Zermatt User Manual

Whilst all reasonable care has been taken to ensure that the details are true and not misleading at the time of publication, no liability whatsoever is assumed by Automature LLC, or any supplier of Automature LLC, with respect to the accuracy or any use of the information provided herein.

Any license, delivery and support of software require entering into separate agreements with Automature LLC.

This document may contain confidential information and may not be modified or reproduced, in whole or in part, or transmitted in any form to any third party, without the written approval from Automature LLC.

Revision History

Ver.	Date	Name	Description of Changes
1.0	09/22/2010	Anuradha Chanda	First Edition
1.1	10/27/2010	Dipayan Sengupta	Revised Edition
1.2	11/06/2010	Dipu Deshmukh	Review
1.3	12/07/2010	Anuradha Chanda	Revised Edition
2.1	18/01/2011	Dipayan Sengupta	New Display layouts

Contents

1 Introduction

ZERMATT is Automature's browser based quality management framework for Planning and Reporting.

ZERMATT is used to store the results of the tests, the test environments and everything else that is contextually relevant to the tests.

It also provides an infrastructure for planning, creating requirements, and grouping test cases.

1.1 Document Purpose

The user manual will help you navigate Zermatt's Work Flows and create reports.

1.2 Intended Audience

This manual is intended for the users who would be using Zermatt for planning and reporting purpose.

1.3 References and Other Related Documents

The following documents provide additional useful information about Automature's other products, and how they relate to Zermatt.

1. Zermatt Install Guide
2. Zug User Manual

2 Concepts and Terminology

2.1 Product

- A company develops and sells several individually releasable software packages. These are the entities, that we call ***Products***. Example of a product could be Adobe's Flash, or Automature's ZERMATT. Products may often be referred to as projects in some contexts.

2.2 Role

- We normally use more than one machine in conducting a specific test. Each machine participates in a specific ***role*** that is assigned to it, for the purpose of a test.

2.3 Release

- A product is developed for customer consumption in increments, known as ***releases***. Each release can have a code name (internal to the organization), and a version number (by which it is identified outside of the organization, where it is developed). Code Names are normally used during development of a specific release.
- A release maybe created for various reasons, viz.
  - Development of new features.
  - Emergency bug fixes, also known as patch or hot fix releases.
  - Rollup bug fixes also known as Service Packs.

2.4 Sprint or Project Phase

- Each development release is organized in a series of sprints or phases, each of which provides incremental levels of functionality. The goal at the end of each sprint is to achieve a reasonably stable version of the product.
- The final sprint is known as the ***Release Sprint***.
- For the purposes of quality management, product sustaining activity, that lead to service packs or patch releases are also considered project phases.

2.5 Test Case

- A ***Test Case*** describes a specific test procedure, in concert with all the variables, identified as being significant for testing the feature. Note that an environmental variable, such as the operating system version, while also significant for the test case, is normally tracked using topologies.

2.6 Test Suite

- A ***Test Suite*** consists of a collection of Test Cases, which somehow are related together for reasons, considered appropriate by the Test Plan designer.
- Usually, Test Cases, inside a specific Test Plan may use similar test procedures, or may test a specific product feature.

2.7 Test Cycle

- A ***Test Cycle*** describes a logical test session, during which tests are conducted. Typically (but not necessarily), some environmental conditions remain constant, such as the build, the test engineer, etc... While certain other environmental conditions may vary, such as the Topology Sets where the tests are being executed.

- All results of Test execution are stored within the context of a test cycle. There can be no more than a single result stored for a specific test case, executed on a specific topology set, inside one test cycle.
- A test cycle usually, but not necessarily refers to a specific build of the software being tested

2.8 Test Suite List

- A collection of Test Suites that might be used for a specific purpose, such as a test cycle, a release regression test plan is called ***Test Suite List***.

2.9 Topology

- A ***topology*** describes a single machine participating in the execution of a Test Case. The machine is described in terms of various significant attributes, such as its role, its operating system, the software installed on it, etc.

2.10 Topology Set

- A ***Topology Se***t describes a collection of machines required to execute a Test Case. Each machine may participate in a different role (e.g. client, server, database, Active Directory, proxy server, etc...). A test case may typically be executed on more than one topology set.

2.11 Topology Set List

- A collection of Topology Sets that might be used for a specific purpose, such as a test cycle, or a release regression test plan is called Topology Set List.

2.12 Test Plan

- A ***Test Plan*** identifies the following entities, viz.
    - The release and the project phase for which the plan exists.
    - The complete list of test suites (and specific test cases, contained in them) that should be executed during the course of the sprint.
    - The complete list of topology sets, where the test cases must be executed during the sprint.
- Multiple Test Cycles (i.e. one or more) may be executed during the course of a single sprint, and contribute to the Test Plan. Some of these tests may be manually executed, while others may be automated.

2.13 Release Test Plan

- A ***Release Test Plan*** or RTP is a special case of a Test Plan. Specifically it is the Test Plan for the final, or Release Sprint. It identifies all the same entities as a normal Test Plan, but is furthermore, considered a formal release document.
- A RTP is normally a superset of all individual Test Plans for that release.

2.14 Feature

- Each product is assumed to contain a number of high level features.

2.15 Detail Requirements

- Each product feature can be further decomposed into a number of individual detail requirements. These requirements may be spread over releases. Such requirements are considered testable entities, and can be associated with one or more test cases. Requirements may also be assigned a priority (LOW, MEDIUM or HIGH).

Assigning a priority to a requirement is often useful for planning and scheduling which test cases need to be developed early in the release cycle, and which ones may be deferred until later.

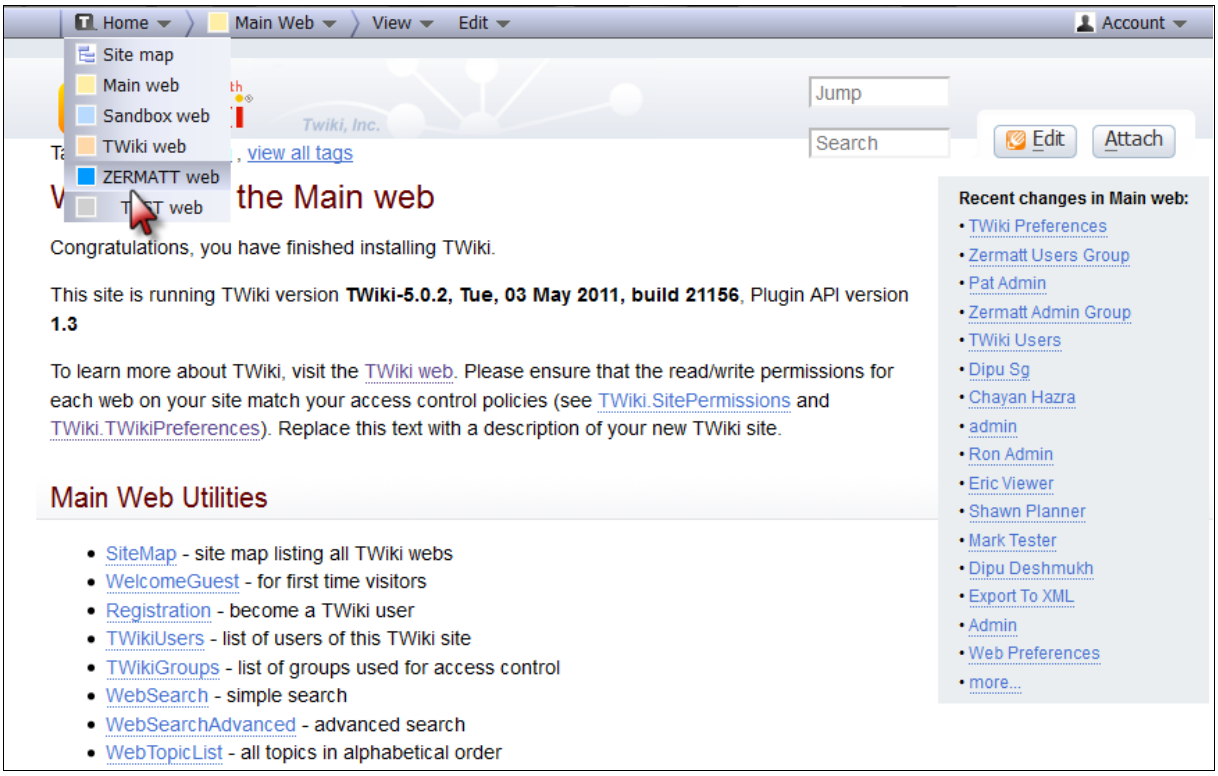
2.16 Requirement Variants

- A specific detail requirement may apply in more than one way to the target. For example, the requirement to copy a file may be dependent on various factors, such as its size, its structure, its permissions, its visibility, etc... These are the variables for the requirement. Each variable can take on a number of variants. For example, the variants for file size could be zero-byte, small, medium, large or huge. Thus a single requirement might be testable under a large number of variant combinations. A test case usually tests one such combination. Coverage is determined not only by the number of requirements tested, but also by the number of variants of those requirements tested.

3 Zermatt Navigation

3.1 Getting Started

Once you have registered in Zermatt or logged in giving the username and password credentials, you will be redirected to the Twiki Main web.



W the Main web

Congratulations, you have finished installing TWiki.

This site is running TWiki version **TWiki-5.0.2, Tue, 03 May 2011, build 21156**, Plugin API version **1.3**

To learn more about TWiki, visit the TWiki web. Please ensure that the read/write permissions for each web on your site match your access control policies (see TWiki.SitePermissions and TWiki.TWikiPreferences). Replace this text with a description of your new TWiki site.

Main Web Utilities

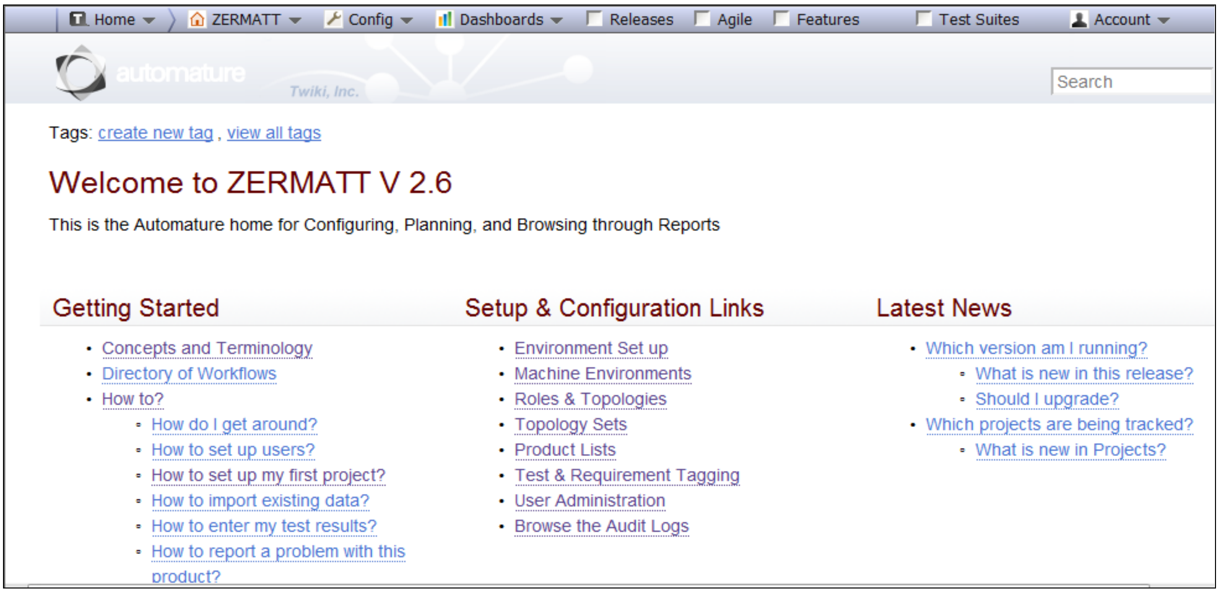
- SiteMap - site map listing all TWiki webs
- WelcomeGuest - for first time visitors
- Registration - become a TWiki user
- TWikiUsers - list of users of this TWiki site
- TWikiGroups - list of groups used for access control
- WebSearch - simple search
- WebSearchAdvanced - advanced search
- WebTopicList - all topics in alphabetical order

Recent changes in Main web:

- TWiki Preferences
- Zermatt Users Group
- Pat Admin
- Zermatt Admin Group
- TWiki Users
- Dipu Sg
- Chayan Hazra
- admin
- Ron Admin
- Eric Viewer
- Shawn Planner
- Mark Tester
- Dipu Deshmukh
- Export To XML
- Admin
- Web Preferences
- more...

Select Zermatt Web from the drop down menu of Home and you will be navigated to the Zermatt home page.

Zermatt Homepage



Tags: create new tag , view all tags

Welcome to ZERMATT V 2.6

This is the Automature home for Configuring, Planning, and Browsing through Reports

Getting Started

- Concepts and Terminology
- Directory of Workflows
- How to?
  - How do I get around?
  - How to set up users?
  - How to set up my first project?
  - How to import existing data?
  - How to enter my test results?
  - How to report a problem with this product?

Setup & Configuration Links

- Environment Set up
- Machine Environments
- Roles & Topologies
- Topology Sets
- Product Lists
- Test & Requirement Tagging
- User Administration
- Browse the Audit Logs

Latest News

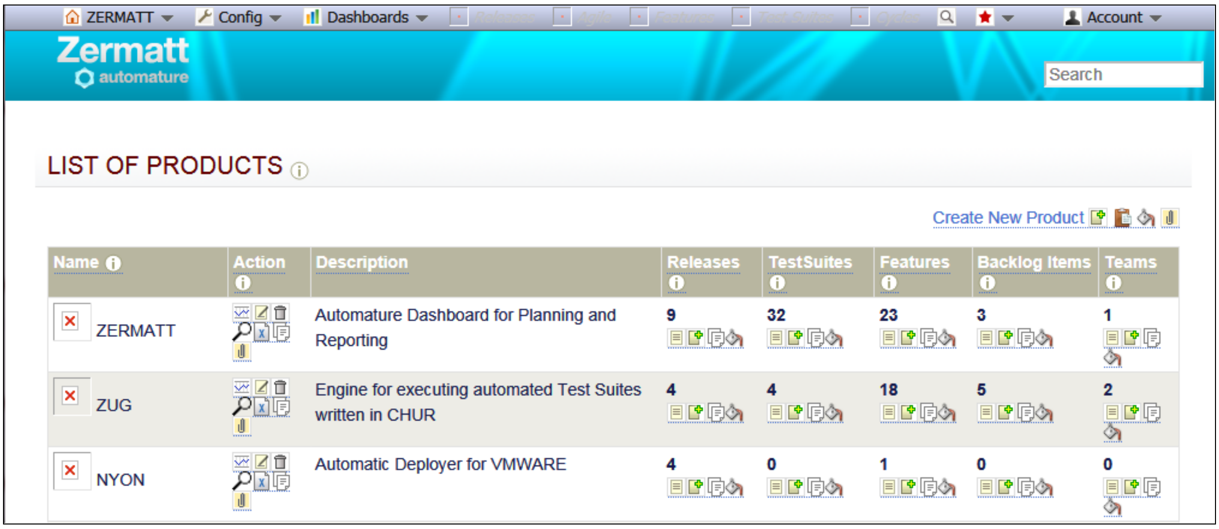
- Which version am I running?
  - What is new in this release?
  - Should I upgrade?
- Which projects are being tracked?
  - What is new in Projects?

This is the Zermatt web home page. It contains all necessary links to tutorials and other pages of Zermatt which are required for Configuring, Planning and Browsing through different reports. If you are a first time user we strongly recommend you to go through the topics under **Getting Started.** If you want to navigate to Releases, Agile or Features or Test Suites, you need to select a product first.

If you want to create a new product, then go the Products page from Zermatt drop down Menu.

3.2 Screen Layout

The screen has been divided into several sections, viz. top menu bar and the search bar containing the Automature logo, and the main display area, that covers the rest of the browser window.



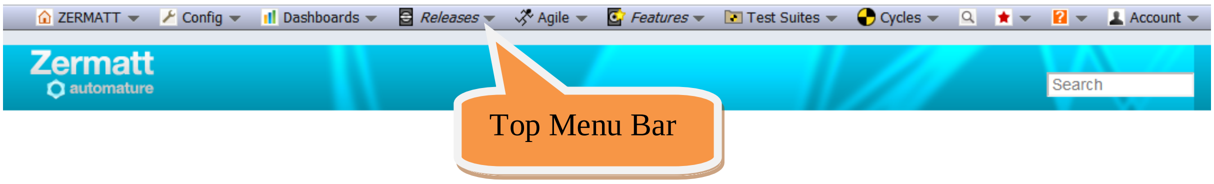
LIST OF PRODUCTS

Create New Product

Name	Action	Description	Releases	TestSuites	Features	Backlog Items	Teams
ZERMATT		Automature Dashboard for Planning and Reporting	9	32	23	3	1
ZUG		Engine for executing automated Test Suites written in CHUR	4	4	18	5	2
NYON		Automatic Deployer for VMWARE	4	0	1	0	0

3.3 The Navigation Bar on the top

The Navigation Bar can be used to quickly move between workflows. There are ten pull down topics in the menu.

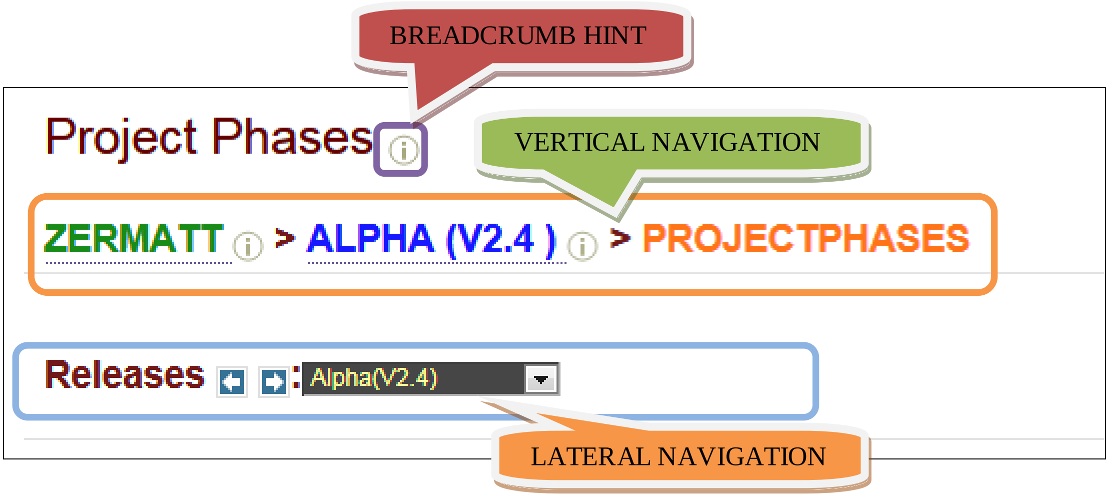


3.4 Breadcrumbs

Breadcrumbs can be found on most pages that are at leaf nodes of a workflow. The purpose of the breadcrumbs is two-fold, viz.

- to identify the objects selected during the navigation of the current workflow, to arrive at the current page
- to allow the user to backtrack easily to a previous step along the current workflow, by clicking on the link

Note that breadcrumbs allow only backward navigation on the currently selected workflow. To jump to the last visited page on a different workflow, one has to select from the Navigation Bar on the left Frame.



3.5 Object Creation Hot Links

Object creation links are usually found on the top right corner, just above the table. These links allow the user to create a new object, either from scratch, or by pasting from the clipboard. The object created depends on the context of the current page. Note, that the Zermatt clipboard has nothing to do with Windows clipboard.

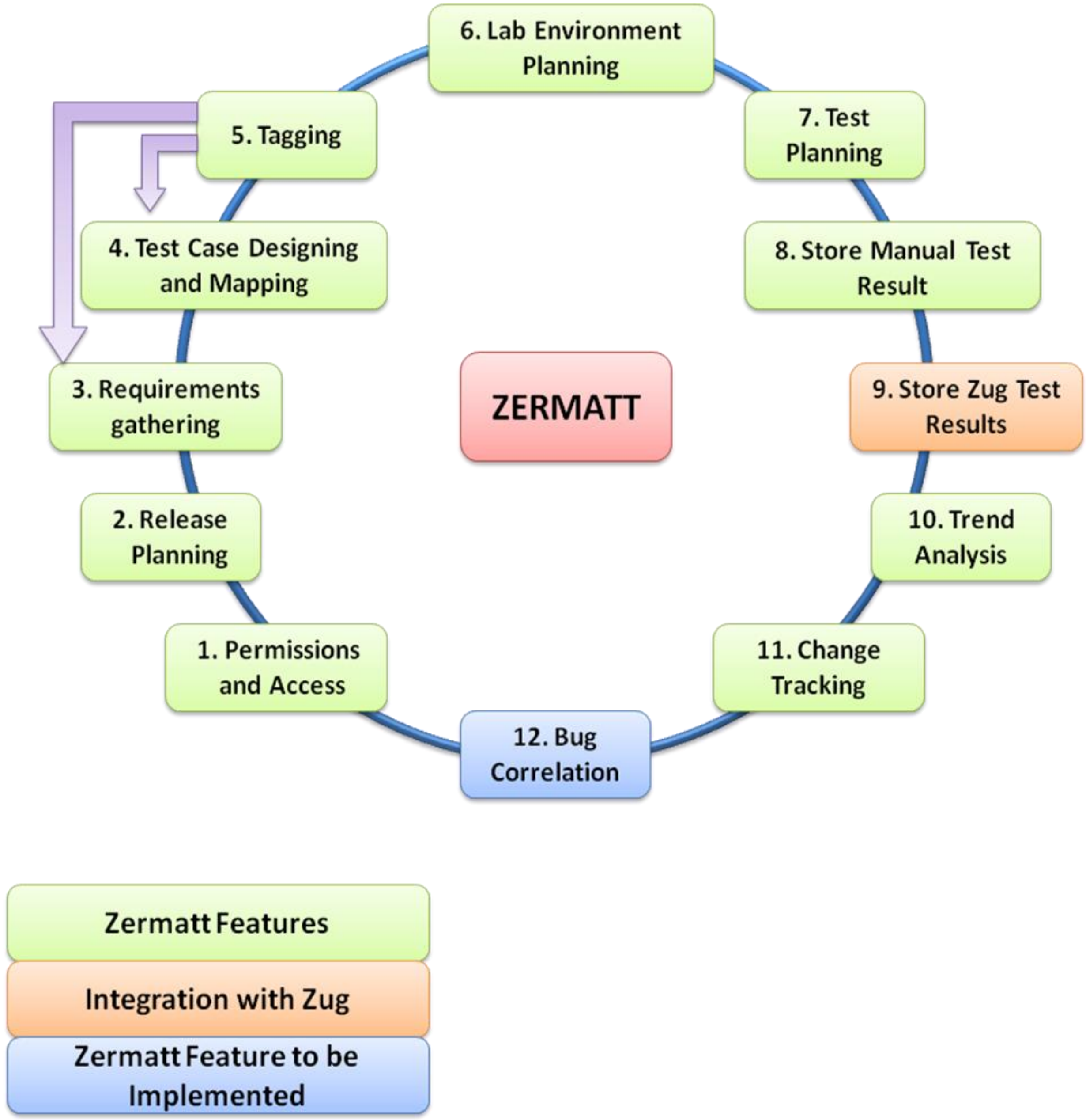
3.6 Icons

In order to provide a consistent user experience, Automature has standardized on the use of a few icons, viz.

Icon	Significance
🎞	Help Video
ⓘ	Help text
📈	Navigate to context dependent dashboard page
✎	Modify the properties of the chosen object
➕	Create a new object of the chosen type
🗑	Delete the chosen object
🖨	Generate a printable view of this object
📄	Generate a XML dump of this object, useful for loading into a spreadsheet, or migrating data into another Zermatt installation
⧉	Make a copy of this object in the clipboard
📋	Paste a copy of the object from the clipboard
⛉	Paste a copy of the clipboard object and all dependent objects under it
⬅ ➡	Move laterally, choose another object of same type as currently chosen
⬇	Drill down one level deeper than object currently chosen
🔍	Detailed View
X	Export to XML
📎	Attach a File
🏷	Selecting by tag value
🔗	For Association and Disassociation
💬	Comments
W	View the attached document
⊘	Cancel Edit mode while assigning a story to a team
◔	Performance Details
⊜	A Release
◉	A Sprint
☰	A Test Plan
◑	A Test Cycle
📁	A Test Suite
▫	A Test Case
★	A Feature
▦	A Requirement

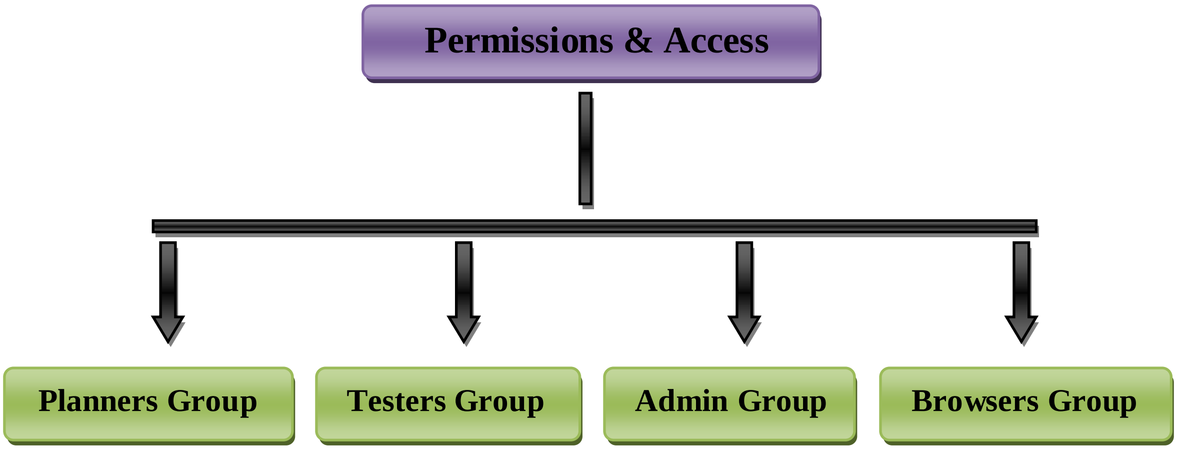
🏃	A Epic
👥	A Team
👤	A User

4 Process Overview


ZERMATT
1. Permissions and Access
2. Release Planning
3. Requirements gathering
4. Test Case Designing and Mapping
5. Tagging
6. Lab Environment Planning
7. Test Planning
8. Store Manual Test Result
9. Store Zug Test Results
10. Trend Analysis
11. Change Tracking
12. Bug Correlation
Zermatt Features
Integration with Zug
Zermatt Feature to be Implemented

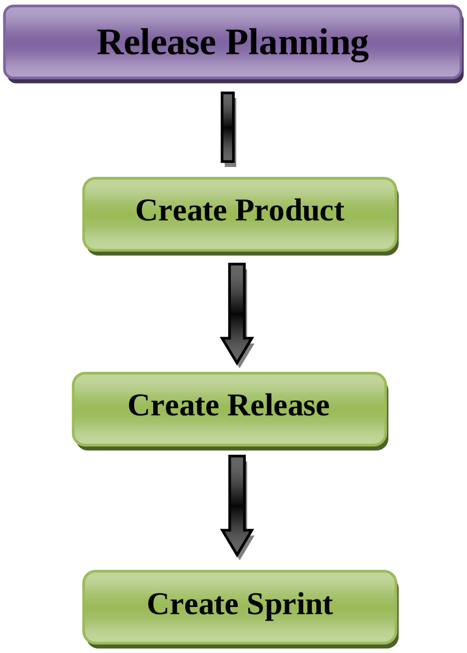
4.1 Step 1: Roles and Permissions (Refer to topic 5.2.1)

- Create users in the Planners group
- Create users in the Testers group
- Create users in the Admin group
- Create users in the Browsers group



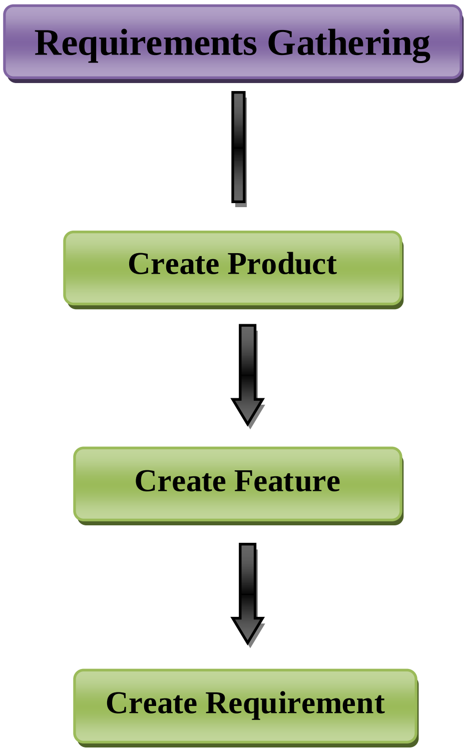
4.2 Step 2: Release Planning (Refer to topic 5.3.1)

- **Create a Product.**
- **Create a Release for the Product.**
- **Create a Sprint for the Release.**



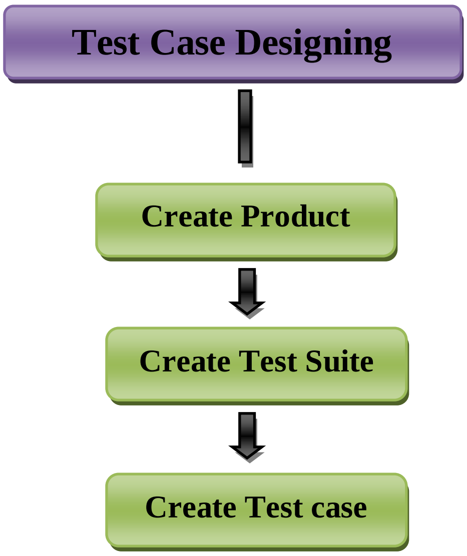
4.3 Step 3: Requirement Planning (Refer to topic 5.3.2)

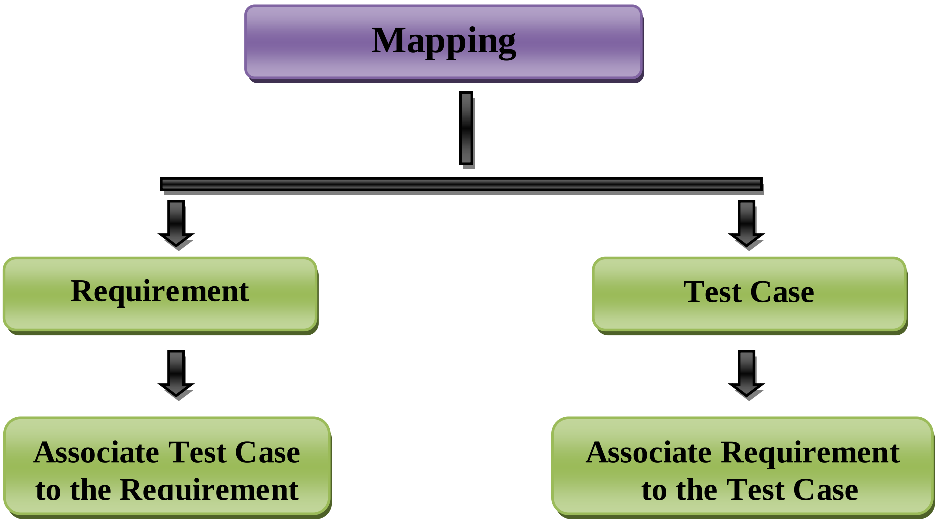
- Capture a product feature in Zermatt.
- Create a Requirement for the feature.
- Requirement can only be created if there is a Release.
- A test case can be written for the requirement.



4.4 Step 4: Test Case Designing & Mapping (Refer to topic 5.3.3, 5.3.4, 5.3.5)

- **A test case can only exist if there is a test suite.**
- **A test suite must belong to a Role. Create the Role, if it does not already exist. Create a test suite in a Product.**
- **Under the test suite write a test case.**
- **Now associate the test case to a requirement.**

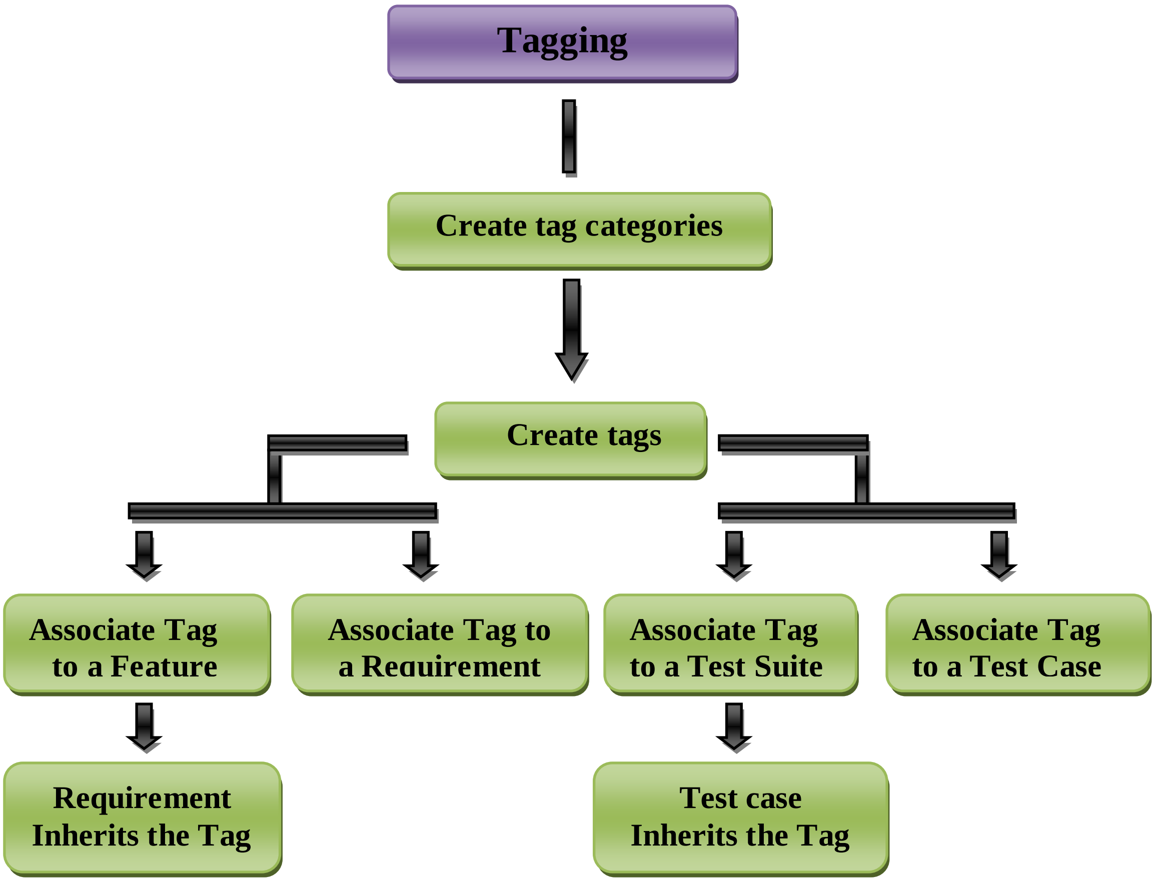



Mapping
Requirement
Test Case
Associate Test Case to the Reauirement
Associate Requirement to the Test Case

4.5 Step 5: Tagging (Refer to topic 5.3.7)

Test cases and requirements may be grouped in many different ways to serve many different purposes, and facilitate quick selection for a particular task.

- Create Tag Categories.
- Create Tags under the category.
- Associate test cases and requirements to the tags.

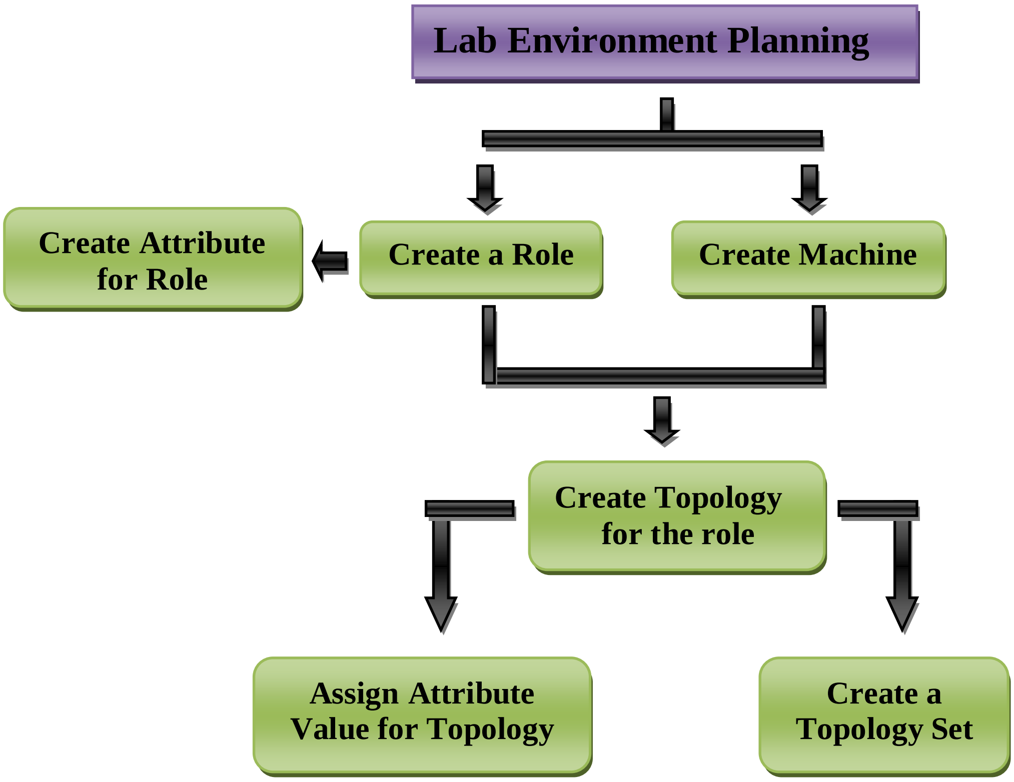

Tagging
Create tag categories
Create tags
Associate Tag to a Feature
Associate Tag to a Requirement
Associate Tag to a Test Suite
Associate Tag to a Test Case
Requirement Inherits the Tag
Test case Inherits the Tag

4.6 Step 6: Lab Environment Planning (Refer to topic 5.3.6)

You can catalog all the equipment used for testing. These maybe either be physical machines or virtual machines and later these machines may be referenced as part of a topology for a test session.

- Add a new Machine.

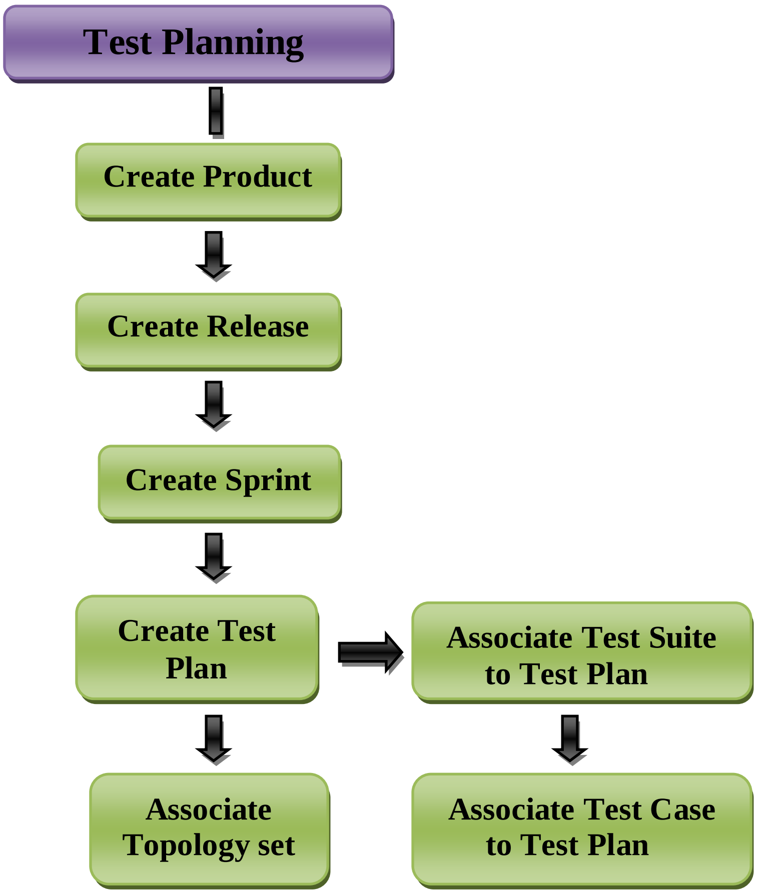
- Add a new Role.
- Add an Attributes for the Role
- Add a new Topology for the Role.
- Add an Attribute Value for the Topology.
- Add a new Topology Set.
- Associate a topology to the Topology Set.


Lab Environment Planning
Create Attribute for Role
Create a Role
Create Machine
Create Topology for the role
Assign Attribute Value for Topology
Create a Topology Set

4.7 Step 7: Test Planning (Refer to topic 5.3.8)

- **Now to test the requirement you need to have a test plan.**
- **Create a test plan under the project phase / Sprint.**
- **A test plan will have test suites and Topology sets.**
- **Associate one or more topology sets to the test plan.**

- **Associate the test suites that have the test cases for the requirements to be tested.**
- **Choose the test cases within the test suites associated with the test plan**

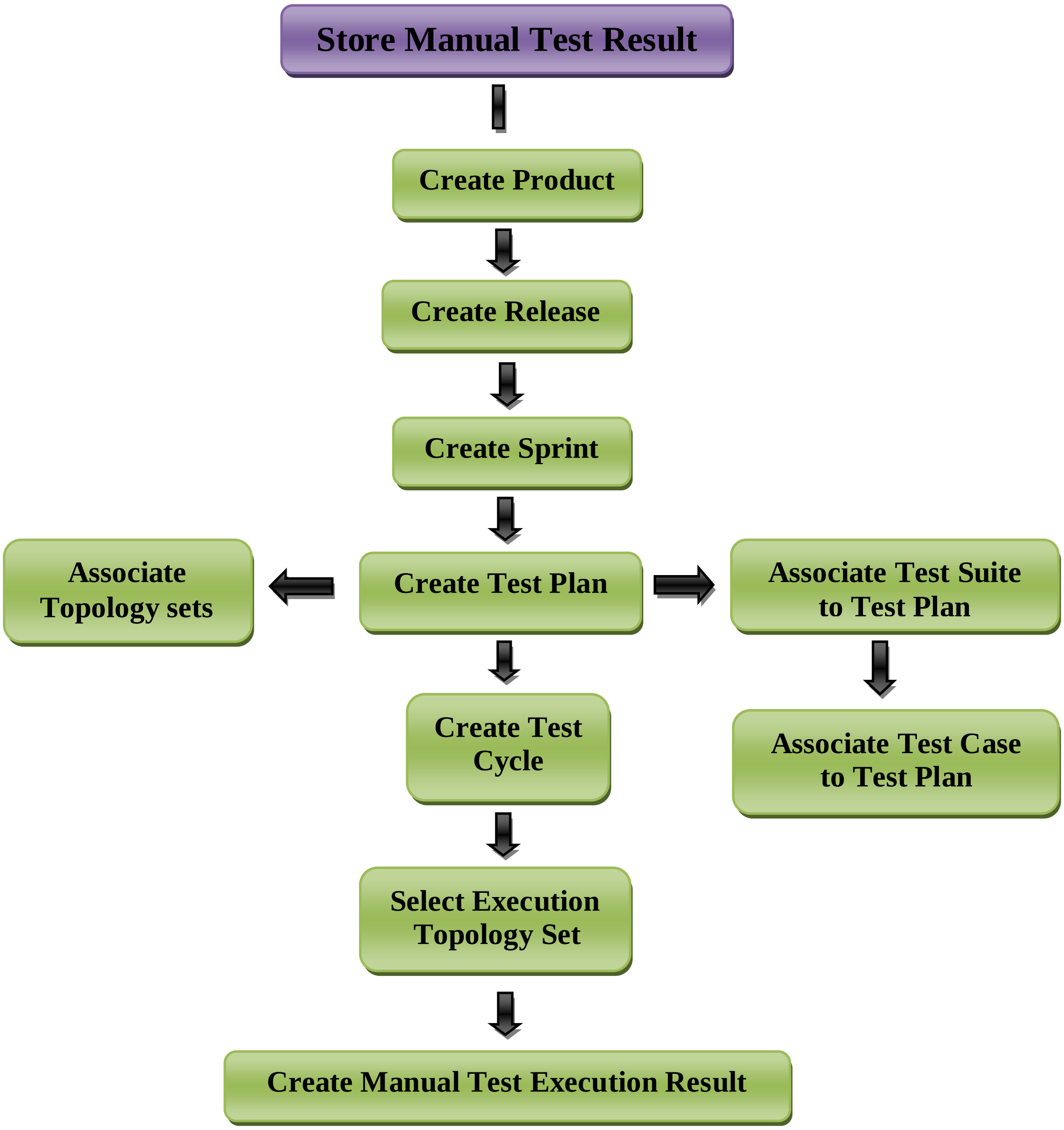


4.8 Step: 8 Store Manual test Result (Refer to topic 5.3.11)

You can store the test results in Zermatt by following the below Step:

- **We assume you have already created a Test Plan, as described in the previous section.**
- **Now create a test cycle under the test plan.**

- **In the execution topology set of the test cycle, create a manual test execution result.**


Store Manual Test Result
Create Product
Create Release
Create Sprint
Associate Topology sets
Create Test Plan
Associate Test Suite to Test Plan
Create Test Cycle
Associate Test Case to Test Plan
Select Execution Topology Set
Create Manual Test Execution Result

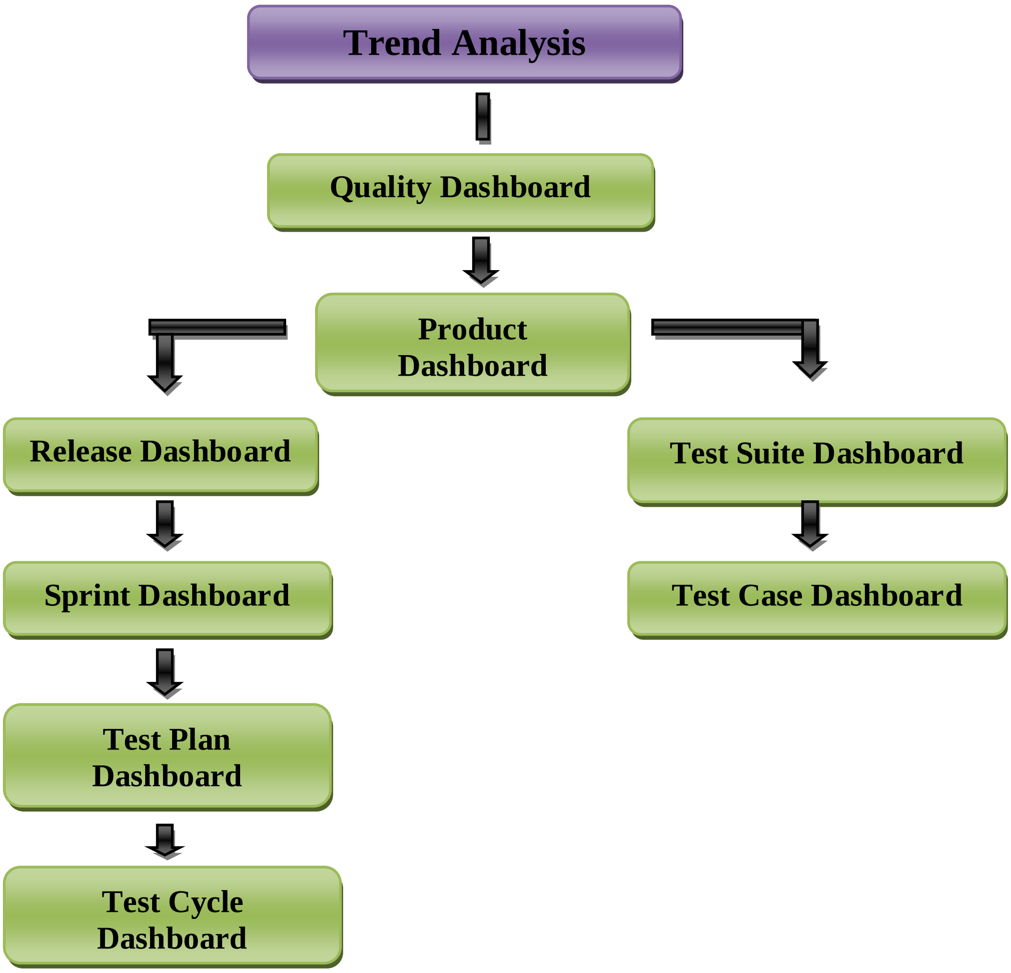
4.9 Step 9: Store Automation Test Results

The results of the tests run by Zug are automatically stored in Zermatt.

(This topic is covered in the Zug User Manual)

4.10 Step 10: Trend Analysis (Refer to topic 5.4)

- Quality Dashboard shows the overview of all the products.
- Product Quality Dashboard shows the overview of individual products.
- Release Quality Dashboard shows the overview of all the releases under a Product.
- Sprint Quality Dashboard shows the overview of all the sprints under a Release.
- Test Plan Quality Dashboard shows the overview of all the Test Plans under a Sprint.
- Test Cycle Quality Dashboard shows the overview of all the Test Cycles under a Test Plan.
- Test Suite Quality Dashboard shows the overview of all the Test Suites under a Product.
- Test Case Quality Dashboard shows the overview of all the Test Cases under a Test Suite.

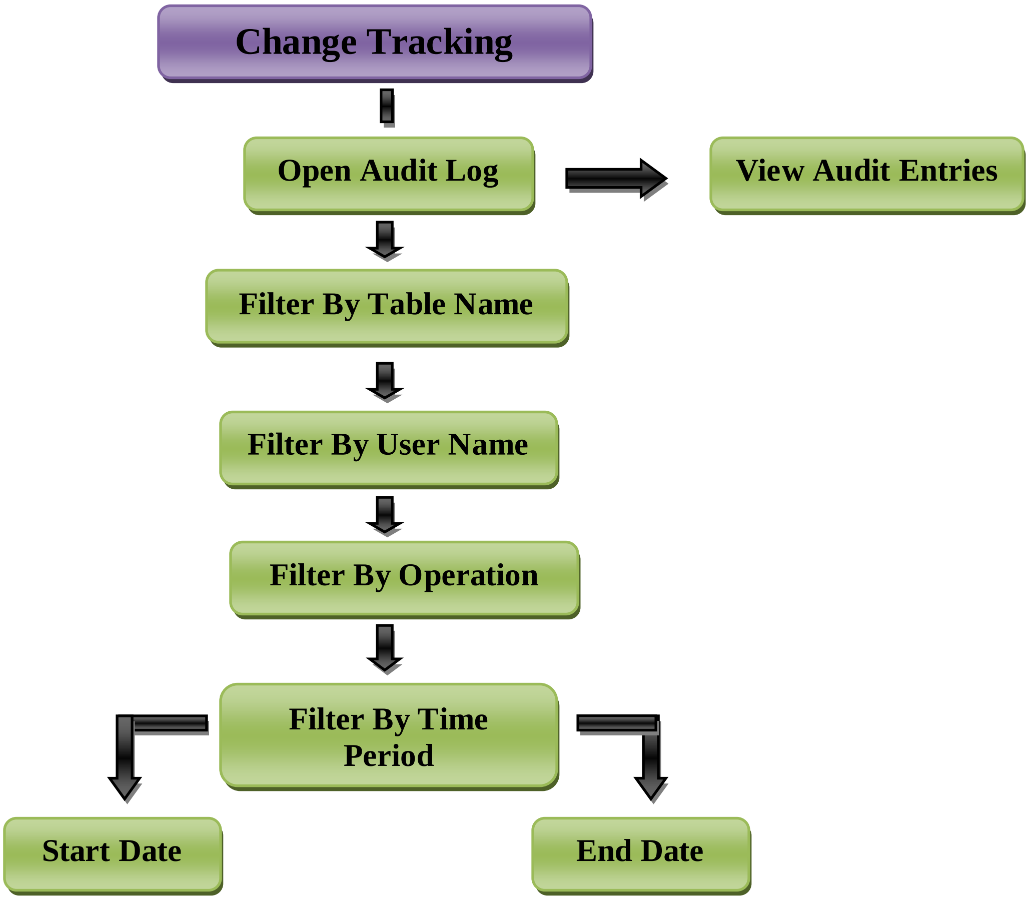


4.11 Step 11: Change Tracking (Refer to topic 5.2.2)

In Zermatt change tracking is done by a feature called Audit Log.

Audit Entries are created whenever any planning data is created, deleted or modified.

- Open the Audit Log Page
- View Audit Entries
- Filter Audit Entries by Table Name
- Filter Audit Entries by User Name
- Filter Audit Entries by Operation Type
- Filter Audit Entries by Time Period (Start Date & End Date)



4.12 Step 12: Bug Correlation

Bug Correlation in Zermatt is an ability to correlate defects encountered during test execution, and defects that are being tracked in a bug tracking system such as Bugzilla or Redmine. Note that this feature is not available in V1 of Zermatt.

5 Work Flows

5.1 Configuration Tasks

5.1.1 Archive Location Setup

This workflow allows you to specify the location on disk, where ZUG log files, as well as any other application specific log files will be compressed and archived. The name of the key is ***ArchiveLocation.*** Note that the location must be a network folder that is accessible from all machines where ZUG might be running.

Check where Logs are being archived

Create a new archival location

Modify current archival location

5.1.2 Lab Environment Planning

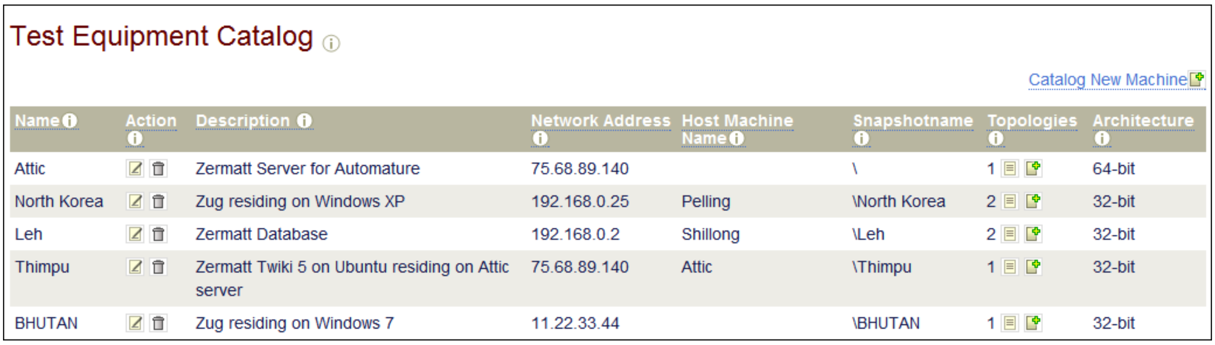
This workflow allows you to catalog all the equipment used for testing. These may be either physical machines, or virtual machines. Later these machines may be referenced as part of a topology for a test session.

List All Lab Physical or Virtual Machines

Add a New Machine to the catalog

Remove a Machine from the existing catalog

Modify the description of a Machine



Test Equipment Catalog

Catalog New Machine

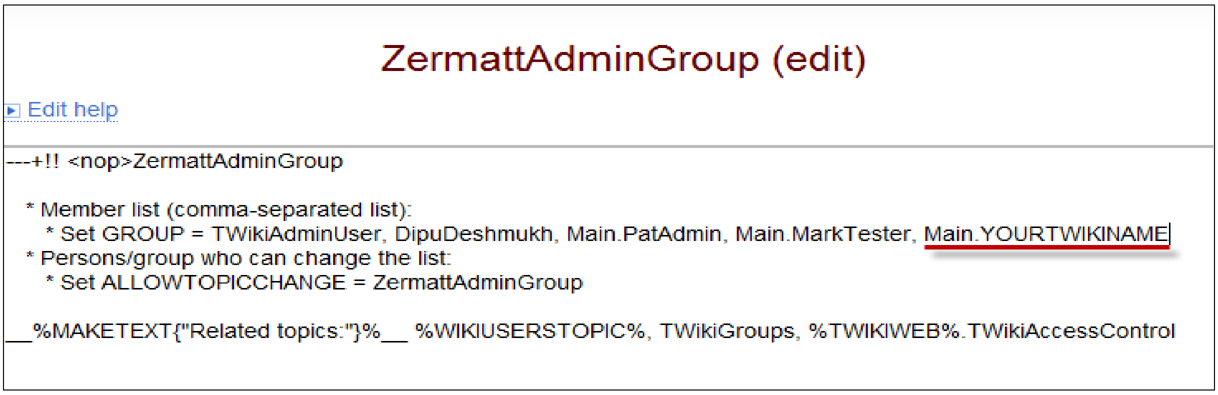
Name	Action	Description	Network Address	Host Machine Name	Snapshotname	Topologies	Architecture
Attic		Zermatt Server for Automature	75.68.89.140		\	1	64-bit
North Korea		Zug residing on Windows XP	192.168.0.25	Pelling	\North Korea	2	32-bit
Leh		Zermatt Database	192.168.0.2	Shillong	\Leh	2	32-bit
Thimpu		Zermatt Twiki 5 on Ubuntu residing on Attic server	75.68.89.140	Attic	\Thimpu	1	32-bit
BHUTAN		Zug residing on Windows 7	11.22.33.44		\BHUTAN	1	32-bit

5.2 Administrative Tasks

5.2.1 Manage User Permissions

Zermatt 2 different authorization levels to distinguish between different classes of users, viz. Administrators and Users. Only the Administrator can make changes in this page.
You can make someone TWiki Administrator or Super user by visiting the page http://localhost/twiki/bin/view/Main/TWikiGroups.

If you want to give administrative privileges to a particular user , click on ZermattAdminGroup and edit the page in the following way :
Set GROUP = Main.YourTWikiName and save this page. This makes the user the administrator for the Zermatt product.



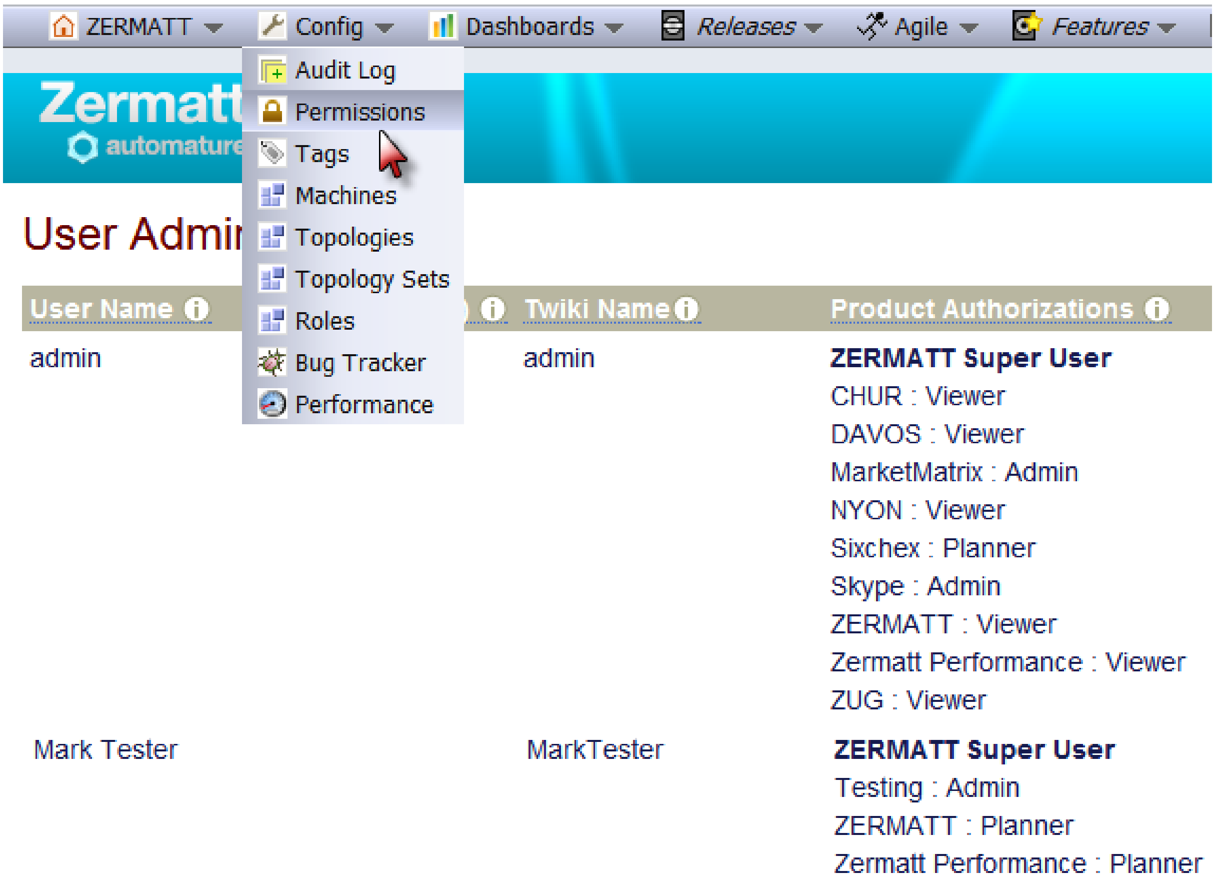
ZermattAdminGroup (edit)

```
---+!! <nop>ZermattAdminGroup

 * Member list (comma-separated list):
   * Set GROUP = TWikiAdminUser, DipuDeshmukh, Main.PatAdmin, Main.MarkTester, Main.YOURTWIKINAME
 * Persons/group who can change the list:
   * Set ALLOWTOPICCHANGE = ZermattAdminGroup

__%MAKETEXT{"Related topics:"}%__ %WIKIUSERSTOPIC%, TWikiGroups, %TWIKIWEB%.TWikiAccessControl
```

Now the administrator can add new users in Zermatt who have registered in TWiki. These users can be given different levels of permissions for the existing products in Zermatt. To give User Permissions for Products navigate in the following way and you will be redirected to the User Admin page.

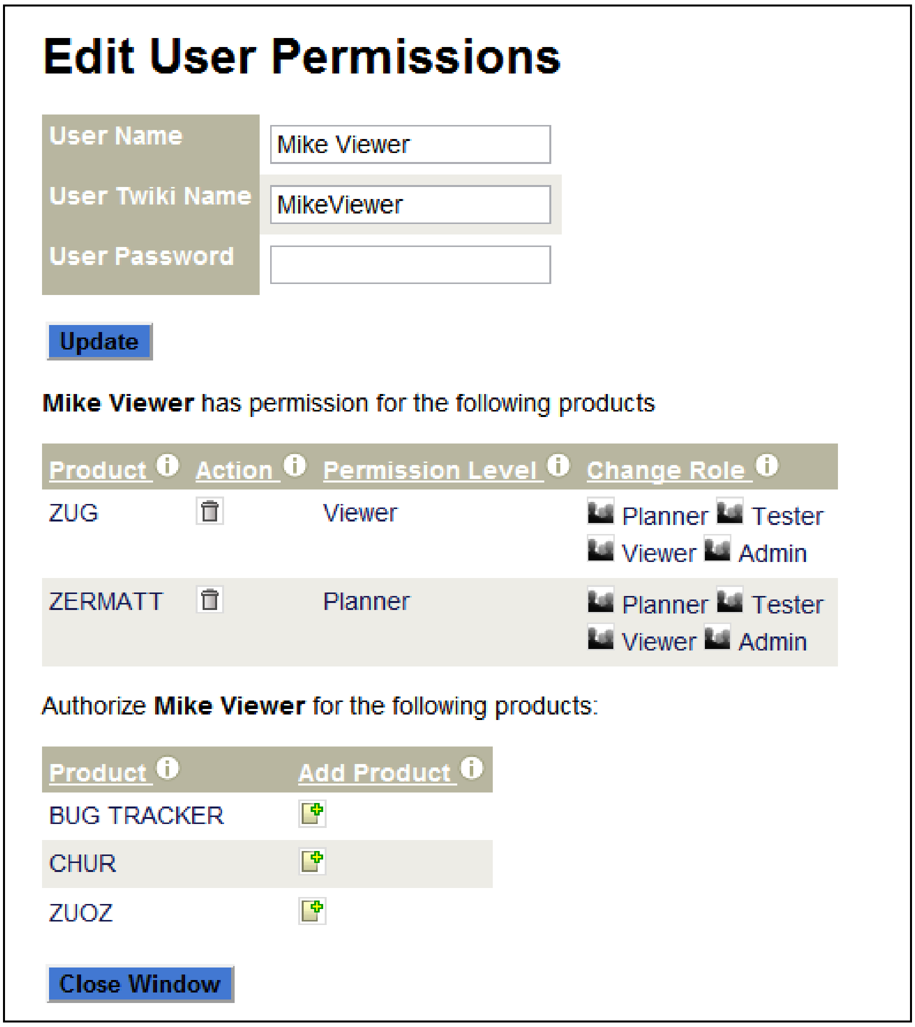


User Admin

User Name	Twiki Name	Product Authorizations
admin	admin	ZERMATT Super User CHUR : Viewer DAVOS : Viewer MarketMatrix : Admin NYON : Viewer Sixchex : Planner Skype : Admin ZERMATT : Viewer Zermatt Performance : Viewer ZUG : Viewer
Mark Tester	MarkTester	ZERMATT Super User Testing : Admin ZERMATT : Planner Zermatt Performance : Planner

Each category of user is allowed to do certain activities. Some authorization levels are subsets of others. For instance, Planners are allowed to do everything Testers and Browsers are allowed to do. **The authorization to different products can only be given by the Zermatt Admin Group.**

Authorization Level	Rights
Planner	Allowed to enter or modify planning data in Zermatt. Planners may also enter or modify test execution related data. A planner can cannot add or delete any product.
Tester	Allowed to enter or modify test execution related data, but not the planning data. Examples of test execution data are Test Cases, Test Cycles, Test Execution Results, etc..
Admin	The Admin has all the privileges to a particular product including deleting the product to which he is assigned as administrator.
Viewer	Allowed to view planning data and execution reports



Edit User Permissions

User Name	Mike Viewer
User Twiki Name	MikeViewer
User Password	

Mike Viewer has permission for the following products

Product	Action	Permission Level	Change Role
ZUG	🗑	Viewer	Planner Tester Viewer Admin
ZERMATT	🗑	Planner	Planner Tester Viewer Admin

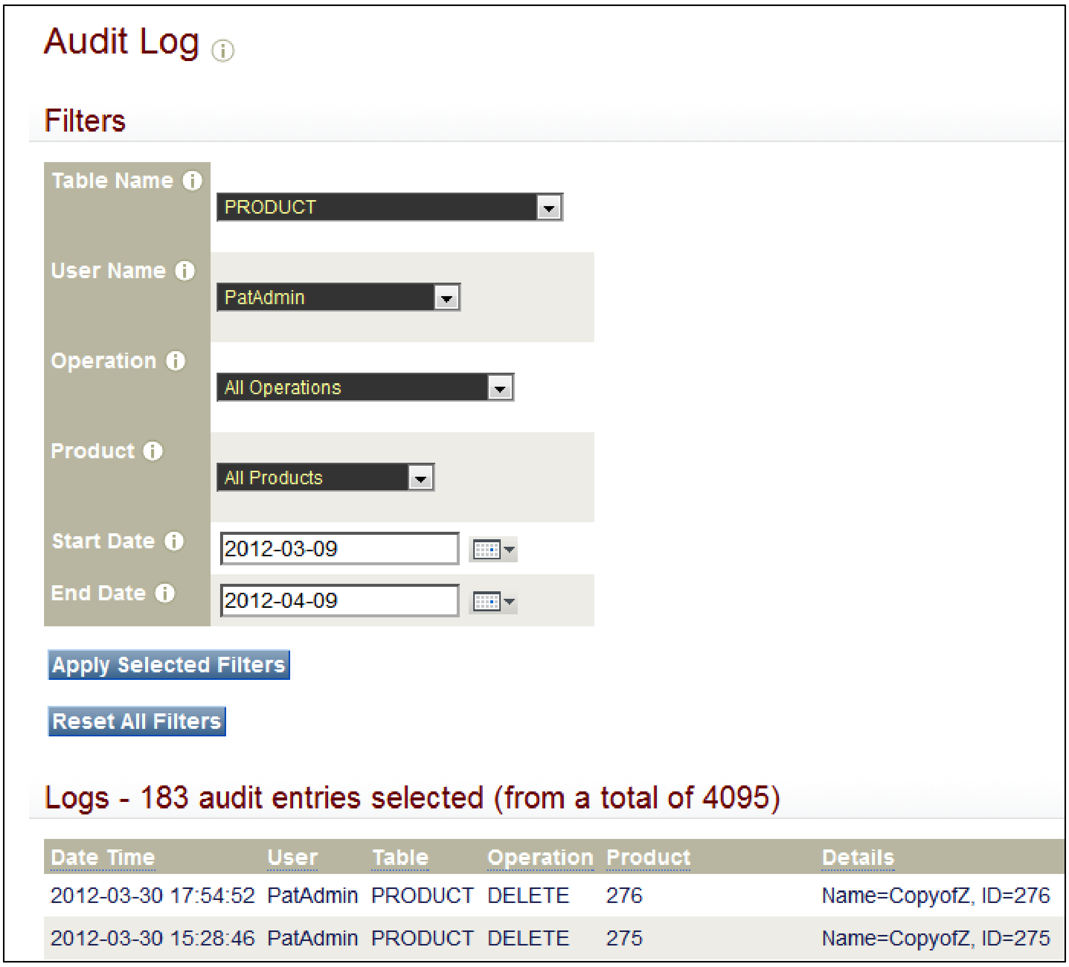
Authorize **Mike Viewer** for the following products:

Product	Add Product
BUG TRACKER	➕
CHUR	➕
ZUOZ	➕

5.2.2 Check Audit Logs

Audit Entries are created whenever any planning data is created, deleted or modified. This workflow allows you to browse through the log entries for all changes to the ZERMATT data store.

List Audit Log Entries



Audit Log

Filters

Table Name	PRODUCT
User Name	PatAdmin
Operation	All Operations
Product	All Products
Start Date	2012-03-09
End Date	2012-04-09

Logs - 183 audit entries selected (from a total of 4095)

Date Time	User	Table	Operation	Product	Details
2012-03-30 17:54:52	PatAdmin	PRODUCT	DELETE	276	Name=CopyofZ, ID=276
2012-03-30 15:28:46	PatAdmin	PRODUCT	DELETE	275	Name=CopyofZ, ID=275

5.3 Planning Tasks

5.3.1 Release Planning for a Product or a Project

All planning data must belong to a product (the Zermatt name for a large project). Product delivery consists of one or more release cycles, and each release cycle may be further subdivided into Sprints[1] or Project Phases. Typical releases may take several sprints to complete, each sprint delivering a functional (but not complete) testable kit. The idea here is that test cycles validate a specific project phase of the release.

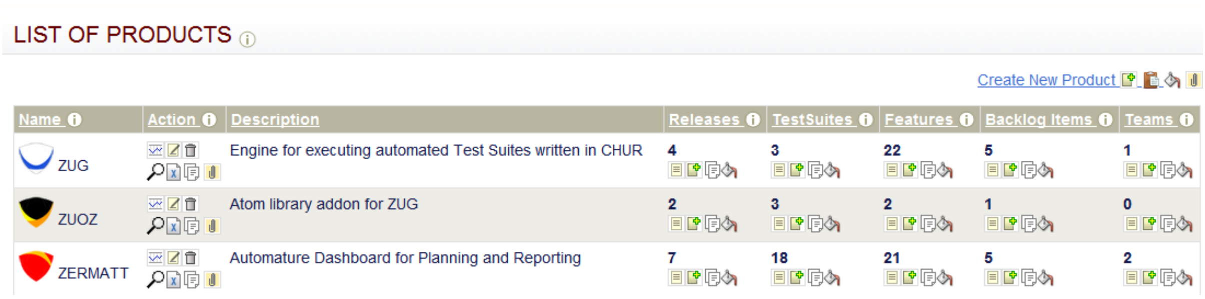
List All Products

Add a New Product, if the desired one does not exist

Remove a product from the existing list

Modify the description of a product

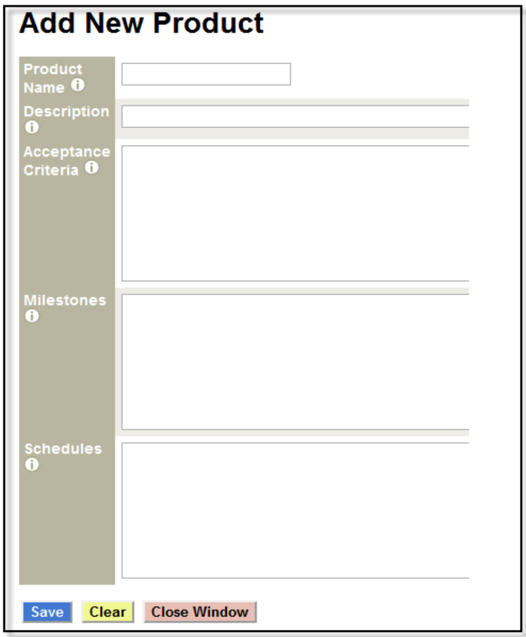
Product Listing Page



LIST OF PRODUCTS

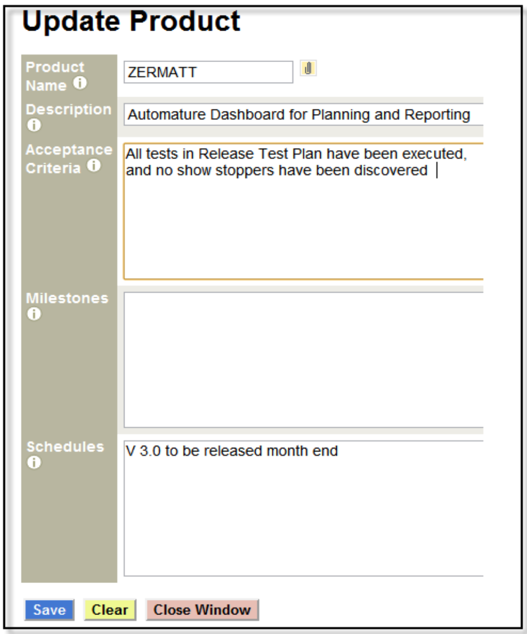
Create New Product

Name	Action	Description	Releases	TestSuites	Features	Backlog Items	Teams
ZUG		Engine for executing automated Test Suites written in CHUR	4	3	22	5	1
ZUOZ		Atom library addon for ZUG	2	3	2	1	0
ZERMATT		Automature Dashboard for Planning and Reporting	7	18	21	5	2



Add New Product

Product Name	
Description	
Acceptance Criteria	
Milestones	
Schedules	



Update Product

Product Name	ZERMATT
Description	Automature Dashboard for Planning and Reporting
Acceptance Criteria	All tests in Release Test Plan have been executed, and no show stoppers have been discovered
Milestones	
Schedules	V 3.0 to be released month end

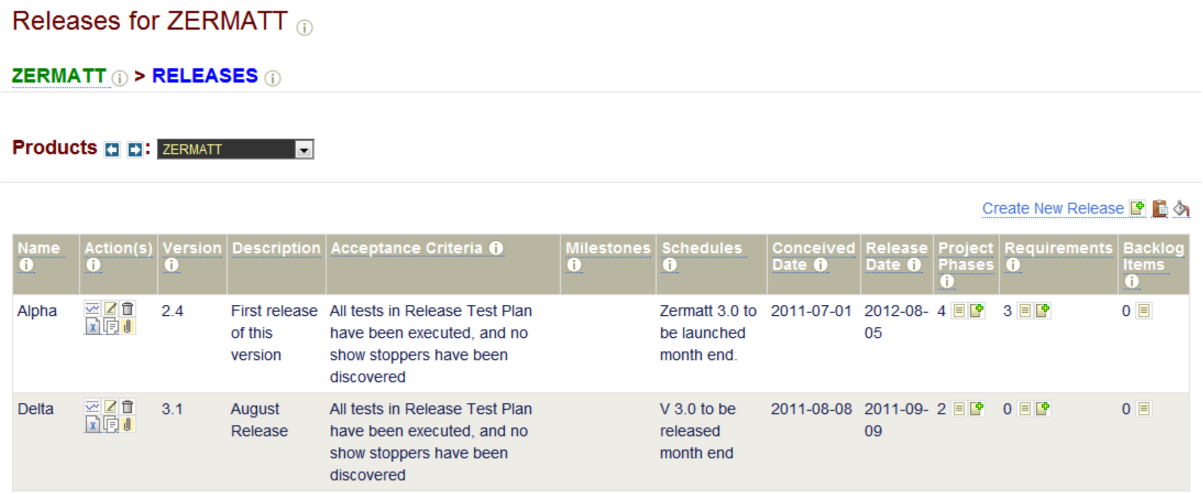
[1] The concept of Sprints is borrowed from Agile Software Development Methodology. One may think of a Sprint as a development phase within a release cycle of a project. In Zermatt, even maintenance releases are referred to as project phases, and belong to a specific release.

List all Releases for a chosen product ☰

Quick create a new release for this product ➕

Remove a release (which does not have any project phases defined) 🗑

Modify the properties of a release ✎



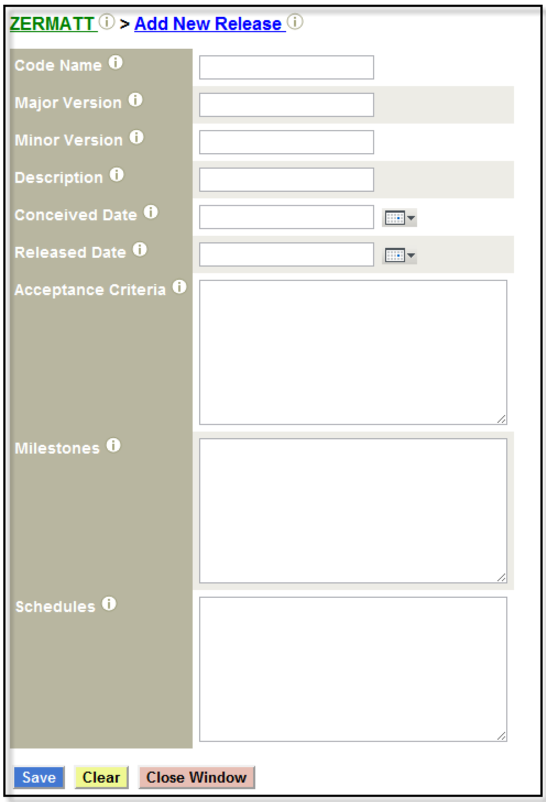
Releases for ZERMATT

ZERMATT > RELEASES

Products: ZERMATT

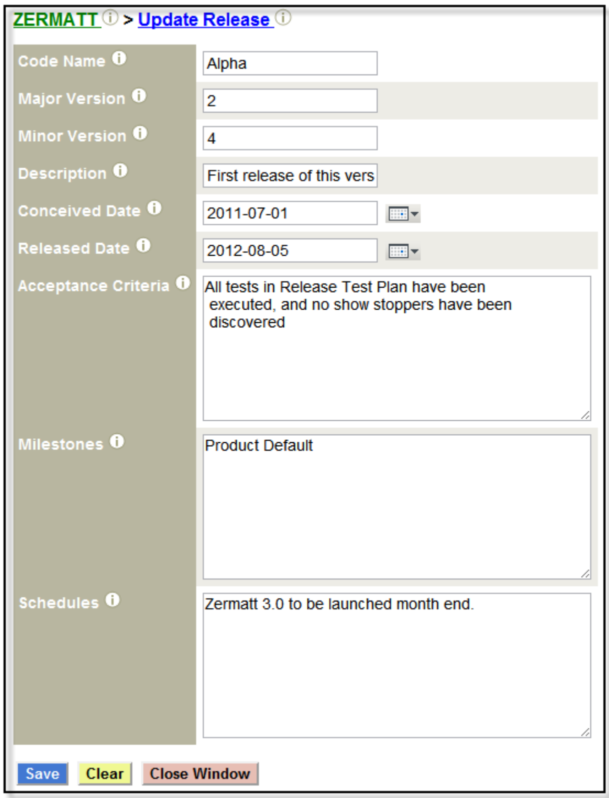
Create New Release

Name	Action(s)	Version	Description	Acceptance Criteria	Milestones	Schedules	Conceived Date	Release Date	Project Phases	Requirements	Backlog Items
Alpha		2.4	First release of this version	All tests in Release Test Plan have been executed, and no show stoppers have been discovered		Zermatt 3.0 to be launched month end.	2011-07-01	2012-08-05	4	3	0
Delta		3.1	August Release	All tests in Release Test Plan have been executed, and no show stoppers have been discovered		V 3.0 to be released month end	2011-08-08	2011-09-09	2	0	0



ZERMATT > Add New Release

Code Name	
Major Version	
Minor Version	
Description	
Conceived Date	
Released Date	
Acceptance Criteria	
Milestones	
Schedules	



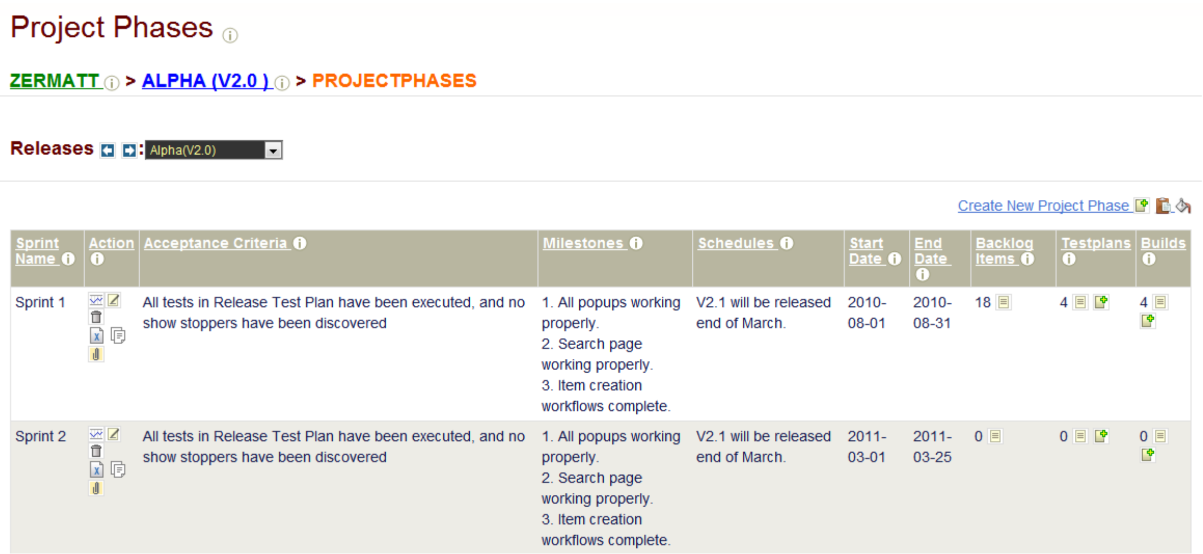
ZERMATT > Update Release

Code Name	Alpha
Major Version	2
Minor Version	4
Description	First release of this vers
Conceived Date	2011-07-01
Released Date	2012-08-05
Acceptance Criteria	All tests in Release Test Plan have been executed, and no show stoppers have been discovered
Milestones	Product Default
Schedules	Zermatt 3.0 to be launched month end.

List all Sprints or Project Phases for a chosen release

Remove a Sprint or Project Phase from a release (not containing a Test Plan)

Modify the properties of a specific Sprint or Project Phase



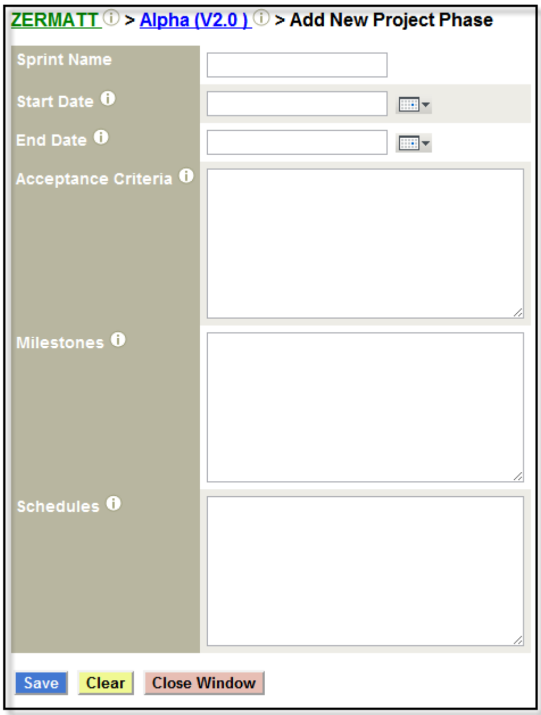
Project Phases

ZERMATT > ALPHA (V2.0) > PROJECTPHASES

Releases : Alpha(V2.0)

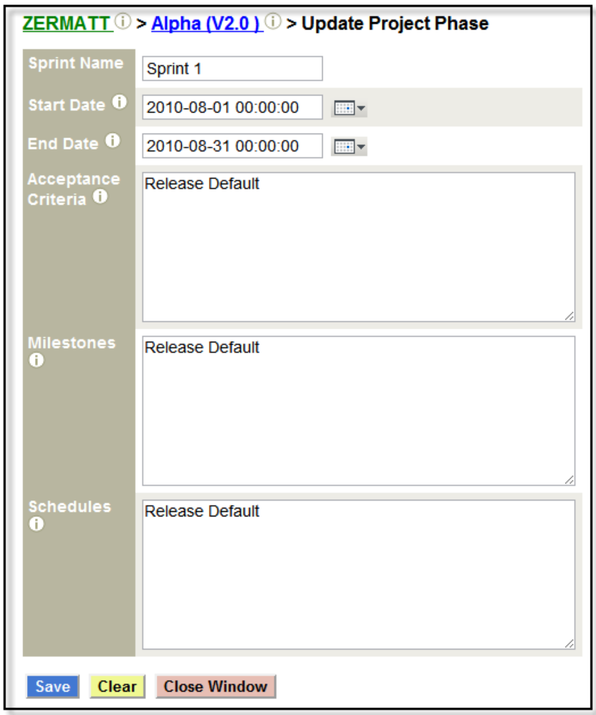
Create New Project Phase

Sprint Name	Action	Acceptance Criteria	Milestones	Schedules	Start Date	End Date	Backlog Items	Testplans	Builds
Sprint 1		All tests in Release Test Plan have been executed, and no show stoppers have been discovered	1. All popups working properly. 2. Search page working properly. 3. Item creation workflows complete.	V2.1 will be released end of March.	2010-08-01	2010-08-31	18	4	4
Sprint 2		All tests in Release Test Plan have been executed, and no show stoppers have been discovered	1. All popups working properly. 2. Search page working properly. 3. Item creation workflows complete.	V2.1 will be released end of March.	2011-03-01	2011-03-25	0	0	0



ZERMATT > Alpha (V2.0) > Add New Project Phase

Sprint Name	
Start Date	
End Date	
Acceptance Criteria	
Milestones	
Schedules	



ZERMATT > Alpha (V2.0) > Update Project Phase

Sprint Name	Sprint 1
Start Date	2010-08-01 00:00:00
End Date	2010-08-31 00:00:00
Acceptance Criteria	Release Default
Milestones	Release Default
Schedules	Release Default

5.3.2 Requirements Planning for a Product

All Projects have requirements that must be delivered. In Zermatt, requirements describe the details belonging to a high level feature. For example, a feature could be that there will be a GUI management interface. The requirements will then spell out all that the management interface will allow you to do. Test cases will typically validate that one or more requirements are adequately implemented.

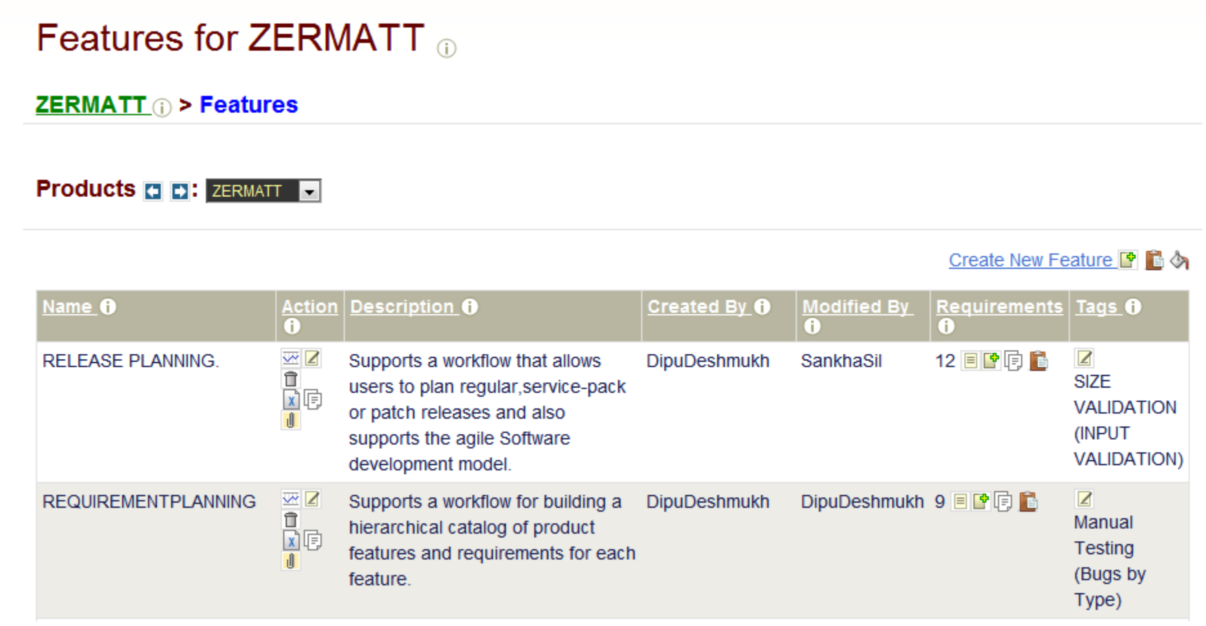
List All Products

List all Features for a chosen product

Add a New (high-level) Feature (e.g. Security) for this product

Remove an existing features (for which, no requirements have been entered)

Modify the name or description of an existing feature

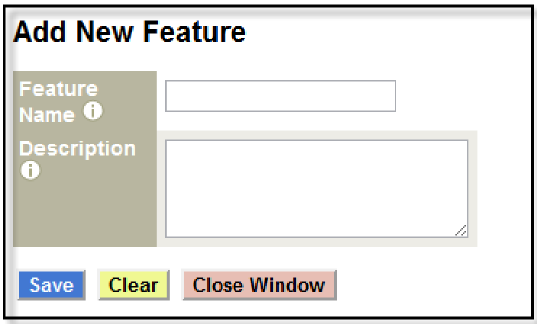


Features for ZERMATT

ZERMATT > Features

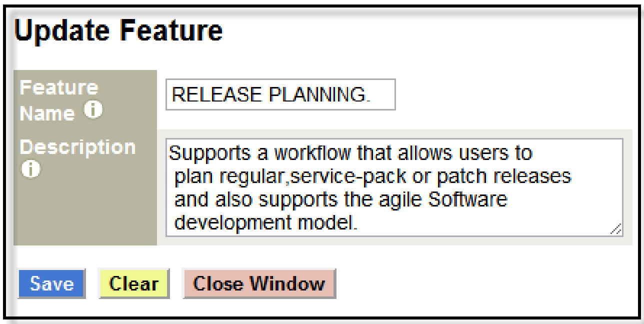
Products : ZERMATT

Name	Action	Description	Created By	Modified By	Requirements	Tags
RELEASE PLANNING.		Supports a workflow that allows users to plan regular,service-pack or patch releases and also supports the agile Software development model.	DipuDeshmukh	SankhaSil	12	SIZE VALIDATION (INPUT VALIDATION)
REQUIREMENTPLANNING		Supports a workflow for building a hierarchical catalog of product features and requirements for each feature.	DipuDeshmukh	DipuDeshmukh	9	Manual Testing (Bugs by Type)



Add New Feature

Feature Name	
Description	



Update Feature

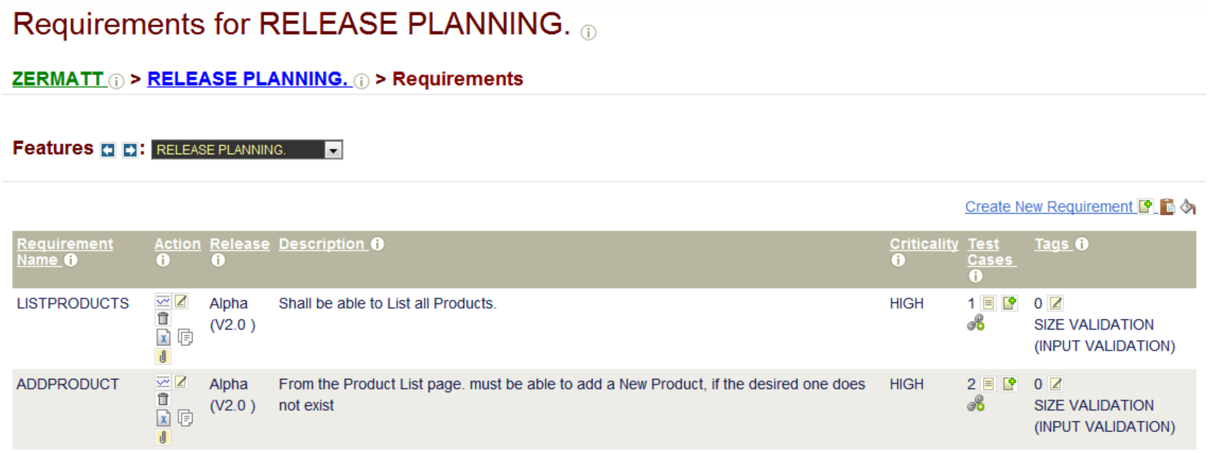
Feature Name	RELEASE PLANNING.
Description	Supports a workflow that allows users to plan regular,service-pack or patch releases and also supports the agile Software development model.

List all Detail Requirements for a chosen Feature

Add a new detail requirement for this feature, and associate a release

Remove an existing detail requirement for this feature

Modify the description or the release association of an existing detail requirement



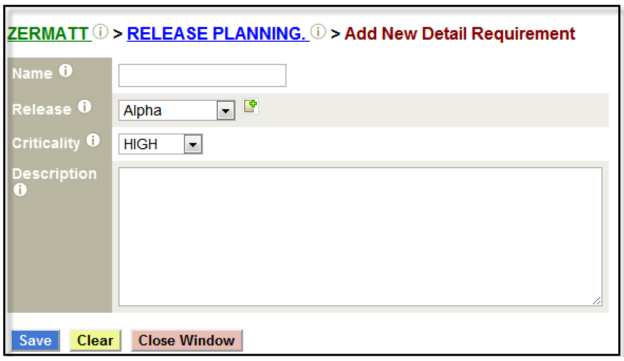
Requirements for RELEASE PLANNING.

ZERMATT > RELEASE PLANNING. > Requirements

Features: RELEASE PLANNING.

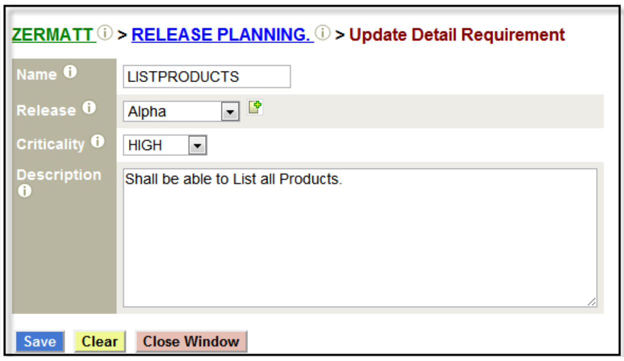
Create New Requirement

Requirement Name	Action	Release	Description	Criticality	Test Cases	Tags
LISTPRODUCTS		Alpha (V2.0)	Shall be able to List all Products.	HIGH	1	0 SIZE VALIDATION (INPUT VALIDATION)
ADDPRODUCT		Alpha (V2.0)	From the Product List page. must be able to add a New Product, if the desired one does not exist	HIGH	2	0 SIZE VALIDATION (INPUT VALIDATION)



ZERMATT > RELEASE PLANNING. > Add New Detail Requirement

Name	
Release	Alpha
Criticality	HIGH
Description	



ZERMATT > RELEASE PLANNING. > Update Detail Requirement

Name	LISTPRODUCTS
Release	Alpha
Criticality	HIGH
Description	Shall be able to List all Products.

5.3.3 Test Case Design

This workflow allows you to specify test cases, and group them under a test suite, so that test cases testing a certain product feature, or similar requirements, can be placed in a common container.

List all Test Suites for a chosen product

Create a New Test Suite for this product

Remove an existing Test Suite that does not have any test cases in it

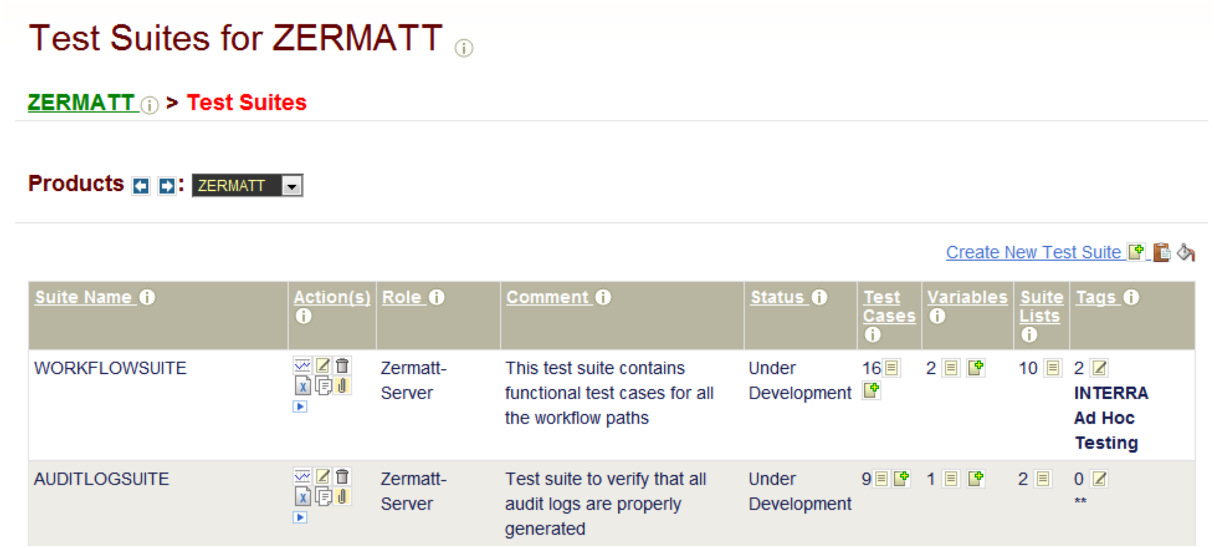
Modify some properties of a Test Suite for this product

List all Test Cases for this Test Suite

Add a New Test Case for this Test Suite

Remove a Test Case that does not have any test execution detail referring to it

Modify properties for a Test Case in this Test Suite



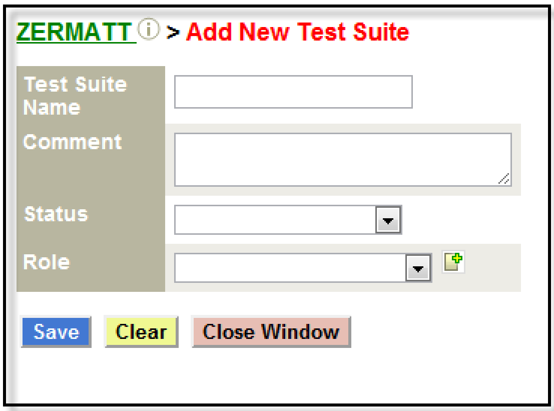
Test Suites for ZERMATT

ZERMATT > Test Suites

Products : ZERMATT

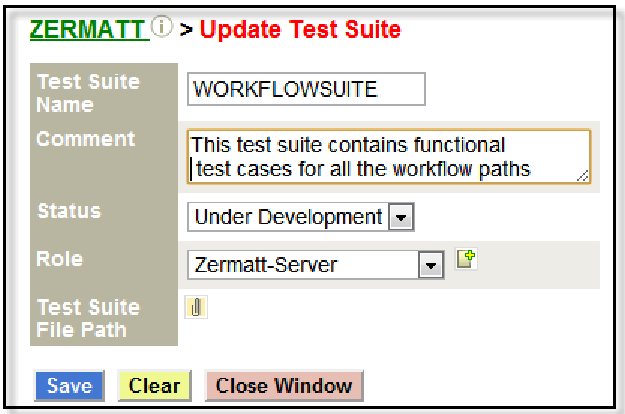
Create New Test Suite

Suite Name	Action(s)	Role	Comment	Status	Test Cases	Variables	Suite Lists	Tags
WORKFLOWSUITE		Zermatt-Server	This test suite contains functional test cases for all the workflow paths	Under Development	16	2	10	2 INTERRA Ad Hoc Testing
AUDITLOGSUITE		Zermatt-Server	Test suite to verify that all audit logs are properly generated	Under Development	9	1	2	0 **



ZERMATT > Add New Test Suite

Test Suite Name	
Comment	
Status	
Role	



ZERMATT > Update Test Suite

Test Suite Name	WORKFLOWSUITE
Comment	This test suite contains functional test cases for all the workflow paths
Status	Under Development
Role	Zermatt-Server
Test Suite File Path	

5.3.4 Test Case to Requirement Mapping

Sometimes the test cases already exist, or have already been specified. In this workflow, the association between a test case and the requirement(s) it validates, begins from the test case.

List all Test Suites for a chosen product

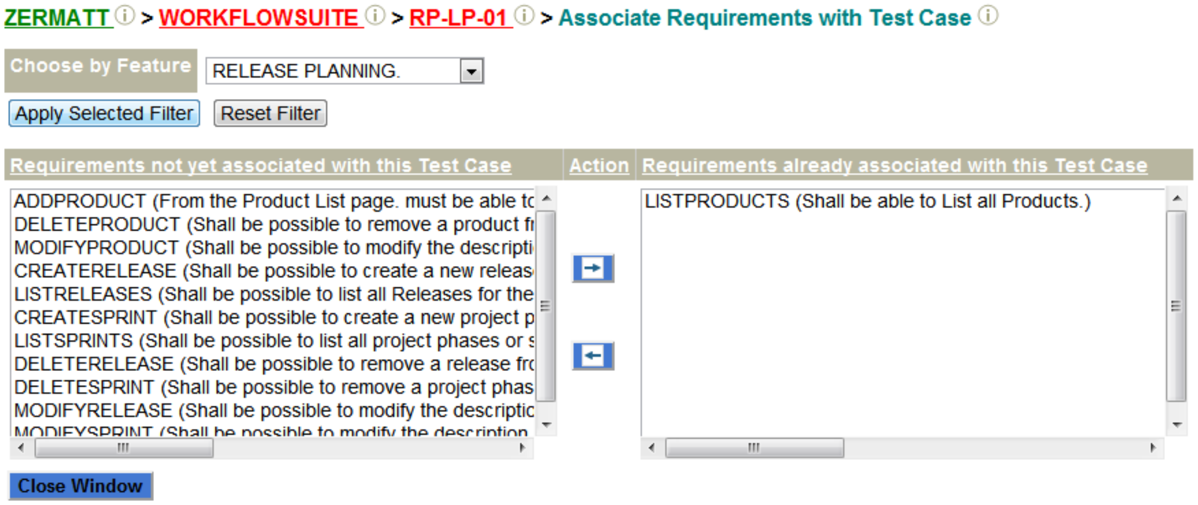
List all Test Cases for this Test Suite

List all Requirements for this Test Case

Add an existing Requirement for this Test Case (choose by Feature)

Remove a Requirement from this Test Case

Create a new Requirement covered by this Test Case


ZERMATT > WORKFLOWSUITE > RP-LP-01 > Associate Requirements with Test Case
Choose by Feature
RELEASE PLANNING.
Apply Selected Filter
Reset Filter
Requirements not yet associated with this Test Case
Action
Requirements already associated with this Test Case
Close Window

ADDPRODUCT (From the Product List page. must be able tc
DELETEPRODUCT (Shall be possible to remove a product fr
MODIFYPRODUCT (Shall be possible to modify the descripti
CREATERELEASE (Shall be possible to create a new releas
LISTRELEASES (Shall be possible to list all Releases for the
CREATESPRINT (Shall be possible to create a new project p
LISTSPRINTS (Shall be possible to list all project phases or s
DELETERELEASE (Shall be possible to remove a release frc
DELETESPRINT (Shall be possible to remove a project phas
MODIFYRELEASE (Shall be possible to modify the descriptic
MODIFYSPRINT (Shall be possible to modify the description

LISTPRODUCTS (Shall be able to List all Products.)

5.3.5 Requirement to Test case mapping

Test cases typically validate that one or more requirements are adequately implemented. This workflow starts from product requirements and identifies which test cases validate them. If a requirement is missing a Testcase, then this workflow will help you create a placeholder test case that can be implemented by automation engineers, or manually executed.

List All Products

List all Features for a chosen product

List all Detail Requirements for chosen Feature

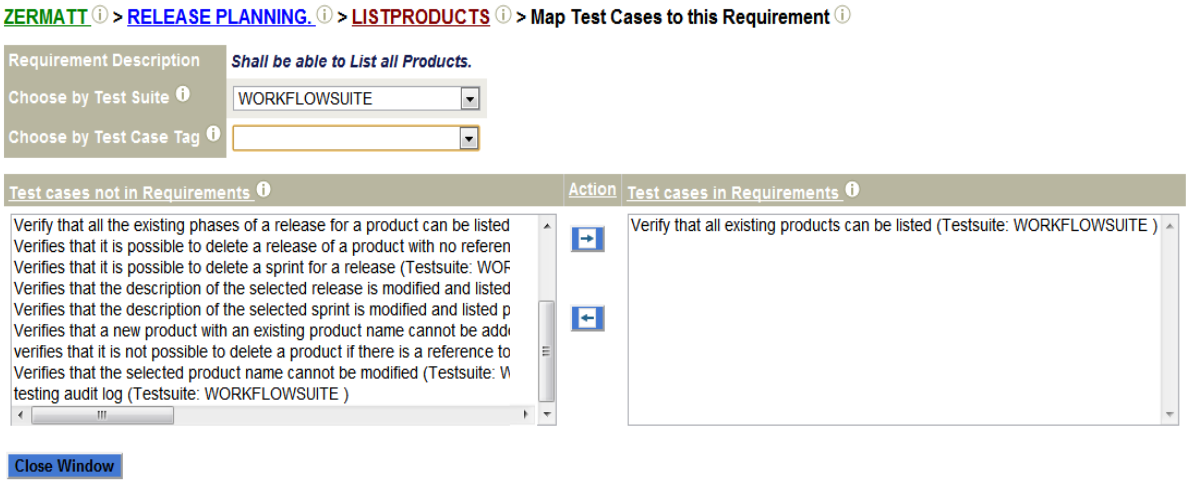
List all Test Cases that test this Requirement

Add an existing Test Case for this Requirement (choose by Tag)

Add an existing Test Case for this Requirement (choose by Test Suite)

Remove a Test Case from this Requirement

Create a new Test Case for this Requirement



ZERMATT > RELEASE PLANNING. > LISTPRODUCTS > Map Test Cases to this Requirement

Requirement Description	*Shall be able to List all Products.*
Choose by Test Suite	WORKFLOWSUITE
Choose by Test Case Tag	

Test cases not in Requirements	Action	Test cases in Requirements
Verify that all the existing phases of a release for a product can be listed Verifies that it is possible to delete a release of a product with no referen Verifies that it is possible to delete a sprint for a release (Testsuite: WOF Verifies that the description of the selected release is modified and listed Verifies that the description of the selected sprint is modified and listed p Verifies that a new product with an existing product name cannot be addı verifies that it is not possible to delete a product if there is a reference to Verifies that the selected product name cannot be modified (Testsuite: W testing audit log (Testsuite: WORKFLOWSUITE)		Verify that all existing products can be listed (Testsuite: WORKFLOWSUITE)

5.3.6 Lab Environment Planning

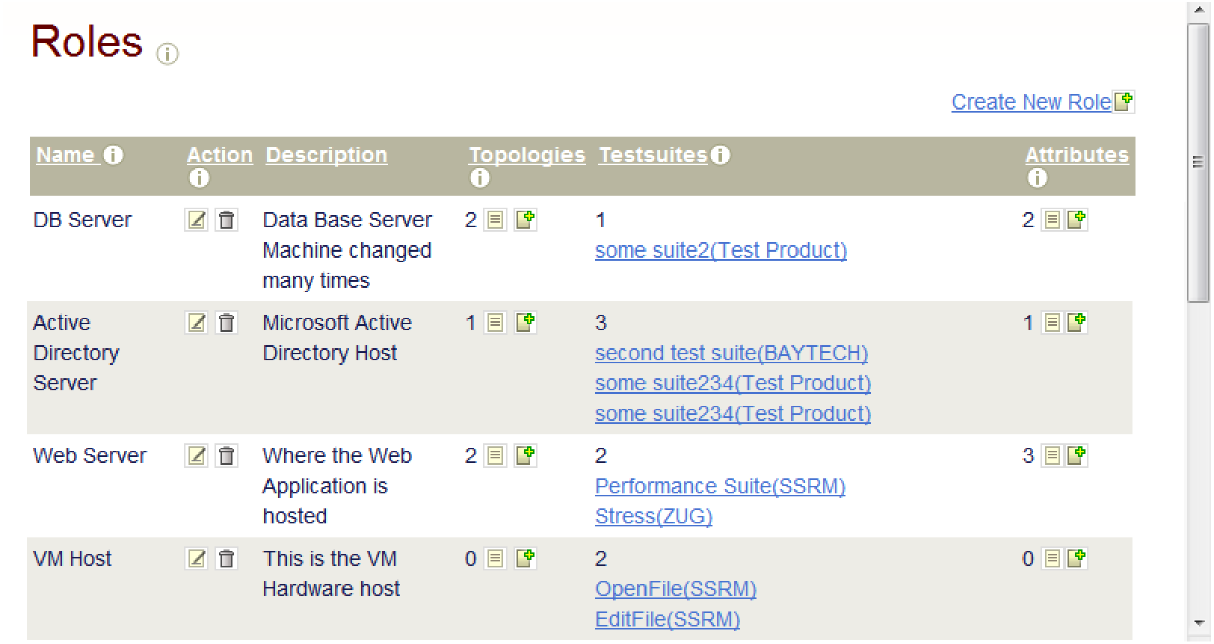
In this workflow section, users can define roles that will participate in a testing environment, and the characteristics of those roles, that may be considered significant for analyzing the effectiveness of the test. To begin, the user will need to create a role, and then define some attributes that will need to be tracked for these roles. Then, various (physical or virtual) machines will need to be selected, and the relevant attribute values will be needed to be entered. Finally, because a test cycle will involve a collection of machines (referred to here as **topologies**) those collections will need to be identified as ***topology sets***.

List All Roles

Add a New Role, if the desired one does not exist

Remove a Role from the existing list

Modify the description of a Role

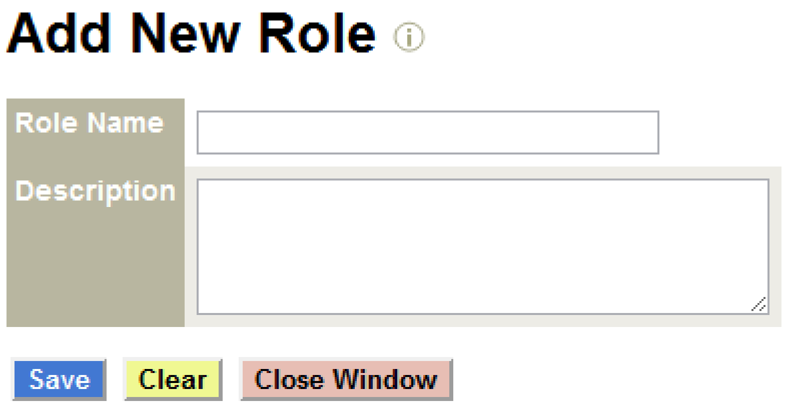


Roles

Create New Role

Name	Action	Description	Topologies	Testsuites	Attributes
DB Server		Data Base Server Machine changed many times	2	1 some suite2(Test Product)	2
Active Directory Server		Microsoft Active Directory Host	1	3 second test suite(BAYTECH) some suite234(Test Product) some suite234(Test Product)	1
Web Server		Where the Web Application is hosted	2	2 Performance Suite(SSRM) Stress(ZUG)	3

VM Host		This is the VM Hardware host	0	2
				OpenFile(SSRM)
				EditFile(SSRM)
					0



Add New Role ⓘ

Role Name

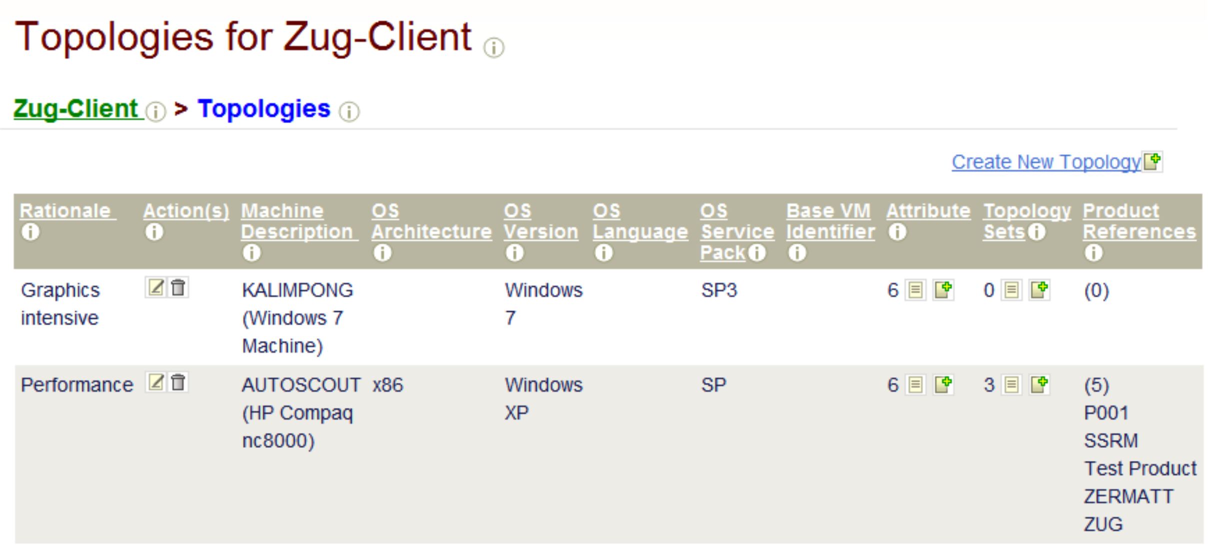
Description

List All (Topology) Attributes for this Role ☰

Add a New Role Attribute, if the desired one does not exist. If a topology already exists for this role, then the attribute will be automatically propagated to that topology ➕

Remove a Role Attribute from the existing list, provided no topologies have been defined for this role 🗑

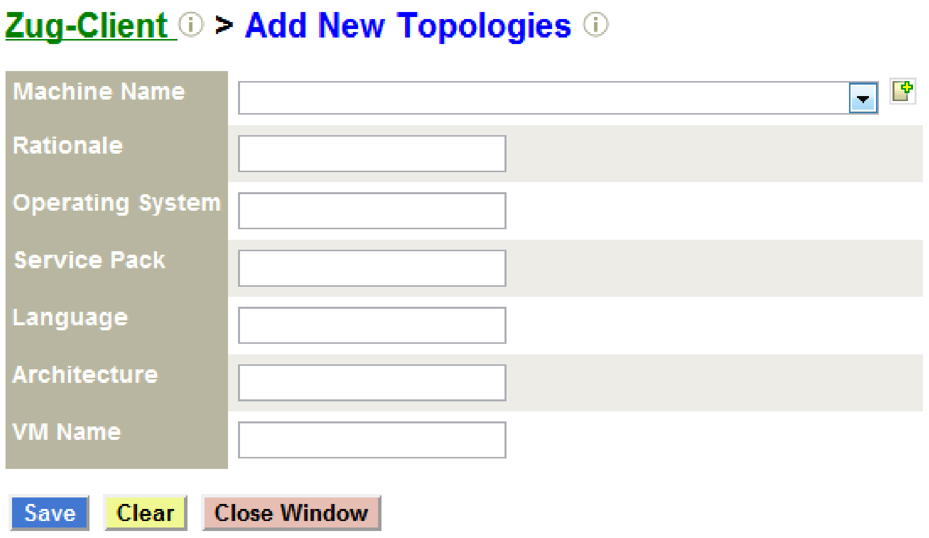
Modify the description or properties of a Role Attribute ✎



Topologies for Zug-Client ⓘ

Zug-Client ⓘ > Topologies ⓘ

Rationale ⓘ	Action(s) ⓘ	Machine Description ⓘ	OS Architecture ⓘ	OS Version ⓘ	OS Language ⓘ	OS Service Pack ⓘ	Base VM Identifier ⓘ	Attribute ⓘ	Topology Sets ⓘ	Product References ⓘ
Graphics intensive	✎ 🗑	KALIMPONG (Windows 7 Machine)		Windows 7		SP3		6 ☰ ➕	0 ☰ ➕	(0)
Performance	✎ 🗑	AUTOSCOUT (HP Compaq nc8000)	x86	Windows XP		SP		6 ☰ ➕	3 ☰ ➕	(5) P001 SSRM Test Product ZERMATT ZUG

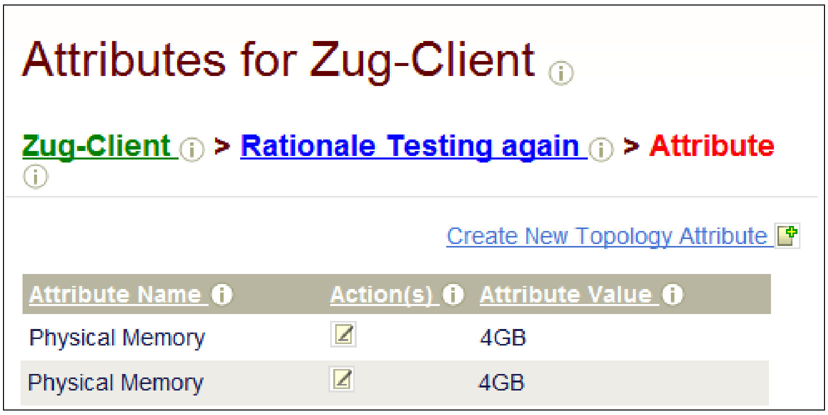


Zug-Client ⓘ > Add New Topologies ⓘ

Machine Name	
Rationale	
Operating System	
Service Pack	
Language	
Architecture	
VM Name	

List attribute value in all Topologies for this Attribute

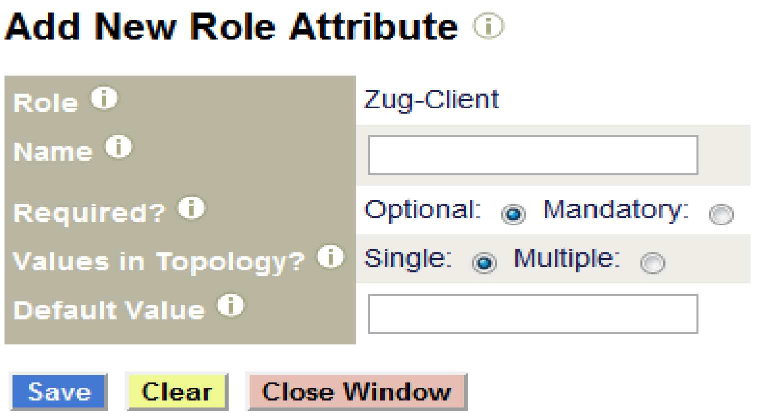
Assign or modify an attribute value for this topology



Attributes for Zug-Client ⓘ

Zug-Client ⓘ > Rationale Testing again ⓘ > Attribute ⓘ

Attribute Name ⓘ	Action(s) ⓘ	Attribute Value ⓘ
Physical Memory		4GB
Physical Memory		4GB



Add New Role Attribute ⓘ

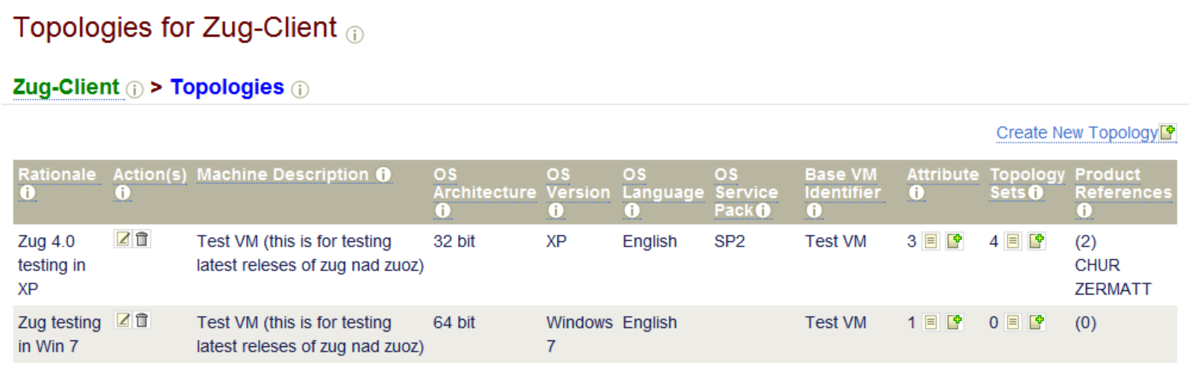
Role ⓘ	Zug-Client
Name ⓘ	
Required? ⓘ	Optional: ● Mandatory: ○
Values in Topology? ⓘ	Single: ● Multiple: ○
Default Value ⓘ	

List All Topologies for this Role

Add a New Topology, if the desired one does not exist. This will automatically inherit all attributes defined for the role represented in this topology

Remove a topology from the existing list

Modify the description of a topology

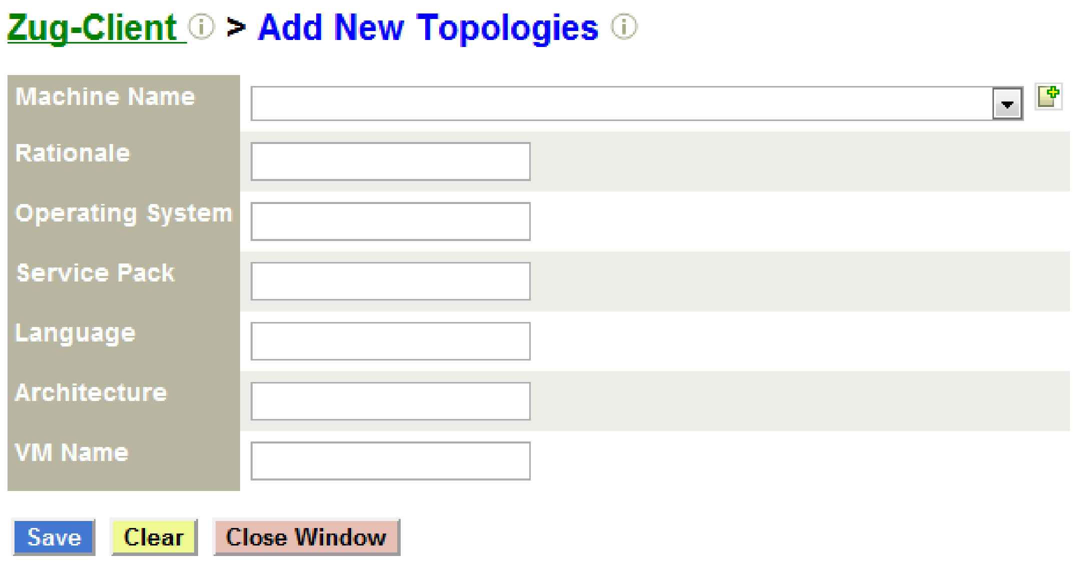


Topologies for Zug-Client

Zug-Client > Topologies

Create New Topology

Rationale	Action(s)	Machine Description	OS Architecture	OS Version	OS Language	OS Service Pack	Base VM Identifier	Attribute	Topology Sets	Product References
Zug 4.0 testing in XP		Test VM (this is for testing latest releses of zug nad zuoz)	32 bit	XP	English	SP2	Test VM	3	4	(2) CHUR ZERMATT
Zug testing in Win 7		Test VM (this is for testing latest releses of zug nad zuoz)	64 bit	Windows 7	English		Test VM	1	0	(0)



Zug-Client > Add New Topologies

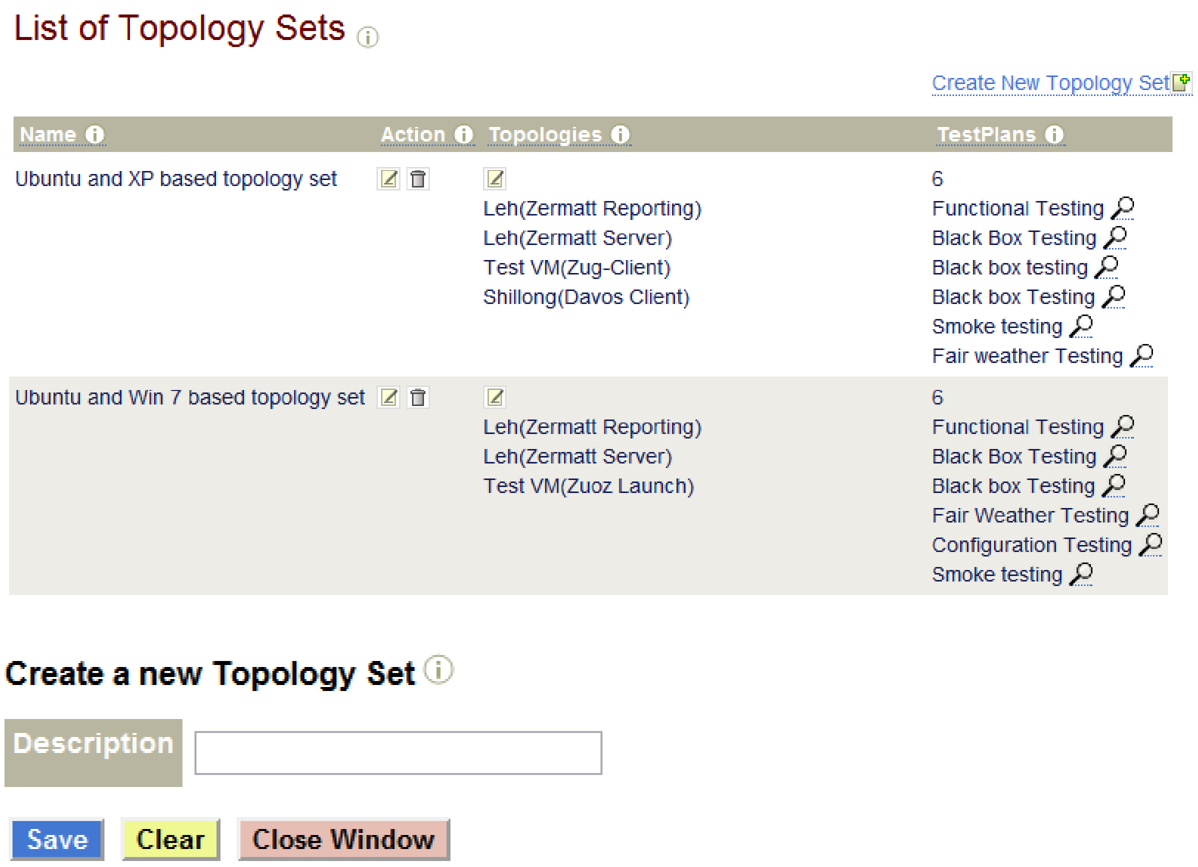
Machine Name	
Rationale	
Operating System	
Service Pack	
Language	
Architecture	
VM Name	

List all Topology Sets this Topology is in

Quick create a new Topology Set

Remove a Topology Set (which does not have any Topologies)

Add a Topology to a Topology Set 1 Remove a Topology from a Topology Set



List of Topology Sets

Create New Topology Set

Name	Action	Topologies	TestPlans

Ubuntu and XP based topology set		Leh(Zermatt Reporting)
		Leh(Zermatt Server)
		Test VM(Zug-Client)
		Shillong(Davos Client)
			6
			Functional Testing
			Black Box Testing
			Black box testing
			Black box Testing
			Smoke testing
			Fair weather Testing

Ubuntu and Win 7 based topology set		Leh(Zermatt Reporting)
		Leh(Zermatt Server)
		Test VM(Zuoz Launch)
			6
			Functional Testing
			Black Box Testing
			Black box Testing
			Fair Weather Testing
			Configuration Testing
			Smoke testing

Create a new Topology Set

Description

5.3.7 Tag Management

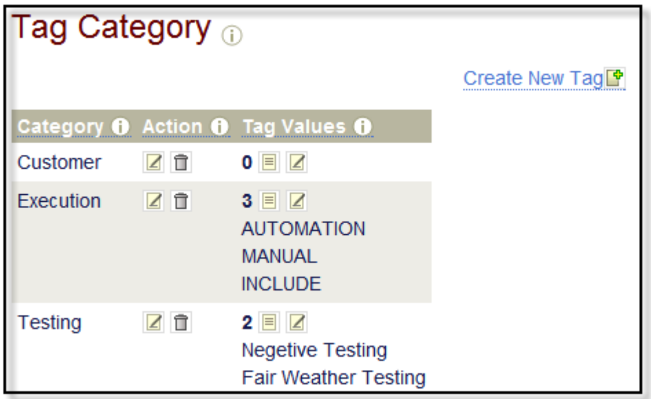
This work flow allows tagging test cases using arbitrary keywords. This feature allows the user to pick and choose test cases by any chosen criteria. For example, you could choose all test cases that verify bugs reported by a specific customer, or all test cases that do stress testing. Tags may be created and assigned at any time during the product lifecycle. Thus test cases may be grouped in many different ways to serve many different purposes, and facilitate quick selection for a particular task.

List all Tag Categories

Create a New Tag Category (e.g. Testplans, Internal-Bugs, Customer-Bugs, Performance, Security, OS, etc...)

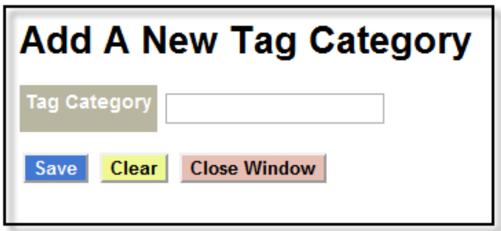
Remove a Tag Category (having no tag values)

Modify the name of a Tag Category



Tag Category

Category	Action	Tag Values
Customer		0
Execution		3 AUTOMATION MANUAL INCLUDE
Testing		2 Negetive Testing Fair Weather Testing



Add A New Tag Category

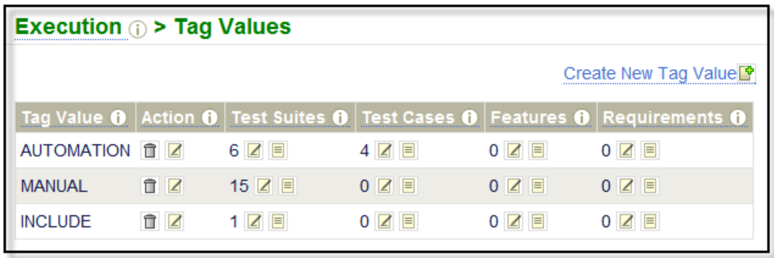
Tag Category	

List all Tags for a given Tag Category

Create a new Tag within the Category (e.g TP5678, Bug1234, Vista, etc...)

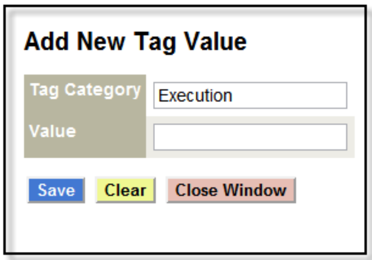
Remove a Tag (not applied to any Test Case or Test Suite)

Modify a Tag



Execution > Tag Values

Tag Value	Action	Test Suites	Test Cases	Features	Requirements
AUTOMATION		6	4	0	0
MANUAL		15	0	0	0
INCLUDE		1	0	0	0



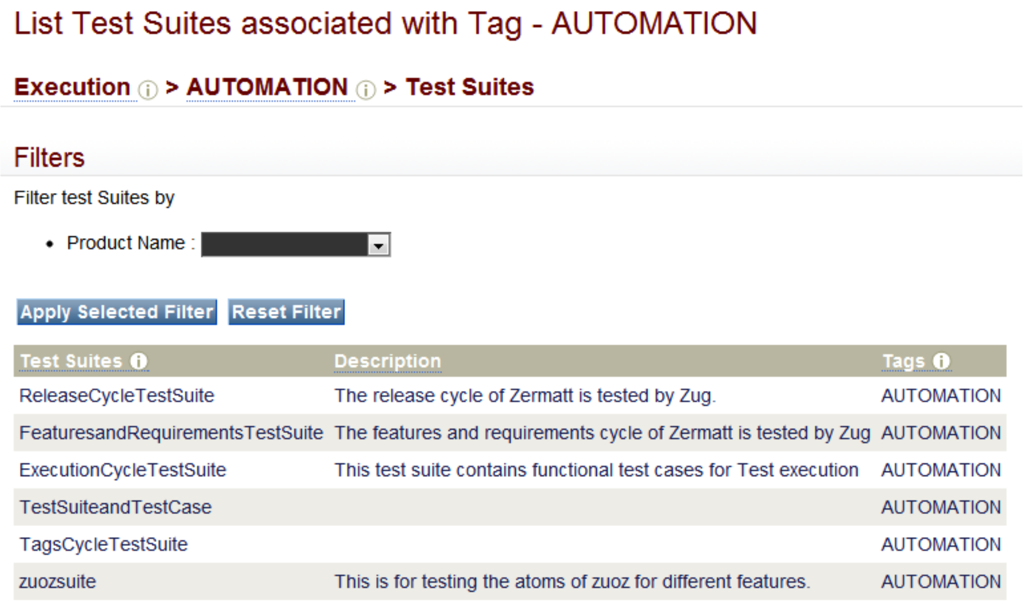
Add New Tag Value

Tag Category	Execution
Value	

List all Test Suites associated with the Tag value

Associate a new Test Suite to this Tag value

Disassociate a Test Suite from this Tag value



List Test Suites associated with Tag - AUTOMATION

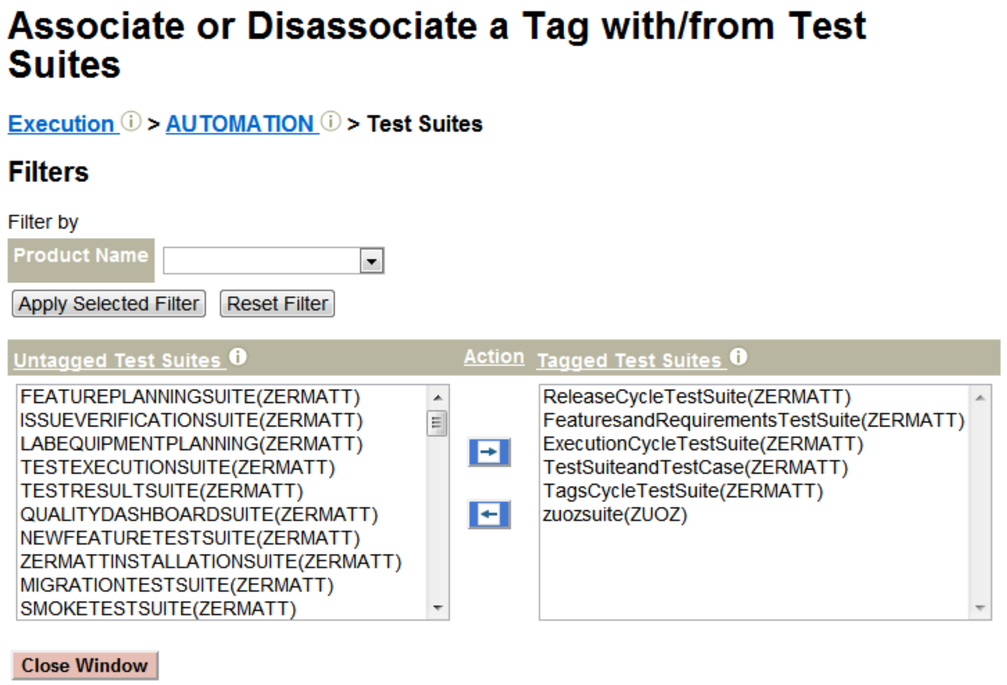
Execution ⓘ > AUTOMATION ⓘ > Test Suites

Filters

Filter test Suites by

- Product Name : [redacted]

Test Suites ⓘ	Description	Tags ⓘ
ReleaseCycleTestSuite	The release cycle of Zermatt is tested by Zug.	AUTOMATION
FeaturesandRequirementsTestSuite	The features and requirements cycle of Zermatt is tested by Zug	AUTOMATION
ExecutionCycleTestSuite	This test suite contains functional test cases for Test execution	AUTOMATION
TestSuiteandTestCase		AUTOMATION
TagsCycleTestSuite		AUTOMATION
zuozsuite	This is for testing the atoms of zuoz for different features.	AUTOMATION



Associate or Disassociate a Tag with/from Test Suites

Execution ⓘ > AUTOMATION ⓘ > Test Suites

Filters

Filter by

Product Name

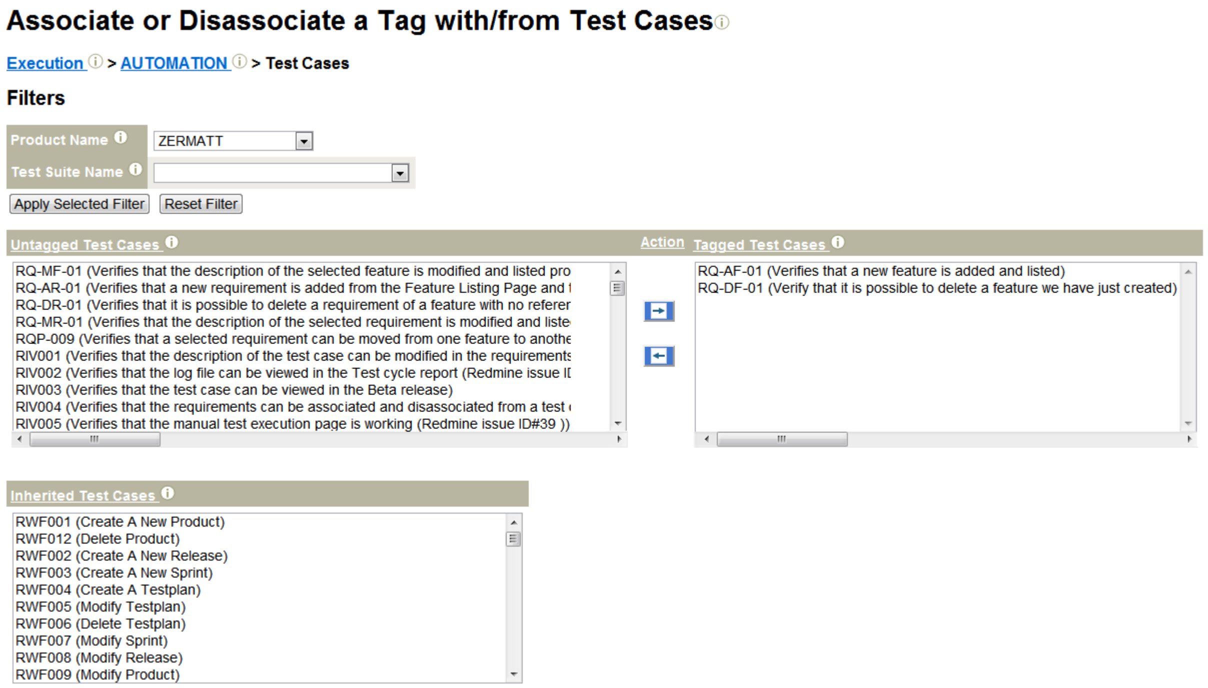
Untagged Test Suites ⓘ	Action	Tagged Test Suites ⓘ

FEATUREPLANNINGSUITE(ZERMATT)
ISSUEVERIFICATIONSUITE(ZERMATT)
LABEQUIPMENTPLANNING(ZERMATT)
TESTEXECUTIONSUITE(ZERMATT)
TESTRESULTSUITE(ZERMATT)
QUALITYDASHBOARDSUITE(ZERMATT)
NEWFEATURETESTSUITE(ZERMATT)
ZERMATTINSTALLATIONSUITE(ZERMATT)
MIGRATIONTESTSUITE(ZERMATT)
SMOKETESTSUITE(ZERMATT)
	→
	←
		ReleaseCycleTestSuite(ZERMATT)
		FeaturesandRequirementsTestSuite(ZERMATT)
		ExecutionCycleTestSuite(ZERMATT)
		TestSuiteandTestCase(ZERMATT)
		TagsCycleTestSuite(ZERMATT)
		zuozsuite(ZUOZ)

List all Test Cases associated with the Tag value

Associate a new Test Case to this Tag value

Disassociate a Test Case from this Tag value



Associate or Disassociate a Tag with/from Test Cases

Execution > AUTOMATION > Test Cases

Filters

Product Name: ZERMATT

Test Suite Name:

Apply Selected Filter   Reset Filter

Untagged Test Cases

RQ-MF-01 (Verifies that the description of the selected feature is modified and listed pro
RQ-AR-01 (Verifies that a new requirement is added from the Feature Listing Page and
RQ-DR-01 (Verifies that it is possible to delete a requirement of a feature with no referer
RQ-MR-01 (Verifies that the description of the selected requirement is modified and liste
RQP-009 (Verifies that a selected requirement can be moved from one feature to anothe
RIV001 (Verifies that the description of the test case can be modified in the requirements
RIV002 (Verifies that the log file can be viewed in the Test cycle report (Redmine issue I
RIV003 (Verifies that the test case can be viewed in the Beta release)
RIV004 (Verifies that the requirements can be associated and disassociated from a test
RIV005 (Verifies that the manual test execution page is working (Redmine issue ID#39))

Action

Tagged Test Cases

RQ-AF-01 (Verifies that a new feature is added and listed)
RQ-DF-01 (Verify that it is possible to delete a feature we have just created)

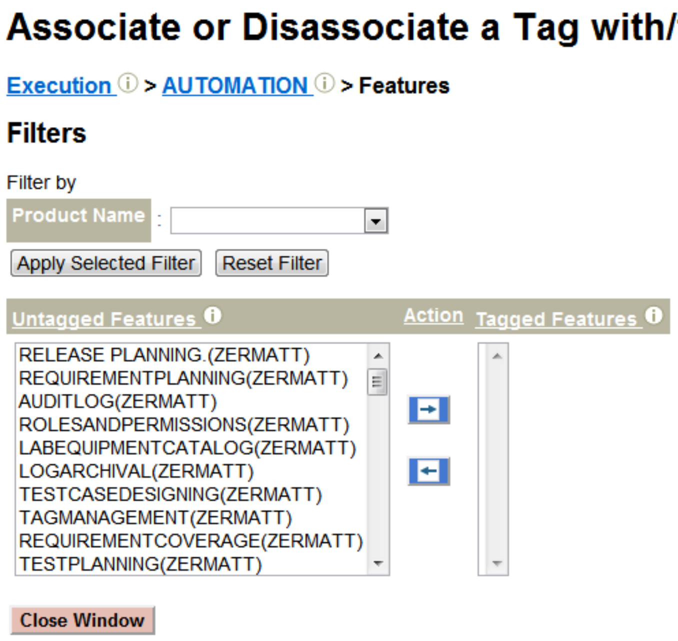
Inherited Test Cases

RWF001 (Create A New Product)
RWF012 (Delete Product)
RWF002 (Create A New Release)
RWF003 (Create A New Sprint)
RWF004 (Create A Testplan)
RWF005 (Modify Testplan)
RWF006 (Delete Testplan)
RWF007 (Modify Sprint)
RWF008 (Modify Release)
RWF009 (Modify Product)

List all Features associated with the Tag value

Associate a new Feature to this Tag value

Disassociate a Feature from this Tag value



Associate or Disassociate a Tag with/from Features

Execution > AUTOMATION > Features

Filters

Filter by

Product Name :

Untagged Features

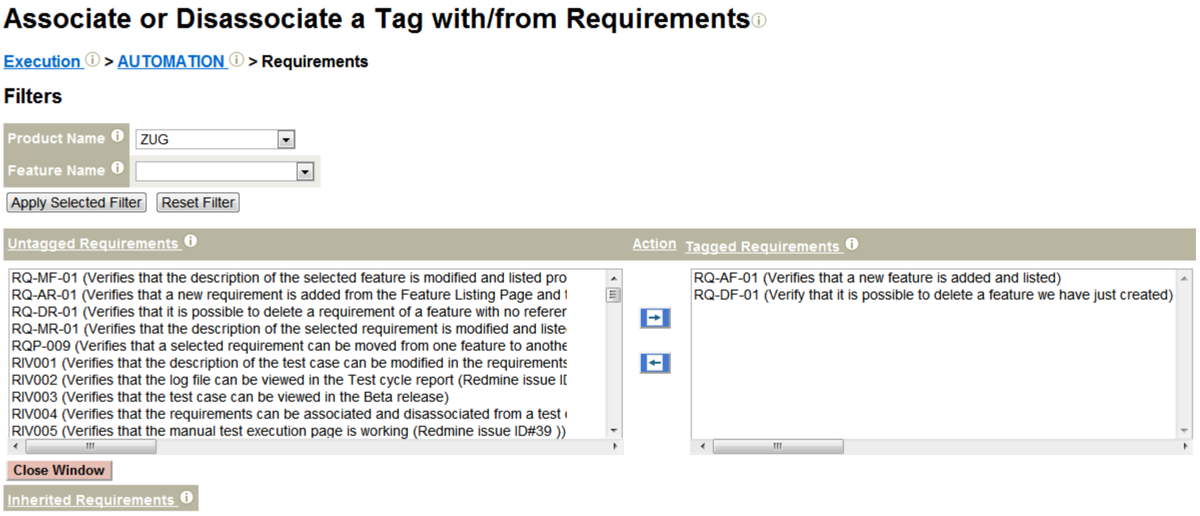
- RELEASE PLANNING.(ZERMATT)
- REQUIREMENTPLANNING(ZERMATT)
- AUDITLOG(ZERMATT)
- ROLESANDPERMISSIONS(ZERMATT)
- LABEQUIPMENTCATALOG(ZERMATT)
- LOGARCHIVAL(ZERMATT)
- TESTCASEDESIGNING(ZERMATT)
- TAGMANAGEMENT(ZERMATT)
- REQUIREMENTCOVERAGE(ZERMATT)
- TESTPLANNING(ZERMATT)

Tagged Features

List all Requirements associated with the Tag value

Associate a new Requirement to this Tag value

Disassociate a Requirement from this Tag value



Associate or Disassociate a Tag with/from Requirements

Execution > AUTOMATION > Requirements

Filters

Product Name: ZUG

Feature Name:

Untagged Requirements

- RQ-MF-01 (Verifies that the description of the selected feature is modified and listed pro
- RQ-AR-01 (Verifies that a new requirement is added from the Feature Listing Page and t
- RQ-DR-01 (Verifies that it is possible to delete a requirement of a feature with no referer
- RQ-MR-01 (Verifies that the description of the selected requirement is modified and liste
- RQP-009 (Verifies that a selected requirement can be moved from one feature to anothe
- RIV001 (Verifies that the description of the test case can be modified in the requirements
- RIV002 (Verifies that the log file can be viewed in the Test cycle report (Redmine issue II
- RIV003 (Verifies that the test case can be viewed in the Beta release)
- RIV004 (Verifies that the requirements can be associated and disassociated from a test
- RIV005 (Verifies that the manual test execution page is working (Redmine issue ID#39))

Tagged Requirements

- RQ-AF-01 (Verifies that a new feature is added and listed)
- RQ-DF-01 (Verify that it is possible to delete a feature we have just created)

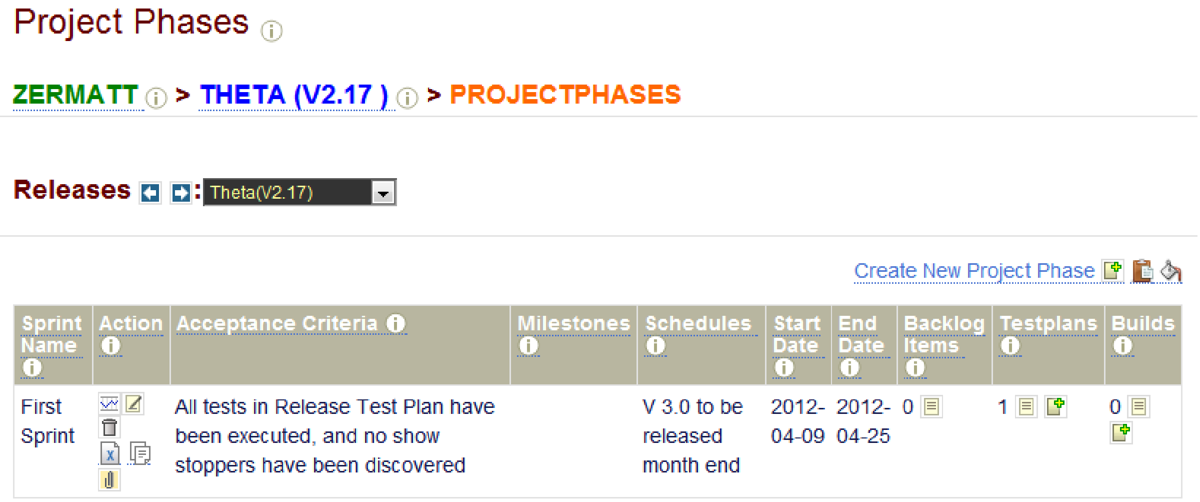
5.3.8 Release/Sprint Planning

This workflow is designed to allow the person responsible for planning the testing for a sprint or a release to choose test suites, test cases, test environments for carrying out a series of tests, designed to meet the objectives of the release or the sprint. This workflow assumes that the test cases already exist, and have been tagged, and the test environments have already been entered into Zermatt.

List All Products

List all Releases for a chosen product

List all Sprints or Project Phases for a chosen release



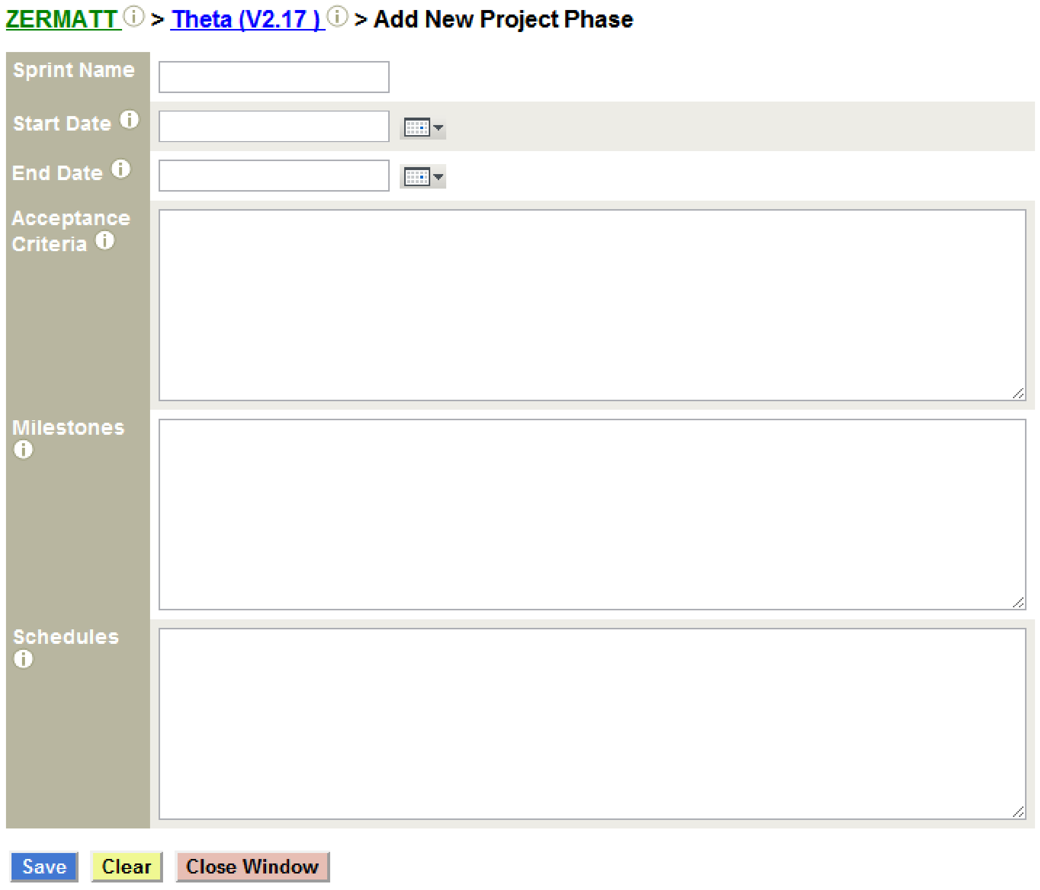
Project Phases

ZERMATT > THETA (V2.17) > PROJECTPHASES

Releases : Theta(V2.17)

Create New Project Phase

Sprint Name	Action	Acceptance Criteria	Milestones	Schedules	Start Date	End Date	Backlog Items	Testplans	Builds
First Sprint		All tests in Release Test Plan have been executed, and no show stoppers have been discovered		V 3.0 to be released month end	2012-04-09	2012-04-25	0	1	0



ZERMATT > Theta (V2.17) > Add New Project Phase

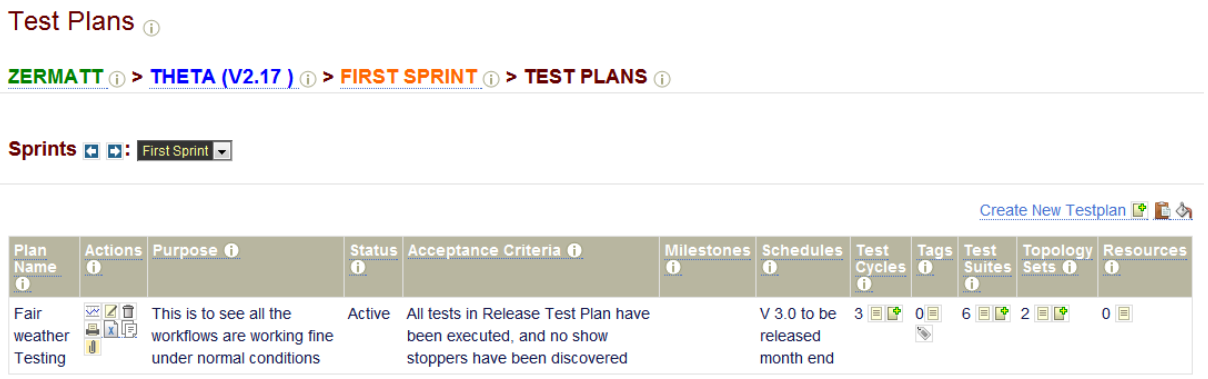
Sprint Name	
Start Date	
End Date	
Acceptance Criteria	
Milestones	
Schedules	

List all Test Plans for a chosen Sprint or Project Phase (Normally there will be at most one such Test Plan)

Create a Test Plan for a chosen Sprint or Project Phase

Remove an existing Test Plan (provided there are no test suite lists or topology set lists referencing this test plan)

Modify an existing Test Plan



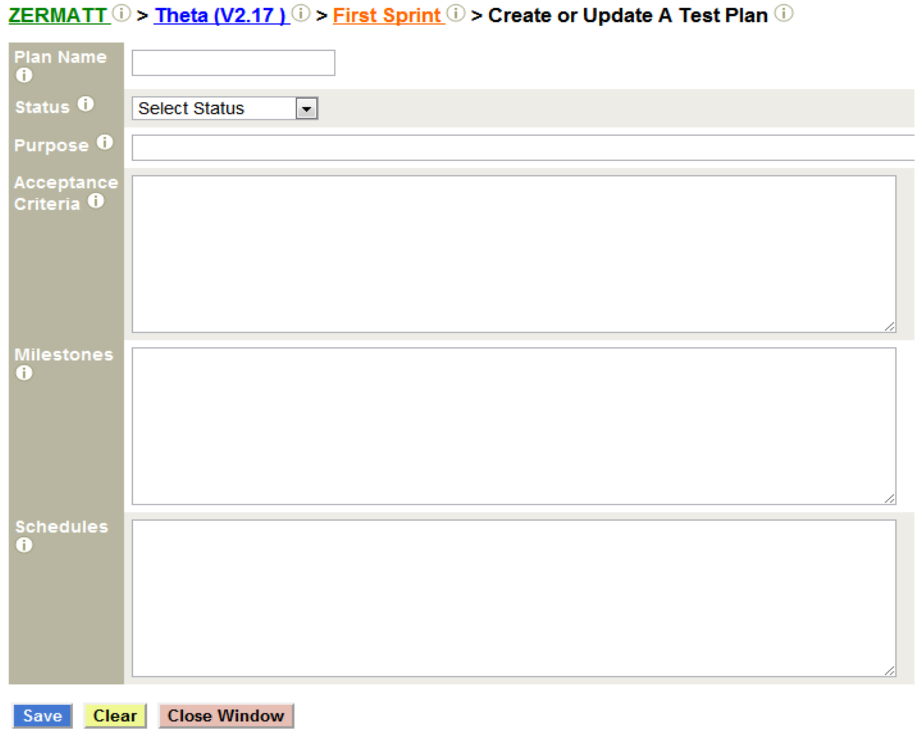
Test Plans

ZERMATT > THETA (V2.17) > FIRST SPRINT > TEST PLANS

Sprints : First Sprint

Create New Testplan

Plan Name	Actions	Purpose	Status	Acceptance Criteria	Milestones	Schedules	Test Cycles	Tags	Test Suites	Topology Sets	Resources
Fair weather Testing		This is to see all the workflows are working fine under normal conditions	Active	All tests in Release Test Plan have been executed, and no show stoppers have been discovered		V 3.0 to be released month end	3	0	6	2	0

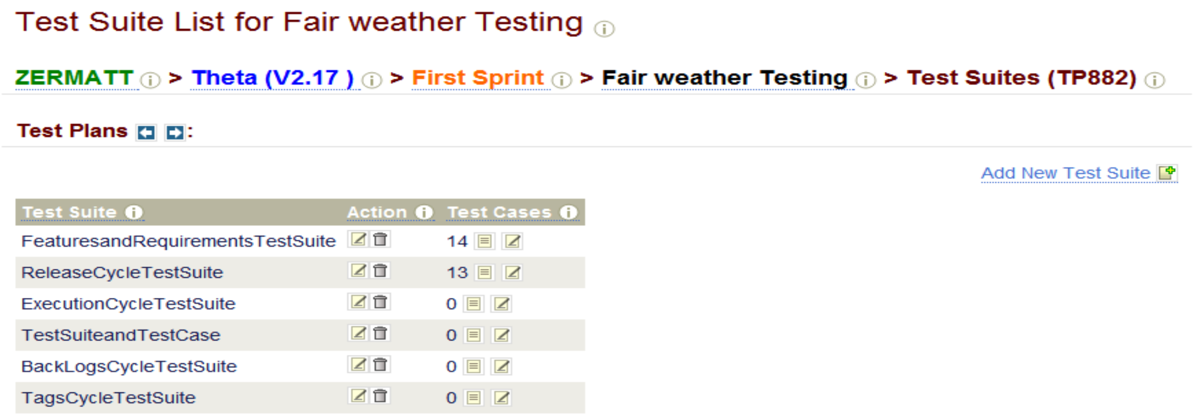


ZERMATT > Theta (V2.17) > First Sprint > Create or Update A Test Plan

Plan Name	
Status	Select Status
Purpose	
Acceptance Criteria	
Milestones	
Schedules	

List all Test Suites for a chosen Test Plan

Add a Test Suite to a chosen Test Plan's List of Test Suites



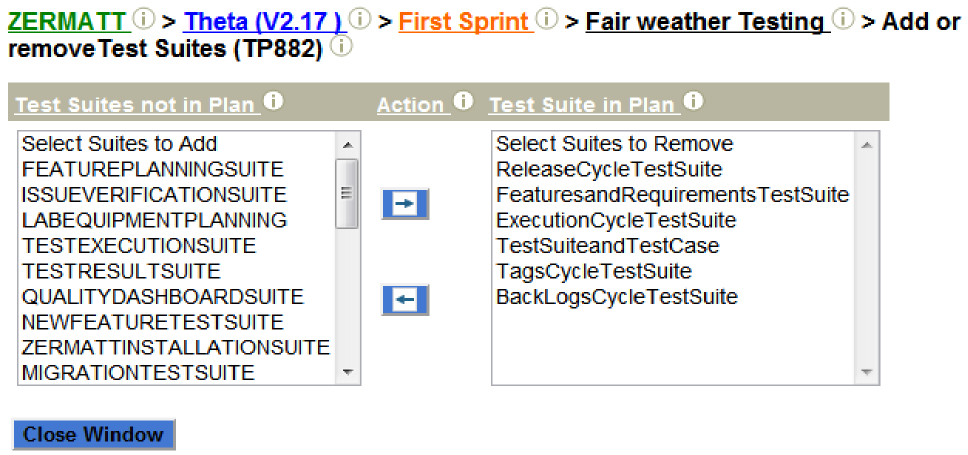
Test Suite List for Fair weather Testing

ZERMATT > Theta (V2.17) > First Sprint > Fair weather Testing > Test Suites (TP882)

Test Plans :

Add New Test Suite

Test Suite	Action	Test Cases
FeaturesandRequirementsTestSuite		14
ReleaseCycleTestSuite		13
ExecutionCycleTestSuite		0
TestSuiteandTestCase		0
BackLogsCycleTestSuite		0
TagsCycleTestSuite		0



ZERMATT > Theta (V2.17) > First Sprint > Fair weather Testing > Add or removeTest Suites (TP882)

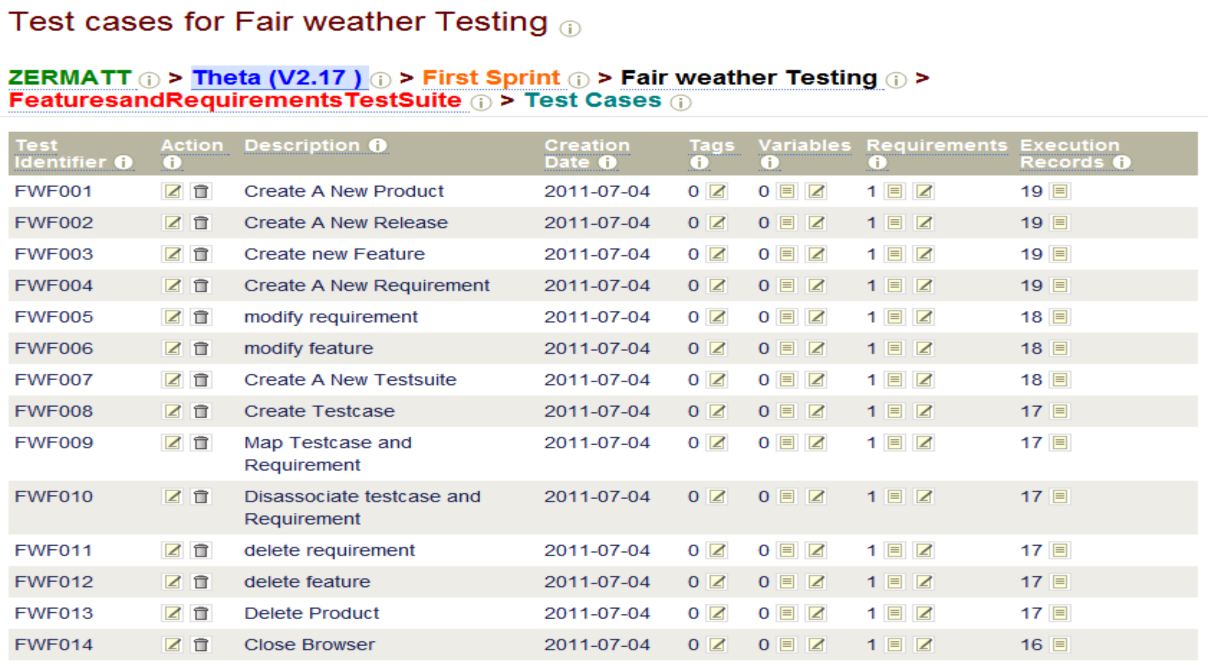
Test Suites not in Plan	Action	Test Suite in Plan
Select Suites to Add FEATUREPLANNINGSUITE ISSUEVERIFICATIONSUITE LABEQUIPMENTPLANNING TESTEXECUTIONSUITE TESTRESULTSUITE QUALITYDASHBOARDSUITE NEWFEATURETESTSUITE ZERMATTINSTALLATIONSUITE MIGRATIONTESTSUITE	→ ←	Select Suites to Remove ReleaseCycleTestSuite FeaturesandRequirementsTestSuite ExecutionCycleTestSuite TestSuiteandTestCase TagsCycleTestSuite BackLogsCycleTestSuite

List all selected Test Cases in the chosen Test Suite for this Test Plan (select the Test Cases, which are tagged for this Test Plan)

Add a Test Case to the Test Plan for this Test Suite (Tag the test case with the Test Plan ID)

Remove a Test Case from the Test Plan for this Test Suite (Untag the Test Plan ID tag from this test case)

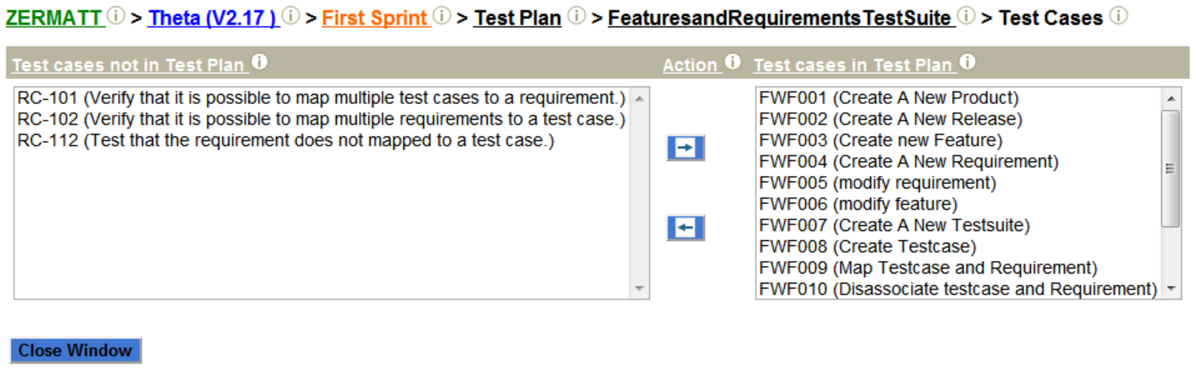
Remove a Test Suite from a chosen Test Plan's list of Test Suites



Test cases for Fair weather Testing ⓘ

ZERMATT ⓘ > Theta (V2.17) ⓘ > First Sprint ⓘ > Fair weather Testing ⓘ > FeaturesandRequirementsTestSuite ⓘ > Test Cases ⓘ

Test Identifier ⓘ	Action ⓘ	Description ⓘ	Creation Date ⓘ	Tags ⓘ	Variables ⓘ	Requirements ⓘ	Execution Records ⓘ
FWF001	✎ 🗑	Create A New Product	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	19 ≣
FWF002	✎ 🗑	Create A New Release	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	19 ≣
FWF003	✎ 🗑	Create new Feature	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	19 ≣
FWF004	✎ 🗑	Create A New Requirement	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	19 ≣
FWF005	✎ 🗑	modify requirement	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	18 ≣
FWF006	✎ 🗑	modify feature	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	18 ≣
FWF007	✎ 🗑	Create A New Testsuite	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	18 ≣
FWF008	✎ 🗑	Create Testcase	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	17 ≣
FWF009	✎ 🗑	Map Testcase and Requirement	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	17 ≣
FWF010	✎ 🗑	Disassociate testcase and Requirement	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	17 ≣
FWF011	✎ 🗑	delete requirement	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	17 ≣
FWF012	✎ 🗑	delete feature	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	17 ≣
FWF013	✎ 🗑	Delete Product	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	17 ≣
FWF014	✎ 🗑	Close Browser	2011-07-04	0 ✎	0 ≣ ✎	1 ≣ ✎	16 ≣



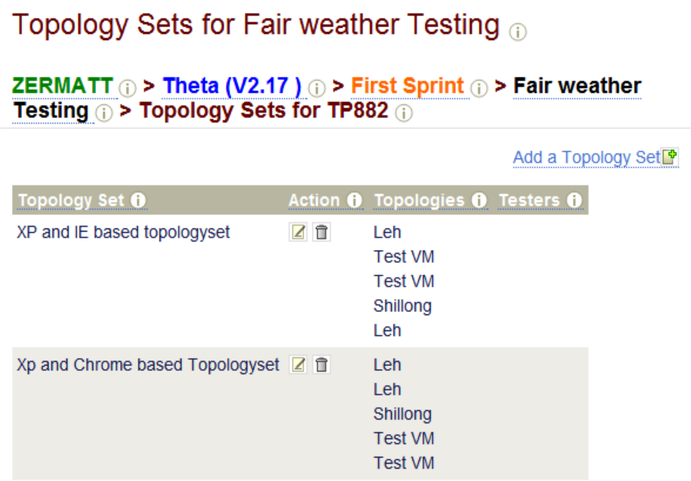
ZERMATT ⓘ > Theta (V2.17) ⓘ > First Sprint ⓘ > Test Plan ⓘ > FeaturesandRequirementsTestSuite ⓘ > Test Cases ⓘ

Test cases not in Test Plan ⓘ	Action ⓘ	Test cases in Test Plan ⓘ
RC-101 (Verify that it is possible to map multiple test cases to a requirement.) RC-102 (Verify that it is possible to map multiple requirements to a test case.) RC-112 (Test that the requirement does not mapped to a test case.)	→ ←	FWF001 (Create A New Product) FWF002 (Create A New Release) FWF003 (Create new Feature) FWF004 (Create A New Requirement) FWF005 (modify requirement) FWF006 (modify feature) FWF007 (Create A New Testsuite) FWF008 (Create Testcase) FWF009 (Map Testcase and Requirement) FWF010 (Disassociate testcase and Requirement)

List all Topology Sets for a chosen Test Plan ≣

Add a Topology Set to a chosen Test Plan's List of Topology Sets ✚

Remove a Topology Set from a chosen Test Plan's list of Topology Sets 🗑

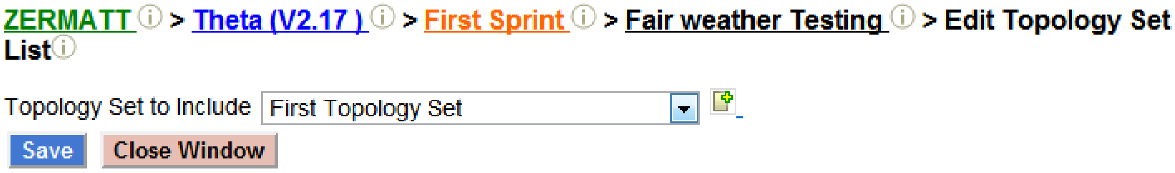


Topology Sets for Fair weather Testing

ZERMATT > Theta (V2.17) > First Sprint > Fair weather Testing > Topology Sets for TP882

Add a Topology Set

Topology Set	Action	Topologies	Testers
XP and IE based topologyset		Leh Test VM Test VM Shillong Leh	
Xp and Chrome based Topologyset		Leh Leh Shillong Test VM Test VM	



ZERMATT > Theta (V2.17) > First Sprint > Fair weather Testing > Edit Topology Set List

Topology Set to Include: First Topology Set

5.3.9 Copy / Paste Function

Zermatt allows you to copy the data under an object and then allows you to paste it in the desired location.

Copy Feature

Allow you to copy an object (E.g. Product, Release, Test Suite, Feature, Project phase, Test plan, Test Case, requirement.

Shallow Paste

Allow you to paste the parent object without the objects under it (E.g.: If you shallow paste a product then only the product gets copied and not the objects under it like test suites, release, features etc.)

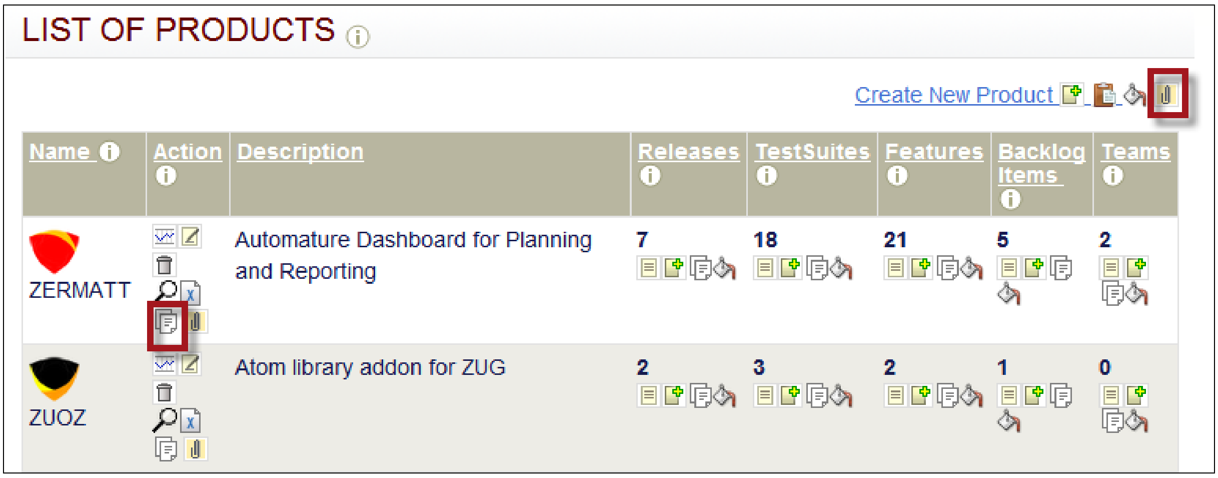
Deep paste

Allow you to paste the parent object with the objects under it (E.g.: If you deep paste a Product then all the Releases, Test suites and features get copied.

Example: For Product

Copy Products / Shallow Paste Products / Deep paste Products

- Product can be copied along with the objects under it/ without the objects under it
- Product can be pasted in the same page without any objects under it
- Product can be pasted in the same page with the objects under it

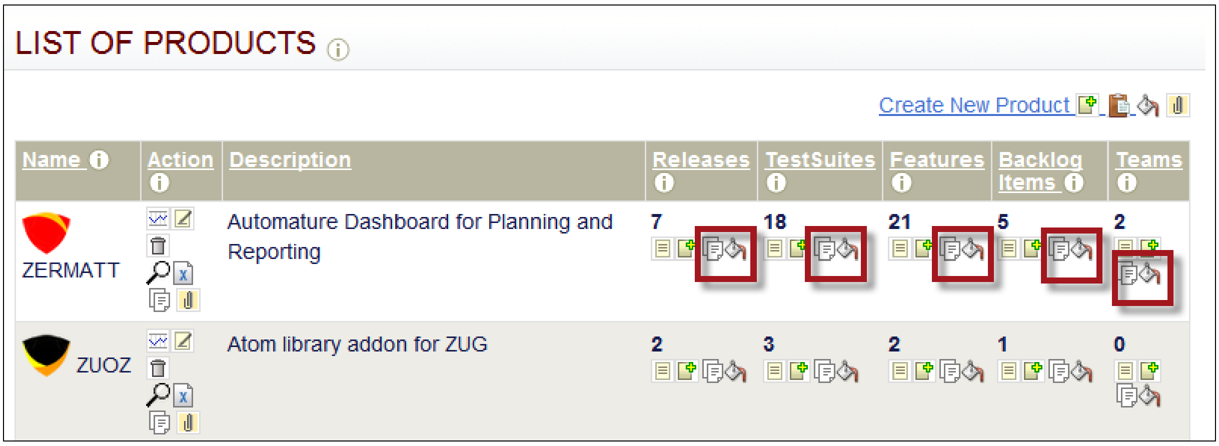


LIST OF PRODUCTS

Create New Product

Name	Action	Description	Releases	TestSuites	Features	Backlog Items	Teams
ZERMATT		Automature Dashboard for Planning and Reporting	7	18	21	5	2
ZUOZ		Atom library addon for ZUG	2	3	2	1	0

Example: To copy and deep paste all releases, Test suites and Features under a product

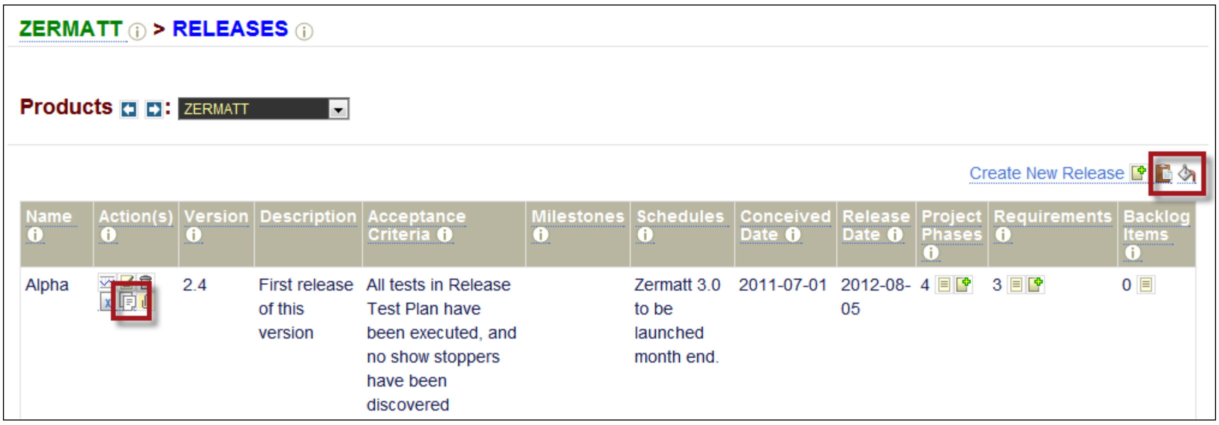


LIST OF PRODUCTS

Create New Product

Name	Action	Description	Releases	TestSuites	Features	Backlog Items	Teams
ZERMATT		Automature Dashboard for Planning and Reporting	7	18	21	5	2
ZUOZ		Atom library addon for ZUG	2	3	2	1	0

Example: To Copy a single release under a product



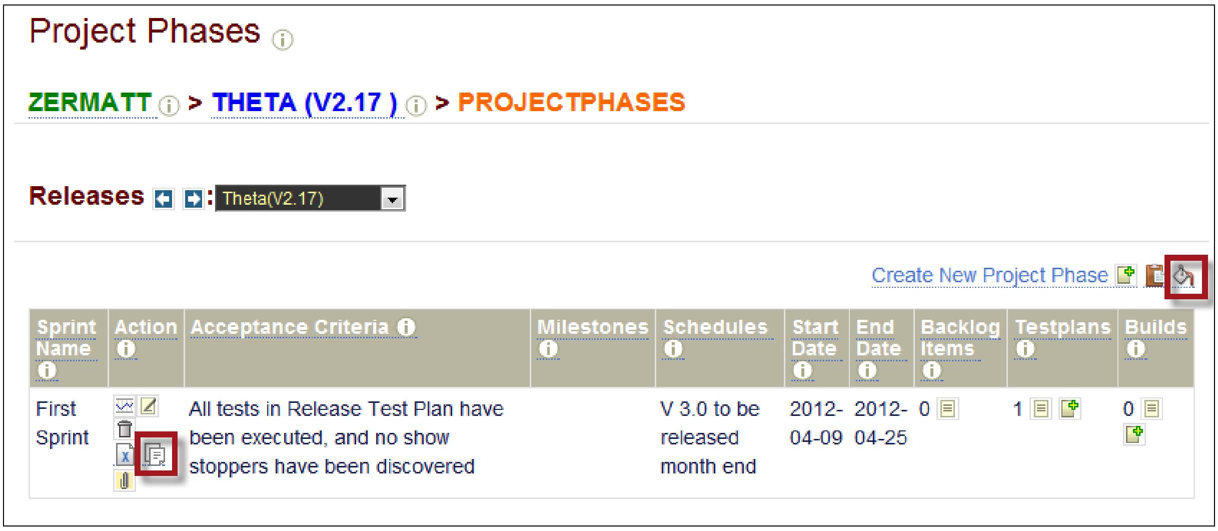
ZERMATT > RELEASES

Products: ZERMATT

Create New Release

Name	Action(s)	Version	Description	Acceptance Criteria	Milestones	Schedules	Conceived Date	Release Date	Project Phases	Requirements	Backlog Items
Alpha		2.4	First release of this version	All tests in Release Test Plan have been executed, and no show stoppers have been discovered		Zermatt 3.0 to be launched month end.	2011-07-01	2012-08-05	4	3	0

Example: To Copy a single project phase under a release



Project Phases

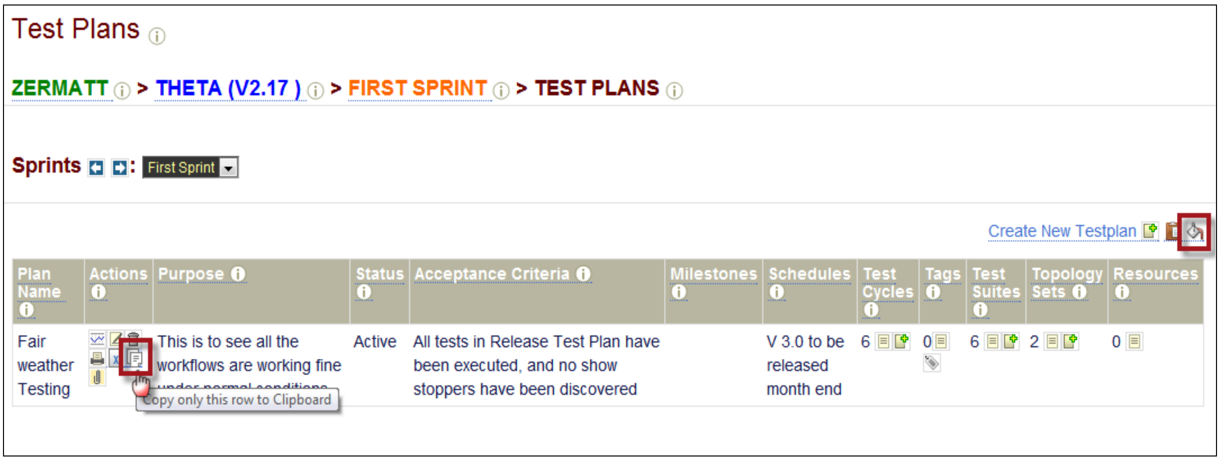
ZERMATT > THETA (V2.17) > PROJECTPHASES

Releases: Theta(V2.17)

Create New Project Phase

Sprint Name	Action	Acceptance Criteria	Milestones	Schedules	Start Date	End Date	Backlog Items	Testplans	Builds
First Sprint		All tests in Release Test Plan have been executed, and no show stoppers have been discovered		V 3.0 to be released month end	2012-04-09	2012-04-25	0	1	0

Example: To Copy a single Test plan under a Project phase



Test Plans

ZERMATT > THETA (V2.17) > FIRST SPRINT > TEST PLANS

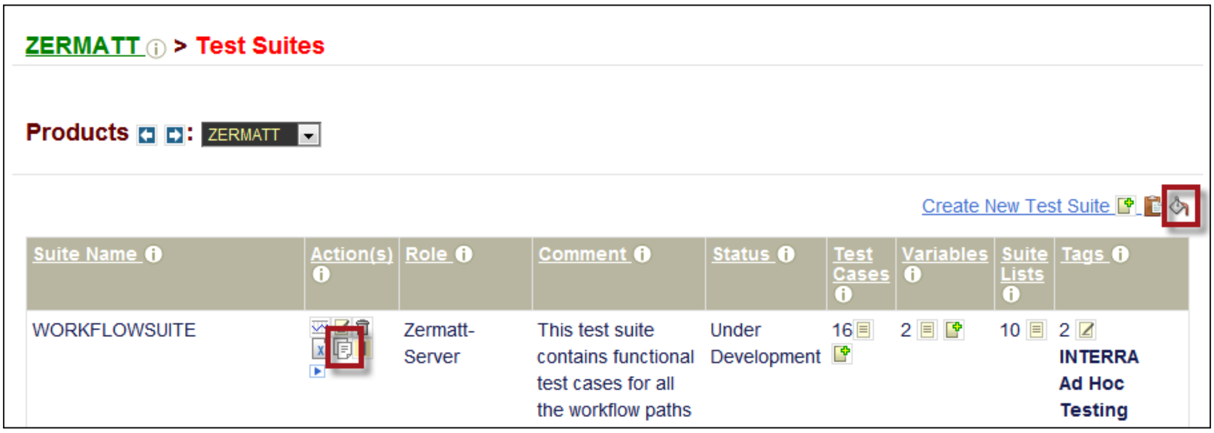
Sprints: First Sprint

Create New Testplan

Plan Name	Actions	Purpose	Status	Acceptance Criteria	Milestones	Schedules	Test Cycles	Tags	Test Suites	Topology Sets	Resources
Fair weather Testing		This is to see all the workflows are working fine under normal conditions	Active	All tests in Release Test Plan have been executed, and no show stoppers have been discovered		V 3.0 to be released month end	6	0	6	2	0

Copy only this row to Clipboard

Example: To Copy a single Test suite under a product



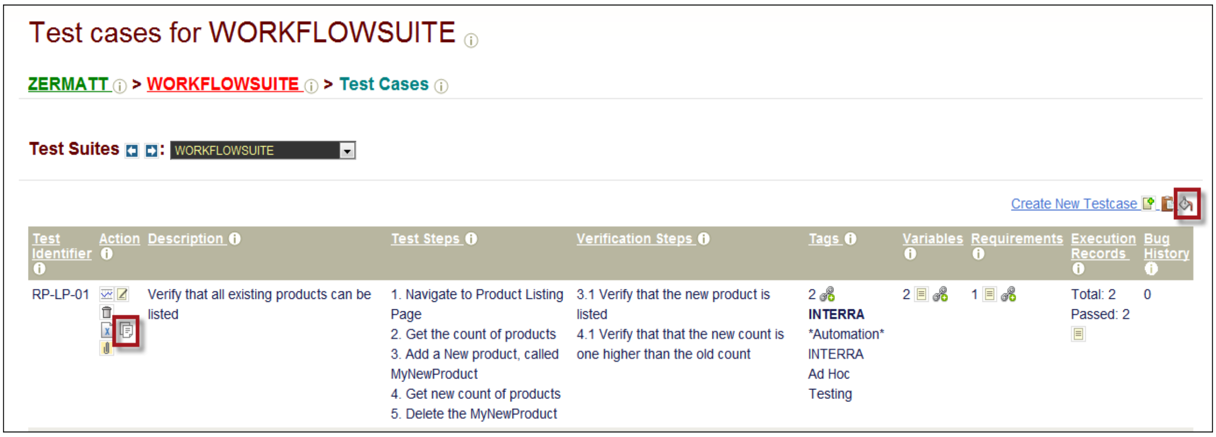
ZERMATT > Test Suites

Products: ZERMATT

Create New Test Suite

Suite Name	Action(s)	Role	Comment	Status	Test Cases	Variables	Suite Lists	Tags
WORKFLOWSUITE		Zermatt-Server	This test suite contains functional test cases for all the workflow paths	Under Development	16	2	10	2 **INTERRA Ad Hoc Testing**

Example: To Copy a single Test case under a Test suite



Test cases for WORKFLOWSUITE

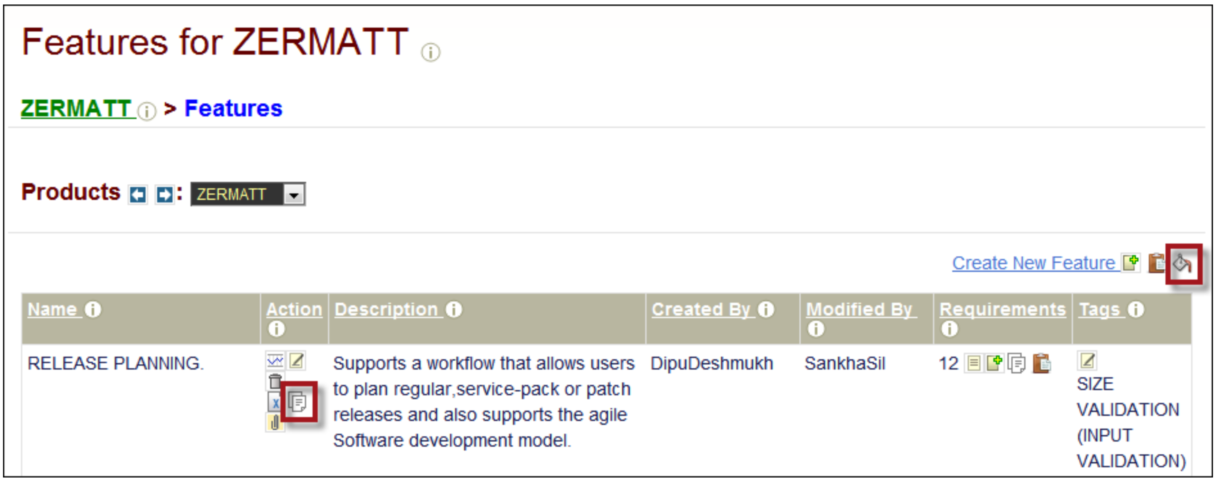
ZERMATT > WORKFLOWSUITE > Test Cases

Test Suites: WORKFLOWSUITE

Create New Testcase

Test Identifier	Action	Description	Test Steps	Verification Steps	Tags	Variables	Requirements	Execution Records	Bug History
RP-LP-01		Verify that all existing products can be listed	1. Navigate to Product Listing Page 2. Get the count of products 3. Add a New product, called MyNewProduct 4. Get new count of products 5. Delete the MyNewProduct	3.1 Verify that the new product is listed 4.1 Verify that that the new count is one higher than the old count	2 **INTERRA** *Automation* INTERRA Ad Hoc Testing	2	1	Total: 2 Passed: 2	0

Example: To Copy a single feature under a product



Features for ZERMATT

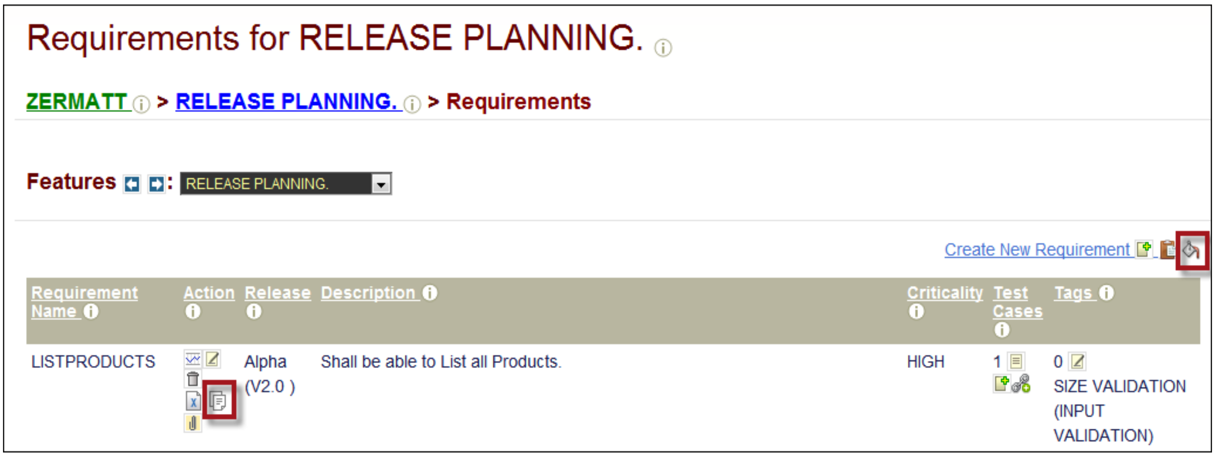
ZERMATT > Features

Products : ZERMATT

Create New Feature

Name	Action	Description	Created By	Modified By	Requirements	Tags
RELEASE PLANNING.		Supports a workflow that allows users to plan regular,service-pack or patch releases and also supports the agile Software development model.	DipuDeshmukh	SankhaSil	12	SIZE VALIDATION (INPUT VALIDATION)

Example: To Copy a single requirement under a feature



Requirements for RELEASE PLANNING.

ZERMATT > RELEASE PLANNING. > Requirements

Features : RELEASE PLANNING.

Create New Requirement

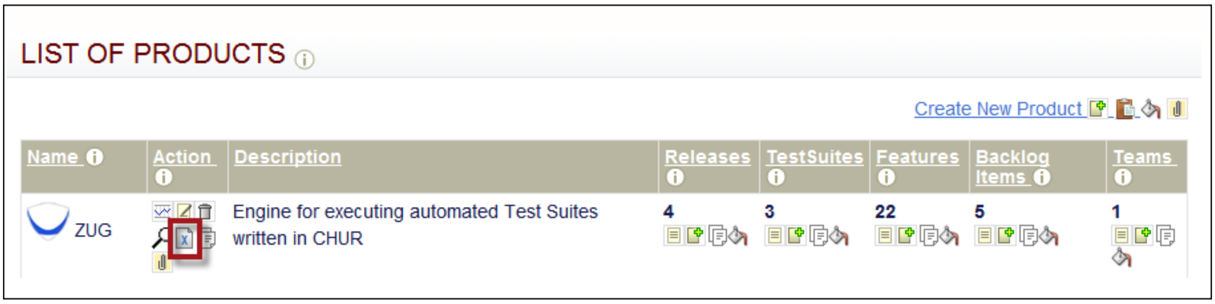
Requirement Name	Action	Release	Description	Criticality	Test Cases	Tags
LISTPRODUCTS		Alpha (V2.0)	Shall be able to List all Products.	HIGH	1	0 SIZE VALIDATION (INPUT VALIDATION)

5.3.10 Data Migration

Zermatt allows you to export data in xml format and import data from an external xml file. So, it becomes easy to migrate data from one Zermatt installation to another Zermatt installation.

Export Feature

To export an object (e.g. product, release, test suite, feature etc) from Zermatt click on the export link in the Action column of the Listing Table.

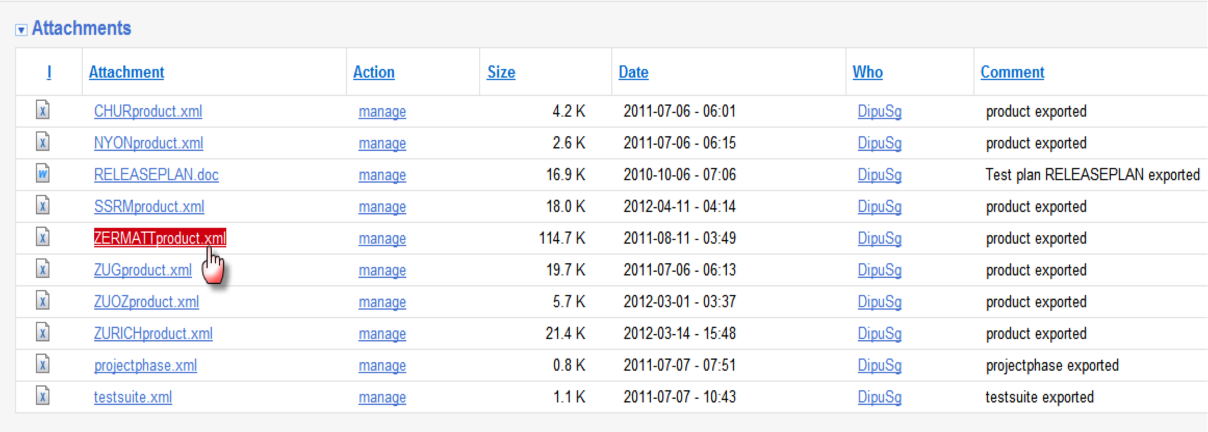


LIST OF PRODUCTS

Create New Product

Name	Action	Description	Releases	TestSuites	Features	Backlog Items	Teams
ZUG		Engine for executing automated Test Suites written in CHUR	4	3	22	5	1

The exported file can be viewed in the user home page.

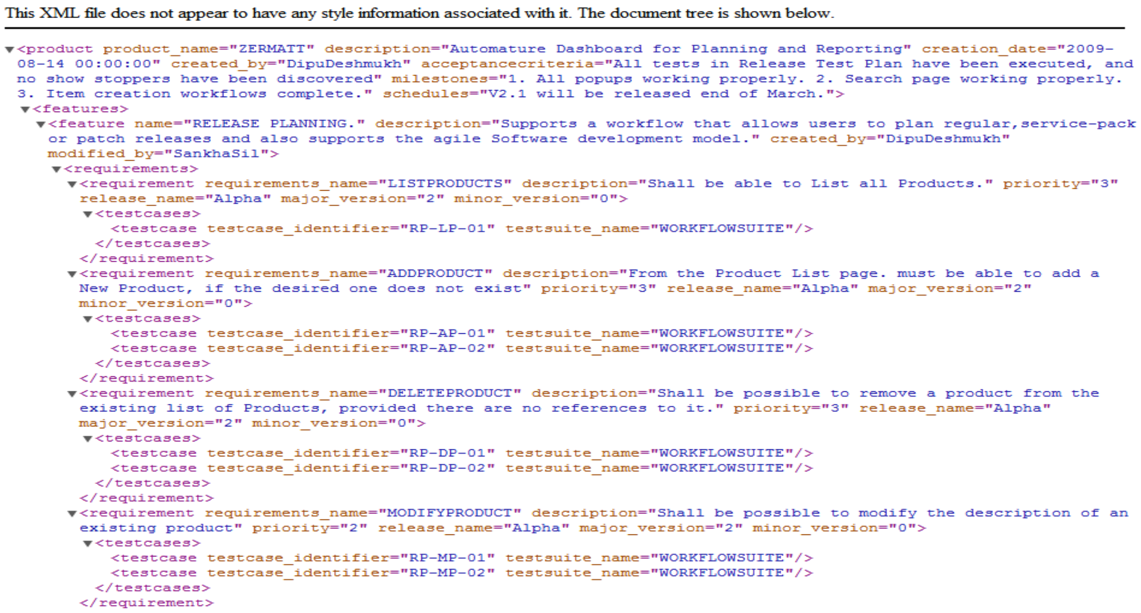


Attachments

I	Attachment	Action	Size	Date	Who	Comment
	CHURproduct.xml	manage	4.2 K	2011-07-06 - 06:01	DipuSg	product exported
	NYONproduct.xml	manage	2.6 K	2011-07-06 - 06:15	DipuSg	product exported
	RELEASEPLAN.doc	manage	16.9 K	2010-10-06 - 07:06	DipuSg	Test plan RELEASEPLAN exported
	SSRMproduct.xml	manage	18.0 K	2012-04-11 - 04:14	DipuSg	product exported
	ZERMATTproduct.xml	manage	114.7 K	2011-08-11 - 03:49	DipuSg	product exported
	ZUGproduct.xml	manage	19.7 K	2011-07-06 - 06:13	DipuSg	product exported
	ZUOZproduct.xml	manage	5.7 K	2012-03-01 - 03:37	DipuSg	product exported
	ZURICHproduct.xml	manage	21.4 K	2012-03-14 - 15:48	DipuSg	product exported
	projectphase.xml	manage	0.8 K	2011-07-07 - 07:51	DipuSg	projectphase exported
	testsuite.xml	manage	1.1 K	2011-07-07 - 10:43	DipuSg	testsuite exported

0After clicking the link to the attachment, the browser opens up the XML file

Now you can save the XML file on your computer.

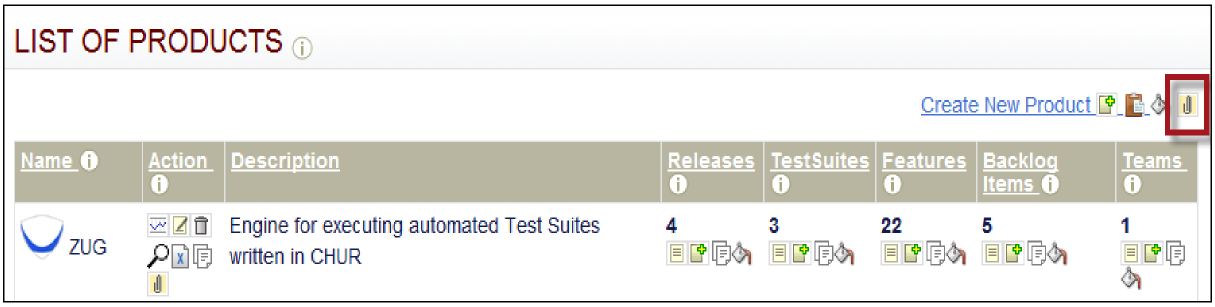


This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<product product_name="ZERMATT" description="Automature Dashboard for Planning and Reporting" creation_date="2009-
 08-14 00:00:00" created_by="DipuDeshmukh" acceptancecriteria="All tests in Release Test Plan have been executed, and
 no show stoppers have been discovered" milestones="1. All popups working properly. 2. Search page working properly.
 3. Item creation workflows complete." schedules="V2.1 will be released end of March.">
  ▼<features>
    ▼<feature name="RELEASE PLANNING." description="Supports a workflow that allows users to plan regular,service-pack
     or patch releases and also supports the agile Software development model." created_by="DipuDeshmukh"
     modified_by="SankhaSil">
      ▼<requirements>
        ▼<requirement requirements_name="LISTPRODUCTS" description="Shall be able to List all Products." priority="3"
         release_name="Alpha" major_version="2" minor_version="0">
          ▼<testcases>
             <testcase testcase_identifier="RP-LP-01" testsuite_name="WORKFLOWSUITE"/>
           </testcases>
         </requirement>
        ▼<requirement requirements_name="ADDPRODUCT" description="From the Product List page. must be able to add a
         New Product, if the desired one does not exist" priority="3" release_name="Alpha" major_version="2"
         minor_version="0">
          ▼<testcases>
             <testcase testcase_identifier="RP-AP-01" testsuite_name="WORKFLOWSUITE"/>
             <testcase testcase_identifier="RP-AP-02" testsuite_name="WORKFLOWSUITE"/>
           </testcases>
         </requirement>
        ▼<requirement requirements_name="DELETEPRODUCT" description="Shall be possible to remove a product from the
         existing list of Products, provided there are no references to it." priority="3" release_name="Alpha"
         major_version="2" minor_version="0">
          ▼<testcases>
             <testcase testcase_identifier="RP-DP-01" testsuite_name="WORKFLOWSUITE"/>
             <testcase testcase_identifier="RP-DP-02" testsuite_name="WORKFLOWSUITE"/>
           </testcases>
         </requirement>
        ▼<requirement requirements_name="MODIFYPRODUCT" description="Shall be possible to modify the description of an
         existing product" priority="2" release_name="Alpha" major_version="2" minor_version="0">
          ▼<testcases>
             <testcase testcase_identifier="RP-MP-01" testsuite_name="WORKFLOWSUITE"/>
             <testcase testcase_identifier="RP-MP-02" testsuite_name="WORKFLOWSUITE"/>
           </testcases>
         </requirement>
```

Import Feature

To import an object (e.g. product) into Zermatt, click on the import link on the top right hand side of the table.

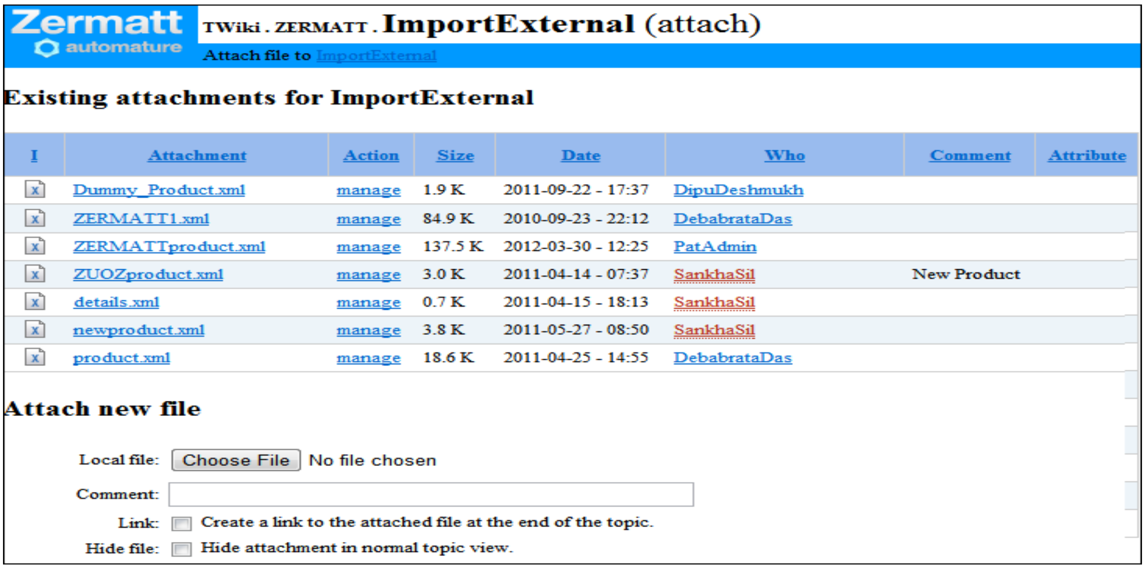


LIST OF PRODUCTS

Create New Product

Name	Action	Description	Releases	TestSuites	Features	Backlog Items	Teams
ZUG		Engine for executing automated Test Suites written in CHUR	4	3	22	5	1

Click on the browse icon and select the file that needs to be imported into Zermatt and then upload the file.



Zermatt automature TWiki . ZERMATT . ImportExternal (attach)

Attach file to ImportExternal

Existing attachments for ImportExternal

I	Attachment	Action	Size	Date	Who	Comment	Attribute
	Dummy_Product.xml	manage	1.9 K	2011-09-22 - 17:37	DipuDeshmukh		
	ZERMATT1.xml	manage	84.9 K	2010-09-23 - 22:12	DebabrataDas		
	ZERMATTproduct.xml	manage	137.5 K	2012-03-30 - 12:25	PatAdmin		
	ZUOZproduct.xml	manage	3.0 K	2011-04-14 - 07:37	SankhaSil	New Product	
	details.xml	manage	0.7 K	2011-04-15 - 18:13	SankhaSil		
	newproduct.xml	manage	3.8 K	2011-05-27 - 08:50	SankhaSil		
	product.xml	manage	18.6 K	2011-04-25 - 14:55	DebabrataDas		

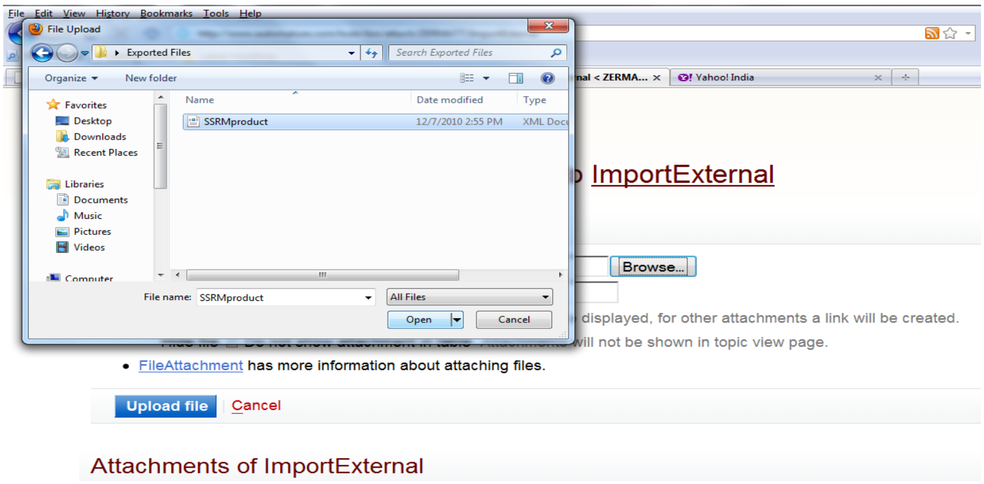
Attach new file

Local file: Choose File No file chosen

Comment:

Link: □ Create a link to the attached file at the end of the topic.

Hide file: □ Hide attachment in normal topic view.



SSRMproduct 12/7/2010 2:55 PM XML Docu

File name: SSRMproduct

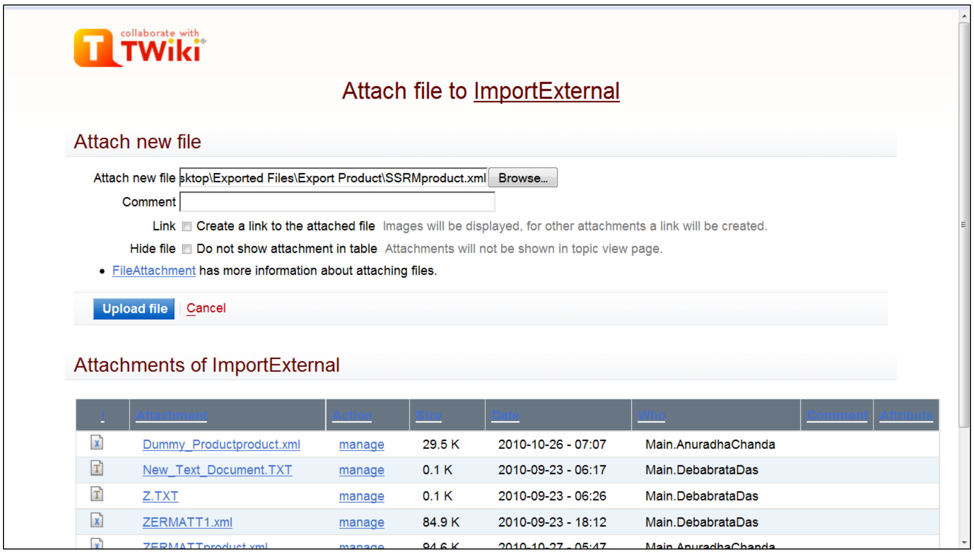
ImportExternal

displayed, for other attachments a link will be created.

will not be shown in topic view page.

- FileAttachment has more information about attaching files.

Attachments of ImportExternal



Attach file to ImportExternal

Attach new file

Attach new file sktop\Exported Files\Export Product\SSRMproduct.xml Browse...

Comment

Link □ Create a link to the attached file Images will be displayed, for other attachments a link will be created.

Hide file □ Do not show attachment in table Attachments will not be shown in topic view page.

- FileAttachment has more information about attaching files.

Upload file Cancel

Attachments of ImportExternal

I	Attachment	Action	Size	Date	Who	Comment	Attribute
	Dummy_Productproduct.xml	manage	29.5 K	2010-10-26 - 07:07	Main.AnuradhaChanda		
	New_Text_Document.TXT	manage	0.1 K	2010-09-23 - 06:17	Main.DebabrataDas		
	Z.TXT	manage	0.1 K	2010-09-23 - 06:26	Main.DebabrataDas		
	ZERMATT1.xml	manage	84.9 K	2010-09-23 - 18:12	Main.DebabrataDas		
	ZERMATTproduct.xml	manage	94.6 K	2010-10-27 - 05:47	Main.AnuradhaChanda		

5.3.11 Manual Test Result Entry

Normally, test execution results are expected to be uploaded by ZUG, Automature's automation execution engine. However, not all tests are automated, and some ad hoc tests may also be performed manually. This workflow facilitates recording of results from tests that were executed manually.

List All Products

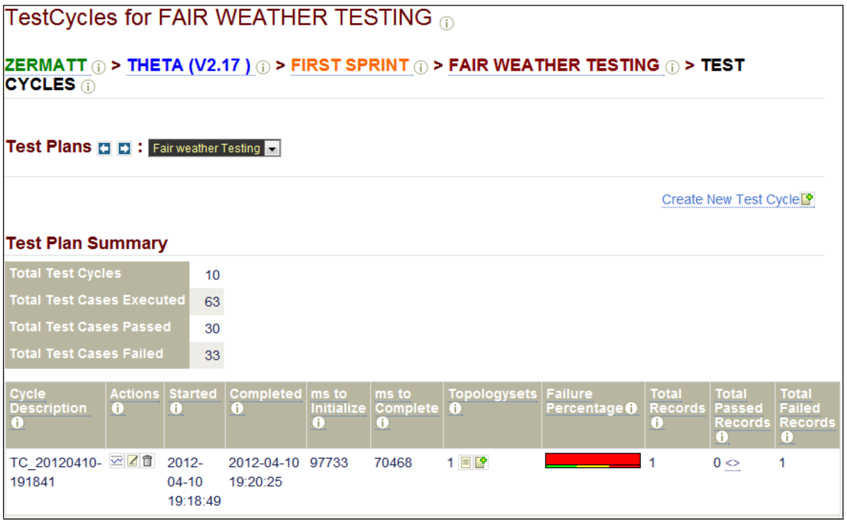
List all Releases for a chosen product

List all Sprints or Project Phases for a chosen release

List all Test Plans for a chosen Sprint or Project Phase (Normally there will be at most one such Test Plan)

List all Test Cycles for the Test Plan

- Create a New Test Cycle for the Test Plan



TestCycles for FAIR WEATHER TESTING

ZERMATT > THETA (V2.17) > FIRST SPRINT > FAIR WEATHER TESTING > TEST CYCLES

Test Plans : Fair weather Testing

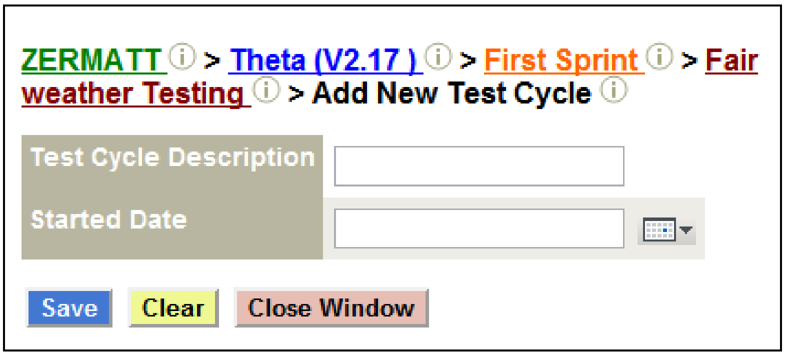
Create New Test Cycle

Test Plan Summary

Total Test Cycles	10
Total Test Cases Executed	63
Total Test Cases Passed	30
Total Test Cases Failed	33

Cycle Description	Actions	Started	Completed	ms to Initialize	ms to Complete	Topologysets	Failure Percentage	Total Records	Total Passed Records	Total Failed Records
TC_20120410-191841		2012-04-10 19:18:49	2012-04-10 19:20:25	97733	70468	1	🟥	1	0 <>	1

- Remove an existing Test Cycle from the Test Plan 🗑

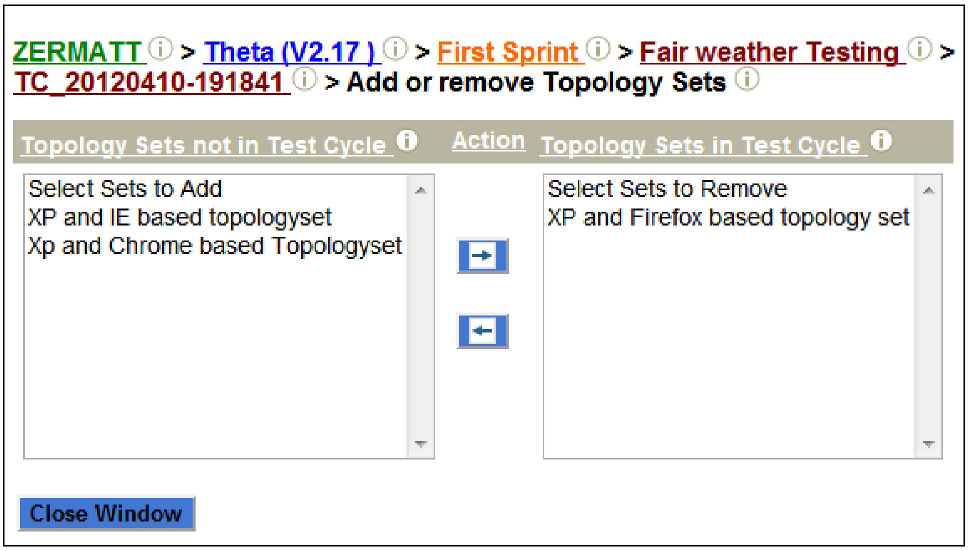


ZERMATT ⓘ > Theta (V2.17) ⓘ > First Sprint ⓘ > Fair weather Testing ⓘ > Add New Test Cycle ⓘ

Test Cycle Description	
Started Date	

List all Test Cycles for the Test Plan

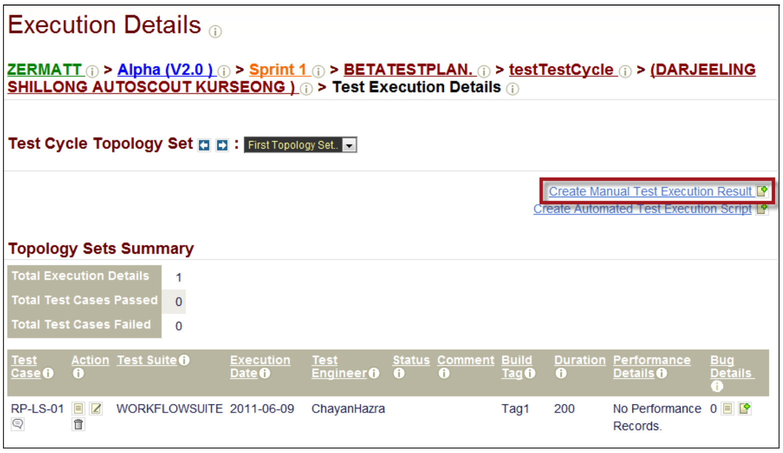
- Add a Topology set to the test Cycle
- Remove an existing Topology Set



ZERMATT ⓘ > Theta (V2.17) ⓘ > First Sprint ⓘ > Fair weather Testing ⓘ > TC_20120410-191841 ⓘ > Add or remove Topology Sets ⓘ

Topology Sets not in Test Cycle ⓘ	Action	Topology Sets in Test Cycle ⓘ
Select Sets to Add XP and IE based topologyset Xp and Chrome based Topologyset	→ ←	Select Sets to Remove XP and Firefox based topology set

- Create an Execution Detail Record for the chosen Test Cycle



Execution Details ⓘ

ZERMATT ⓘ > Alpha (V2.0) ⓘ > Sprint 1 ⓘ > BETATESTPLAN. ⓘ > testTestCycle ⓘ > (DARJEELING SHILLONG AUTOSCOUT KURSEONG) ⓘ > Test Execution Details ⓘ

Test Cycle Topology Set : First Topology Set..

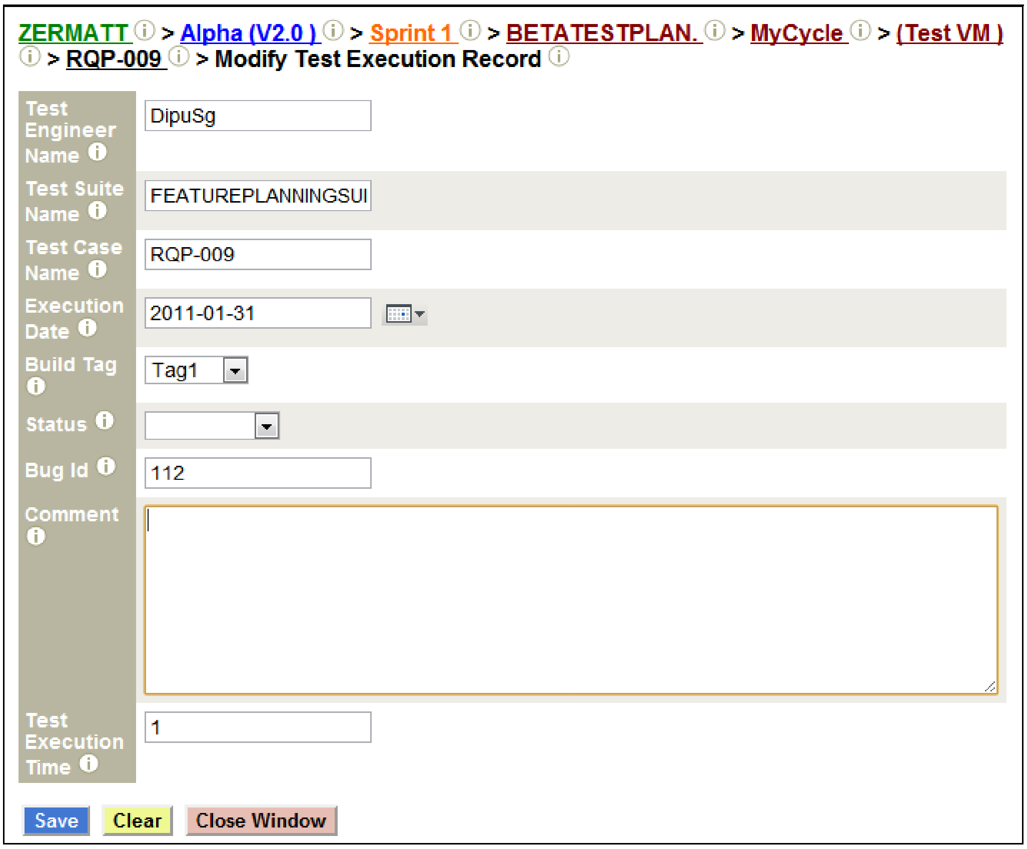
Create Manual Test Execution Result
Create Automated Test Execution Script

Topology Sets Summary

Total Execution Details	1
Total Test Cases Passed	0
Total Test Cases Failed	0

Test Case ⓘ	Action ⓘ	Test Suite ⓘ	Execution Date ⓘ	Test Engineer ⓘ	Status ⓘ	Comment ⓘ	Build Tag ⓘ	Duration ⓘ	Performance Details ⓘ	Bug Details ⓘ
RP-LS-01		WORKFLOWSUITE	2011-06-09	ChayanHazra			Tag1	200	No Performance Records.	0

- Modify an Execution Detail Record for the chosen Test Cycle 🗑



ZERMATT ⓘ > Alpha (V2.0) ⓘ > Sprint 1 ⓘ > BETATESTPLAN. ⓘ > MyCycle ⓘ > (Test VM) ⓘ > RQP-009 ⓘ > Modify Test Execution Record ⓘ

Test Engineer Name ⓘ	DipuSg
Test Suite Name ⓘ	FEATUREPLANNINGSUI
Test Case Name ⓘ	RQP-009
Execution Date ⓘ	2011-01-31
Build Tag ⓘ	Tag1
Status ⓘ	
Bug Id ⓘ	112
Comment ⓘ	
Test Execution Time ⓘ	1

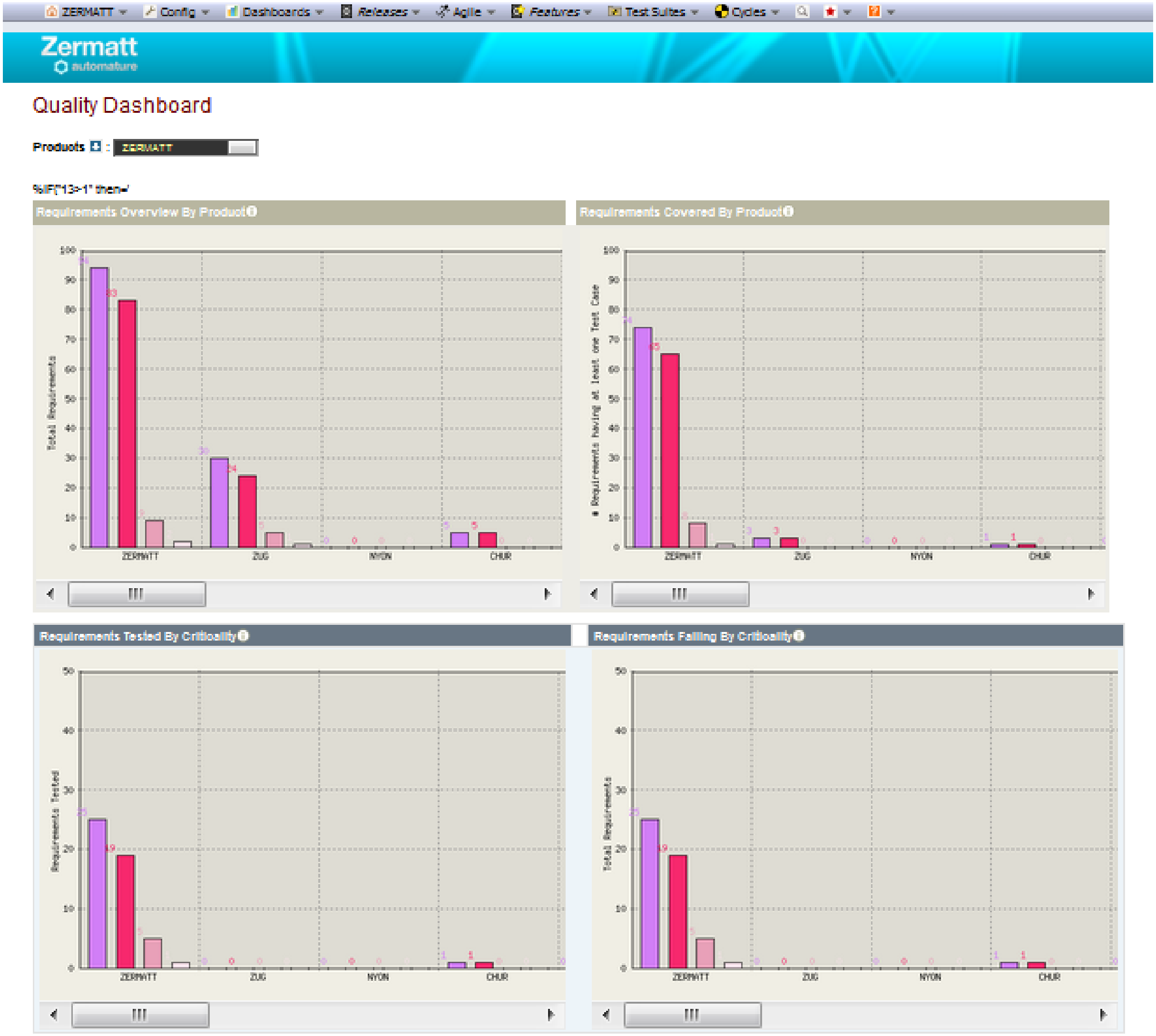
- Remove an Execution Detail Record from the chosen Test Cycle ✎

5.4 Reporting tasks

5.4.1 Total Quality Dashboard

The Total Quality Dashboard represents all the products with

- Requirements Overview by Product
- Requirements Coverage by Product
- Requirements Tested by Criticality
- Requirements Failing by Criticality
- Bugs Outstanding by Product



Quality Dashboard

Products : ZERMATT

%IF("13>1" then='

Requirements Overview By Product

Requirements Covered By Product

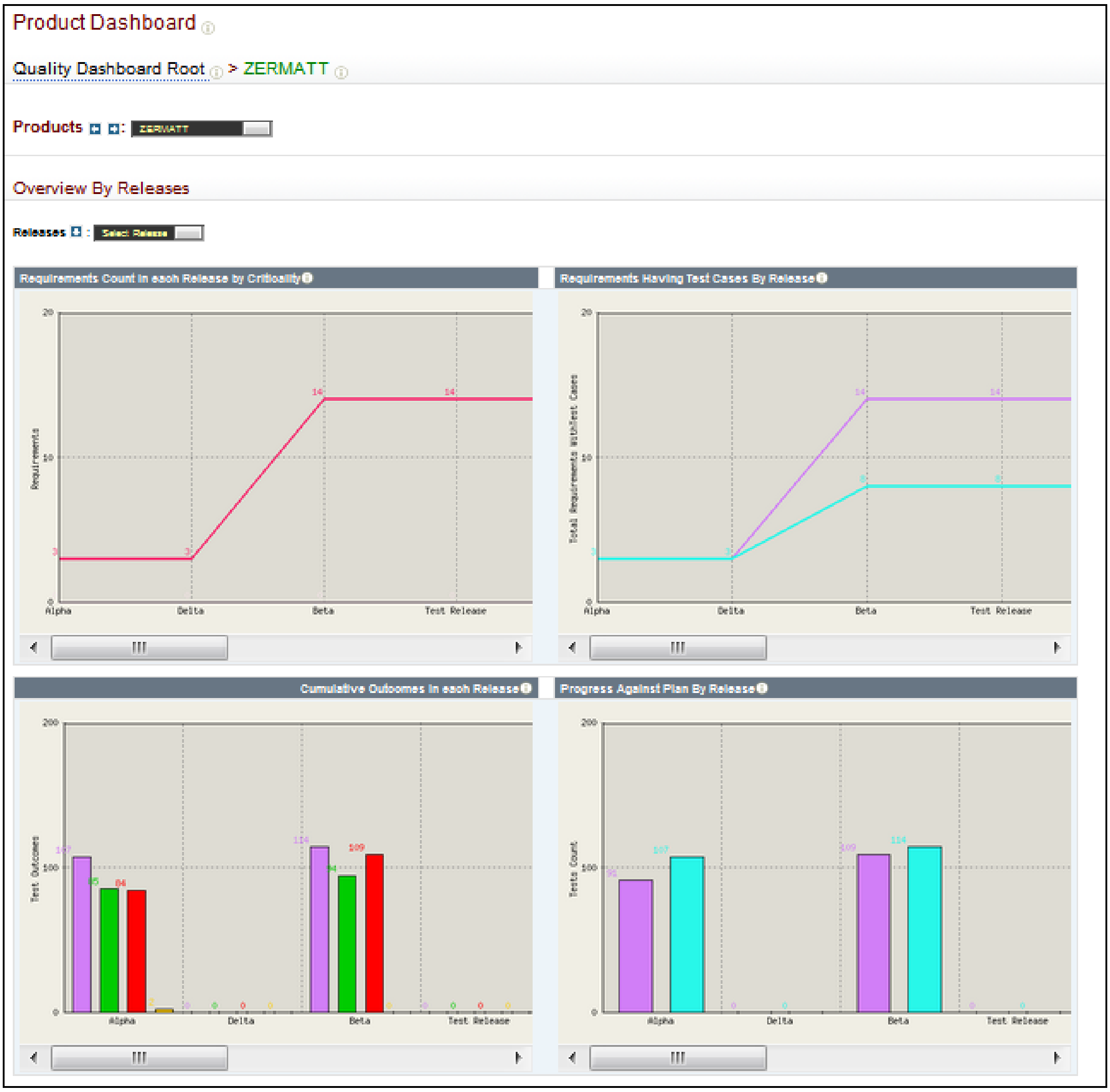
Requirements Tested By Criticality

Requirements Failing By Criticality

5.4.2 Product Quality Dashboard

The Product Quality dashboard represents an individual Product with

- Requirements having Test Cases by Release
- Cumulative outcomes in each Release
- Requirement count by Criticality
- Progress against Plan by Release
- Critical Requirements tested in each Release
- Requirements Overview by Test Suites
- Outcome of tests in each Test Suite
- Total Variables Declared and Tags Used By Test Suites
- Test Case Count by Test Suites



Product Dashboard

Quality Dashboard Root > ZERMATT

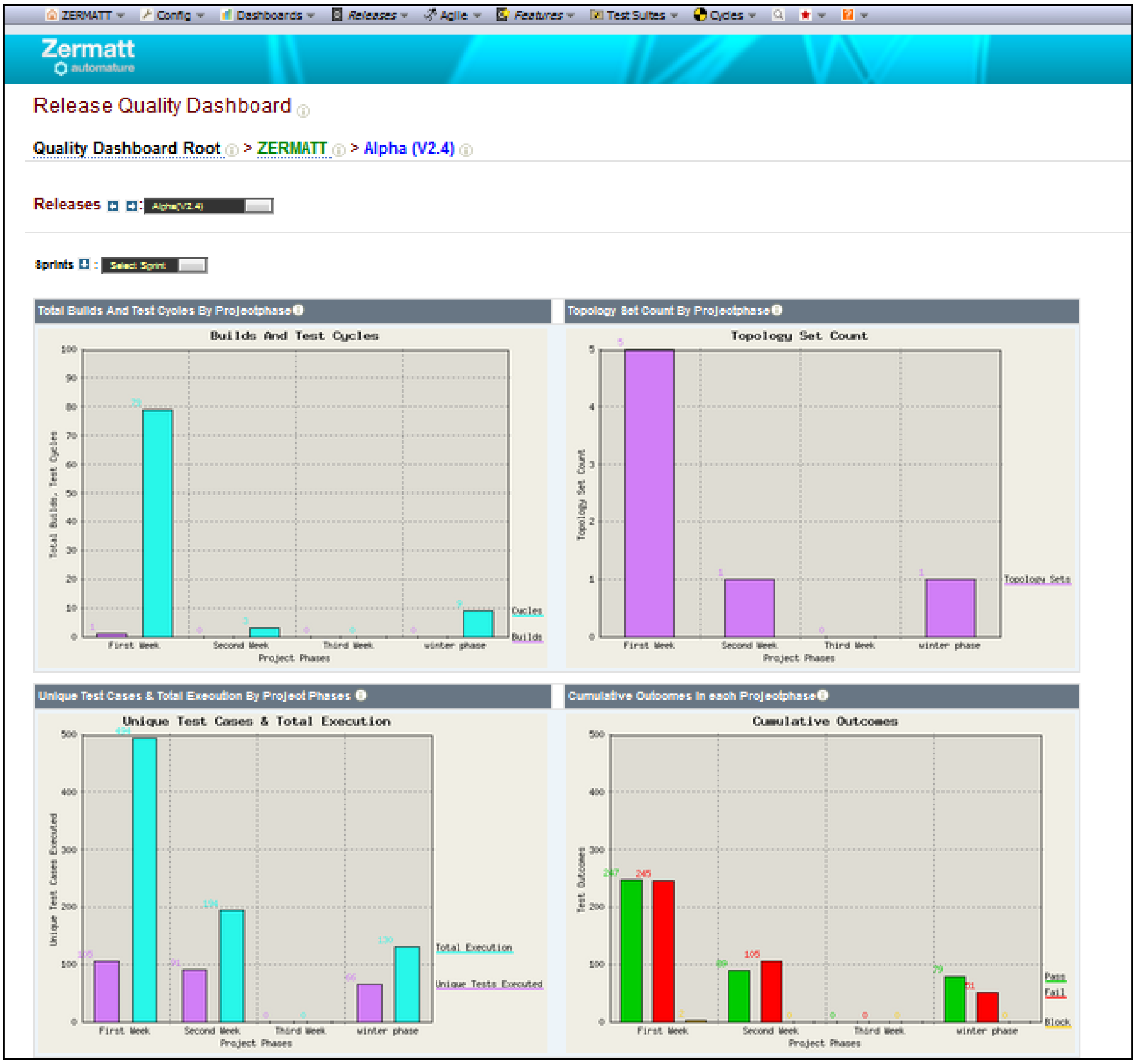
Products: ZERMATT

Overview By Releases

Releases : Select Release

5.4.3 Release Quality Dashboard

- The Release Quality Dashboard Represents an individual Release with
- Total Builds and Test Cycles by Project Phases
- Topology set Count by Project Phase
- Unique Test Cases & Total Execution in Each Release
- Cumulative Outcome in each Project Phase



Release Quality Dashboard

Quality Dashboard Root > ZERMATT > Alpha (V2.4)

Releases : Alpha(V2.4)

Sprints : Select Sprint

Total Builds And Test Cycles By Projectphase

Topology Set Count By Projectphase

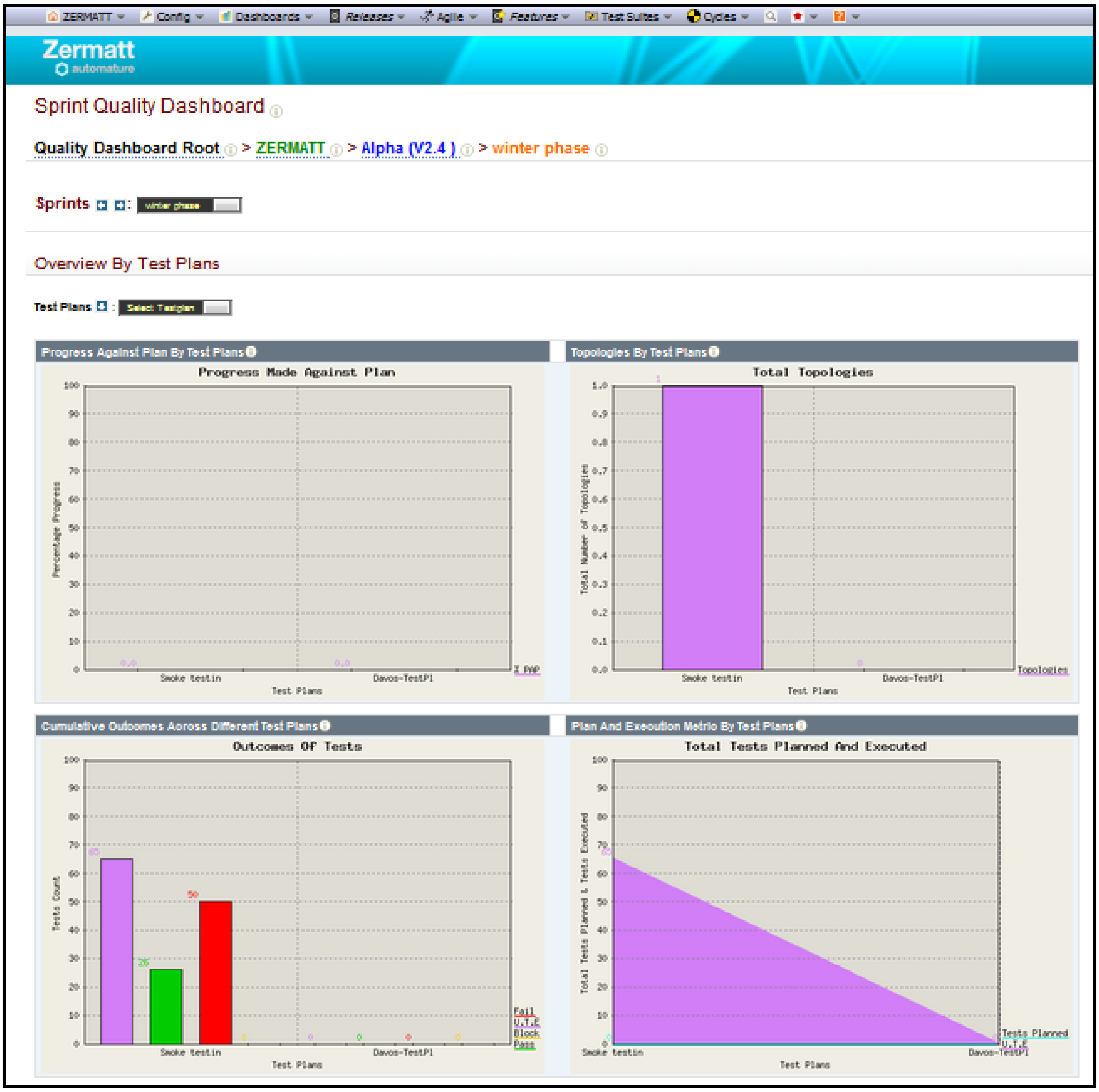
Unique Test Cases & Total Execution By Project Phases

Cumulative Outcomes In each Projectphase

5.4.4 Sprint Quality Dashboard

The Sprint Quality Dashboard represents the individual project phases with

- Sprint overview by Test Plans
- Topologies by Test Plans
- Outcome of Test by Test Plans
- Plan and Execution Metric by test Plans
- Progress against plan by Test Plans
- Sprint Overview by Build
- Total Topologies by Builds
- Outcome of Test by Builds



Sprint Quality Dashboard

Quality Dashboard Root > ZERMATT > Alpha (V2.4) > winter phase

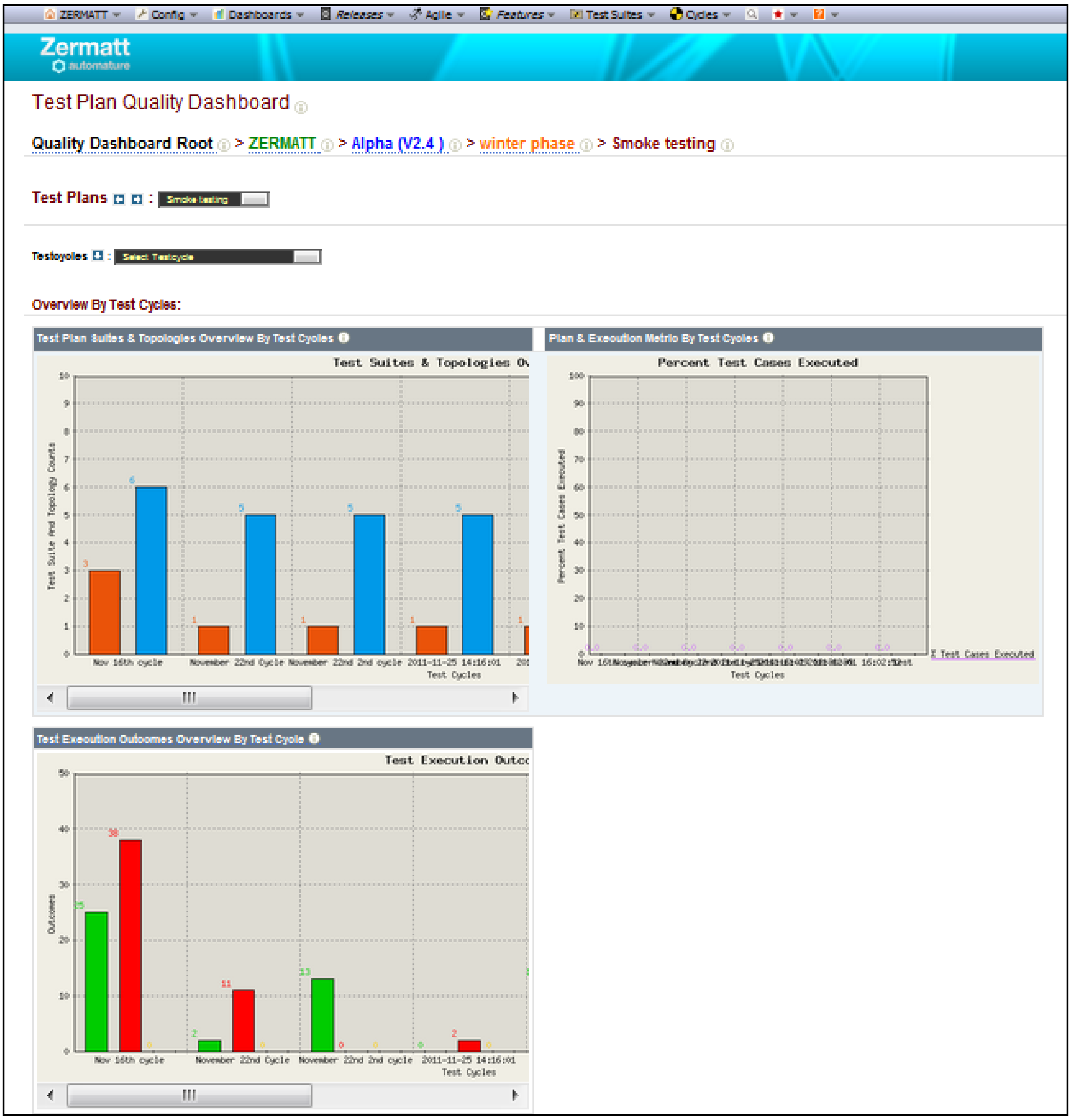
Sprints: winter phase

Overview By Test Plans

Test Plans : Select Testplan

5.4.5 Test Plan Report

- The Test Plan Quality Dashboard represents a Test Suite with
- Test Plan Suites & Topologies Overview by test Cycles
- Plan & Execution Metric by Test Cycles
- Test Execution Outcomes Overview by Test Cycles



Test Plan Quality Dashboard

Quality Dashboard Root > ZERMATT > Alpha (V2.4) > winter phase > Smoke testing

Test Plans : Smoke testing

Testcycles : Select Testcycle

Overview By Test Cycles:

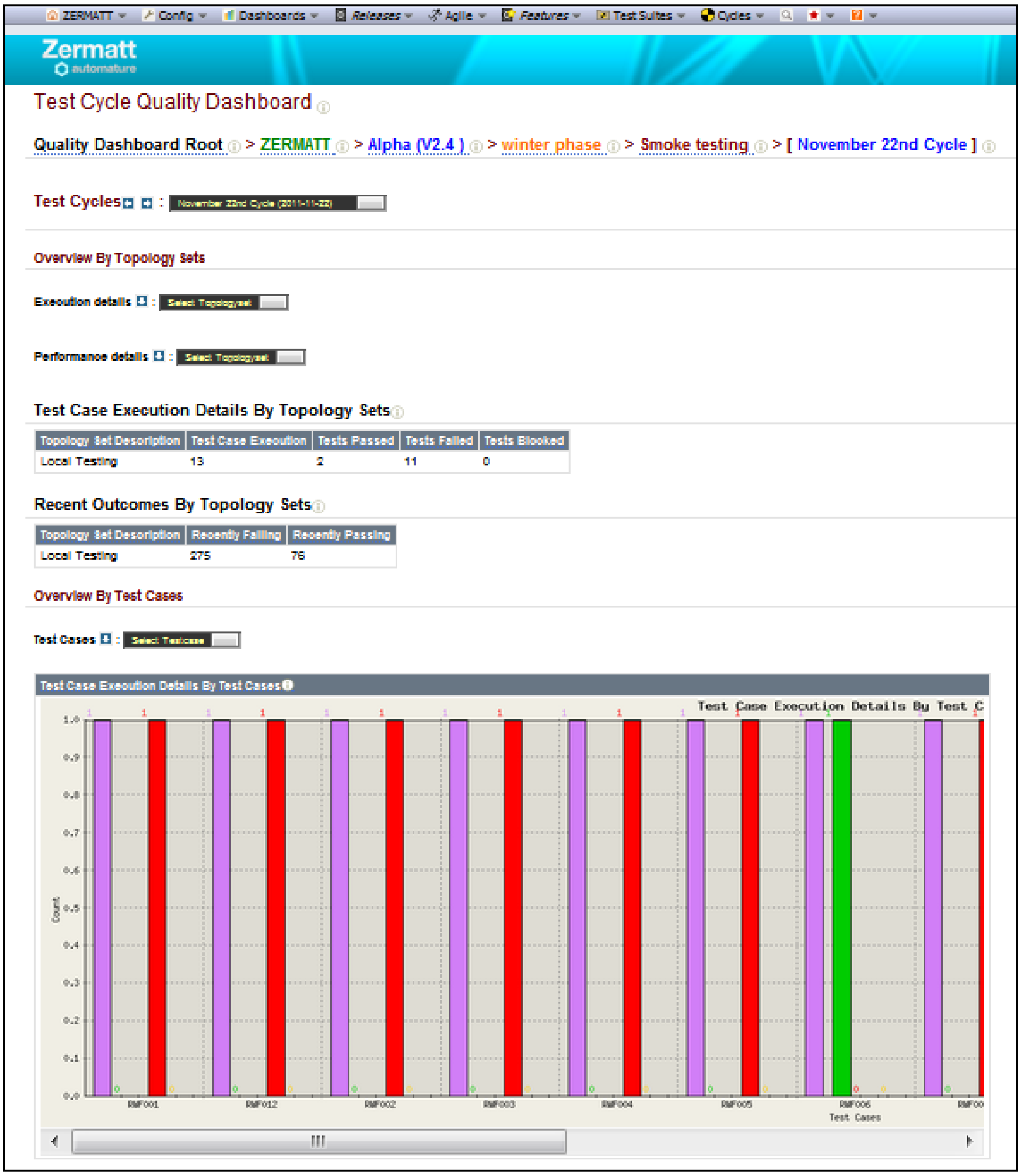
Test Plan Suites & Topologies Overview By Test Cycles

Plan & Execution Metric By Test Cycles

Test Execution Outcomes Overview By Test Cycle

5.4.6 Test Cycle Report

- The Test Cycle Quality dashboard represents a Test Cycle with
- Recent Outcomes by Topology Sets
- Test Case execution Details By Test Cases
- Test Case Outcomes in Topologies



Test Cycle Quality Dashboard

Quality Dashboard Root > ZERMATT > Alpha (V2.4) > winter phase > Smoke testing > [November 22nd Cycle]

Test Cycles : November 22nd Cycle (2011-11-22)

Overview By Topology Sets

Execution details : Select Topologyset

Performance details : Select Topologyset

Test Case Execution Details By Topology Sets

Topology Set Description	Test Case Execution	Tests Passed	Tests Failed	Tests Blocked
Local Testing	13	2	11	0

Recent Outcomes By Topology Sets

Topology Set Description	Recently Failing	Recently Passing
Local Testing	275	76

Overview By Test Cases

Test Cases : Select Testcase

Test Case Execution Details By Test Cases