient Case from the RippleDown Server, ideally sending all available, relevant, current and historical data for the patient.

3. If necessary, prior to interpreting the Case, to enhance the accuracy and precision of the interpretation the RippleDown Server can enrich the patient Case with additional data from Primary or Secondary Data Sources.

4. The RippleDown Server interprets the Case using its configured rules to create a Report (being a combination of Comments and/or Actions). If eligible for automation, the RippleDown Server returns a Report to the Primary Data Source in real-time. If not eligible for automation, the RippleDown Server holds the Report until manually approved (see X. Reports Requiring Manual Validation).

5. The Primary Data Source accepts the Report, stores the Comments against the patient record and executes any Actions. The Physician sees an enhanced patient record through the Primary Data Source without disruption to their user experience or test turnaround times.

X. Reports not eligible for automation are sent to the RippleDown Auditor / Validator for review, modification (if necessary) and approval by Auditors / Validators.

Y. Approved Reports are returned to the RippleDown Server and immediately forwarded to the Primary Data Source (see 6. Approved Report)

Z. Modified Reports are returned to the RippleDown Server and immediately forwarded to the Primary Data Source (see 6. Approved Report) and forwarded to the RippleDown Knowledge Builder for rule refinement (see B. Modified Reports).

A. New cases are sent to the RippleDown Knowledge Builder for the Domain Expert to consider writing rules against.

B. Modified Reports are sent to the RippleDown Knowledge Builder to consider refining the existing rules (i.e. continual improvement).

C. The Domain Expert creates and modifies rules in the RippleDown Knowledge Builder, which automatically checks for any impacts to existing rules. New and modified rules are then returned to the RippleDown Server and applied immediately and safely.

Business workflow

The figure below describes the key activities within the RippleDown-enabled process:

1	Test Request	An Attending Physician requests one or more tests to be conducted through the clinical data source (e.g. Hospital Information System (HIS), Laboratory Information System (LIS)). Potentially a Middleware system orchestrates the Instruments to conduct the requested test. If External Integration is being used, the test request may also be sent to it.
2	Test Result	On completion of the test, the result is returned to the clinical data source. The test result is available to the Attending Physician. If External Integration is being used, the test result may also be sent to it.
3	Enrich Case	If External Integration is being used, for each test request and test result it receives, it may request relevant historical test requests, test results and other patient data (e.g. age, clinical notes, medication history) in order to create the patient case.
4	Case	The clinical data source or External Integration collates the test request and/or result with relevant historical test requests, test results and other patient data (e.g. age, clinical notes, medication history) into a case and sends this to the RippleDown Server for interpretation.
5	Approved Report	The RippleDown Server compares the conditions of the case with the conditions recorded against stored cases. Where there is a match, it applies the corresponding recorded output into a report. When the case has been compared to all stored cases, a report is generated and assessed against auto-validation rules. If the generated report is eligible for auto-validation as per the Pathologists / Clinicians applied prior configuration, an approved report is released to the clinical data source (potentially through External Integration if it is being used) else the report is sent to the Validator application for manual approval.
X	Reports Requiring Manual Validation	Any first-of-a-kind report is sent to the Validator application for manual validation. The Pathologists / Clinicians may also scale the auto-validation configuration for any previously generated report to reflect their confidence in the appropriateness of the report. For example, a report with an auto-validation setting of 90% equates to 1 in 10 cases that receive the report being sent to the Validator application.
Y	Approved Reports	Pathologists / Clinicians or their delegated Laboratory Managers review reports requiring manual validation. If they are satisfied with the report they may approve it, which releases the report to the clinical data source (potentially through External Integration if it is being used).
Z	Modified Reports	Pathologists / Clinicians or their delegated Laboratory Managers review reports requiring manual validation. If they are not satisfied with the report they may modify it, which releases the modified report to the clinical data source (potentially through External Integration if it is being used) and sends a copy of the case and modified report to the Knowledge Builder application for the Pathologists / Clinicians to review and potentially remediate the rules.
A	New Cases	Copies of new cases and their reports are available to the Pathologists / Clinicians through the Knowledge Builder application to review and configure new rules and reports against.

B	Modified Reports	Copies of cases whose reports have been modified during manual validation are available to the Pathologists / Clinicians through the Knowledge Builder application to review the reason the original report was not accepted and potentially remediate or refine the rules to generate the modified report (or a new report if appropriate).
C	Rules	New rules, represented as cases, conditions and reports, are returned from the Knowledge Builder application to the RippleDown Server. Any new or modified rule is compared to stored cases and their rules to ensure the Pathologists / Clinicians are prompted to accept any impacts the new or modified rules on stored cases and their reports.

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. 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.

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

- Patient demographics
- Episode details
- Test codes and results
- Historical results

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

Attributes

A case in RippleDown consists of a set of attributes each with a sequence of sample values reflecting current and historical results. There are several different types of attributes:

Primary attributes represent the "raw" data sent to RippleDown by the Online Information System/s. The integrity, quality and accuracy of the primary attribute values received into RippleDown is completely dependent on the Online Information system. All rules executed by RippleDown are solely based on these values. It is imperative that the Online Information System/s is used, managed and maintained in accordance to the manufacturer specifications to ensure the accuracy of the attribute values.

Derived attributes can be created from the primary attributes to capture examples of key concepts.

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 value attributes** 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. Go to Edit | Features... to build your features.

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.

Comments

Comments are added to a specific report section through rules. Go to Edit | Comments... to build your comments.

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.

Please note: It is imperative that all project components are given unique names.

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
- Text values in clinical notes or other fields

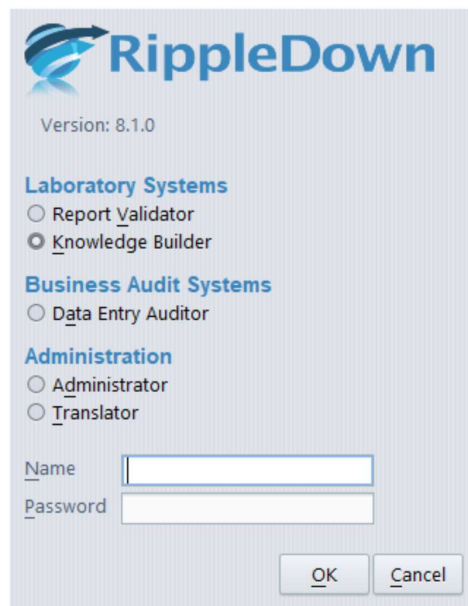The full condition syntax is provided in the RippleDown Help utility.

ALL conditions in an expression must be met for a case before a rule is 'true' (or is valid).

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 screen shown below ensure the Knowledge Builder is selected:

3. Enter your username and password and click OK (For creation of login details please contact your RippleDown Administrator)

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.

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 Attributes, Features, Report Sections, Comments and Queues
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 andSimulation 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 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	The amount of cases retained is configurable in **Options	Settings	Days to keep archived cases**. The default is set at 90 days, however, it is recommended that no more than 100,000 cases are kept for any individual project. Cases in the archive cannot be deleted manually.
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 / folder name.

The next column displays the **reference ranges** (if available) for the analyte attribute in the current episode.

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 is the right-most column.

There is a way to filter and search for a desired attribute or result in the case view. In the top right-hand side of the screen is a filter box that will filter the case view with the characters that are entered and only show the attribute names and results that match the entered criteria.

Information / Action Panel

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

Comments	The pane used to add comments and workflow 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 and so on.

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 function. Go to File | Find (Ctrl + F). Alternatively, click on the search icon. This search is not case-sensitive. The first attribute that matches your search will be highlighted. To find the next matching attribute (and any subsequent ones) use the F3 function key.

Searching for attributes within the Attribute editor can be achieved by clicking on the search bar and typing the name of the desired attribute. This search is not case sensitive. The attributes matching the search criteria will filter to the top of the list.

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.

Other case view properties of existing attributes can be customised. Right click on the attribute and select Attribute Display (or click on the attribute and press CTRL+Y):

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. A colour can be chosen from the drop-down menu. The default 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 in the case view however may still be used in creating rules	
Format	Default	The attribute will not show in the body of a case if all of the samples of 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 defined as a patient with an age in days between 7 and 365 days:

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 is one from the list below:

From the Case View it is possible to add elements to a pre-existing list. To do this right click on the sample value, select 'Add value to' and select the appropriate List derived attribute.

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.

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.
- The reference range should not be left 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:

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 the word "diabetic" in the clinical notes. Facilities are provided to refine the search so as to exclude similar but unwanted words or phrases, e.g. not diabetic.

 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.

In the above example if the Requesting Dr code is H546 and the Provider number is I06883 or I8764, the attribute will evaluate as true

 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.

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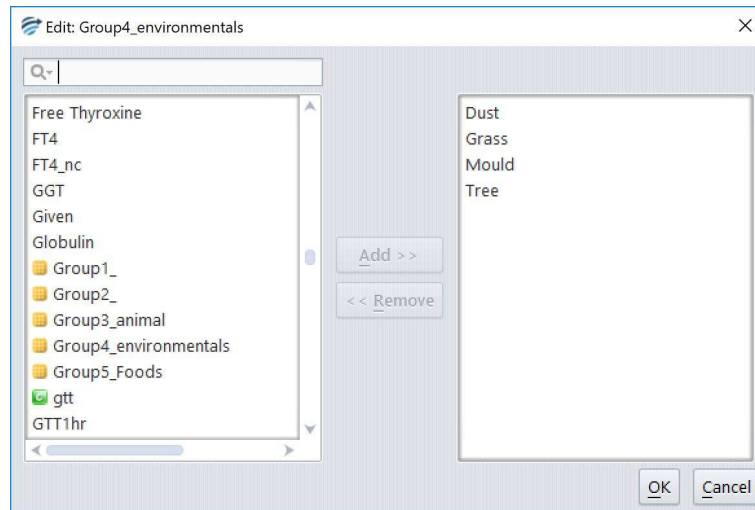
For example, we can define the following groups of allergen attributes:

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 or one of a number of defined conditions apply. Along with primary attributes, the conditions can include other derived attributes. These are useful in identifying clinical patterns. For example, suppose that we need to identify patients who have had at least one abnormal lipid test or all lipid tests are normal. Examples of these are below:

Note: 'Some_Lipids_Abnormal' will evaluate as true when the "At least one condition is true" radio button is selected AND any of the 4 lipid components above are not within the normal range.

Note: 'All_Measured_Lipids_Normall' will evaluate as true when the "All conditions are true" radio button is selected AND all of the lipid components performed were within the normal range

Attribute Mapping withing RippleDown

RippleDown will automatically add any new attributes received by the data source to the relevant project. The attribute mapping for all projects can be found in RippleDown as follows:

1. Login to the Knowledge Builder module.
2. Open the desired project.
3. Go to '**Edit | Attributes**'.
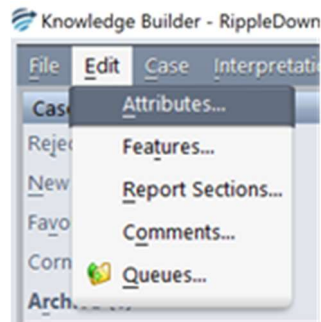
4. The attribute editor will display all attributes ever received by RippleDown for the installation instance. The display name within RippleDown will be shown.
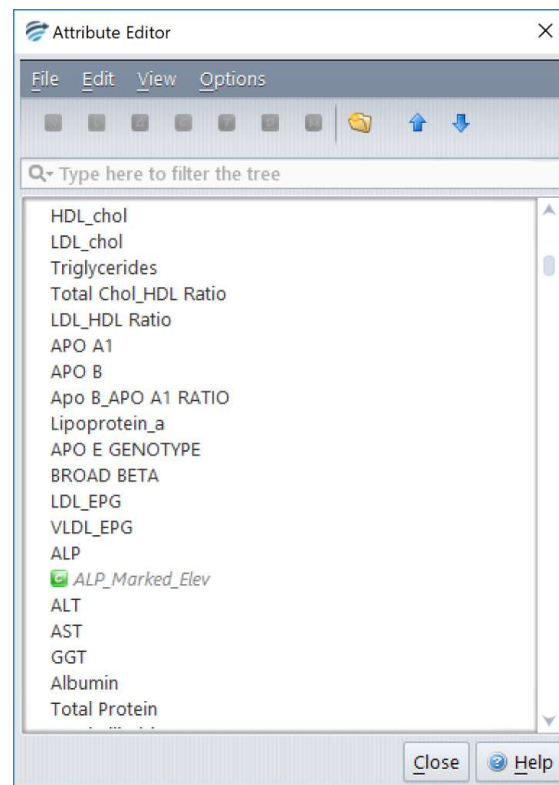
5. From the Options menu select "Show external names".
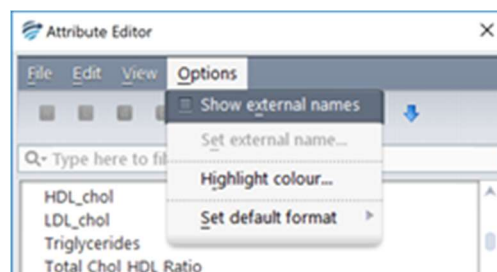
6. The external name mapped to each attribute will now be shown beside the display name.
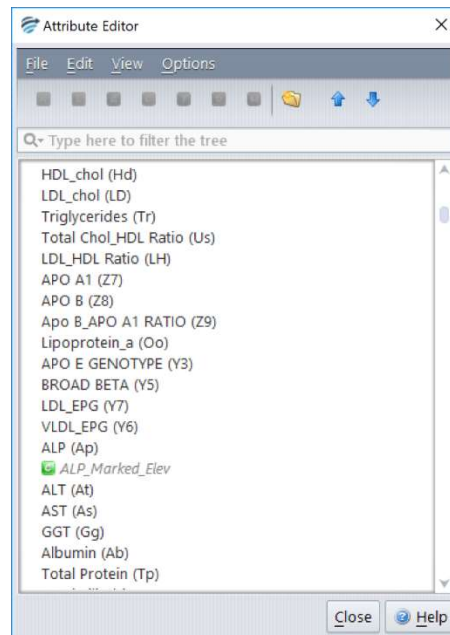
7. The external name of an attribute can be changed by first selecting the attribute, then selecting "Set external name" from the Option menu.
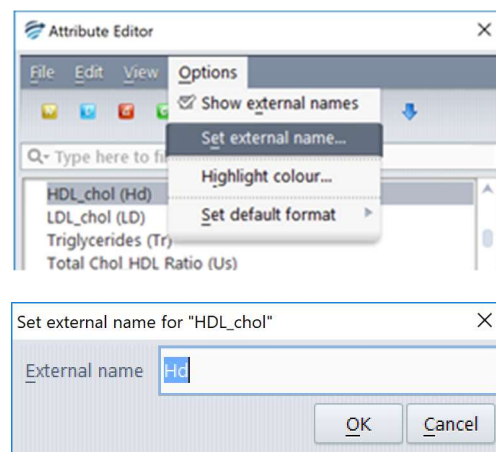
Features and Derived Features

Features

Features are high-level concepts that you can define (with rules) to aggregate concepts that will be common to many case types. Further use of these concepts makes subsequent rule building easier. A feature can be used as a condition in rules that adds comments to the interpretive report. The feature itself does not appear in the report.

Features are indicated by the grey light globe icon:

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" for example can be added, when the 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

Creating Features

To create a Feature, select Add Feature in the right click menu from the Features box in the lower left corner.
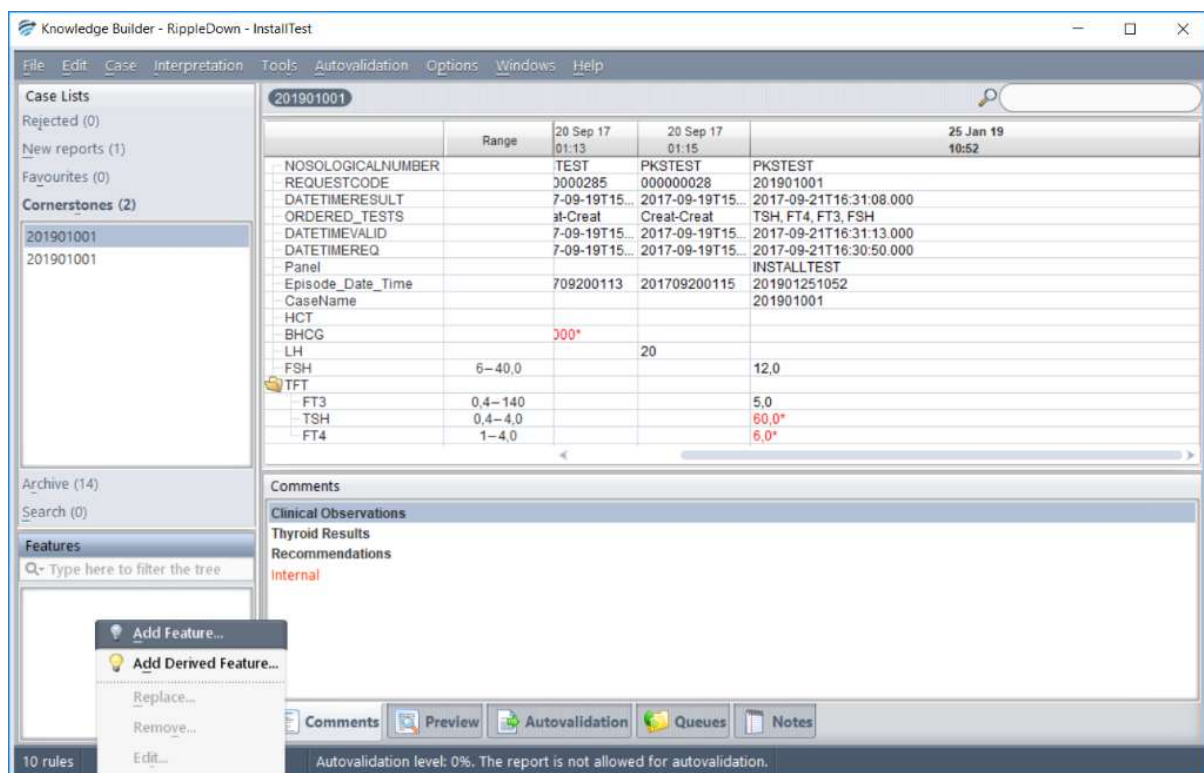
This will open the rule wizard where the user can select an existing feature or create a new feature. Following this step, the user will then be required to add the specific conditions for this rule like they would for any other rule.
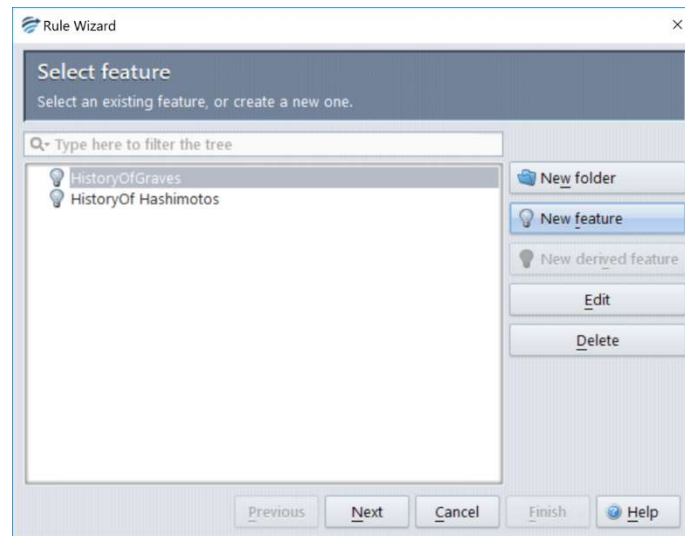
Derived Features

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 when the other three are given, thus simplifying the rule conditions.

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…".
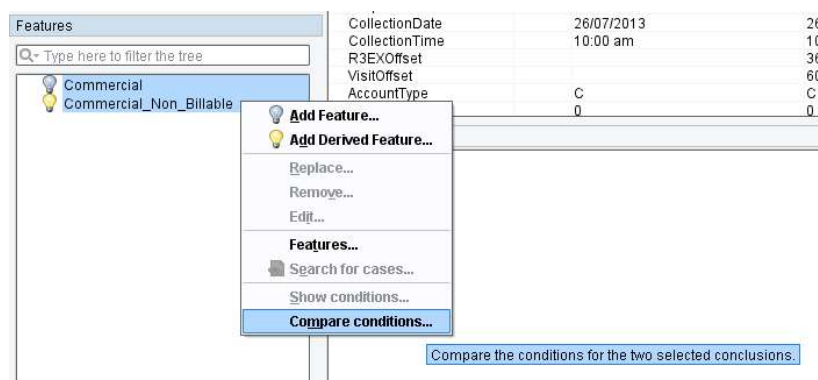
Feature and Derived Feature Organisation

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 by using Drag and Drop.
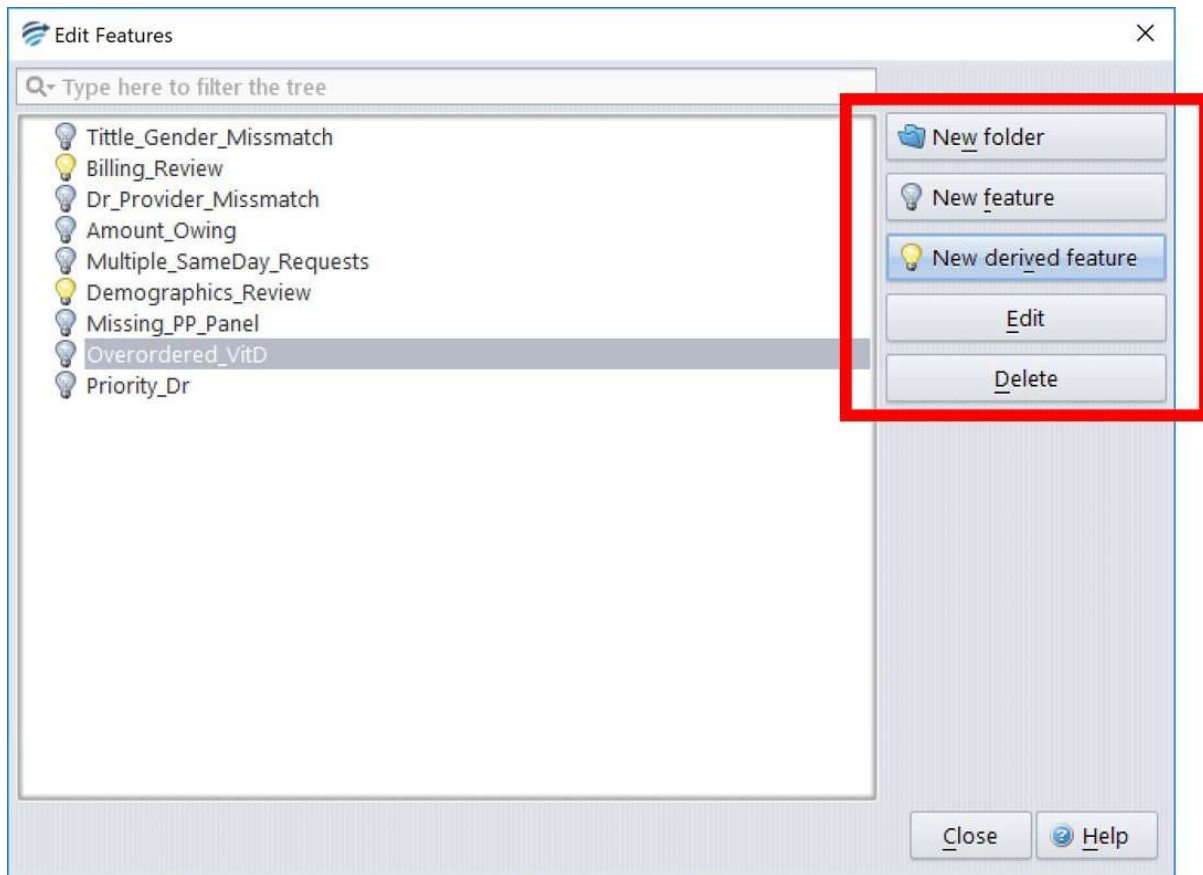
Rules and the Rule Wizard (See Appendix 2 - Reference Guide)

A rule is a set of conditions put in place to create an action, for example to add or delete 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 to guide the rule building process and updating the knowledge base using the current case. 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

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	▪ Adding a review queue ▪ Replacing the selected review queue by another one ▪ Removing the selected review queue

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

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:

The list of conditions to be applied

These fields are used to add conditions

Cornerstones

Search Bar

The Differences List

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)
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 which will place the condition to the completed pane.

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
- Report Sections
- Comments (only appears in the Queues Rule Wizard window)

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 is 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 record of cornerstone cases that help prevent rules having unintended side-effects.

Cornerstone cases affected by the conditions are presented as 'Impacted cornerstones'. The total number is shown in the Cornerstones: dialog.

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

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 affected if the rule is applied can also be seen by clicking the 'Explain cornerstone' button..

Once all Cornerstone cases have been reviewed and the Cornerstone count is shown as 0. The Rule Wizard screen is shown again. Click Finish to apply the rule.

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. By right clicking on the conditions on the right-hand side of the screen the user can select to edit these conditions.
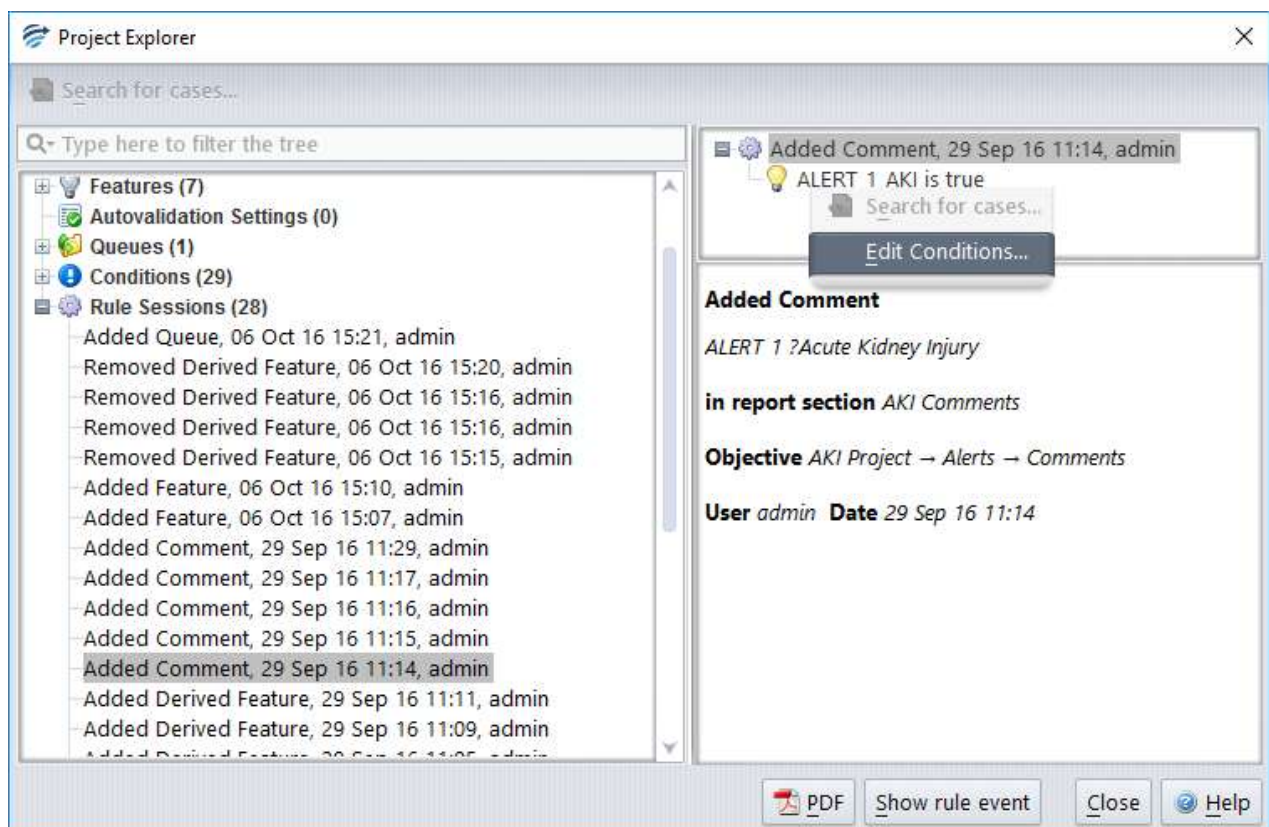
The following window will then appear, giving you the option to add, remove or edit the conditions.
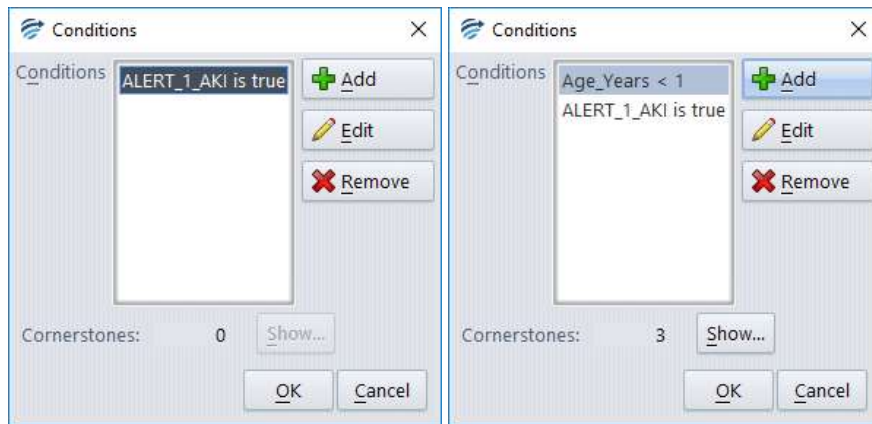
Similar to writing the rule in the Rule Wizard, as the conditions are edited, cornerstone cases will be compared against to identify examples that would be impacted by this change. Once the edited conditions are completed then the user can select the 'Show' button and review the cornerstone cases that 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. For full syntax see Appendix A

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

Table 1 shows some results for a patient:

	Range	04 Sep 11 08:47	20 Aug 12 08:05	16 Feb 14 07:40	27 Mar 18 09:00		17 May 18 09:30
FastingStatus				Non Fasting	Fasting	Fasting	
Therapy			Y		N		
CLINICALNOTES			?DM			DIABETIC, ON STATIN	
Glucose	3.0−5.4		6.3*	5.9*	5.8*	5.2	
HbA1c	3.5−6.0		5.1			5.6	
Cholesterol			6.8*	5.0	5.8*		
CaseName		88461234	88461234	88461234	88461234	88461234	

Previous Results — Current Episode

Table 1: Current and past results for a patient

Table 2 shows examples of simple syntax based on the above results.

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	Glucose is normal
	Is low/is high	
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	Some Glucose ≥ 6.2
	None/no	No HbA1c is high
	At most,	At most 3 Glucose are high
	At least	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"

Table 2: Syntax examples that could be applied to the case in table 1

Reports

A "Report" is the ordered set of comments provided by the Knowledge Base for a case and may be divided into report sections, each containing none, one or more comments.

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

1. A Report Section called "Comments" and 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:
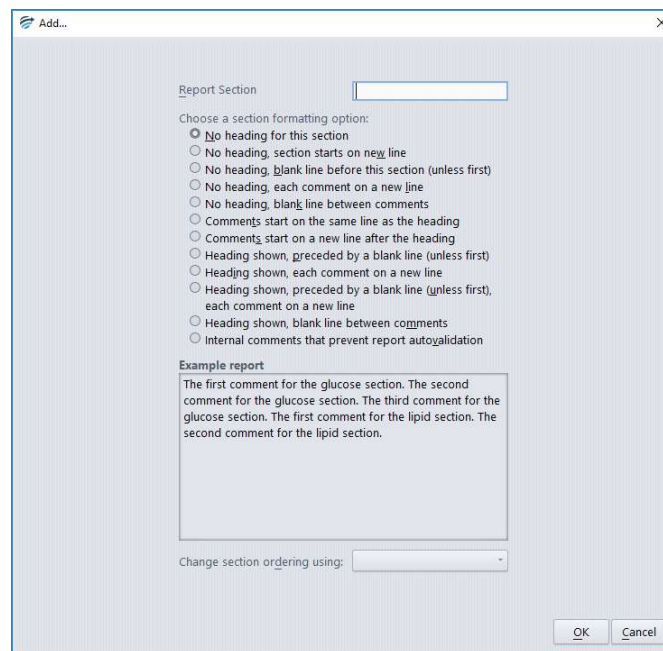
Here it is possible to name the report section and to select one of the predefined format options. An example of the format type is shown in the lower 'Example report' portion once an option is selected.

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 detailed "comments" view or in "preview" mode. Preview mode allows viewing of the interpretation as it is presented to the Validator or Online Information System.

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 user specific 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, a report section that is currently disabled can be enabled.

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 to an existing comment
- Replacing one comment with another
- 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 create a new one.
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.

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 (see next section)
- Naming or renaming a comment
- 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
- A filter for searching 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 from the case.

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.

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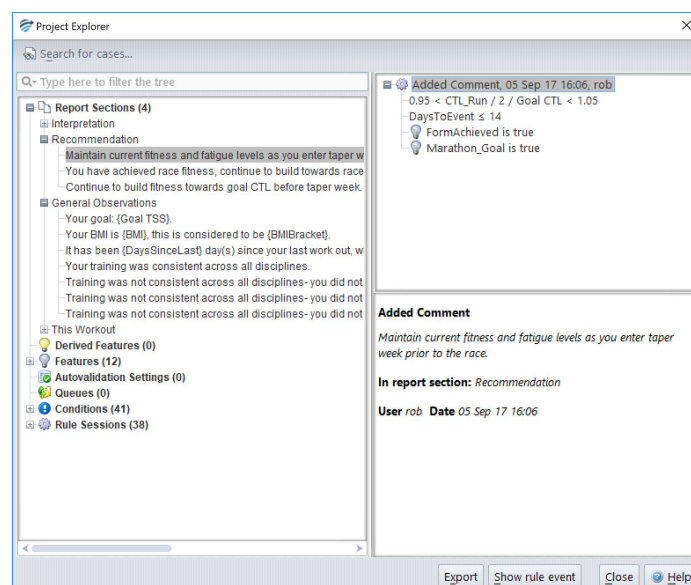
In the upper right pane it shows the history of the comment and all the reasons for it being added or removed.

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 a 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 track 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 can be sorted and in the top right hand corner there is a free text search box.
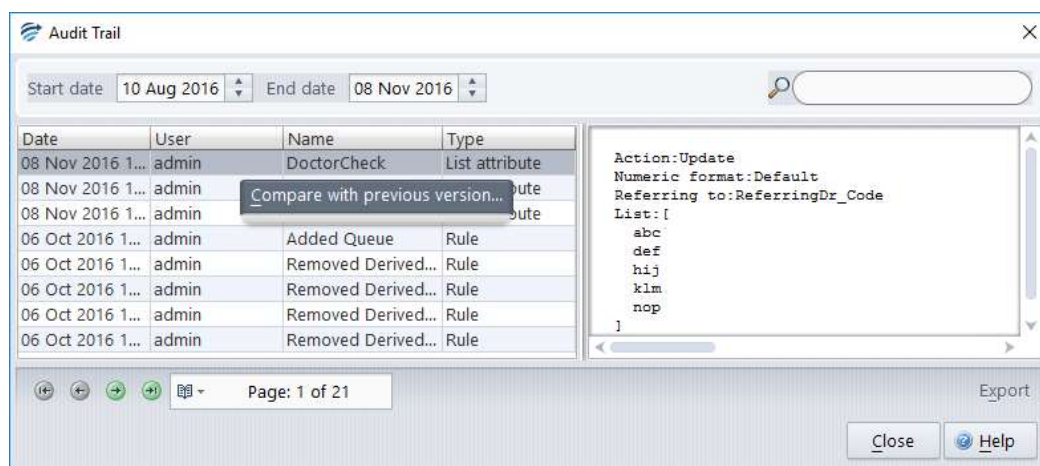
Once a desired entry has been located, a list derived attribute in this example, in order to drill down and compare the current version and the previous version of the list simply right click.
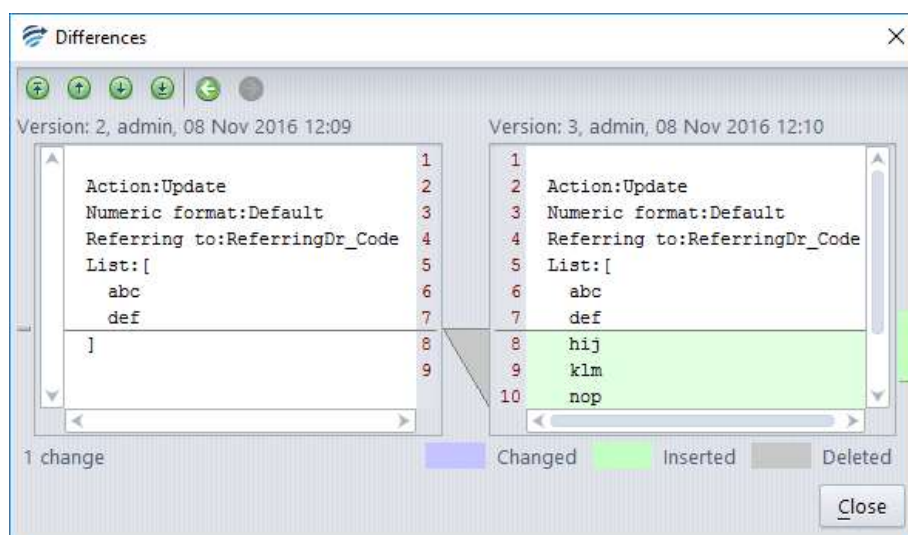
Changes are clearly highlighted so you can see in this example that three new ReferringDr codes were added to the list attribute at 12:10 on November 8th by the user: admin.

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; by reports, by report sections or by rules.

1. On a "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. On a "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 makes use of the rule wizard to add, modify or remove auto-validation settings that are given by the project, in exactly the same way the user would add, modify or remove comments using rules. This is the most flexible approach.

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

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. Using the Preview pane on the Knowledge Builder and selecting the 'Allow' button

b. Opening Auto-validation/Report Settings and double clicking the report to open the screen shown below. Click the checkbox to allow auto-validation:
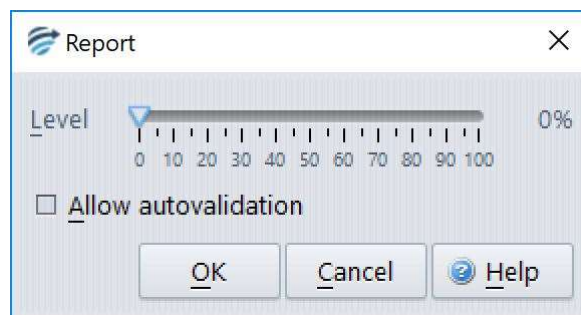
c. Using the Validator to approve the case through normal lab workflow (this will only apply if the user has Knowledge Builder priviledge)

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.
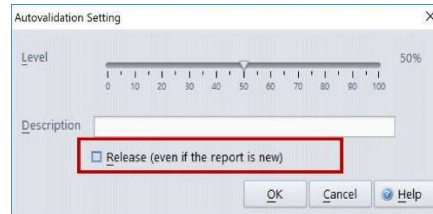
Auto-validation Options
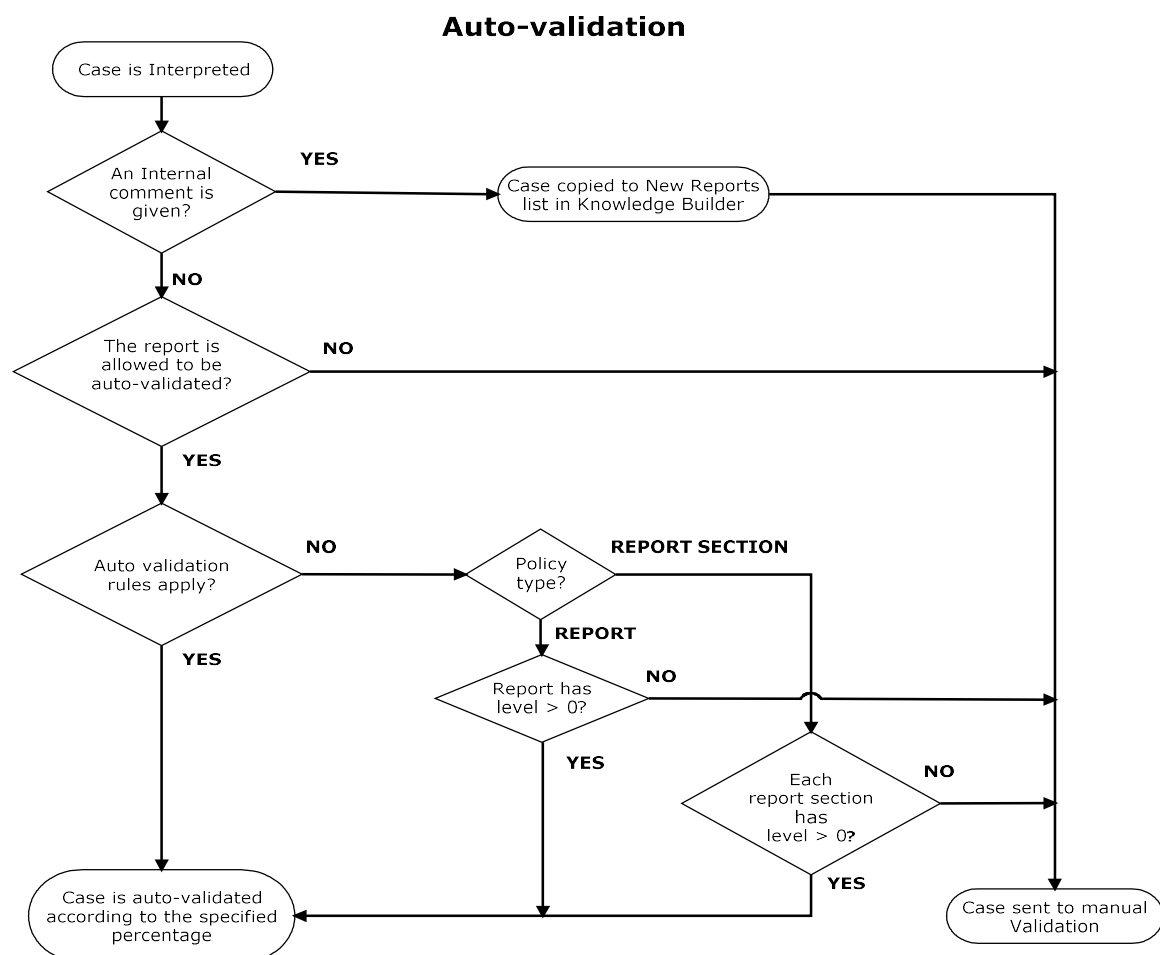
Auto-Validation levels

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: Normally, it is advisable 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 a continuousquality audit of the knowledge base.

Policy

The choice between the "by report' and "by report section" policy is done using the menu Autovalidation | Policy.
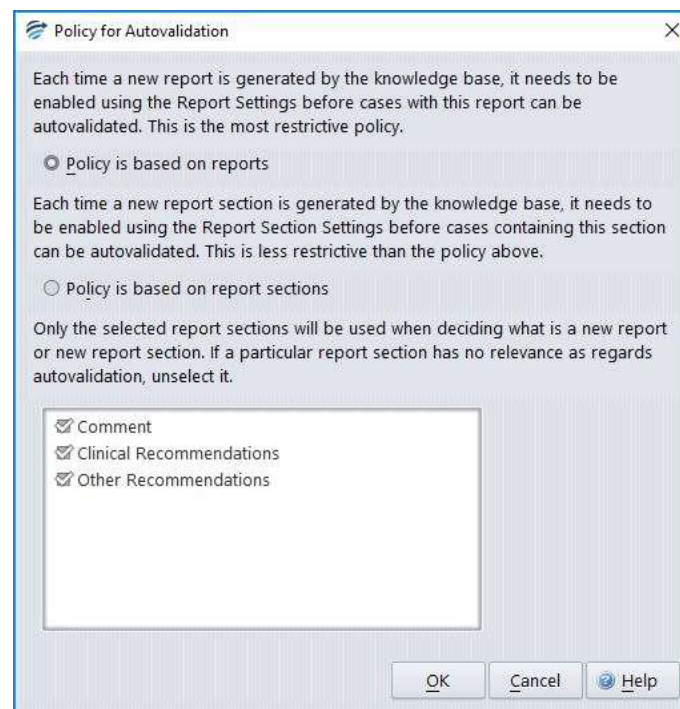
The default setting for a project is "Policy based on reports" with all report sections being selected.

Autovalidation 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 'by-report' basis:

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

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

Autovalidation rule settings

Release	Level	Description	Case	Given	Manual	Reviewed	Rejected
✔	30%	Likely false - positive alpha Transg...	●	0	0	0	0
✔	20%	Coeliac Disease is Unlikely		0	0	0	0
	10%	Order IgA		0	0	0	0
		Order tissue transglutaminase IgA		0	0	0	0
		Order tissue transglutaminase IgG		0	0	0	0

Buttons: Add, Edit, Search, Rules

Full description
Likely false - positive alpha Transglutaminase

In the last 31 days
Total: 0 | Manual: 0 | Auto: 0 | Reviewed: 0 | Rejected: 0 | None: 0 | Search

Close | Help

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.

Autovalidation 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.

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

1. Set the policy to 'Policy is 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, the user must add an auto-validation setting to a case using the rule wizard, in exactly the same way a comment would be added 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 be safely 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. 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 a comment would be replaced for a case, giving only the conditions relevant to the change.

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

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

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.

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, the following possibilities for reviewing for the case are available, 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. 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.

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.

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 from backup

It is possible to search for a backed-up copy of a case that has been interpreted by a project. To do this, activate the 'Archive' case list and select 'Search from backup...'
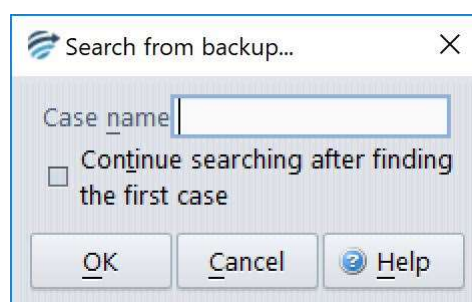
A dialog will appear, prompting you for the name (lab number) of the case. When you click 'OK', the server begins the search as a background process. A progress bar is shown at the bottom left hand corner of the screen to indicate how many of the backup files have been searched. As the cases are found, they are put into the Search case list.
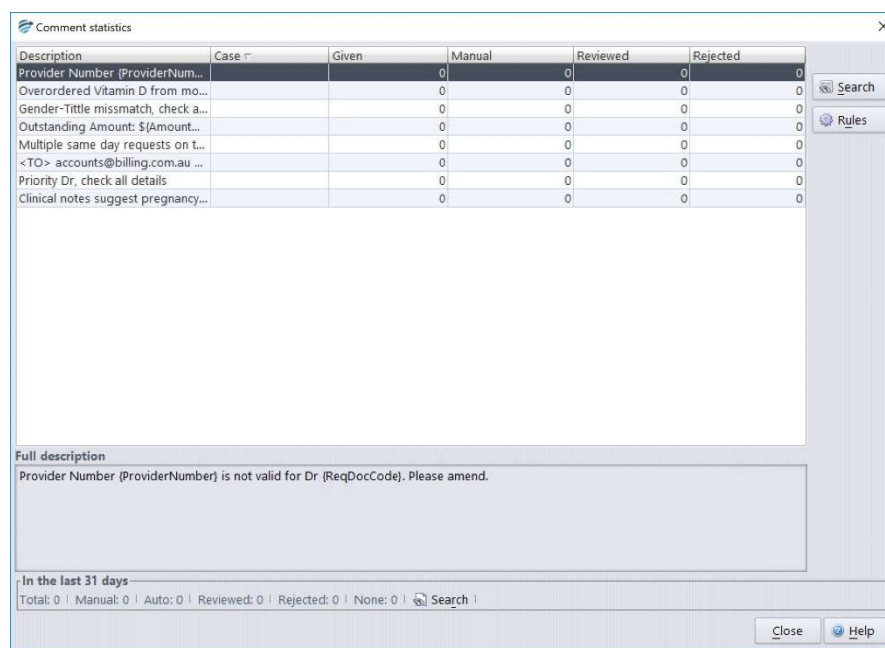
Sometimes a case is sent several times by the online information system, over the course of a few days. If you want to see all of these versions of the case, choose the option 'Continue the search

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.

The statistics for each comment are:

- Given: the number of cases with this comment that have been given by the project
- Manual: the number of cases with this comment that have been manually validated
- Reviewed: the number of cases with this comment that have been reviewed by the validator
- Rejected: the number of cases with this comment that have been rejected by the validator

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

The **Search** button will activate a search facility for the comment that is highlighted.
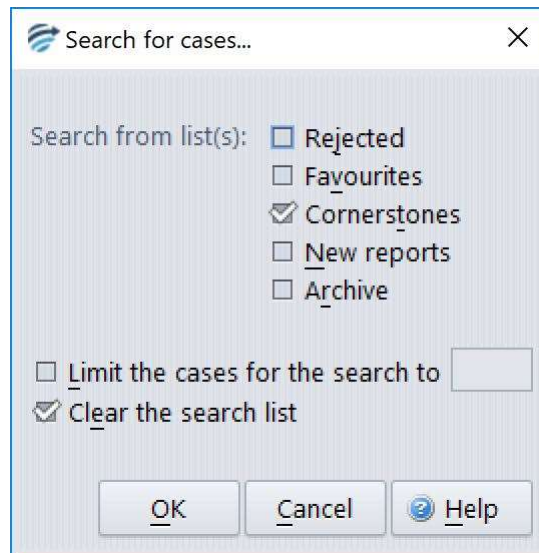
Select the case list for the search and enter a limit, if required. Click OK to commence the search.

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:
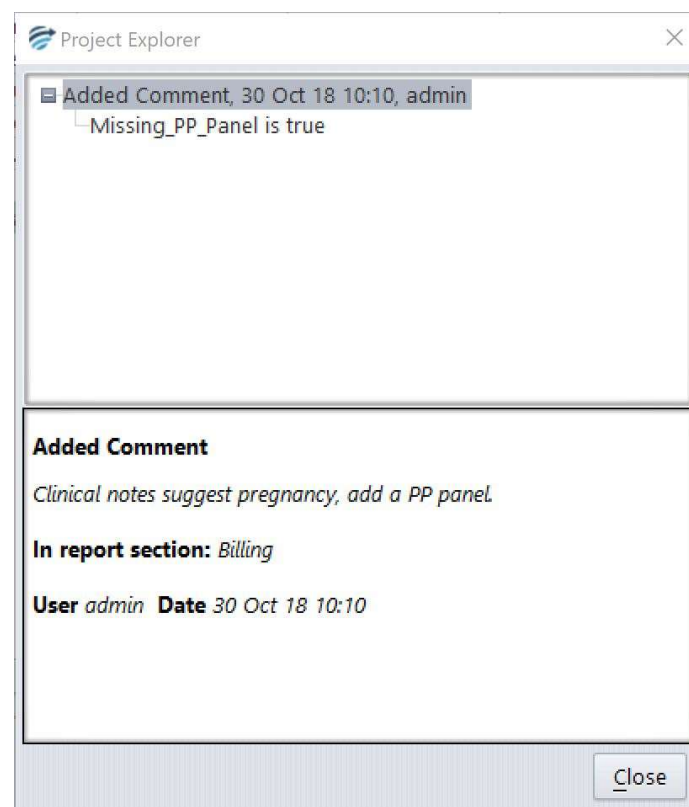
Simulation reports

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 desired case list and go to Tools | Simulation Report. A window with following statistics will appear:

- A summary statement showing the number of cases containing an interpretationfrom the total as well as a percentage
- 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:

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

Housekeeping

Housekeeping of the system is performed automatically once per day and typically takes roughly 15 minutes 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.
- The log file will be sent for evaluation to a specified monitoring address.
- The licence file will be validated
- All statistics will be updated

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 | Help Topics

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.
	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.

Category	Expression	Description
	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.
	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.

Category	Expression	Description
	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'
	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.

Category	Expression	Description
	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
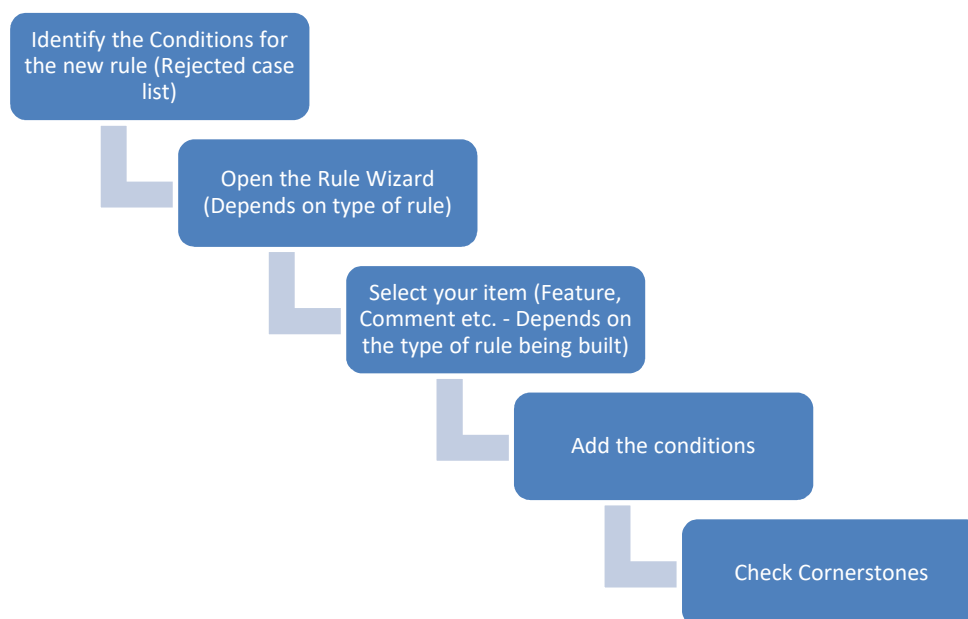
Steps to building a rule in RippleDown

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:
     o Write the condition in the current window which is located in the middle of the left hand side of the screen.
     o Click on Add.
     o The Wizard will then add the condition to the Completed Window.
   - If assistance is required with syntax for the condition:
     o Select the tab which contains the attribute, feature etc. that is to be used for the condition.
     o Select the attribute, feature etc. for the condition.
     o Click on Next.
     o Select an expression from one of the 6 available tabs. Complete the expression if needed. Click Next.
     o 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
P8-3096	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 sequence is followed: • An existing comment within a particular report section is modified by the knowledge builder • The report section that contains the edited comment is then modified directly afterwards • No other screens or tabs are selected in between the above actions • The Report in the Preview pane will not reflect the newly selected report section format. **Evaluation:** The format will only be displayed correctly once the daily housekeeping is performed. Exact sequence of events must be followed for issue to occur. If any other screen or option is selected between the edit of the comment and the edit of the report section, the issue will not occur. 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.
P8-2699	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.
P8-2921	Client application freezes when resetting password after an inactivity timeout - V8	**Description:** Bring up window to reset your password, but don't reset it Wait until the client timeout inactivity period has elapsed The client application is now frozen and the application will need to be restarted. **Evaluation:** A new client can still be started if the issue occurs. No impact on patient reports. Client can be stopped

		through task manager. Hung client will be disconnected at housekeeping in the client is not stopped manually.
P8-3091	Incorrect window rendering of Numeric Attribute window when using some Asian characters	**Description:** When Korean, Japanese or Chinese characters have been used for a primary attribute, if the attribute name is very long, then when a Numeric derived attribute is created based off this long named primary attribute, the Numeric attribute window does not render correctly. **Evaluation:** Workaround is to rename the primary attribute to a shorter length and set values for the numeric attribute. The name of the primary attribute can then be set back to the original. Patient reports are not affected.
P8-3106	Searching by condition in Config Archive	**Description:** When "Search by condition" is used in the Archive case list of the Config knowledge base, the search does not return any results even if there are valid entries in the case list. **Evaluation:** Workaround is to copy cases to the favourites case list which can be searched. The Config knowledge base is commonly only used for creating derived attributes to then flow to other knowledge bases as primary attributes. Config knowledge base are rarely used for rule building. Once a case is copied to the favourites, it can then be manually modified to fulfil any required rule building session. This issue has no impact on patient reports.