DevSpec User Guide

Author:

Table of Content

DevSpec User Guide

Key Benefits

Define product or project requirements.
Track what requirements are not covered by a development work items or test cases.
Identify which features or functions in the design are not required internally or externally.
Control requirement changes and view the impact of the implementation of such changes.
Provide lifecycle traceability and review requirement implementation and validation

Features Overview

Create, manage, discuss and link project requirements and features.
Fully supports Scrum, Iterative development, Waterfall, and many other methodologies
Automatic requirement versioning whenever specified changes are made combined with approval tracking and a robust workflow engine.
Requirements, specification and other key digital assets are stored in a reliable and secure central data repository (with support for storage in external SCM systems from Perforce and Subversion.)
Out of the box integrations with DevTrack and DevTest allowing project managers to view the complete development lifecycle for all project requirements.
Poll stake holders on requirement and feature value, risk or other user-defined criteria.
Create "what if" scenarios for product feature sets and time estimates using DevSpec "Options".
Define requirement and feature interfaces you want with extensive customization options including user-defined field labels, field types, drop-down menu options, master-detail relationships, and custom reports.
Integration with Microsoft Office and Adobe PDF
Built-in presentation quality reports enable you to easily report on all requirement and feature data, change control and change impact as well as implementation and testing data for requirements and features.

Chapter 1 - Requirement Management with DevSpec

This chapter serves as an introduction to the design goals of DevSpec.

1 TechExcel DevSpec

DevSpec is a software tool for requirements management. It is built around modern collaboration technologies. Managing requirements with DevSpec is easy and fun with a WYSIWYG wiki editing tool, coupled with the full power and flexibility of workflow controls, process automation tools, a definable GUI, and built in support of popular development methodologies including Agile, iterative, and traditional waterfall methods.

Before explaining too much about what DevSpec is, it would be best to define the two terms used most in this guide.

Requirements- A requirement represents a description of some feature or set of features. Requirements can take many forms, and these differ based on your processes and your team. We will cover the various ways to use requirements later in the document.

Specifications- A specification details work to be done. This is the easiest way to understand specs in DevSuite. A specification should be detailed enough so that the person responsible for implementing it knows what their role should be. In most cases, specifications should cover 1-5 days of work.

DevSpec is not only a tool for gathering and refining ideas, but also for defining the repeatable processes that ensure that good ideas are consistently realized and delivered in the final product. DevSpec enables businesses to define and manage a process for transforming good ideas into requirements and specifications, so that those ideas may be fully realized and delivered to the customer.

Requirements, specifications, and other key digital assets are stored in a reliable, scalable, and secure central data repository. In addition, requirements and specifications may be linked to development work items and test cases, so that all project stakeholders -- management, development, and QA -- may collaborate and track changing requirements and specifications throughout the project lifecycle.

The system also enables project team members to organize, prioritize, and monitor project requirements at every stage in the project lifecycle, manage multiple versions of requirements, track requirement changes, and ensure the accurate implementation of specifications. DevSpec facilitates access to information and collaboration between internal and external stakeholders, so that good ideas are expressed in thorough requirements and well-defined specifications, so that the end product realizes the initial vision.

2 DevSpec as a Part of DevSuite

TechExcel DevSuite is a family of integrated application lifecycle management (ALM) tools that place knowledge management - from ideas, to formal specifications, to competitive information, to issue resolution and customer insight - at the core of any product development initiative.

The DevSuite knowledge-centric strategy enables improved communication, ensures users are up-to-date on changes, and reduces the development cycles, so that businesses may deliver the right products for the right markets in the shortest possible time.

DevSuite places knowledge management at the core of all development processes. TechExcel KnowledgeWise provides for the easy and efficient collection and organization of informal ideas, gathered from a wide variety of sources, that area shared across multiple DevSpec, DevTrack, and DevTest projects.

KnowledgeWise projects provide controlled access to documents, improve communication and coordination between distributed development teams, and facilitate the management and sharing information between development teams and project stakeholders.

TechExcel DevSuite leverages intellectual assets with KnowledgeWise, communicating a clear product vision and tactical execution strategy by linking ideas and customer feedback, specifications, requirements, designs, prototypes, and other documents to specific areas of work.

3 TechExcel ALM Solutions

TechExcel DevSuite features five ALM solutions that operate in an n-tier architecture: TechExcel KnowledgeWise, TechExcel

DevTrack, TechExcel DevPlan, TechExcel DevSpec, and TechExcel DevTest.

All DevSuite products share the same core architecture - the DevSuite Database Server, DevSuite Application Server, DevSuite Document Server, and DevSuite Web Services - and are fully integrated enabling every branch of the development organization to communicate and work together.

KnowledgeWise

TechExcel DevSuite leverages intellectual assets with KnowledgeWise, communicating a clear product vision and tactical execution strategy by linking ideas and customer feedback, specifications, requirements, designs, prototypes, and other documents to specific areas of work during a development project. Documents are shared with all resources involved in the execution of the project allowing for an uncompromised vision to direct the path of any development project.

DevPlan

TechExcel DevPlan manages the transformation of concepts into formal strategic plans. DevPlan offers an intuitive planning hierarchy to formalize scope and optimize resource usage, team-based planning and calendaring capabilities. These features enable complete control over all product development projects from design planning to implementation and enables increased team efficiency and collaboration.

DevSpec

TechExcel DevSpec is an integrated requirements management solution that is specifically designed to provide visibility, traceability and validation of your product or project requirements. DevSpec provides a framework to create new requirements, specifications and features that can be linked to development and testing implementation projects.

DevTrack

TechExcel DevTrack enables development teams to manage every aspect of the development process including issue management, team management, and communications management.

DevTest

TechExcel DevTest is a test management solution that enables test organizations to manage every stage of testing lifecycle - from test case design, to test execution, to test analysis. DevTest provides testing groups with the tools they need work more effectively and efficiently, hold down costs, and to deliver higher quality products.

While each element is valuable even when functioning independently, the DevSuite architecture is founded upon the assumption of future scaling and is capable of advanced inter-module integration. Whether used as separate components or as an incorporated set of applications, the DevSuite solutions are ideal for companies at any stage in their product development.

4 Administration of DevSpec

DevSpec is customized and configured using the DevSuite Admin. This tool is used to configure various aspects of the system.

4.1 Workflow in DevSpec

TechExcel DevSuite solutions enable development organizations to define custom workflow rules for managing every aspect of the ALM life cycle. DevSpec requirement workflow defines how requirements are created, managed, and tracked in a DevSpec project.

Administrator-defined workflow rules determine the sequence of workflow states, how and when a requirement may pass from one workflow state to the next, and who may own and edit a requirement in each stage of its life cycle.

A typical requirement workflow might consist of eight workflow states. Each state is either an active or inactive state: {New}, Proposed, Approved, Delayed, Dropped, Implemented, In Development, and Change Requested.

In DevSpec, every workflow state is defined as either an active or inactive workflow state based on its status - active or inactive. The status assigned to a workflow state determines how the specification is managed in that workflow state.

The requirement life cycle consists of at least two workflow states: an active workflow state and an inactive workflow state. By default, all new requirements start in the New workflow state, an active state.

4.2 GUI Customization

DevSpec project administration is largely the process of defining structures for recording, managing, and tracking work items - requirements, specifications, change requests - in a project.

Customized project management structures are manifested in the DevSpec client as views, panels, folders, and pages.

A view is an interface that displays and organizes data in the client workspace. DevSpec supports three customizable views: the requirements views, the specifications view, and the change request view.

Views enable project members to manage and track different types of data using structures that are specifically designed to manage that data. The building blocks of each view - folders, pages, and controls - may be customized to support custom business processes.

The building blocks of client interface include: controls, folder pages, and GUI pages.

Control: A control is a GUI element displayed in a folder page or custom page that enables the user to define, update, and track data. DevSpec supports system controls and custom controls. Controls are displayed in system pages and custom pages.

System Page: A system page is a predefined form that enables project members to collect, manage, and track project data in a project. A system page displays a predefined set of system controls.

Custom Page: A custom page is a form conceived, designed, and built by a project administrator that enables project members to collect, manage, and track project data. A custom page may display system controls and custom controls.

Function Page: A function page is a compound page composed of multiple custom pages and system pages. In the DevSpec client, function pages are displayed as multiple-page form that enable the user to perform a particular function such as adding or updating a folder or a requirement.

4.3 Sample Projects and Templates

DevSuite builds on TechExcel's commitment to the rapid deployment of its ALM by providing development organizations with the tools they need to quickly configure and implement projects that enforce good development practices and drive the quality of releases.

To facilitate the configuration of a fully integrated site and projects that support your development processes, DevSuite provides develpers with tools for evaluating and testing DevSuite features.

Sample Projects: A sample project is a project that provides development organizations with a sandbox for configuring and evaluating DevSuite features and integrations in a risk free environment.

Project Templates: A project template is a blueprint for creating a project of a specific type.

TechExcel strongly recommends that development organizations deploy and test DevSuite in a test environment prior to your DevSuite implementation. A test environment enables development organizations to test project configurations and to train users in new processes prior to the "go live" date in a production environment.

4.3.1 Sample Projects

In DevSuite, a sample project is a project that provides the development organization with a sandbox for configuring and evaluating features and integrations in a risk free environment.

DevSuite features ten predefined projects that provide devleopment organizations with a sandbox to experiment with KnowledgeWise, DevSpec, and DevTrack tools and features. All sample project include sample data, sample project team members, and fully configured workflow rules. Project administrators may freely configure sample projects to represent their development processes.

Project configurations and settings tried and tested in sample projects may be imported into newly created "live projects". Sample projects provide development organizations with a tool for training new users or to introduce existing users to new features or changes in businesses processes.
Using controls in the DevSuite Admin client, administrators may create additional sample KnowledgeWise, DevSpec, and DevTrack projects.

4.3.2 Project Templates

In DevSuite, a project template is a blueprint for creating a project of a specific type - KnowledgeWise knowledge management projects, DevSpec requirements management projects, or DevTrack development projects.

Each project template defines a complete set of project-level settings including the definition of all project business objects, workflow rules, team representation, and project integrations.

Project templates may be used to quickly configure new projects whenever they are created in a DevSuite site.

5 Additional Help and Information

More information about DevSpec can be found by visitinghttp://www.techexcel.com/.

Chapter 2 - DevSpec Client Basics

This chapter provides an introduction on using the DevSpec client, including how to log in, how the DevSpec client is designed graphically, and how DevSpec can be customized for each user.

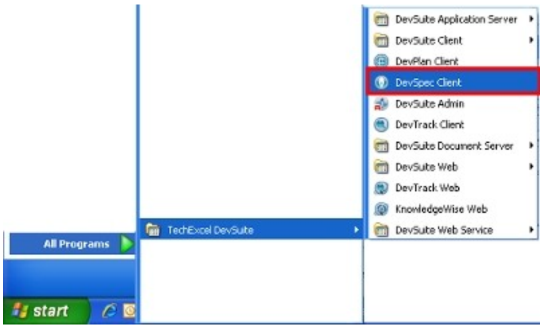
1 Getting Started

DevSpec is accessed through a smart-client: a system that uses HTTP requests to communicate from the client to the server. This means that a connection from anywhere is a guaranteed good performance.

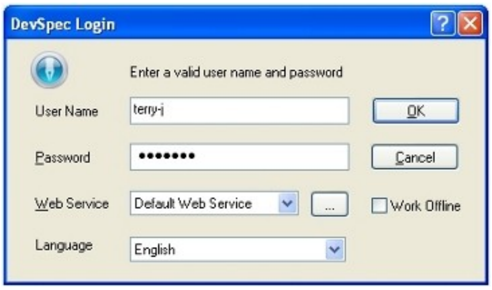
DevSpec implements project-level security by assigning a unique username, password, and account type to all project team members. Individual logins enable the DevSpec system to identify, control, and track the changes that each user makes to project data. Passwords provide accountability for all transactions and other changes to project data and enable the organization to ensure that only authorized individuals may access and change project data. Every user is assigned an account type that defines the role and responsibilities of that user in the project.

Login

1. To login to DevSpec, select the DevSpec icon from your DevSuite program folder.



2. DevSpec client login screen will be prompted. Enter a username and password to login.


DevSpec Login

Enter a valid user name and password

User Name: terry-j

Password: •••••••

Web Service: Default Web Service

□ Work Offline

Language: English

Users who do not have a username and password can login with the built-in sample user that TechExcel offers:

User Name: terry-j
Password: terry-j

Optional: To select a web service, select an option from the*Web Service*dropdown list. Click the ellipses button (…) to define a new web service connection.

Note: Additional controls such as*Language*,*LDAP server*, and the*Work Offline*check box would also be displayed in the login dialog box if these features are turned on by the system administrator:

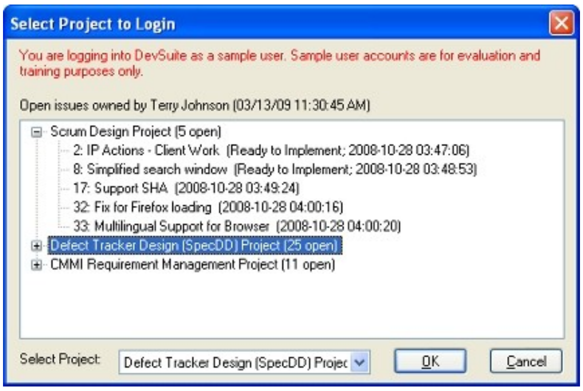
> Language:To choose the language used in DevSpec client, select an option from the*Language*dropdown list.
>
> LDAP Server**:**To connect to the LDAP server to authenticate logins, select an option from the*LDAP Server*dropdown list.

Work Offline:Selecting the*Work Offline*check box enables users to work on DevSpec projects locally by connecting to the local web service, while the work will be synced to the live system later.

3. Click the*OK*button to log into the DevSpec client.

Project Selection

If the user is a member of multiple DevSpec projects, the*Select Project to Login*dialog box appears. The*Select Project to Login*dialog box displays the user's DevSpec projects and qll open work items (requirements, specifications, and change requests) in each project.
Select a DevSpec project or work item in the project list and click the*OK*button. The DevSpec smart client will then display the selected project (and work item).



Select Project to Login

You are logging into DevSuite as a sample user. Sample user accounts are for evaluation and training purposes only.

Open issues owned by Terry Johnson (03/13/09 11:30:45 AM)

- Scrum Design Project (5 open)
  - 2: IP Actions - Client Work (Ready to Implement; 2008-10-28 03:47:06)
  - 8: Simplified search window (Ready to Implement; 2008-10-28 03:48:53)
  - 17: Support SHA (2008-10-28 03:49:24)
  - 32: Fix for Firefox loading (2008-10-28 04:00:16)
  - 33: Multilingual Support for Browser (2008-10-28 04:00:20)
- Defect Tracker Design (SpecDD) Project (25 open)
- CMMI Requirement Management Project (11 open)

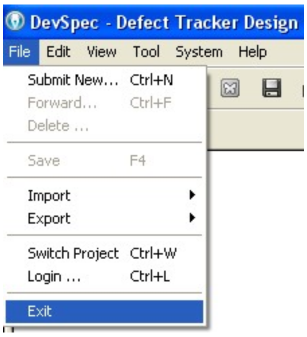
Select Project: Defect Tracker Design (SpecDD) Projec

OK Cancel

Note: DevSpec includes several pre-configured sample projects with different settings to help users understand and manage product design using various types of development methods. DevSpec initially comes with three sample projects: Scrum Design Project, Defect Tracker Design, and CMMI Requirement Management Projects.

Exiting DevSpec

To exit DevSpec, select*File>Exit*in the menu bar.



2 Understanding the DevSpec Client User Interface

The DevSpec Client is a web service-based client that enables users to view, manage, and track requirements, specifications, and change requests.

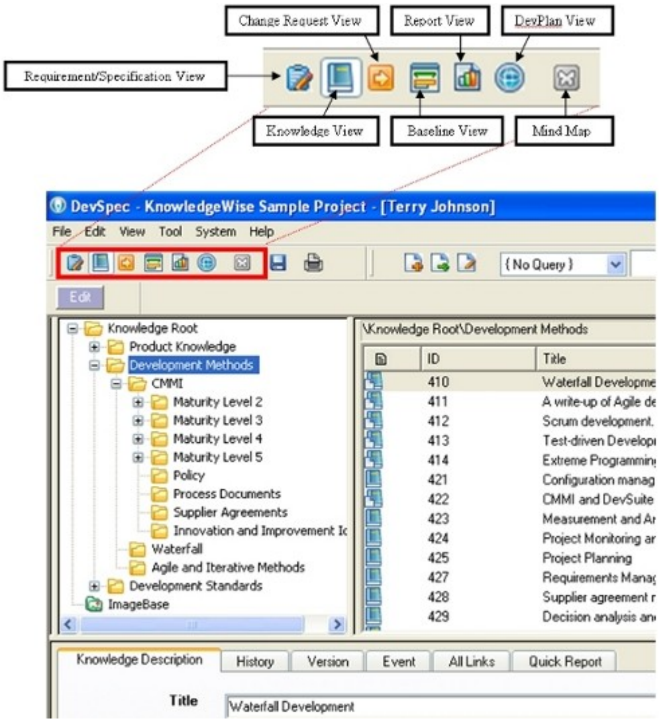
The client workspace organizes project data in multiple views. A view is an interface that displays and organizes data in the client workspace. Each view displays tools and controls that enable the user to process the work items managed in that view.

The workspace of each view is organized into panels (the specification tree panel, list panel, and detail panel) and bars (the menu bar, tool bar, and status bar).

2.1 Views

Every DevSpec project is represented in the DevSpec client by a customized graphical user interface (GUI) that provides project members with the tools that they need to manage and track specifications, requirements, change requests and changes to specifications and requirements.

The DevSpec client may display seven different views: the Requirement/Specification view, the Knowledge view, the Change Request view, the Report view, the Baseline View, the DevPlan view and the Mind Map view.



\Knowledge Root\Development Methods

ID	Title
410	Waterfall Developme
411	A write-up of Agile de
412	Scrum development.
413	Test-driven Develop
414	Extreme Programmin
421	Configuration manag
422	CMMI and DevSuite
423	Measurement and Ar
424	Project Monitoring ar
425	Project Planning
427	Requirements Manag
428	Supplier agreement r
429	Decision analysis an

Title: Waterfall Development

Requirement/Specification View:

The specification view is the primary view for creating, managing, and tracking requirements/specifications in the DevSpec client.

Knowledge View:

The knowledge view is the primary view for creating, managing, and tracking knowledge items in the DevSpec client.

Change Request View:

The change request view is the primary view for creating, managing, and tracking change requests in the DevSpec client.

Baseline View:

The baseline view is used to compare one of the previous specifications and the current specification, and view the differences between them.

Report View:

The report view is used to generate various types of reports against Requirements, Specifications and Knowledge.

DevPlan View:

The DevPlan view allows users to view the specifications/requirements implementation schedule.

Mind Map View:

The Mind Map is used to represent ideas and concepts in a graphical way.

2.2 Bars

DevSpec client bars enables the user to manage the work items displayed in the view where the user is currently situated. For instance, if a user is in the specification view, he or she would be able to use the bars to perform various actions to the specification items.

Every view displays three tool bars:

Menu Bar: The menu bar organizes DevSpec commands into six different menus: the*File*menu,*Edit*menu,*View*menu,*Tool*menu,*System*menu, and*Help*menu.

Tool Bar: The tool bar displays buttons and controls that enable users to perform common tasks, such as switching between views or filtering the items displayed in the list panel.

Status Bar: The status bar displays information about the number of items displayed in the list panel.

Menu Bar Functions

The menu bar organizes DevSpec commands into seven different menus. The commands displayed in the menu bar can be frequently accessible by shortcut keys or using commands elsewhere in the application.

The menu bar displays six different menus, each including multiple functions:

File Menu

New: creating a new work item
Forward: forwarding an existing work item to a different member
Delete: deleting an existing work item
Save: saving changes made in DevSpec client
Import: importing data from either a text or a XML file to DevSpec
Export: exporting data in DevSpec to either a text or a XML file
Switch Project: switching between DevSpec projects
Login: logging in again with different username
Exit: exiting the DevSpec client

Edit Menu

Select All: selecting all the work items in the list panel
Search: bringing up the search window
Go To: bringing up the quick search*Go to*box
Load Query: bringing up the query loading box

View Menu

Tool Bar: displaying or hiding the tool bar
Status Bar: displaying or hiding the status bar
Refresh: refreshing the list panel
Reload Project Setting: reloading project setting to reflect changes made in the Admin client

Tool Menu

Product Release Management: bringing up the product release management window
Project Member Directory: bringing up the team member contact information window
KnowledgeWise Add-In Setup: setting up the Microsoft Office add-in
Mind Map Manager: switching to the Mind Map view
Mind Map Manager Setup: Installing/uninstalling the Mind Map tool

System Menu

User Preferences: configuring the DevSpec client personalization

Help Menu

Help Topics: accessing the DevSpec help articles
Check for Updates: checking to see if the DevSpec client application is up to date
About DevSpec: viewing the DevSpec client version and build info

Tool Bar Functions

The tool bar displays buttons and controls that enable users to perform common tasks

The*Knowledge View*button enables the user to display the knowledge view in the workspace.

The*Specification View*button enables the user to display the specification view in the workspace.

The*Change Request View*button enables the user to display the change request view in the workspace.

The*Baseline View*button enables the user to display the baseline view in the workspace.

The*DevPlan View*button enables the user to launch the DevPlan client from DevSpec.

The*Report View*button enables the user to display the report view in the workspace.

The*Mind Map View*button enables the user to display the mind map view in the workspace.

The*Save*button enables the user to save changes made to a work item (knowledge item, requirement, specification, or change request).

The*Print*button enables the user to print work item (knowledge item, requirement, specification, or change request) details.

The*New*button enables the user to submit a new work item (knowledge item, requirement, specification, or change request) to the project.

The*Forward*button enables the user to forward a selected work item (knowledge item, requirement, specification, or change request) to another project member.

The*Edit*button enables the user to edit a selected work item (knowledge item, requirement, specification, or change request).

The*Search*button enables the user to filter the work items displayed in the list panel using a custom-defined query.

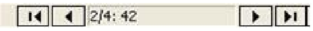
The*Cancel Search*button enables the user to cancel the query used to filter the work items displayed in the list panel.

The*Product Version Tree*button enables the user to display the product tree panel in the workspace.

The*Display folder properties as detail tabs*button enables the user to switch the display of detailed information between a selected work item and a work item folder in the detail panel.

Status Bar

The status bar displays the number of items displayed in the list panel and the sequence number of the selected item.



To display or hide the status bar in the workspace of the DevSpec client, select*View>Status Bar*in the menu bar. The status bar is displayed in the workspace if a black check mark is displayed next to the*Status Bar*command in the*View*menu.

Using status bar control buttons, users may view the first item, previous item, next item, and last item in the list.

To go to the first item in the list, click the Start button.

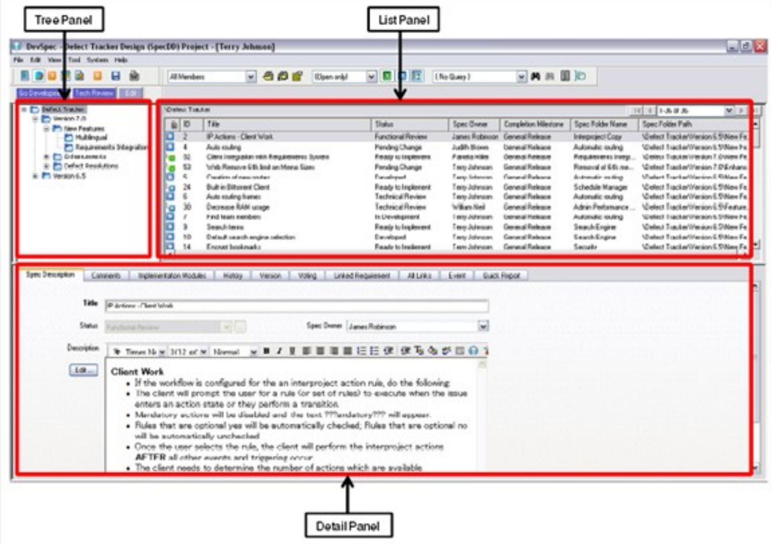
To go to the previous item in the list, click the Previous button.

To go to the next item in the list, click the Next button.

To go to the last item in the list, click the Last button.

2.3 Panels

Panels are important concepts in the knowledge view, specification view, change request view and the baseline view. In these views, the DevSpec workspace is divided into three, or sometimes four, panels for efficient data management.



\Defect Tracker

ID	Title	Status	Spec Owner	Completion Milestone	Spec Folder Name	Spec Folder Path
2	IP Actions - Client Work	Functional Review	James Robinson	General Release	Interproject Copy	\Defect Tracker\Version 6.5\New Fe
4	Auto routing	Pending Change	Judith Brown	General Release	Automatic routing	\Defect Tracker\Version 6.5\New Fe
32	Client Integration with Requirements System	Ready to Implement	Pamela Hiler	General Release	Requirements Integr...	\Defect Tracker\Version 7.0\New Fe
52	Web Remove 64k limit on Memo Sizes	Pending Change	Terry Johnson	General Release	Removal of 64k me...	\Defect Tracker\Version 7.0\Enhanc
5	Creation of new router	Developed	Terry Johnson	General Release	Automatic routing	\Defect Tracker\Version 6.5\New Fe
24	Built in Bittorrent Client	Ready to Implement	Terry Johnson	General Release	Schedule Manager	\Defect Tracker\Version 6.5\New Fe
6	Auto routing frames	Technical Review	Terry Johnson	General Release	Automatic routing	\Defect Tracker\Version 6.5\New Fe
30	Decrease RAM usage	Technical Review	William Neil	General Release	Admin Performance ...	\Defect Tracker\Version 6.5\Feature
7	Find team members	In Development	Terry Johnson	General Release	Automatic routing	\Defect Tracker\Version 6.5\New Fe
9	Search terms	Ready to Implement	Terry Johnson	General Release	Search Engine	\Defect Tracker\Version 6.5\New Fe
10	Default search engine selection	Developed	Terry Johnson	General Release	Search Engine	\Defect Tracker\Version 6.5\New Fe
14	Encrypt bookmarks	Ready to Implement	Terry Johnson	General Release	Security	\Defect Tracker\Version 6.5\New Fe

Title: IP Actions - Client Work

Status: Functional Review

Spec Owner: James Robinson

Description:

Client Work

- If the workflow is configured for the an interproject action rule, do the following:
- The client will prompt the user for a rule (or set of rules) to execute when the issue enters an action state or they perform a transition.
- Mandatory actions will be disabled and the text ???andatory??? will appear.
- Rules that are optional yes will be automatically checked; Rules that are optional no will be automatically unchecked
- Once the user selects the rule, the client will perform the interproject actions **AFTER** all other events and triggering occur
- The client needs to determine the number of actions which are available

Specification Tree Panel: The specification tree panel is a hierarchal structure composed of folders and subfolders that organizes work items into distinct areas of work.

List Panel: The list panel shows high-level information for multiple work items in a tabular format of rows and columns.

Detail Panel: The detail panel displays detailed information for a single work item or work item folder in multiple tabs.

Product Tree Panel: The product tree panel organizes development items by product, version, and build.

Note: The product version tree panel may be optionally displayed in the requirement view, specification view, and change request view. Use the*Product Version Tree*button in the tool bar to display or hide the product version tree panel.

Specification Tree Panel

The specification tree panel is a hierarchal structure composed of folders and subfolders that organizes work items into distinct areas of work.

For more information about the specification tree panel, please see chapter 3, section 1.2,*Specification Details*.

List Panels

DevSpec list panels enable the user to quickly view high-level information for multiple development work items (knowledge items, requirements, specifications, or change requests) in a tabular list of rows and intersecting columns.

Using controls in the tool bar, the tree panel, and the product tree panel, users may filter and sort records based on various work item property values.

Tracking/Filtering Work Items in the List Panel

In DevSpec, list panels enable project members to quickly view high-level information for multiple work items in each view. The list panel shows high-level information for work items (knowledge item, specifications, requirements, and change requests) in a tabular list of rows and intersecting columns. Note that detailed information for the highlighted item in the list panel is displayed in the detail panel.

Since there may be a long list of items displayed in the list panel, work item filtering and querying is therefore crucial. To work effectively, users must be able to quickly identify and execute appropriate filters so that they can quickly access relevant work items based on key indicators. Filtering minimizes the time needed to review large numbers of records, as well as to maximize effectiveness.

In DevSpec, a filter is a simple query that returns development work items matching a defined set of criteria. Records matching the criteria are displayed in the report—all other work items are “filtered out” and not shown. Filtering controls, such as the owner list and status list, enable project members to filter work items by owner, state, status, and other properties:

> **Owner List:**The Owner List enables project members to view a subset of work items based on the current owner or original submitter.
>
> **Status List:**The Status List enables project members to view a subset of work items based on the work item status or

workflow state.

For more information about filtering the list panel, see chapter 4,*Searches and Queries*.

Refreshing the List Panel

The refresh command enables project members to refresh the items displayed in the list panel with the latest data from the DevSpec database. Project members can refresh the list panel by two methods:

- Press F5.
- Select*View>Refresh*in the menu bar.

Detail Panels

In DevSpec, the detail panel displays detailed information for a single work item or work item folder. The data displayed in the detail panel is view-specific: knowledge item data is displayed in the knowledge view, requirement and specification data in the specifications view, and change request data in the change request view.

The*Switch Detail*button enables the user to display detailed information for a selected work item or work item folder in the detail panel. When the detail panel is set to display the work item information, the detail panel will display the detailed information about the highlighted work item in the list panel. And when the detail panel is set to display the work item folder properties, the detail panel will display the detailed information for the highlighted work item folder in the tree panel.

For more information about the specification tree panel, please see chapter 3, section 1.2,*Specification Details*.

Product Version Tree Panel

The product tree represents the products, versions, and under-development builds in a DevSuite site in a hierarchical structure of folders and subfolders. Each folder in the product tree represents a specific product, version, or build.

Project members may filter the work items (knowledge items, requirements, specifications, and change requests) displayed in the list panel by selecting a work item folder in the product version tree panel.

For more information about the specification tree panel, please see chapter 3, section 2.2,*Product/Version Folder Tree*.

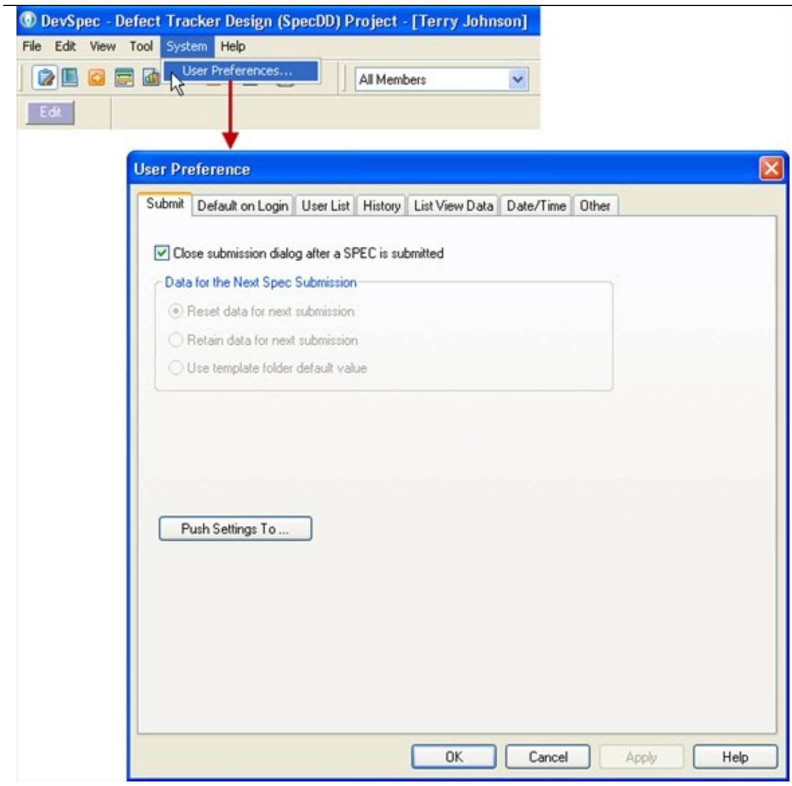
3 DevSpec Personalization

DevSpec users are able to easily customize their own personal DevSpec interface. Since each user has their own set of unique needs, it is important that everyone is able to use DevSpec in a way that is optimal to them individually–thus saving time and maximizing efficiency. Some of the ways users can personalize DevSpec are how they login, how specifications are submitted, and how they view items in certain lists and tabs. Users can also change their DevSpec password here.

User preferences are used to define the settings for the current project only.

Note: This section covers how*users*customize DevSpec. For more information on how*project administrators*customize DevSpec, please see the*DevSuite Admin Guide*.

Users can set their preferences using the*User Preference*dialog, which consists of seven different tabs. To open the*User Preference*dialog, select*System>User Preferences…*in the menu bar.



☑ Close submission dialog after a SPEC is submitted

Data for the Next Spec Submission

● Reset data for next submission

○ Retain data for next submission

○ Use template folder default value

When finished making changes to the user preferences, the*OK*button must be clicked in order to close and save.

Note: User preferences may be edited in multiple tabs before pressing the*OK*button.

***Push Settings To*button**

Sometimes a project manager will want other users to view DevSpec just as he/she does. By using the*Push Settings To*button in any tab, the settings on that tab (and that tab only) will be copied, or “pushed”, to other users’ settings. This action is privileged controlled.

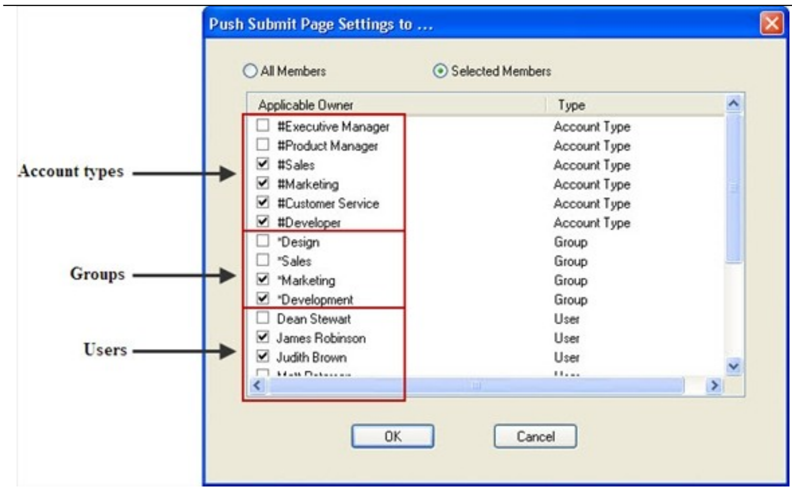
To push settings to another user:

1. In any tab of the*User Preference*dialog, set the settings that are to be shared.

2. Click the*Push Settings To…*button.

3. In the newly opened dialog, select the users, to whom the settings are to be pushed.

> Click the*All Members*radio button to push settings to all users in the project, or click the*Selected Members*radio button to push settings to selected account types, groups, or users. Account types are defined by a preceding number sign (#). Groups are defined by a preceding asterisk (*). All others are users.



Push Submit Page Settings to ...

○ All Members ● Selected Members

Applicable Owner	Type
□ #Executive Manager	Account Type
□ #Product Manager	Account Type
☑ #Sales	Account Type
☑ #Marketing	Account Type
☑ #Customer Service	Account Type
☑ #Developer	Account Type
□ *Design	Group
□ *Sales	Group
☑ *Marketing	Group
☑ *Development	Group
□ Dean Stewart	User
☑ James Robinson	User
☑ Judith Brown	User

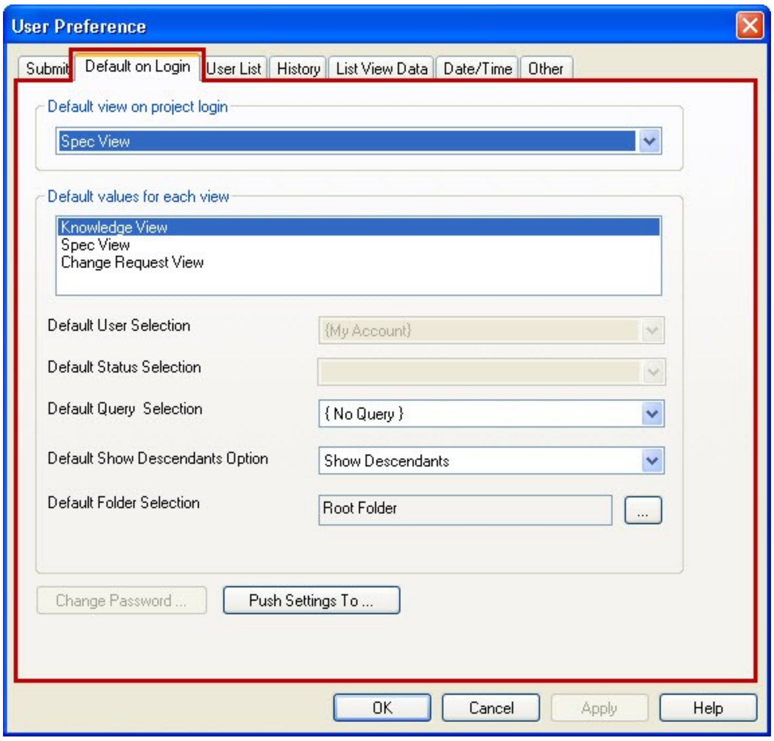
In the above screenshot, all users with a*Sales*account type will receive the pushed settings, while all users with a*Product Manager*account type will not.

4. Click the*OK*button to return to the*User Preference*dialog. This action only accepts the list of receiving users.

5. Click the*OK*button in the*User Preference*dialog to push the settings.**If the dialog is not closed via the*OK*button, the settings will not be pushed.**

3.1 Personalizing Default Settings on Login

Under the*Default on Login*tab, users can set which view will open first when they log in to DevSpec, set the default filter settings for each view, and change their DevSpec password.



User Preference

Default view on project login: Spec View

Default values for each view: Knowledge View (selected), Spec View, Change Request View

Default User Selection: {My Account}

Default Status Selection:

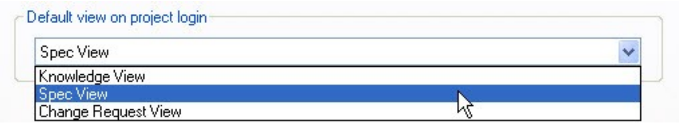
Default Query Selection: { No Query }

Default Show Descendants Option: Show Descendants

Default Folder Selection: Root Folder

Default view on login

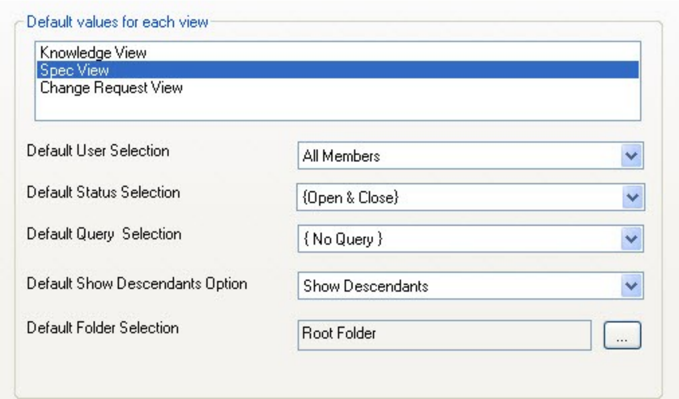
To set the default view after logging in to DevSpec, select a view from the*Default view on project login*dropdown control. There are up to three options:*Knowledge View*,*Spec View*, or*Change Request View*(if user has access).



Default view on project login: Spec View

Default filter settings

To set the default filter settings for each view, under the*Default values for each view*section, select a view. Then set the default value for each filter for that particular view. Repeat for the other views as well.



Default values for each view

Knowledge View
Spec View
Change Request View

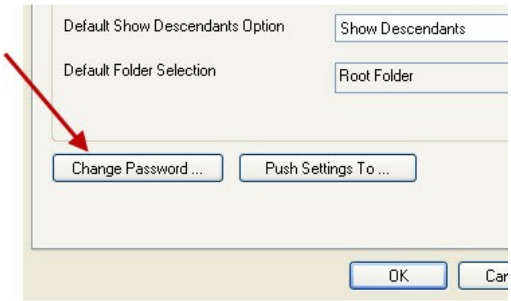
Default User Selection: All Members
Default Status Selection: {Open & Close}
Default Query Selection: { No Query }
Default Show Descendants Option: Show Descendants
Default Folder Selection: Root Folder

For more information on using filters in DevSpec, please see chapter 4, section 1,*Quick Filters*.

Change password

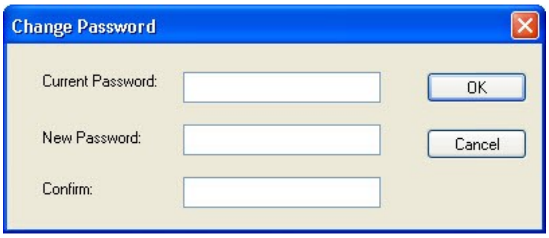
If a user wishes to change their password:

1. Click the*Change Password…*button.



Default Show Descendants Option: Show Descendants
Default Folder Selection: Root Folder

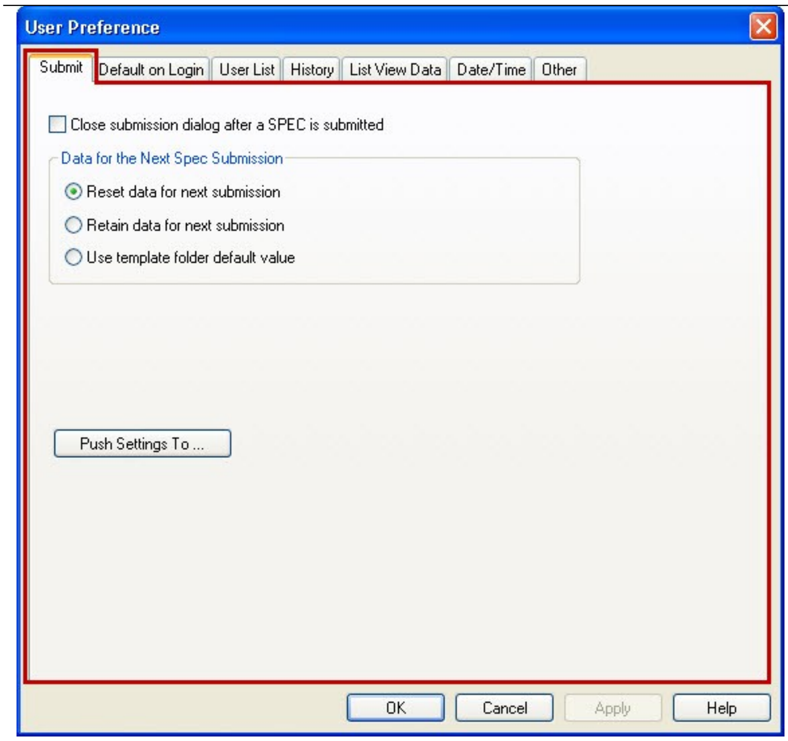
2. In the newly opened*Change Password*dialog, enter the current password, enter the new password, and then confirm the new password again.



3. Click the*OK*button to submit the password change.

3.2 Personalizing the Work Item Submission Dialog

Under the*Submit*tab, users can set how data is retained after the submission of a new specification. For more information on submitting specifications, please see chapter 3, section 1.1,*Specification Workflow*.



□ Close submission dialog after a SPEC is submitted

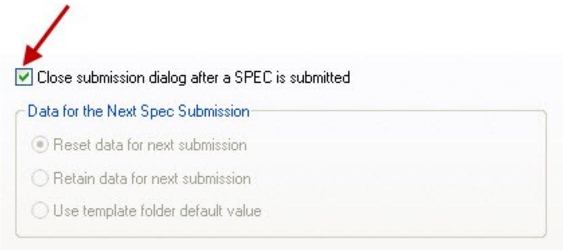
Data for the Next Spec Submission

● Reset data for next submission

○ Retain data for next submission

○ Use template folder default value

In the dialog used to submit new specifications, there is a checkbox to close the window after submission. To have this box (in the submission dialog) checked by default, check the*Close submission dialog after a SPEC is submitted*checkbox (in the*User Preference*dialog).



☑ Close submission dialog after a SPEC is submitted

Data for the Next Spec Submission

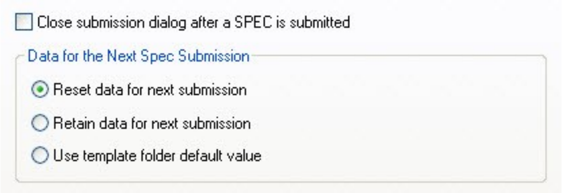
● Reset data for next submission

○ Retain data for next submission

○ Use template folder default value

When this checkbox is unchecked, the submission dialog will not close after a submission. In this case, users may also set how the submitted data is handled in regards to the next specification. In the*Data for the Next Submission*control, users may choose from one of three radio buttons:

Reset data for next submission: All fields and controls are cleared and/or returned to their default values.
Retain data for next submission: The folder, item type, and template fields remain the same from the previous submission. The title, status, and description fields remain the same too only when a template is selected; otherwise, they are cleared.
Use template folder default value: When a template is selected, after submission, all fields return to their template default values, regardless if these values have been edited. However, if no template is selected, this is equivalent to*Retain data for next submission*.



□ Close submission dialog after a SPEC is submitted

Data for the Next Spec Submission

● Reset data for next submission

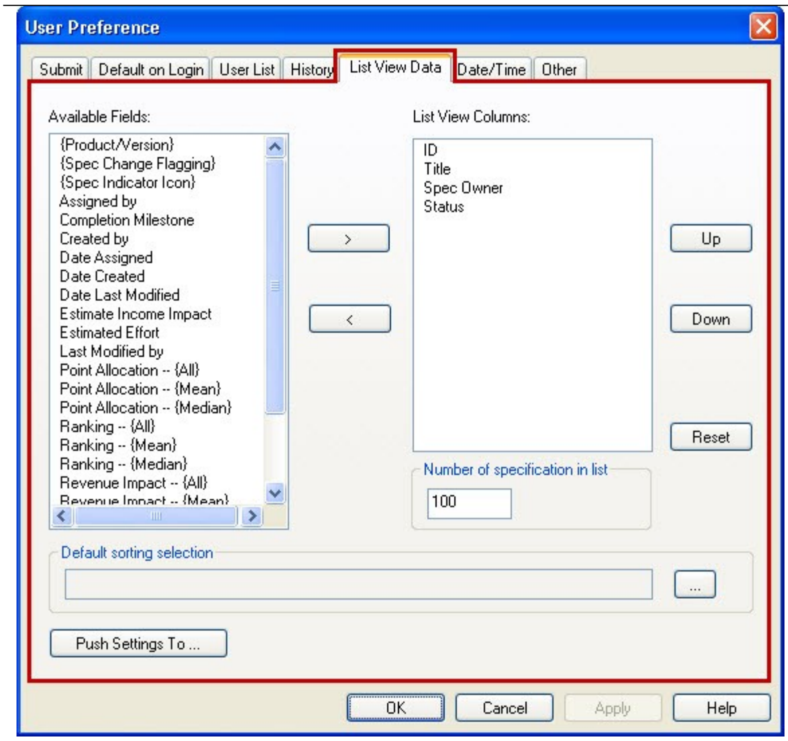
○ Retain data for next submission

○ Use template folder default value

For more information on using templates, please see chapter 3, section 1.1,*Specification Workflow*.

3.3 Personalizing the List Panel

Under the*List View Data*tab, users can set how they view the list panel: adding, removing, and ordering visible property fields; setting the number of specifications that are displayed per page; and defining the default sorting rules.



Available Fields:

{Product/Version}
{Spec Change Flagging}
{Spec Indicator Icon}
Assigned by
Completion Milestone
Created by
Date Assigned
Date Created
Date Last Modified
Estimate Income Impact
Estimated Effort
Last Modified by
Point Allocation -- {All}
Point Allocation -- {Mean}
Point Allocation -- {Median}
Ranking -- {All}
Ranking -- {Mean}
Ranking -- {Median}
Revenue Impact -- {All}
Revenue Impact -- {Mean}

List View Columns:

ID
Title
Spec Owner
Status

Number of specification in list: 100

Default sorting selection:

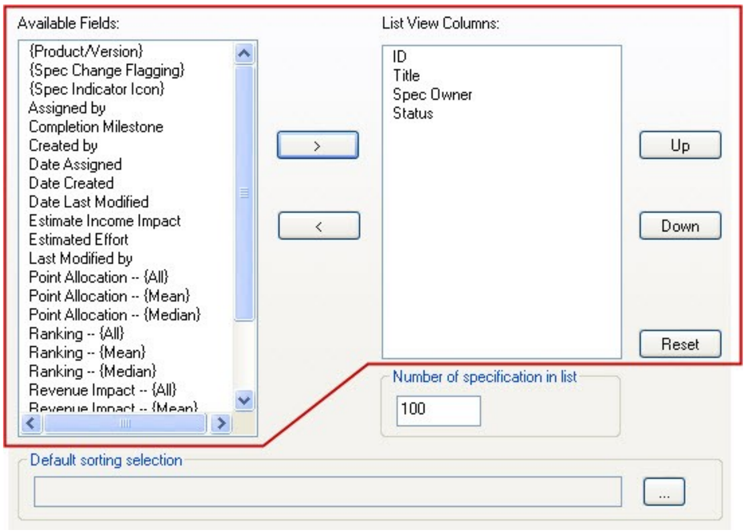
Adding and removing fields

To add a field, highlight a field name from the*Available Fields*list, and then click the right arrow button to move it to the*List View Column*list.

To remove a field, highlight a field name from the*List View Column*list, and then click the left arrow button to move it back to the*Available Fields*list.

To define the field order, highlight a field name from the*List View Column*list, and use the*Up*and*Down*buttons to set the field’s position.

Click the*Reset*button to restore the project’s default settings, as defined by the project administrator.



Available Fields:

{Product/Version}
{Spec Change Flagging}
{Spec Indicator Icon}
Assigned by
Completion Milestone
Created by
Date Assigned
Date Created
Date Last Modified
Estimate Income Impact
Estimated Effort
Last Modified by
Point Allocation -- {All}
Point Allocation -- {Mean}
Point Allocation -- {Median}
Ranking -- {All}
Ranking -- {Mean}
Ranking -- {Median}
Revenue Impact -- {All}
Revenue Impact -- {Mean}

List View Columns:

ID
Title
Spec Owner
Status

Number of specification in list: 100

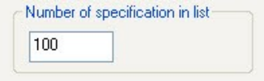
Default sorting selection:

In the above screenshot,*ID*,*Title*,*Spec Owner*, and*Status*are the properties that will be displayed in the list view, in that order.

Note: To select multiple fields, press and hold Ctrl while selecting other fields. To select all fields, click on the first field, and then while pressing and holding Ctrl and Shift, select the final field.

Number of specifications in list

To set the number of specifications that are displayed per page in the list panel, enter a number between 10 and 500 in the*Number of specifications in list*field. The list panel will not show more than 500 specifications in order to preserve optimal performance.



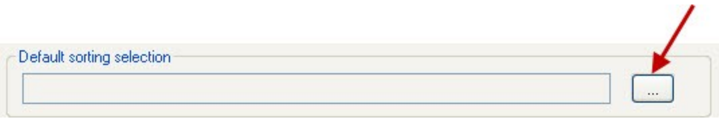
Number of specification in list: 100

Default sorting selection

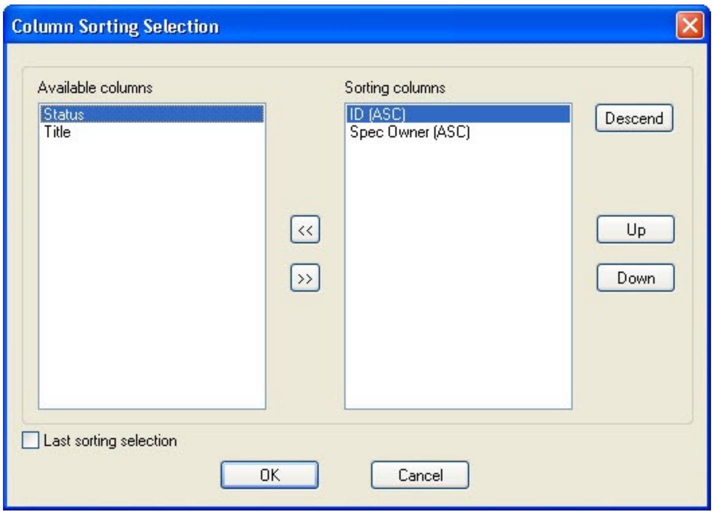
Users can also set the default specification sorting rules. For more information on sorting, please see chapter 4, section 1.7,*Sorting*.

To define default sorting rules:

1. Click the ellipses button (…) in the*Default sorting selection*section. The*Column Sorting Selection*dialog will open.

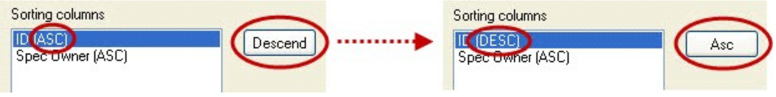


2. To select a default property by which to be sorted, highlight a property name from the*Available columns*list, and then click the right arrow button to move it to the*Sorting columns*list. To remove a property, highlight a property name from the*Sorting columns*list, and then click the left arrow button to move it back to the*Available columns*list.



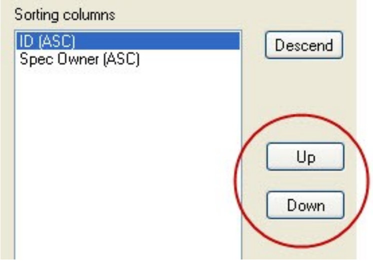
Note: To select multiple fields, press and hold Ctrl while selecting other fields. To select all fields, click on the first field, and then while pressing and holding Ctrl and Shift, select the final field.

Ascending or descending order: Once a property is added, the order in which the specifications are sorted, ascending or descending, can be defined. The default value is ascending order, but to switch to descending order, highlight a property in the*Sorting columns*list, and then click the*Descend*button. The button will change to the*Asc*button–used to switch back to ascending order. Each property's current sorting value is display with a proceeding*(ASC)*or*(DESC)*.

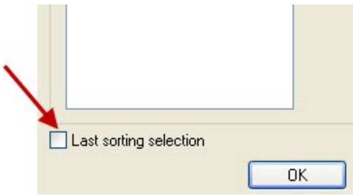


The screenshot above demonstrates what happens when the*Descend*button is clicked.

Secondary sorting: In cases when multiple specifications have the same value of the property that is being sorted, users can add multiple properties and define an order for secondary sorting. To define the sorting order, highlight a property name from the*List View Column*list, and use the*Up*and*Down*buttons to set the property's sorting priority.



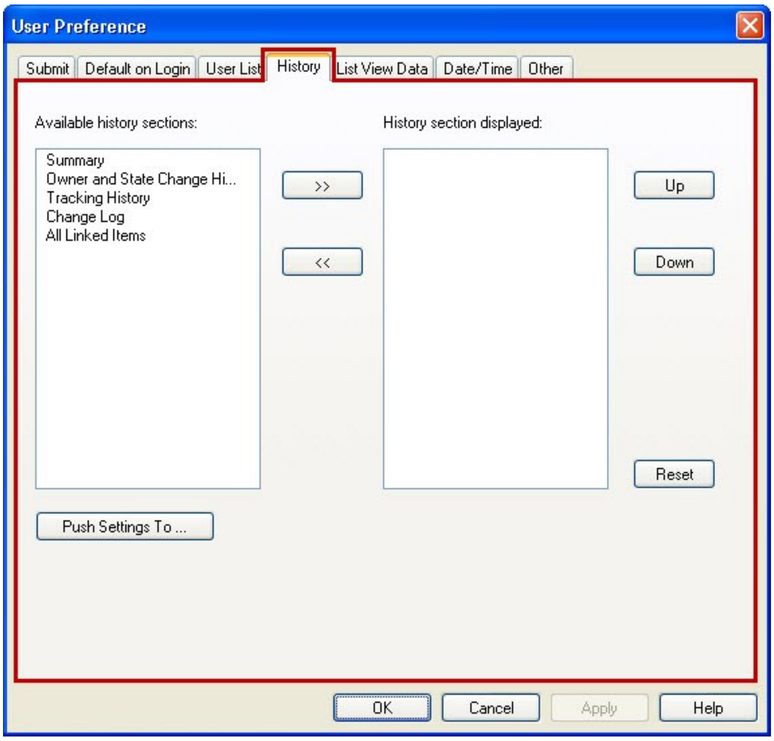
Last sorting selection: Users can choose the default sort to simply be however the list panel was sorted when the list panel was last used, even when that may be a manual sort, and even when users have logged off and on. To select this option, check the*Last sorting selection*checkbox. This will make all other previously mentioned controls in the*Column Sorting Selection*dialog inactive, or “grayed-out”. Since this option simply applies the most recent sort, all other sorting rules would be overridden.



3. Click the*OK*button to submit the default sorting rules, and to return to the*User Preference*dialog.

3.4 Personalizing the History Page in the Detail Panel

Under the*History*tab (in the*User Preference*dialog), users can add and remove sections to the*History*tab**in the detail panel**. For more information on using the*History*tab in the detail panel, please see chapter 3, section 1.2,*Specification Details*.



There are five possible sections in the*History*tab in the detail panel:
Summary
Owner and State Change History
Tracking History
Change Log
All Linked Items
To add a section, highlight a section name from the*Available history sections*list, and then click the right arrow button to move it to the*History section displayed*list.

To remove a section, highlight a field name from the*History section displayed*list, and then click the left arrow button to move it back to the*Available history sections*list.

To define the section order, highlight a field name from the*History section displayed*list, and use the*Up*and*Down*buttons to set the section's position.

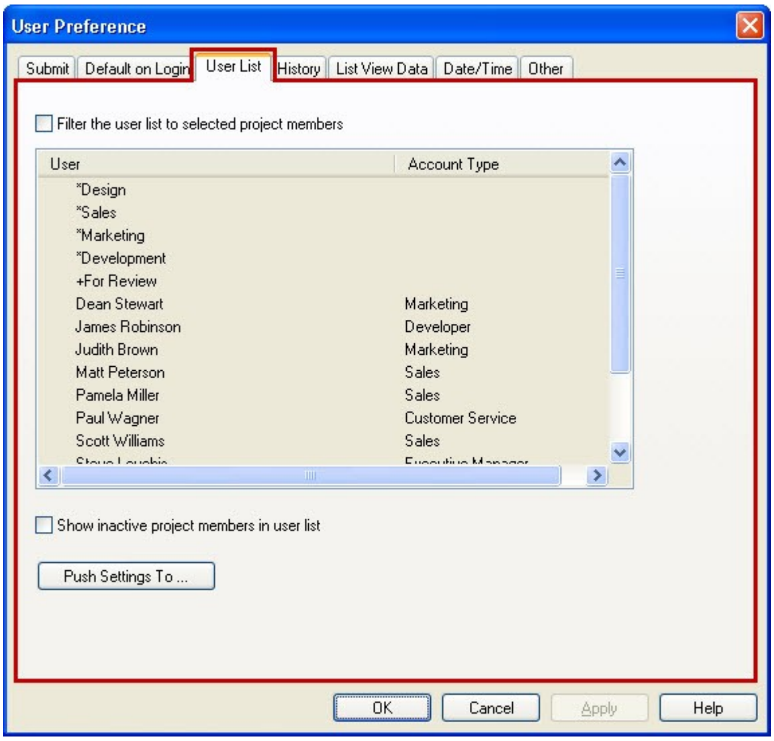
Click the*Reset*button to restore the project's default settings, as defined by the project administrator.

Note: If no section is added to the*History section displayed*list, the*Owner and State Change History*and*Summary*sections will be displayed by default, in that order.

Note: To select multiple fields, press and hold Ctrl while selecting other fields. To select all fields, click on the first field, and then, while pressing and holding Ctrl and Shift, select the final field.

3.5 Personalizing the Owner Dropdown List

Under the*User List*tab, users can define which users, groups, and group folders of the project are displayed in the owner filter dropdown control in the tool bar. This could be beneficial to users who don't need to see items owned by other particular owners. For more information on filtering with the owner filter, please see chapter 4, section 1.2,*Filtering by Owner*.



□ Filter the user list to selected project members

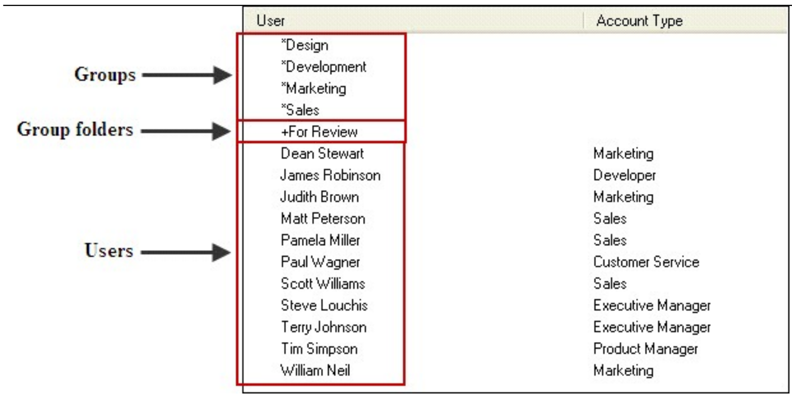
User	Account Type
*Design	
*Sales	
*Marketing	
*Development	
+For Review	
Dean Stewart	Marketing
James Robinson	Developer
Judith Brown	Marketing
Matt Peterson	Sales
Pamela Miller	Sales
Paul Wagner	Customer Service
Scott Williams	Sales

□ Show inactive project members in user list

To simply select all users, groups, and group folders, leave the*Filter the user list to selected project members*checkbox unchecked, as in the previous screenshot. The owners list will remain inactive, or "grayed-out".

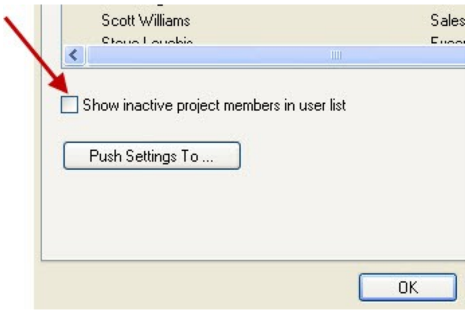
To select the owners one-by-one, check the*Filter the user list to selected project members*checkbox, which will make the owners list active and accessible. Then click on the users, groups, and group folders that are to be shown in the owner filter (a red check appears next to each selected owner); click again to uncheck.

Users are denoted by their project member name.
Groups are denoted by a preceding asterisk (*).
Group folders are denoted by a preceding plus sign (+).



User	Account Type
*Design	
*Development	
*Marketing	
*Sales	
+For Review	
Dean Stewart	Marketing
James Robinson	Developer
Judith Brown	Marketing
Matt Peterson	Sales
Pamela Miller	Sales
Paul Wagner	Customer Service
Scott Williams	Sales
Steve Louchis	Executive Manager
Terry Johnson	Executive Manager
Tim Simpson	Product Manager
William Neil	Marketing

Whether all owners are selected or not, users have the option to include inactive project members in the owner list. To show inactive users, check the*Show inactive project members in user list*checkbox. To hide inactive users, uncheck the box.



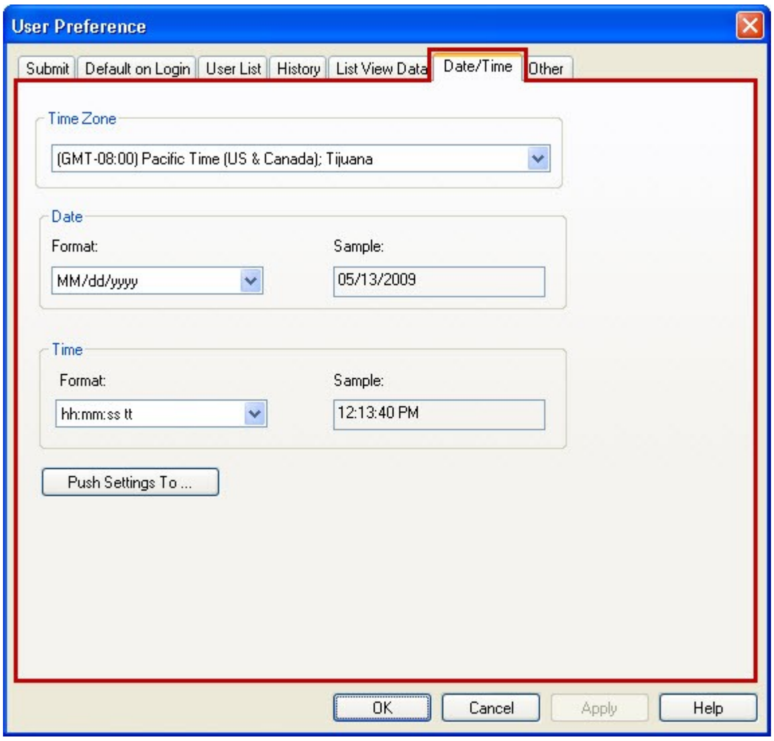
Scott Williams	Sales

□ Show inactive project members in user list

Note: Although inactive users may not be displayed in the owner filter in the tool bar, they still will always be displayed in the user list on the*User List*tab of the*User Preferences*dialog.

3.6 Personalizing Date/Time Settings

Under the*Date/Time*tab, users can define the format of time stamps, including selecting a time zone. These settings can be set only when the project administrator allows it. If not allowed, these controls are inactive, or “grayed-out”.

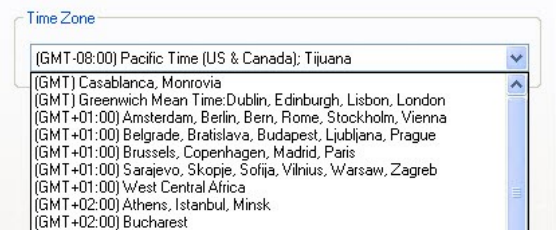


Time Zone: (GMT-08:00) Pacific Time (US & Canada); Tijuana

Date
Format: MM/dd/yyyy
Sample: 05/13/2009

Time
Format: hh:mm:ss tt
Sample: 12:13:40 PM

To select a time zone, choose a time zone from the*Time Zone*dropdown control.

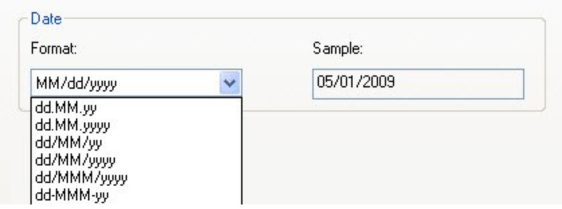


Time Zone

(GMT-08:00) Pacific Time (US & Canada); Tijuana

(GMT) Casablanca, Monrovia
(GMT) Greenwich Mean Time:Dublin, Edinburgh, Lisbon, London
(GMT+01:00) Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna
(GMT+01:00) Belgrade, Bratislava, Budapest, Ljubljana, Prague
(GMT+01:00) Brussels, Copenhagen, Madrid, Paris
(GMT+01:00) Sarajevo, Skopje, Sofija, Vilnius, Warsaw, Zagreb
(GMT+01:00) West Central Africa
(GMT+02:00) Athens, Istanbul, Minsk
(GMT+02:00) Bucharest

To select a date format, choose a format from the*Date*section. The current date will appear as a sample of the selected format.



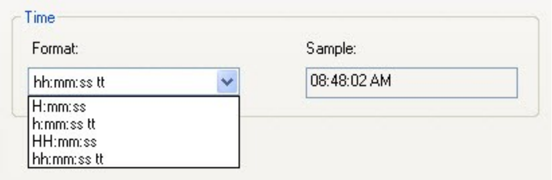
Date

Format: MM/dd/yyyy

Sample: 05/01/2009

dd.MM.yy
dd.MM.yyyy
dd/MM/yy
dd/MM/yyyy
dd/MMM/yyyy
dd-MMM-yy

To select a time format, choose a format from the*Time*section. The current time will appear as a sample of the selected format.



Time

Format: hh:mm:ss tt

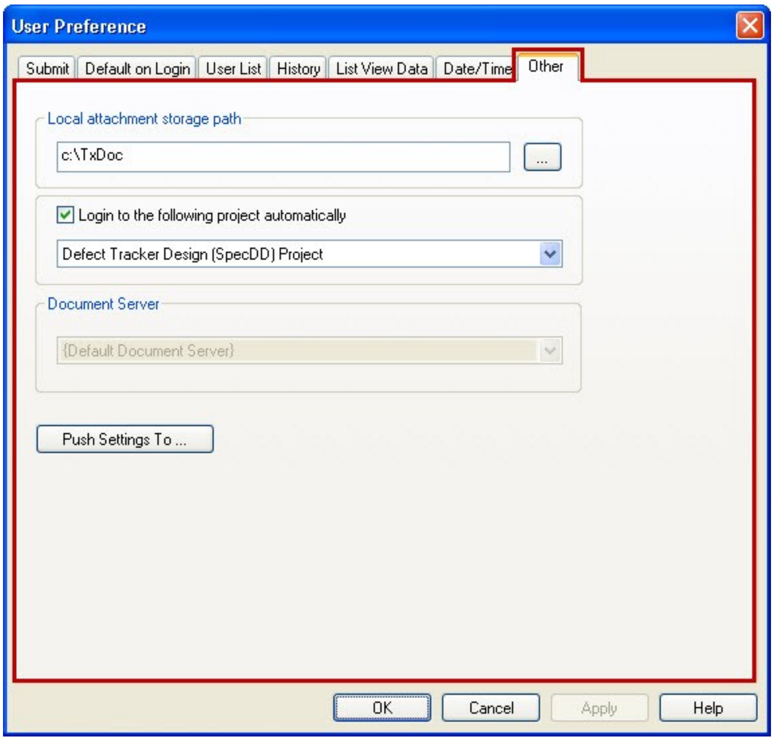
Sample: 08:48:02 AM

H:mm:ss
h:mm:ss tt
HH:mm:ss
hh:mm:ss tt

Note: Regardless of who submits, forwards, or edits a work item, or even where it happened, time stamps in DevSpec, when the format is user-defined, will always be displayed according to the time zone defined by the user who is viewing the time stamp.

3.7 Additional Personalization Settings

Under the *Other*tab, users can define the attachment storage path on their local computer, set up DevSpec to login in automatically to a specific project, and choose a document server.



Local attachment storage path: c:\TxDoc

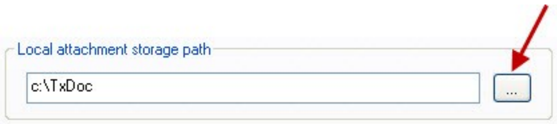
☑ Login to the following project automatically: Defect Tracker Design (SpecDD) Project

Document Server: {Default Document Server}

Local attachment storage path

In DevSpec users can download, or check out, specification attachments from the document server and save to their local computer. To define the path to where these attachments are saved:

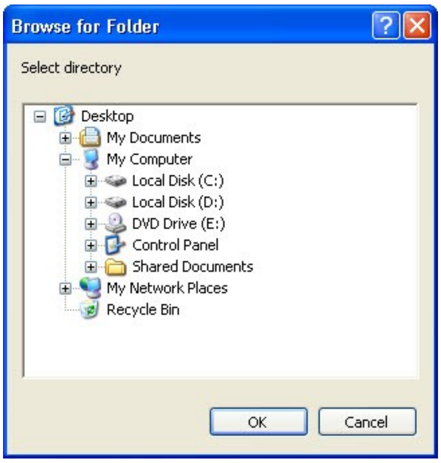
1. In the*Local attachment storage path*section, click the ellipses button (…).



Local attachment storage path

c:\TxDoc

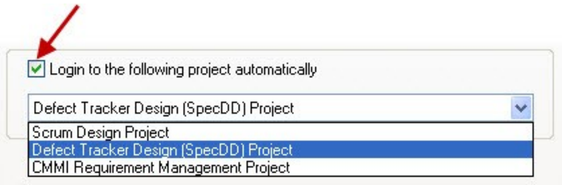
2. In the newly opened*Browse for Folder*dialog, navigate the local computer for the desired directory, and highlight it.



3. Click the*OK*button to save, close the*Browse for Folder*dialog, and return to the*User Preference*dialog.

Automatically login to a specific project

To skip the*Select Project to Login*dialog on login, and to be automatically logged in to a specific project, check the*Login to the following project automatically*checkbox and select a project. Uncheck to choose a project on login.



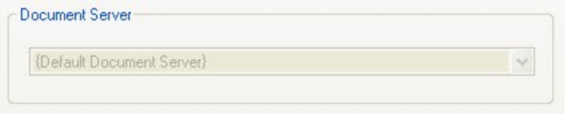
☑ Login to the following project automatically

Defect Tracker Design (SpecDD) Project

Scrum Design Project
Defect Tracker Design (SpecDD) Project
CMMI Requirement Management Project

Document server

Project administrators can create multiple document servers based on location. The*Document Server*control allows users to choose one that is closest to them for optimal performance. If there is only one document server, this control is inactive, or “grayed-out”, and*{Default Document Server}*is displayed.



Document Server

{Default Document Server}

Chapter 3 - Using DevSpec

This chapter covers basic management of specifications, requirements, and the folder trees.

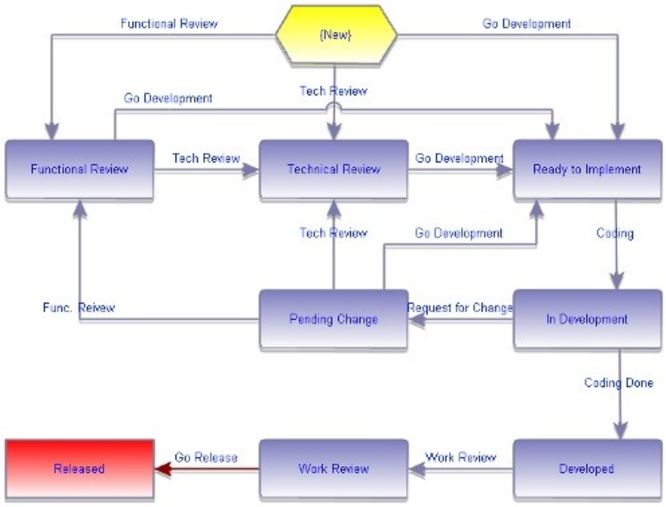
1 Basic Usage

1.1 Specification Workflow

The DevSpec workflow, managed in the specification view, defines how requirements or specifications are created, managed, and tracked in a DevSpec project.

Administrator-defined workflow determines the sequence of workflow states-how and when a specification may pass from one workflow state to the next. Each state is also privileged controlled-who may submit, forward, edit, or delete a specification at each stage of its lifecycle. This provides traceability for DevSpec projects.

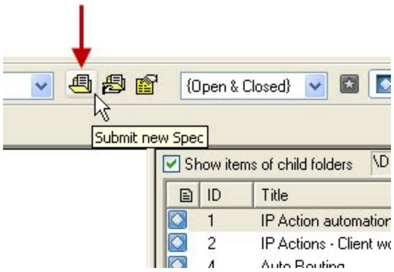
The follow diagram is a possible workflow. For example, after a new specification is created, it can start in the*Functional Review*state. Then it can move on to the*Technical Review*state (via the*Tech Review*transition), or even further along to the*Ready to Implement*state (via the*Go Development*transition).



1.1.1 Submitting

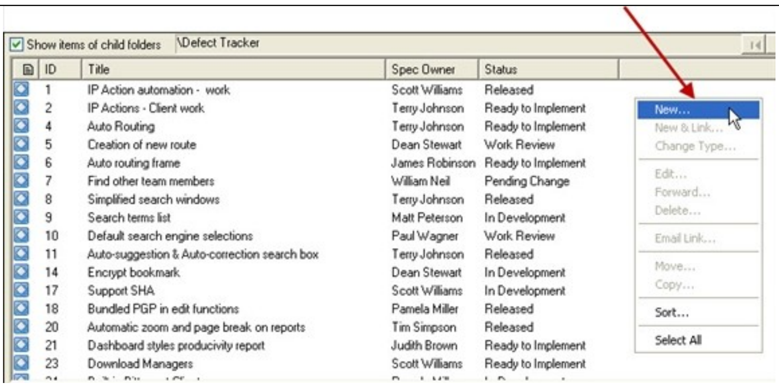
(Mandatory steps in bold)

1. Open the*Submission Pages*window by one of the following commands:
Click the*Submit New Spec*button in the tool bar



ID	Title
1	IP Action automatior
2	IP Actions - Client w
4	Auto Routing

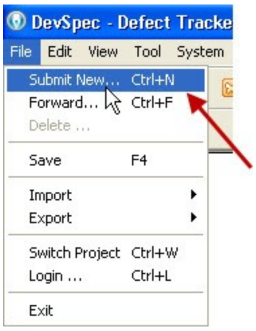
Right click in the specification list panel and click*New…*



☑ Show items of child folders \Defect Tracker

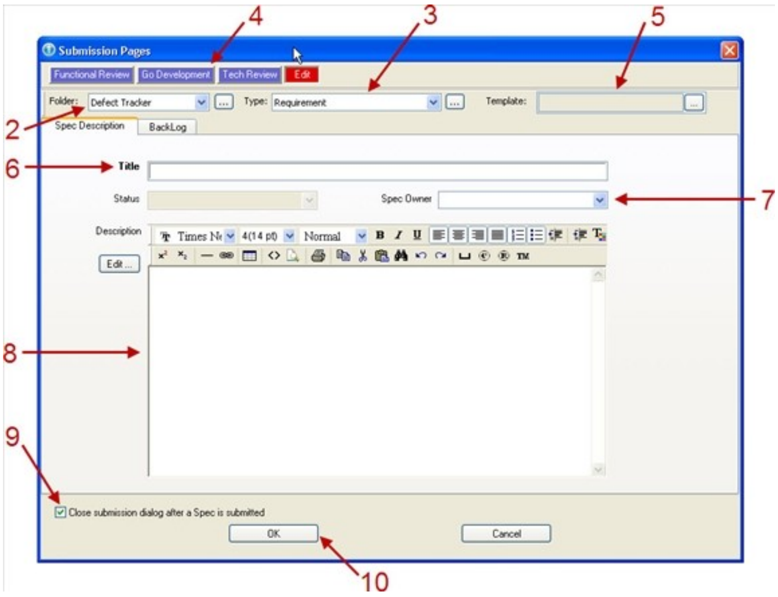
ID	Title	Spec Owner	Status
1	IP Action automation - work	Scott Williams	Released
2	IP Actions - Client work	Terry Johnson	Ready to Implement
4	Auto Routing	Terry Johnson	Ready to Implement
5	Creation of new route	Dean Stewart	Work Review
6	Auto routing frame	James Robinson	Ready to Implement
7	Find other team members	William Neil	Pending Change
8	Simplified search windows	Terry Johnson	Released
9	Search terms list	Matt Peterson	In Development
10	Default search engine selections	Paul Wagner	Work Review
11	Auto-suggestion & Auto-correction search box	Terry Johnson	Released
14	Encrypt bookmark	Dean Stewart	In Development
17	Support SHA	Scott Williams	In Development
18	Bundled PGP in edit functions	Pamela Miller	Released
20	Automatic zoom and page break on reports	Tim Simpson	Released
21	Dashboard styles producivity report	Judith Brown	Ready to Implement
23	Download Managers	Scott Williams	Ready to Implement

From the menu bar, select*File*>*Submit New…*



Press Ctrl + N.

* To auto-populate the value of the folder field in the submission dialog, click on a folder in the specification folder tree prior to performing any of the above actions.



2. Select the folder to which the specification is to be added. New specification may be created within a folder.

3. Select whether the new item is a requirement or a specification.

4. Select the state of the specification. By default it will start in the*New*state.

5. Select a specification template. For more information about creating specification templates, see the ensuing section,*Creating a Template*.

6. Define the specification title.

7. Define the specification owner.

8. Define the specification description.

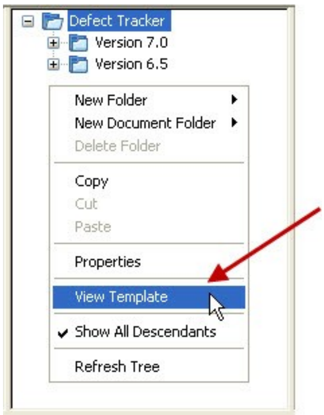
9. To close the window upon submission, check the*Close submission dialog after a Spec is submitted*box. To keep the window open to submit another new specification, leave the box unchecked.

10. Click the*OK*button.

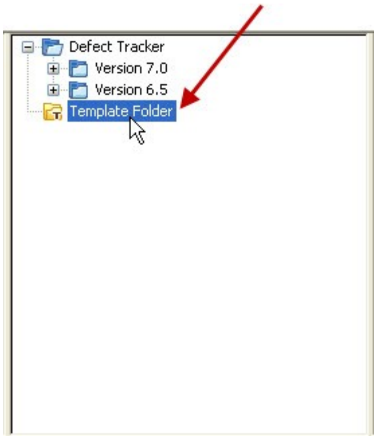
1.1.2 Creating a Template

A template is a predefined collection of definitions-namely the item type, title, owner, and description-that may be used to submit a new specification, quickly and easily.

1. Right-click in the folder tree panel and click*View Template*.



2. Click on the newly appeared*Template Folder*.



3. Create a new specification template just as if an actual specification were being created (see the previous section,*Submitting*). Although a template will appear as a specification in the list panel, it is not an actual specification.

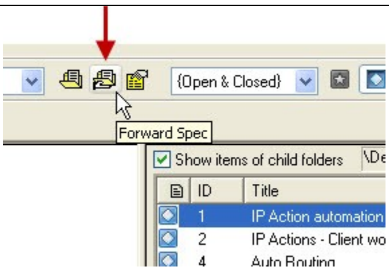
1.1.3 Forwarding

Forwarding is primarily done to change the owner of the specification, but can also be used to change the state of the specification as well.

(Mandatory steps in bold)

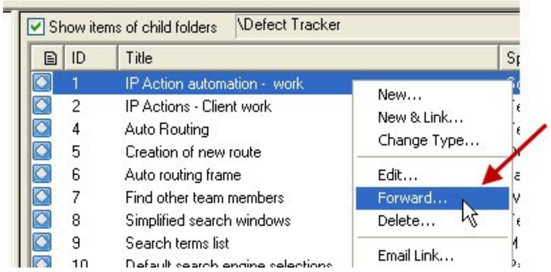
1. Open the*Forward Pages*window for the specification to be forwarded. This can be done by one of the follow options:

Highlight a specification in the list panel and press the*Forward Spec*button in the tool bar.



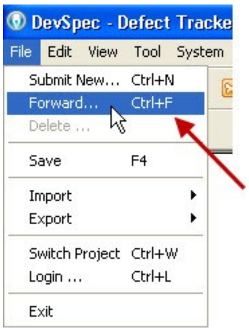
ID	Title
1	IP Action automation
2	IP Actions - Client wo
4	Auto Routing

Right-click a specification in the list panel and click*Forward…*

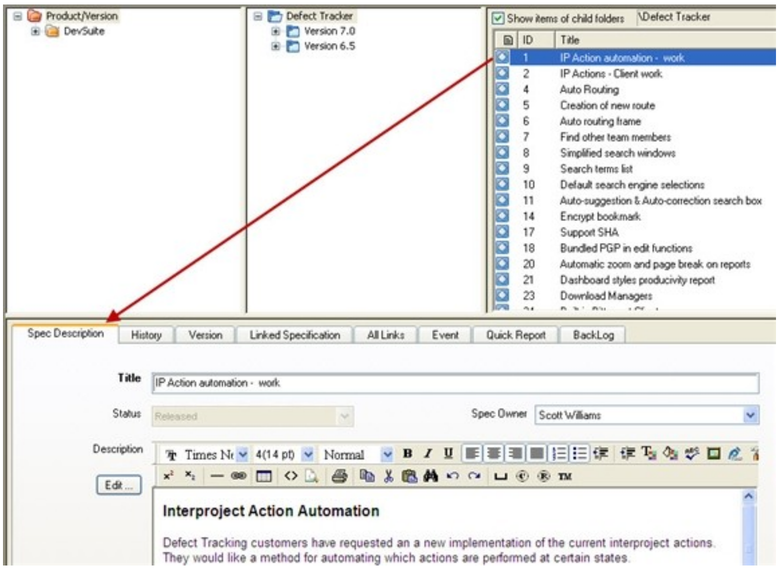


ID	Title
1	IP Action automation - work
2	IP Actions - Client work
4	Auto Routing
5	Creation of new route
6	Auto routing frame
7	Find other team members
8	Simplified search windows
9	Search terms list
10	Default search engine selections

Highlight a specification in the list panel and from the menu bar, select*File>Forward…*



Highlight a specification in the list panel and press Ctrl + F.
Highlight a specification in the list panel and click the*Spec Description*tab in the detail panel. (No window opens, but rather the specification is forwarded directly from the detail panel.)



ID	Title
1	IP Action automation - work
2	IP Actions - Client work
4	Auto Routing
5	Creation of new route
6	Auto routing frame
7	Find other team members
8	Simplified search windows
9	Search terms list
10	Default search engine selections
11	Auto-suggestion & Auto-correction search box
14	Encrypt bookmark
17	Support SHA
18	Bundled PGP in edit functions
20	Automatic zoom and page break on reports
21	Dashboard styles producivity report
23	Download Managers

Title: IP Action automation - work

Status: Released

Spec Owner: Scott Williams

Description:

Interproject Action Automation

Defect Tracking customers have requested an a new implementation of the current interproject actions. They would like a method for automating which actions are performed at certain states.

2. Once the*Forward Pages*window opens up, define the new specification owner, to whom the specification will be forwarded.

3. Make any other necessary changes to the specification properties, such as title, state, and description.

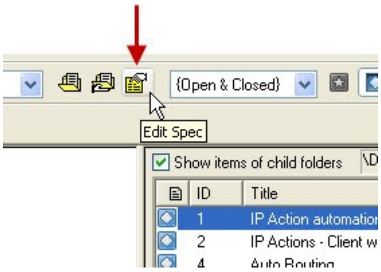
4. Click the*OK*button to forward the specification to the next owner and/or next state.

1.1.4 Editing

(Mandatory steps in bold)

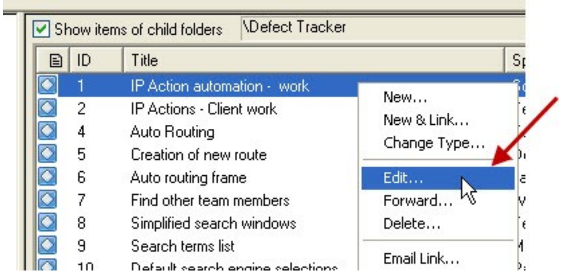
1. Open the*Editing Pages*window for the specification to be edited. This can be done by one of the follow options:

Highlight a specification in the list panel and press the*Edit Spec*button in the tool bar.



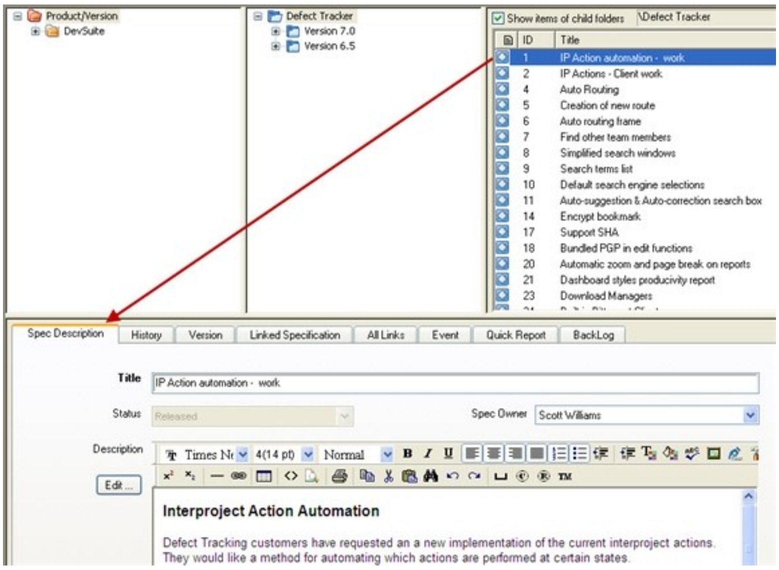
ID	Title
1	IP Action automatio
2	IP Actions - Client w
4	Auto Routing

Right-click a specification in the list panel and click*Edit…*



ID	Title
1	IP Action automation - work
2	IP Actions - Client work
4	Auto Routing
5	Creation of new route
6	Auto routing frame
7	Find other team members
8	Simplified search windows
9	Search terms list
10	Default search engine selections

Highlight a specification in the list panel and click the*Spec Description*tab in the detail panel. (No window opens, but rather the specification is forwarded directly from the detail panel.)



ID	Title
1	IP Action automation - work
2	IP Actions - Client work
4	Auto Routing
5	Creation of new route
6	Auto routing frame
7	Find other team members
8	Simplified search windows
9	Search terms list
10	Default search engine selections
11	Auto-suggestion & Auto-correction search box
14	Encrypt bookmark
17	Support SHA
18	Bundled PGP in edit functions
20	Automatic zoom and page break on reports
21	Dashboard styles producivity report
23	Download Managers

Title: IP Action automation - work

Status: Released

Spec Owner: Scott Williams

Description:

Interproject Action Automation

Defect Tracking customers have requested an a new implementation of the current interproject actions. They would like a method for automating which actions are performed at certain states.

2. Once the*Editing Pages*window opens up, the user can change the properties of the specification, such as folder, type, title, state, owner, and description. For example, using the*Type*control, a requirement can be changed to a specification (and vice versa).

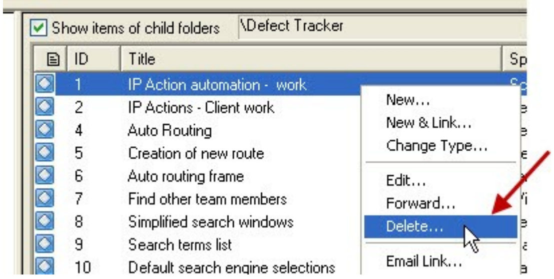
3. Click the*OK*button to save the changes made to the specification.

1.1.5 Deleting

Warning: Deleting a specification is a non-recoverable action!

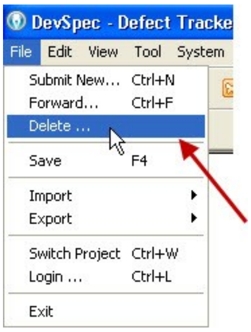
1. A specification can be deleted by one of the follow options:

Right-click a specification in the list view and click*Delete…*



ID	Title	Sp
1	IP Action automation - work	
2	IP Actions - Client work	
4	Auto Routing	
5	Creation of new route	
6	Auto routing frame	
7	Find other team members	
8	Simplified search windows	
9	Search terms list	
10	Default search engine selections	

Highlight a specification in the list view and from the menu bar, select*File>Delete…*



2. A confirmation box will appear. Press the OK button.

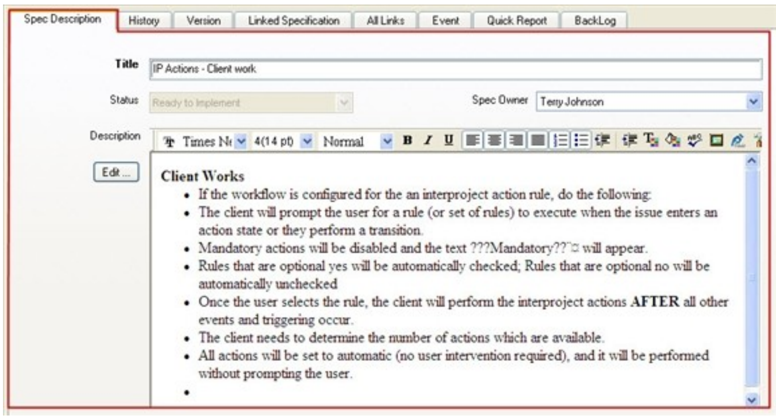
1.2 Specification Details

The detail panel (bottom pane with tabs) displays detailed information about the highlighted specification or requirement in the specification view. Details of each item is further divided and distributed across multiple tabs, allowing you to quickly view different information for the highlighted requirement or specification.

***Note*: The label for each specification detail tab is easily customizable in the DevSuite Admin.**

1.2.1 Description

The*Spec Description*tab is the main page that displays key specification details. It consists of standard fields and custom defined fields.



Title: IP Actions - Client work

Status: Ready to Implement

Spec Owner: Terry Johnson

Description:

Client Works

- If the workflow is configured for the an interproject action rule, do the following:
- The client will prompt the user for a rule (or set of rules) to execute when the issue enters an action state or they perform a transition.
- Mandatory actions will be disabled and the text ???Mandatory??¨¤ will appear.
- Rules that are optional yes will be automatically checked; Rules that are optional no will be automatically unchecked
- Once the user selects the rule, the client will perform the interproject actions **AFTER** all other events and triggering occur.
- The client needs to determine the number of actions which are available.
- All actions will be set to automatic (no user intervention required), and it will be performed without prompting the user.
-

Standard Fields:

Title: A short, one line description of the specification (used in the list view). It’s a single line edit box.

Description: Detailed description of the specification. It’s a multi-line edit box. This field can be enabled for editing with time stamps. Administrators can enable this field for HTML editing, spell check, support image base and URLs.

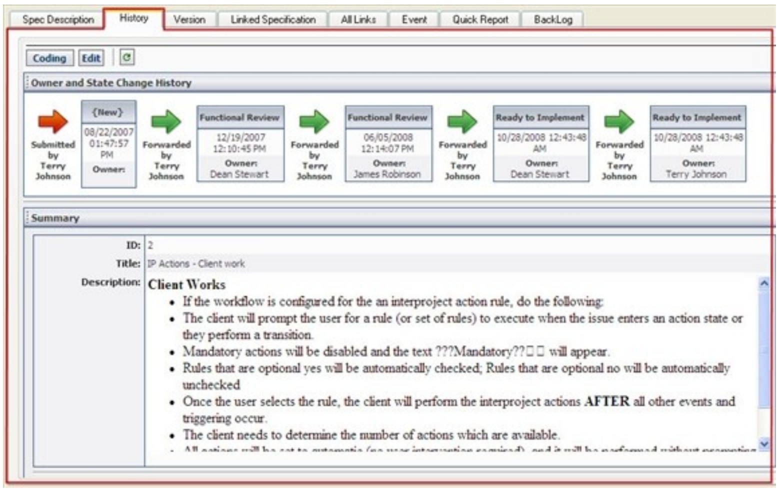
Status: This field indicates the current state of a specification within the specification workflow. The specification state is normally changed using the transition buttons on the secondary tool bar.

Owner: This field indicates the current owner of a specification. It's a dropdown field.

Other custom defined fields: DevSpec administrator may define and place additional custom fields on this tab as needed.

1.2.2 History

As a specification progresses through states and is modified, every change to the specification is tracked and logged under the*History*tab. It consists of different sections that allow users to easily view the information. This tab is a consolidated "one-page stop" for viewing all specification details that are distributed across all other tabs.



ID: 2

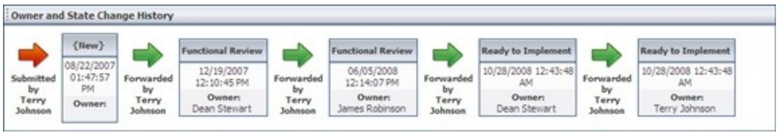
Title: IP Actions - Client work

Description: Client Works

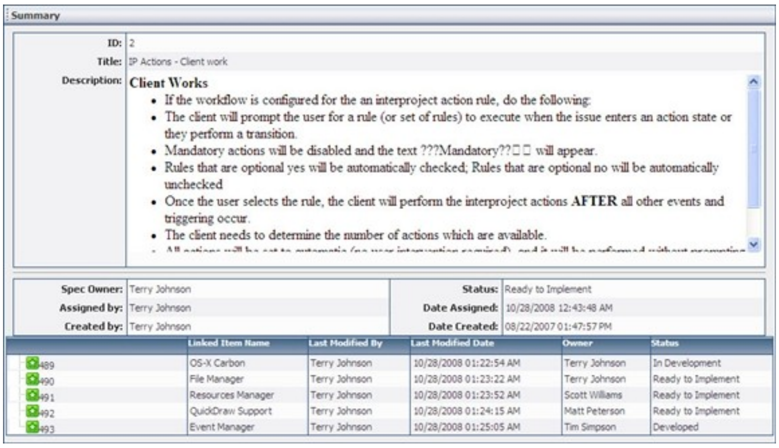
- If the workflow is configured for the an interproject action rule, do the following:
- The client will prompt the user for a rule (or set of rules) to execute when the issue enters an action state or they perform a transition.
- Mandatory actions will be disabled and the text ???Mandatory??□□ will appear.
- Rules that are optional yes will be automatically checked; Rules that are optional no will be automatically unchecked
- Once the user selects the rule, the client will perform the interproject actions **AFTER** all other events and triggering occur.
- The client needs to determine the number of actions which are available.
- All actions will be set to automatic (no user intervention required), and it will be performed without prompting

History Sections:

Owner and State Change History --- This section is a graphical representation of the specification flow through different states and different owners, along with time stamps.



Summary --- This section provides information on all specification fields being tracked for the highlighted specification.



ID: 2

Title: IP Actions - Client work

Description: Client Works

- If the workflow is configured for the an interproject action rule, do the following:
- The client will prompt the user for a rule (or set of rules) to execute when the issue enters an action state or they perform a transition.
- Mandatory actions will be disabled and the text ???Mandatory??□□ will appear.
- Rules that are optional yes will be automatically checked; Rules that are optional no will be automatically unchecked
- Once the user selects the rule, the client will perform the interproject actions **AFTER** all other events and triggering occur.
- The client needs to determine the number of actions which are available.
- All actions will be set to automatic (no user intervention required), and it will be performed without prompting

Spec Owner: Terry Johnson

Assigned by: Terry Johnson

Created by: Terry Johnson

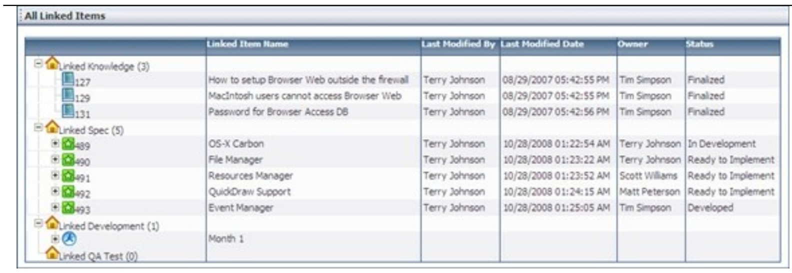
Status: Ready to Implement

Date Assigned: 10/28/2008 12:43:48 AM

Date Created: 08/22/2007 01:47:57 PM

	Linked Item Name	Last Modified By	Last Modified Date	Owner	Status
489	OS-X Carbon	Terry Johnson	10/28/2008 01:22:54 AM	Terry Johnson	In Development
490	File Manager	Terry Johnson	10/28/2008 01:23:22 AM	Terry Johnson	Ready to Implement
491	Resources Manager	Terry Johnson	10/28/2008 01:23:52 AM	Scott Williams	Ready to Implement
492	QuickDraw Support	Terry Johnson	10/28/2008 01:24:15 AM	Matt Peterson	Ready to Implement
493	Event Manager	Terry Johnson	10/28/2008 01:25:05 AM	Tim Simpson	Developed

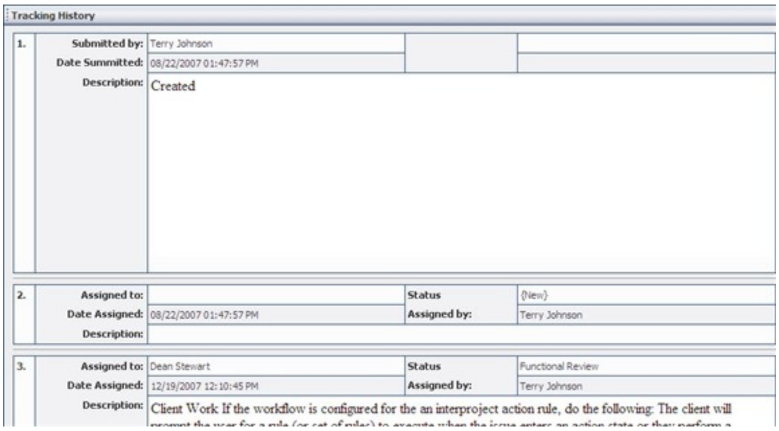
All Linked Items --- This section provides information on other items currently linked to the highlighted specification. This includes other specifications, requirements, knowledge items, development items, and test tasks linked to the current specification.



All Linked Items

	Linked Item Name	Last Modified By	Last Modified Date	Owner	Status
⊟ Linked Knowledge (3)					
127	How to setup Browser Web outside the firewall	Terry Johnson	08/29/2007 05:42:55 PM	Tim Simpson	Finalized
129	MacIntosh users cannot access Browser Web	Terry Johnson	08/29/2007 05:42:55 PM	Tim Simpson	Finalized
131	Password for Browser Access DB	Terry Johnson	08/29/2007 05:42:56 PM	Tim Simpson	Finalized
⊟ Linked Spec (5)					
⊞ 489	OS-X Carbon	Terry Johnson	10/28/2008 01:22:54 AM	Terry Johnson	In Development
⊞ 490	File Manager	Terry Johnson	10/28/2008 01:23:22 AM	Terry Johnson	Ready to Implement
⊞ 491	Resources Manager	Terry Johnson	10/28/2008 01:23:52 AM	Scott Williams	Ready to Implement
⊞ 492	QuickDraw Support	Terry Johnson	10/28/2008 01:24:15 AM	Matt Peterson	Ready to Implement
⊞ 493	Event Manager	Terry Johnson	10/28/2008 01:25:05 AM	Tim Simpson	Developed
⊟ Linked Development (1)					
⊞	Month 1				
Linked QA Test (0)					

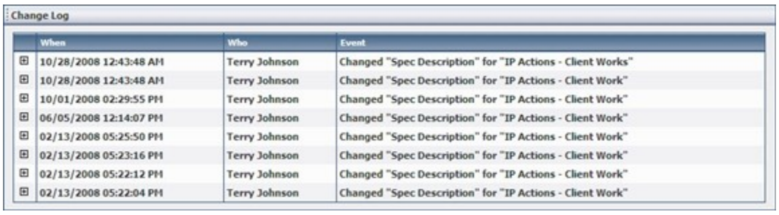
Tracking History --- As the specification is forwarded from one owner to another, each owner's comment is recorded and the new owner assignment is time stamped. This section provides a complete log of owner changes and owner comments, along with time stamps.



Tracking History

1.	Submitted by:	Terry Johnson		
	Date Summitted:	08/22/2007 01:47:57 PM		
	Description:	Created		
2.	Assigned to:		Status	{New}
	Date Assigned:	08/22/2007 01:47:57 PM	Assigned by:	Terry Johnson
	Description:			
3.	Assigned to:	Dean Stewart	Status	Functional Review
	Date Assigned:	12/19/2007 12:10:45 PM	Assigned by:	Terry Johnson
	Description:	Client Work If the workflow is configured for the an interproject action rule, do the following: The client will		

Change Log --- GUI fields defined to track requirements or specifications may be modified or updated during the lifecycle of the DevSpec item. Any change to a field value is tracked in this section, along with the time stamp. It also records the user who changed the field value.

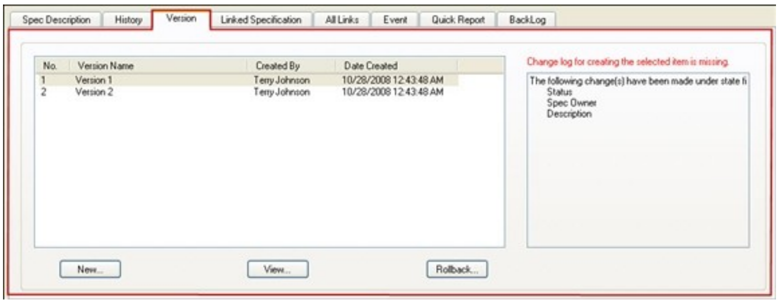


Change Log

	When	Who	Event
⊞	10/28/2008 12:43:48 AM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Works"
⊞	10/28/2008 12:43:48 AM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"
⊞	10/01/2008 02:29:55 PM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"
⊞	06/05/2008 12:14:07 PM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"
⊞	02/13/2008 05:25:50 PM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"
⊞	02/13/2008 05:23:16 PM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"
⊞	02/13/2008 05:22:12 PM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"
⊞	02/13/2008 05:22:04 PM	Terry Johnson	Changed "Spec Description" for "IP Actions - Client Work"

1.2.3 Version

A specification may be modified several times before it is completed or closed. In some cases, you may want to refer back to a previous version before further modification, or even rollback completely. For complete traceability and flexibility, DevSpec allows users to save multiple versions of a specification at different points of time throughout the specification lifecycle.



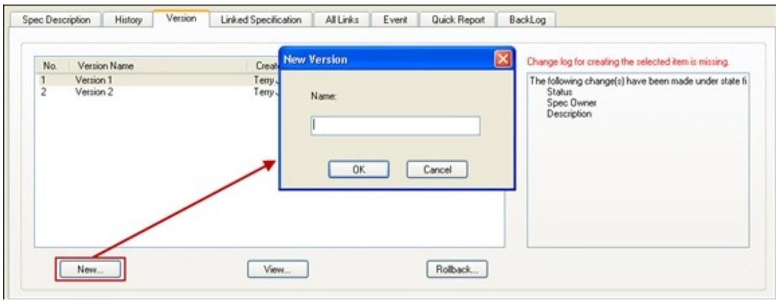
No.	Version Name	Created By	Date Created
1	Version 1	Terry Johnson	10/28/2008 12:43:48 AM
2	Version 2	Terry Johnson	10/28/2008 12:43:48 AM

Change log for creating the selected item is missing.

The following change(s) have been made under state fi
Status
Spec Owner
Description

To save the current version:

Select a requirement or specification
In theDetailtab, go toVersiontab
Click theNewbutton
Enter the version name, and add a comment if necessary
Click theOK button
The current copy of the requirement or specification is saved under this version



1 Version 1 Terry J
2 Version 2 Terry J

Change log for creating the selected item is missing.

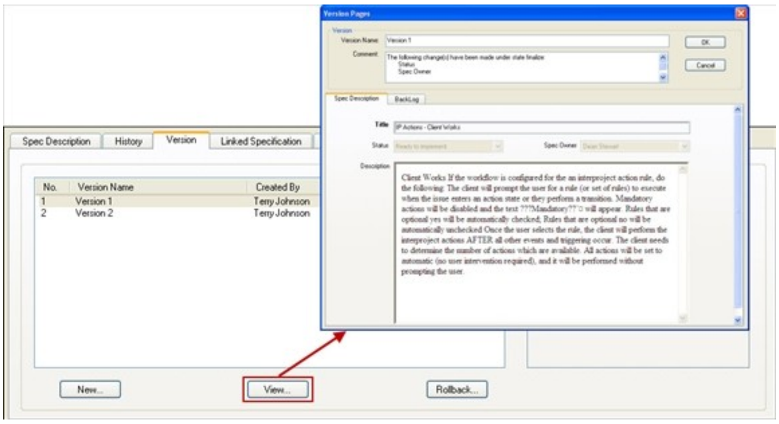
The following change(s) have been made under state fi
Status
Spec Owner
Description

To view a previous version:

Select a requirement or specification

In theDetailpanel, go to theVersiontab

Click theViewbutton



Version Name: Version 1

Comment: The following change(s) have been made under state finalize
Status
Spec Owner

Title: IP Actions - Client Works

Description: Client Works If the workflow is configured for the an interproject action rule, do the following: The client will prompt the user for a rule (or set of rules) to execute when the issue enters an action state or they perform a transition. Mandatory actions will be disabled and the text ???Mandatory??'□ will appear. Rules that are optional yes will be automatically checked; Rules that are optional no will be automatically unchecked Once the user selects the rule, the client will perform the interproject actions AFTER all other events and triggering occur. The client needs to determine the number of actions which are available. All actions will be set to automatic (no user intervention required), and it will be performed without prompting the user.

1 Version 1 Terry Johnson
2 Version 2 Terry Johnson

To rollback the current version:

Select a requirement or specification

In theDetailpanel, go to theVersiontab

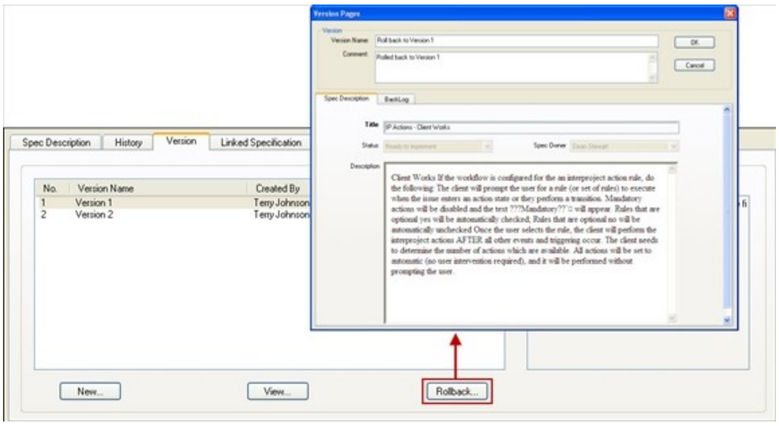
Highlight a previous version to which you want the requirement or specification to be rolled back

Click theRollbackbutton

Enter the version name, and add a comment if necessary

Click theOKbutton

The requirement or specification is rolled back to the selected version



Version Name: Roll back to Version 1

Comment: Rolled back to Version 1

Title: IP Actions - Client Works

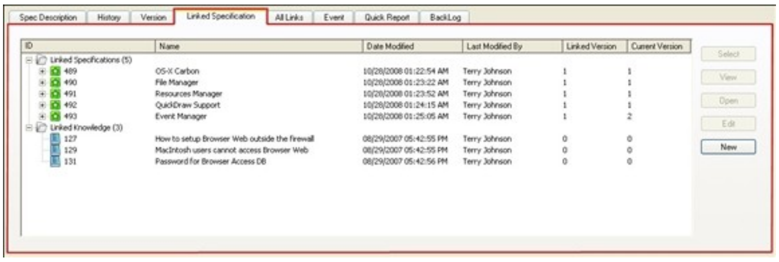
Description: Client Works If the workflow is configured for the an interproject action rule, do the following: The client will prompt the user for a rule (or set of rules) to execute when the issue enters an action state or they perform a transition. Mandatory actions will be disabled and the text ???Mandatory??'□ will appear. Rules that are optional yes will be automatically checked; Rules that are optional no will be automatically unchecked Once the user selects the rule, the client will perform the interproject actions AFTER all other events and triggering occur. The client needs to determine the number of actions which are available. All actions will be set to automatic (no user intervention required), and it will be performed without prompting the user.

1 Version 1 Terry Johnson
2 Version 2 Terry Johnson

Specification workflow can be configured such that the system automatically generates a new version when certain changes are made to a specification.

1.2.4 Linked Specification

Often there is a need to link two or more specifications, because of a relationship or dependency amongst them. Users can also link a specification to a requirement, as well as link two requirements. Such linking of requirements and specifications is done on this tab.



ID	Name	Date Modified	Last Modified By	Linked Version	Current Version
Linked Specifications (5)					
489	OS-X Carbon	10/28/2008 01:22:54 AM	Terry Johnson	1	1
490	File Manager	10/28/2008 01:23:22 AM	Terry Johnson	1	1
491	Resources Manager	10/28/2008 01:23:52 AM	Terry Johnson	1	1
492	QuickDraw Support	10/28/2008 01:24:15 AM	Terry Johnson	1	1
493	Event Manager	10/28/2008 01:25:05 AM	Terry Johnson	1	2
Linked Knowledge (3)					
127	How to setup Browser Web outside the firewall	08/29/2007 05:42:55 PM	Terry Johnson	0	0
129	MacIntosh users cannot access Browser Web	08/29/2007 05:42:55 PM	Terry Johnson	0	0
131	Password for Browser Access DB	08/29/2007 05:42:56 PM	Terry Johnson	0	0

To link the current specification to an existing specification or requirement:

Select a requirement or specification

In theDetailtab, go to theLinked Specificationtab

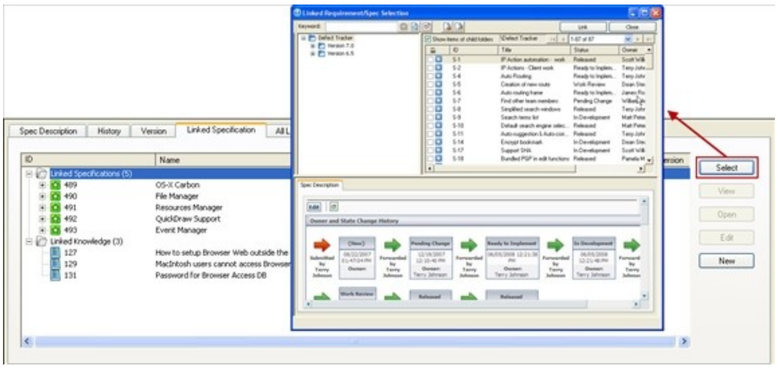
Click theSelectbutton

Highlight a folder under which the other specification is located

Select the checkbox next to the specification that is to be linked

(Multiple items can be linked at the same time by selecting multiple checkboxes)

6. Click theLinkbutton



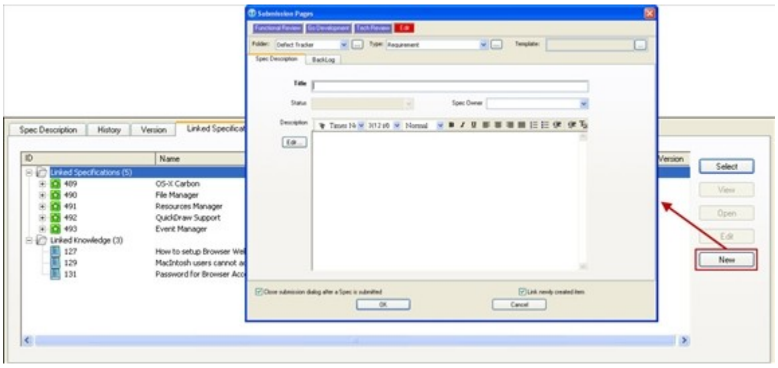
To link a new specification or requirement to the current specification:

Select a requirement or specification

In theDetailtab, go to theLinked Specificationtab

Click theNewbutton

Fill out the information to submit and link the new specification or requirement (For more information on submitting specifications, see chapter 3, section 1.1,Specification Workflow)



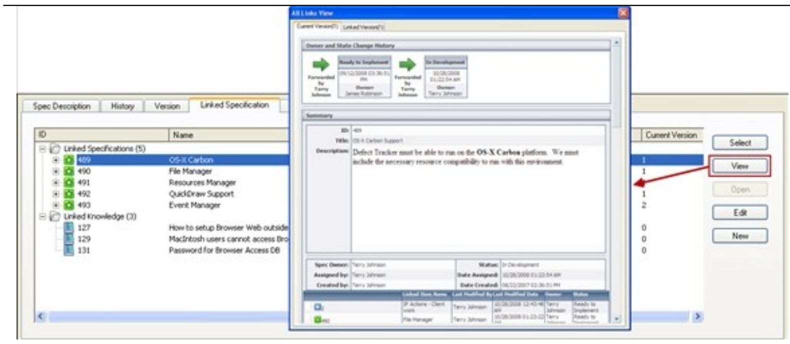
To view a linked specification or requirement:

Select a requirement or specification

In theDetailtab, go to theLinked Specificationtab

Highlight a linked specification or requirement

Click theViewbutton

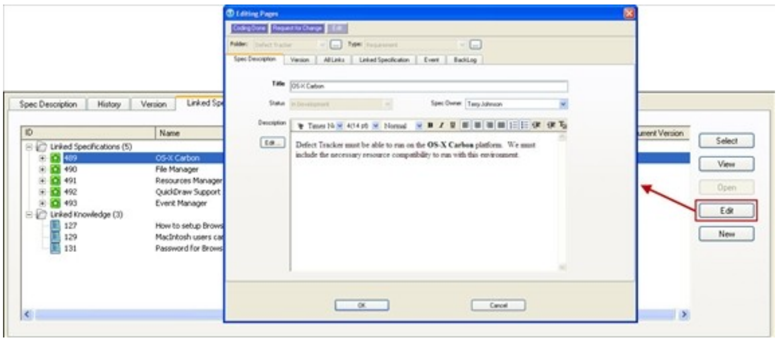


Description: Defect Tracker must be able to run on the **OS-X Carbon** platform. We must include the necessary resource compatibility to run with this environment.

ID	Name	Current Version
Linked Specifications (5)		
489	OS-X Carbon	1
490	File Manager	1
491	Resources Manager	
492	QuickDraw Support	1
493	Event Manager	2
Linked Knowledge (3)		
127	How to setup Browser Web outside	0
129	MacIntosh users cannot access Bro	0
131	Password for Browser Access DB	0

To edit a linked specification or requirement:

> ***Select a requirement or specification***
> ***In theDetailtab, go to theLinked Specificationtab***
> ***Highlight a linked specification or requirement***
> ***Click theEditbutton***



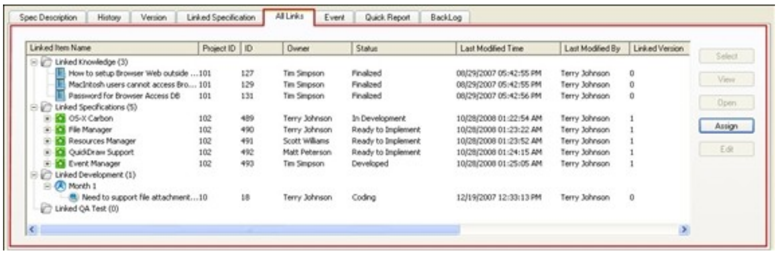
Title: OS-X Carbon

Spec Owner: Terry Johnson

Defect Tracker must be able to run on the **OS-X Carbon** platform. We must include the necessary resource compatibility to run with this environment.

1.2.5 All Links

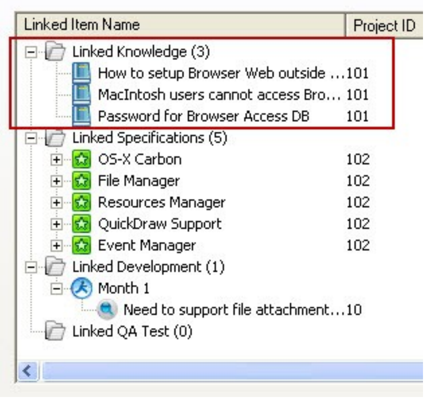
This tab is an extension of theLinked Specificationtab. It is used to link not only specifications and requirements, but also other items, such as knowledge items, development items, and test task items. Multiple knowledge items can be linked to one specification, and multiple specifications can be linked to one knowledge item. This facilitates better visibility to all items throughout DevSpec.



Linked Item Name	Project ID	ID	Owner	Status	Last Modified Time	Last Modified By	Linked Version
Linked Knowledge (3)							
How to setup Browser Web outside ...	101	127	Tim Simpson	Finalized	08/29/2007 05:42:55 PM	Terry Johnson	0
MacIntosh users cannot access Bro...	101	129	Tim Simpson	Finalized	08/29/2007 05:42:55 PM	Terry Johnson	0
Password for Browser Access DB	101	131	Tim Simpson	Finalized	08/29/2007 05:42:56 PM	Terry Johnson	0
Linked Specifications (5)							
OS-X Carbon	102	489	Terry Johnson	In Development	10/28/2008 01:22:54 AM	Terry Johnson	1
File Manager	102	490	Terry Johnson	Ready to Implement	10/28/2008 01:23:22 AM	Terry Johnson	1
Resources Manager	102	491	Scott Williams	Ready to Implement	10/28/2008 01:23:52 AM	Terry Johnson	1
QuickDraw Support	102	492	Matt Peterson	Ready to Implement	10/28/2008 01:24:15 AM	Terry Johnson	1
Event Manager	102	493	Tim Simpson	Developed	10/28/2008 01:25:05 AM	Terry Johnson	1
Linked Development (1)							
Month 1							
Need to support file attachment...	10	18	Terry Johnson	Coding	12/19/2007 12:33:13 PM	Terry Johnson	0
Linked QA Test (0)							

To link knowledge item(s):

> ***Select a requirement or specification***
> ***In theDetailtab, go to theAll Linkstab***
> ***Highlight theLinked Knowledgefolder***
> ***Click theSelectbutton***
> ***Highlight a folder under which the knowledge item is located***



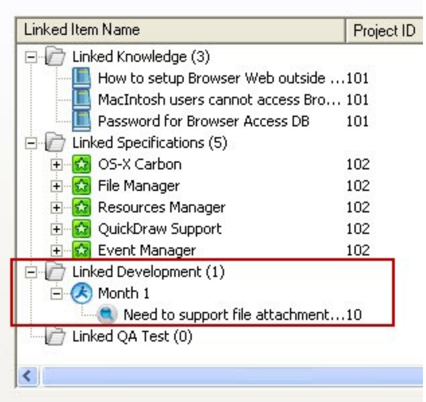
Linked Item Name	Project ID
⊟ Linked Knowledge (3)	
How to setup Browser Web outside ...	101
MacIntosh users cannot access Bro...	101
Password for Browser Access DB	101
⊟ Linked Specifications (5)	
⊞ OS-X Carbon	102
⊞ File Manager	102
⊞ Resources Manager	102
⊞ QuickDraw Support	102
⊞ Event Manager	102
⊟ Linked Development (1)	
⊟ Month 1	
Need to support file attachment...	10
Linked QA Test (0)	

Select the checkbox next to the knowledge item that is to be linked

(Multiple items can be linked at the same time by selecting multiple checkboxes)
Click theLinkbutton

To link development item(s):

Select a requirement or specification
In theDetailtab, go to theAll Linkstab
Highlight theLinked Developmentfolder
Click theSelectbutton
Highlight a folder under which the development item is located

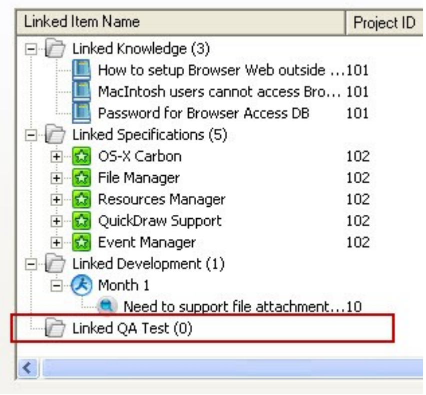


Linked Item Name	Project ID
⊟ Linked Knowledge (3)	
How to setup Browser Web outside ...	101
MacIntosh users cannot access Bro...	101
Password for Browser Access DB	101
⊟ Linked Specifications (5)	
⊞ OS-X Carbon	102
⊞ File Manager	102
⊞ Resources Manager	102
⊞ QuickDraw Support	102
⊞ Event Manager	102
⊟ Linked Development (1)	
⊟ Month 1	
Need to support file attachment...	10
Linked QA Test (0)	

Select the checkbox next to the development item that is to be linked
Click theLinkbutton

To link QA test item(s):

Select a requirement or specification
In theDetailtab, go to theAll Linkstab
Highlight theLinkedQA Testfolder
Click theSelectbutton
Highlight a folder under which the QA test item is located

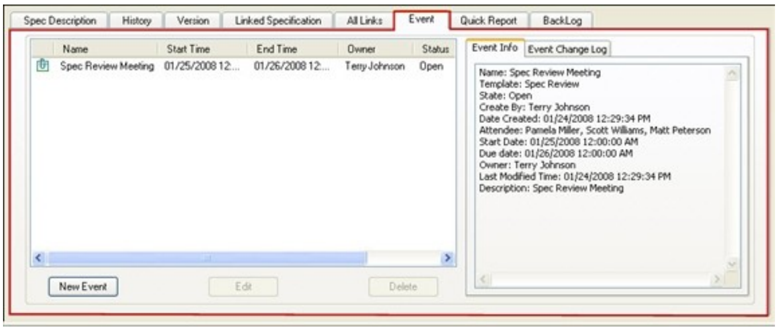


Linked Item Name	Project ID
Linked Knowledge (3)	
How to setup Browser Web outside …	101
MacIntosh users cannot access Bro…	101
Password for Browser Access DB	101
Linked Specifications (5)	
OS-X Carbon	102
File Manager	102
Resources Manager	102
QuickDraw Support	102
Event Manager	102
Linked Development (1)	
Month 1	
Need to support file attachment…	10
Linked QA Test (0)	

Select the checkbox next to the QA test item that is to be linked
Click theLinkbutton

1.2.6 Events

Events facilitate team collaboration in DevSpec. Events are tasks that need to be completed before a specification is finalized, approved and/or committed. All events for a specification are easily managed under theEventstab.



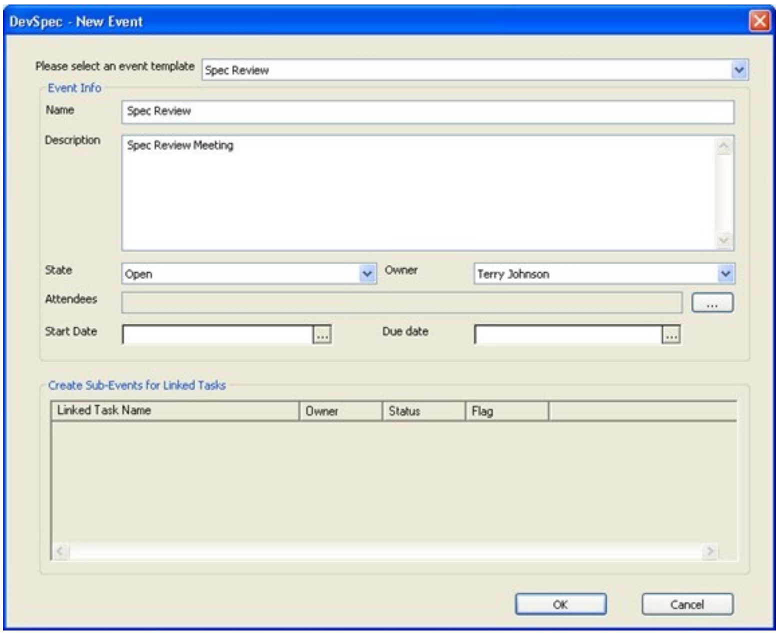
	Name	Start Time	End Time	Owner	Status
	Spec Review Meeting	01/25/2008 12:...	01/26/2008 12:...	Terry Johnson	Open

Name: Spec Review Meeting
Template: Spec Review
State: Open
Create By: Terry Johnson
Date Created: 01/24/2008 12:29:34 PM
Attendee: Pamela Miller, Scott Williams, Matt Peterson
Start Date: 01/25/2008 12:00:00 AM
Due date: 01/26/2008 12:00:00 AM
Owner: Terry Johnson
Last Modified Time: 01/24/2008 12:29:34 PM
Description: Spec Review Meeting

To create a new event:

Select a requirement or specification
In theDetailtab, go to theEventstab
Click theNew Eventbutton
Select an event template from the dropdown list

(Some examples of possible event may be design meetings, customer request discussions, or approval events)



Please select an event template: Spec Review

Event Info

Name: Spec Review

Description: Spec Review Meeting

State: Open

Owner: Terry Johnson

Attendees:

Start Date:

Due date:

Create Sub-Events for Linked Tasks

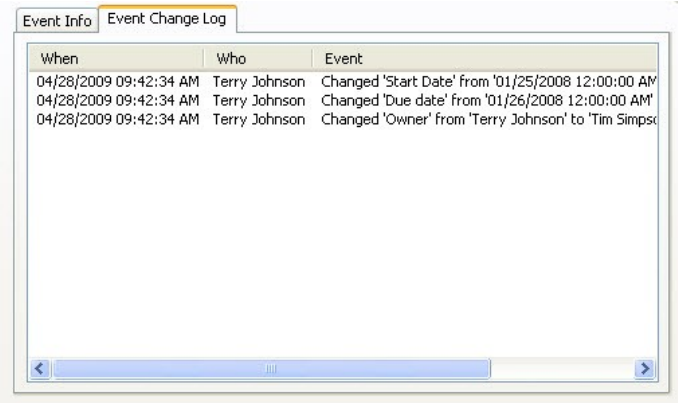
Linked Task Name	Owner	Status	Flag

Fill out the event properties, such as name, description, state, owner, start date, and due date
If the event requires other users to be notified about a meeting they should attend, click the ellipses button to

select the even attendees
Click theOKbutton

Events can also be auto-generated based on conditions defined in the Admin.

TheEvent Change Logtab under theEventstab allows you to track any changes made to the highlighted event.



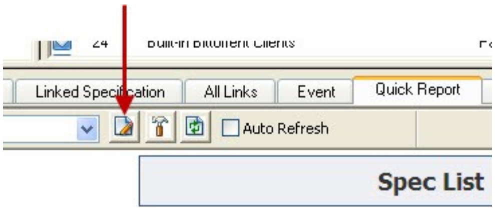
When	Who	Event
04/28/2009 09:42:34 AM	Terry Johnson	Changed 'Start Date' from '01/25/2008 12:00:00 AM
04/28/2009 09:42:34 AM	Terry Johnson	Changed 'Due date' from '01/26/2008 12:00:00 AM'
04/28/2009 09:42:34 AM	Terry Johnson	Changed 'Owner' from 'Terry Johnson' to 'Tim Simps

1.2.7 Quick Reports

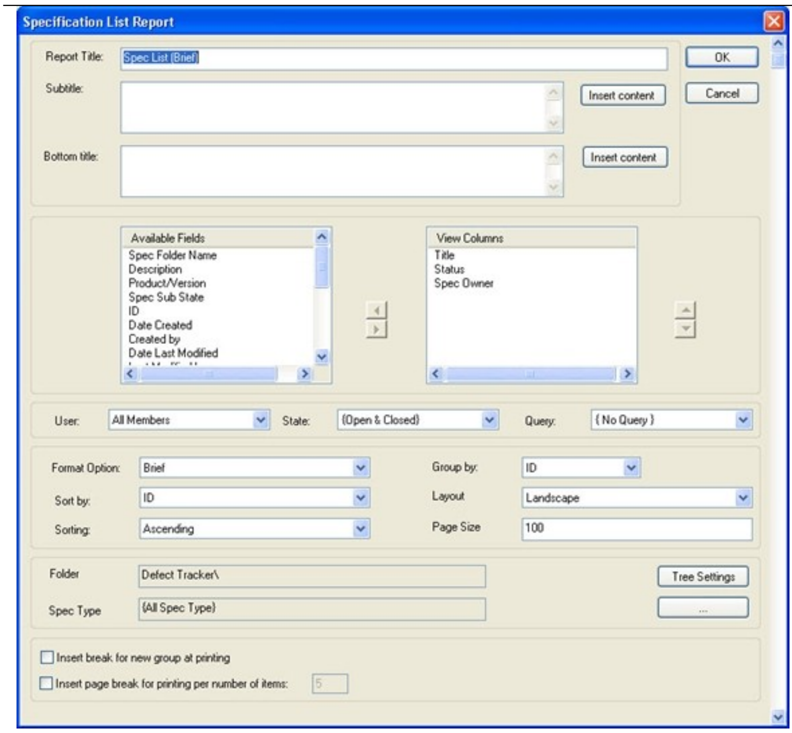
Users can quickly generate simple reports under theQuick Reporttab. Users can generate list reports or burn down reports under this tab.

List Reports
There are two types of list reports:
Brief list reports
Detailed list report

To configure a list report:
SelectSpec List (Brief)orSpec List (Detailed)in the report dropdown list underQuick Reporttab
Click the✎ button



Users can customize this report as needed
Apart from using the regular filters, such as user, state and query, to filter the items, users can also select a branch of theproduct/versiontree to filter out irrelevant items in the report
Click theTree Settingsbutton and select the applicableproduct/versionbranch



Specification List Report

Report Title: Spec List (Brief)

Subtitle:

Bottom title:

Available Fields: Spec Folder Name, Description, Product/Version, Spec Sub State, ID, Date Created, Created by, Date Last Modified

View Columns: Title, Status, Spec Owner

User: All Members

State: (Open & Closed)

Query: { No Query }

Format Option: Brief

Group by: ID

Sort by: ID

Layout: Landscape

Sorting: Ascending

Page Size: 100

Folder: Defect Tracker\

Spec Type: (All Spec Type)

□ Insert break for new group at printing

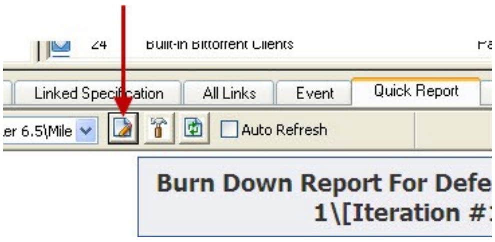
□ Insert page break for printing per number of items: 5

Burn Down Report

To configure a burn down report:

Select a burn down report in thereportdropdown list underQuick Reporttab

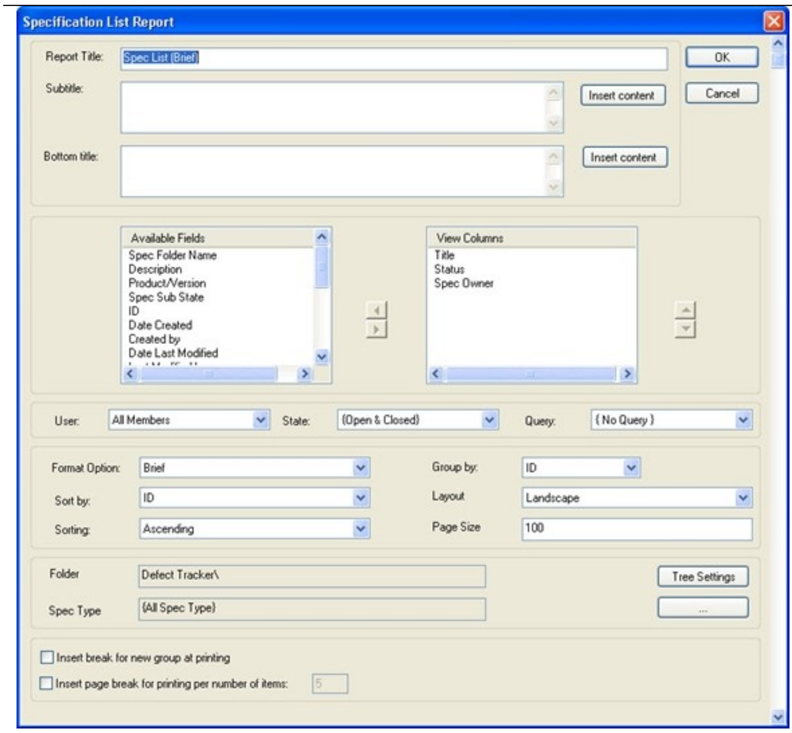
Click the button



Users can customize this report as needed

Users can also select a specific subproject or sprint in the project tree view (defined in DevPlan) to get a burn down analysis report of selected items in the project tree

Click theTree Settingsbutton and select the applicable subproject



Specification List Report

Report Title:	Spec List (Brief)
Subtitle:	
Bottom title:	

Available Fields	View Columns
Spec Folder Name	Title
Description	Status
Product/Version	Spec Owner
Spec Sub State	
ID	
Date Created	
Created by	
Date Last Modified	

User:	All Members	State:	(Open & Closed)
Query:	{ No Query }		
Format Option:	Brief	Group by:	ID
Sort by:	ID	Layout	Landscape
Sorting:	Ascending	Page Size	100
Folder	Defect Tracker\		
Spec Type	(All Spec Type)		

☐ Insert break for new group at printing

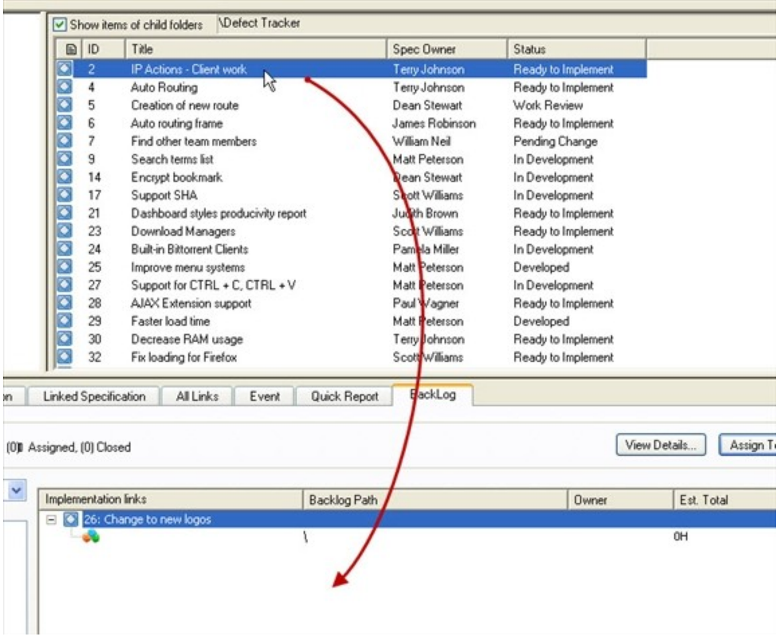
☐ Insert page break for printing per number of items: 5

1.2.8 Backlog

When a specification or requirement is finalized and approved in DevSpec, it is ready to be scheduled in a subproject to be implemented.*A product backlog is created to facilitate smooth transition through the design, planning, and implementation phases. It consists of a list of prioritized items that needs to be developed in the next implementation cycle.As specifications reach their final approval state, they are dropped into the backlog folder, indicating the project managers that these specifications are ready for implementation.*

To add a specification to the product backlog:

In theDetailtab, go to theBacklogtab
Highlight a linked specification or requirement
In the specification list view, drag any open specification and drop it into aSpecification Backlogfolder



☑ Show items of child folders \Defect Tracker

ID	Title	Spec Owner	Status
2	IP Actions - Client work	Terry Johnson	Ready to Implement
4	Auto Routing	Terry Johnson	Ready to Implement
5	Creation of new route	Dean Stewart	Work Review
6	Auto routing frame	James Robinson	Ready to Implement
7	Find other team members	William Neil	Pending Change
9	Search terms list	Matt Peterson	In Development
14	Encrypt bookmark	Dean Stewart	In Development
17	Support SHA	Scott Williams	In Development
21	Dashboard styles producivity report	Judith Brown	Ready to Implement
23	Download Managers	Scott Williams	Ready to Implement
24	Built-in Bittorrent Clients	Pamela Miller	In Development
25	Improve menu systems	Matt Peterson	Developed
27	Support for CTRL + C, CTRL + V	Matt Peterson	In Development
28	AJAX Extension support	Paul Wagner	Ready to Implement
29	Faster load time	Matt Peterson	Developed
30	Decrease RAM usage	Terry Johnson	Ready to Implement
32	Fix loading for Firefox	Scott Williams	Ready to Implement

(0) Assigned, (0) Closed

Implementation links	Backlog Path	Owner	Est. Total
26: Change to new logos			
	\		0H

1.2.9 Item Voting

DevSpec users may vote on a requirement or specification. This allows distributed teams to evaluate the impact and need for a requirement that is going to be implemented.

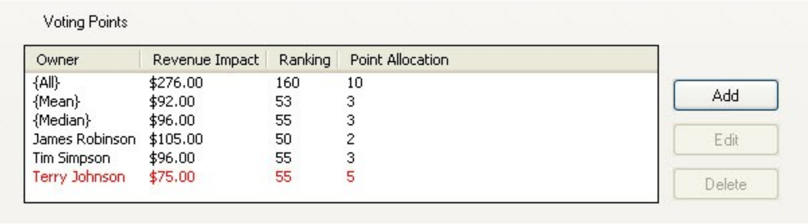
To add voting points to a specification or requirement:

Select a specification or requirement in the list view

In theDetailtab, go to theItem Votingtab

Note:The DevSuite administrator needs to turn on and configure the Voting Feature before users can add their votes.

Click theAddbutton next to theVoting Pointsfield



Voting Points

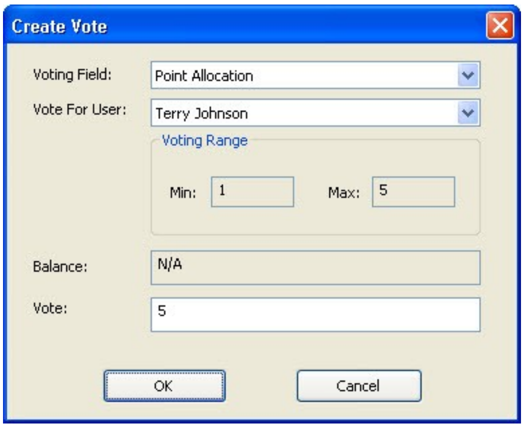
Owner	Revenue Impact	Ranking	Point Allocation
{All}	$276.00	160	10
{Mean}	$92.00	53	3
{Median}	$96.00	55	3
James Robinson	$105.00	50	2
Tim Simpson	$96.00	55	3
Terry Johnson	$75.00	55	5

From theVoting Fielddropdown list, select aVoting Type:

Revenue Impact

Ranking

Point Allocation



Create Vote

Voting Field: Point Allocation

Vote For User: Terry Johnson

Voting Range

Min: 1 Max: 5

Balance: N/A

Vote: 5

Enter the vote points as a numeric value and click theOKbutton

Voting types

Revenue Impact - Users can contribute to the evaluation of the revenue impact for a proposed requirement. The product design team, stakeholders and other DevSpec users can enter their estimates of the revenue impact towards the implementation of a suggested feature.

Ranking - Each DevSpec user can rank a requirement based on urgency, usefulness, and triviality of a requirement.

Point Allocation - Apart from ranking and revenue impact, users can also allocate points to each requirement.

The system then automatically calculates the total, mean and median for each of the previously mentioned voting options.

The DevSpec Administrator can restrict the range for voting points based on users' privileges in the system. For example, executive managers and product managers will have a larger voting range than other users who belong to marketing, sales, or service groups.

2 Advanced Usage

2.1 Specification Folder Tree

The DevSpec folder tree is a hierarchical structure composed of multiple folders and subfolders that organize specifications into distinct areas of work. Project members with the appropriate privileges may create any number of nested specification folders to any level of depth. Every specification category and subcategory is represented by a specification folder. Specifications may be grouped by specification class, stakeholder, functional area, or any other category that is useful to a business.

The tree allows users to view subsets of work items in the list panel. Users can view just the work items in the selected folder or the work items contained within descendant folders. In addition, the DevSpec folder structure allows users to define a set of access rules to secure the contained work items, define applicable work item owners, and sort the folder tree. All changes made to the DevSpec folders are stored in the built-in change log easily accessible by the appropriate manager.

Creating a Folder

To create a folder, right-click on an existing folder and clickNew Folder:
SelectNew Child Folderto create a subfolder underneath the current folder.

SelectNew Sibling Folder Aboveto create a folder in the same level but above the current folder.

SelectNew Sibling Folder Belowto create a folder in the same level but below the current folder.

*Refer to theFolder Propertiessection.

Copying/Moving Folders

Users can create an identical set of folders and its specifications to another folder from the current selected folder.

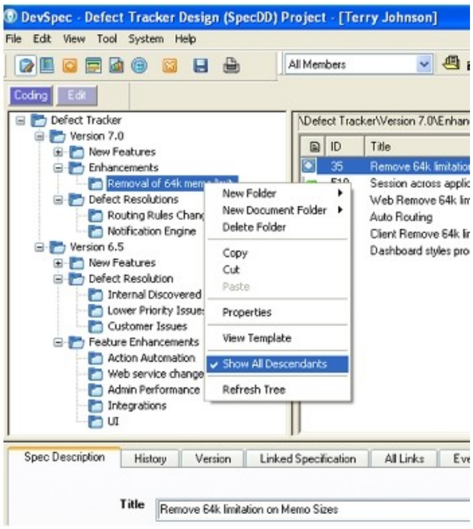
1. Right-click on a folder and selectCopy.
2. Right-click on a destination folder and selectPaste.

Users can also choose to move a folder and its contents to a different directory.

1. Right-click on a folder and selectCut.
2. Right-click on a destination folder, and selectPaste

Filtering Work Items

To filter work items, begin by selecting a folder in the tree panel.



Title: Remove 64k limitation on Memo Sizes

To display or hide work items in descendent folder(s):

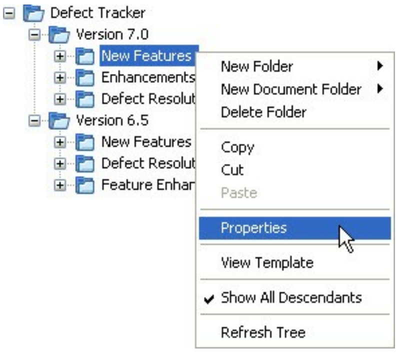
1. ***Right-click a work item folder in a tree panel.***
2. ***Define the scope of the folder filter:***

To display the work items in that folder and the work items of every descendent folder, select theShow All Descendentcommand. (Alternatively, check theshow items of child folderscheckbox on the top-left corner of the list panel.)

To display the work items in that folder and that folder only, deselect theShow All Descendentcommand.

2.1.1 Folder Properties

To access folder properties, right-click on the folder and selectProperties.



Folder Description

This section allows users to accurately describe the details of a folder.

1. ***Provide an accurate folder name so that work items can be easily filtered.***
2. ***Provide a folder status to indicate whether or not this folder should still be in use.***
3. ***Provide a priority value.***

****Administrators can add custom fields and pages to track additional details.***

Access Control:

This section allows users with sufficient privileges to secure the contents within a DevSpec folder and the folder itself.

Managers can select 1 out of 3 different folder access types. All folder access types are split into two panes. This allows the manager to view the permissions defined by the administrator for folders and work items within the folders.

Public Folder:
A set of account types defined in the Admin.

No Access – users will not be able to see existing folder/work items

Read-Only – users cannot update existing folder/work items

Can Edit – users can only update existing folder/work items

Can Create and Edit – users can submit new folder/work items

Can Delete, Create and Edit – users can submit new folder/work items as well as delete existing items
Private Folder:
A second set of account types defined in the Admin.
No Access – users will not be able to see existing folder/work items

Read-Only – users cannot update existing folder/work items

Can Edit – users can only update existing folder/work items

Can Create and Edit – users can submit new folder/work items

Can Delete, Create and Edit – users can submit new folder/work items as well as delete existing items
Secured Folder:
This folder access type is used if the public / private folder access types are not sufficient. Administrators can

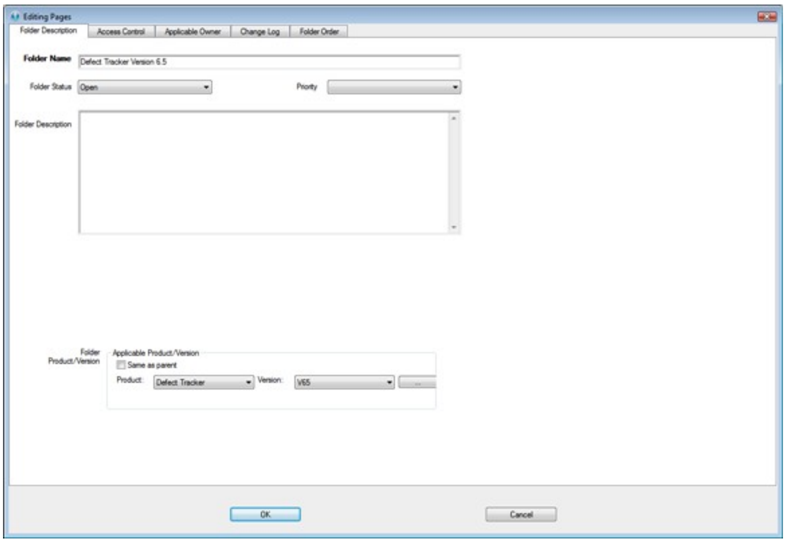
define different sets of custom access levels for account types and team groups that can be applied to any folder. This is beneficial if privileges may need to be changed later.

To view privileges for each access type, left-clickview access type.

In addition, individual users can also be added as an exception to the account type and team group privileges defined in the access level:

1. ***To add a user, clickAdd User, select the user(s) and give applicable privileges.***
2. ***To remove user(s) from the access level, clickRemove User.***

****Check offsame as parentto inherit the access control from the parent folder***



Folder Name: Defect Tracker Version 6.5

Folder Status: Open

Priority:

Folder Description:

Folder Product/Version

Applicable Product/Version

□ Same as parent

Product: Defect Tracker

Version: V65

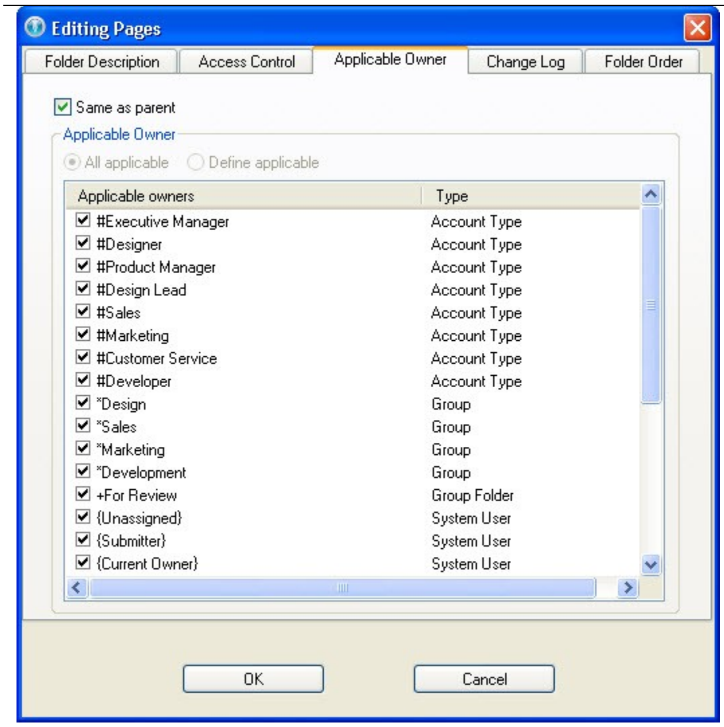
Applicable Owner:

This section allows managers to define account types, groups and individuals that can own a work item in the folder. Users that do not belong in the account types or groups defined here cannot be selected as a specification owner, even though the workflow permits them.

SelectAll Applicableto quickly allow all DevSpec users to be able to own work items if the workflow permits them.

SelectDefine Applicableto define specific account types, groups and users to be able to own work items if the workflow permits them.

****Check offsame as parentto inherit the access control from the parent folder***



☑ Same as parent

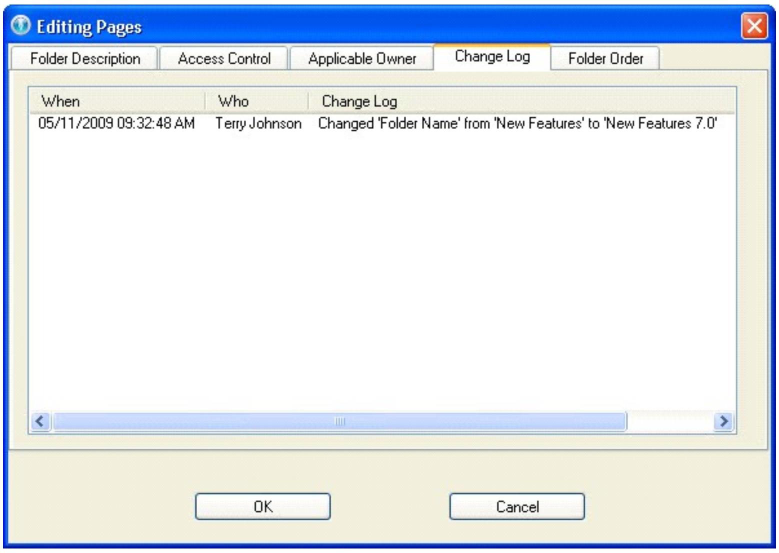
Applicable Owner

● All applicable ○ Define applicable

Applicable owners	Type
☑ #Executive Manager	Account Type
☑ #Designer	Account Type
☑ #Product Manager	Account Type
☑ #Design Lead	Account Type
☑ #Sales	Account Type
☑ #Marketing	Account Type
☑ #Customer Service	Account Type
☑ #Developer	Account Type
☑ *Design	Group
☑ *Sales	Group
☑ *Marketing	Group
☑ *Development	Group
☑ +For Review	Group Folder
☑ {Unassigned}	System User
☑ {Submitter}	System User
☑ {Current Owner}	System User

Change Log:

This section allows managers to easily see accurate information on all changes performed on the folder. The date, the user who performed the change, and the change description are displayed.



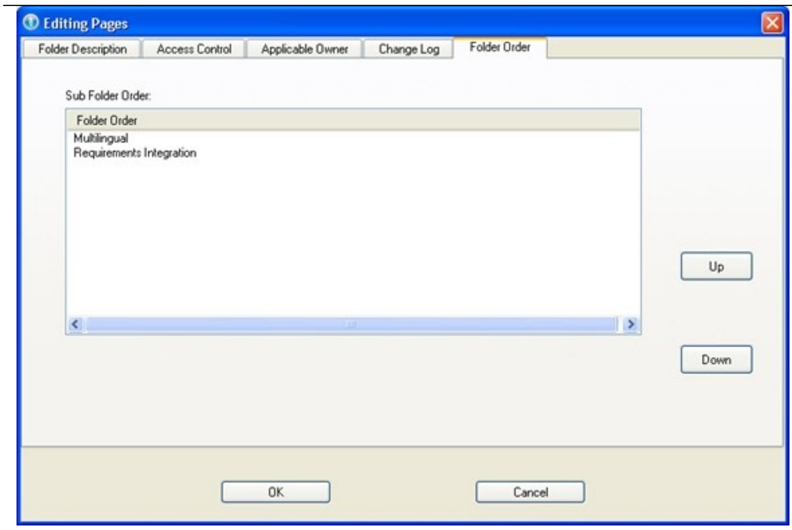
When	Who	Change Log
05/11/2009 09:32:48 AM	Terry Johnson	Changed 'Folder Name' from 'New Features' to 'New Features 7.0'

Folder Order:

This section allows managers to be able to sort the subfolders underneath the selected folder.

Click the up button to move a subfolder higher in the tree.

Click the down button to move a subfolder lower in the tree.



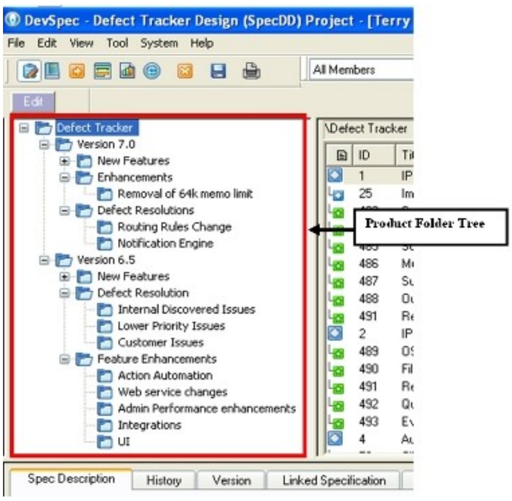
Sub Folder Order:

Folder Order
Multilingual
Requirements Integration

2.2 Product/Version Folder Tree

A product tree filter is a simple query that returns work items that are associated with a specific product, version, or build and displays those work items in the list panel - all other work items are "filtered out" and not displayed.

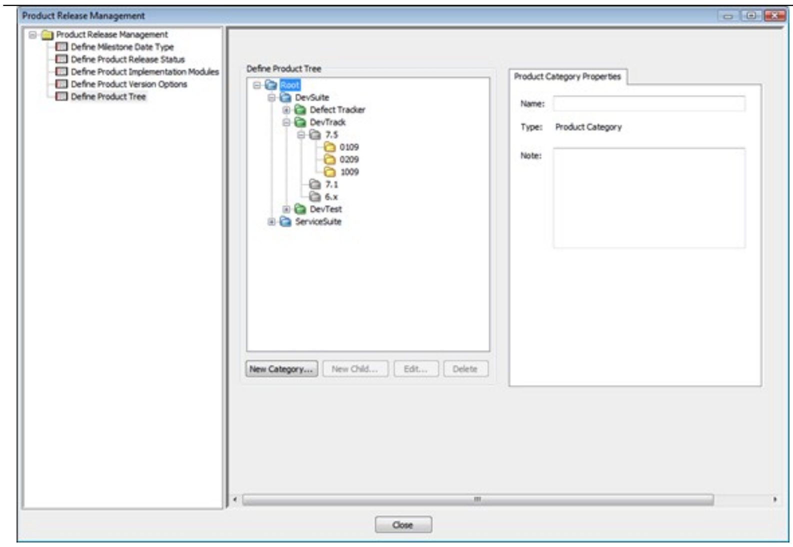
Project members may use the product tree in conjunction with the DevSpec folder tree to filter work items. All work items (requirements, specifications, and change requests) may be defined as "applicable" to a select set of products, versions, or builds.



Defect Tracker
- Version 7.0
  - New Features
  - Enhancements
    - Removal of 64k memo limit
  - Defect Resolutions
    - Routing Rules Change
    - Notification Engine
- Version 6.5
  - New Features
  - Defect Resolution
    - Internal Discovered Issues
    - Lower Priority Issues
    - Customer Issues
  - Feature Enhancements
    - Action Automation
    - Web service changes
    - Admin Performance enhancements
    - Integrations
    - UI

Maintaining Products/Versions

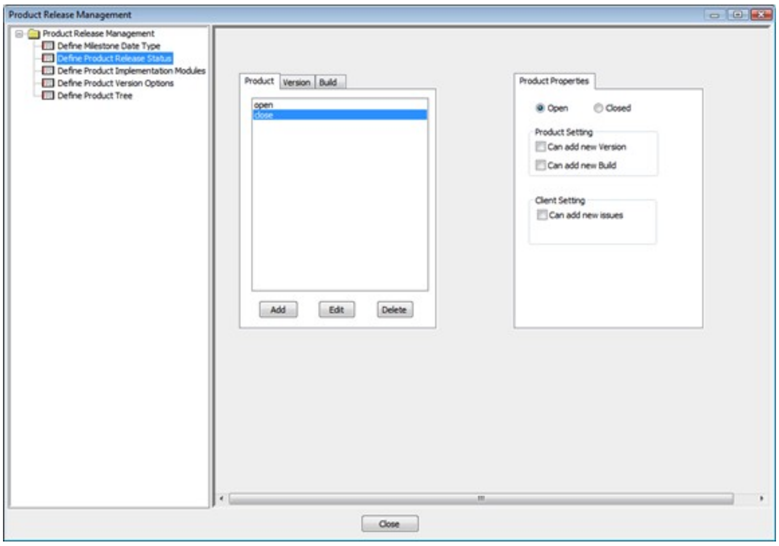
The product / version tree is organized into a hierarchy of 5 types of folders. The root folder represents the highest level and consists of all products that are tracked in the current project. The category folder is defined by all applicable products within the category. The product folder, 3rdin the hierarchy, contains the different versions and builds that are applicable.


Product Release Management
Product Release Management
Define Milestone Date Type
Define Product Release Status
Define Product Implementation Modules
Define Product Version Options
Define Product Tree
Define Product Tree
Root
DevSuite
Defect Tracker
DevTrack
7.5
0109
0209
1009
7.1
6.x
DevTest
ServiceSuite
New Category...
New Child...
Edit...
Delete
Product Category Properties
Name:
Type:
Product Category
Note:
Close

To begin creating products and versions, users must go toTool>Product Release Management.

1. ***ClickDefine Product Tree.***
2. ***Highlight theRootfolder (if there are no existing product versions). Click onNew Categoryto create a new category underneath the root.***
3. ***Highlight the newly created category and click onNew Productto create a new product under the root.***
4. ***Highlight the newly created product and click onNew Versionto create a new version under the product.***
5. ***Highlight the newly created version and click onNew Buildto create a new build under the version.***

For each folder type, you can define the properties by clicking on the folder. You can add a note to the properties to provide a description for the folder. For the product/version/build folder types, you can also add a status to the properties of the folder. The status indicates whether or not users can add additional products/versions/ build, and also allows users to be able to add additional issues to the product/version.



To define the status:

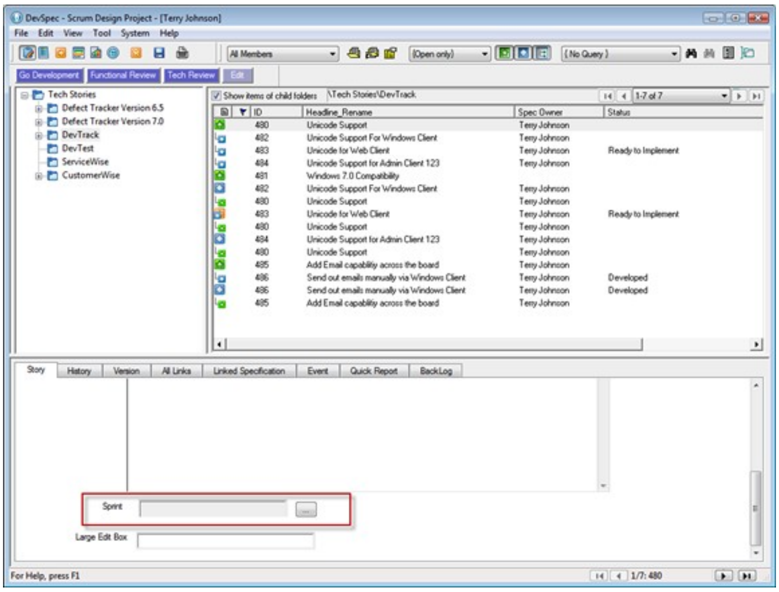
1. ***ClickDefine Product Release Statusin theProduct Release Managementdialogue.***
2. ***Click on the product tab and clickAdd. This will bring up a dialogue box for the user to input a new status name.***
3. ***To the left of the newly created status, users can indicate whether or not this status is open or closed. Check offcan add new version,canadd newbuild, andcan add new issueswhere appropriate.***
4. ***ClickDefine Product Treein theProduct Release Managementdialogue.***
5. ***Click on a product, version, or build folder, and provide the status.***

To rename a folder, click on the edit button to rename a folder. To delete a folder, click on the delete button. Users can only delete folders that have not yet been associated with any work item.

Note: The Product Release Management module is useable with the correct permissions given to the user in the DevSuite Admin.

Associating Product/Version

Prior to filtering by products and versions, users must associate work items with the correct product and version.



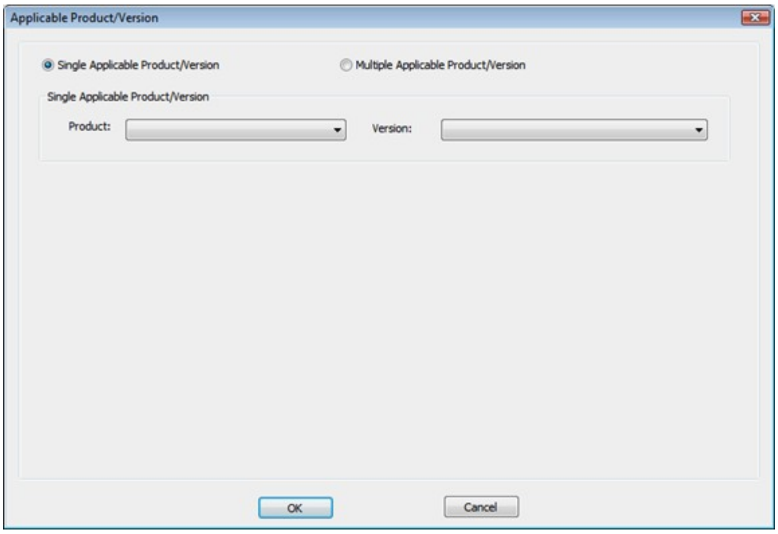
☑ Show items of child folders \Tech Stories\DevTrack 1-7 of 7

ID	Headline_Rename	Spec Owner	Status
480	Unicode Support	Terry Johnson	
482	Unicode Support For Windows Client	Terry Johnson	
483	Unicode for Web Client	Terry Johnson	Ready to Implement
484	Unicode Support for Admin Client 123	Terry Johnson	
481	Windows 7.0 Compatibility		
482	Unicode Support For Windows Client	Terry Johnson	
480	Unicode Support	Terry Johnson	
483	Unicode for Web Client	Terry Johnson	Ready to Implement
480	Unicode Support	Terry Johnson	
484	Unicode Support for Admin Client 123	Terry Johnson	
480	Unicode Support	Terry Johnson	
485	Add Email capability across the board	Terry Johnson	
486	Send out emails manually via Windows Client	Terry Johnson	Developed
486	Send out emails manually via Windows Client	Terry Johnson	Developed
485	Add Email capability across the board	Terry Johnson	

To assign a product/version to a work item:

1. ***Click on the work item.***
2. ***Click on the product/version ellipse button.***

Note: If you do not see the product/version field, please contact the administrator to add this field into the GUI.

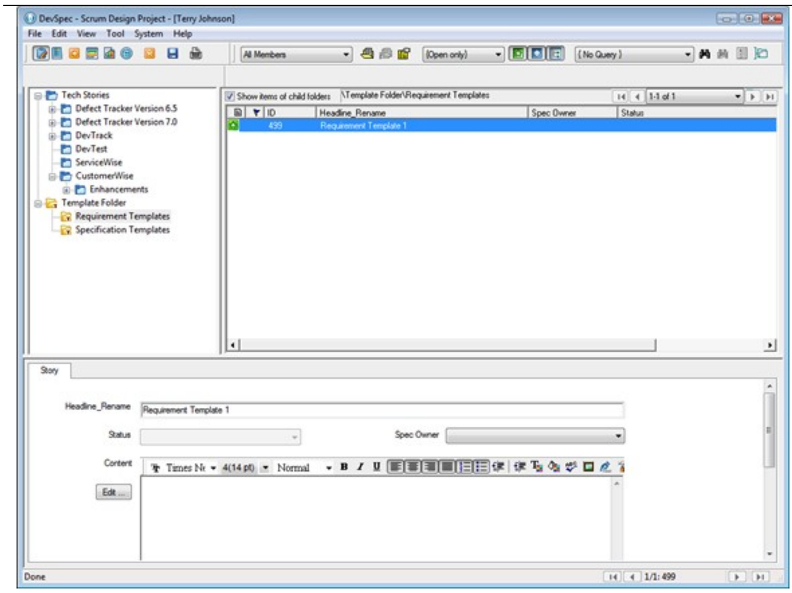

Applicable Product/Version
● Single Applicable Product/Version
○ Multiple Applicable Product/Version
Single Applicable Product/Version
Product:
Version:
OK
Cancel

3. ***Click the radio button to indicate whether or not this item is applicable to a single product/version or multiple applicable products/versions.***
4. ***Select the applicable product(s) and version(s).***

Note: If you cannot select the product or version, please check with the administrator to make sure the created products/versions are listed in the product tree, and are defined as applicable to use for the current project.

Template Folders

In DevSpec, users can define templates to use when creating a new work item. A template is used to pre-fill all the required fields during the submission of a work item.



\Template Folder\Requirement Templates

ID	Headine_Rename	Spec Owner	Status
499	Requirement Template 1		

Headine_Rename: Requirement Template 1

Status:

Spec Owner:

Content:

To begin:

1. ***Right-click on a folder in the folder tree.***
2. ***SelectView Template. The template folder structure will appear.***
3. ***Right-click on the template folder to create a new child folder***
4. ***Highlight the template folder in which you want to create templates.***
5. ***Right click on the list pane, and selectNew.***
6. ***Define all applicable fields for the template.***
7. ***ClickOKto finish.***

Please refer to chapter 3 section 3.1.1 on more information on how to create new specifications

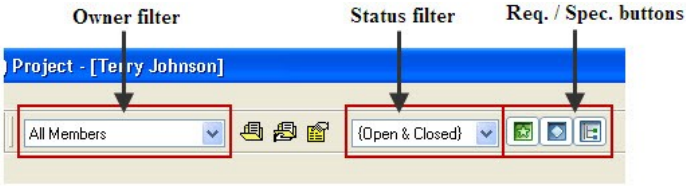
Chapter 4 - Searches and Queries

With a large collection of specifications and requirements, it is imperative to users that they are able to quickly and easily find exactly what they need. With DevSpec's rich searching abilities, users can apply quick filters, define searches, and use queries to do just that.

1 Quick Filters

In the specification view the user can apply filers and queries on work items. To work effectively, and to minimize the time needed to review large numbers of records, project members must be able to quickly identify relevant work items based on key indicators, such as item owner, status, or type (i.e. requirement or specification).

In DevSpec, a filter is a simple query that returns work items matching a defined set of criteria. Records matching the criteria are displayed in the list panel.

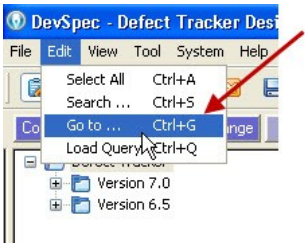


1.1 Go To

The*Go to*feature may be used to quickly filter work items by their ID numbers.

1. Open the*Go To*dialog by doing one of the follow:

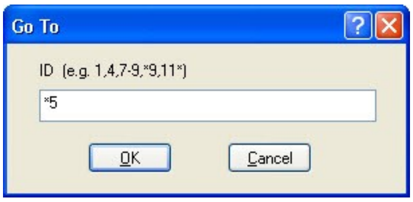
From the menu bar, select*Edit*>*Go to…*



Press Ctrl + G.

2. Type an ID number in the text box.

3. Press the*OK*button.



The*Go to*feature also comes equipped with different operators to expand the search.

A comma (,) or a space may be used as separators for multiple single IDs. For example, inputting*2,3,4*would return items 2, 3, and 4.
A hyphen (-) may be used for an interval of IDs. For example, inputting*5-8*would return items 5, 6, 7, and 8.
An asterisk (*) may be used to denote wildcard characters; that is, any string of numbers. For example, inputting**5*would return items: 5, 15, 25 … 105, etc.

Note: Multiple operators may be used in a single search, as shown in the example provided in the*Go to*dialog.

1.2 Filtering by Owner

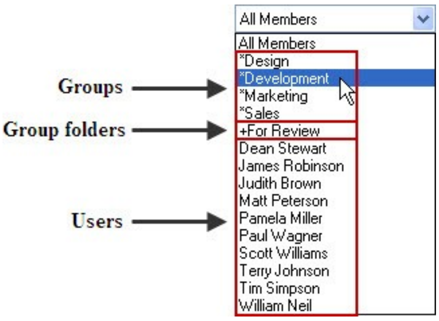
In DevSpec, all specifications, requirements, and change requests are at all times owned by one-and-only-one user or group folder. Using the owner filter, the work items displayed in the list panel may be filtered by the current owner.

The owner filter in the search bar displays the names of all users, groups, and group folders.

Users: Users are the individual project team members. When a user is selected, all items owned by that user are displayed. Users are identified by the users' names.

Groups: Groups are administrator-defined teams of users who share a common set of responsibilities within a project. When a group is selected, all items owned by any individual belonging to that group are displayed. Groups are identified by a leading asterisk (*).

Group folders: Group folders give groups the ability to own items, while preserving a set of access rights within the group. Group folder ownership and user ownership are exclusive; therefore, when a group folder is selected, all items owned by that group folder, and not any other user, are displayed. Group folders are identified by a leading addition sign (+).

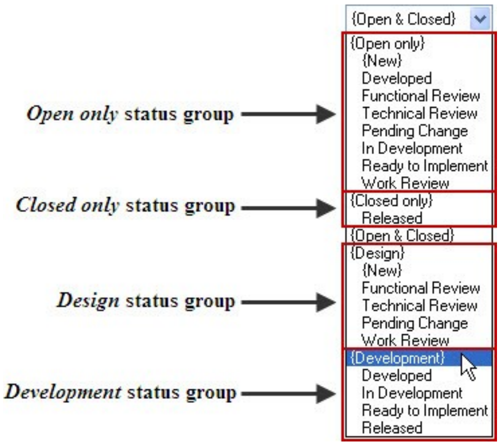


In the above screenshot, where the*Development*group is selected, only the work items currently being owned by that group will be returned in the search results.

1.3 Filtering by Status

In DevSpec, all work items, specifications, requirements, and change requests, are defined by their workflow status and whether that status is opened or closed.

Using the status filter, the work items displayed in the list panel may be filtered by the current status. The status filter in the search bar displays every workflow state that is applicable to the work items managed in the view.



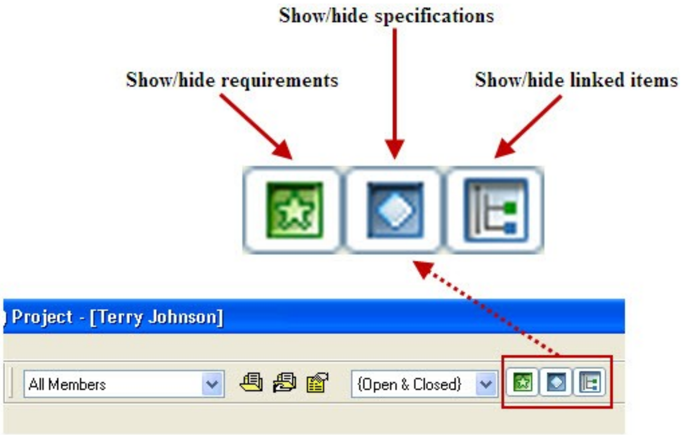
Status groups: The status control may also be used to filter multiple common statuses, a status group. These are identified by the statuses which are not indented in the status list, and are enclosed by braces ({…}).

In the above screenshot,*Open only*,*Closed only*,*Design*and*Development*are defined as status groups. Also in the example, where the*Development*status group is selected, only the items in that status group (i.e. the statuses listed thereunder) will be returned in the search results.

***Note**: All DevSpec projects have the predefined status groups,OpenandClosed, as all workflow states are defined as such. Any further status groups must be set up by the project administrator in the Admin.*

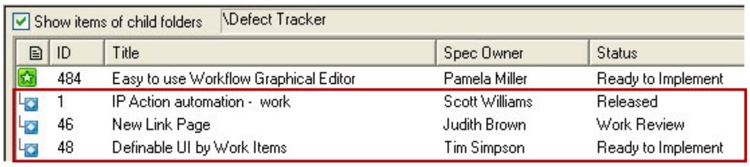
1.4 Filtering by Work Item Type

Using the requirement and specification buttons in the tool bar, work items may be filtered by their type, displaying only requirements, only specifications, or both. Linked requirements and specifications may also be shown or hidden in the list panel.



In the example above, all three buttons are shown as clicked. When not clicked, the buttons are grayed-out. At least one item type is always displayed. It is not possible to uncheck both the requirement and specification buttons.

When linked items are shown in the list panel, they are displayed as child items to each parent item.



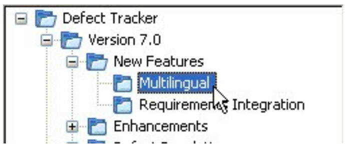
☑ Show items of child folders \Defect Tracker

	ID	Title	Spec Owner	Status
	484	Easy to use Workflow Graphical Editor	Pamela Miller	Ready to Implement
	1	IP Action automation - work	Scott Williams	Released
	46	New Link Page	Judith Brown	Work Review
	48	Definable UI by Work Items	Tim Simpson	Ready to Implement

1.5 Filtering Using the Folder Tree

Using the folder tree as a filter, the user may view subsets of work items in the list panel. Only the work items contained in the selected work item folder are displayed in the list panel.

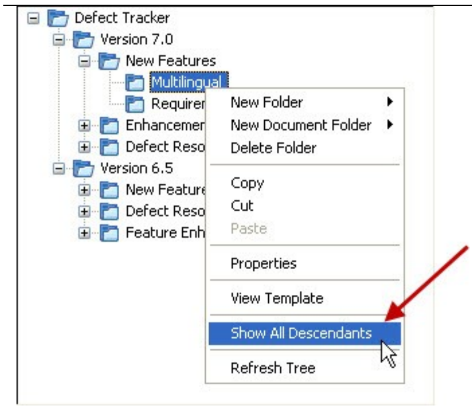
Select a folder in the folder tree to filter the work items displayed in the list panel.



To display or hide work items in the descendent folder(s), follow one of the following methods:

1. Right-click on a folder in the folder tree, and choose the scope of the folder tree filter:

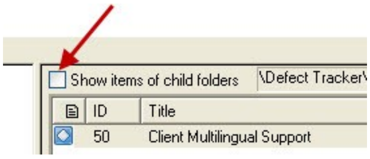
To display the work items in that folder and the work items of every descendant folder, select theShow All Descendentsoption.
To display the work items in that folder and that folder only, deselect theShow All Descendantsoption.



OR

2. Highlight a folder in the folder tree, and choose the scope of the folder tree filter:

To display the work items in that folder and the work items of every descendant folder, check theShow items of childfolders box.
To display the work items in that folder and that folder only, uncheck theShow items of childfolders box.



Organizing and managing work items in the folder tree:

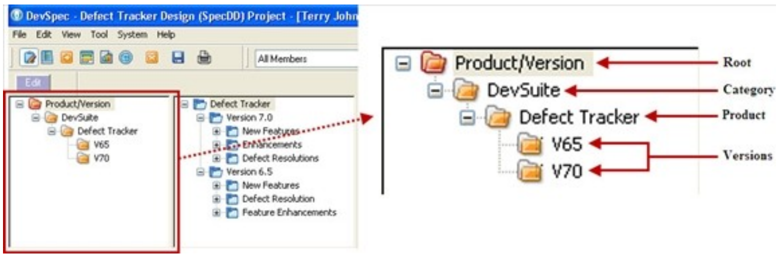
Folder tree management is the task of defining the structures that organizes requirements and specifications in a DevSpec project. All DevSpec items are stored and managed in the folder tree.

The folder tree is a hierarchical structure composed of multiple folders and subfolders that organize items into distinct areas of work. Project members with the appropriate privileges may create any number of nested folders to any level of depth. Each folder is defined by a set of access rights, applicable products/version rules, and applicable owner rules.

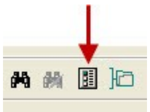
Each category and subcategory is represented by a folder. Items may be grouped by item class, stakeholder, functional area, or any other category that is useful to a business. However, TechExcel recommends that the folder structure is not organized by product, version, and build; this is done by the product version tree.

1.6 Filtering Using the Product Version Tree

As an extension of the folder tree, the product version tree may also be used to further filter work items. The product version tree structure represents all categories, products, versions, and builds throughout the entire application lifecycle.



To view the product version tree, click theShow Product Version Treebutton in the tool bar. Click it again to hide.

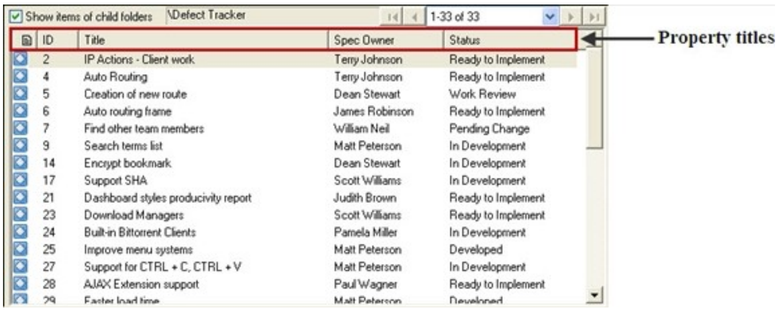


Note: To use the product version tree to filter work items, items must be defined with the product/version property. This is not a default item property, so this must first be set up by the project administrator.

1.7 Sorting

The results of a filter are displayed in the list panel. To find all relevant information quickly, work items may be sorted by its properties. Depending on the data type, this will sort a particular property alphabetically or numerically.

To sort by a work item property, click on its title in the header section.



☑ Show items of child folders \Defect Tracker 1-33 of 33

ID	Title	Spec Owner	Status
2	IP Actions - Client work	Terry Johnson	Ready to Implement
4	Auto Routing	Terry Johnson	Ready to Implement
5	Creation of new route	Dean Stewart	Work Review
6	Auto routing frame	James Robinson	Ready to Implement
7	Find other team members	William Neil	Pending Change
9	Search terms list	Matt Peterson	In Development
14	Encrypt bookmark	Dean Stewart	In Development
17	Support SHA	Scott Williams	In Development
21	Dashboard styles producivity report	Judith Brown	Ready to Implement
23	Download Managers	Scott Williams	Ready to Implement
24	Built-in Bittorrent Clients	Pamela Miller	In Development
25	Improve menu systems	Matt Peterson	Developed
27	Support for CTRL + C, CTRL + V	Matt Peterson	In Development
28	AJAX Extension support	Paul Wagner	Ready to Implement
29	Faster load time	Matt Peterson	Developed

The first time a property is sorted, the data is displayed in ascending order (A to Z). If clicked a second time, the data will be displayed in descending order (Z to A). An arrow will arrear next to the title of the property that is being sorted. This denotes ascending or descending order.

△ Ascending order

▽ Descending order

The user may customize which work item properties are displayed in the list panel and also define a default property by which to sort on login. For more information, see chapter 2, section 6, DevSpec Preferences.

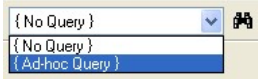
Note: Text fields, such as item description, are not sortable.

2 Defining Search

There are different ways to searching for an item in DevSpec. Users can use the quick filters (described in the previous section) to filter out or search for items based on the item owner and/or item status. Users can also utilize the extensive search functions available in DevSpec.

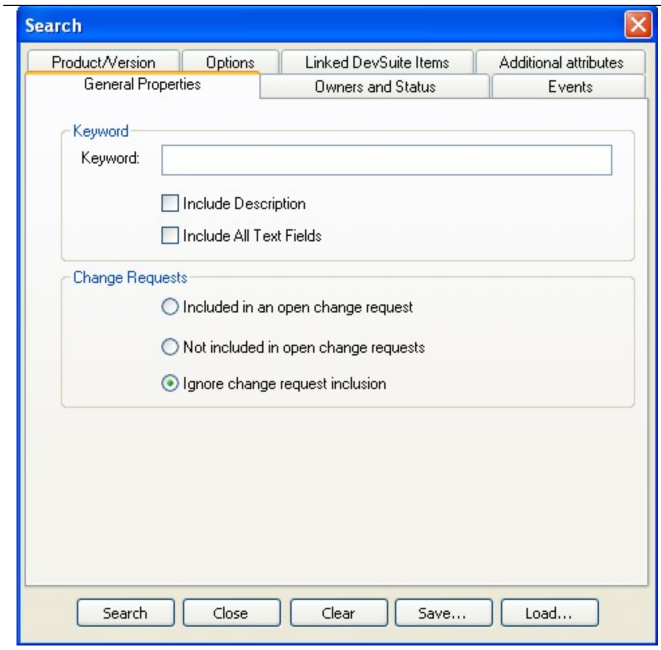
To launch the search dialog in DevSpec:

Click the button on the tool bar (or)
Select {Ad-hoc Query} from the dropdown list as shown below (or)



Select Edit > Search from the menu bar (or)
Press CTRL + S

The Search dialog is displayed:



Keyword

Keyword:

☐ Include Description

☐ Include All Text Fields

Change Requests

○ Included in an open change request

○ Not included in open change requests

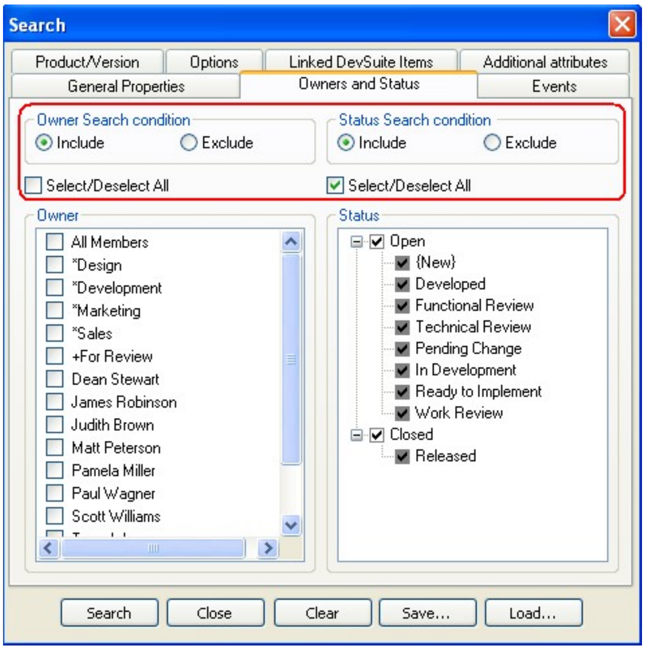
● Ignore change request inclusion

The Search dialog consists of up to seven different pages, represented by tabs. Users can define unique search condition(s) within each tab, and even combine conditions defined on multiple tabs, and execute as one final search.

Common search options

There are a few search options that are common across most of the tabs on the search dialog.

The Select/Deselect All checkbox enables project members to quickly select or deselect all field values displayed in the field value list.



Owner Search condition: ● Include ○ Exclude

☐ Select/Deselect All

Status Search condition: ● Include ○ Exclude

☑ Select/Deselect All

Owner

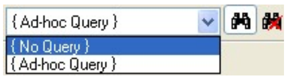
- ☐ All Members
- ☐ *Design
- ☐ *Development
- ☐ *Marketing
- ☐ *Sales
- ☐ +For Review
- ☐ Dean Stewart
- ☐ James Robinson
- ☐ Judith Brown
- ☐ Matt Peterson
- ☐ Pamela Miller
- ☐ Paul Wagner
- ☐ Scott Williams

Status

- ☑ Open
  - ☑ {New}
  - ☑ Developed
  - ☑ Functional Review
  - ☑ Technical Review
  - ☑ Pending Change
  - ☑ In Development
  - ☑ Ready to Implement
  - ☑ Work Review
- ☑ Closed
  - ☑ Released

The Include option returns only those records that contain the selected field values in the search field.
The Exclude option returns only those records that do not contain the selected field values in the search field.
To cancel a search:

Click the button on the tool bar (or)
Select {No Query} from the dropdown list as shown below


{ Ad-hoc Query }

{ No Query }
{ Ad-hoc Query }

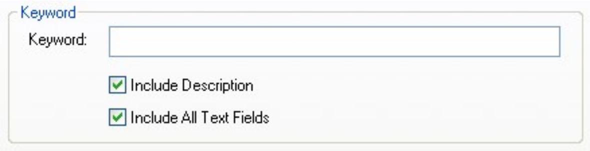
2.1 Search by Keyword

In DevSpec, a keyword is a term (word, phrase, or alphanumeric string) that is used as a search condition in a query. The DevSpec search engine searches for instances of a keyword in a record set and returns those records in which the keyword is found.

Using the DevSpec Search feature, the user may search for requirements, specifications, knowledge items and change requests based on the text strings found in text fields. These text fields include title, description, history, note title, note description, link comments, event title, and event description fields.

To search using keywords:

Launch the search dialog in DevSpec (use button).
Go to the General Properties tab.
Enter one or more keywords in the keyword text box. The search engine can search for keywords in the description control, and other multiple line text box controls as well. By default the keyword search is executed against the title field of an item.



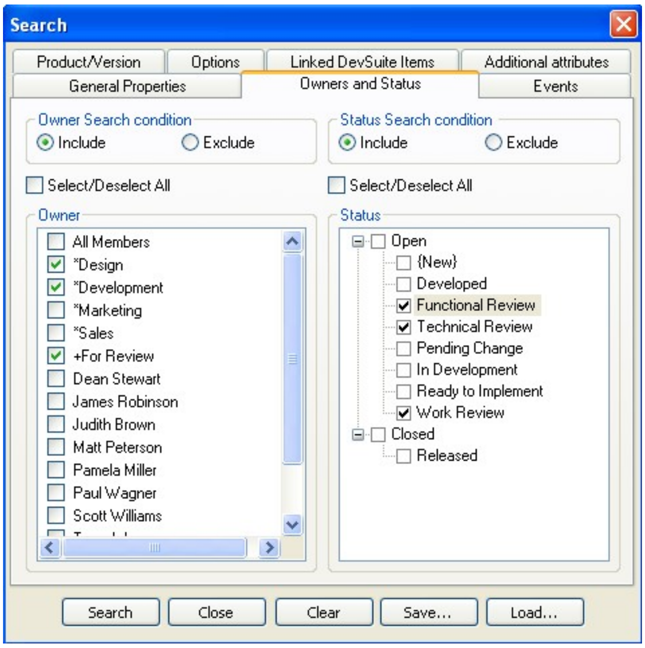
Keyword
Keyword:
☑ Include Description
☑ Include All Text Fields

Check the Include Description checkbox to search for a keyword in the description field of an item.
Check the Include All Text Fields checkbox to search for a keyword in any text/memo field (defined by the DevSpec Administrator).

2.2 Search by Owner and Status

This search is an extension of the owner and status filters in the tool bar. Unlike these dropdown lists, users can select more than one owner and/or status to filter specifications.

To define owner and status search condition:
Launch the search dialog in DevSpec (use button).
Go to the Owner and Status tab.
Select one or more owners and/or workflow states.



Owner Search condition: ● Include ○ Exclude
Status Search condition: ● Include ○ Exclude

□ Select/Deselect All
□ Select/Deselect All

Owner
□ All Members
☑ *Design
☑ *Development
□ *Marketing
□ *Sales
☑ +For Review
□ Dean Stewart
□ James Robinson
□ Judith Brown
□ Matt Peterson
□ Pamela Miller
□ Paul Wagner
□ Scott Williams

Status
□ Open
□ {New}
□ Developed
☑ Functional Review
☑ Technical Review
□ Pending Change
□ In Development
□ Ready to Implement
☑ Work Review
□ Closed
□ Released

Click the Search button.

In the above example, the resulting list would be items currently owned by any user in the Design group, Development group, or For Review group folder, and which are currently in the Functional

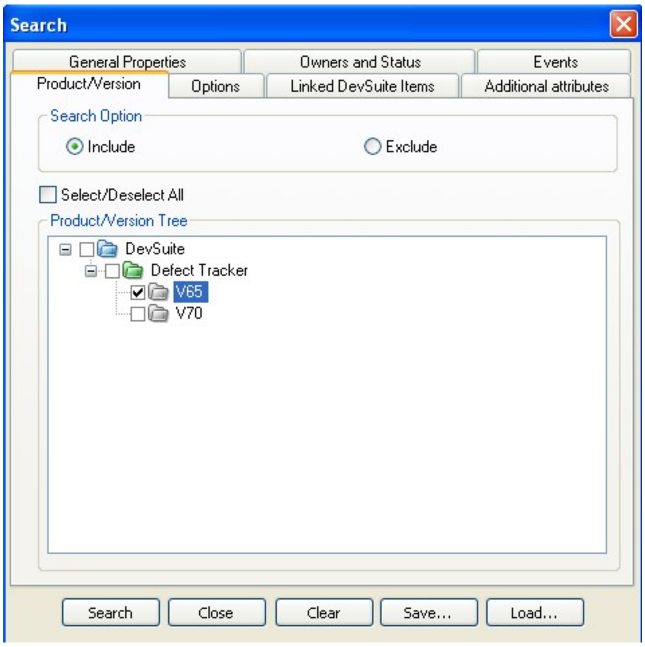
Review, Technical Review, or Work Review states.

2.3 Search by Product/Version

This search is an extension of using the product/version tree as a filter (described in section 4.1 Quick Filters). The product/version tree structure is defined by the DevSpec administrator, and is available for users to link specifications and requirements to applicable products and versions. This search allows users to find items related to a set of products and product versions easily.

To define product/version search conditions:

Launch the search dialog in DevSpec (use button).
Go to the Product/Version tab.
Select a specific version under a product folder to find all related specifications; OR select the checkbox next to a product name to find specifications for all versions of the selected product.



Search Option: ● Include ○ Exclude

□ Select/Deselect All

Product/Version Tree
□ DevSuite
□ Defect Tracker
☑ V65
□ V70

Click the Search button.

In the above example, the resulting list will include all specifications for Defect Tracker, version 65.

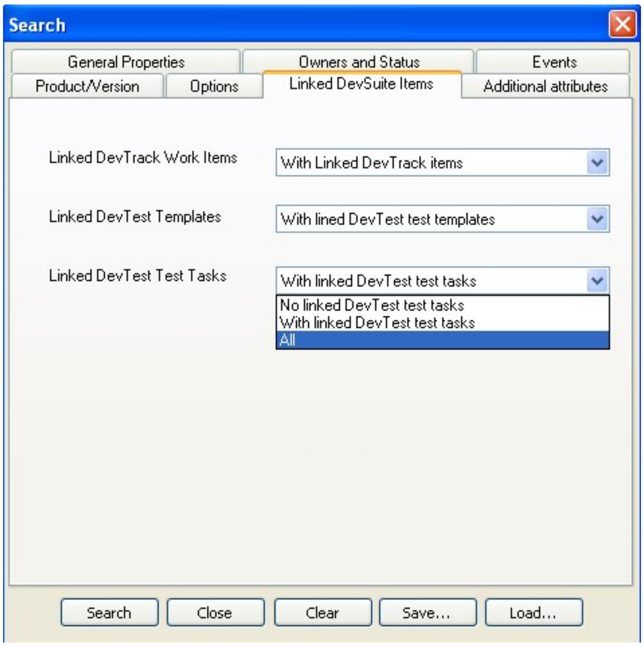
2.4 Search by Linked Item

In DevSpec, a specification or requirement can be linked to a development item in DevTrack. It can also be linked to a test case template, or even test tasks in DevTest. Searching for linked DevSuite items allows users to find any specifications that are currently linked to either a development item or a test case/task. For more information on linking items across DevTrack and DevTest, please see chapter 7, DevSuite Integration.

To define linked item search conditions:

Launch the search dialog in DevSpec (use button).
Go to the Linked DevSuite Items tab.
Select a search condition based on linked DevTrack and DevTest items.
To find specifications with a linked DevTrack item, select With Linked DevTrack items in the drop-down list.
To find specifications with no linked DevTrack items, select No Linked DevTrack work items in the drop-down list.
To find specifications with or without linked DevTrack item, select All in drop-down list.

Use the same principle for the other two dropdown lists: Linked DevTest Templates and Linked DevTest Test Tasks.



Linked DevTrack Work Items: With Linked DevTrack items

Linked DevTest Templates: With lined DevTest test templates

Linked DevTest Test Tasks: With linked DevTest test tasks
- No linked DevTest test tasks
- With linked DevTest test tasks
- All

Click the Search button.

In the above example, the resulting list will include all specifications that have a linked DevTrack item, a linked DevTest template, or a linked DevTest test task.

2.5 Search by Event

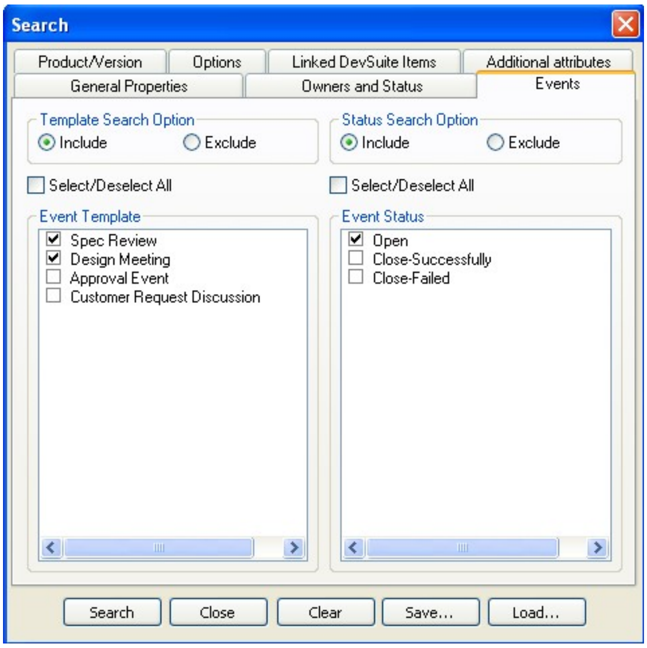
This search allows users to search for specifications and requirements that have associated events. Users may define search conditions based on event templates and event workflow states. For more information on events in DevSpec, please see chapter 8, section 3, Events.

To define event search conditions:

Launch the search dialog in DevSpec (use button).

Go to the Events tab.

Select one or more event templates and/or event workflow states.



Template Search Option: ● Include ○ Exclude

Status Search Option: ● Include ○ Exclude

□ Select/Deselect All

□ Select/Deselect All

Event Template
- ☑ Spec Review
- ☑ Design Meeting
- □ Approval Event
- □ Customer Request Discussion

Event Status
- ☑ Open
- □ Close-Successfully
- □ Close-Failed

Click the Search button.

In the above example, the resulting list would be items that currently have an associated Spec Review or Design Meeting event, and which are currently in the Open state.

2.6 Search by Field

DevSpec administrators may define a variety of field types to track and manage items.These field types typically include:

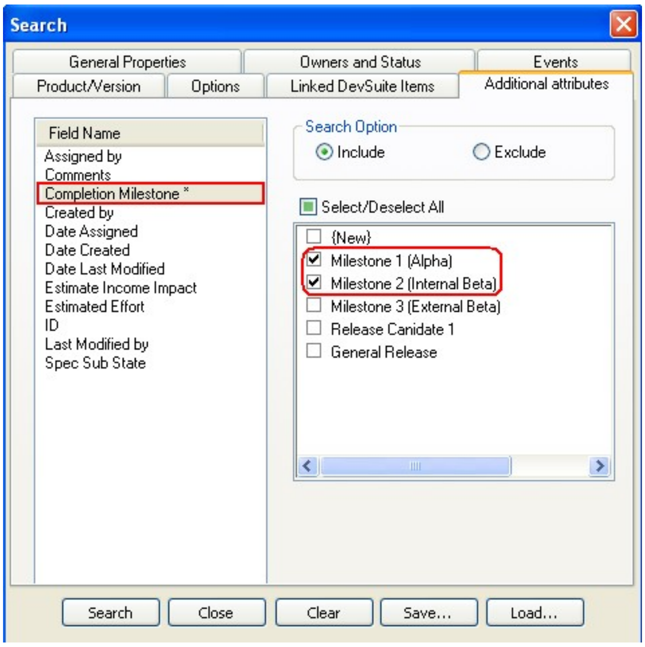
- Dropdown lists
- Combo boxes
- Date-time fields
- Single line edit boxes
- Multiple line edit boxes
- Multiple selection list boxes
- Checkboxes

Searching by fields allows users to search for items with a specific value in these admin-defined fields.

2.6.1 Dropdown Field Search

To define dropdown field search conditions:

Launch the search dialog in DevSpec (use button).
Go to the Additional attributes tab.
Highlight a field name of the dropdown list, multiple selection, or combo box type, and select its values.



Field Name
- Assigned by
- Comments
- Completion Milestone *
- Created by
- Date Assigned
- Date Created
- Date Last Modified
- Estimate Income Impact
- Estimated Effort
- ID
- Last Modified by
- Spec Sub State

Search Option: ● Include ○ Exclude

☑ Select/Deselect All

- □ {New}
- ☑ Milestone 1 (Alpha)
- ☑ Milestone 2 (Internal Beta)
- □ Milestone 3 (External Beta)
- □ Release Canidate 1
- □ General Release

Click the Search button.

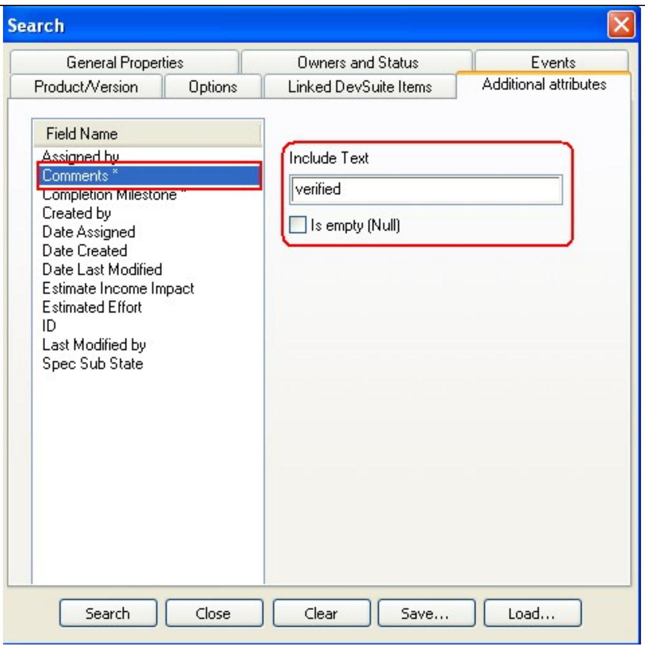
In the above example, the search result would include all specifications where the Completion Milestone field value is Milestone 1 or Milestone 2.

Note: In the Additional Attributes tab, all fields on which a search condition has been defined is indicated by an asterisk (*) after the field name.

2.6.2 Edit Box/Text Field Search

To define edit box/test field search conditions:

Launch the search dialog in DevSpec (use button).
Select the Additional attributes tab.
Highlight a field of the edit box type.
Enter a text to find all items where this text is found in the edit box field.
To find all items where the edit box field is empty (no text), check the box Is empty (Null).



Include Text: verified

☐ Is empty (Null)

Click the Search button.

In the above example, the search result would include all specifications where the Comments field contains the string, "verified". If we wanted to search for all specifications where the Comments field is blank/null, we would check the Is empty (Null) checkbox.

Note: In the Additional Attributes tab, the fields on which a search condition has been defined is indicated by an asterisk (*) after the field name.

2.6.3 Date-Time Field Search

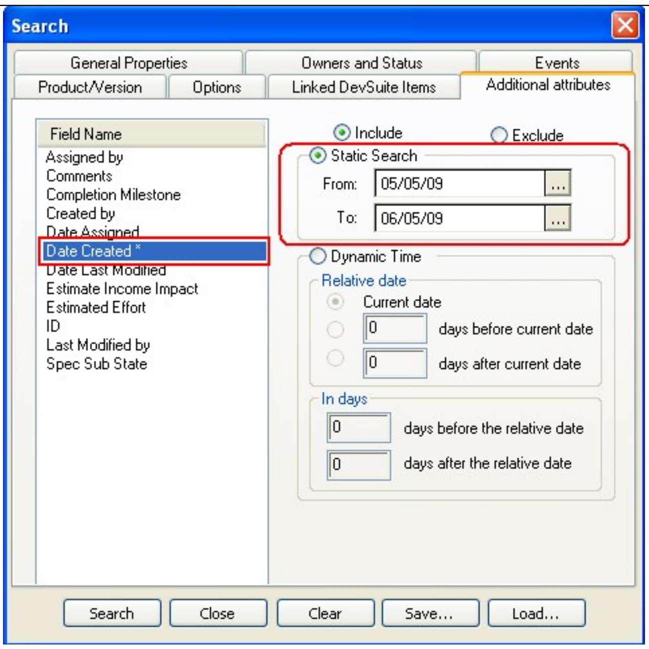
A date-time search enables the user to search for items based on the date and time that those items were submitted, assigned, last edited, or any other custom defined date-time field.

Static Date-Time Search

A static date-time condition returns all items that fall within a fixed time period. Static date-time search conditions are defined by a fixed starting date and fixed ending date.

To define Static Date Time field search condition:

Launch the search dialog in DevSpec (use button).
Go to the Additional attributes tab.
Highlight a field of the date-time type.
Enter a From date and a To date.



● Include ○ Exclude

● Static Search

From: 05/05/09

To: 06/05/09

○ Dynamic Time

● Current date

○ 0 days before current date

○ 0 days after current date

0 days before the relative date

0 days after the relative date

Click the Search button.

In the above example, the search result would include all specifications that were created between May 5, 2009 and June 5, 2009.

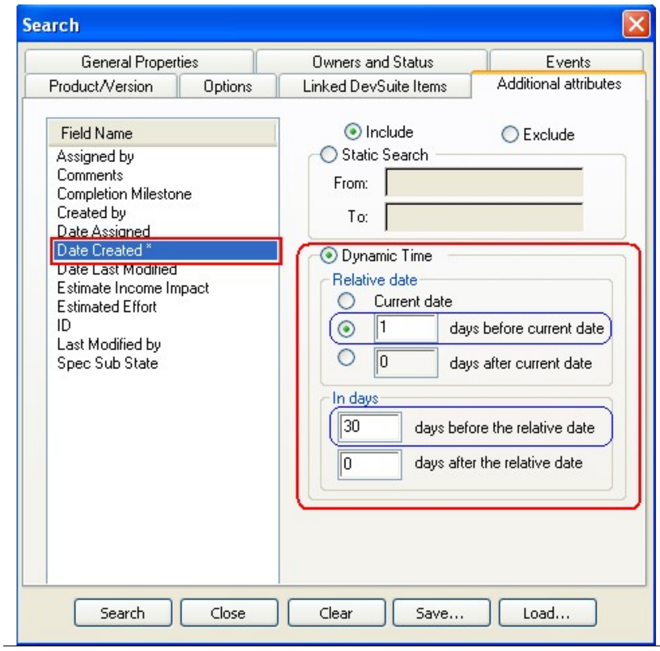
Note: In the Additional Attributes tab, the fields on which a search condition has been defined is indicated by an asterisk (*) after the field name.

Dynamic Date Time Search

A dynamic date-time condition returns all items that fall within a dynamically defined time period, a range relative to the current date.

To define dynamic date-time field search conditions:

Launch the search dialog in DevSpec (use button).
Go to the Additional attributes tab.
Highlight a field of the date-time type.
Select the Dynamic Time radio button.
Select the Current date radio button if today is the pivot/starting point. Otherwise, select the days before or days after radio button. Then enter a positive numeric value to set the pivot/starting point.



● Include ○ Exclude

○ Static Search

From:

To:

● Dynamic Time

○ Current date

● 1 days before current date

○ 0 days after current date

30 days before the relative date

0 days after the relative date

Enter the number of days in the In days section. This number could be before or after the relative date (or pivot/starting point) defined in the previous step.

In the above example, the search result would include all specifications that were created 30 days prior to yesterday. In this case, yesterday is defined by setting the relative date to be one day before the current date. Therefore, if the current day is May 5, 2009, the search will find all specifications created between April 4, 2009 and May 4, 2009.

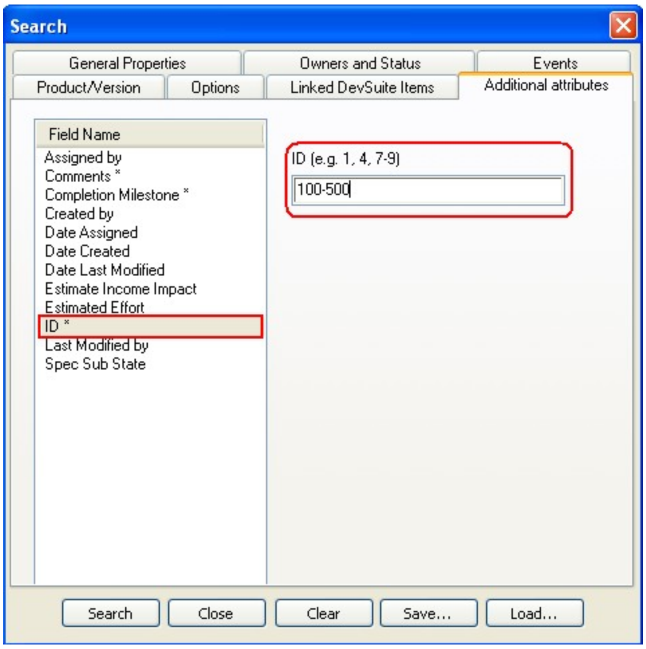
Note: In the Additional Attributes tab, the fields on which a search condition has been defined is indicated by an asterisk (*) after the field name.

2.6.4 Search by ID

Search conditions may also be used to retrieve items by their system-defined ID numbers. Each item in DevSpec is assigned a unique sequential ID number when submitted.

Using basic punctuation marks (commas, hyphens, and asterisks) as logical operators, project members may define complex queries to locate multiple work items based on a range of work item ID numbers.

This search is equivalent to the Go To feature in DevSpec. For more information, including the use of logical operators, please see section 1.1 in this chapter, Go To.



ID (e.g. 1, 4, 7-9): 100-500

The above example will result in a list of specifications with IDs ranging from 100 to 500.

Note: In the Additional Attributes tab, the fields on which a search condition has been defined is indicated by an asterisk (*) after the field name.

2.6.5 Search by Change Request Inclusion

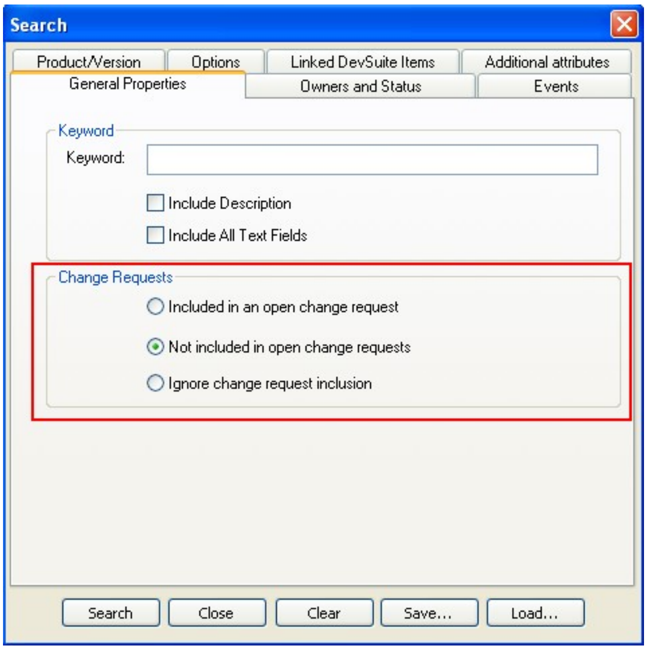
In DevSpec, one or more specifications may be grouped together and flagged for a change. All change request items are tracked under the change request view in DevSpec (must be enabled by the project administrator). For a complete explanation of the change request feature in DevSpec, please see chapter 8, section 1, Change Request.

In the specification view, users can filter specifications based on whether they are linked to a change request. To search for specifications/requirements linked to a change request:

Launch the search dialog in DevSpec (use button).

Go to the General Properties tab.

In the Change Requests section, select a radio button.



Keyword

Keyword:

□ Include Description

□ Include All Text Fields

Change Requests

○ Included in an open change request

● Not included in open change requests

○ Ignore change request inclusion

Select Included in an open change request to find all specifications that are currently linked to a change request item.
Select Not included in open change requests to find all specifications that are not linked to any change request item.
Select Ignore change request inclusion to find all specifications, regardless whether they are linked to a change request item.

Click the Search button.

3 Using Queries

A DevSpec query is a set of instructions (search conditions), for retrieving and displaying the desired data in the list panel or in a report.

Each query consists of one or more search conditions, which identify the searched data fields and the field value criteria for each field.

3.1 Query vs. Search

In DevSpec, a "query" is sometimes distinguished from a "search" in that queries are saved to the database and many be accessed and used again and again to retrieve work items that meet its search conditions, while a search is an ad-hoc query that is used once if not saved.

3.2 Query Types

Queries are also distinguished from searches in that queries are defined by its type. A query type defines query access rights; that is, who may access and use the queries saved in a project. DevSpec supports two types of queries:

Private Query: A private query is only available to the user who created it.

Public Query: Public queries are available to all other DevSpec users. To create a public query, a project team member must belong to an account type that has been granted the Can Define Public Query privilege by a DevSpec administrator.

Running Queries

To run a private or public query, select a query from the query dropdown list in the tool bar of the DevSpec client.

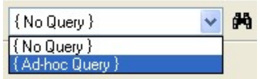
Creating Queries

Project members may save user-defined search parameters as queries using the search function in the DevSpec client.

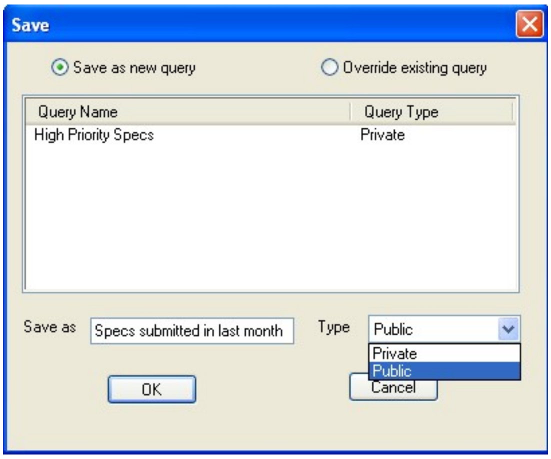
3.3 Defining and Saving Queries

To create a query:

Click the button in the tool bar (or)
Select {Ad-hoc Query} from the dropdown list as shown below (or)



Select Edit > Search in the menu bar (or)
Use CTRL + S.
Define search conditions across one or more tabs in the Search dialog.
Click the Save button at the bottom of the Search dialog.
The Save dialog box appears.



● Save as new query ○ Override existing query

Query Name	Query Type
High Priority Specs	Private

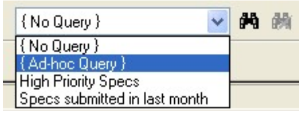
Save as: Specs submitted in last month
Type: Public (Private / Public)

Define a name for your query in the Save As text field.
Select an option from the Type drop-down list.
To make the query available to all users, select the Public option.
To create a query that is only available to yourself, select the Private option.

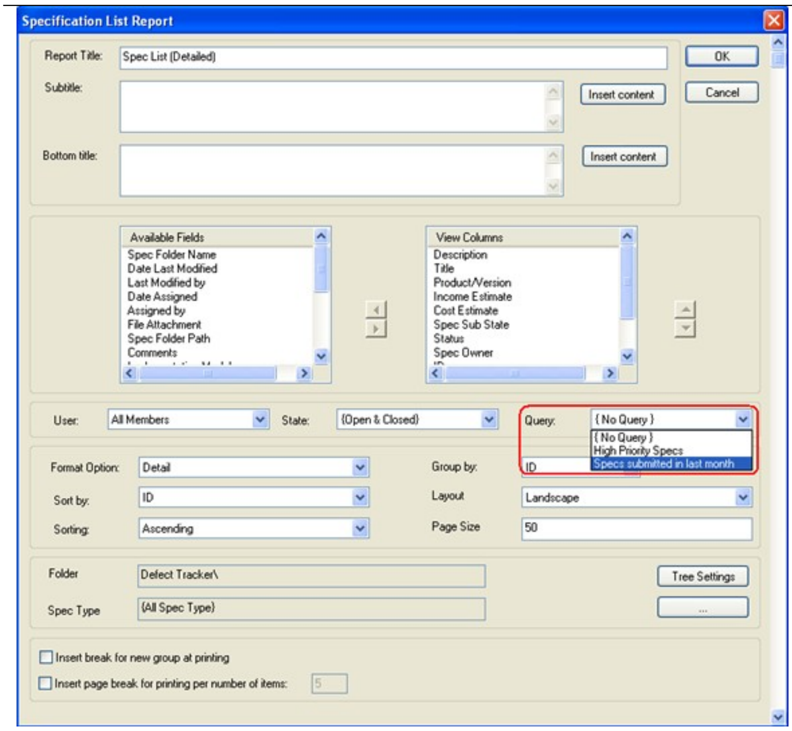
Note: Only the users who have been granted the Can Define Public Query privilege by the DevSpec administrator will be able to save a query as Public.

Click the OK button.
To run the search

All saved queries will be listed in the search dropdown list in tool bar.



The saved queries can also be used in the report view when the properties of a report are being defined.



Report Title: Spec List (Detailed)

Subtitle:

Bottom title:

Available Fields	View Columns
Spec Folder Name	Description
Date Last Modified	Title
Last Modified by	Product/Version
Date Assigned	Income Estimate
Assigned by	Cost Estimate
File Attachment	Spec Sub State
Spec Folder Path	Status
Comments	Spec Owner

User: All Members
State: (Open & Closed)
Query: (No Query)
(No Query)
High Priority Specs
Specs submitted in last month

Format Option: Detail
Group by: ID
Sort by: ID
Layout: Landscape
Sorting: Ascending
Page Size: 50

Folder: Defect Tracker\
Spec Type: (All Spec Type)

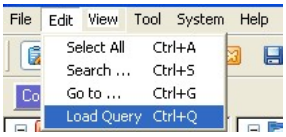
□ Insert break for new group at printing
□ Insert page break for printing per number of items: 5

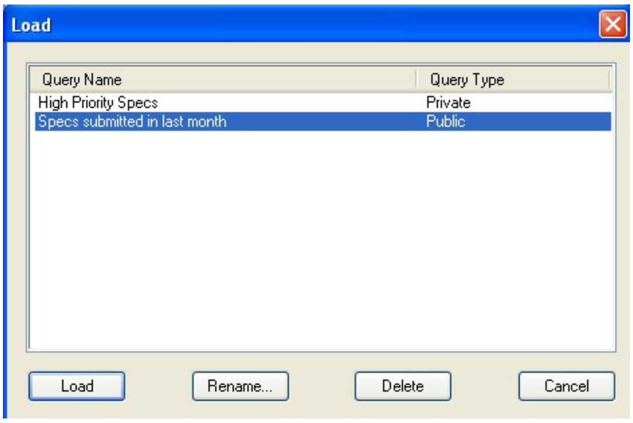
3.4 Loading Queries

To load a saved query:

Launch the search dialog in DevSpec (use button).
Click the Load button. The Load dialog appears.

Note: The Load dialog can also be opened by selecting Edit > Load Query in the menu bar.





Query Name	Query Type
High Priority Specs	Private
Specs submitted in last month	Public

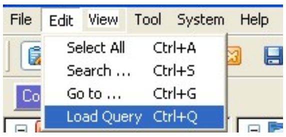
Select a query in theQuery Namelist.
Click theLoadbutton (in theLoaddialog).
Click theSearchbutton (in theSearchdialog).

3.5 Editing/Renaming Queries

To edit/rename a saved query:

Launch the search dialog in DevSpec (use button).
Click the Load button. The Load dialog appears.

Note: The Load dialog can also be opened by selecting Edit > Load Query in the menu bar.



Select a query in the Query Name list.
Click the Rename button.
Edit or rename the query.
Click the OK button.
Click the Cancel button (in the Load dialog).
Click the Close button (in the Search dialog).

Note: Users may only rename public queries if they have been granted the Can Define Public Query privilege by the DevSpec administrator.

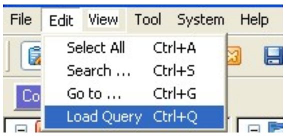
3.6 Deleting Queries

Project members may delete private and public queries. Project members may only delete public queries if they have been granted theCan Define Public Queryprivilege by a project administrator.

To delete a saved query:

Launch the search dialog in DevSpec (use button).
Click theLoadbutton. TheLoaddialog appears.

Note:TheLoaddialog can also be opened by selectingEdit>Load Queryin the menu bar.



Select a query in theQuery Namelist.
Click theDeletebutton.
Click theCancelbutton (in theLoaddialog).
Click theClosebutton (in theSearchdialog).

Note:Users may delete public queries only if they have been granted theCan Define Public Queryprivilege by the DevSpec administrator.

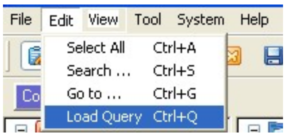
3.7 Adding More Search Conditions to Queries

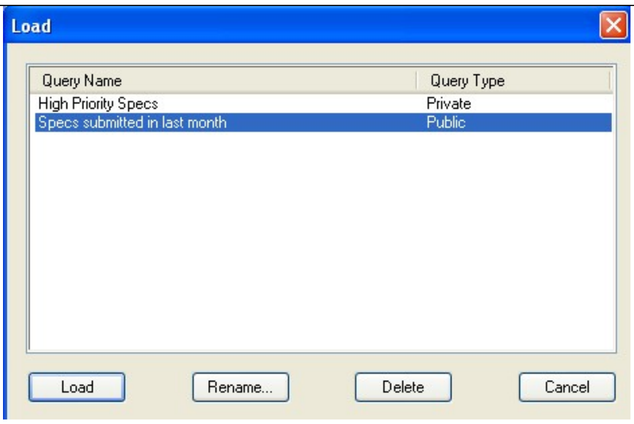
DevSpec allows users to define one or more search conditions and save it as a private or public query for re-use in future. Users can load a previously-defined query, add more search conditions to it, and either search or save the new query.

To add more search conditions to a saved query:

Launch the search dialog in DevSpec (use button).
Click theLoadbutton. TheLoaddialog appears.

Note:TheLoaddialog can also be opened by selectingEdit>Load Queryin the menu bar.

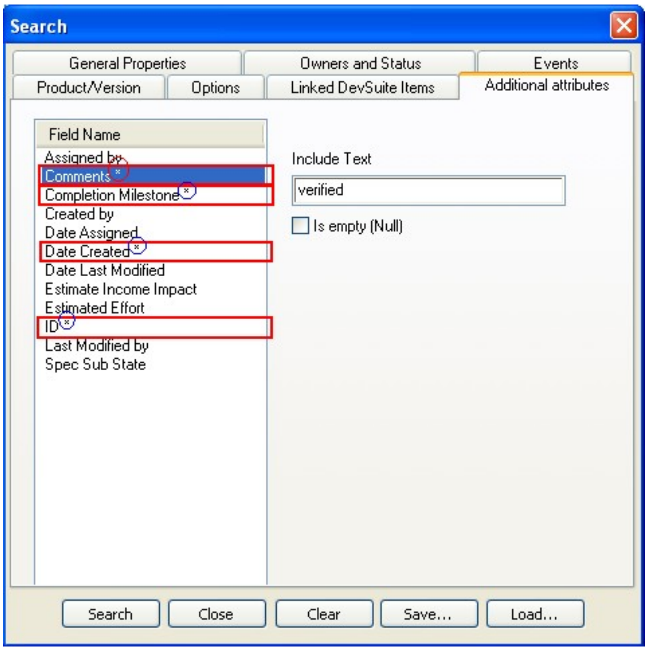




Query Name	Query Type
High Priority Specs	Private
Specs submitted in last month	Public

Select a query in theQuery Namelist.
Click theLoadbutton (in theLoaddialog).
In theSearchdialog, users can browse through different tabs and view the existing search conditions.

Note:In theAdditional Attributestab, the fields on which a search condition has been defined is indicated by an asterisk (*) after the field name.

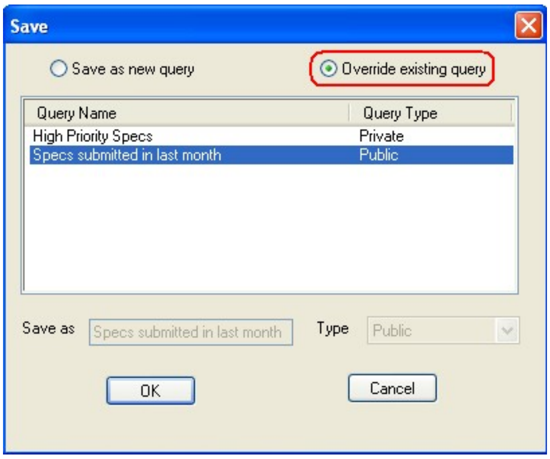


Field Name
Assigned by
Comments *
Completion Milestone *
Created by
Date Assigned
Date Created *
Date Last Modified
Estimate Income Impact
Estimated Effort
ID *
Last Modified by
Spec Sub State

Include Text: verified

□ Is empty (Null)

Users can then add more conditions across different tabs.
Click theSavebutton.
In theSavedialog, select theOverride existing queryradio button.



○ Save as new query ● Override existing query

Query Name	Query Type
High Priority Specs	Private
Specs submitted in last month	Public

Save as: Specs submitted in last month
Type: Public

Select the query in theQuery Namelist to be updated with new search conditions.

Click theOKbutton.

In theSearchdialog, click theSearchbutton to run the search, or click theClosebutton to finish saving the new query.

Chapter 5 - Importing

Most of the requirements and specifications tracked in a DevSpec project are manually submitted to the project from within the DevSpec client. DevSpec also provides users with a few other methods for submitting items in DevSpec. These alternative methods are especially important to organizations that need to import existing requirements and specifications into DevSpec.

1 Import data using Microsoft Office Add-on

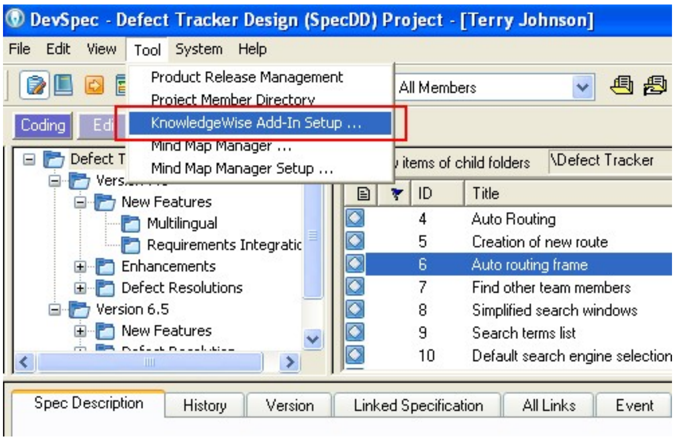
Requirements, specifications and other knowledge items can come from a variety of sources and may initially be stored in Microsoft Office applications, such as Word, Outlook, PowerPoint, etc. A good requirements management tool should facilitate easy import of data from such commonly used applications.

The DevSpec Office Add-in module allows users to integrate Microsoft Office tools with DevSpec.

2 Installing the Microsoft Office Add-In Module

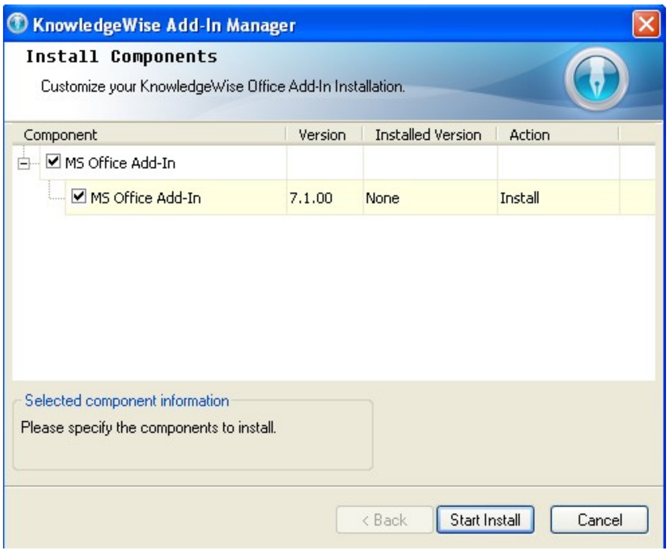
To install the Microsoft Office Add-In module:

Select*Tools*>*KnowledgeWiseAdd-In Setup*.



ID	Title
4	Auto Routing
5	Creation of new route
6	Auto routing frame
7	Find other team members
8	Simplified search windows
9	Search terms list
10	Default search engine selection

The KnowledgeWise Add-In installation wizard appears.



Customize your KnowledgeWise Office Add-In Installation.

Component	Version	Installed Version	Action
☑ MS Office Add-In			
☑ MS Office Add-In	7.1.00	None	Install

Selected component information

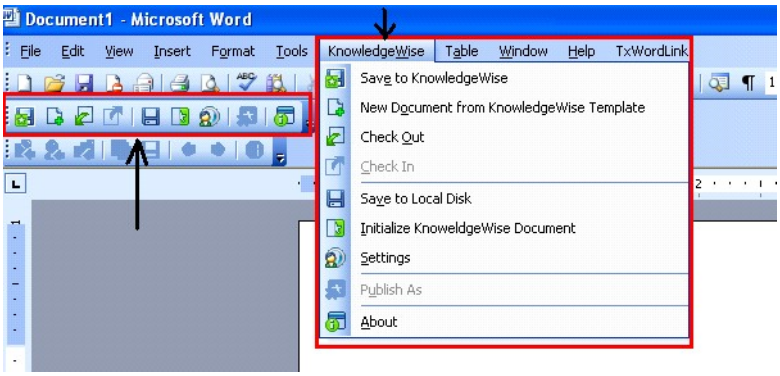
Please specify the components to install.

Click the*Start Install*button, and complete the installation.

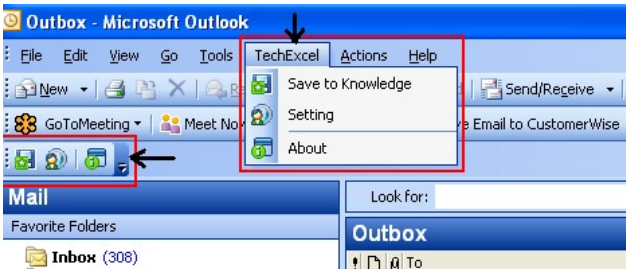
Note: Make sure all Microsoft Office applications are closed before running this installation.

After successful installing the DevSpec Office Add-in module, users can see DevSpec tools in Microsoft Office applications:

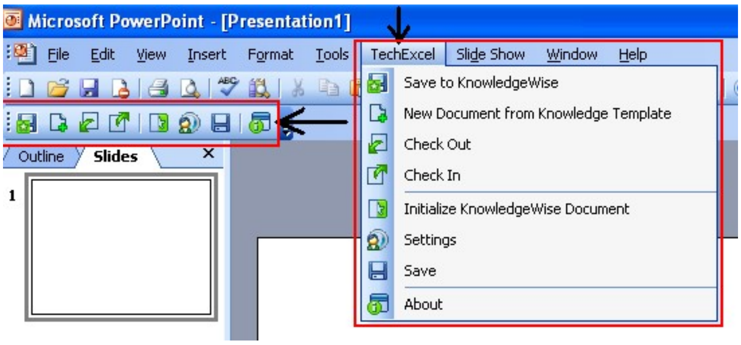
DevSpec tools in Microsoft Word:



DevSpec tools in Microsoft Outlook:



DevSpec tools in Microsoft PowerPoint:

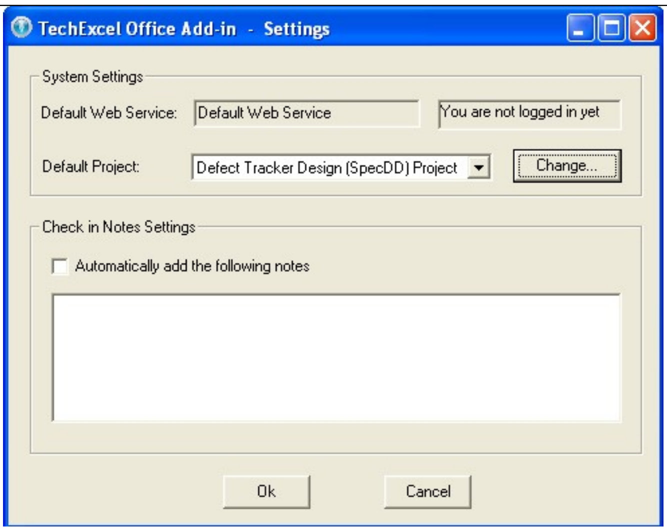


3 DevSpec and Microsoft Word

3.1 Default Settings

To define DevSpec default settings in Microsoft Word:

Open Microsoft Word and select*KnowledgeWise>Settings*in the main menu; or click the icon on the tool bar. The*Settings*dialog appears.


TechExcel Office Add-in - Settings

System Settings

Default Web Service: Default Web Service

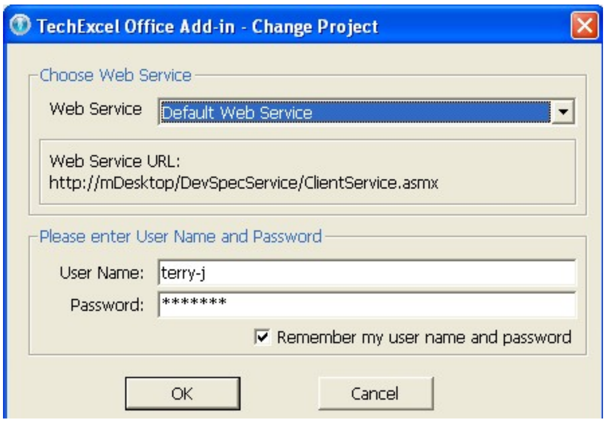
You are not logged in yet

Default Project: Defect Tracker Design (SpecDD) Project

Check in Notes Settings

☐ Automatically add the following notes

Select a default DevSpec project in the dropdown list
Click the*Change*button. The*Change Project*dialog appears.


TechExcel Office Add-in - Change Project

Choose Web Service

Web Service: Default Web Service

Web Service URL:
http://mDesktop/DevSpecService/ClientService.asmx

Please enter User Name and Password

User Name: terry-j

Password: *******

☑ Remember my user name and password

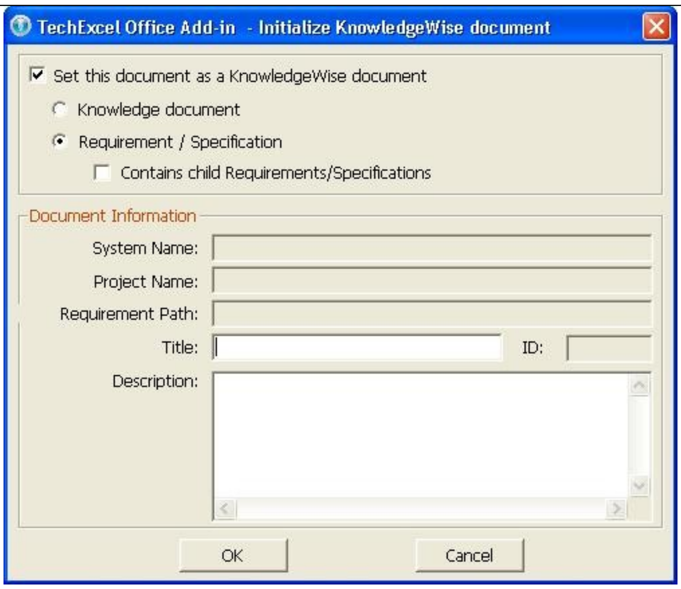
Select a web service in the*Web Service*dropdown list.
Enter a username and password.
Optional: To save the login settings, select the*Remember my user name and password*check box.
Click the*OK*button. The*Change Project*dialog closes.
In the*Settings*dialog, select the*Automatically add the following notes*check box.
Enter a brief message in the*Notes*text box. This will be used as the default note when checking in items to DevSpec.
Click the*OK*button.

3.2 Initializing Requirement Documents

Using controls in the*Initialize KnowledgeWise Document*dialog box, project team members may create (or initialize) new KnowledgeWise knowledge documents and DevSpec requirement documents.

To initialize a requirement document:

Click the*Initialize KnowledgeWise Document*button in the DevSpec tool bar or the Microsoft Word tool bar. The*Initialize KnowledgeWise Document*dialog appears.



TechExcel Office Add-in - Initialize KnowledgeWise document

☑ Set this document as a KnowledgeWise document

○ Knowledge document

● Requirement / Specification

□ Contains child Requirements/Specifications

Document Information

System Name:

Project Name:

Requirement Path:

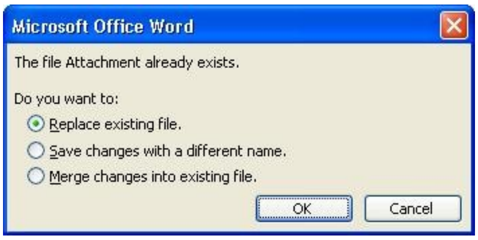
Title: ID:

Description:

Define the title and provide a brief description of the requirement in the*Title*and*Description*text boxes. The*Save As*window appears.

Identify an existing Microsoft Word document.

Click the*Save*button. The*Microsoft Word*dialog appears.



Microsoft Office Word

The file Attachment already exists.

Do you want to:

● Replace existing file.

○ Save changes with a different name.

○ Merge changes into existing file.

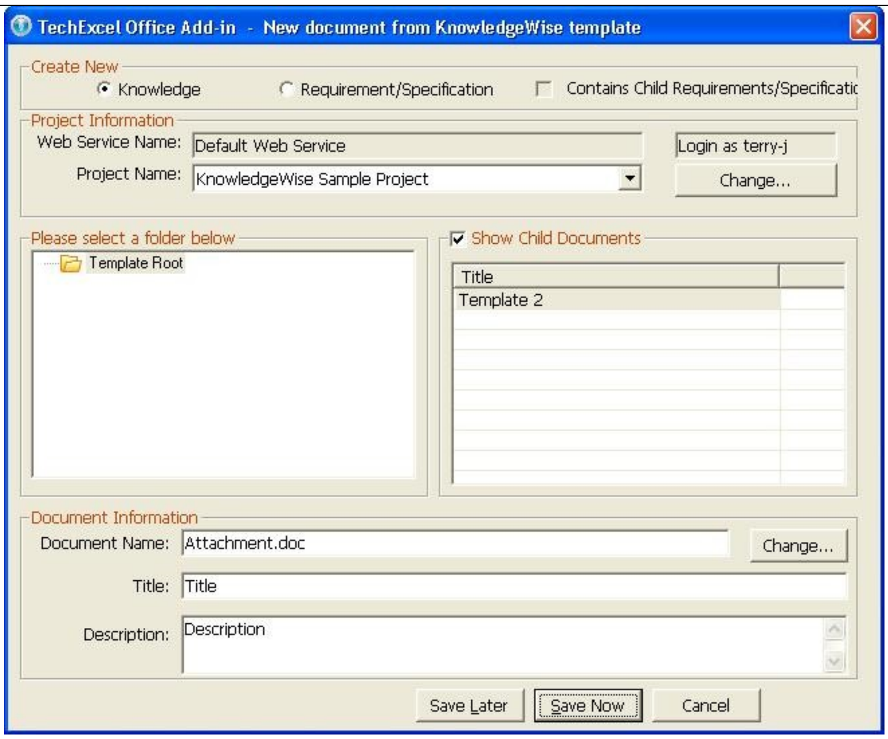
Select an option for saving the requirement document.
To overwrite the selected file with text, select the*Replace Existing File*radio button.
To save text as a different file, select the Save Changes With Different Name option button.
To merge text into the selected file, select the Merge Changes Into Existing File option button.

Click the*OK*button.

Note: Using controls in the*New Document from KnowledgeWise Template*window, project team members may add new requirements, child requirements, or knowledge
documents.

To create a requirement document from a template:

Click the*New Document*button in the DevSpec tool bar or the Microsoft Word tool bar. The*New Document*dialog box appears.



TechExcel Office Add-in - New document from KnowledgeWise template

Create New: ● Knowledge ○ Requirement/Specification □ Contains Child Requirements/Specificatic

Project Information
Web Service Name: Default Web Service — Login as terry-j
Project Name: KnowledgeWise Sample Project

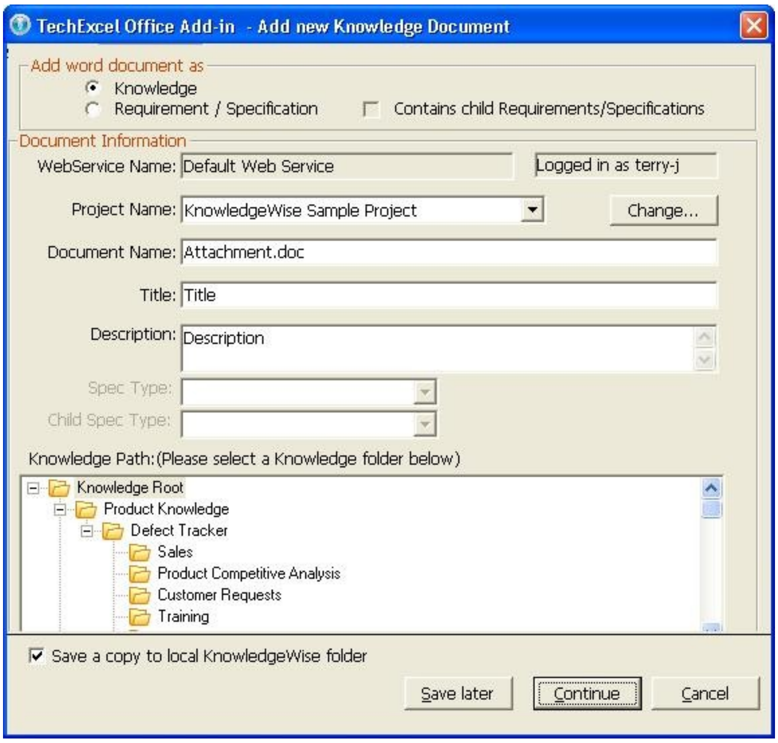
Please select a folder below: Template Root

☑ Show Child Documents

Title	
Template 2	

Document Information
Document Name: Attachment.doc
Title: Title
Description: Description

Click the Add button. The*Add New* dialog appears.



TechExcel Office Add-in - Add new Knowledge Document

Add word document as: ● Knowledge ○ Requirement / Specification □ Contains child Requirements/Specifications

Document Information
WebService Name: Default Web Service — Logged in as terry-j
Project Name: KnowledgeWise Sample Project
Document Name: Attachment.doc
Title: Title
Description: Description
Spec Type:
Child Spec Type:

Knowledge Path:(Please select a Knowledge folder below)

```
Knowledge Root
  Product Knowledge
    Defect Tracker
      Sales
      Product Competitive Analysis
      Customer Requests
      Training
```

☑ Save a copy to local KnowledgeWise folder

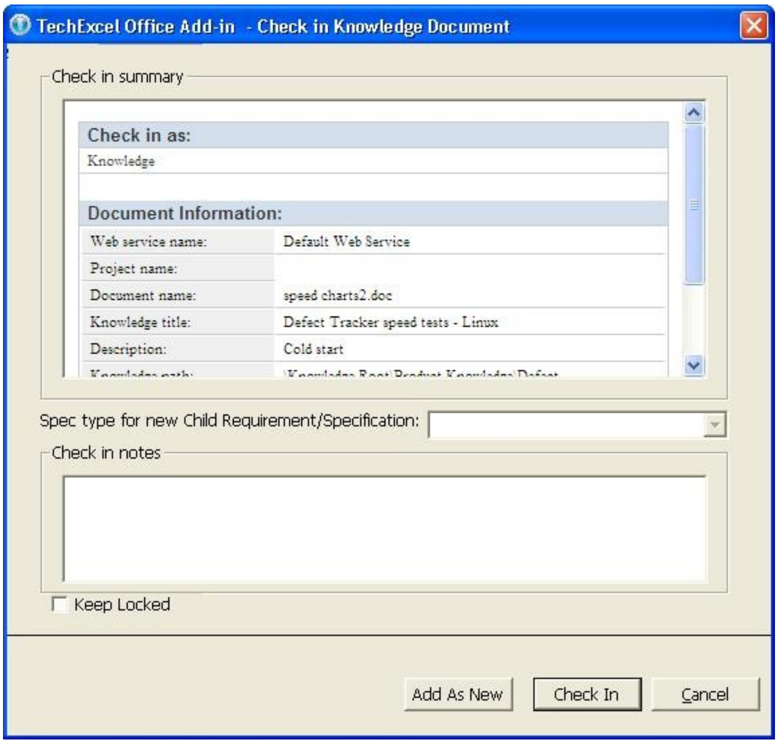
Optional: To change projects, select the*Change*button and update the web service, user name, and password in the*Change Project*dialog box.
Select the*Requirement*option in the *Add word document as* section.
Optional: If the requirement contains child requirements, select the*Contains Child Requirements*check box.
Optional: To switch projects, select a project in the*Project Name*dropdown list.
Select a requirement template folder in the requirement template tree panel.

3.3 Creating Items from Checked Out Documents

To create a new requirement document from checked out document:

Check out a requirement document.

Click the*Check In*button in the Word tool bar. The*Check In Requirement Document*dialog appears.



TechExcel Office Add-in - Check in Knowledge Document

Check in summary

Check in as:

Knowledge

Document Information:

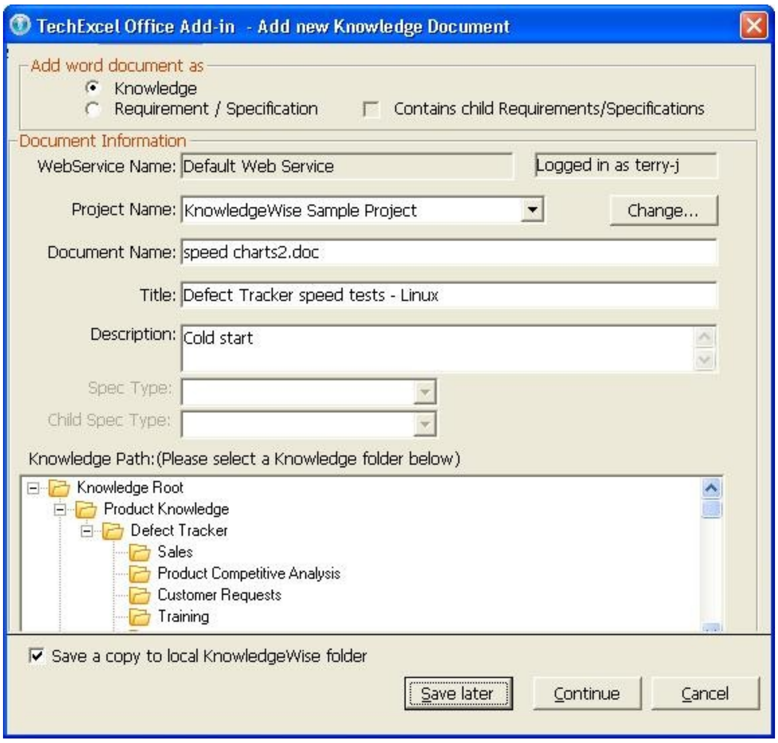
Web service name:	Default Web Service
Project name:	
Document name:	speed charts2.doc
Knowledge title:	Defect Tracker speed tests - Linux
Description:	Cold start
Knowledge path:	\Knowledge Root\Product Knowledge\Defect

Spec type for new Child Requirement/Specification:

Check in notes

□ Keep Locked

Enter check in notes in the*Check In Notes*text box.
Optional: To keep the requirement document locked, select the*Keep Locked*check box.
Click the*Add As New*button. The*Add New Requirement Document* dialog appears.



TechExcel Office Add-in - Add new Knowledge Document

Add word document as

● Knowledge
○ Requirement / Specification □ Contains child Requirements/Specifications

Document Information

WebService Name: Default Web Service Logged in as terry-j

Project Name: KnowledgeWise Sample Project

Document Name: speed charts2.doc

Title: Defect Tracker speed tests - Linux

Description: Cold start

Spec Type:

Child Spec Type:

Knowledge Path:(Please select a Knowledge folder below)

- Knowledge Root
  - Product Knowledge
    - Defect Tracker
      - Sales
      - Product Competitive Analysis
      - Customer Requests
      - Training

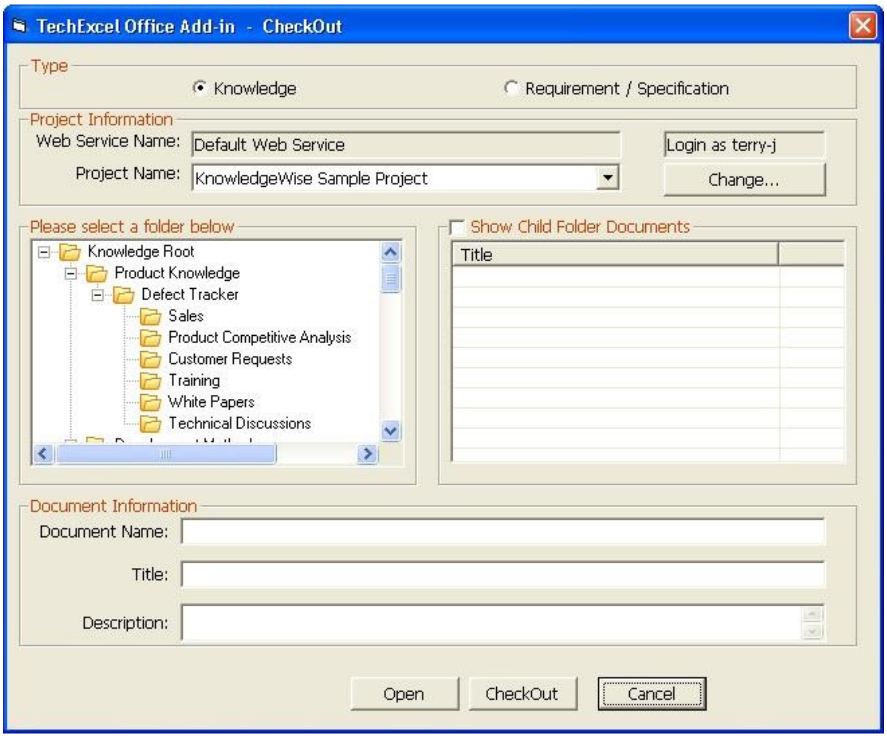
☑ Save a copy to local KnowledgeWise folder

Optional: To change the login profile, select the*Change*button and update the web service, user name, and password in the*Change Project*dialog.
Select the requirement option in the *Add word document as* section.
Optional: If the requirement contains child requirements, select the*Contains Child Requirements*check box.
Optional: To switch projects, select a project in the*Project Name*dropdown list.
Select a requirement template folder in the requirement template tree panel.
Click the*Continue*button.

3.4 Checking Out Requirement Documents

To check out a requirement document:

Click the*Check Out*button in the Word tool bar. The*Check Out* dialog appears.



TechExcel Office Add-in - CheckOut

Type: ● Knowledge ○ Requirement / Specification

Project Information

Web Service Name: Default Web Service

Login as terry-j

Project Name: KnowledgeWise Sample Project

Please select a folder below

- Knowledge Root
  - Product Knowledge
    - Defect Tracker
      - Sales
      - Product Competitive Analysis
      - Customer Requests
      - Training
      - White Papers
      - Technical Discussions

□ Show Child Folder Documents

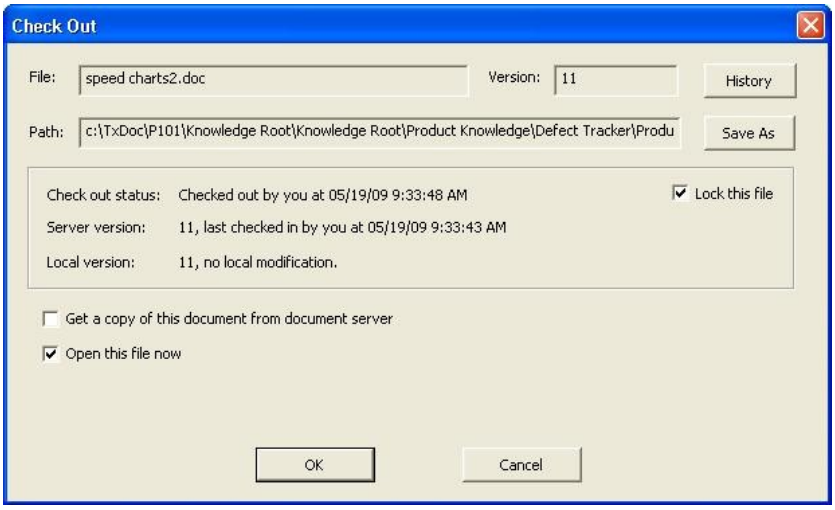
Document Information

Document Name:

Title:

Description:

To change the login profile, select the*Change*button and update the web service, user name, and password in the*Change Project*dialog.
Select the*Requirement*option in the*Type*section.
Optional: To switch projects, select a project in the*Project Name*dropdown list.
Select a requirement in the requirement template tree panel.
Select a requirement in the requirement document list.
Optional: To display child requirements, select the*Show Child Documents*check box.
Click the*Check Out*button. The*Check Out*dialog appears.



Check Out

File: speed charts2.doc

Version: 11

Path: c:\TxDoc\P101\Knowledge Root\Knowledge Root\Product Knowledge\Defect Tracker\Produ

Check out status: Checked out by you at 05/19/09 9:33:48 AM

☑ Lock this file

Server version: 11, last checked in by you at 05/19/09 9:33:43 AM

Local version: 11, no local modification.

□ Get a copy of this document from document server

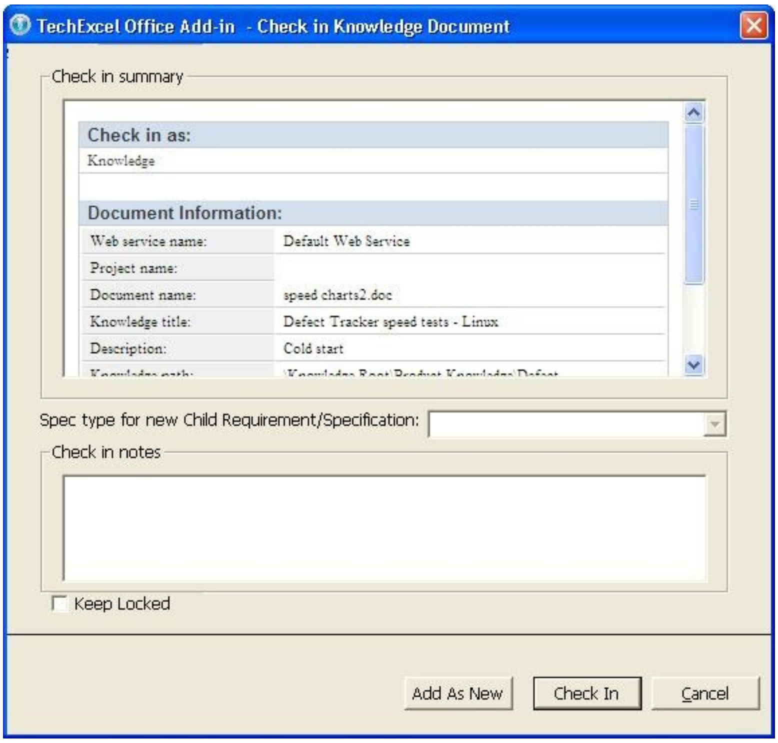
☑ Open this file now

Optional: To open the file, select the*Open This File Now*check box.
Optional: To place a lock on the file, select the*Lock This File*check box.
Click the*OK*button.

3.5 Checking In Requirement Documents

To check in a requirement document:.

Click the*Check In*button in the Word tool bar. The*Check In Requirement Document*dialog appears.


TechExcel Office Add-in - Check in Knowledge Document
Check in summary
Spec type for new Child Requirement/Specification:
Check in notes
Add As New
Check In
Cancel

Check in as:

Knowledge

Document Information:

Web service name:	Default Web Service
Project name:	
Document name:	speed charts2.doc
Knowledge title:	Defect Tracker speed tests - Linux
Description:	Cold start
Knowledge path:	\Knowledge Root\Product Knowledge\Defect

□ Keep Locked

Enter check in notes in the*Check In Notes*text box.
Optional: To keep the requirement document locked, select the*Keep Locked*check box.
Click the*Check In*button.

4 Managing Word Requirements

A Word requirement is a line item that defines a specific attribute, quality, or capability of a system. A requirement document may represent multiple Word requirements (child requirements) that may be managed and tracked independently in the DevSpec client.

Every Word requirement in a requirement document may be published independently of the other line items in that requirement document. Word requirements may be published as requirements, knowledge items, or specifications.

4.1 Adding Word Requirements to Requirement Documents

To add a Word requirement to a requirement document:

Create or open a requirements document in Microsoft Word.
Highlight one or more lines of text in the document.
Click the*Add New*button in the Word tool bar. The*New Requirement*dialog appears.

Define the name and a brief description of the requirement line item.
Click the*OK*button.

The selected text is enclosed in DevSpec markup tags:

4.2 Editing Word Requirements

To edit a Word requirement:

Create or open a requirements document in Microsoft Word.
Highlight one or more lines of text in the document.
Click the*Edit*button in the Word tool bar. The*Edit Requirement*dialog appears.

Update the title or description of the requirement.
Click the*OK*button. The selected Word requirement is updated.

4.3 Deleting Word Requirements

To delete a Word requirement:

Create or open a requirements document in Microsoft Word.
Highlight one or more lines of text in the document.
Click the*Delete*button in the Word tool bar. A confirmation dialog appears.

Click the Yes button. The selected Word requirement is deleted.

4.4 Browsing Word Requirements

To browse Word requirements in a Word document, select the*Previous Requirement*button or*Next Requirement*button in the DevSpec bar.

The Previous Requirement button enables the user to select the previous Word requirement in a document.
The Next Requirement button enables the user to select the next Word requirement in a document.

4.5 Browsing Word Requirement Summaries

Using controls in the*Requirement Summary*window, project team members may quickly browse the Word requirements in a requirements document, go to selected Word requirements, or delete selected Word requirements.

To browse requirement summaries:

Create or open a requirements document in Microsoft Word.
Highlight one or more lines of text in the document.
Click the*Requirement Summary*button in the Word tool bar. The*Requirement Summary*dialog appears.

Select the check box next to the title of the Word requirement.
Optional: To go to the selected Word requirement, select the*Go To*button.
Optional: To delete the selected Word requirement, select the*Delete*button.

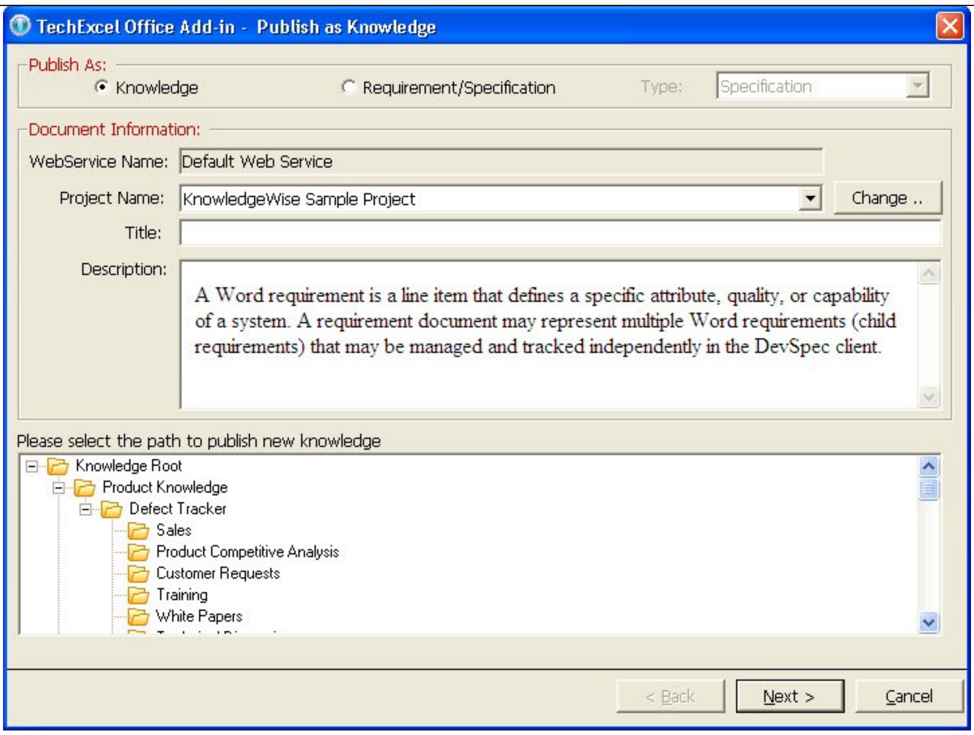
4.6 Publishing Word Requirements As Knowledge

Every Word requirement in a requirement document may be published independently of the other line items in that requirement document. Word requirements may be
published as requirements, knowledge items, or specifications.

Using controls in the*Publish As Requirement*dialog, project team members may publish Word requirement-line items in a requirement document-as independent standard requirements.

To publish a Word requirement as knowledge:

Create or open a requirements document in Microsoft Word.
Select a requirement line item-a Word requirement-in the requirement document.
Click the*Publish*button in the Word tool bar. The*Publish* dialog appears.



TechExcel Office Add-in - Publish as Knowledge

Publish As:
● Knowledge ○ Requirement/Specification Type: Specification

Document Information:

WebService Name: Default Web Service

Project Name: KnowledgeWise Sample Project

Title:

Description: A Word requirement is a line item that defines a specific attribute, quality, or capability of a system. A requirement document may represent multiple Word requirements (child requirements) that may be managed and tracked independently in the DevSpec client.

Please select the path to publish new knowledge

- Knowledge Root
  - Product Knowledge
    - Defect Tracker
      - Sales
      - Product Competitive Analysis
      - Customer Requests
      - Training
      - White Papers

Select the*Publish As Knowledge*option button.
Optional: To switch projects, select a project in the*Project Name*dropdown list.
Optional: To change login profile, click the Change button and define the web service, user name, and password.
Select a knowledge folder in the knowledge template tree panel.
Click the*Next*button.

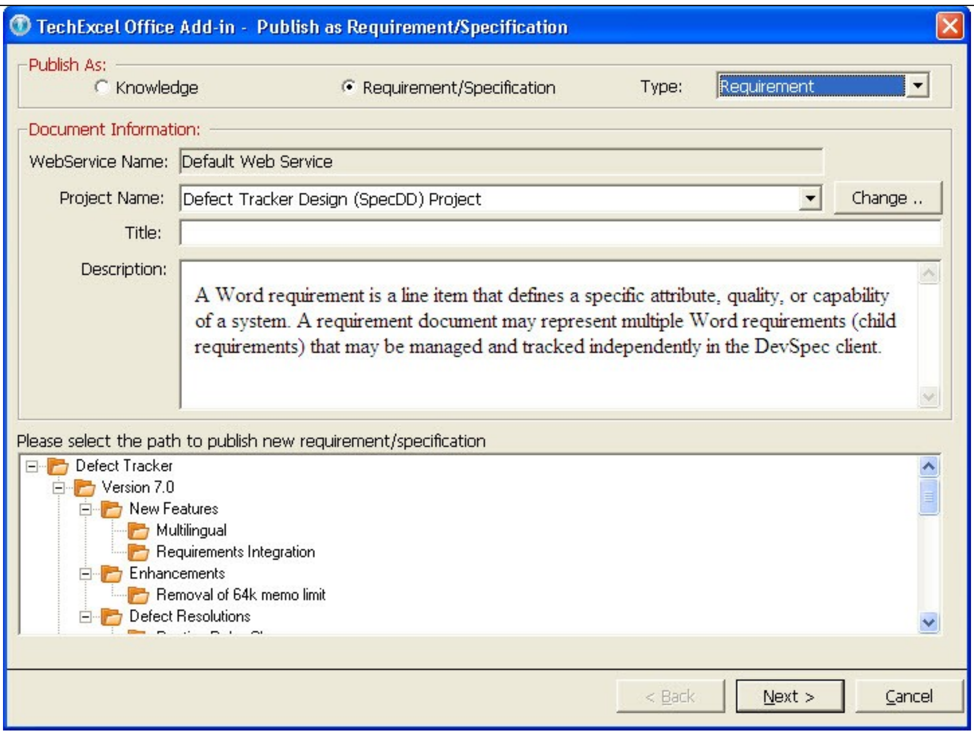
4.7 Publishing Word Requirements As Requirements

Every Word requirement in a requirement document may be published independently of the other line items in that requirement document. Word requirements may be
published as requirements, knowledge items, or specifications.

Using controls in the*Publish As Requirement*dialog, project team members may publish a Word requirement-a line item in a requirement document-as a standard requirement.

To publish a Word requirement as a standard requirement:

Create or open a requirements document in Microsoft Word.
Select a requirement line item-a Word requirement-in the requirement document.
Click the*Publish*button in the Word tool bar. The*Publish* dialog appears.


TechExcel Office Add-in - Publish as Requirement/Specification
Publish As:
Document Information:
Please select the path to publish new requirement/specification
< Back
Next >
Cancel
Change ..

Publish As: ○ Knowledge ● Requirement/Specification Type: Requirement

WebService Name: Default Web Service

Project Name: Defect Tracker Design (SpecDD) Project

Title:

Description: A Word requirement is a line item that defines a specific attribute, quality, or capability of a system. A requirement document may represent multiple Word requirements (child requirements) that may be managed and tracked independently in the DevSpec client.

- Defect Tracker
  - Version 7.0
    - New Features
      - Multilingual
      - Requirements Integration
    - Enhancements
      - Removal of 64k memo limit
    - Defect Resolutions

Select the*Publish As Knowledge*option button.
Optional: To switch projects, select a project in the*Project Name*dropdown list.
Optional: To change login profile, click the Change button and define the web service, user name, and password.
Select a requirement folder in the knowledge template tree panel.
Click the*Next*button.

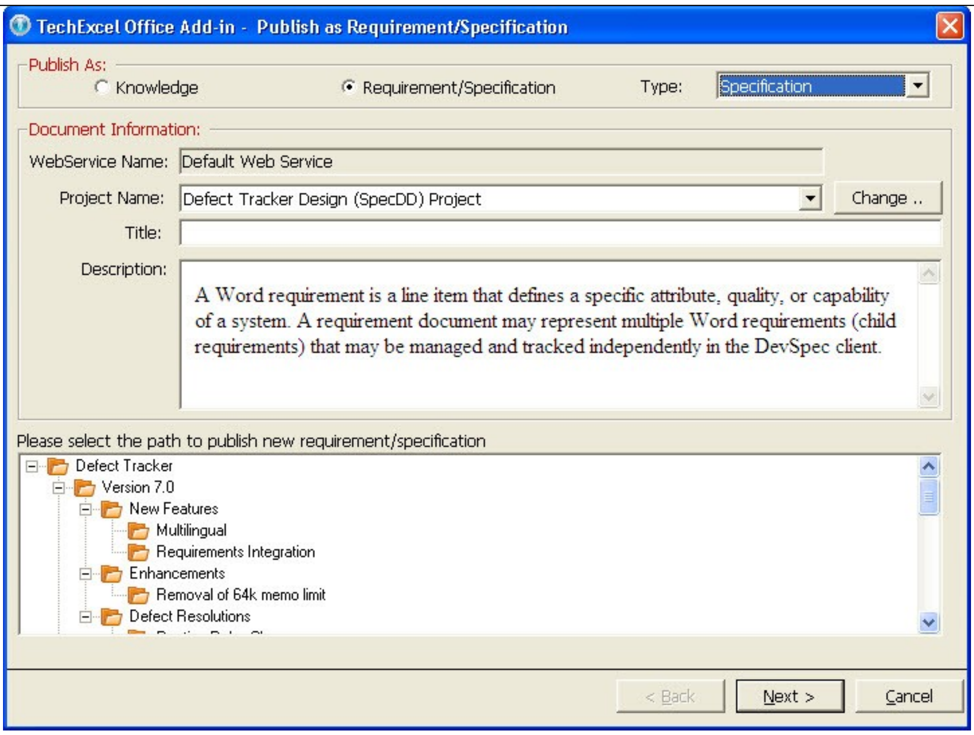
4.8 Publishing Word Requirements as Specifications

Every Word requirement in a requirement document may be published independently of the other line items in that requirement document. Word requirements may be
published as requirements, knowledge items, or specifications.

Using controls in the*Publish As Specification* dialog, project team members may publish a Word requirement-a line item in a requirement document-as a specification.

To publish a Word requirement as a specification:

Create or open a requirements document in Microsoft Word.
Select a requirement line item-a Word requirement-in the requirement document.
Click the*Publish*button in the Word tool bar. The*Publish* dialog appears.



TechExcel Office Add-in - Publish as Requirement/Specification
Publish As:
○ Knowledge
● Requirement/Specification
Type: Specification
Document Information:
WebService Name: Default Web Service
Project Name: Defect Tracker Design (SpecDD) Project
Change ..
Title:
Description:
Please select the path to publish new requirement/specification
Defect Tracker
Version 7.0
New Features
Multilingual
Requirements Integration
Enhancements
Removal of 64k memo limit
Defect Resolutions
< Back
Next >
Cancel

A Word requirement is a line item that defines a specific attribute, quality, or capability of a system. A requirement document may represent multiple Word requirements (child requirements) that may be managed and tracked independently in the DevSpec client.

Select the*Publish As Knowledge*option button.
Optional: To switch projects, select a project in the*Project Name*dropdown list.
Optional: To change login profile, click the*Change*button and define the web service, user name, and password.
Select a specification folder in the specification template tree panel.
Click the*Next*button.

Chapter 6 - Backlog Management

When the design of a specification is complete, it is ready to be implemented. This transition is facilitated by the backlog. From assigning specifications to subprojects, to initiating an implementation, this section covers the entire process and concept of the DevSpec backlog.

1 Backlog Management - Concept

DevSpec provides rich functions and features and can be used as a stand alone requirement and product design management tool. However, the true value of DevSpec is to provide better product design management. This allows enterprises to achieve more mature development process control. An essential part of agile development is backlog management.

The product backlog can be formally defined as the specifications that are ready to be implemented. While a specification in DevSpec represents a feature, enhancement, change or defect to be fixed, it becomes a backlog item once it is ready to be implemented. Backlog items represent a link for a specification to be implemented within a certain development iteration. A specification may need to be implemented multiple times, and therefore can be linked with multiple implementation links.

DevSpec is designed and developed based on TechExcel's conceptual modeling ideology: requirements should be formally represented and can be quantified. Quantified requirements can better drive implementation and testing.

With specifications serving as the quantified requirements, application development project planning, implementation, and QA testing can all be managed with specificaitons as their foundation.

- Project planning with specifications is about committing a set of features for development iterations, planning the resources needed for each specification, assigning the proper start and end time for the development work, and providing estimates on time for the development team to finish the new features, enhancements, and bug fixing specifications.
- Development implementation tracking becomes about task management around the committed specifications. Each specification may require one or many development tasks, and development teams can update the development task's status, update the finish date, the time spent and the time remaining. As a result, the planned time and resources can be used to display the time and resources used for each development task.
- QA testing becomes requirement-driven thorough specifications. As specifications are committed, planned, and implemented, QA test tasks are created for each specification. Specifications can be used to define a QA test library, and to better quantify and standardize the QA testing process.

All the above mentioned functions are facilitated by the Backlog feature in DevSpec. It enables DevSpec users to easily move a committed Specification or Feature to a temporary repository of prioritized items, before these items are actually scheduled for implementation and testing.

2 Managing the Specification Backlog

In DevSuite, a specification defines a conceptual product that may be implemented in one or more development subprojects. Each subproject represents a distinct area of development and defines the framework that development organization uses to manage, schedule, and track iterations of development within that area.

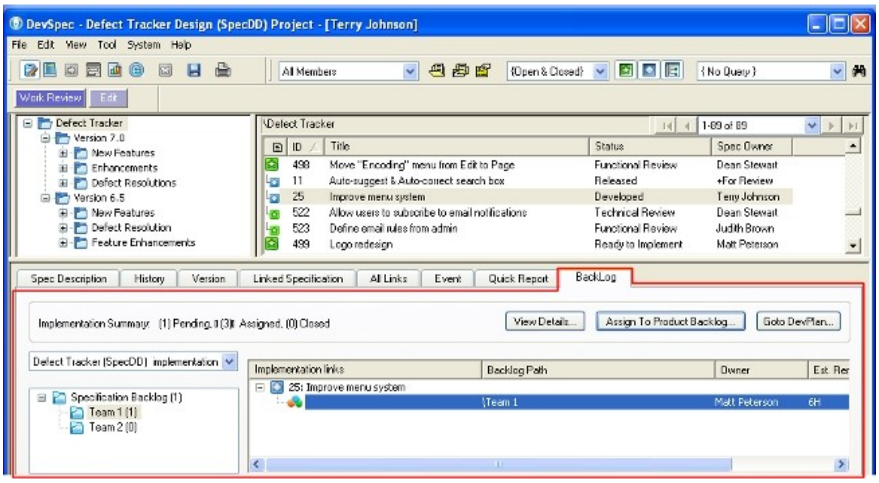
Implementation linking enables development organizations to schedule the development of designed product - as defined by a specification - in development subprojects. Every development issue managed in a subproject may be linked to a single "implementation specification" - a pairing of a specification and an implementation module.

A specification may define one or more subprojects. In DevSuite, every subproject may be defined by one or more implementation modules, as well as its subproject type (regular, iteration, iteration group, and product defect), a product and version, and multiple specifications.

A requirement, that is evaluated by users in DevSpec and well defined through Specification, is ready for implementation as soon as it is finalized and approved.

Specification is one of the key components in Product Design. It clearly defines the development goals. Before the implementation team begins working on a committed specification, Project Managers need to properly plan when the actual development work will commence. They need to allocate resources to work on the implementation of this specification. Project Managers also need to specify which Product Module will include this Specification.

A Product Backlog is created to facilitate smooth transition from the Design phase to the Planning and Implementation phase. It consists of a list of prioritized items that need to be developed in the next implementation cycle. As Specifications reach their final approval state, they are dropped into the Backlog folder, indicating Project Managers that these Specifications are ready for implementation.

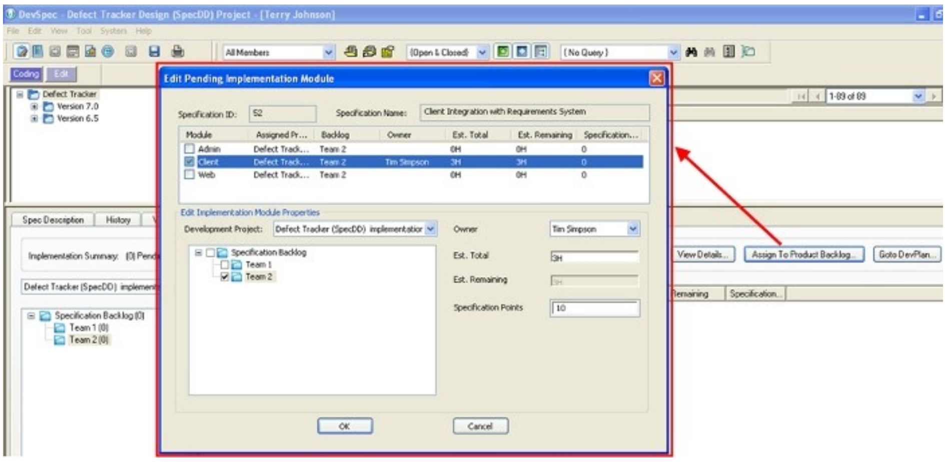


ID	Title	Status	Spec Owner
498	Move "Encoding" menu from Edit to Page	Functional Review	Dean Stewart
11	Auto-suggest & Auto-correct search box	Released	+For Review
25	Improve menu system	Developed	Terry Johnson
522	Allow users to subscribe to email notifications	Technical Review	Dean Stewart
523	Define email rules from admin	Functional Review	Judith Brown
499	Logo redesign	Ready to Implement	Matt Peterson

Implementation Summary: (1) Pending, (3) Assigned, (0) Closed

Implementation links	Backlog Path	Owner	Est. Rer
25: Improve menu system			
	\Team 1	Matt Peterson	6H

In DevSpec, the Backlog tab allows users to select a specification and easily add it to the appropriate Product Backlog folder. To do so, highlight a specification in the List View, go to the Backlog tab, select the appropriate Backlog folder and click "Assign to Product Backlog" button. You can also drag-and-drop a Specification from List View into one of the Specification Backlog Folder under Backlog.



Specification ID: 52 Specification Name: Client Integration with Requirements System

Module	Assigned Pr...	Backlog	Owner	Est. Total	Est. Remaining	Specification...
☐ Admin	Defect Track...	Team 2		0H	0H	0
☑ Client	Defect Track...	Team 2	Tim Simpson	3H	3H	0
☐ Web	Defect Track...	Team 2		0H	0H	0

Edit Implementation Module Properties

Development Project: Defect Tracker (SpecDD) implementatior

☐ Specification Backlog
☐ Team 1
☑ Team 2

Owner: Tim Simpson

Est. Total: 3H

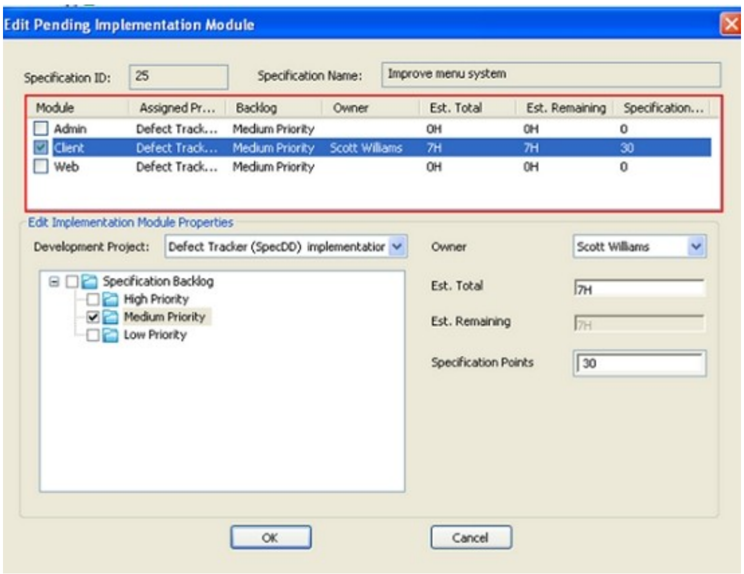
Est. Remaining: 3H

Specification Points: 10

3 Tracking Implemented Specifications

DevSpec further uses a concept called implementation modules, which can be used to more formally represent the relationship between areas of the product design and the implementation backlog.

While adding a Specification to the Backlog, users can also select different areas of development in which the particular Specification needs to be completed.



Edit Pending Implementation Module

Specification ID: 25

Specification Name: Improve menu system

Module	Assigned Pr...	Backlog	Owner	Est. Total	Est. Remaining	Specification...
□ Admin	Defect Track...	Medium Priority		0H	0H	0
☑ Client	Defect Track...	Medium Priority	Scott Williams	7H	7H	30
□ Web	Defect Track...	Medium Priority		0H	0H	0

Edit Implementation Module Properties

Development Project: Defect Tracker (SpecDD) implementatior

Owner: Scott Williams

Est. Total: 7H

Est. Remaining: 7H

Specification Points: 30

- □ Specification Backlog
  - □ High Priority
  - ☑ Medium Priority
  - □ Low Priority

In the DevSpec client, project team members may manage and track the implementation of specifications across multiple development projects in the Implementation Tracking tab.

Using controls in the Specification tab, project team members may manage and track implementation links between specifications and iterations (subprojects).

Implemented specifications are organized into three folders. Each specification is identified by a distinct icon.

> **Pending: The pending folder contains specifications that have been assigned to the specification backlog, but not yet committed to an iteration.**
>
> **Assigned: The assigned folder contains specifications that have been implemented in an iteration. A specification is automatically moved to the closed folder when the linked iterative subproject is closed.**
>
> **Closed:The closed folder shows implementations of the specification that have been implemented. A specification is automatically moved to the closed folder when the linked iterative subproject is closed.**

The Implementation Tracking tab displays high-level information about specifications that are in various stages of development, including the status (open or closed) of the specification, the status of linked development issues, owner, estimated time, estimated time remaining, and specification points.

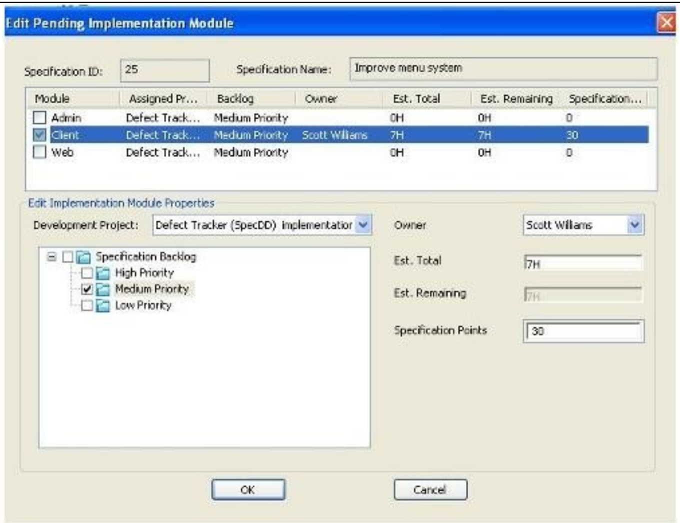
4 Managing Pending Implementations of Specifications

In DevSuite, a pending implementation identifies a implementation of a specification that has been approved for development, but which has not yet been committed to a particular iterative subproject (iteration or sprint).

A pending implementation is defined by a specification, implementation module, specification backlog folder, owner, estimated time, and optionally, specification points.

Using controls in the Edit Pending Implementation window, project team members may add implementation specifications - paired specifications and modules - to the specification backlog and define preliminary time estimates and specification points for the feature.

The Specification backlog is a hierarchical tree structure that enables development organizations to prioritize the implementation of features in a DevTrack development project.



Edit Pending Implementation Module

Specification ID: 25

Specification Name: Improve menu system

Module	Assigned Pr...	Backlog	Owner	Est. Total	Est. Remaining	Specification...
□ Admin	Defect Track...	Medium Priority		0H	0H	0
☑ Client	Defect Track...	Medium Priority	Scott Williams	7H	7H	30
□ Web	Defect Track...	Medium Priority		0H	0H	0

Edit Implementation Module Properties

Development Project: Defect Tracker (SpecDD) implementatior

- □ Specification Backlog
  - □ High Priority
  - ☑ Medium Priority
  - □ Low Priority

Owner: Scott Williams

Est. Total: 7H

Est. Remaining: 7H

Specification Points: 30

To add a pending implementation:

1. Select a specification in the specification list panel. Specification may be added to the specification backlog only if they are in an applicable open workflow state. Closed specifications may not be scheduled for development.

2. Select the Backlog tab in the specification detail panel.

3. Click the Edit Pending Implementation button. The Edit Pending Implementation window appears.

4. Select one or more implementation modules in the module list. An implementation module is a tool for organizing a distinct area of development - such as a platform (Windows, Linux), an application (thick client, thin client, smart client), plug-in, tool, widget, or any other development item. A specification may be scheduled for implementation in multiple modules.

5. Optional:To choose the development project, select a development project in the Development Project drop-down list.

6. Select a specification backlog folder in the specification backlog tree. The specification backlog is a hierarchical tree structure that enables development organizations to prioritize the implementation of features in a DevTrack development project.

7. Optional:To define a project team member as the owner of the implementation specification, select a project team member in the Owner drop-down list. A distinct owner may be defined for each pending implementation.

8.Input the estimated time required to complete the development task in the Est. Total text box. Numbers are automatically converted into days, weeks, and hours. For example, the number 90 will equate 2W 1D 2H. Distinct time estimates may be defined for each specification/module pairing.

9. Optional: To define specification points for the feature to be implemented, input the number in the Story Points text box. A specification point is a tool for rating the relative complexity of a feature - as defined by the specification and the module in which it is to be implemented - so that the team may better estimate the work that may be completed in an iteration. Distinct specification points may be defined for each specification/module pairing.

10. Click the OK button. One implementation is displayed in the Pending folder for each implementation module selected.

To edit a pending implementation specification:

1. Click the Edit Pending Implementation button. The Edit Pending Implementation window appears.

2.Select an implementation modules in the module list.

3. Update implementation properties.

4. Click the OK button. The update definitions are displayed in the implementation list.

To delete a pending implementation specification:

1. Select the Implementation Tracking tab in the specification detail panel.

2. Select a pending implementation in the implementation list.

3. Click the Delete button. Confirmation dialog box appears.

4. Click the Yes button. The pending implementation is deleted from the implementation list.

5 Adding Specifications to the Specification Backlog

In DevSuite, the specification backlog is a prioritized list of features that have been cued for development, but not committed to a specific iteration.

The specification backlog enables development organizations to control which specifications may be assigned to iterative subprojects - iterations - in development projects using state-based specification assignment rules.

Using controls in the Edit pending Implementation window, project team members may link paired specifications and modules to one or more specification backlog folders and define preliminary time estimate and specification points for the feature.

The Edit pending Implementation Module window displays a list of applicable implementation modules, preliminary owners, time estimates, and specification points for each implementation module, and the specification backlog tree for the selected development project.

To link a pending implementation specification:

1. Select a specification in the specification list panel. Specifications may be added to the specification backlog only if they are in an applicable open workflow state. Closed specifications may not be scheduled for development.

2.Select the Implementation Tracking tab in the specification detail panel.

3. Click the Edit Pending Implementation button. The Edit Pending Implementation window appears.

4. Select one or more implementation modules in the module list. An implementation module is a tool for organizing a distinct area of development - such as platform (Windows, Linux), an application (thick client, thin client, smart client), plug-in, tool, widget, or any other development item. A specification may be scheduled for implementation in multiple modules.

5. Optional:To choose the development project, select a development project in the Development Project drop-down list.

6. Select a specification backlog folder in the specification backlog tree. The specification backlog is a project-specific list of the features that have been cued for development, but not committed to a specific iteration.

7. Optional: To define a project team member as the owner of the implementation specification, select a project team member in the Owner drop-down list. A distinct owner may be defined for each specification/module pairing.

8. Input the estimated time required to complete the development task in the estimated Total text box. Numbers are automatically converted into days, weeks, and hours. For example, the number 90 will equate to 2W 1D 2H. Distinct time estimates may be defined for each specification/module pairing.

9. Optional: To define specification points for the feature to be implemented, input the number in the Story Points text box. A specification point is a tool for rating the relative complexity of a feature - as defined by the specification and the module in which it is to be implemented - so that the team can better estimate the work that may be completed in an iteration. Distinct specification points may be defined for each specification/module pairing.

10. Click the OK button. One implementation is displayed in the Pending folder for each implementation module selected.

To edit a pending implementation specification:

1.Click the Edit Pending Implementation button. The Edit Pending Implementation window appears.

2.Select an implementation modules in the module list.

3.Update implementation properties.

4.Click the OK button. The updated definitions are displayed in the implementation list.

To delete a pending implementation specificaiton:

1. Select the Implementation Tracking tab in the specification detail panel.

2. Select a pending implementation in the implementation list.

3. Click the Delete button. A confirmation dialog box appears.

4. Click the Yes button. The pending implementation is deleted from the implementation list.

6 Assigning Implementation to Subprojects

In DevSuite, an assigned implementation identifies a specification that has been assigned to development subproject. An assigned implementation link is a development task that has been assigned to a subproject. Assigned implementation links are defined by a owner, estimated time, and optionally, story points.

Using controls in the Assign Implementation Links to Subproject window, project team members may link specifications to development subprojects, define the owner of development issues, the time required to complete the development task, and the story points for that task.

A specification may be assigned to one subproject per implementation module. In DevSuite, an implementation module is a tool for organizing a distinct area of development - such as a platform (Windows, Linux), an application (thick client, thin client, smart client), plug-in, tool, widget, or any other development item.

A single specification may be implemented multiple times in many different areas of development. For example, the same feature may implemented in a Windows-based client and a browser-based client.

To assign an Implementation specification to a development subproject:

1.Select a specification in the specification list panel.

2.Select the Implementation Tracking tab of the specification detail panel.

3. Click the Assign Pending button. The Assign Implementation Module to Subproject window appears. Select one or more implementation modules in the implementation list. Pending implementations are selected by default.

4. Select a project.

5. Select a subproject.

6. Select a development project team member in the Owner drop-down list.

7. Define the estimated time to complete the development task in the Estimated Time text box.

8. Input the number of story points in the Story Points text box.

9. Click the Assign button. The Assign Implementation Module to Subproject window closes.

To delete an assigned implementation specification:

1. Select the Implementation Tracking tab in the specification detail panel.

2. Select an assigned implementation in the implementation list.

3. Click the Delete button. A confirmation dialog box appears.

4. Click the Yes button. The pending implementation is deleted from the implementation list.

7 Editing Assigned Implementation Specifications

Using controls in the Edit Implementation Links to Subproject window, project team members may link specifications to a development subproject and define the owner of development issues, the time required to complete the development task, and the story points for the task.

To edit an implementation specification:

1. Select a specification in the specification list panel.
2. Select the Implementation Tracking tab of the specification detail panel.
3. Click the Edit Existing button. The Edit Implementation window appears. The pending implementation is deleted from the implementation list.

To assign implementation specification to a different subproject:

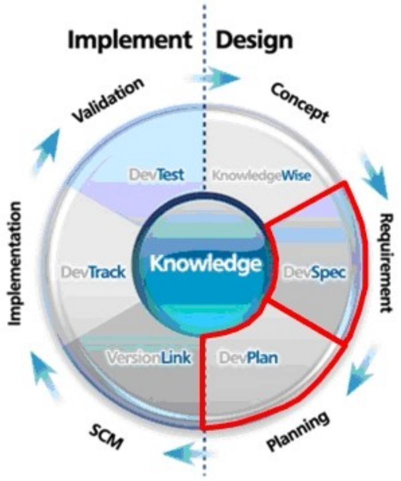
1. Select a specification in the specification list panel.
2. Select the Implementation Tracking tab of the specification detail panel.
3. Click the Edit Existing button. The Edit Implementation window appears. The pending implementation is deleted from the implementation list.

Chapter 7 - DevSuite Integration

TechExcel's DevSuite is a family of integrated application lifecycle management (ALM) tools. When DevSpec is integrated with DevPlan, DevTrack, and DevTest, specifications can be used to drive the entire application lifecycle: from design, to planning, to implementation, and to testing.

1 DevSpec and DevPlan as part of DevSuite

TechExcel DevSuite is a comprehensive Application Lifecycle Management Tool. It consists of integrated tools to manage different phases in the application lifecycle.



DevSuite consists of following integrated tools:

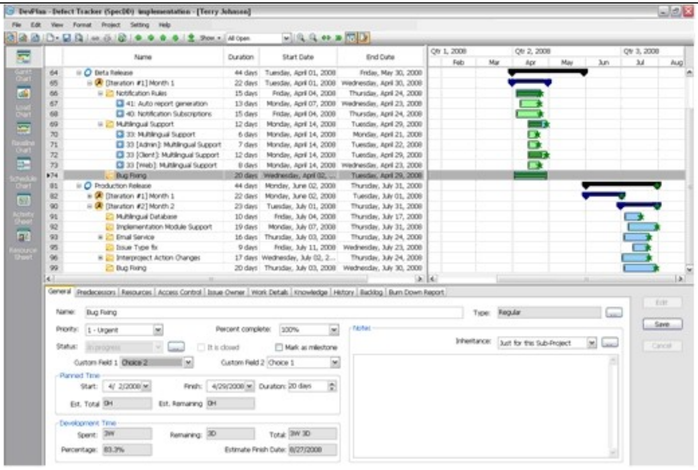
DevSpec (Requirement Management Tool)
DevPlan (Project Planning Tool)
DevTrack (Implementation Tool for Development Team)
DevTest (Test Management Tool)

1.1 DevPlan

DevPlan is DevSuite's project planning tool. DevPlan users, such as product and project managers, can ensure that a proposed initiative/feature completes within a defined time frame at a managed cost.

Some of the key activities of Product and Project Managers in DevPlan are listed below:

Schedule and define projects/sub-projects. Sub-project tree represents functional breakdown of projects.
Allocate timelines for Projects and define dependencies.
Allocate Resources for Project. Through resource allocation, estimate the man-hours required for the completion of a project or area of work.
Schedule events, such as meetings, brainstorming sessions, discussions, designer presentation, etc.
Make sure that knowledge, concepts, and design ideas are shared across all teams and properly linked to the development tasks.
Generate reports and charts to monitor progress and balance work load.
Adjust timelines based on design changes and other estimates.


DevPlan - Defect Tracker (SpecDD) implementation - [Terry Johnson]
File Edit View Format Project Setting Help
Show
All Open
Qtr 1, 2008
Qtr 2, 2008
Qtr 3, 2008
Feb Mar Apr May Jun Jul Aug
Gantt Chart
Load Chart
Baseline Chart
Schedule Chart
Activity Sheet
Resource Sheet
General Predecessors Resources Access Control Issue Owner Work Details Knowledge History Backlog Burn Down Report
Edit
Save
Cancel

	Name	Duration	Start Date	End Date
64	Beta Release	44 days	Tuesday, April 01, 2008	Friday, May 30, 2008
65	[Iteration #1] Month 1	22 days	Tuesday, April 01, 2008	Wednesday, April 30, 2008
66	Notification Rules	15 days	Friday, April 04, 2008	Thursday, April 24, 2008
67	41: Auto report generation	13 days	Monday, April 07, 2008	Wednesday, April 23, 2008
68	40: Notification Subscriptions	15 days	Friday, April 04, 2008	Thursday, April 24, 2008
69	Multilingual Support	12 days	Monday, April 14, 2008	Tuesday, April 29, 2008
70	33: Multilingual Support	6 days	Monday, April 14, 2008	Monday, April 21, 2008
71	33 [Admin]: Multilingual Support	7 days	Monday, April 14, 2008	Tuesday, April 22, 2008
72	33 [Client]: Multilingual Support	12 days	Monday, April 14, 2008	Tuesday, April 29, 2008
73	33 [Web]: Multilingual Support	8 days	Monday, April 14, 2008	Wednesday, April 23, 2008
74	Bug Fixing	20 days	Wednesday, April 02, ...	Tuesday, April 29, 2008
81	Production Release	44 days	Monday, June 02, 2008	Thursday, July 31, 2008
82	[Iteration #1] Month 1	22 days	Monday, June 02, 2008	Tuesday, July 01, 2008
90	[Iteration #2] Month 2	23 days	Tuesday, July 01, 2008	Thursday, July 31, 2008
91	Multilingual Database	10 days	Friday, July 04, 2008	Thursday, July 17, 2008
92	Implementation Module Support	19 days	Monday, July 07, 2008	Thursday, July 31, 2008
93	Email Service	16 days	Thursday, July 03, 2008	Thursday, July 24, 2008
95	Issue Type fix	9 days	Friday, July 11, 2008	Wednesday, July 23, 2008
96	Interproject Action Changes	17 days	Wednesday, July 02, 2...	Thursday, July 24, 2008
99	Bug Fixing	20 days	Thursday, July 03, 2008	Wednesday, July 30, 2008

Name: Bug Fixing
Type: Regular
Priority: 1 - Urgent
Percent complete: 100%
Status: In progress
□ It is closed
□ Mark as milestone
Custom Field 1 Choice 2
Custom Field 2 Choice 1
Notes
Inheritance: Just for this Sub-Project
Planned Time
Start: 4/ 2/2008
Finish: 4/29/2008
Duration: 20 days
Est. Total 0H
Est. Remaining 0H
Development Time
Spent: 3W
Remaining: 3D
Total: 3W 3D
Percentage: 83.3%
Estimate Finish Date: 8/27/2008

1.2 Need for DevSpec - DevPlan Integration

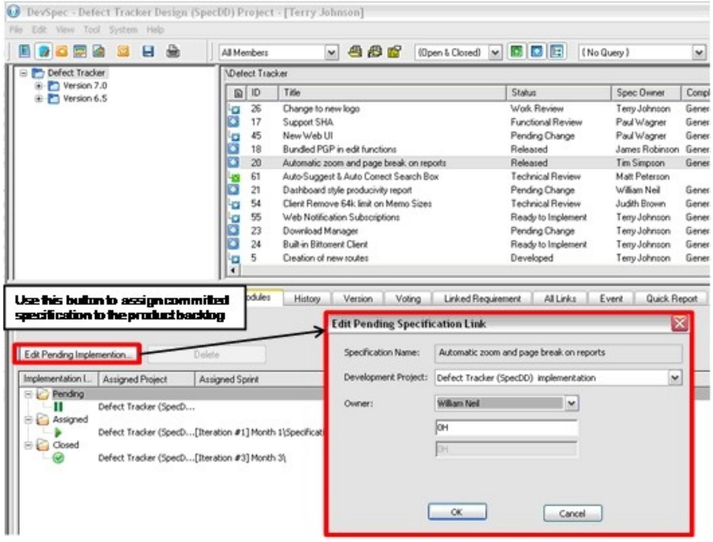
For DevPlan users to plan projects efficiently, it is essential that they clearly understand the proposed design, features, and specifications that will drive the future implementations. Since all requirements, design, and concepts are well tracked and managed in DevSpec, it would be nice to have a view or window to DevSpec from within DevPlan. This will empower Project Managers working in DevPlan to better understand the features and therefore better plan a project.

Generally, a product design consists of complete specifications that include:

Functional requirements and design documents
Product components and breakdown
Design parameters and feature specifications
Architecture and database design documents
Programming logic and QA test case documents
Business logic and user interface design
All these items are managed in DevSpec and serve as guidelines for developers during product implementation phase. So it is equally important to link these documents with appropriate development work items. DevSpec-DevPlan integration facilitates accomplishing this goal.

1.3 Understanding and Managing DevSpec and DevPlan Integration

When a Requirement/Specification in DevSpec is ready to be implemented, it is assigned to a Product Backlog through the Implementation Module feature.


DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]
File Edit View Tool System Help
All Members
(Open & Closed)
(No Query)
Defect Tracker
Version 7.0
Version 6.5
\Defect Tracker
odules History Version Voting Linked Requirement All Links Event Quick Report
Edit Pending Implemention...
Delete
OK
Cancel

ID	Title	Status	Spec Owner	Compl
26	Change to new logo	Work Review	Terry Johnson	Gener
17	Support SHA	Functional Review	Paul Wagner	Gener
45	New Web UI	Pending Change	Paul Wagner	Gener
18	Bundled PGP in edit functions	Released	James Robinson	Gener
20	Automatic zoom and page break on reports	Released	Tim Simpson	Gener
61	Auto-Suggest & Auto Correct Search Box	Technical Review	Matt Peterson	
21	Dashboard style producivity report	Pending Change	William Neil	Gener
54	Client Remove 64k limit on Memo Sizes	Technical Review	Judith Brown	Gener
55	Web Notification Subscriptions	Ready to Implement	Terry Johnson	Gener
23	Download Manager	Pending Change	Terry Johnson	Gener
24	Built-in Bittorrent Client	Ready to Implement	Terry Johnson	Gener
5	Creation of new routes	Developed	Terry Johnson	Gener

Use this button to assign commited specification to the product backlog

Implementation I...	Assigned Project	Assigned Sprint
Pending		
	Defect Tracker (SpecD...	
Assigned		
	Defect Tracker (SpecD...[Iteration #1] Month 1\Specificat	
Closed		
	Defect Tracker (SpecD...[Iteration #3] Month 3\	

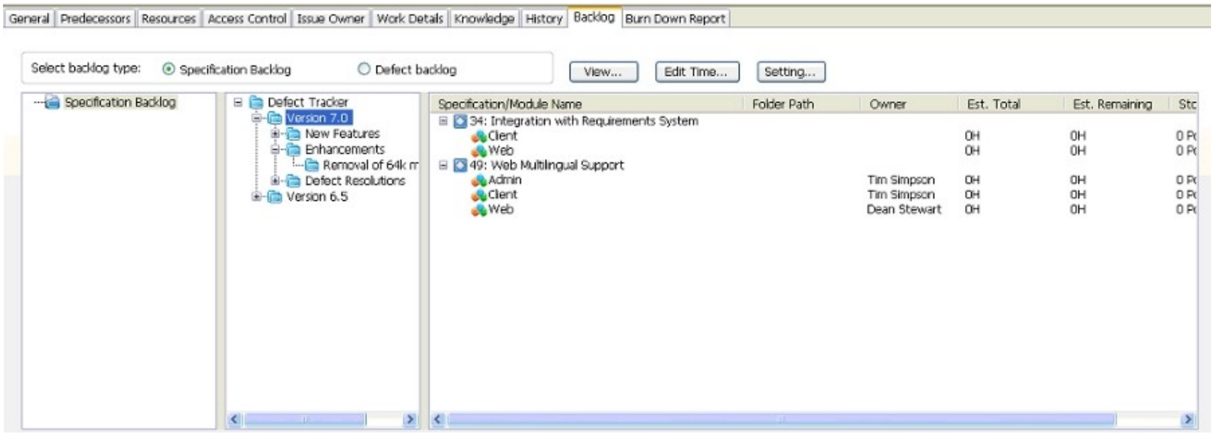
Edit Pending Specification Link
Specification Name: Automatic zoom and page break on reports
Development Project: Defect Tracker (SpecDD) implementation
Owner: William Neil
0H
0H

With the help of DevSpec-DevPlan integration, this Backlog is made available to the DevPlan users.

Specification Backlog in DevPlan

The specification backlog enables development organizations to organize and prioritize features - as defined by DevSpec specifications - that have been approved for development, but which have not yet been committed to a particular subproject.

Using the backlog, product and project managers may identify those features that are highest priority for the business, schedule those specifications for development by assigning them to a sub-project or a sprint. In DevPlan, this is done on the "Backlog" page for a highlighted sub-project. This requires simply dragging a Specification from Backlog and dropping it into a sub-project or sprint.

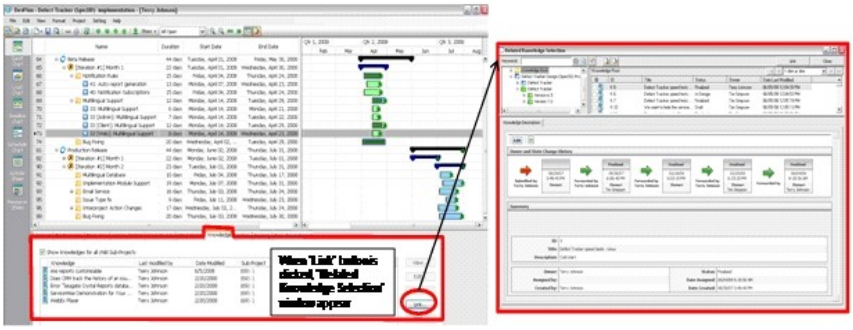


Select backlog type: ● Specification Backlog ○ Defect backlog

Specification/Module Name	Folder Path	Owner	Est. Total	Est. Remaining	Stc
34: Integration with Requirements System					
Client			0H	0H	0 P
Web			0H	0H	0 P
49: Web Multilingual Support					
Admin		Tim Simpson	0H	0H	0 P
Client		Tim Simpson	0H	0H	0 P
Web		Dean Stewart	0H	0H	0 P

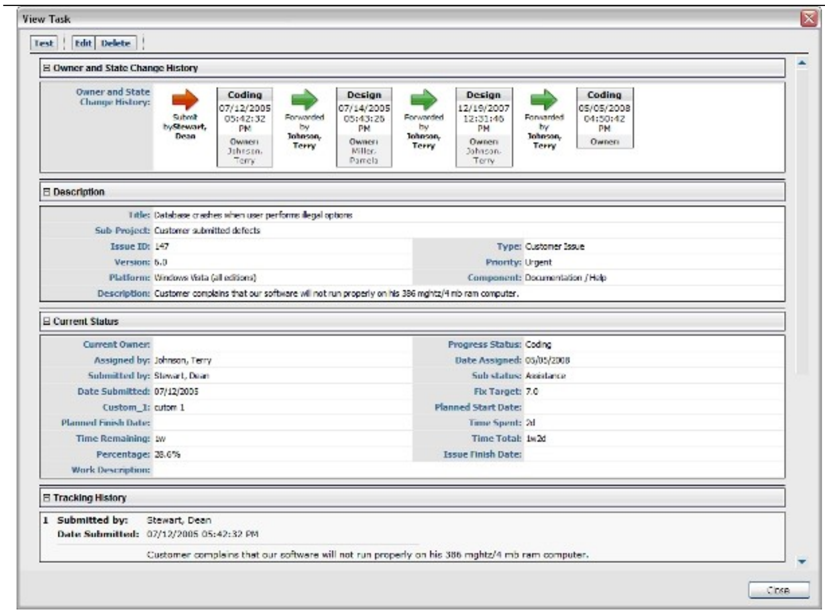
Knowledge sharing in DevPlan

DevPlan users can also link relevant Knowledge artifacts, managed in DevSpec, to the development project.


When 'Link' button is clicked, 'Related Knowledge Selection' window appear

Specification History and Work Details in DevPlan

DevPlan can also display the properties and history of a specification through the work details view. Project managers can view this information by selecting "View" on the Work Details tab.



⊟ Owner and State Change History

Owner and State Change History:

Submit by**Stewart, Dean**	Coding 07/12/2005 05:42:32 PM Owner: Johnson, Terry	Forwarded by **Johnson, Terry**	Design 07/14/2005 05:43:25 PM Owner: Miller, Pamela	Forwarded by **Johnson, Terry**	Design 12/19/2007 12:31:46 PM Owner: Johnson, Terry	Forwarded by **Johnson, Terry**	Coding 05/05/2008 04:50:42 PM Owner:	

⊟ Description

Title:	Database crashes when user performs illegal options		
Sub Project:	Customer submitted defects		
Issue ID:	147	Type:	Customer Issue
Version:	6.0	Priority:	Urgent
Platform:	Windows Vista (all editions)	Component:	Documentation / Help
Description:	Customer complains that our software will not run properly on his 386 mghtz/4 mb ram computer.		

⊟ Current Status

Current Owner:		Progress Status:	Coding
Assigned by:	Johnson, Terry	Date Assigned:	05/05/2008
Submitted by:	Stewart, Dean	Sub status:	Assistance
Date Submitted:	07/12/2005	Fix Target:	7.0
Custom_1:	cutom 1	Planned Start Date:	
Planned Finish Date:		Time Spent:	2d
Time Remaining:	1w	Time Total:	1w2d
Percentage:	28.6%	Issue Finish Date:	
Work Description:			

⊟ Tracking History

1 **Submitted by:** Stewart, Dean
Date Submitted: 07/12/2005 05:42:32 PM

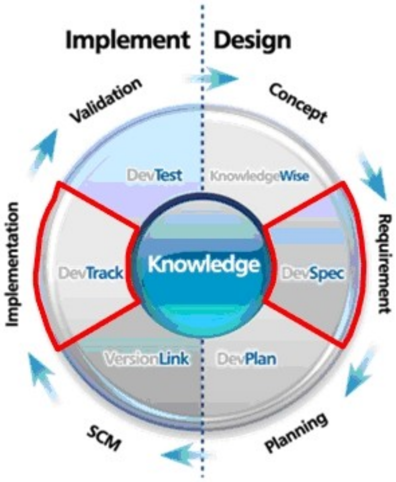
Customer complains that our software will not run properly on his 386 mghtz/4 mb ram computer.

Each Development work item that is created for this subproject will be linked to the specification driving the development work. So everyone on the development side and project planning side are always aware of clear requirements and in sync.

2 DevSpec and DevTrack as part of DevSuite

DevTrack is a tool to manage and track development work. It comprehensively tracks and manages any type of issue, from a new feature to IT activities to product defects. It consists of an intelligent transition-based workflow system that allows development teams to remain agile, yet react quickly to process and design changes.

The picture below shows DevSuite's ALM model and DevTrack's role in the process:

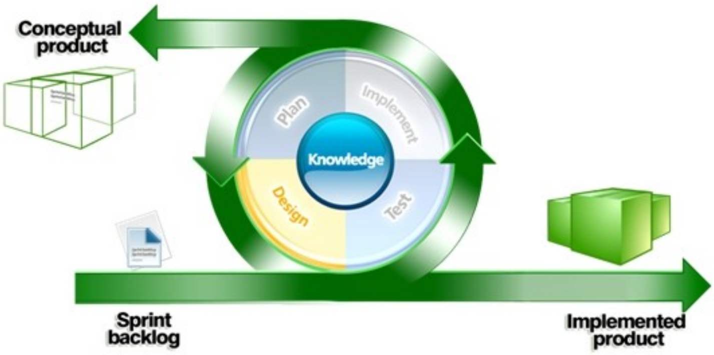


Requirements managed in DevSpec and shceduled in DevPlan actually get implemented in DevTrack. DevSuite provides a way to quantify requirements by defining specification for each requirement. Committed specifications/features are added to the Product Backlog. Project planning team then assigns the specifications to a project and schedules the implementation. Actual development of the features/specificaiton is carried out and tracked in DevTrack.

2.1 Need for DevSpec - DevTrack Integration

It is essential for the implementation team in DevTrack to clearly understand the features and concepts to make sure that the final implemented product precisely represents the designed product (or the conceptual product).

The picture below illustrates the relationship between the conceptual product and the implemented product in an iterative development model.



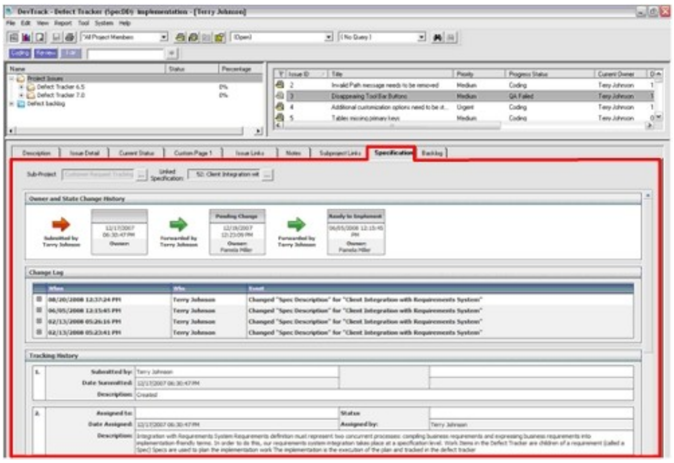
The conceptual product, is an illustration to quantitively represent the requirements and business logic of an application.

The implemented product is the end result of the implementation teams working towards a final deliverable product, but building upon the conceptual product.

Using this model, both the design team and the implemented team can always be in sync by sharing clear and common goals. This is facilitated through DevSpec - DevTrack integration.

2.2 Understanding DevSpec - DevTrack Integration

Requirements and specifications tracked and committed in DevSpec drive the implementation work in DevTrack. As each development item progresses through workflow towards completion, it is appropriately associated with a specification defined in DevSpec. The visibility to the specification details is facilitated by the "Specification" tab in DevTrack Client.



Name	Status	Percentage
Project Issues		
Defect Tracker 6.5		0%
Defect Tracker 7.0		0%
Defect backlog		

Issue ID	Title	Priority	Progress Status	Current Owner	D
2	Invalid Path message needs to be removed	Medium	Coding	Terry Johnson	1
3	Disappearing ToolBar Buttons	Medium	QA Failed	Terry Johnson	1
4	Additional customization options need to be st...	Urgent	Coding	Terry Johnson	1
5	Tables missing primary keys	Medium	Coding	Terry Johnson	0

Sub-Project: Customer Request Tracking

Linked Specification: 52: Client Integration wit

Owner and State Change History

Submitted by Terry Johnson → 12/17/2007 06:30:47 PM Owner: → Forwarded by Terry Johnson → Pending Change 12/19/2007 12:23:09 PM Owner: Pamela Miller → Forwarded by Terry Johnson → Ready to Implement 06/05/2008 12:15:45 PM Owner: Pamela Miller

Change Log

When	Who	Event
08/20/2008 12:37:24 PM	Terry Johnson	Changed "Spec Description" for "Client Integration with Requirements System"
06/05/2008 12:15:45 PM	Terry Johnson	Changed "Spec Description" for "Client Integration with Requirements System"
02/13/2008 05:26:16 PM	Terry Johnson	Changed "Spec Description" for "Client Integration with Requirements System"
02/13/2008 05:23:41 PM	Terry Johnson	Changed "Spec Description" for "Client Integration with Requirements System"

Tracking History

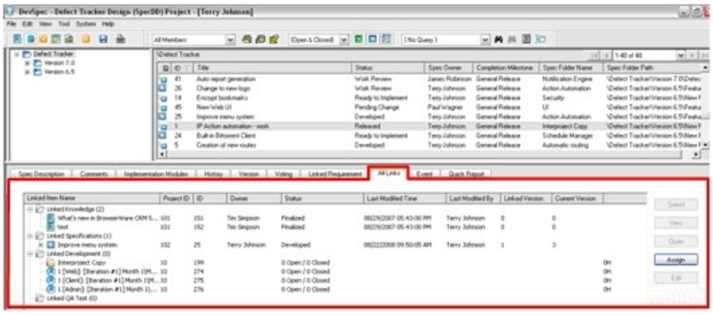
1. Submitted by: Terry Johnson
Date Submitted: 12/17/2007 06:30:47 PM
Description: Created

2. Assigned to:
Status:
Date Assigned: 12/17/2007 06:30:47 PM
Assigned by: Terry Johnson
Description: Integration with Requirements System Requirements definition must represent two concurrent processes: compiling business requirements and expressing business requirements into implementation-friendly terms. In order to do this, our requirements system integration takes place at a specification level. Work Items in the Defect Tracker are children of a requirement (called a Spec) Specs are used to plan the implementation work The implementation is the execution of the plan and tracked in the defect tracker

DevTrack users can quickly refer to the linked specification and get a better understanding of the design requirement.

On the subproject links tab for a highlighted development item, users can view, edit, and link all other linked documents, knowledge, requirements.

In DevSpec, users can view all the development items for a requirement/specification under the "All Links" tab -> Linked Development folder.



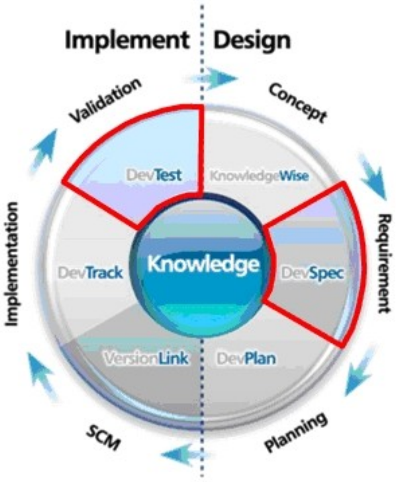
ID	Title	Status	Spec Owner	Completion Milestone	Spec Folder Name	Spec Folder Path
41	Auto report generation	Work Review	James Robinson	General Release	Notification Engine	\Defect Tracker\Version 7.0\Defec
26	Change to new logo	Work Review	Terry Johnson	General Release	Action Automation	\Defect Tracker\Version 6.5\Featu
14	Encrypt bookmarks	Ready to Implement	Terry Johnson	General Release	Security	\Defect Tracker\Version 6.5\New F
45	New Web UI	Pending Change	Paul Wagner	General Release	UI	\Defect Tracker\Version 6.5\Featu
25	Improve menu system	Developed	Terry Johnson	General Release	Action Automation	\Defect Tracker\Version 6.5\Featu...
1	IP Action automation - work	Released	Terry Johnson	General Release	Interproject Copy	\Defect Tracker\Version 6.5\New F
24	Built-in Bittorent Client	Ready to Implement	Terry Johnson	General Release	Schedule Manager	\Defect Tracker\Version 6.5\New F
5	Creation of new routes	Developed	Terry Johnson	General Release	Automatic routing	\Defect Tracker\Version 6.5\New F

Linked Item Name	Project ID	ID	Owner	Status	Last Modified Time	Last Modified By	Linked Version	Current Version	
Linked Knowledge (2)									
What's new in BrowserWare CRM 5...	101	151	Tim Simpson	Finalized	06/29/2007 05:43:00 PM	Terry Johnson	0	0	
test	101	152	Tim Simpson	Finalized	06/29/2007 05:43:00 PM	Terry Johnson	0	0	
Linked Specifications (1)									
Improve menu system	102	25	Terry Johnson	Developed	06/22/2008 09:50:05 AM	Terry Johnson	1	3	
Linked Development (0)									
Interproject Copy	10	199		0 Open / 0 Closed					0H
1 [Web]: [Iteration #1] Month 1(M...	10	274		0 Open / 0 Closed					0H
1 [Client]: [Iteration #1] Month 1(M...	10	275		0 Open / 0 Closed					0H
1 [Admin]: [Iteration #1] Month 1(...	10	276		0 Open / 0 Closed					0H
Linked QA Test (0)									

3 DevSpec and DevTest as part of DevSuite

TechExcel DevTest is an integrated test management solution that is specifically designed to manage the entire software test lifecycle. DevTest gives you total control of every aspect of your testing process from test planning and team management to analyzing your test results. DevTest enables you to create and manage release and test cycles, plan and assign test tasks to your testing teams, execute test coverage, and submit product defects - all from a single application.

The picture below shows the DevSuite model and the significant role played by DevSpec and DevTest in the ALM process:



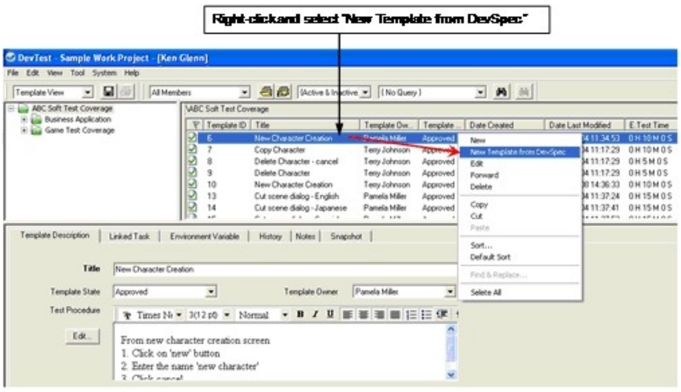
DevTest-centric approach to test management ensures that the test team has access to the complete picture of the project they are testing for each phase of the test lifecycle. From test design to test execution and analysis, DevTest provides test teams with a connection to the knowledge items they need to implement an effective test management process.

In addition to the global knowledge access, DevTest team can also refer to the requirements/specifications driving the implementation and validation work. The requirements and specifications managed in DevSpec are easily linked to the test cases managed in DevTest.

3.1 Understanding DevSpec - DevTest Integration

After feature specifications get committed in DevSpec, QA Test Team can start preparing test coverage for these committed feature specifications. Users in DevTest can easily create a test complete from an existing requirement/specification.

A new Test Complete can be created by right-clicking in the Template List View and selecting "New Template from DevSpec" and select a requirement/specification.



Right-clickand select "New Template from DevSpec"

DevTest - Sample Work Project - [Ken Glenn]

Template View | All Members | (Active & Inactive) | (No Query)

\ABC Soft Test Coverage

Template ID	Title	Template Ow...	Template ...	Date Created	Date Last Modified	E.Test Time
6	New Character Creation	Pamela Miller	Approved		4 11:34:53	0 H 10 M 0 S
7	Copy Character	Terry Johnson	Approved		4 11:17:29	0 H 10 M 0 S
8	Delete Character - cancel	Terry Johnson	Approved		4 11:17:29	0 H 5 M 0 S
9	Delete Character	Terry Johnson	Approved		4 11:17:29	0 H 5 M 0 S
10	New Character Creation	Terry Johnson	Approved		0 14:36:33	0 H 10 M 0 S
13	Cut scene dialog - English	Pamela Miller	Approved		4 11:37:24	0 H 15 M 0 S
14	Cut scene dialog - Japanese	Pamela Miller	Approved		4 11:37:41	0 H 15 M 0 S

Title: New Character Creation

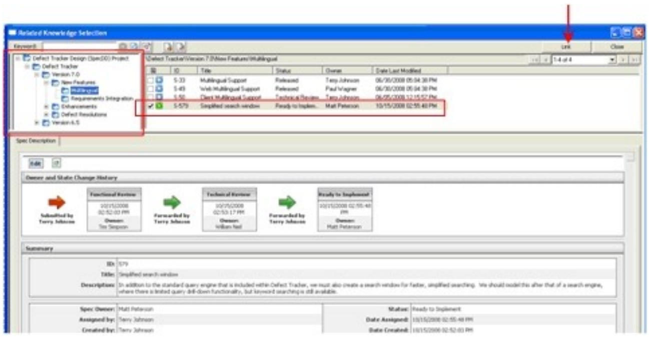
Template State: Approved

Template Owner: Pamela Miller

Test Procedure:

From new character creation screen
1. Click on 'new' button
2. Enter the name 'new character'
3. Click cancel

An existing test template in DevTest can be linked to the requirement/specification defined in DevSpec. All test tasks that are generated from a test template under Test Cycle automatically show the requirement/specification driving the testing and validation work. DevTest users with special privileges can also edit the linked requirement/specification.



Related Knowledge Selection

\Defect Tracker\Version 7.0\New Features\Multilingual

ID	Title	Status	Owner	Date Last Modified
S-33	Multilingual Support	Released	Terry Johnson	06/30/2008 05:04:38 PM
S-49	Web Multilingual Support	Released	Paul Wagner	06/30/2008 05:04:38 PM
S-50	Client Multilingual Support	Technical Review	Terry Johnson	06/05/2008 12:15:57 PM
S-579	Simplified search window	Ready to Implem...	Matt Peterson	10/15/2008 02:55:40 PM

Owner and State Change History

Submitted by Terry Johnson → Functional Review 10/15/2008 02:52:03 PM Owner: Tim Simpson → Forwarded by Terry Johnson → Technical Review 10/15/2008 02:53:17 PM Owner: William Neil → Forwarded by Terry Johnson → Ready to Implement 10/15/2008 02:55:48 PM Owner: Matt Peterson

Summary

ID: 579

Title: Simplified search window

Description: In addition to the standard query engine that is included within Defect Tracker, we must also create a search window for faster, simplified searching. We should model this after that of a search engine, where there is limited query drill-down functionality, but keyword searching is still available.

Spec Owner: Matt Peterson

Assigned by: Terry Johnson

Created by: Terry Johnson

Status: Ready to Implement

Date Assigned: 10/15/2008 02:55:48 PM

Date Created: 10/15/2008 02:52:03 PM

QA team in DevTest can generate reports on test tasks/templates and also include information on linked requirements/specifications. Product management and design teams in DevSpec can easily refer to all the test coverage defined for the highlighted requirement/specification.

Chapter 8 - Advanced Features

This chapter covers the advanced features of DevSpec:

Change requests

Baseline

Events

Voting

Mind Map

1 Change Request

Any software development project should welcome changes to the design at any point during the development lifecycle. Customers and other stakeholders may demand a requirement/specification change at any time. A requirement/specification management tool should be able to accommodate these changes with minimal impact, minimal risk, and minimal overhead.

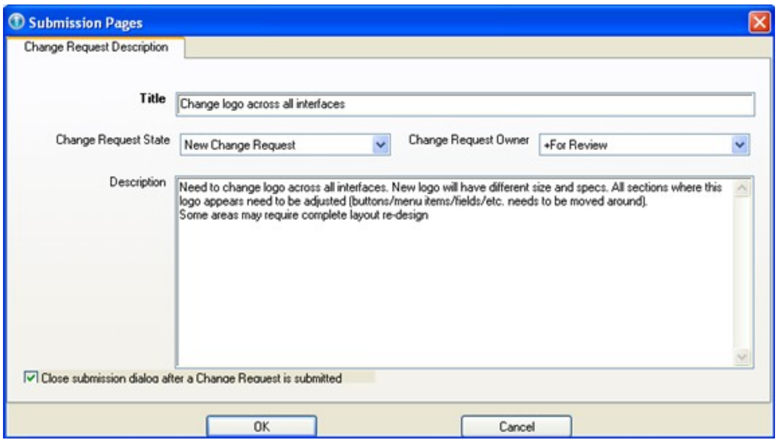
In DevSpec requirement changes are completely tracked and controlled through a change process. The change request view in DevSpec allows users to define a change and link all affected specifications. Users can be notified about the suggested change, and meetings and events can be scheduled to gather stakeholders' opinion. The suggested change follows a strict workflow to make sure each intricate detail is considered and everyone is on the same page and well prepared for the suggested change to the product Design.

Just as with other views in DevSpec, the change request view is divided into three panels:
List panel
Folder panel
Detail panel

1.1 Submitting a Change Request

To submit a change request:

Switch to the change request view by clicking the button in the tool bar
Click the button in the tool bar, or select*File >Submit New…*in the menu bar



Title: Change logo across all interfaces

Change Request State: New Change Request

Change Request Owner: +For Review

Description: Need to change logo across all interfaces. New logo will have different size and specs. All sections where this logo appears need to be adjusted (buttons/menu items/fields/etc. needs to be moved around).
Some areas may require complete layout re-design

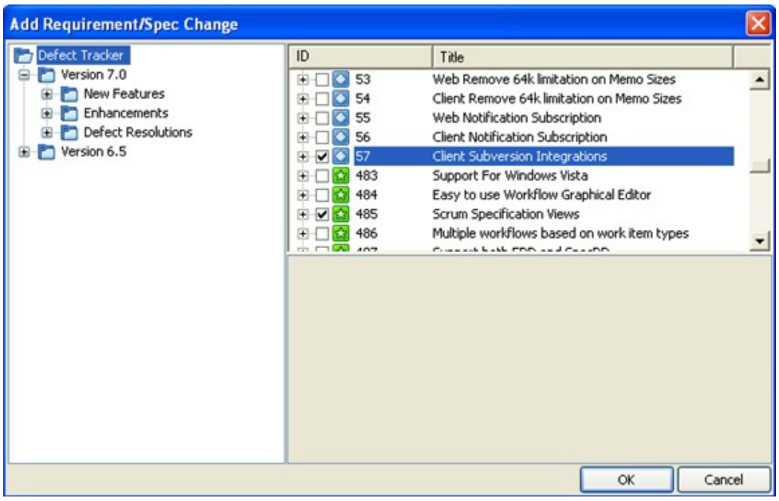
☑ Close submission dialog after a Change Request is submitted

Complete the change request submission form and click*OK.*

1.2 Linking Specifications to a Change Request

To link specifications and requirements to a change request:

In the change request view, right-click on a change request in the list panel
Select*Add Requirement/Spec Change*.
In the new dialog, highlight a specification folder in the tree panel and select the requirements/specifications that will be effected by the proposed change



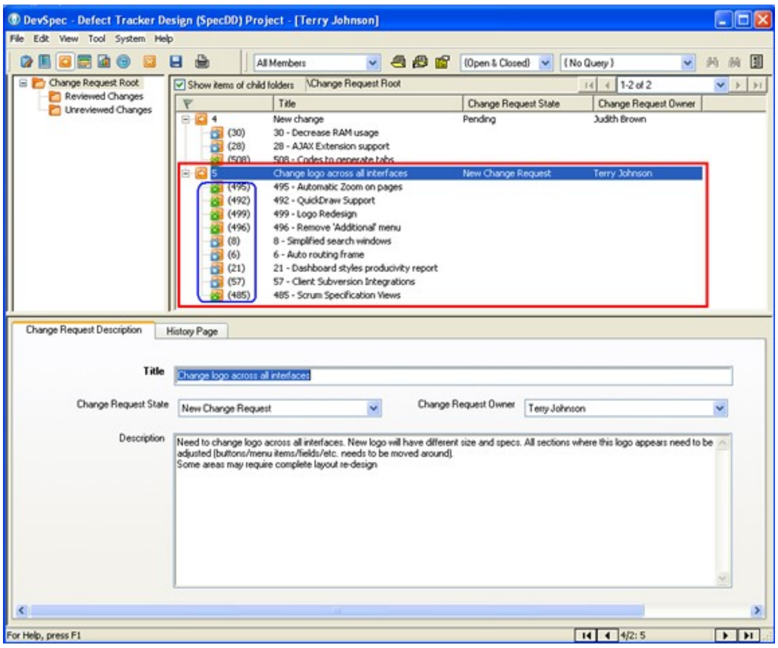
Add Requirement/Spec Change

- Defect Tracker
  - Version 7.0
    - New Features
    - Enhancements
    - Defect Resolutions
  - Version 6.5

	ID	Title
□	◆ 53	Web Remove 64k limitation on Memo Sizes
□	◆ 54	Client Remove 64k limitation on Memo Sizes
□	◆ 55	Web Notification Subscription
□	◆ 56	Client Notification Subscription
☑	◆ 57	Client Subversion Integrations
□	★ 483	Support For Windows Vista
□	★ 484	Easy to use Workflow Graphical Editor
☑	★ 485	Scrum Specification Views
□	★ 486	Multiple workflows based on work item types

Note:The ◆ icon represents a*Specification*and the ★ icon represents a*Requirement*.

Click OK
All linked requirements and specifications are displayed under the change request item in the list panel



DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]

☑ Show items of child folders \Change Request Root 1-2 of 2

	Title	Change Request State	Change Request Owner
4	New change	Pending	Judith Brown
(30)	30 - Decrease RAM usage		
(28)	28 - AJAX Extension support		
(508)	508 - Codes to generate tabs		
5	Change logo across all interfaces	New Change Request	Terry Johnson
(495)	495 - Automatic Zoom on pages		
(492)	492 - QuickDraw Support		
(499)	499 - Logo Redesign		
(496)	496 - Remove 'Additional' menu		
(8)	8 - Simplified search windows		
(6)	6 - Auto routing frame		
(21)	21 - Dashboard styles producivity report		
(57)	57 - Client Subversion Integrations		
(485)	485 - Scrum Specification Views		

Change Request Description | History Page

Title: Change logo across all interfaces

Change Request State: New Change Request

Change Request Owner: Terry Johnson

Description: Need to change logo across all interfaces. New logo will have different size and specs. All sections where this logo appears need to be adjusted (buttons/menu items/fields/etc. needs to be moved around).
Some areas may require complete layout re-design

For Help, press F1 4/2: 5

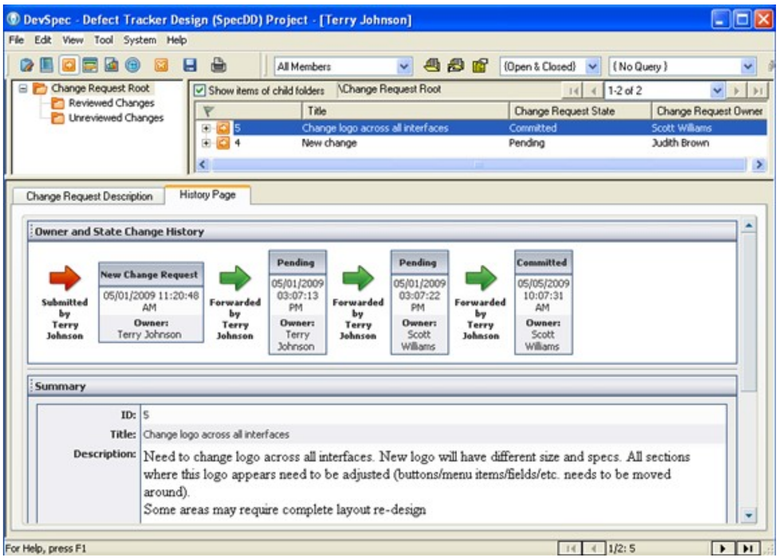
Note:In the list panel of the change request view, the ◆ icon represents a*linked specification*and the ★ icon represents a*linked requirement*(linked to a change request item).

1.3 The Change Request Process

After a change request has been submitted and requirements/specifications are linked, users can view the change request details in the detail panel. The detail panel for a change request consists of two tabs:
Change Request Description
History Page
The*Change Request Description*tab provides users with general information about the selected change

request record. It consists of standard fields, such as *Title*, *Description*, *State* and *Owner*. DevSpec Administrators can add more fields to this tab as needed.

The *History Page* provides a summary and tracking history of the selected change request record. It consists of multiple sections that can be included or excluded by each user as needed.



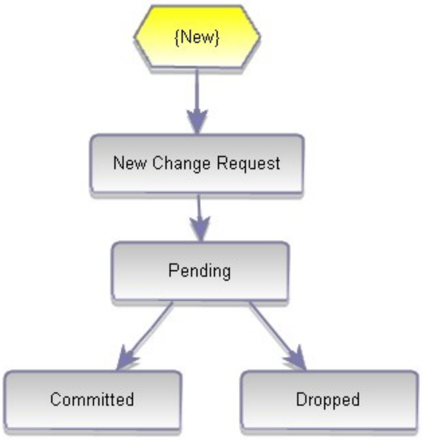
	Title	Change Request State	Change Request Owner
5	Change logo across all interfaces	Committed	Scott Williams
4	New change	Pending	Judith Brown

ID: 5

Title: Change logo across all interfaces

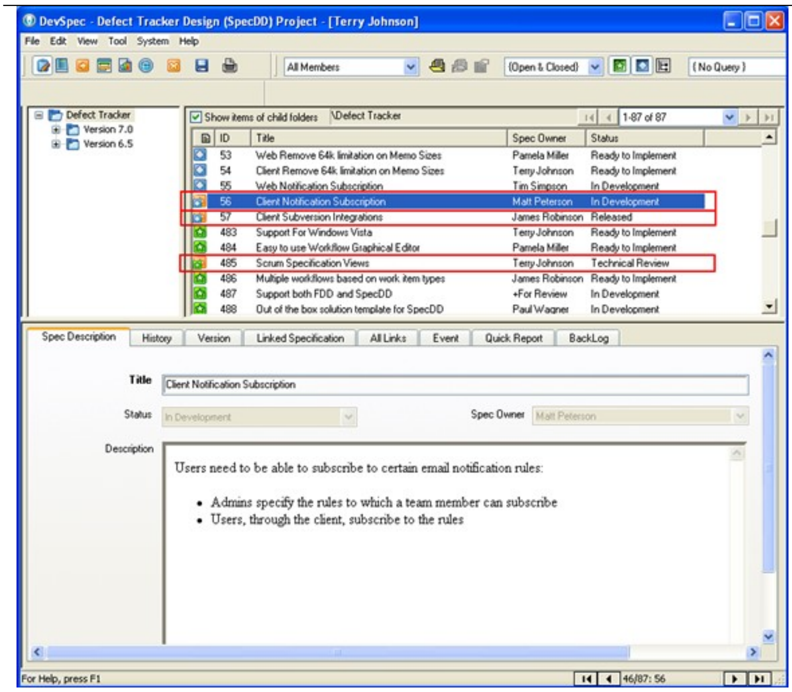
Description: Need to change logo across all interfaces. New logo will have different size and specs. All sections where this logo appears need to be adjusted (buttons/menu items/fields/etc. needs to be moved around).
Some areas may require complete layout re-design

Each change request follows the workflow defined by the DevSpec administrator. A typical change request workflow is shown in the picture below.



In the diagram above, *New Change Request* and *Pending* are workflow states where a change request remains open. *Committed* and *Dropped* are workflow states where a change request remains closed.

All linked requirements and specifications are flagged and rendered read-only as long as the change request item remains open. While in the specification view, users can easily differentiate these linked specifications from other regular, non-linked specifications.



ID	Title	Spec Owner	Status
53	Web Remove 64k limitation on Memo Sizes	Pamela Miller	Ready to Implement
54	Client Remove 64k limitation on Memo Sizes	Terry Johnson	Ready to Implement
55	Web Notification Subscription	Tim Simpson	In Development
56	Client Notification Subscription	Matt Peterson	In Development
57	Client Subversion Integrations	James Robinson	Released
483	Support For Windows Vista	Terry Johnson	Ready to Implement
484	Easy to use Workflow Graphical Editor	Pamela Miller	Ready to Implement
485	Scrum Specification Views	Terry Johnson	Technical Review
486	Multiple workflows based on work item types	James Robinson	Ready to Implement
487	Support both FDD and SpecDD	+For Review	In Development
488	Out of the box solution template for SpecDD	Paul Wagner	In Development

Title: Client Notification Subscription

Status: In Development

Spec Owner: Matt Peterson

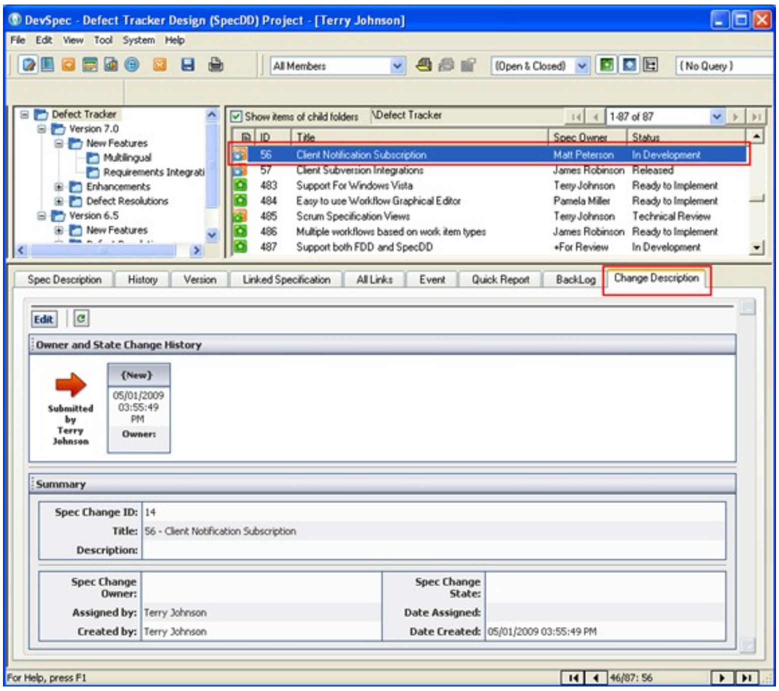
Description:

Users need to be able to subscribe to certain email notification rules:

- Admins specify the rules to which a team member can subscribe
- Users, through the client, subscribe to the rules

Note:In the list panel of the specification view, the icon represents a specification linked to a change request item and the icon represents a requirement linked to a change request.

For each specification that is currently linked to a change request item, users will see an additional tab in the list panel of the specification view. This tab is labeled*Change Description*. In this tab users can view the details of the change request.



ID	Title	Spec Owner	Status
56	Client Notification Subscription	Matt Peterson	In Development
57	Client Subversion Integrations	James Robinson	Released
483	Support For Windows Vista	Terry Johnson	Ready to Implement
484	Easy to use Workflow Graphical Editor	Pamela Miller	Ready to Implement
485	Scrum Specification Views	Terry Johnson	Technical Review
486	Multiple workflows based on work item types	James Robinson	Ready to Implement
487	Support both FDD and SpecDD	+For Review	In Development

Owner and State Change History

Submitted by Terry Johnson

{New}
05/01/2009 03:55:49 PM
Owner:

Summary

Spec Change ID:	14
Title:	56 - Client Notification Subscription
Description:	

Spec Change Owner:		Spec Change State:	
Assigned by:	Terry Johnson	Date Assigned:	
Created by:	Terry Johnson	Date Created:	05/01/2009 03:55:49 PM

1.4 Change Flagging Events

For each specification that is linked to a change request, a special type of event can be created to track the progress with respect to the suggested change. These events are called**change flagging events**in DevSpec, and are different from other regular events. Change flagging events are only applicable to any specification or requirement that is linked to a proposed change request. When defining an event template, the DevSpec Administrator can customize the label of such special events.

The purpose of a change flagging event is to notify users (including the owner of the linked specification) about the proposed change request. Users can conduct meetings to review the proposed change and the impact on the linked specifications. After evaluating and making the appropriate adjustments to the linked specification, the event owner confirms that the specification is ready for the proposed change. This is done by closing the change flagging event.

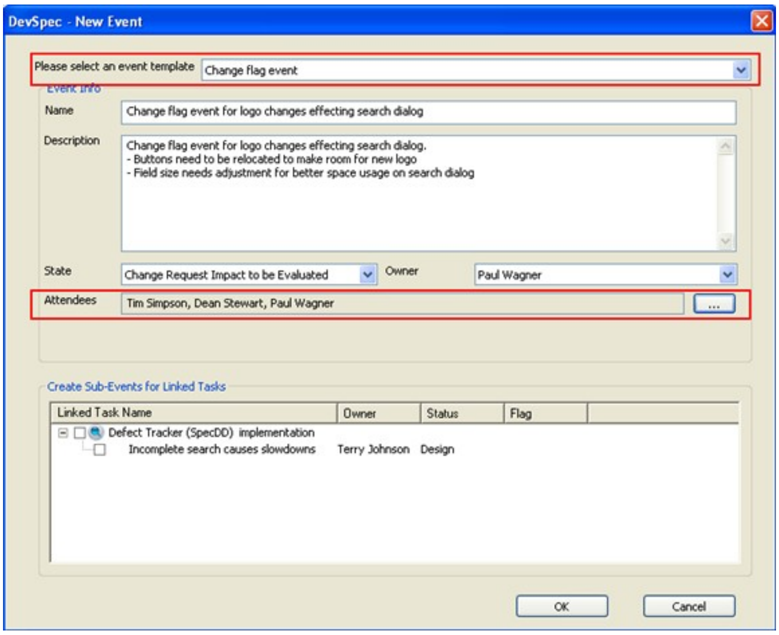
To create a change flagging event:

In the list panel of the change request view, highlight a specification linked to the proposed change request item

In the*Detail*panel, click the*Event*tab

Select the*Change Event*radio button

Click the*New Event*button. The*New Event*dialog pops up



DevSpec - New Event

Please select an event template: Change flag event

Event Info

Name: Change flag event for logo changes effecting search dialog

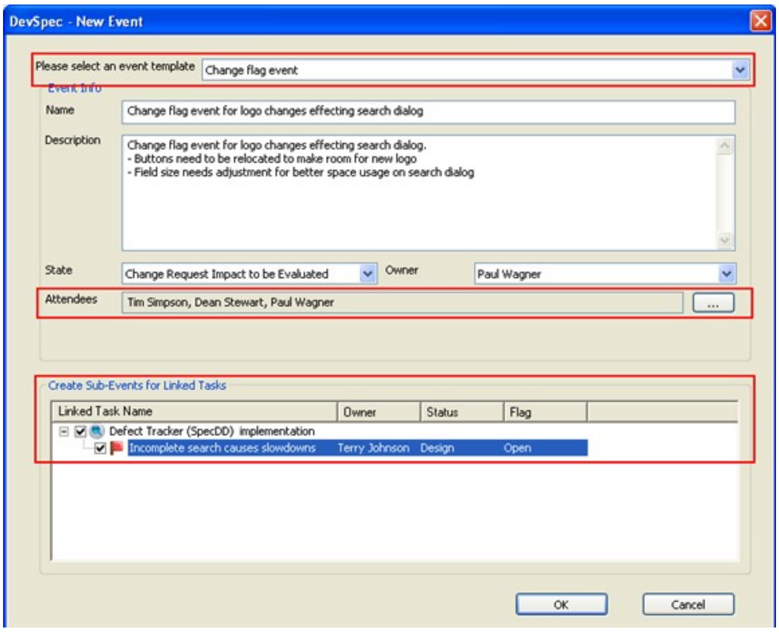
Description: Change flag event for logo changes effecting search dialog.
- Buttons need to be relocated to make room for new logo
- Field size needs adjustment for better space usage on search dialog

State: Change Request Impact to be Evaluated
Owner: Paul Wagner

Attendees: Tim Simpson, Dean Stewart, Paul Wagner

Create Sub-Events for Linked Tasks

Linked Task Name	Owner	Status	Flag
□ Defect Tracker (SpecDD) implementation			
□ Incomplete search causes slowdowns	Terry Johnson	Design	



DevSpec - New Event

Please select an event template: Change flag event

Event Info

Name: Change flag event for logo changes effecting search dialog

Description: Change flag event for logo changes effecting search dialog.
- Buttons need to be relocated to make room for new logo
- Field size needs adjustment for better space usage on search dialog

State: Change Request Impact to be Evaluated
Owner: Paul Wagner

Attendees: Tim Simpson, Dean Stewart, Paul Wagner

Create Sub-Events for Linked Tasks

Linked Task Name	Owner	Status	Flag
☑ Defect Tracker (SpecDD) implementation			
☑ Incomplete search causes slowdowns	Terry Johnson	Design	Open

Select a change flagging event from the dropdown list at the top

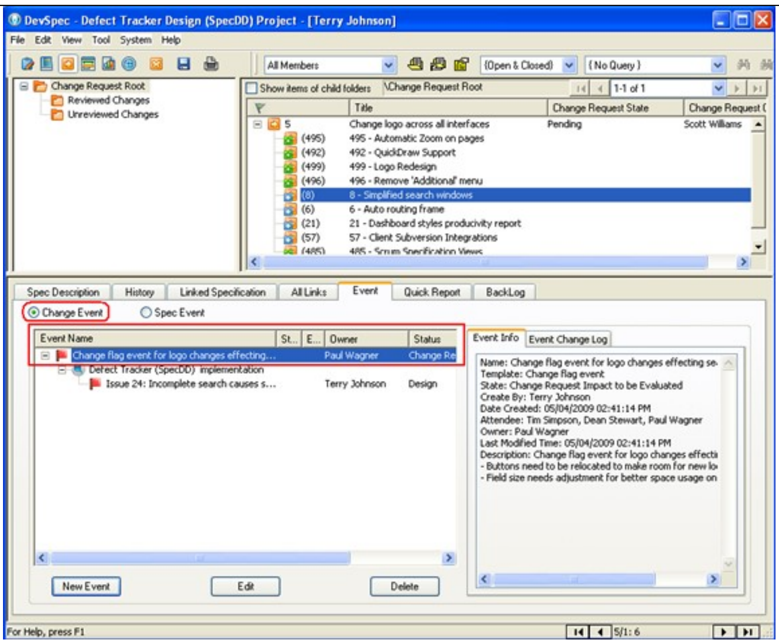
Complete other fields on the form

Users can also choose a list of attendees for this event meeting. To do so, click the ... button

Users can also create sub-events for any development tasks that are linked to the selected specification. This is done under the section labeled*Create Sub-Events for Linked Tasks*

Click the*OK*button

The*Event*tab now displays all change flagging events, along with any sub-events for linked development tasks



	Title	Change Request State	Change Request (
5	Change logo across all interfaces	Pending	Scott Williams
(495)	495 - Automatic Zoom on pages		
(492)	492 - QuickDraw Support		
(499)	499 - Logo Redesign		
(496)	496 - Remove 'Additional' menu		
(8)	8 - Simplified search windows		
(6)	6 - Auto routing frame		
(21)	21 - Dashboard styles producivity report		
(57)	57 - Client Subversion Integrations		
(485)	485 - Scrum Specification Views		

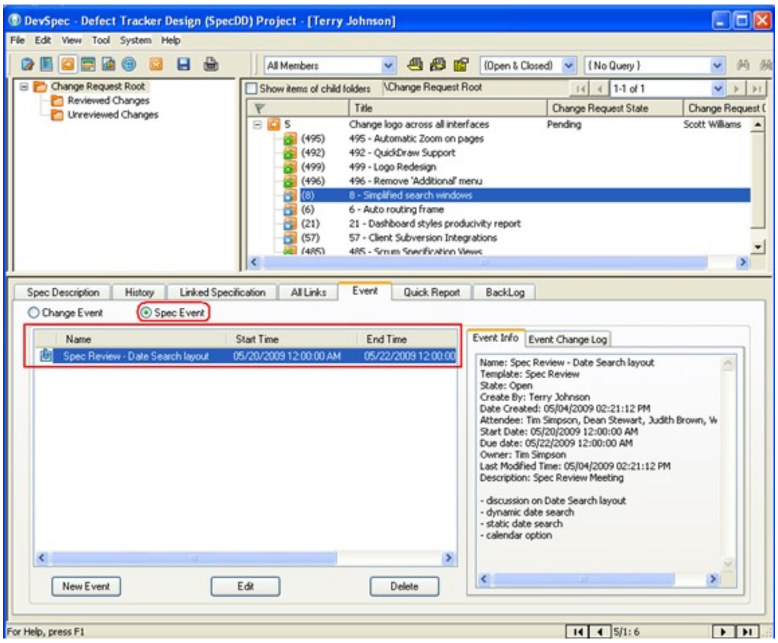
● Change Event ○ Spec Event

Event Name	St...	E...	Owner	Status
Change flag event for logo changes effecting...			Paul Wagner	Change Re
Defect Tracker (SpecDD) implementation				
Issue 24: Incomplete search causes s...			Terry Johnson	Design

Name: Change flag event for logo changes effecting se
Template: Change flag event
State: Change Request Impact to be Evaluated
Create By: Terry Johnson
Date Created: 05/04/2009 02:41:14 PM
Attendee: Tim Simpson, Dean Stewart, Paul Wagner
Owner: Paul Wagner
Last Modified Time: 05/04/2009 02:41:14 PM
Description: Change flag event for logo changes effecti
- Buttons need to be relocated to make room for new lo
- Field size needs adjustment for better space usage on

Note:All change flagging events are indicated by the 🚩 icon in the*Event*tab.

On the*Event*tab, users can select the*Spec Event*radio button to view any regular events associated with the highlighted linked specification



	Title	Change Request State	Change Request (
5	Change logo across all interfaces	Pending	Scott Williams
(495)	495 - Automatic Zoom on pages		
(492)	492 - QuickDraw Support		
(499)	499 - Logo Redesign		
(496)	496 - Remove 'Additional' menu		
(8)	8 - Simplified search windows		
(6)	6 - Auto routing frame		
(21)	21 - Dashboard styles producivity report		
(57)	57 - Client Subversion Integrations		
(485)	485 - Scrum Specification Views		

○ Change Event ● Spec Event

Name	Start Time	End Time
Spec Review - Date Search layout	05/20/2009 12:00:00 AM	05/22/2009 12:00:00

Name: Spec Review - Date Search layout
Template: Spec Review
State: Open
Create By: Terry Johnson
Date Created: 05/04/2009 02:21:12 PM
Attendee: Tim Simpson, Dean Stewart, Judith Brown, W
Start Date: 05/20/2009 12:00:00 AM
Due date: 05/22/2009 12:00:00 AM
Owner: Tim Simpson
Last Modified Time: 05/04/2009 02:21:12 PM
Description: Spec Review Meeting

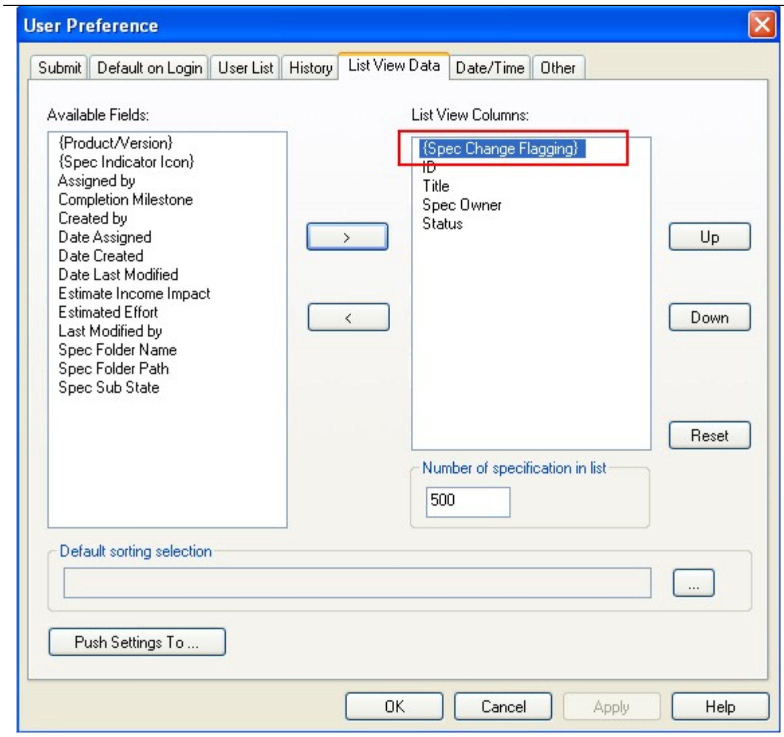
- discussion on Date Search layout
- dynamic date search
- static date search
- calendar option

Note:For more details on events, please see section 3 in this chapter,*Events*.

Identifying Specifications with Change Flagging Events

Users can easily differentiate between regular specifications and specifications with change flagging events in the list panel of the specification view. Each user must add an extra specification property before utilizing this feature:

1. In the menu bar, select*System*>*User Preferences*...
2. In the new dialog, go to the*List View Data*tab
3. Add*{Spec Change Flagging}*to the*List View Columns*on the right side



Available Fields:

{Product/Version}
{Spec Indicator Icon}
Assigned by
Completion Milestone
Created by
Date Assigned
Date Created
Date Last Modified
Estimate Income Impact
Estimated Effort
Last Modified by
Spec Folder Name
Spec Folder Path
Spec Sub State

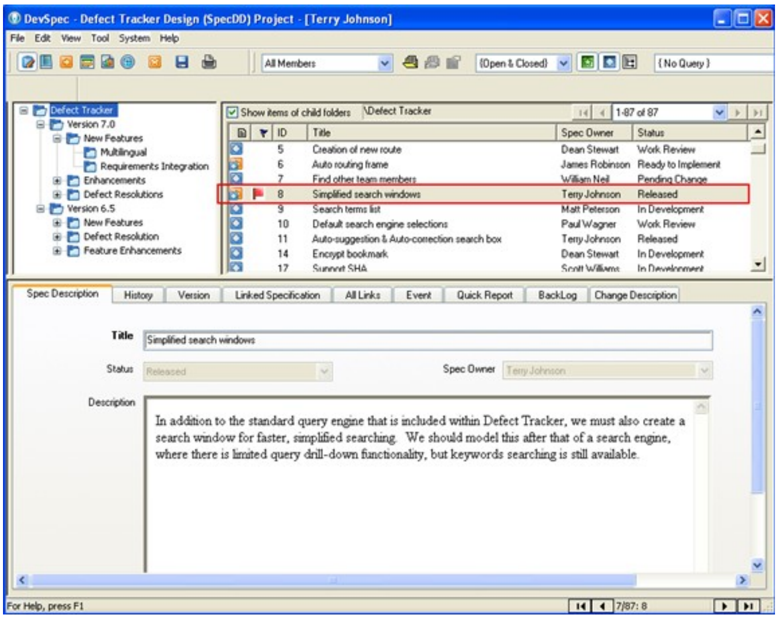
List View Columns:

{Spec Change Flagging}
ID
Title
Spec Owner
Status

Number of specification in list: 500

Default sorting selection:

Now such specifications with change flagging events will be identified with a flag icon in the list view.



☑ Show items of child folders \Defect Tracker 1-87 of 87

	ID	Title	Spec Owner	Status
	5	Creation of new route	Dean Stewart	Work Review
	6	Auto routing frame	James Robinson	Ready to Implement
	7	Find other team members	William Neil	Pending Change
🚩	8	Simplified search windows	Terry Johnson	Released
	9	Search terms list	Matt Peterson	In Development
	10	Default search engine selections	Paul Wagner	Work Review
	11	Auto-suggestion & Auto-correction search box	Terry Johnson	Released
	14	Encrypt bookmark	Dean Stewart	In Development
	17	Support SHA	Scott Williams	In Development

Title: Simplified search windows

Status: Released

Spec Owner: Terry Johnson

Description: In addition to the standard query engine that is included within Defect Tracker, we must also create a search window for faster, simplified searching. We should model this after that of a search engine, where there is limited query drill-down functionality, but keywords searching is still available.

Note:For more details on events, please see section 3 in this chapter,*Events*.

1.5 Change Request Workflow vs. Specification Change Workflow

When a change request is open, there are three separate workflows involved:

Specification workflow: Specifications follow the specification workflow defined by the DevSpec administrator.

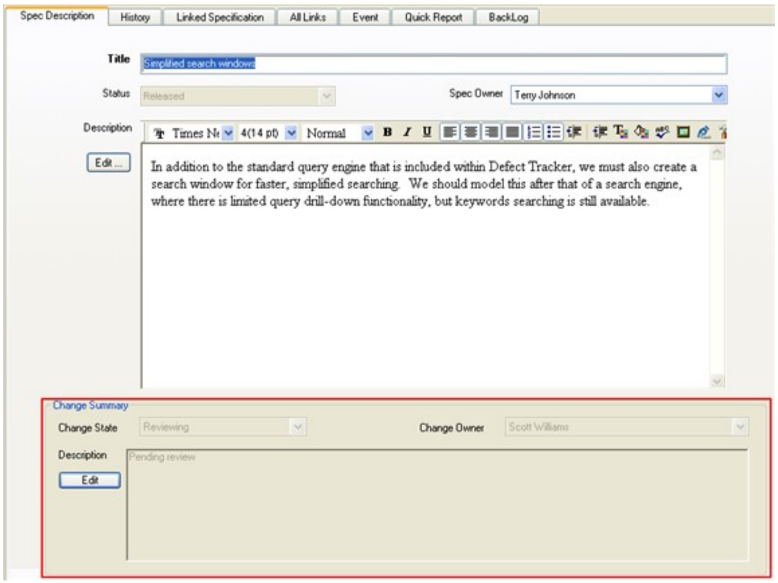
Change request workflow: Change request items follow the change request workflow defined the DevSpec administrator.

Specification change workflow: Specifications linked to a change request follow the specification change workflow while the change request is open.

An open change request item can be linked to multiple specifications. Since each of these linked specifications can potentially be in different states in the specification workflow, until the change request item is closed or committed, all its linked specifications are frozen from further modifications. During this period, each linked specification follows the specification change workflow.

To edit the linked specifications and the specification change workflow:

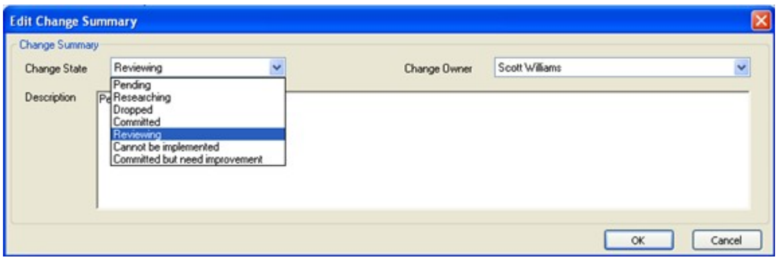
In the list view of the change request view, highlight a specification linked to the proposed change request item.
In the detail panel, click the*Spec Description*tab.



Title: Simplified search windows
Status: Released
Spec Owner: Terry Johnson
Description: In addition to the standard query engine that is included within Defect Tracker, we must also create a search window for faster, simplified searching. We should model this after that of a search engine, where there is limited query drill-down functionality, but keywords searching is still available.

Change Summary
Change State: Reviewing
Change Owner: Scott Williams
Description: Pending review

Click the*Edit*button inside the*Change Summary*section



Change Summary
Change State: Reviewing
Change Owner: Scott Williams
Description: Pe

Users can edit the*Spec Change Description*,*Owner*and*State*.
Click*OK*button.

1.6 Specification History vs. Specification Change History

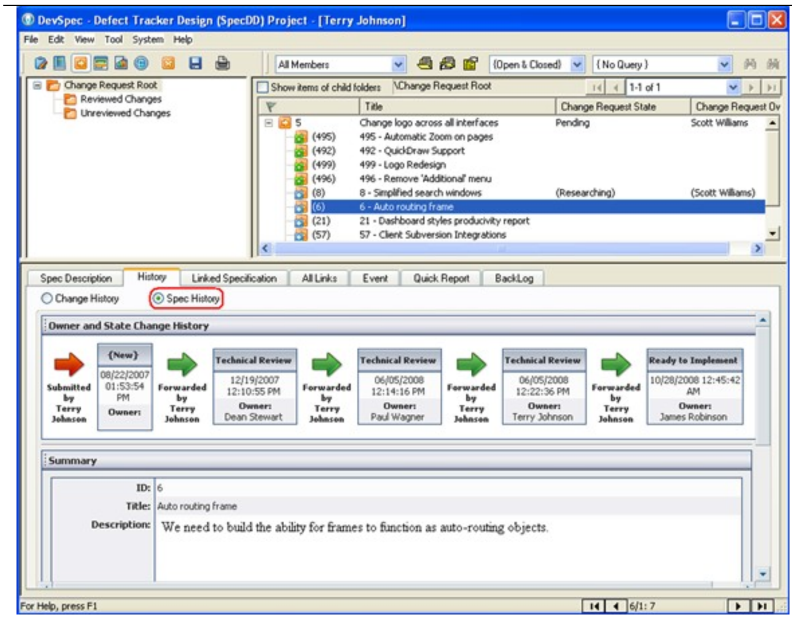
All specifications linked to a change request item maintain two workflows: the specification workflow and the specification change workflow.

The specification workflow is frozen as soon as a specification gets included in the proposed change request. This specification workflow remains dormant as long as the change request item remains open. While the specification workflow remains dormant, DevSpec allows users to track changes to each linked specification via a separate workflow, called the specification change workflow.

Users can easily view both workflows for each specification on the*History*tab in the detail panel of the change request view.

To view the history of a specification linked to a change request item:

In the list panel of the change request view, highlight a specification linked to the proposed change request item.
In the detail panel, click the*History*tab.
To view the specification workflow history, click the*Spec History*radio button.



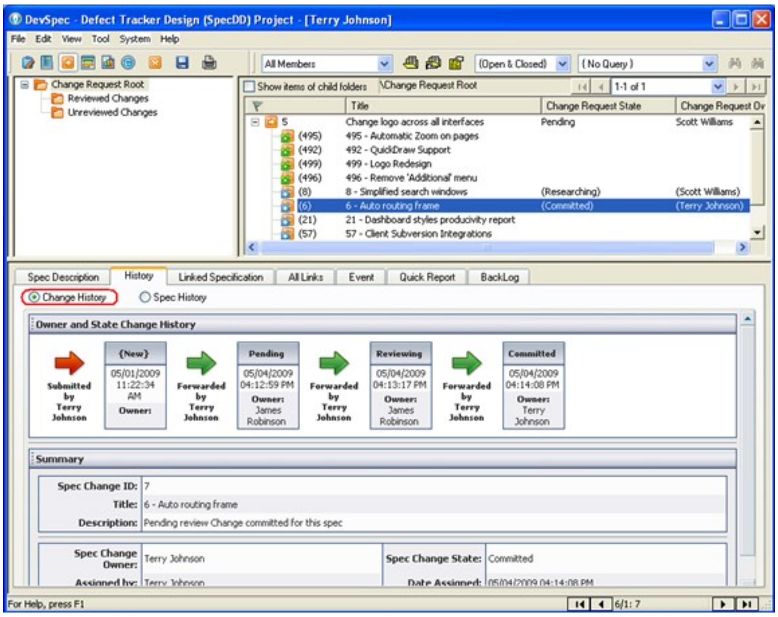
\Change Request Root

	Title	Change Request State	Change Request Ov
5	Change logo across all interfaces	Pending	Scott Williams
(495)	495 - Automatic Zoom on pages		
(492)	492 - QuickDraw Support		
(499)	499 - Logo Redesign		
(496)	496 - Remove 'Additional' menu		
(8)	8 - Simplified search windows	(Researching)	(Scott Williams)
(6)	6 - Auto routing frame		
(21)	21 - Dashboard styles producivity report		
(57)	57 - Client Subversion Integrations		

Submitted by Terry Johnson → {New} 08/22/2007 01:53:54 PM Owner: → Forwarded by Terry Johnson → Technical Review 12/19/2007 12:10:55 PM Owner: Dean Stewart → Forwarded by Terry Johnson → Technical Review 06/05/2008 12:14:16 PM Owner: Paul Wagner → Forwarded by Terry Johnson → Technical Review 06/05/2008 12:22:36 PM Owner: Terry Johnson → Forwarded by Terry Johnson → Ready to Implement 10/28/2008 12:45:42 AM Owner: James Robinson

ID: 6
Title: Auto routing frame
Description: We need to build the ability for frames to function as auto-routing objects.

To view the specification change workflow history, click the *Change History* radio button.



\Change Request Root

	Title	Change Request State	Change Request Ov
5	Change logo across all interfaces	Pending	Scott Williams
(495)	495 - Automatic Zoom on pages		
(492)	492 - QuickDraw Support		
(499)	499 - Logo Redesign		
(496)	496 - Remove 'Additional' menu		
(8)	8 - Simplified search windows	(Researching)	(Scott Williams)
(6)	6 - Auto routing frame	(Committed)	(Terry Johnson)
(21)	21 - Dashboard styles producivity report		
(57)	57 - Client Subversion Integrations		

Submitted by Terry Johnson → {New} 05/01/2009 11:22:34 AM Owner: → Forwarded by Terry Johnson → Pending 05/04/2009 04:12:59 PM Owner: James Robinson → Forwarded by Terry Johnson → Reviewing 05/04/2009 04:13:17 PM Owner: James Robinson → Forwarded by Terry Johnson → Committed 05/04/2009 04:14:08 PM Owner: Terry Johnson

Spec Change ID: 7
Title: 6 - Auto routing frame
Description: Pending review Change committed for this spec

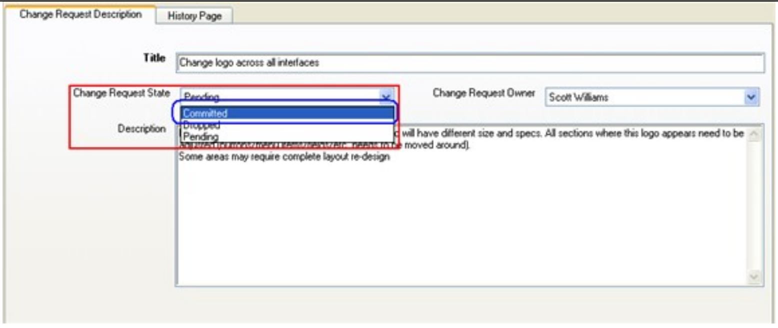
Spec Change Owner: Terry Johnson
Spec Change State: Committed
Assigned by: Terry Johnson
Date Assigned: 05/04/2009 04:14:08 PM

1.7 Committing/Closing a Change Request

Before a change request item can be closed, the owner of the change request item must make sure all linked requirements and specifications have been reviewed and are ready for the proposed change to be committed. Owners of each linked requirement/specification must close all change flagging events and other meeting events prior to change request closure.

To close a change request:

In the list panel of the change request view, highlight a change request item.
In the detail panel, click the *Change Request Description* tab.
Select a closing state from the *State* dropdown list (e.g. select “Committed”).



Change Request Description | History Page

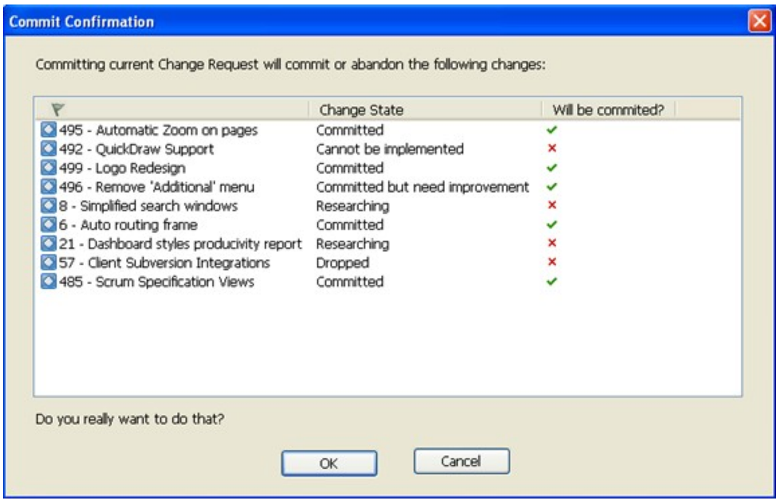
Title: Change logo across all interfaces

Change Request State: Pending (Committed / Dropped / Pending)

Change Request Owner: Scott Williams

Description: c will have different size and specs. All sections where this logo appears need to be moved around).
Some areas may require complete layout re-design

In the newly opened *Commit Conformation* dialog, review the status of all linked requirements and specifications.



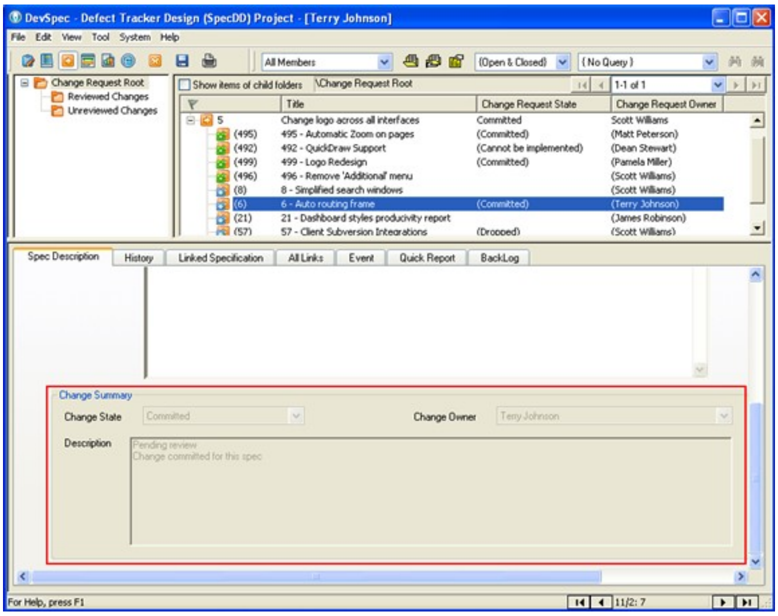
Commit Confirmation

Committing current Change Request will commit or abandon the following changes:

	Change State	Will be commited?
495 - Automatic Zoom on pages	Committed	✓
492 - QuickDraw Support	Cannot be implemented	✗
499 - Logo Redesign	Committed	✓
496 - Remove 'Additional' menu	Committed but need improvement	✓
8 - Simplified search windows	Researching	✗
6 - Auto routing frame	Committed	✓
21 - Dashboard styles producivity report	Researching	✗
57 - Client Subversion Integrations	Dropped	✗
485 - Scrum Specification Views	Committed	✓

Do you really want to do that?

Click the *OK* button to confirm and commit the change, or click the *Cancel* button to go back.
After the change request item is closed, the *Change Summary* section for each linked specification is rendered read-only, and the specification change workflow is frozen.



DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]

	Title	Change Request State	Change Request Owner
5	Change logo across all interfaces	Committed	Scott Williams
(495)	495 - Automatic Zoom on pages	(Committed)	(Matt Peterson)
(492)	492 - QuickDraw Support	(Cannot be implemented)	(Dean Stewart)
(499)	499 - Logo Redesign	(Committed)	(Pamela Miller)
(496)	496 - Remove 'Additional' menu		(Scott Williams)
(8)	8 - Simplified search windows		(Scott Williams)
(6)	6 - Auto routing frame	(Committed)	(Terry Johnson)
(21)	21 - Dashboard styles producivity report		(James Robinson)
(57)	57 - Client Subversion Integrations	(Dropped)	(Scott Williams)

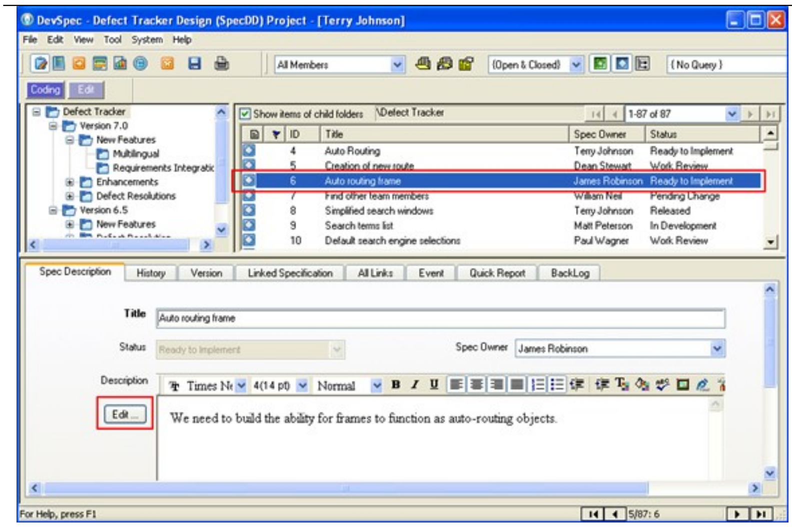
Change Summary

Change State: Committed

Change Owner: Terry Johnson

Description: Pending review
Change committed for this spec

All specifications linked to the committed change request item return to their prior states in the specification workflow, and all specification fields become active again (editable).



☑ Show items of child folders \Defect Tracker 1-87 of 87

ID	Title	Spec Owner	Status
4	Auto Routing	Terry Johnson	Ready to Implement
5	Creation of new route	Dean Stewart	Work Review
6	Auto routing frame	James Robinson	Ready to Implement
7	Find other team members	William Neil	Pending Change
8	Simplified search windows	Terry Johnson	Released
9	Search terms list	Matt Peterson	In Development
10	Default search engine selections	Paul Wagner	Work Review

Title: Auto routing frame

Status: Ready to Implement

Spec Owner: James Robinson

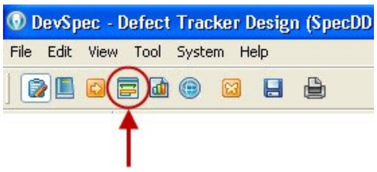
Description: We need to build the ability for frames to function as auto-routing objects.

Edit ...

2 Baseline

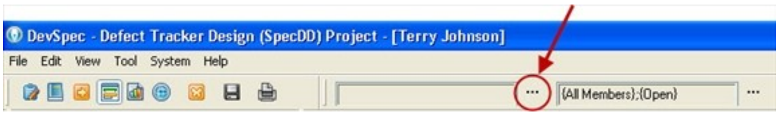
As a project progresses through workflow states, new specifications may be added, some specifications may be omitted, and some specification definitions may be changed. By creating a baseline, a set of specifications, users are able to “take a snapshot” of the project design at any point of time during the development lifecycle, and then compare it with future designs. This allows teams to measure the deviation of the product design from the original goal, and foresee the direction in which the product design may be heading.

To use the baseline feature in DevSpec, switch to the baseline view by clicking on the*Select Baseline View*button in the tool bar (only after the project administrator has enabled baseline support for the current project).

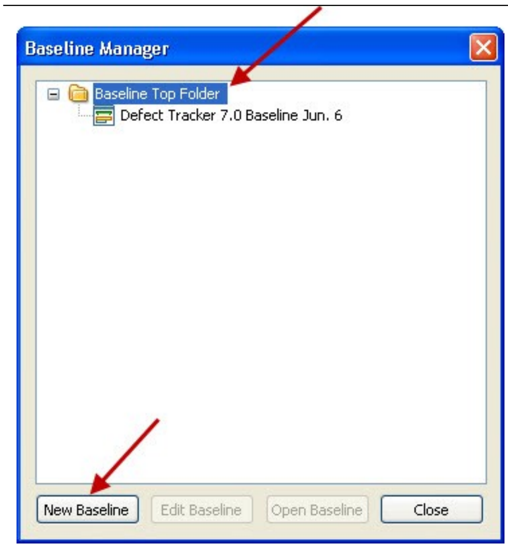


2.1 Creating a New Baseline

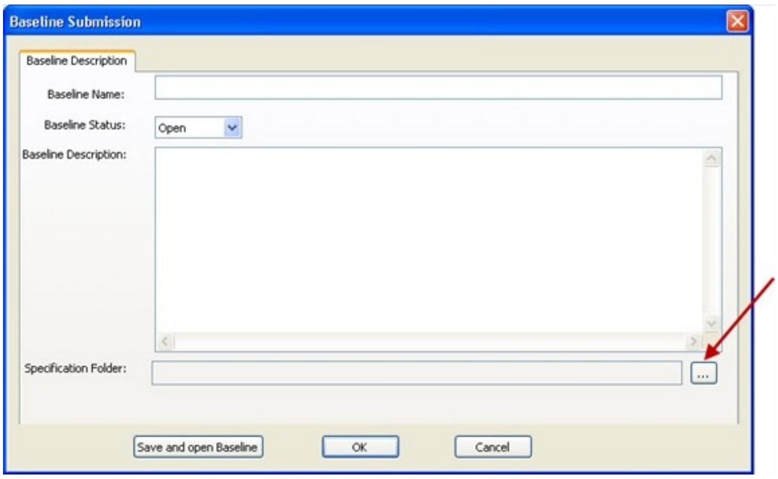
1. In the baseline view, click the ellipsis button (…) in the tool bar. This will open the*Baseline Manager*dialog.


DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]
File
Edit
View
Tool
System
Help
...
{All Members};{Open}
...

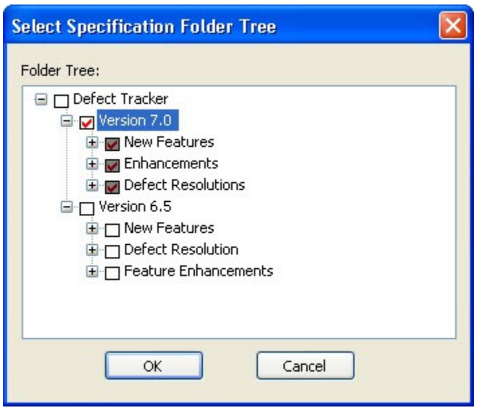
2. In the*Baseline Manager*dialog, highlight*Baseline Top Folder*, and click the*New Baseline*button. This will open the*Baseline Submission*dialog.



3. In the*Baseline Submission*dialog, the appropriate folder(s) need to be selected first. To do so, click the ellipses button next to the*Specification Folder*control. This will open the*Select Specification Folder Tree*dialog.



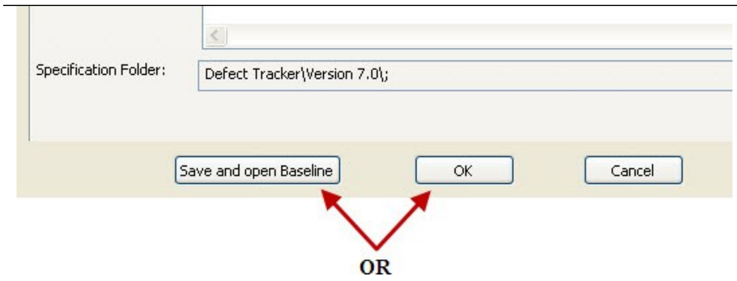
4. In the*Select Specification Folder Tree*dialog, select the appropriate folder(s) for the new baseline and press the*OK*button. Since a baseline is a set of specifications, this action will cause the system to capture all current details for all specification within the selected folders.



In the screenshot above, where the*Version 7.0*folder and each of its child folders are selected, all specifications belonging to these checked folders will be included in the baseline.

5. Once back in the*Baseline Submission*dialog, define the new baseline name, status, and description.

6. Click the*Save and open baseline*to save the new baseline, close all dialogs, and view the saved baseline in the detail panel; OR click the*OK*button to save and return to the*Baseline Manager*dialog.

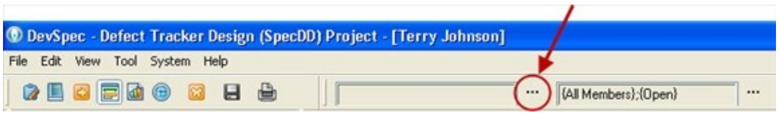

Specification Folder: Defect Tracker\Version 7.0\;

***Note*: A new baseline cannot be saved if the*Specification Folder*and*Baseline Name*fields are not filled out.**

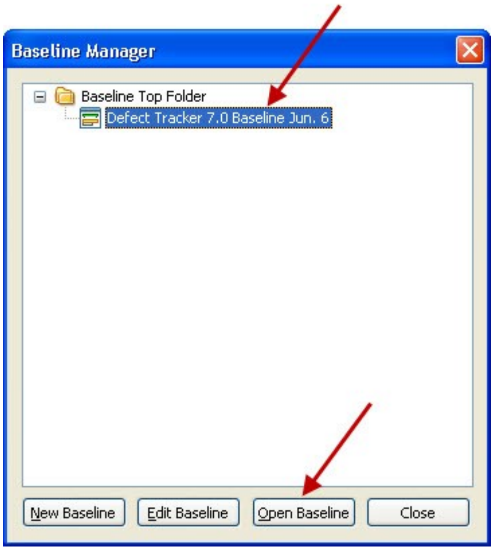
2.2 Opening a Baseline

After a baseline is created and saved, it can be opened at any point in future. When needing to compare current and previous designs, a previously saved baseline can easily be opened and compared with the current design.

1. In the baseline view, click the ellipsis button (…) in the tool bar. This will open the*Baseline Manager*dialog.


DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]
File Edit View Tool System Help
...
(All Members);(Open)
...

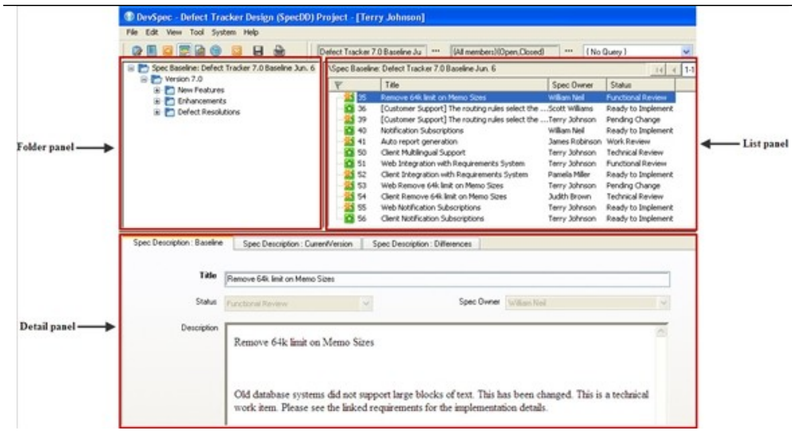
2. In the*Baseline Manager*dialog, highlight a previously saved baseline, and click the*Open Baseline*button.



2.3 Baseline View Interface

The baseline view in DevSpec is divided into three separate panels:
Folder panel
List panel
Detail panel

If no baseline is opened, these panels will be empty. At any instance, only one baseline can be opened in the baseline view.


Folder panel
List panel
Detail panel
DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]
File Edit View Tool System Help
Spec Baseline: Defect Tracker 7.0 Baseline Jun. 6
Version 7.0
New Features
Enhancements
Defect Resolutions
Spec Description : Baseline
Spec Description : CurrentVersion
Spec Description : Differences

Defect Tracker 7.0 Baseline Ju ··· {All members}{Open,Closed} ··· { No Query }

\Spec Baseline: Defect Tracker 7.0 Baseline Jun. 6

	Title	Spec Owner	Status
35	Remove 64k limit on Memo Sizes	William Neil	Functional Review
36	[Customer Support] The routing rules select the ...	Scott Williams	Ready to Implement
39	[Customer Support] The routing rules select the ...	Terry Johnson	Pending Change
40	Notification Subscriptions	William Neil	Ready to Implement
41	Auto report generation	James Robinson	Work Review
50	Client Multilingual Support	Terry Johnson	Technical Review
51	Web Integration with Requirements System	Terry Johnson	Functional Review
52	Client Integration with Requirements System	Pamela Miller	Ready to Implement
53	Web Remove 64k limit on Memo Sizes	Terry Johnson	Pending Change
54	Client Remove 64k limit on Memo Sizes	Judith Brown	Technical Review
55	Web Notification Subscriptions	Terry Johnson	Ready to Implement
56	Client Notification Subscriptions	Terry Johnson	Ready to Implement

Title: Remove 64k limit on Memo Sizes

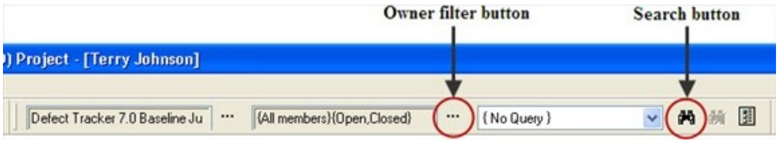
Status: Functional Review   Spec Owner: William Neil

Description: Remove 64k limit on Memo Sizes

Old database systems did not support large blocks of text. This has been changed. This is a technical work item. Please see the linked requirements for the implementation details.

The folder panel shows the selected folder tree for the opened baseline in a tree structure. Selecting a folder in the folder tree will work as a filter; that is, all specifications that belong to that folder will appear in the list panel (specifications in child folders are displayed too).

Using the controls in the tool bar can also filter specifications in the list panel. Click the ellipses button (…) for the owner control to filter by owner, or click the search button (binoculars icon) to search by other attributes. For more information on filtering and searching in DevSpec, please see chapter 4,*Searches and Queries*.



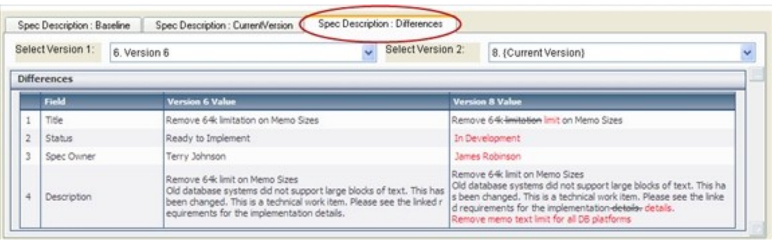
Defect Tracker 7.0 Baseline Ju ··· {All members}{Open,Closed} ··· { No Query }

The specification icons in the list panel denote whether the specification has been changed since the current baseline. An orange asterisk will appear over an icon in the case that there has been a change.

◇ **Specificationhas notbeen changed since the current baseline.**

✱ **Specificationhasbeen changed since the current baseline.**

When a specification in the list panel is highlighted, its comparison details will appear in the detail panel. The detail panel has three tabs:
The*Baseline*tab displays information for the highlighted specification at the time of baseline creation (saved sometime in past).
The*Current Version*tab displays the most updated and/or latest information for the highlighted specification.
The*Differences*tab displays the differences between the baseline information and the current information for the selected specification.


Spec Description : Baseline
Spec Description : CurrentVersion
Spec Description : Differences
Differences

Select Version 1: 6. Version 6   Select Version 2: 8. {Current Version}

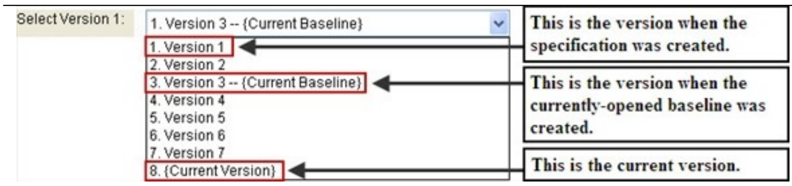
	Field	Version 6 Value	Version 8 Value
1	Title	Remove 64k limitation on Memo Sizes	Remove 64k ~~limitation~~ limit on Memo Sizes
2	Status	Ready to Implement	In Development
3	Spec Owner	Terry Johnson	James Robinson
4	Description	Remove 64k limit on Memo Sizes Old database systems did not support large blocks of text. This has been changed. This is a technical work item. Please see the linked requirements for the implementation details.	Remove 64k limit on Memo Sizes Old database systems did not support large blocks of text. This has been changed. This is a technical work item. Please see the linked requirements for the implementation ~~details.~~ details. Remove memo text limit for all DB platforms

2.4 Version Comparison

This information shown in the*Differences*tab allows the users to select two different versions of the selected specification, and compare the two.

Versions may be created manually in the*Versions*tab. For more information on managing versions in the*Versions*tab, please see chapter 3, section 1.2,*Specification Details*. Versions are also created automatically by a few events:

- **When an specification is created or edited (most common)**
- **When a new baseline is created**
- **When a specification is rolled back to a previous version**



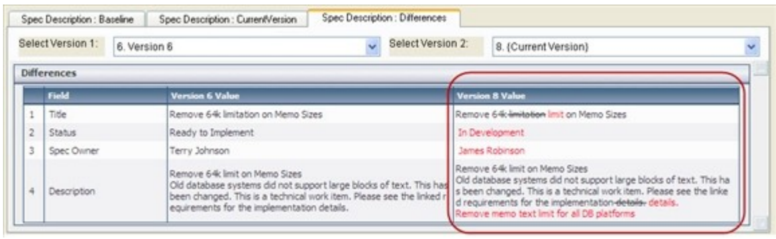
This is the version when the specification was created.

This is the version when the currently-opened baseline was created.

This is the current version.

The screenshot above is an expanded dropdown list in the*Differences*tab. After taking a look at this, we know that:
***Version 1*represents the first version of the specification (when it was submitted).**
The currently-opened baseline, denoted by*{Current Baseline}*, was created when the specification was on*Version 3*.
During the lifetime of this specification, there have been seven versions.
To compare two versions:

Select two different versions in the version dropdown lists. In the section below, the version on the right-hand side will display the difference of the specification properties, such as title, status, owner, and description. Added text will be in red (example), while removed text will be struck through (~~example~~). Even if only one character has been added or removed, the entire word will be denoted as a change.


Spec Description : Baseline
Spec Description : CurrentVersion
Spec Description : Differences
Select Version 1:
6. Version 6
Select Version 2:
8. {Current Version}
Differences

	Field	Version 6 Value	Version 8 Value
1	Title	Remove 64k limitation on Memo Sizes	Remove 64k ~~limitation~~ limit on Memo Sizes
2	Status	Ready to Implement	In Development
3	Spec Owner	Terry Johnson	James Robinson
4	Description	Remove 64k limit on Memo Sizes Old database systems did not support large blocks of text. This has been changed. This is a technical work item. Please see the linked requirements for the implementation details.	Remove 64k limit on Memo Sizes Old database systems did not support large blocks of text. This has been changed. This is a technical work item. Please see the linked requirements for the implementation ~~details.~~ details. Remove memo text limit for all DB platforms

In the above screenshot, in the*Differences*section, under the*Version 8 Value*row, the differences from*Version 6*to*Version 8*are shown. For example, in the title, “limitation” has been changed to “limit”. The status and owner have been changed as well.

2.5 Editing a Baseline

To edit the properties of an existing baseline, open the*Baseline Manager*, highlight the baseline to be edited, and click the*Edit Baseline*button. The*Baseline Edit*dialog will open for the selected baseline. Make the desired changes and save by following the similar steps for creating a baseline.

The*Specification Folder*field cannot be edited. Since a new version was created for all involving specifications upon baseline creation, prior versions cannot be added or deleted.

2.6 Deleting a Baseline

To delete a baseline, open the*Baseline Manager*, right-click on the baseline to be deleted, and select*Delete*. In the conformation dialog, click the*Yes*button.

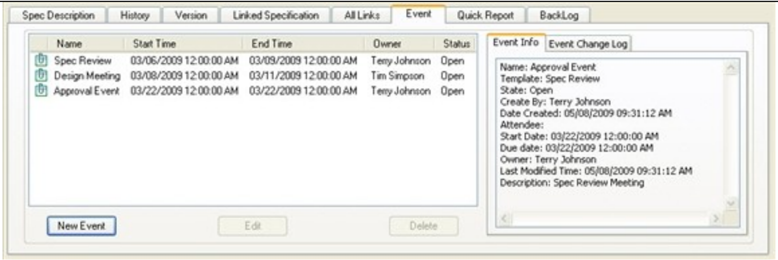
***Note*: A baseline cannot be deleted if it is currently open.**

3 Events

In DevSpec, an event is a management taskor activitythat facilitates communication and collaboration between project stakeholders so that a greatergoalmay be achieved.

3.1 Understanding DevSpec Events

Eventsusually representa list oftasks that need to be completedor confirmedasarequirement or specification isfinalized, approved,and/or committed. All events are easily managed in the*Event*tabin the detail panel.



Name	Start Time	End Time	Owner	Status
Spec Review	03/06/2009 12:00:00 AM	03/09/2009 12:00:00 AM	Terry Johnson	Open
Design Meeting	03/08/2009 12:00:00 AM	03/11/2009 12:00:00 AM	Tim Simpson	Open
Approval Event	03/22/2009 12:00:00 AM	03/22/2009 12:00:00 AM	Terry Johnson	Open

Name: Approval Event
Template: Spec Review
State: Open
Create By: Terry Johnson
Date Created: 05/08/2009 09:31:12 AM
Attendee:
Start Date: 03/22/2009 12:00:00 AM
Due date: 03/22/2009 12:00:00 AM
Owner: Terry Johnson
Last Modified Time: 05/08/2009 09:31:12 AM
Description: Spec Review Meeting

By utilizing events in DevSpec, users can createmanydifferenttasksassociated with a work item, assign each of those tasks to a different project member, define separate start and due dates for each subtask, and manage and track each of those tasksindependently in its own workflow.

Each event is based on an administrator-defined event template.An event template is a blueprint for creating a specific type of event, such as a meeting,apresentation, orademo.

Event templates exist in DevSpec in one of these twotypes: standard events and change flagging events.

> Standardevents: Standard events are used to track the activities that are going on during the specification management process. It can be a design review meeting, a presentation, or any other activity that needs to take place. This type of events may be created and tracked in both the specification and the change request viewsof the DevSpec client.
>
> Changeflagging event: Change flagging events are used to track the activities that are going on during the specification change process. This type of events may be created and tracked only in the change request view of the DevSpec client.

Each event is also defined by a unique workflow that is applicable to the template it derives from.The life cycle of an event is defined by two or more workflow states: an initial open state, a final closed state, and any number of intermediary states. An event state represents a specific stage of the life cycle of that event. Each event state is defined by its status: open, closed-successfully, or closed-failed.

The following is a list of event icons that identify the tasks by type and status in the event list:

Regular events are tagged with an event icon.

Changeflaggingevents with an*open*status are tagged with a red flag icon.

Changeflaggingevents with a*closed: failed*status are tagged with a yellow flag icon.

Changeflaggingevents with a*closed: successfully*status are tagged with a green flag icon.

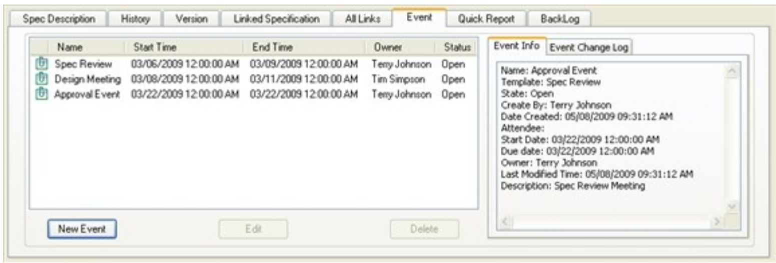
Linked DevTrack development issues are tagged with a DevTrack icon. Each changeflaggingevent subtask is the child of a linked development issue.

At each stage in its life cycle, an eventcan beowned by one (and only one) project member. The event owner is responsible for that specific event andshould make sure it iscompleted successfully.

*Note:*For information on creating event templates, please see the*DevSpec Admin Guide*

3.2 Tracking Standard Events

Project managers may create, edit, and review standardrequirement/specificationevents in the*Events*tab of the detail panel.



Name	Start Time	End Time	Owner	Status
Spec Review	03/06/2009 12:00:00 AM	03/09/2009 12:00:00 AM	Terry Johnson	Open
Design Meeting	03/08/2009 12:00:00 AM	03/11/2009 12:00:00 AM	Tim Simpson	Open
Approval Event	03/22/2009 12:00:00 AM	03/22/2009 12:00:00 AM	Terry Johnson	Open

Name: Approval Event
Template: Spec Review
State: Open
Create By: Terry Johnson
Date Created: 05/08/2009 09:31:12 AM
Attendee:
Start Date: 03/22/2009 12:00:00 AM
Due date: 03/22/2009 12:00:00 AM
Owner: Terry Johnson
Last Modified Time: 05/08/2009 09:31:12 AM
Description: Spec Review Meeting

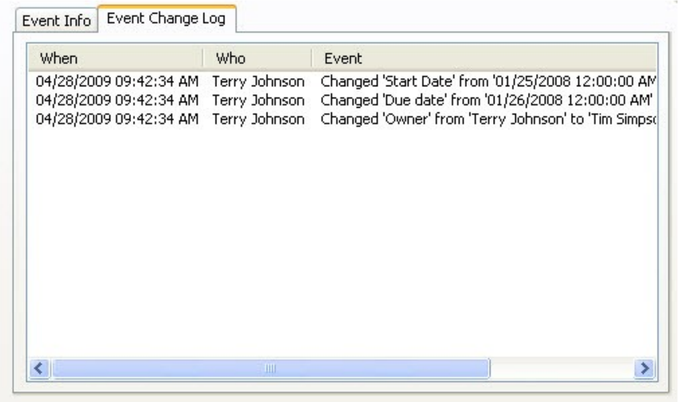
The event tab displays high-level and detailed information about the events associated with a particular requirement or specification.

The*Event*tab consists of three main areas:

Eventlist: The event list displays high-level information about events in a tabular format. Each column represents an event property, such as name, state time, end time, owner, and status.Each row represents an event and displays the event property valuesofthat event.

Eventinfo: The*Event Info*tab displays high-level information about an event, including its name, status, owner, start time, end time, and description.

Eventchangelog: The*Event Change Log*tab displays a list of all changes made to an event from creation closure. It logsthe change that was made, the person who made the change, and the time the change took place.



When	Who	Event
04/28/2009 09:42:34 AM	Terry Johnson	Changed 'Start Date' from '01/25/2008 12:00:00 AM
04/28/2009 09:42:34 AM	Terry Johnson	Changed 'Due date' from '01/26/2008 12:00:00 AM'
04/28/2009 09:42:34 AM	Terry Johnson	Changed 'Owner' from 'Terry Johnson' to 'Tim Simps

Eachgeneralevent consists of one or more of the followingproperties:

Name: The*Name*property identifies the title of the event. Events may inherit their name from the event template that was used to create them.

Start Time: The*Start Time*property identifies the date and time the event is scheduled to begin.

End Time: The*End Time*property identifies the date and time the event is scheduled to end.

Owner: At every stage in its life cycle, an event is owned by one project member. The event owner is responsible for the event and seeing that it successfully completed.

State: An event represents a specific stage of the life cycle of that event. Each event workflow state is defined by its status: open, closed successfully, or closed failed.

3.3 Managing Standard Events

Eventsrepresenta list oftasks that need to be completedor confirmedasawork item isfinalized, approvedand/or committed.

Examplesof standard eventsinclude:

- Brainstorming session
- Design review
- Management review
- Presentation
- Product demo

Using controls in the*New Event*dialog, project team members maycreate a new event associated with a work item,definetheevent schedule,andinvite project team members.

As mentioned, each standard event is based on an administrator-defined event template. An event template is a blueprint for creating a specific type of event, such as a meeting, presentation, or demo, that defines the business rules that determine how that event is managed in the project.

To create a standard event:

1. Select a work item in the list panel of the specification view

2. Go to the*Events*tab in the detail panel.

3. Select the*New*button in the Events tab. The*New Event*dialog appears.

4. Select an event template from the*Event Template*drop-down list.

5. Define a unique and descriptive name for the event in the*New*text box.

6. Provide the event details in the*Description*text box. Event details may include the event agenda, the event location, or other key information.

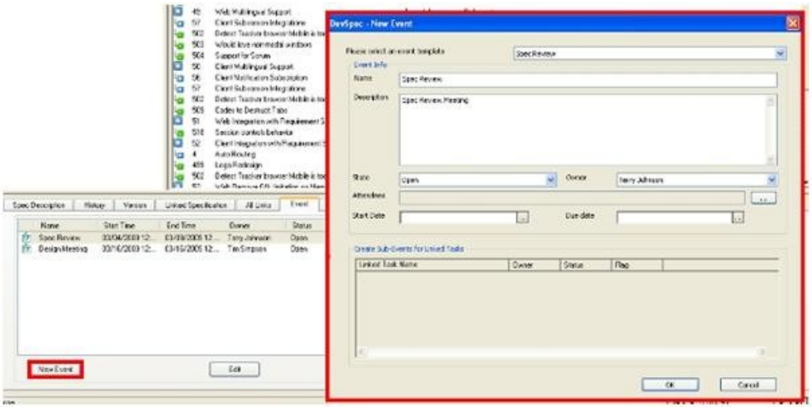
7. Select an option from the*State*drop-down list. Each standard event is defined by its event workflow state. An event state represents a specific stage of the life cycle of that event.

8. Select an option from the*Owner*drop-down list to populate the ownership of the event. The*Owner*drop-down list displays the names of every project member that has been designated as an applicable owner.

9. To define the date, time, and duration of the event, select the date and time of the event in the*Start Date*and*End Date*controls.

10. Optional: To define the event attendee list, click the ellipsis button (…) and add attendees to the event in the*Add Attendee*dialog box. The standard event attendee list may be used to define which project members are invited to the event by e-mail notification.

11. Click the OK button. A general event associated with a work item is successfully created.



DevSpec - New Event

Please select an event template: Spec Review

Name: Spec Review

Description: Spec Review Meeting

State: Open

Owner: Terry Johnson

Name	Start Time	End Time	Owner	Status
Spec Review	03/04/2003 12...	03/09/2005 12...	Terry Johnson	Open
Design Meeting	03/16/2003 12...	03/16/2005 12...	Tim Simpson	Open

To edit a standard event:

1. Select an event in the*Event*tab.

2. Click the*Edit*button in the*Event*tab. The*Edit Event*dialog appears.

3. Optional: To update the event details, edit the text in the*Description*text box. Event details may include the event agenda, the event location, or other key information.

4. Optional: To change the status of an event, select an event workflow state in the*State*drop-down list.

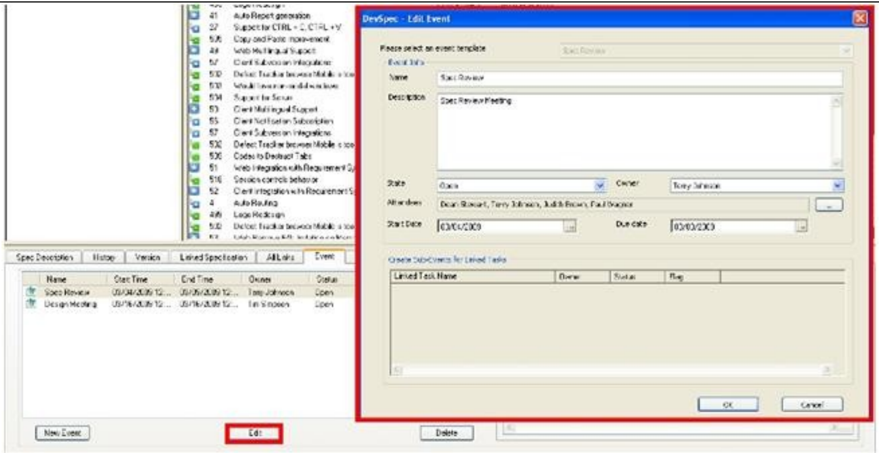
5. Optional: To assign the event to another project member, select a user name in the*Owner*drop-down list. The*Owner*drop-down list displays the names of each project member that has been designated as an applicable owner.

6. Optional: To update the date, time, and duration of the event, define the event start and end times in the*Start Date*and*End Date*controls.

7. Optional: To define the event attendee list, click the ellipsis button (...) and add attendees to the event in the*Add Attendee*dialog box. The standard event attendee list may be used to define which project members are invited to the event by e-mail notification.

For information on defining the attendee list, please see the section below,*Defining Event Attendee Lists*.

8. Click the OK button to submit the changes made to the event.



DevSpec - Edit Event

Please select an event template: Spec Review

Event Info

Name: Spec Review

Description: Spec Review Meeting

State: Open Owner: Terry Johnson

Attendees: Dean Stewart, Terry Johnson, Judith Brown, Paul Wagner

Start Date: 03/04/2009 Due date: 03/03/2009

Create Sub-Events for Linked Tasks

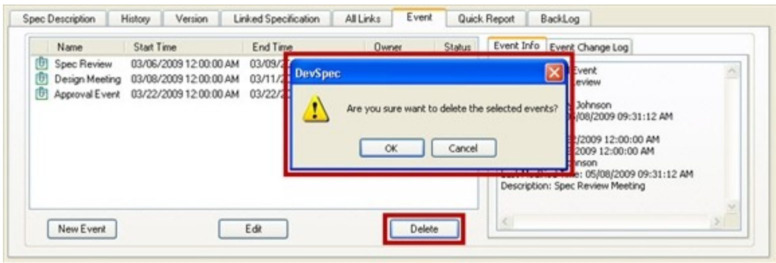
Linked Task Name	Owner	Status	Flag

To delete a standard event:

1. Select an event in the*Event*tab.

2. Click the*Delete*button in the*Event*tab. A warning dialog box appears.

3. Click the OK button to confirm the action.



Name	Start Time	End Time	Owner	Status
Spec Review	03/06/2009 12:00:00 AM	03/09/2		
Design Meeting	03/08/2009 12:00:00 AM	03/11/2		
Approval Event	03/22/2009 12:00:00 AM	03/22/2		

DevSpec

Are you sure want to delete the selected events?

Description: Spec Review Meeting

3.4 Defining Event Attendee Lists

An event attendee is a project member that is invited to attend a particular event. Once a project member has been designated as an event attendee, they may be formally invited to the event by e-mail notification.

Project members may add attendees to events whenever they create or edit a standard event in the*Event*tab of the detail panel.

The standard event attendee list may be used to define the e-mail address list for the meeting request.

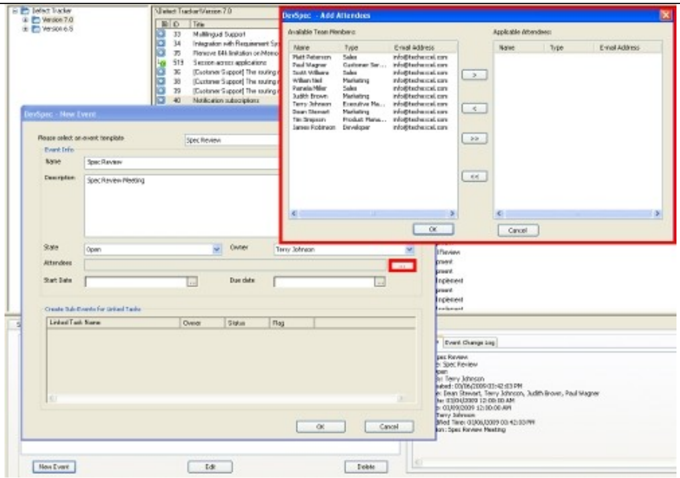
To define the event attendee list:

1. Submit a new event or select an existing event in the*Event*tab.

2. Click the*Attendee*button in the*Edit Event*dialog. The*Add Attendees*dialog box appears.

3. Add or remove project members to the attendee list. To add project members as attendees, select the names of the project members in the*Available Team members*list and click the right arrow button. To remove project members as attendees, select the names of the project members in the*Applicable Attendees*list and click the left arrow button.

4. Click the OK button.



DevSpec - New Event

Please select an event template: Spec Review

Event Info

Name: Spec Review

Description: Spec Review Meeting

State: Open

Owner: Terry Johnson

Attendees:

Start Date:

Due date:

Create Sub-Events for Linked Tasks

Linked Task Name	Owner	Status	Flag

DevSpec - Add Attendees

Available Team Members

Name	Type	E-mail Address
Matt Peterson	Sales	info@techexcel.com
Paul Magner	Customer Ser...	info@techexcel.com
Scott Williams	Sales	info@techexcel.com
William Neil	Marketing	info@techexcel.com
Pamela Miller	Sales	info@techexcel.com
Judith Brown	Marketing	info@techexcel.com
Terry Johnson	Executive Ma...	info@techexcel.com
Dean Stewart	Marketing	info@techexcel.com
Tim Simpson	Product Mana...	info@techexcel.com
James Robinson	Developer	info@techexcel.com

Applicable Attendees

Name	Type	E-mail Address

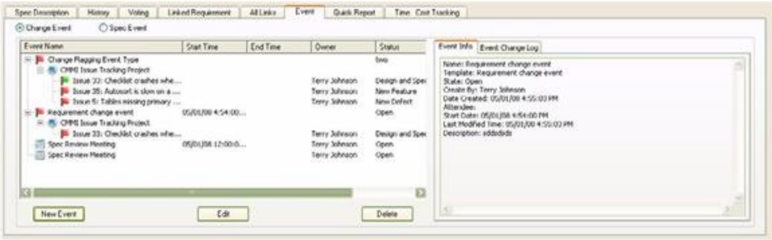
ID	Title
33	Multilingual Support
34	Integration with Requirement Sy...
35	Remove 64k limitation on Memo...
519	Session across applications
36	[Customer Support] The routing ...
38	[Customer Support] The routing ...
39	[Customer Support] The routing ...
40	Notification subscriptions

3.5 Tracking Change Flagging Events

Change flagging events may be created in support of requirement or specification change management processes. It is identified as a development task that links a requirement change or specification change to one or more DevTrack development issues.

Change flagging events are useful in notifying linked development task owners that the original requirement or specification is under change review, and thus the linked development tasks should temporarily be placed on hold until the requirement or specification is reviewed and finalized.

Change flagging events may be managed and tracked in the change request view of the DevSpec client.



○ Change Event ○ Spec Event

Event Name	Start Time	End Time	Owner	Status
Change Flagging Event Type				two
CMMI Issue Tracking Project				
Issue 33: Checklist crashes whe...			Terry Johnson	Design and Spec
Issue 35: Autosort is slow on a ...			Terry Johnson	New Feature
Issue 5: Tables missing primary ...			Terry Johnson	New Defect
Requirement change event	05/01/08 4:54:00...			Open
CMMI Issue Tracking Project				
Issue 33: Checklist crashes whe...			Terry Johnson	Design and Spec
Spec Review Meeting	05/01/08 12:00:0...		Terry Johnson	Open
Spec Review Meeting			Terry Johnson	Open

Name: Requirement change event
Template: Requirement change event
State: Open
Create By: Terry Johnson
Date Created: 05/01/08 4:55:03 PM
Attendee:
Start Date: 05/01/08 4:54:00 PM
Last Modified Time: 05/01/08 4:55:03 PM
Description: sddsdsds

The *Event*tab displays high-level and detailed information about the events associated with a particularchange request item.

Note that there are two radio buttons available for selection at the top of the*Event*tab:*Change Event*and*Spec Event*. Click the*Change Event*radio button to display all the events (both the regular events and the change flagging events) associated with the change request item; or click the*Spec Event*radio button to display the events (only the regular events) associated with the specification item.

Similar to the*Event*tab found in the specification view, the*Event*tabin the change request viewconsists of three main areas:

Eventlist: The Event list displays high-level information about events in a tabular format.Use the*Change Event*and the*Spec Event*radio buttons to switch the display of events.

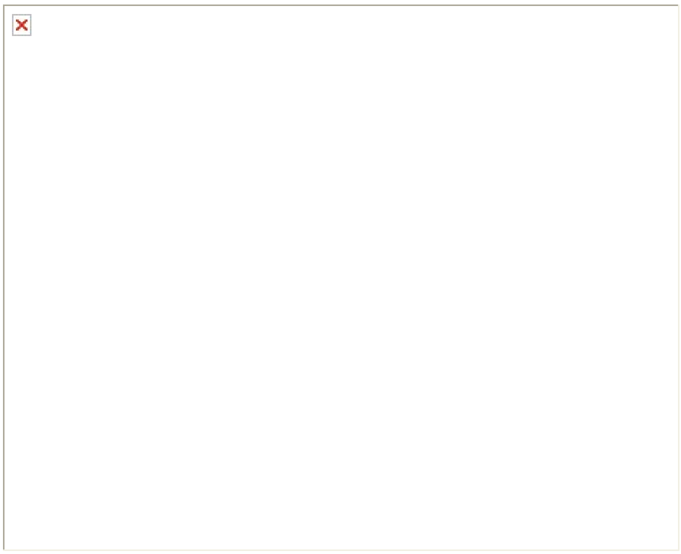
Eventinfo: The*Event Info*tab displays high-level information about an event, including its name, status, owner, start time, end time, and description.

Eventchangelog: The*Event Change Log*displays a list of all of the changes made to an event from its creation to its closure. It logsthe change that was made, the person who made the change, and the time the change took place.

3.6 Managing Change Flagging Events

In DevSpec, a change flagging event is a development task that links a requirement change or specification change to one or more DevTrack development issues. The links between the change flagging event and the development issues are managed and tracked as change event flagging subtasks.

Using controls in the*New Event*window, project team members maycreate a change flagging event, define theevent schedule, invite project team members, and identify change flagging event subtasks.



To create a change flagging event:

1. **Select a requirement or specification change in the list panel of the change request view.**

2. **Select the*Event*tabin thedetail panel.**

3. **Select the*ChangeEvent*radiobutton.**

4. **Select the*New Event*button. The*New Event*dialog appears.**

5. **Select a change flagging event template in the*Event Template*dropdown list. The*Event Template*dropdown list displays all event templates (standard event template or change event template) that may be used to create an event for the selected change request.**

   Note: **The scope of a change flagging event template may be limited to requirement changes or specification changes, based on administrator-defined rules.**

6. **Define a unique name for the event in the Name text box and a brief description in the Description text box.**

7. **Select the event workflow state in the*State*dropdown list. Each change flagging event is definedbyits event workflow state. An event state represents a specific stage of the life cycle of that event.**

8. **Select an option from the*Owner*dropdown list. The*Owner*dropdown list displays the names of all project members that have been designated as an applicable owner of thework item.**

9. **Optional:Toupdate the date, time, and duration of the event, define the event start and end times in the*Start Date*and*End Date*controls.**

10. **Optional: To define the event attendee list, click the ellipsis button (…) and add attendees to the event in the*Add Attendee*dialog box. The change flagging event attendee list may be used to define which project members are invited to the event by e-mail notification. For step-by-step instructions, please see the prior section,*Defining Event Attendee Lists*.**

11. **Select an integrated DevTrack development project and one or more development issues in the*Create Subverts for Linked Tasks*area.**

12. **Click the OK button.**

Note:**Some of the controls mentioned above might not be accessible to users if project administrators set them to be invisible.**

To edit a change flagging event:

1. **Select an event in the*Event*tab of the change request detail panel.**

2. Clickthe*Edit Event*button in the*Event*tab. The*Edit Event*dialog appears.

3. Optional: To update the event details, edit the text in the*Description*text box. Event details may include the event agenda, the event location, or other key information.

4. Optional: To change the status of an event, select an event workflow state inthe*State*dropdown list. An event state represents a specific stage of the life cycle of that event.

5. Optional: To assign the event to another project member, select a user name in the*Owner*dropdown list.

6. Optional:Toupdate the date, time, and duration of the event, define the event start and end times in the*Start Date*and*End Date*controls.

7. Optional: To define the event attendee list, click the ellipsis(…)button and add attendees to the event in the*Add Attendee*dialog box. The attendee list may be used to define which project members are invited to the event by e-mail notification. For step-by-step instructions, please see the prior section,***Defining Event Attendee Lists***.

8. Click the OK button.

To delete a change flagging event:

1. Select a change flagging event in the*Event*tab of the change request detail panel.

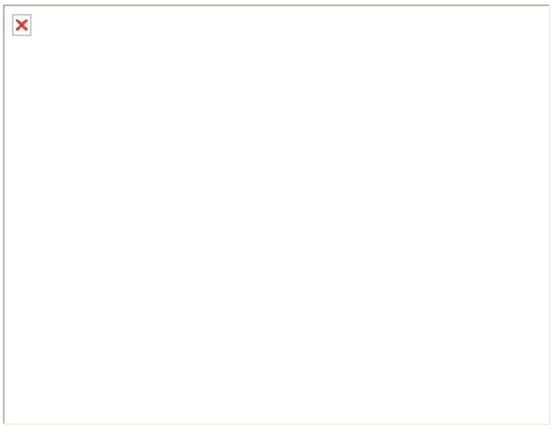
2. Click the*Delete*button in the Event tab. A warning dialog box appears.

3. Click the OK button.

3.7 Managing Change Flagging Event Subtasks

A change flagging event subtask isadevelopment event that links the change made to a requirement or specification to a specific DevTrack development issue. Change flagging event subtask may be independently managed and tracked in the DevSpec and DevTrack clients.

All change flagging event subtasks are managed in a workflow consisting of three states: the*Open*state, the*Close Success*state, and the*CloseFail*state. Using controls in the*Edit Linked Task Flagging*dialog, project team members may change the workflow state of change flagging event subtasks and add flagging notes.



To edit a change flagging event subtask:

1. Select a change flagging event subtask in theevent list of the change request detail panel. Every change flagging event is defined by one or more event subtasks.

2. Click the*Edit*button in the*Event*tab. The*Edit Linked Task Flagging*dialog appears.

3. To change the workflow state of the subtask, select an option in the*Status*dropdown list. The lifecycle of change flagging event subtasks is defined by three workflow states:

   Open

   When the*Open*state is selected, the□ icon (a red flag) will be displayed in front of the subtask.

Close Success

When the*Close Success*state is selected, the□icon (a green flag) will be displayed in front of the subtask.

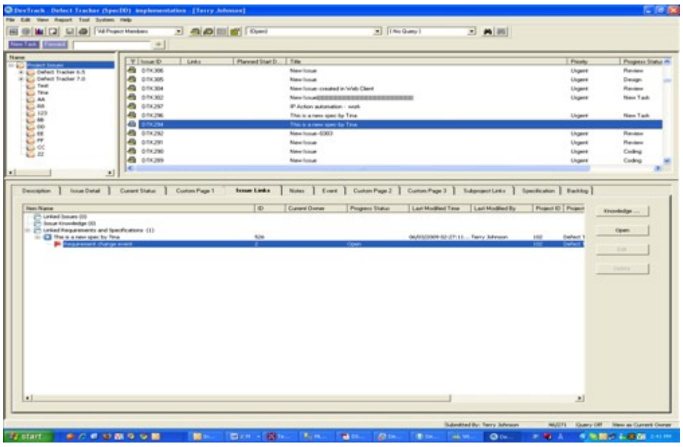
Close Fail

When the*Close Fail*state is selected, the□ icon (a yellow flag) will be displayed in front of the subtask.

4. Optional: Toadd a note to this subtask,entersometext in the*Flagging Note*text box.

5. Click the*OK*button.

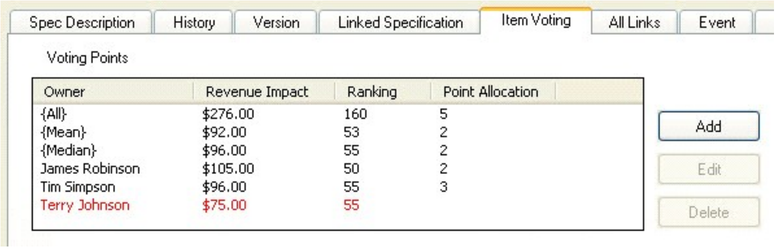
3.8 Tracking Change Flagging Events in DevTrack



4 Voting

In DevSpec, the approval and scheduling ofwork itemsmay be managed using the votingfeature.Voting is an optional feature in DevSpec thatallows distributed teams to evaluate the impact and need for awork item. Project managers can then collect and track the numerical data submitted by team members to further analyze the pending work item.

Using customized voting grids, project teams may track “votes” from individual project members, as well as the median, mean, and total value of allavailable voting types.



Owner	Revenue Impact	Ranking	Point Allocation
{All}	$276.00	160	5
{Mean}	$92.00	53	2
{Median}	$96.00	55	2
James Robinson	$105.00	50	2
Tim Simpson	$96.00	55	3
Terry Johnson	$75.00	55	

A voting type is a custom data field that may be used to track numerical data in a voting grid control. Each column in a voting grid represents an administrator-defined voting type. A voting type is defined by three properties: its title, its field type (integer, decimal, or currency), and its scope–a range of acceptable values.

Examples of voting types include:

Revenue Impact -Users can contribute to the evaluation of the revenue impact for a proposedwork item. The product design team, stakeholders, and other DevSpec users can enter their estimates of the revenue impact towards the implementation of a suggested feature.

Ranking -Each DevSpec user can rank a requirement based on urgency, usefulness, and triviality of aproposed work item.

Point Allocation -Apart from ranking and revenue impact, users can also allocate points to eachwork item.The more points allocated by a member, the more important and urgent he or she thinks this work item is.

Note:Votingis anoptional DevSpec feature that must be enabled in each DevSpec project and requiresthat the project administratorconfigure the voting types, as well ascreate a custom page to manage and track the "votes" submitted bytheteam members.

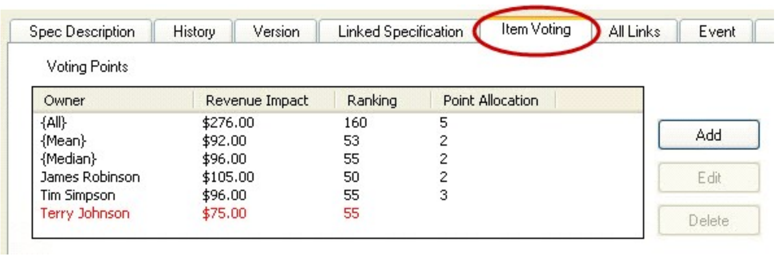
For information on administering voting types, see theDevSuite Admin Guide.

4.1 Voting

To vote on a specification or requirement:

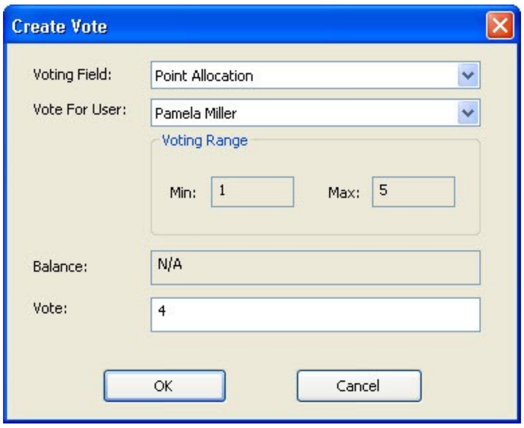
Highlight awork itemin the listpanel of the specification view.

In theDetailpanel, go to theItem Votingtab.



Owner	Revenue Impact	Ranking	Point Allocation
{All}	$276.00	160	5
{Mean}	$92.00	53	2
{Median}	$96.00	55	2
James Robinson	$105.00	50	2
Tim Simpson	$96.00	55	3
Terry Johnson	$75.00	55	

Click theAddbutton next to theVoting Pointsfield. TheCreate Votedialog appears.



Voting Field: Point Allocation

Vote For User: Pamela Miller

Voting Range

Min: 1 Max: 5

Balance: N/A

Vote: 4

Select an availablevoting typefrom theVoting Fielddropdown list:

Revenue Impact

Ranking

Point Allocation
Once a voting field is selected, the administrator-defined voting range, showing the minimum as well as the maximum value of the vote, will be displayed in theVotingRangesection. In the example provided in the above screenshot, the voting range for thePoint Allocationis between 1 and 5, which means that voters can input any number that is within this range.

Select a user name from theVote For Userdropdown list. Normally users will only see their own names in the list unless the users have privileges to vote for others.

Enter a value in theVotetext field. Note that the value entered has to be within the voting range.

If a voting type has the balance feature turned on, the current balance for the selected voting type and the selected user will be displayed in the*Balance*field. Note that the value entered in the*Vote*field will be subtracted fro

Chapter 9 - Reporting

The DevSpec reporting tool comes with a variety of reports, used to dynamically analyze project data, gauge the progress of an development iteration, or export data that can be viewed by non-project members.

1 Managing reports in the report view

DevSpec is a powerful tool in tracking requirements, specifications and change requests. Built-in is a robust and customizable reporting tool to further assist project members in analyzing DevSpec data. Reports can also be used to analyze how items in DevSpec relate to their linked development and QA tasks in other DevSuite components.

With this reporting too, project members may:

Select from twelve different styles of reports.

Define report parameters and data ranges.

Dynamically modify reports using quick filters.

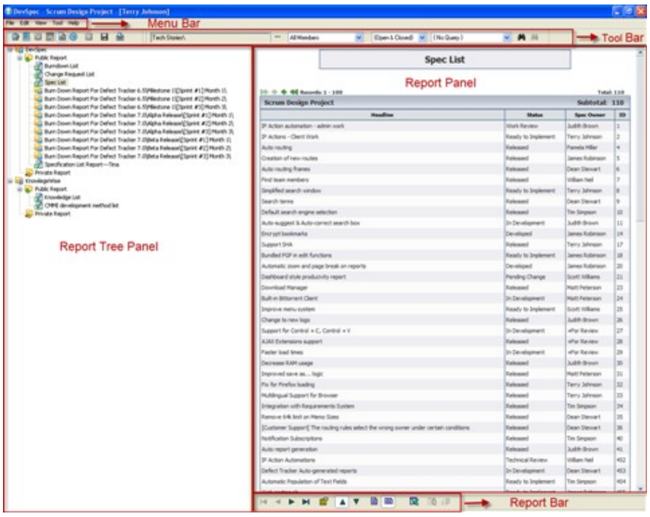
Print and export reports.

By utilizing the reporting feature, DevSpec users can generate the information they are looking for accurately and efficiently.

1.1 Understanding the report view interface

A report is an organized presentation of data displayed in either tabular or graphical formats. Reports are managed in the DevSpec report view.

To access the report view, click the report view icon in the tool bar. The report view workspace is organized into two panels (the tree panel and the report panel) and three bars (the menu bar, the tool bar, and the report bar).


DevSpec - Scrum Design Project - [Terry Johnson]
File Edit View Tool Help
Menu Bar
Tech Stories\
All Members
(Open & Closed)
(No Query)
Tool Bar
DevSpec
Public Report
Burndown List
Change Request List
Spec List
Burn Down Report For Defect Tracker 6.5\Milestone 1([Sprint #1] Month 1)
Burn Down Report For Defect Tracker 6.5\Milestone 1([Sprint #2] Month 2)
Burn Down Report For Defect Tracker 6.5\Milestone 1([Sprint #3] Month 3)
Burn Down Report For Defect Tracker 7.0\Alpha Release([Sprint #1] Month 1)
Burn Down Report For Defect Tracker 7.0\Alpha Release([Sprint #2] Month 2)
Burn Down Report For Defect Tracker 7.0\Alpha Release([Sprint #3] Month 3)
Burn Down Report For Defect Tracker 7.0\Beta Release([Sprint #1] Month 1)
Burn Down Report For Defect Tracker 7.0\Beta Release([Sprint #2] Month 2)
Burn Down Report For Defect Tracker 7.0\Beta Release([Sprint #3] Month 3)
Specification List Report---Tina
Private Report
KnowledgeWise
Public Report
Knowledge List
CMMI development method list
Private Report
Report Tree Panel
Report Panel
Report Bar

Spec List

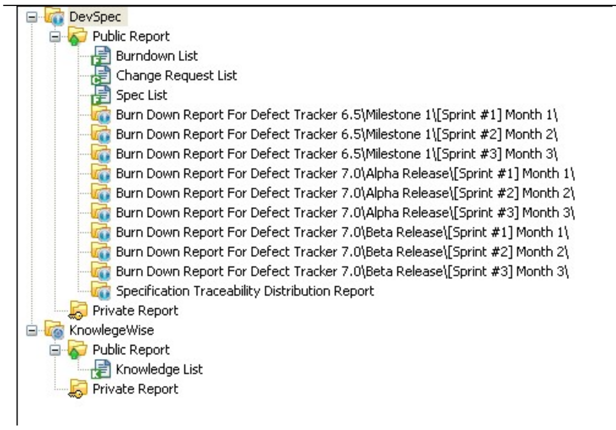
Records 1 - 100 Total: 110

Scrum Design Project Subtotal: 110

Headline	Status	Spec Owner	ID
IP Action automation - admin work	Work Review	Judith Brown	1
IP Actions - Client Work	Ready to Implement	Terry Johnson	2
Auto routing	Released	Pamela Miller	4
Creation of new routes	Released	James Robinson	5
Auto routing frames	Released	Dean Stewart	6
Find team members	Released	William Neil	7
Simplified search window	Ready to Implement	Terry Johnson	8
Search terms	Released	Dean Stewart	9
Default search engine selection	Released	Tim Simpson	10
Auto-suggest & Auto-correct search box	In Development	Judith Brown	11
Encrypt bookmarks	Developed	James Robinson	14
Support SHA	Released	Terry Johnson	17
Bundled PGP in edit functions	Ready to Implement	James Robinson	18
Automatic zoom and page break on reports	Developed	James Robinson	20
Dashboard style productivity report	Pending Change	Scott Williams	21
Download Manager	Released	Matt Peterson	23
Built-in Bittorrent Client	In Development	Matt Peterson	24
Improve menu system	Ready to Implement	Scott Williams	25
Change to new logo	Released	Judith Brown	26
Support for Control + C, Control + V	In Development	<For Review	27
AJAX Extensions support	Released	<For Review	28
Faster load times	In Development	<For Review	29
Decrease RAM usage	Released	Judith Brown	30
Improved save as.... logic	Released	Matt Peterson	31
Fix for Firefox loading	Released	Terry Johnson	32
Multilingual Support for Browser	Released	Terry Johnson	33
Integration with Requirements System	Released	Tim Simpson	34
Remove 64k limit on Memo Sizes	Released	Dean Stewart	35
[Customer Support] The routing rules select the wrong owner under certain conditions	Released	Dean Stewart	36
Notification Subscriptions	Released	Tim Simpson	40
Auto report generation	Released	Judith Brown	41
IP Action Automations	Technical Review	William Neil	452
Defect Tracker Auto-generated reports	In Development	Dean Stewart	453
Automatic Population of Text Fields	Ready to Implement	Tim Simpson	454

Tree Panel

The tree panelin the report vieworganizes reports into two system-defined project folders:*DevSpec*and*KnowledgeWise*. They are placeholders for specification and knowledge related reports, respectively. Reports created under either of the project folders can be categorized as public or private. Reports created under the*Public Reports folder*can be accessed by all users in DevSpec, while those created under the*Private Reports*folder can only be accessed by the report author.



- DevSpec
  - Public Report
    - Burndown List
    - Change Request List
    - Spec List
    - Burn Down Report For Defect Tracker 6.5\Milestone 1\[Sprint #1] Month 1\
    - Burn Down Report For Defect Tracker 6.5\Milestone 1\[Sprint #2] Month 2\
    - Burn Down Report For Defect Tracker 6.5\Milestone 1\[Sprint #3] Month 3\
    - Burn Down Report For Defect Tracker 7.0\Alpha Release\[Sprint #1] Month 1\
    - Burn Down Report For Defect Tracker 7.0\Alpha Release\[Sprint #2] Month 2\
    - Burn Down Report For Defect Tracker 7.0\Alpha Release\[Sprint #3] Month 3\
    - Burn Down Report For Defect Tracker 7.0\Beta Release\[Sprint #1] Month 1\
    - Burn Down Report For Defect Tracker 7.0\Beta Release\[Sprint #2] Month 2\
    - Burn Down Report For Defect Tracker 7.0\Beta Release\[Sprint #3] Month 3\
    - Specification Traceability Distribution Report
  - Private Report
- KnowlegeWise
  - Public Report
    - Knowledge List
  - Private Report

Report Panel

The report panel displays the report data.

Menu Bar

The menu bar organizes DevSpec commands into five different menus: the*File*menu,*Edit*menu,*View*menu,*Tool*menu, and*Help*menu.For details about the menu bar, please see Chapter 2,*DevSpec Client basics*.

Tool Bar

The tool bar displays four filters that enable report viewers to define the scope of a report based on work item property values. Toolbar filters include:

Tree folder filter: define the scope of the report by selecting one or more folders.

Owner filter: filter reports by current owner

State and status filter: filter reports by workflow state or status

Query filter: filter reports by detailed search queries

For more information on filters and queries, please see chapter 4,*Searches and Queries.*

Report Bar

The report bar displays tools and controls that enable the report creators to customize the display of the report data. The controls displayed in the report bar are report style-specific.

Common functionalities include:

First: Go to the first page of the report.

Previous: Go to the previous page of the report.

Next: Go to the next page of the report.

Last: Go to the last page of the report.

Properties: Define report-specific properties including report title, fields to display and item types.

Ascending Sort: Display the report list in ascending order.

Descending Sort: Display the report list in descending order.

Export Report: Export the report in display to the Excel format.

For more information on using the report bar to customize a specific kind of report, please see sections 2 and 3, later in this chapter.

1.2 Report Basics

This section shows how to create, edit and delete a report or report folder.

Adding a report folder

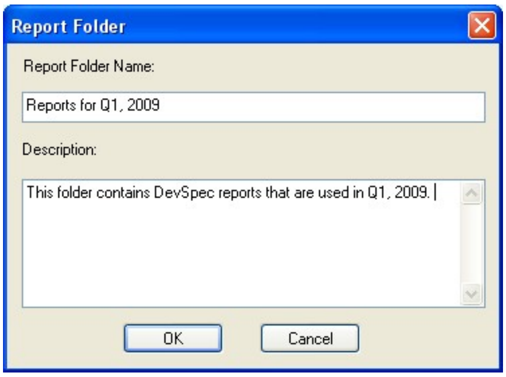
Using commands in the tree panel shortcut menu, users may add subfolders within the system-defined root folders to further organize the reports.

To create a report folder:

Choose whether the report folder is public or private:

To create a public report folder, right-click on the root*Public Report*folder and select*New Folder*.

To create a private report folder, right-click on the root*Private Report*folder and select*New Folder*.
The*Report Folder*dialog box appears. Define the report name and description.



Report Folder Name:
Reports for Q1, 2009

Description:
This folder contains DevSpec reports that are used in Q1, 2009.

3. Click the*OK*button. A new report folder is created.

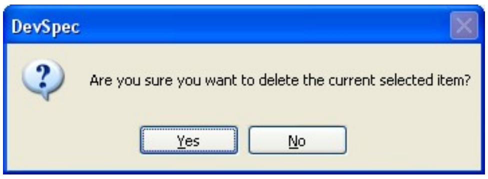
Editing a report folder

To edit an existing report folder, right-click on a report folder in the report tree panel, and select*Properties*in the shortcut menu. In the newly opened dialog, update the report folder properties, and click the*OK*button to save the changes.

Deleting a report folder

Right-click a report folder in the tree panel of the report view.

Select the*Delete*command in the shortcut menu. A confirmation dialog box appears.



Are you sure you want to delete the current selected item?

Click the*Yes*button to confirm.

Note: Report folders that contain reports cannot be deleted. All reports need to be moved to the root folder or other subfolders before the report folder can be deleted. To move the report(s), simply drag and drop the report(s) in the tree panel.

Adding a report

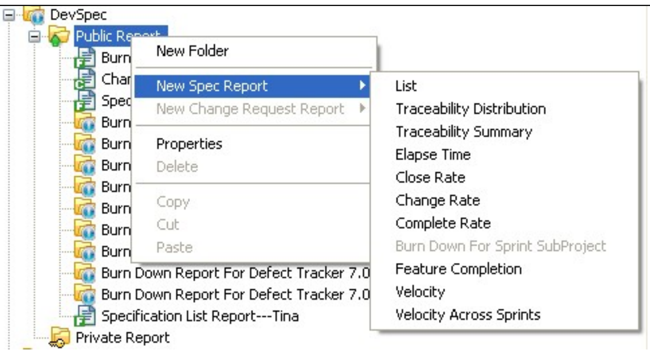
Using commands in the tree panel shortcut menu, a report author may add requirement, specification, and change request reports.

To create a report:

Right-click on a report folder in the report tree panel. The folder selected determines the report type:

To create a public report, select a subfolder within the root*Public Report*folder.

To create a private report, select a subfolder within the root*Private Report*folder.



2. Select a report style in the shortcut menu

   The shortcut menu displays commands that enable the report author to create and manage report styles that are appropriate to the selected folder. Once the report style is defined, the report properties dialog appears.

   ***Note*:**This dialog is report style-specific. The tools and controls displayed in the report manager depend on the style of report open in the report panel.

   Define report properties, such as report title, item type, report query or filter, and other report style-specific attributes.

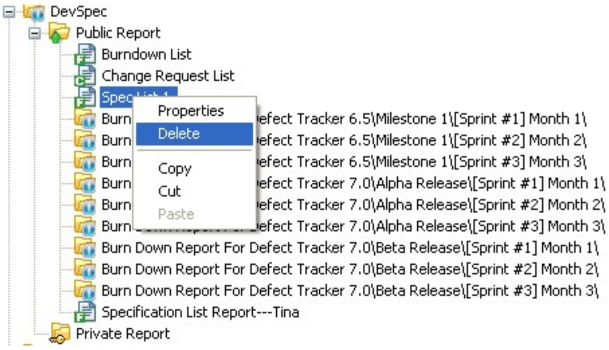
4. Click the*OK*button. The report is displayed in the report panel.

Editing a report

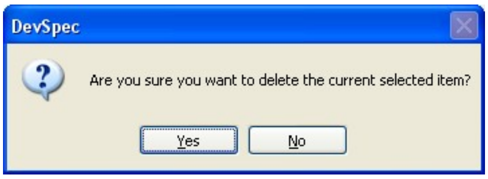
To edit an existing report, select a report in the report tree panel and click the properties button in the report bar. Update report properties in the properties dialog.

Deleting a report:

Right-click on a report in the tree panel of the report view.


DevSpec
Public Report
Burndown List
Change Request List
Spec
Properties
Delete
Copy
Cut
Paste
Burn ...efect Tracker 6.5\Milestone 1\[Sprint #1] Month 1\
Burn ...efect Tracker 6.5\Milestone 1\[Sprint #2] Month 2\
Burn ...efect Tracker 6.5\Milestone 1\[Sprint #3] Month 3\
Burn ...efect Tracker 7.0\Alpha Release\[Sprint #1] Month 1\
Burn ...efect Tracker 7.0\Alpha Release\[Sprint #2] Month 2\
Burn ...efect Tracker 7.0\Alpha Release\[Sprint #3] Month 3\
Burn Down Report For Defect Tracker 7.0\Beta Release\[Sprint #1] Month 1\
Burn Down Report For Defect Tracker 7.0\Beta Release\[Sprint #2] Month 2\
Burn Down Report For Defect Tracker 7.0\Beta Release\[Sprint #3] Month 3\
Specification List Report---Tina
Private Report

Select the*Delete*command in the shortcut menu. A confirmation dialog box appears.



Are you sure you want to delete the current selected item?

3. Click the*Yes*button to confirm the deletion.

***Note*:**Deleting a report is an unrecoverable procedure!

2 DevSpec Reports

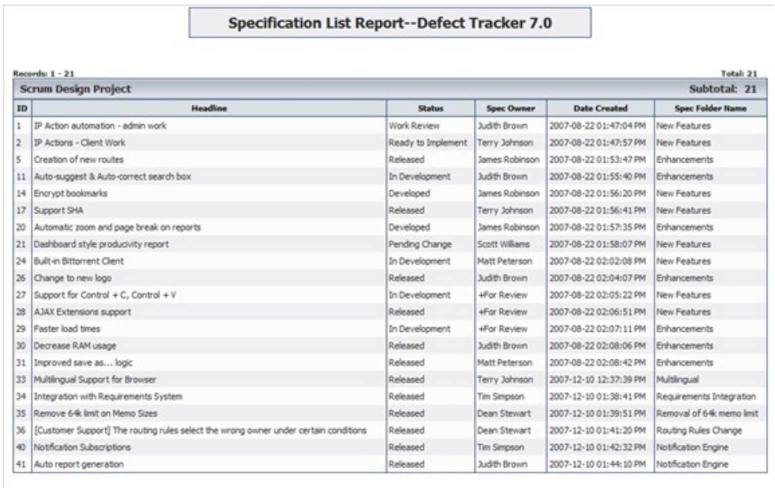
DevSpec currently supports the following report styles:
Specification list report

Traceability distribution report
Traceability summary report
Elapse time report
Close rate report
Change rate report
Complete rate report
Burn down report
Feature completion report
Velocity report
Velocity across report
Change request list report
Each report is explained in details in the sections below.

2.1 Specification List Reports

The specification list report displays information tracked under the specification view in DevSpec. Any field or attribute related to a specification or requirement can be included in the report.

Below is an example of a specification list report:



Specification List Report--Defect Tracker 7.0

Records: 1 - 21 Total: 21

Scrum Design Project Subtotal: 21

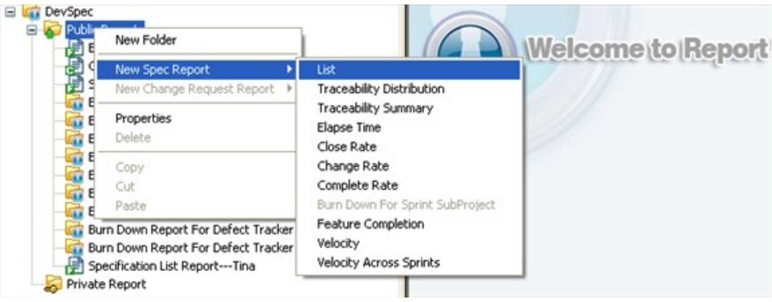
ID	Headline	Status	Spec Owner	Date Created	Spec Folder Name
1	IP Action automation - admin work	Work Review	Judith Brown	2007-08-22 01:47:04 PM	New Features
2	IP Actions - Client Work	Ready to Implement	Terry Johnson	2007-08-22 01:47:57 PM	New Features
5	Creation of new routes	Released	James Robinson	2007-08-22 01:53:47 PM	Enhancements
11	Auto-suggest & Auto-correct search box	In Development	Judith Brown	2007-08-22 01:55:40 PM	Enhancements
14	Encrypt bookmarks	Developed	James Robinson	2007-08-22 01:56:20 PM	New Features
17	Support SHA	Released	Terry Johnson	2007-08-22 01:56:41 PM	New Features
20	Automatic zoom and page break on reports	Developed	James Robinson	2007-08-22 01:57:35 PM	Enhancements
21	Dashboard style producivity report	Pending Change	Scott Wiliams	2007-08-22 01:58:07 PM	New Features
24	Built-in Bittorrent Client	In Development	Matt Peterson	2007-08-22 02:02:08 PM	New Features
26	Change to new logo	Released	Judith Brown	2007-08-22 02:04:07 PM	Enhancements
27	Support for Control + C, Control + V	In Development	+For Review	2007-08-22 02:05:22 PM	New Features
28	AJAX Extensions support	Released	+For Review	2007-08-22 02:06:51 PM	New Features
29	Faster load times	In Development	+For Review	2007-08-22 02:07:11 PM	Enhancements
30	Decrease RAM usage	Released	Judith Brown	2007-08-22 02:08:06 PM	Enhancements
31	Improved save as... logic	Released	Matt Peterson	2007-08-22 02:08:42 PM	Enhancements
33	Multilingual Support for Browser	Released	Terry Johnson	2007-12-10 12:37:39 PM	Multilingual
34	Integration with Requirements System	Released	Tim Simpson	2007-12-10 01:38:41 PM	Requirements Integration
35	Remove 64k limit on Memo Sizes	Released	Dean Stewart	2007-12-10 01:39:51 PM	Removal of 64k memo limit
36	[Customer Support] The routing rules select the wrong owner under certain conditions	Released	Dean Stewart	2007-12-10 01:41:20 PM	Routing Rules Change
40	Notification Subscriptions	Released	Tim Simpson	2007-12-10 01:42:32 PM	Notification Engine
41	Auto report generation	Released	Judith Brown	2007-12-10 01:44:10 PM	Notification Engine

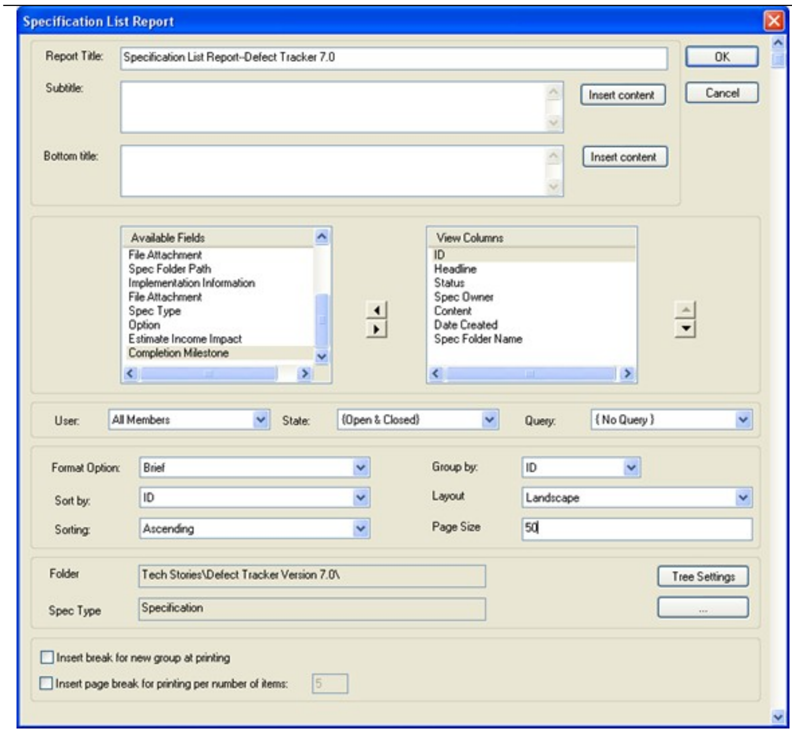
Creating a specification list report

To create a specification list report:

1. Go to the*Report*view by clicking the 📊 button on the tool bar.

2. In the*Tree*panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder (or any subfolder underneath them). Select*New Spec Report>List*. The*Specification List Report*dialog appears.





Specification List Report

Report Title: Specification List Report–Defect Tracker 7.0

Subtitle:

Bottom title:

Available Fields	View Columns
File Attachment	ID
Spec Folder Path	Headline
Implementation Information	Status
File Attachment	Spec Owner
Spec Type	Content
Option	Date Created
Estimate Income Impact	Spec Folder Name
Completion Milestone	

User: All Members State: {Open & Closed} Query: { No Query }

Format Option: Brief

Sort by: ID

Sorting: Ascending

Group by: ID

Layout: Landscape

Page Size: 50

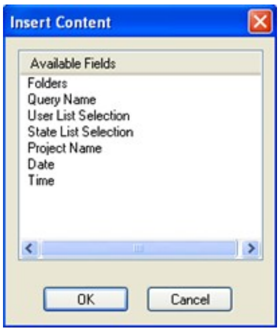
Folder: Tech Stories\Defect Tracker Version 7.0\

Spec Type: Specification

□ Insert break for new group at printing

□ Insert page break for printing per number of items: 5

3. In theSpecification List Reportdialog, define the report title, subtitle, and bottom title. Users can also insert pre-defined system fields by clicking the*Insert content*button. Highlight the fields that are to be added to the subtitle or bottom title, and click the*OK*button.



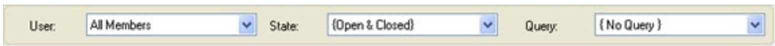
Insert Content

Available Fields

- Folders
- Query Name
- User List Selection
- State List Selection
- Project Name
- Date
- Time

4. Define the report columns. Users can click the ◀▶ buttons to add or remove fields between the*Available Fields*and*View Columns*lists. Users can also define the column order by using the ▲▼ buttons.

5. The*User, State*,and*Query*dropdown lists allow the user to filter the report data. These three filters can also be found in the tool bar in the report view.

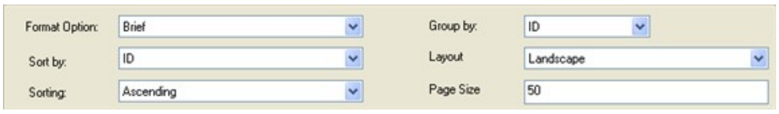


User: All Members State: {Open & Closed} Query: { No Query }

6. Users can also define the report as brief or detailed by using the*Format Option*dropdown list. More columns can be included in a detailed report than a brief one.

7. Define how the specification list is displayed. Users can customize the report by configuring the*Group By*,*Sort By*,*Layout*, and*Sorting*options.

8. The*Page Size*value allows users to determine the number of records displayed per page.



Format Option: Brief

Sort by: ID

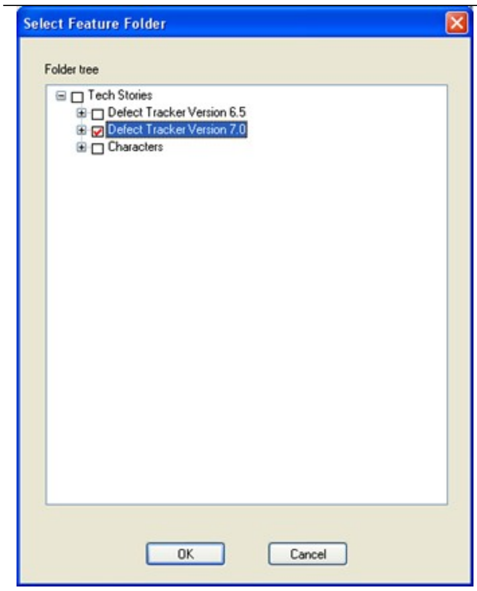
Sorting: Ascending

Group by: ID

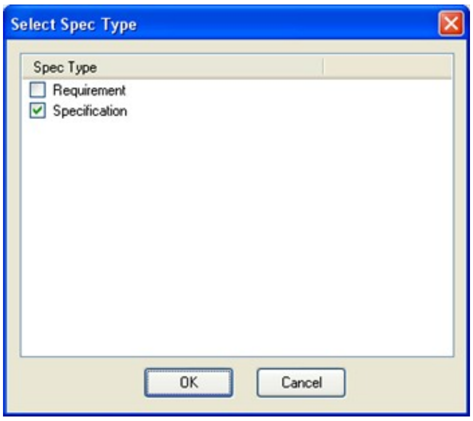
Layout: Landscape

Page Size: 50

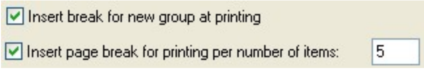
9. Define the report scope. Click the*Tree Settings*button to select a specific branch of the specification folder tree. When a parent folder is selected, all the descendent child folders will be selected automatically.



10. Define which item types, specifications and/or requirements, are covered in the report. Click the ellipses (…) button to select the item types. Choosing at least one item type in the report is mandatory. If no item type is selected, no data will be returned.



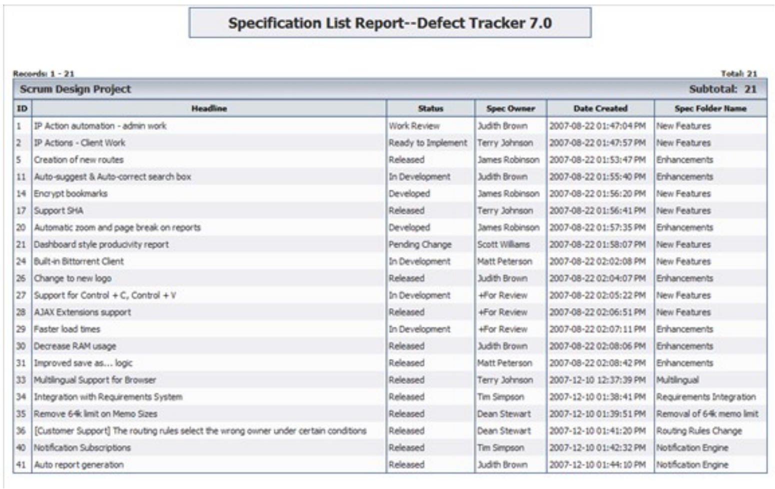
11. Users can choose to insert a page break after every grouping of records when printing the report. Page breaks can also be inserted based on the number of records. This can be configured by selecting the checkboxes, as shown below:



☑ Insert break for new group at printing

☑ Insert page break for printing per number of items: 5

12. Click theOKbutton.



Specification List Report--Defect Tracker 7.0

Records: 1 - 21 Total: 21

Scrum Design Project Subtotal: 21

ID	Headline	Status	Spec Owner	Date Created	Spec Folder Name
1	IP Action automation - admin work	Work Review	Judith Brown	2007-08-22 01:47:04 PM	New Features
2	IP Actions - Client Work	Ready to Implement	Terry Johnson	2007-08-22 01:47:57 PM	New Features
5	Creation of new routes	Released	James Robinson	2007-08-22 01:53:47 PM	Enhancements
11	Auto-suggest & Auto-correct search box	In Development	Judith Brown	2007-08-22 01:55:40 PM	Enhancements
14	Encrypt bookmarks	Developed	James Robinson	2007-08-22 01:56:20 PM	New Features
17	Support SHA	Released	Terry Johnson	2007-08-22 01:56:41 PM	New Features
20	Automatic zoom and page break on reports	Developed	James Robinson	2007-08-22 01:57:35 PM	Enhancements
21	Dashboard style producivity report	Pending Change	Scott Williams	2007-08-22 01:58:07 PM	New Features
24	Built-in Bittorrent Client	In Development	Matt Peterson	2007-08-22 02:02:08 PM	New Features
26	Change to new logo	Released	Judith Brown	2007-08-22 02:04:07 PM	Enhancements
27	Support for Control + C, Control + V	In Development	+For Review	2007-08-22 02:05:22 PM	New Features
28	AJAX Extensions support	Released	+For Review	2007-08-22 02:06:51 PM	New Features
29	Faster load times	In Development	+For Review	2007-08-22 02:07:11 PM	Enhancements
30	Decrease RAM usage	Released	Judith Brown	2007-08-22 02:08:06 PM	Enhancements
31	Improved save as... logic	Released	Matt Peterson	2007-08-22 02:08:42 PM	Enhancements
33	Multilingual Support for Browser	Released	Terry Johnson	2007-12-10 12:37:39 PM	Multilingual
34	Integration with Requirements System	Released	Tim Simpson	2007-12-10 01:38:41 PM	Requirements Integration
35	Remove 64k limit on Memo Sizes	Released	Dean Stewart	2007-12-10 01:39:51 PM	Removal of 64k memo limit
36	[Customer Support] The routing rules select the wrong owner under certain conditions	Released	Dean Stewart	2007-12-10 01:41:20 PM	Routing Rules Change
40	Notification Subscriptions	Released	Tim Simpson	2007-12-10 01:42:32 PM	Notification Engine
41	Auto report generation	Released	Judith Brown	2007-12-10 01:44:10 PM	Notification Engine

In the screenshot above, a specification list report, called*Specification List Report--Defect Track 7.0*, is created. The list is sorted by specification ID, and is displayed in an ascending order.

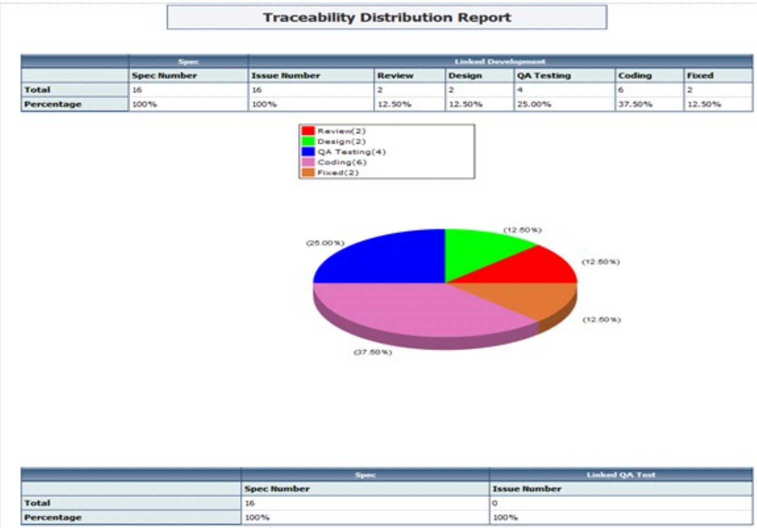
This report allows project members to view all specifications in the*Defect Tracker 7.0*folder of the*Scrum Design Project*. Project members are also able to tell from this report when a specification was created, who currently owns it, and what its status is.

Tip: Once a report is created, users can use the filter controls in the report bar to dynamically customize the report. For more information on using these filter controls in the report bar, please see section 1,*Report Basics*, earlier in this chapter.

2.2 Traceability Distribution Report

A traceability distribution report shows the relative distribution of a set of linked work items within a population, based on its property values. The traceability distribution report displays the raw data of the report in a tabular list, and a graphical representation of the report data as a pie chart. The traceability distribution report enables project team members to view the distribution of the work items linked to specifications based on their workflow state.

Below is an example of a traceability distribution report:



Traceability Distribution Report

	Spec	Linked Development					
	Spec Number	Issue Number	Review	Design	QA Testing	Coding	Fixed
Total	16	16	2	2	4	6	2
Percentage	100%	100%	12.50%	12.50%	25.00%	37.50%	12.50%

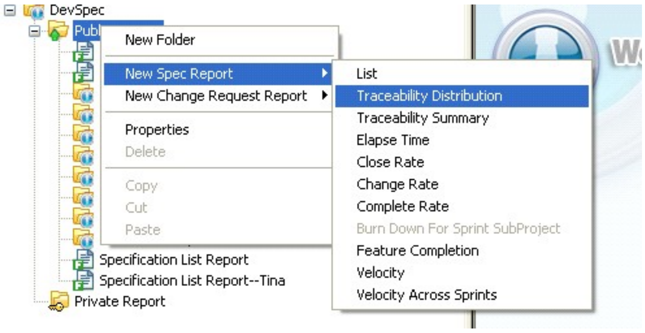
	Spec	Linked QA Test
	Spec Number	Issue Number
Total	16	0
Percentage	100%	100%

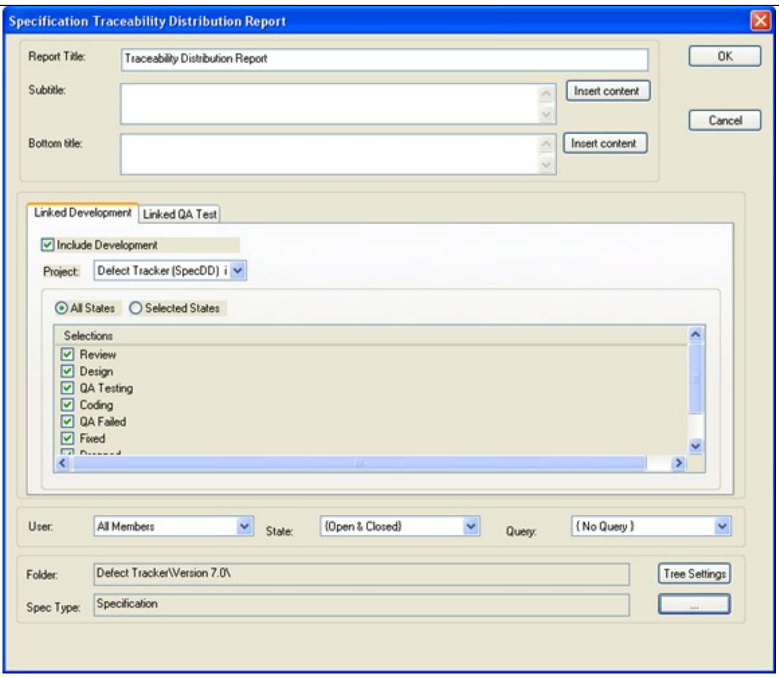
Creating a traceability distribution report

To create a traceability distribution report:

Go to the*Report*view by clicking the button on the tool bar

In the tree panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder (or any subfolder underneath them). Select*NewSpec Report>Traceability Distribution.*The*SpecificationTraceability Distribution Report*window appears.




Specification Traceability Distribution Report

Report Title: Traceability Distribution Report

Subtitle:

Bottom title:

Linked Development | Linked QA Test

☑ Include Development

Project: Defect Tracker (SpecDD)

● All States ○ Selected States

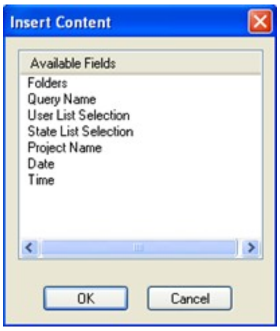
Selections
☑ Review
☑ Design
☑ QA Testing
☑ Coding
☑ QA Failed
☑ Fixed

User: All Members State: (Open & Closed) Query: (No Query)

Folder: Defect Tracker\Version 7.0\

Spec Type: Specification

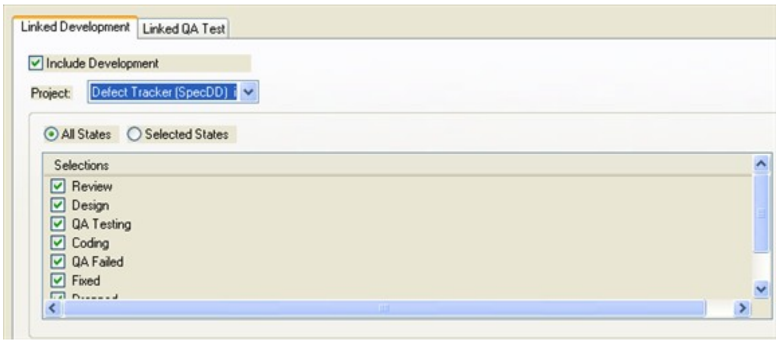
Define the report title, subtitle, and bottom title. Users can also insert system fields by clicking the*Insert content*button. Highlight the fields that you would like to add, and click the*OK*button.


Insert Content

Available Fields
Folders
Query Name
User List Selection
State List Selection
Project Name
Date
Time

Select the linked development project associated with this DevSpec project. Select the*Include Development*checkbox, and choose the correct DevTrack project from the*Project*dropdown list.

The report can be based on all states in the selected DevTrack project, or users can select a set of states to narrow down the report coverage. To include all states in the report, select the*All States*radio button. To include only a set of states, select the*Selected States*radio button.


Linked Development | Linked QA Test

☑ Include Development

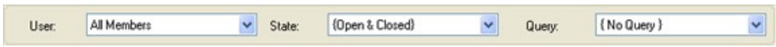
Project: Defect Tracker (SpecDD)

● All States ○ Selected States

Selections
☑ Review
☑ Design
☑ QA Testing
☑ Coding
☑ QA Failed
☑ Fixed

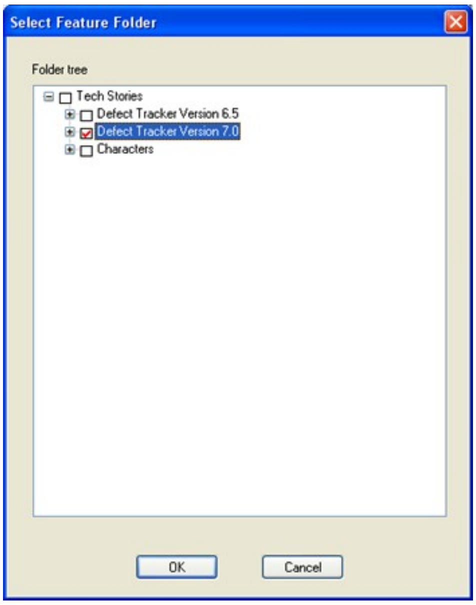
Go to the*Linked QA Test*tab to configure the linked DevTest project. Follow steps 4 and 5 to display the DevTest linked issue distribution in the report. Otherwise, uncheck the*Linked DevTest*checkbox.

The*User, State*and*Query*dropdown list fields allow you to filter the data fetched in the report.


User: All Members State: (Open & Closed) Query: (No Query)

Note: These three filters can also be found in the tool bar in the report view.

Define the report scope. Click the*Tree Settings*button to select a specific branch of the specification folder tree.



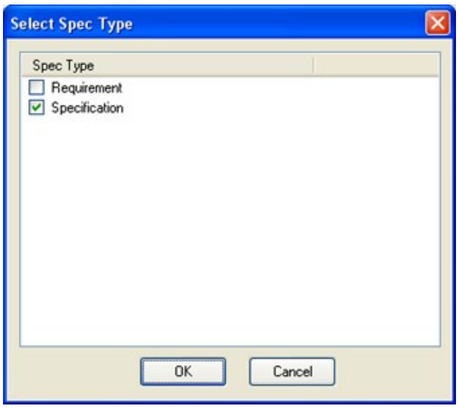
Select Feature Folder

Folder tree

- ⊟ □ Tech Stories
  - ⊞ □ Defect Tracker Version 6.5
  - ⊞ ☑ Defect Tracker Version 7.0
  - ⊞ □ Characters

Note: When a parent folder is selected, all child folders will be selected automatically.

Define which item types are covered in the report. Click the ellipses (…) button to select one or more item types.



Select Spec Type

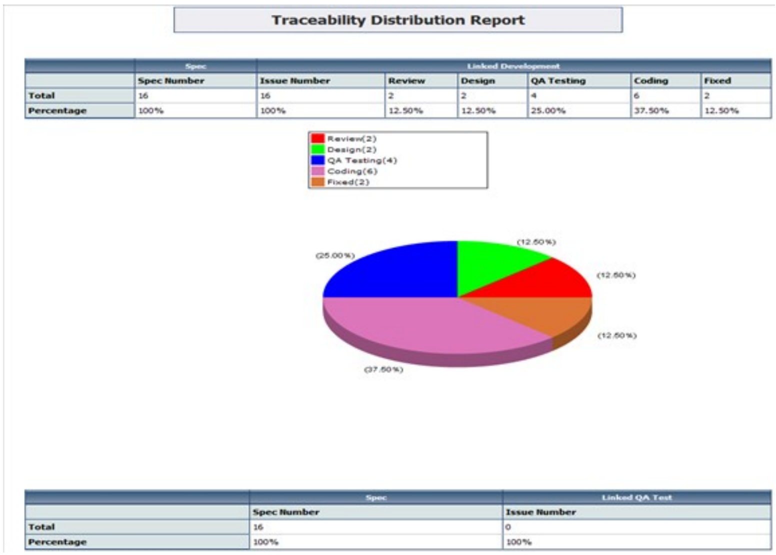
Spec Type

□ Requirement

☑ Specification

Note: Choosing the spec type to report on is mandatory. If no spec type is selected, no data will be returned.

10. Click the*OK*button.



Traceability Distribution Report

	Spec	Linked Development					
	Spec Number	Issue Number	Review	Design	QA Testing	Coding	Fixed
Total	16	16	2	2	4	6	2
Percentage	100%	100%	12.50%	12.50%	25.00%	37.50%	12.50%

	Spec	Linked QA Test
	Spec Number	Issue Number
Total	16	0
Percentage	100%	100%

In the example shown here, a traceability distribution report is generated. Project members can view the total number of specifications, as well as the status breakdown of each linked development item. Percentages are also displayed in the pie chart.

Of the 16 specifications in the report, there are 16 linked development issues, with four in QA Testing (25% of the total). From reading the chart, project members can easily get an overall idea on the status of development issues for a specific area of work.

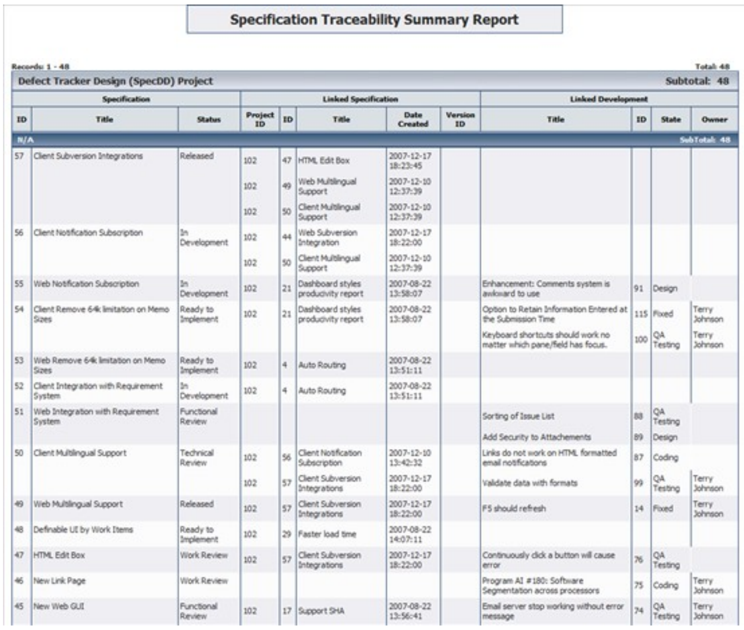
Tip: Once a report is created, users can use controls in the report bar to dynamically customize the report. For more information on using controls in the report bar, please see section 1 of this chapter.

2.3 Traceability Summary Report

A traceability summary report provides a summary of a subset of linked work items in a list report. The traceability summary report displays raw data for linked work items in a tabular list report of rows and columns.

In specification traceability summary reports, selected data for linked work items (specifications, development issues, and test tasks) is displayed in a tabular list report.

Below is an example of a traceability distribution report:



Specification Traceability Summary Report

Records: 1 - 48 Total: 48

Defect Tracker Design (SpecDD) Project Subtotal: 48

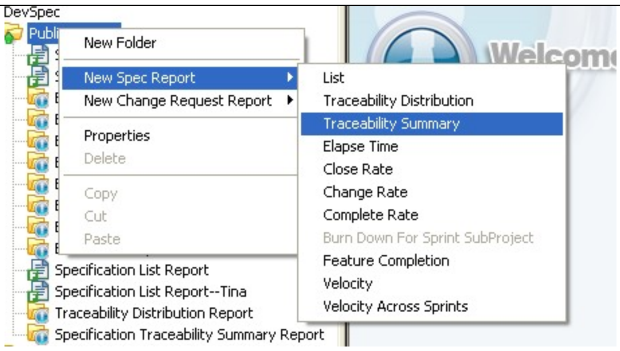
Specification			Linked Specification					Linked Development			
ID	Title	Status	Project ID	ID	Title	Date Created	Version ID	Title	ID	State	Owner
N/A											SubTotal: 48
57	Client Subversion Integrations	Released	102	47	HTML Edit Box	2007-12-17 18:23:45					
			102	49	Web Multilingual Support	2007-12-10 12:37:39					
			102	50	Client Multilingual Support	2007-12-10 12:37:39					
56	Client Notification Subscription	In Development	102	44	Web Subversion Integration	2007-12-17 18:22:00					
			102	50	Client Multilingual Support	2007-12-10 12:37:39					
55	Web Notification Subscription	In Development	102	21	Dashboard styles producivity report	2007-08-22 13:58:07		Enhancement: Comments system is awkward to use	91	Design	
54	Client Remove 64k limitation on Memo Sizes	Ready to Implement	102	21	Dashboard styles producivity report	2007-08-22 13:58:07		Option to Retain Information Entered at the Submission Time	115	Fixed	Terry Johnson
								Keyboard shortcuts should work no matter which pane/field has focus.	100	QA Testing	Terry Johnson
53	Web Remove 64k limitation on Memo Sizes	Ready to Implement	102	4	Auto Routing	2007-08-22 13:51:11					
52	Client Integration with Requirement System	In Development	102	4	Auto Routing	2007-08-22 13:51:11					
51	Web Integration with Requirement System	Functional Review						Sorting of Issue List	88	QA Testing	
								Add Security to Attachements	89	Design	
50	Client Multilingual Support	Technical Review	102	56	Client Notification Subscription	2007-12-10 13:42:32		Links do not work on HTML formatted email notifications	87	Coding	
			102	57	Client Subversion Integrations	2007-12-17 18:22:00		Validate data with formats	99	QA Testing	Terry Johnson
49	Web Multilingual Support	Released	102	57	Client Subversion Integrations	2007-12-17 18:22:00		F5 should refresh	14	Fixed	Terry Johnson
48	Definable UI by Work Items	Ready to Implement	102	29	Faster load time	2007-08-22 14:07:11					
47	HTML Edit Box	Work Review	102	57	Client Subversion Integrations	2007-12-17 18:22:00		Continuously click a button will cause error	76	QA Testing	
46	New Link Page	Work Review						Program AI #180: Software Segmentation across processors	75	Coding	Terry Johnson
45	New Web GUI	Functional Review	102	17	Support SHA	2007-08-22 13:56:41		Email server stop working without error message	74	QA Testing	Terry Johnson

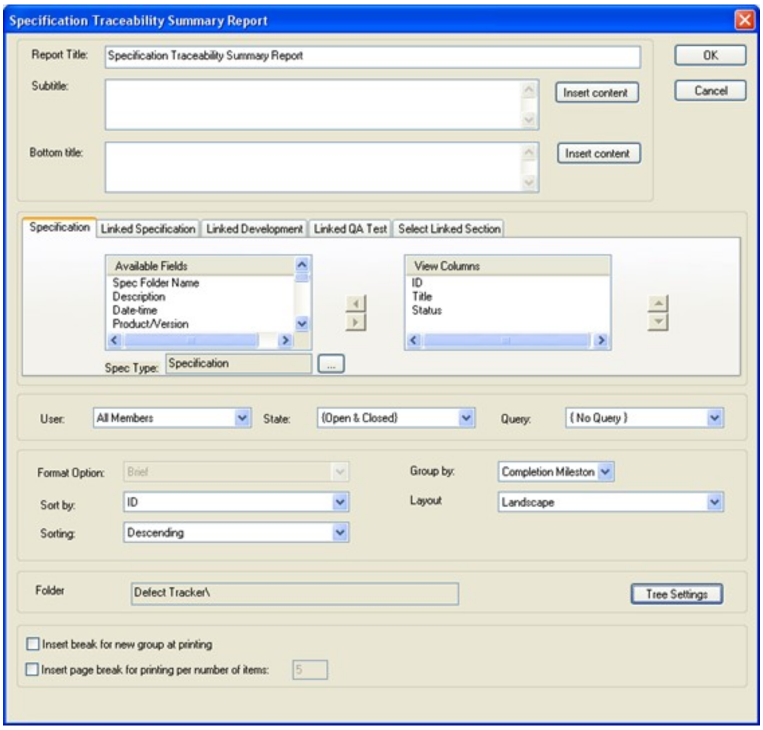
Creating a traceability summary report

To create a traceability summary report:

1. Go to the*Report*view by clicking the button in the tool bar.

2. In the*Tree*panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder (or any subfolder underneath them). Select New*Spec Report>Traceability Summary*. The*SpecificationTraceability Summary Report*dialog appears.


DevSpec
New Folder
New Spec Report
New Change Request Report
Properties
Delete
Copy
Cut
Paste
List
Traceability Distribution
Traceability Summary
Elapse Time
Close Rate
Change Rate
Complete Rate
Burn Down For Sprint SubProject
Feature Completion
Velocity
Velocity Across Sprints
Specification List Report
Specification List Report--Tina
Traceability Distribution Report
Specification Traceability Summary Report


Specification Traceability Summary Report

Report Title: Specification Traceability Summary Report

Subtitle:

Bottom title:

Specification | Linked Specification | Linked Development | Linked QA Test | Select Linked Section

Available Fields: Spec Folder Name, Description, Date-time, Product/Version

View Columns: ID, Title, Status

Spec Type: Specification

User: All Members State: (Open & Closed) Query: (No Query)

Format Option: Brief Group by: Completion Mileston

Sort by: ID Layout: Landscape

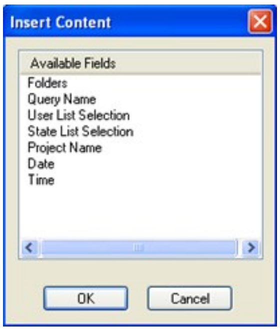
Sorting: Descending

Folder: Defect Tracker\

□ Insert break for new group at printing

□ Insert page break for printing per number of items: 5

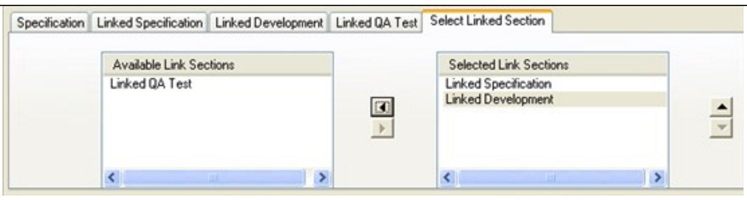
3. Define the report title, subtitle, and bottom title. Users can also insert system fields by clicking the*Insert content*button. Highlight the fields to be added to thesubtitle or bottom title, and click the*OK*button.


Insert Content

Available Fields: Folders, Query Name, User List Selection, State List Selection, Project Name, Date, Time

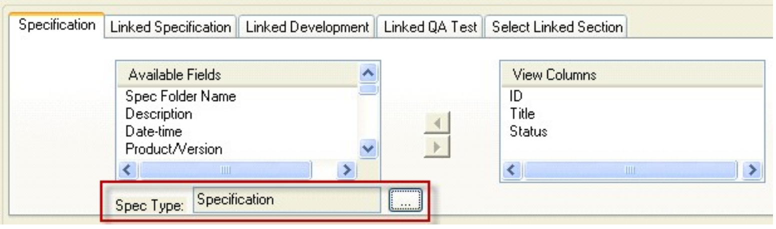
4. Define which linked items will be shown in the traceability summary report. Go to the*Select Linked Section*tab and move the desired linked work types (*Linked Specification*,*Linked Development*, and*Linked QA Test*) to the*Selected Link Sections*list on the right.

In the example shown below, the report would only show the linked specifications and the linked DevTrack development items associated with the filtered specifications.



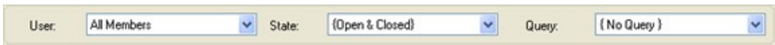
5. Go to the*Specification*tab and define which properties will be shown for the specifications, as well as each linked work types (linked specification, linked DevTrack issues, and linked DevTest issues). Users can click the ◀ ▶ buttons to add or remove fields from the*Available Fields*list to the*View Columns*list, and vice versa. Users can also set the field order by using the ▲ ▼ buttons.

Use the*Spec Type*control to define the type(s) of linked items to include in the report.**When this field is left blank, the report will display no data.**



Spec Type: Specification

6. The*User, State*and*Query*dropdown lists allow yu to filter the data fetched in the report.

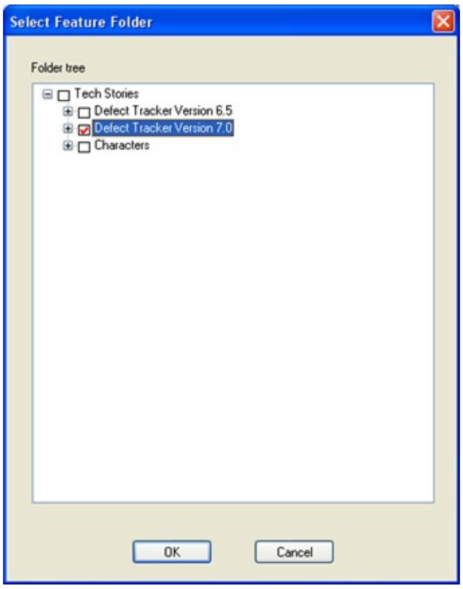


User: All Members State: (Open & Closed) Query: (No Query)

Note: These three filters can also be found in the tool bar in the report view.

7. Define how the specification list is displayed. Users can customize the report by configuring the*Group By, Sort By, Layout*and*Sorting*options.

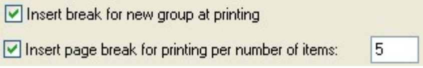
8. Define the report scope. Click the*Tree Settings*button to select a specific branch of the specification folder tree.



- □ Tech Stories
  - □ Defect Tracker Version 6.5
  - ☑ Defect Tracker Version 7.0
  - □ Characters

Note: When a parent folder is selected, all child folders will be automatically selected.

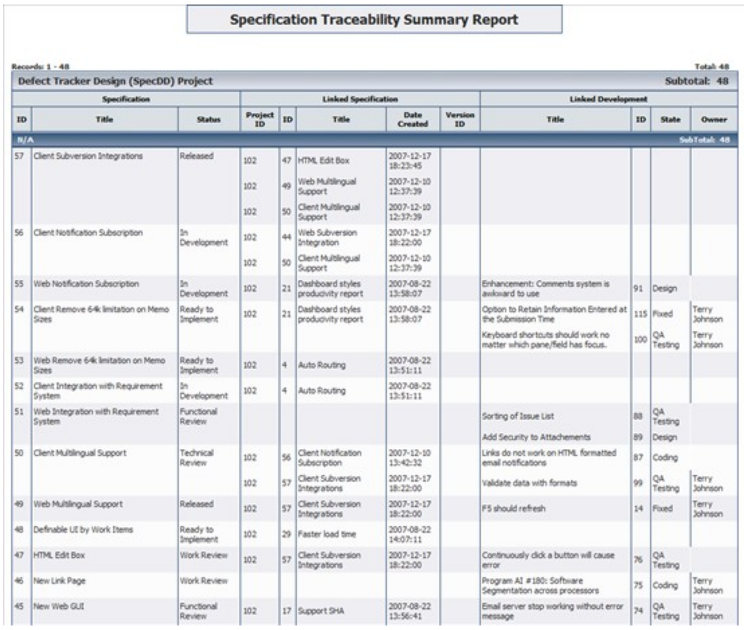
11. When printing the report, users can choose to insert a page break after each grouping of records. Page breaks can also be inserted based on the number of records. This can be configured by selecting the checkboxes, as shown below:



☑ Insert break for new group at printing

☑ Insert page break for printing per number of items: 5

12. Click the*OK*button.



Specification Traceability Summary Report

Records: 1 - 48 Total: 48

Defect Tracker Design (SpecDD) Project Subtotal: 48

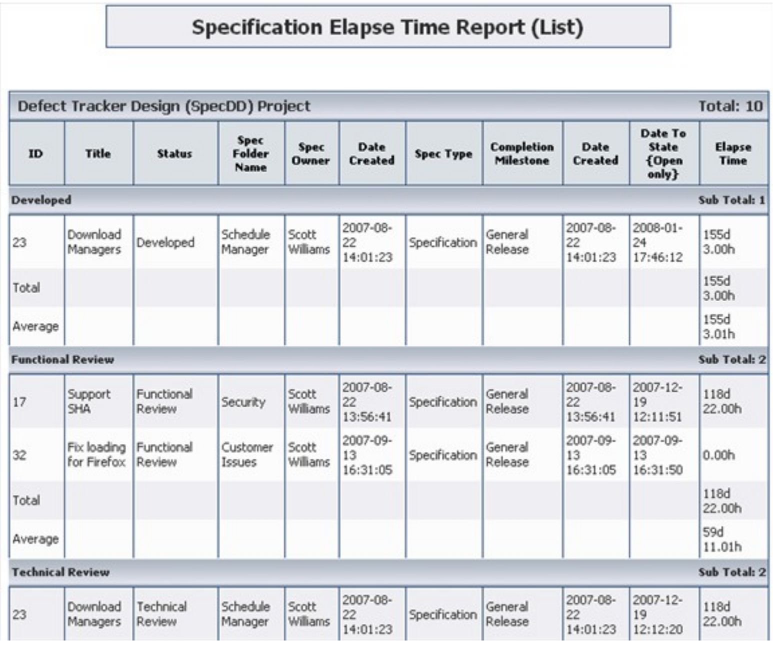
Specification			Linked Specification					Linked Development			
ID	Title	Status	Project ID	ID	Title	Date Created	Version ID	Title	ID	State	Owner
N/A											SubTotal: 48
57	Client Subversion Integrations	Released	102	47	HTML Edit Box	2007-12-17 18:23:45					
			102	49	Web Multilingual Support	2007-12-10 12:37:39					
			102	50	Client Multilingual Support	2007-12-10 12:37:39					
56	Client Notification Subscription	In Development	102	44	Web Subversion Integration	2007-12-17 18:22:00					
			102	50	Client Multilingual Support	2007-12-10 12:37:39					
55	Web Notification Subscription	In Development	102	21	Dashboard styles productivity report	2007-08-22 13:58:07		Enhancement: Comments system is awkward to use	91	Design	
54	Client Remove 64k limitation on Memo Sizes	Ready to Implement	102	21	Dashboard styles productivity report	2007-08-22 13:58:07		Option to Retain Information Entered at the Submission Time	115	Fixed	Terry Johnson
								Keyboard shortcuts should work no matter which pane/field has focus.	100	QA Testing	Terry Johnson
53	Web Remove 64k limitation on Memo Sizes	Ready to Implement	102	4	Auto Routing	2007-08-22 13:51:11					
52	Client Integration with Requirement System	In Development	102	4	Auto Routing	2007-08-22 13:51:11					
51	Web Integration with Requirement System	Functional Review						Sorting of Issue List	88	QA Testing	
								Add Security to Attachements	89	Design	
50	Client Multilingual Support	Technical Review	102	56	Client Notification Subscription	2007-12-10 13:42:32		Links do not work on HTML formatted email notifications	87	Coding	
			102	57	Client Subversion Integrations	2007-12-17 18:22:00		Validate data with formats	99	QA Testing	Terry Johnson
49	Web Multilingual Support	Released	102	57	Client Subversion Integrations	2007-12-17 18:22:00		F5 should refresh	14	Fixed	Terry Johnson
48	Definable UI by Work Items	Ready to Implement	102	29	Faster load time	2007-08-22 14:07:11					
47	HTML Edit Box	Work Review	102	57	Client Subversion Integrations	2007-12-17 18:22:00		Continuously click a button will cause error	76	QA Testing	
46	New Link Page	Work Review						Program AI #180: Software Segmentation across processors	75	Coding	Terry Johnson
45	New Web GUI	Functional Review	102	17	Support SHA	2007-08-22 13:56:41		Email server stop working without error message	74	QA Testing	Terry Johnson

Tips: Once a report is created, users can use the controls in the report bar to dynamically customize it. For more information on using controls in the report bar, please see Section 1 of this chapter.

2.4 Specification Elapse Time Report

Specification elapse time reports show the time that has elapsed between the date a specification was submitted and the date the specification first entered, last entered, or last left a particular workflow state or state status (open or closed).

The list report shows detailed information about each specification that has passed through a selected state or state status, including the date and time the specification was submitted, the date and time it entered or left the selected state, and the total elapsed time between these two actions.



Specification Elapse Time Report (List)

Defect Tracker Design (SpecDD) Project Total: 10

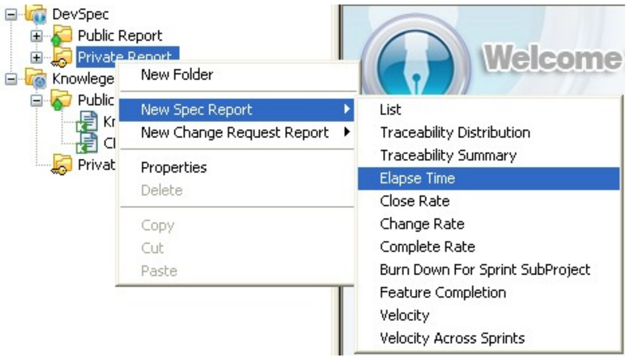
ID	Title	Status	Spec Folder Name	Spec Owner	Date Created	Spec Type	Completion Milestone	Date Created	Date To State {Open only}	Elapse Time
Developed										Sub Total: 1
23	Download Managers	Developed	Schedule Manager	Scott Williams	2007-08-22 14:01:23	Specification	General Release	2007-08-22 14:01:23	2008-01-24 17:46:12	155d 3.00h
Total										155d 3.00h
Average										155d 3.01h
Functional Review										Sub Total: 2
17	Support SHA	Functional Review	Security	Scott Williams	2007-08-22 13:56:41	Specification	General Release	2007-08-22 13:56:41	2007-12-19 12:11:51	118d 22.00h
32	Fix loading for Firefox	Functional Review	Customer Issues	Scott Williams	2007-09-13 16:31:05	Specification	General Release	2007-09-13 16:31:05	2007-09-13 16:31:50	0.00h
Total										118d 22.00h
Average										59d 11.01h
Technical Review										Sub Total: 2
23	Download Managers	Technical Review	Schedule Manager	Scott Williams	2007-08-22 14:01:23	Specification	General Release	2007-08-22 14:01:23	2007-12-19 12:12:20	118d 22.00h

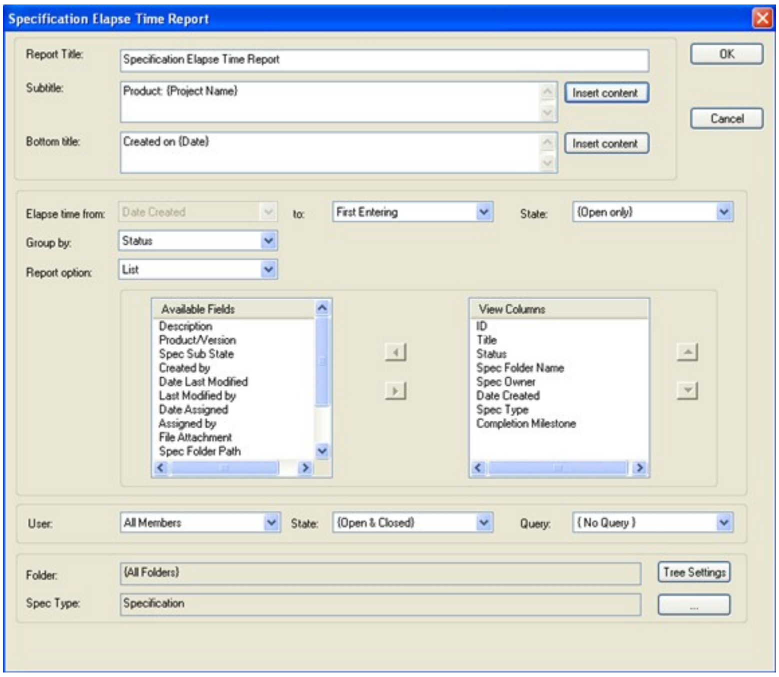
Creating a specification elapse time report

To create a specification elapse time report:

1. Go to the*Report*view by clicking the button in the tool bar..

2. In the*Tree*panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report>Elapse Time*.




Specification Elapse Time Report

Report Title: Specification Elapse Time Report

Subtitle: Product: {Project Name}

Bottom title: Created on {Date}

Elapse time from: Date Created to: First Entering State: (Open only)

Group by: Status

Report option: List

Available Fields: Description, Product/Version, Spec Sub State, Created by, Date Last Modified, Last Modified by, Date Assigned, Assigned by, File Attachment, Spec Folder Path

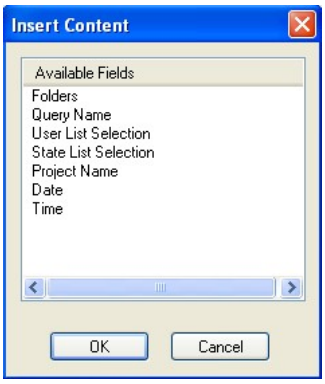
View Columns: ID, Title, Status, Spec Folder Name, Spec Owner, Date Created, Spec Type, Completion Milestone

User: All Members State: (Open & Closed) Query: (No Query)

Folder: (All Folders)

Spec Type: Specification

3. Edit the title, subtitle, and bottom title fields. Users can also insert certain values from the database, by clicking the*Insert content*button.

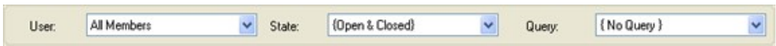

Insert Content

Available Fields: Folders, Query Name, User List Selection, State List Selection, Project Name, Date, Time

4. Users can click the ◀ ▶ buttons to add or remove fields from the*Available Fields*list to the*View Columns*list, and vice versa.

Users can also set the field order by using the ▲ ▼ buttons.

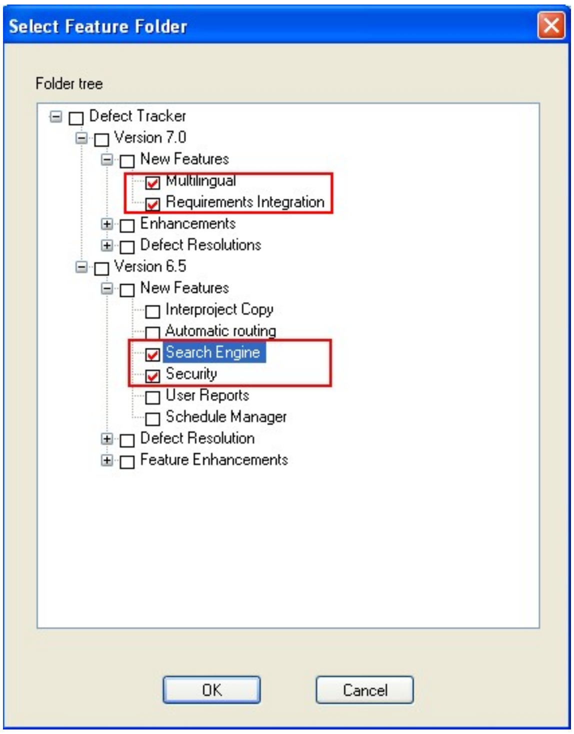
5. The*User, State*and*Query*dropdown lists allow you to filter the data fetched in the report.



User: All Members State: (Open & Closed) Query: (No Query)

6. Users can select a field in the*Group By*dropdown list to define the grouping of data in the report.

7. Click the*Tree Settings*button to select a specific branch of the specification folder tree that will be used to fetch the data in the report.

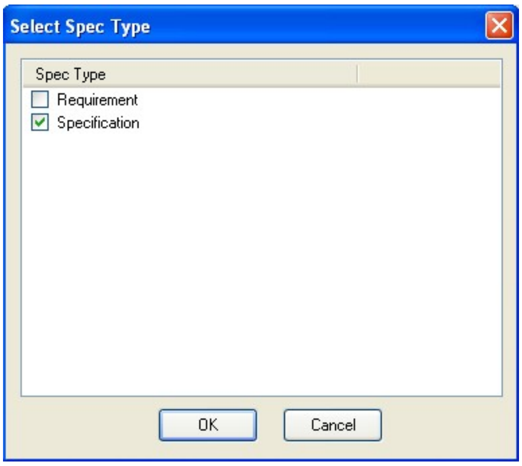


Select Feature Folder

Folder tree

- □ Defect Tracker
  - □ Version 7.0
    - □ New Features
      - ☑ Multilingual
      - ☑ Requirements Integration
    - □ Enhancements
    - □ Defect Resolutions
  - □ Version 6.5
    - □ New Features
      - □ Interproject Copy
      - □ Automatic routing
      - ☑ Search Engine
      - ☑ Security
      - □ User Reports
      - □ Schedule Manager
    - □ Defect Resolution
    - □ Feature Enhancements

8. Click the ellipses (...)button to select the type: specification, requirement, or both.



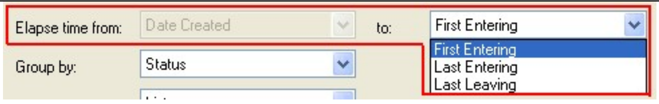
Select Spec Type

Spec Type

□ Requirement

☑ Specification

9. Select the Elapse time to option from the drop down list

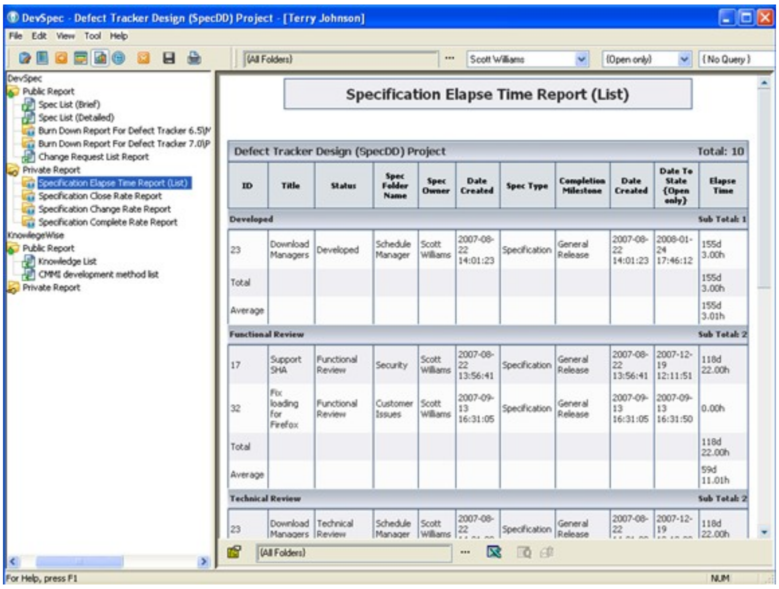


Elapse time from: Date Created to: First Entering

Group by: Status

10. Users can also choose between list or summary report in the*Report Option*dropdown field.

List Report



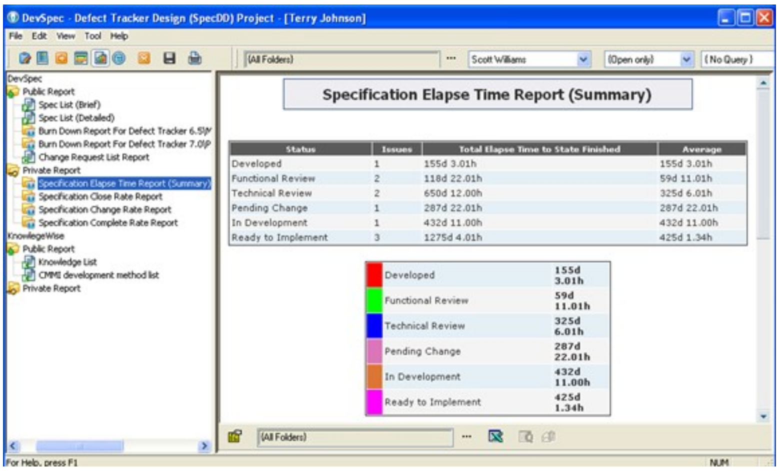
DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]
File Edit View Tool Help
(All Folders)
Scott Williams
(Open only)
{ No Query }
DevSpec
Public Report
Spec List (Brief)
Spec List (Detailed)
Burn Down Report For Defect Tracker 6.5\M
Burn Down Report For Defect Tracker 7.0\P
Change Request List Report
Private Report
Specification Elapse Time Report (List)
Specification Close Rate Report
Specification Change Rate Report
Specification Complete Rate Report
KnowledgeWise
Public Report
Knowledge List
CMMI development method list
Private Report
For Help, press F1
NUM

Specification Elapse Time Report (List)

Defect Tracker Design (SpecDD) Project Total: 10

ID	Title	Status	Spec Folder Name	Spec Owner	Date Created	Spec Type	Completion Milestone	Date Created	Date To State {Open only}	Elapse Time
Developed										**Sub Total: 1**
23	Download Managers	Developed	Schedule Manager	Scott Williams	2007-08-22 14:01:23	Specification	General Release	2007-08-22 14:01:23	2008-01-24 17:46:12	155d 3.00h
Total										155d 3.00h
Average										155d 3.01h
Functional Review										**Sub Total: 2**
17	Support SHA	Functional Review	Security	Scott Williams	2007-08-22 13:56:41	Specification	General Release	2007-08-22 13:56:41	2007-12-19 12:11:51	118d 22.00h
32	Fix loading for Firefox	Functional Review	Customer Issues	Scott Williams	2007-09-13 16:31:05	Specification	General Release	2007-09-13 16:31:05	2007-09-13 16:31:50	0.00h
Total										118d 22.00h
Average										59d 11.01h
Technical Review										**Sub Total: 2**
23	Download Managers	Technical Review	Schedule Manager	Scott Williams	2007-08-22	Specification	General Release	2007-08-22	2007-12-19	118d 22.00h

Summary Report



DevSpec - Defect Tracker Design (SpecDD) Project - [Terry Johnson]
File Edit View Tool Help
(All Folders)
Scott Williams
(Open only)
{ No Query }
DevSpec
Public Report
Spec List (Brief)
Spec List (Detailed)
Burn Down Report For Defect Tracker 6.5\M
Burn Down Report For Defect Tracker 7.0\P
Change Request List Report
Private Report
Specification Elapse Time Report (Summary)
Specification Close Rate Report
Specification Change Rate Report
Specification Complete Rate Report
KnowledgeWise
Public Report
Knowledge List
CMMI development method list
Private Report
For Help, press F1
NUM

Specification Elapse Time Report (Summary)

Status	Issues	Total Elapse Time to State Finished	Average
Developed	1	155d 3.01h	155d 3.01h
Functional Review	2	118d 22.01h	59d 11.01h
Technical Review	2	650d 12.00h	325d 6.01h
Pending Change	1	287d 22.01h	287d 22.01h
In Development	1	432d 11.00h	432d 11.00h
Ready to Implement	3	1275d 4.01h	425d 1.34h

🟥	Developed	155d 3.01h
🟩	Functional Review	59d 11.01h
🟦	Technical Review	325d 6.01h
🟪	Pending Change	287d 22.01h
🟧	In Development	432d 11.00h
🟪	Ready to Implement	425d 1.34h

11. The button at the bottom of the report can be used to change the report properties.

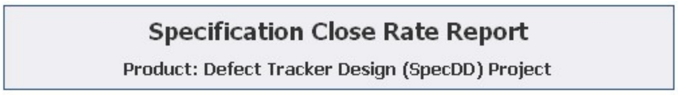
12. The button at the bottom of the report facilitates exporting to an Excel, Word, or CSV format.

13. The ··· button at the bottom of the report can be used to select a specific branch of the specification folder tree that is used to fetch the data in the report.

2.5 Close Rate Report

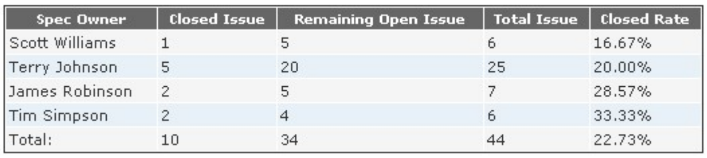
In DevSpec, a close rate report shows the number of work items that have been closed within a defined time period.

The close rate report shows the total number of work items, the number of work items closed, and the percent of work items closed within a specific time period in a tabular list report and a column chart.

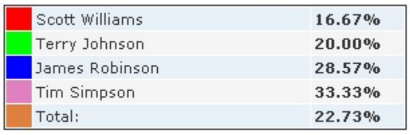


Specification Close Rate Report

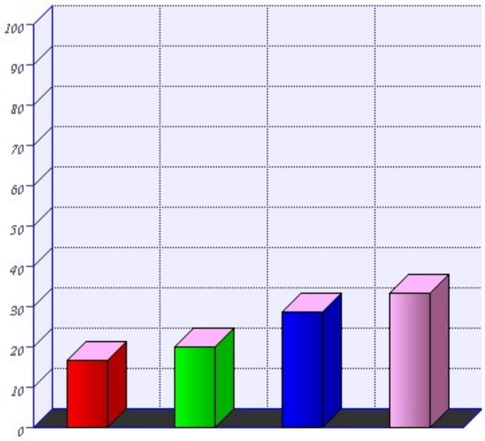
Product: Defect Tracker Design (SpecDD) Project



Spec Owner	Closed Issue	Remaining Open Issue	Total Issue	Closed Rate
Scott Williams	1	5	6	16.67%
Terry Johnson	5	20	25	20.00%
James Robinson	2	5	7	28.57%
Tim Simpson	2	4	6	33.33%
Total:	10	34	44	22.73%



🟥	Scott Williams	**16.67%**
🟩	Terry Johnson	**20.00%**
🟦	James Robinson	**28.57%**
🟪	Tim Simpson	**33.33%**
🟧	Total:	**22.73%**



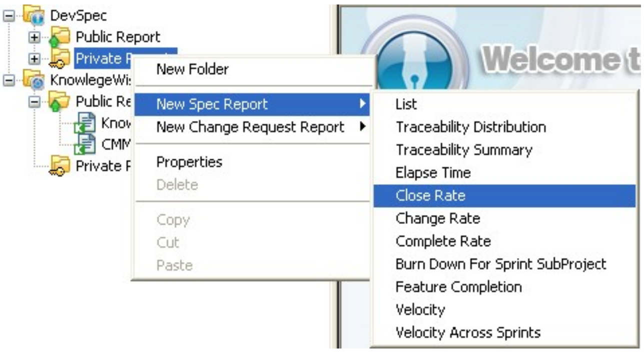
Close rate report data may be grouped by seven different work item properties. The*Group By*control defines the x-axis in the column chart. The y-axis represents the number of work items closed within the report time period.

Creating a specification close rate report

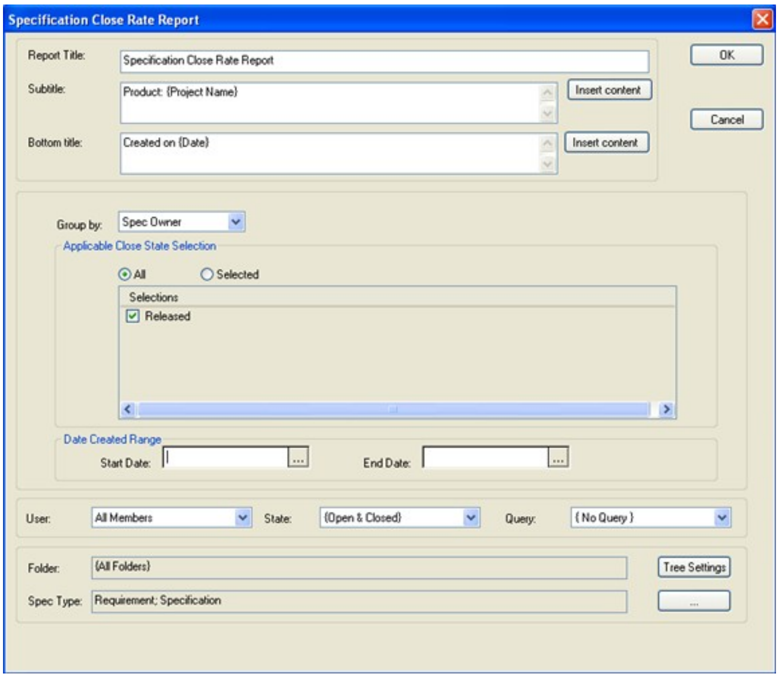
To create a specification close ratereport:

1. Go to the*Report*view by clicking the button in the tool bar.

2. In the*Tree*panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report>Close Rate*.



Note:Only users with the required privileges are able to create public reports. All reports created under this folder are accessible by all other DevSpec users. Private reports are only available to the report author. For more information on reports, please see section 1 at the beginning of this chapter.



Specification Close Rate Report

Report Title: Specification Close Rate Report

Subtitle: Product: (Project Name)

Bottom title: Created on (Date)

Group by: Spec Owner

Applicable Close State Selection

● All ○ Selected

Selections

☑ Released

Date Created Range

Start Date:

End Date:

User: All Members

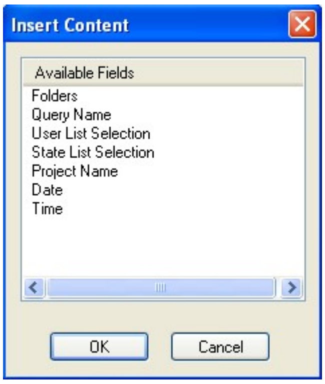
State: (Open & Closed)

Query: (No Query)

Folder: (All Folders)

Spec Type: Requirement; Specification

3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.

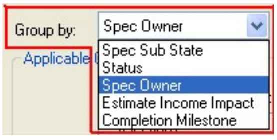


Insert Content

Available Fields

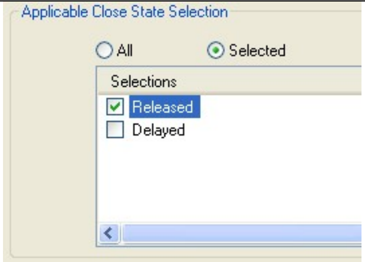
Folders
Query Name
User List Selection
State List Selection
Project Name
Date
Time

4. Users can select a field in the*Group By*dropdown list to define the grouping of data in the report.



Group by: Spec Owner

5. DevSpec administrators can define a specification workflow, so that there are multiple closed states in the specification lifecycle. Users generating reports can choose to include closed specifications in only selected closed states or all closed states.



Applicable Close State Selection

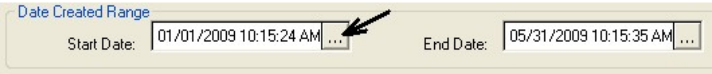
○ All ● Selected

Selections

☑ Released

□ Delayed

6. Define a date range for the report in the*Date Created Range*section. Only those specifications submitted in this date range will be included in the report.

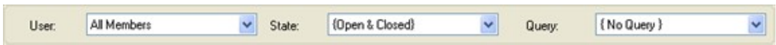


Date Created Range

Start Date: 01/01/2009 10:15:24 AM ...

End Date: 05/31/2009 10:15:35 AM ...

7. The*User*,*State*,and*Query*dropdown lists allow you to filter the data fetched in the report.

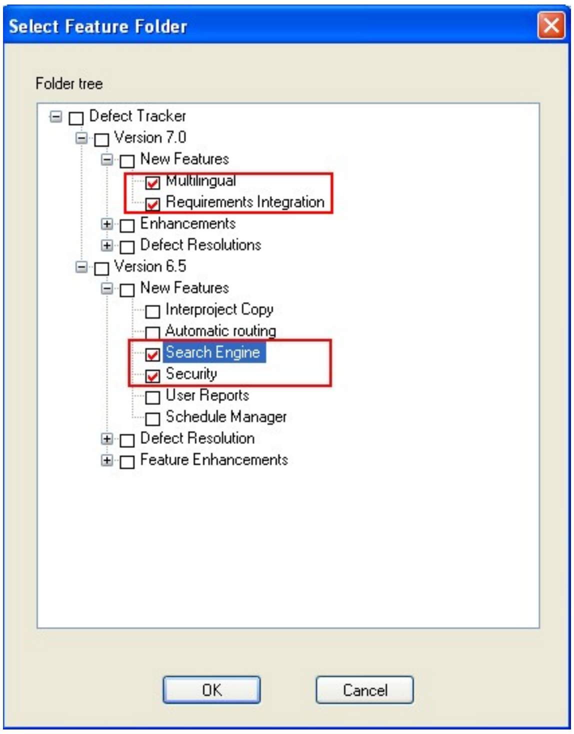


User: All Members

State: (Open & Closed)

Query: (No Query)

8. Click the*Tree Settings*button to select a specific branch of the specification folder tree that will be used to fetch the data in the report.

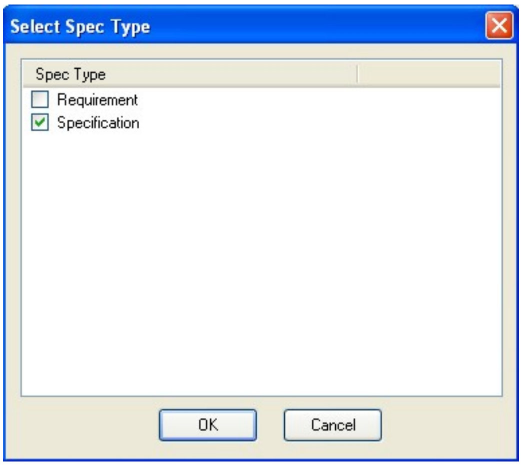


Select Feature Folder

Folder tree

- □ Defect Tracker
  - □ Version 7.0
    - □ New Features
      - ☑ Multilingual
      - ☑ Requirements Integration
    - □ Enhancements
    - □ Defect Resolutions
  - □ Version 6.5
    - □ New Features
      - □ Interproject Copy
      - □ Automatic routing
      - ☑ Search Engine
      - ☑ Security
      - □ User Reports
      - □ Schedule Manager
    - □ Defect Resolution
    - □ Feature Enhancements

9. Click the ellipses (...)button to select the type: specification, requirement, or both.



Select Spec Type

Spec Type

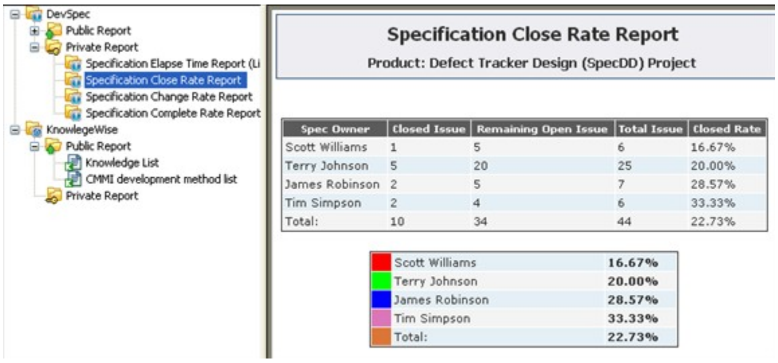
□ Requirement

☑ Specification

10. The 📝 button at the bottom of the report can be used to change the report properties.

11. The ☒ button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

12. The ··· button at the bottom of the report can be used to select a specific branch of the specification folder tree that is used to fetch the data in the report.

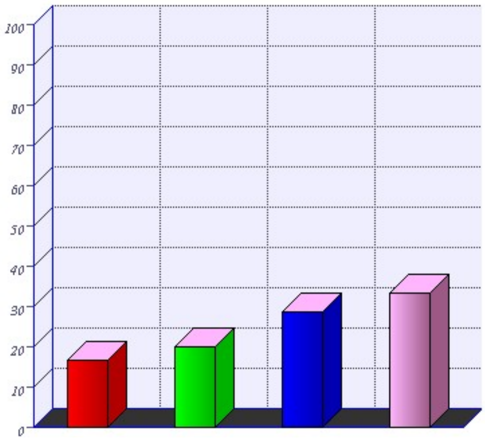

DevSpec
Public Report
Private Report
Specification Elapse Time Report (Li
Specification Close Rate Report
Specification Change Rate Report
Specification Complete Rate Report
KnowlegeWise
Public Report
Knowledge List
CMMI development method list
Private Report

Specification Close Rate Report

Product: Defect Tracker Design (SpecDD) Project

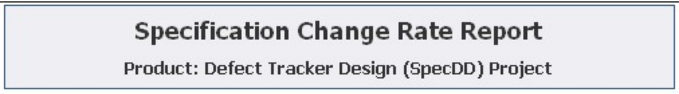
Spec Owner	Closed Issue	Remaining Open Issue	Total Issue	Closed Rate
Scott Williams	1	5	6	16.67%
Terry Johnson	5	20	25	20.00%
James Robinson	2	5	7	28.57%
Tim Simpson	2	4	6	33.33%
Total:	10	34	44	22.73%

🟥	Scott Williams	**16.67%**
🟩	Terry Johnson	**20.00%**
🟦	James Robinson	**28.57%**
🟪	Tim Simpson	**33.33%**
🟧	Total:	**22.73%**



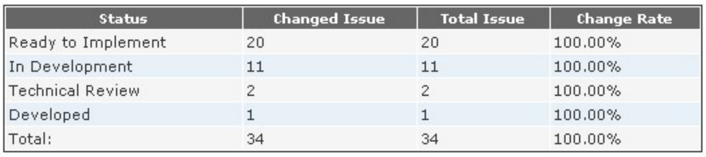
2.6 Change Rate Report

In DevSpec, a change rate report shows the number of work items that have changed their workflow state within a defined period of time.

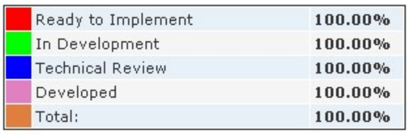


Specification Change Rate Report

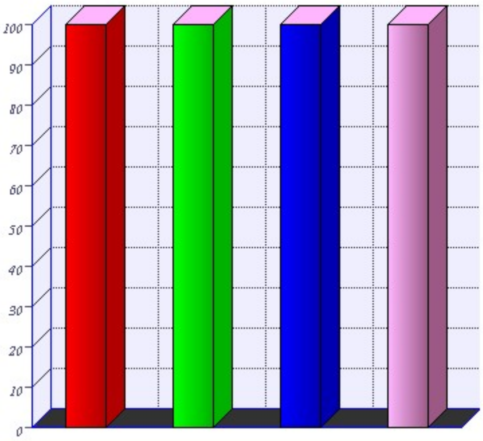
Product: Defect Tracker Design (SpecDD) Project



Status	Changed Issue	Total Issue	Change Rate
Ready to Implement	20	20	100.00%
In Development	11	11	100.00%
Technical Review	2	2	100.00%
Developed	1	1	100.00%
Total:	34	34	100.00%



🟥	Ready to Implement	100.00%
🟩	In Development	100.00%
🟦	Technical Review	100.00%
🟪	Developed	100.00%
🟧	Total:	100.00%



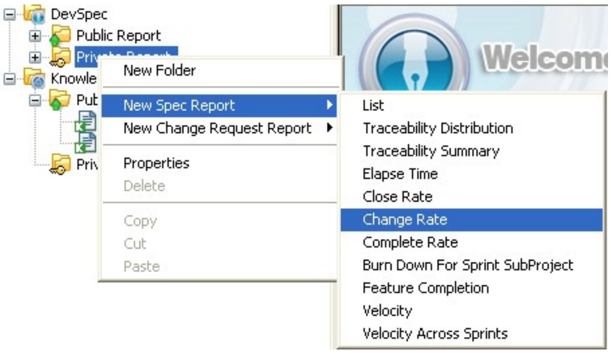
Change rate report data may be grouped by seven different work item properties. The*Group By*control defines the x-axis in the column chart. The y-axis represents the number of work items closed within the report time period.

Creating a specification change rate report

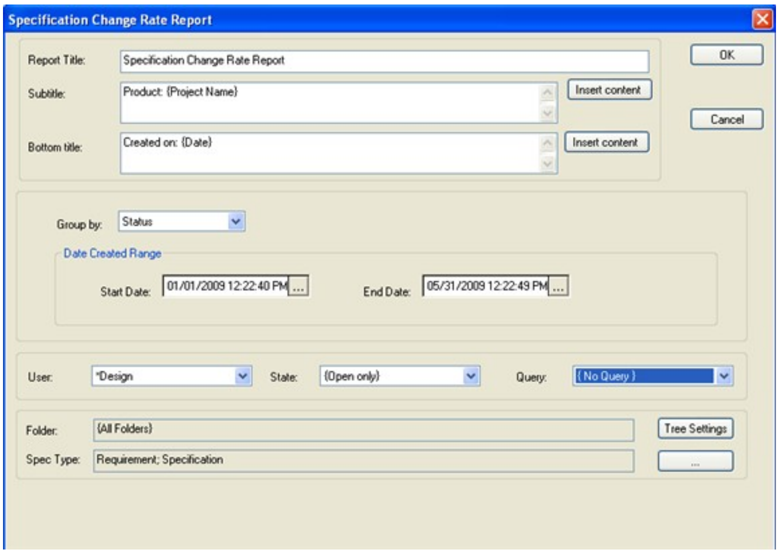
To create a specification change ratereport:

1. Go to the report view by clicking the button in the tool bar.

2. In the tree panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report*>*Change Rate*.



Note:Only users with the required privileges are able to create reports in the*Public Report*folder. Public reports are accessible to all other DevSpec users, while private reports are only available to the report author. For more information, please see section 1, at the beginning of this chapter.


Specification Change Rate Report

Report Title: Specification Change Rate Report

Subtitle: Product: {Project Name}

Bottom title: Created on: {Date}

Group by: Status

Date Created Range

Start Date: 01/01/2009 12:22:40 PM

End Date: 05/31/2009 12:22:49 PM

User: *Design

State: {Open only}

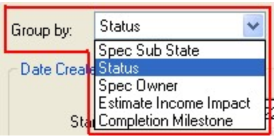
Query: { No Query }

Folder: {All Folders}

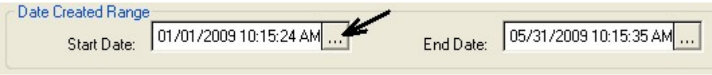
Spec Type: Requirement; Specification

3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.

4. Users can select a field in the*Group By*dropdown list to define the grouping of data in the report.


Group by: Status

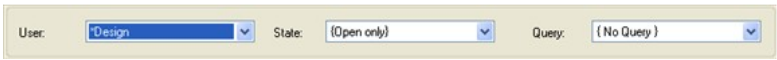
5. Define the date range. Only those specifications submitted in this date range will be included in the report.


Date Created Range

Start Date: 01/01/2009 10:15:24 AM

End Date: 05/31/2009 10:15:35 AM

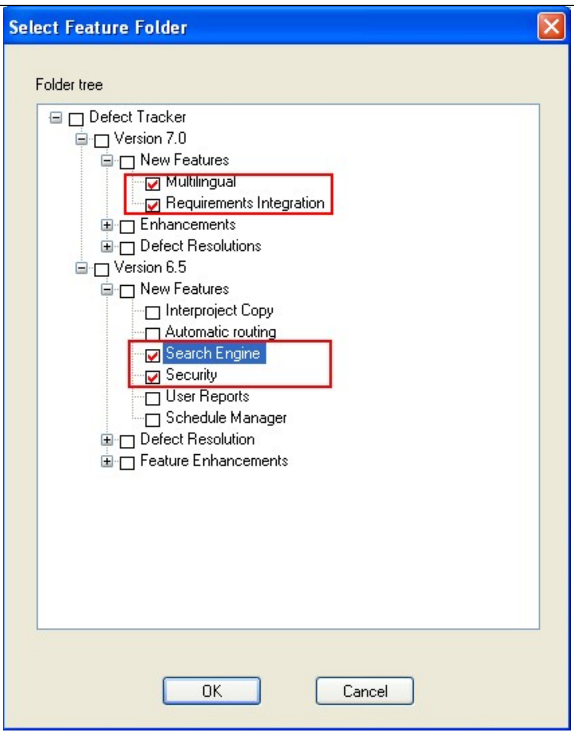
6. The*User, State*and*Query*dropdown lists allow the user to filter the data fetched in the report.


User: *Design

State: {Open only}

Query: { No Query }

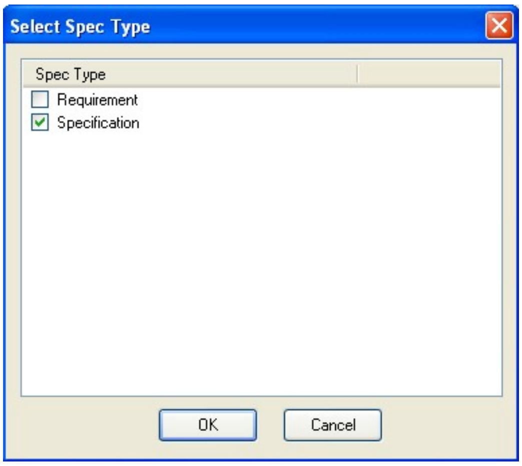
7. Click the*Tree Settings*button to select a specific branch of the specification folder tree that will be used to fetch the report data.



Folder tree

- ☐ Defect Tracker
  - ☐ Version 7.0
    - ☐ New Features
      - ☑ Multilingual
      - ☑ Requirements Integration
    - ☐ Enhancements
    - ☐ Defect Resolutions
  - ☐ Version 6.5
    - ☐ New Features
      - ☐ Interproject Copy
      - ☐ Automatic routing
      - ☑ Search Engine
      - ☑ Security
      - ☐ User Reports
      - ☐ Schedule Manager
    - ☐ Defect Resolution
    - ☐ Feature Enhancements

8. Click the ellipses (...)button to select the item type: specification, requirement, or both.



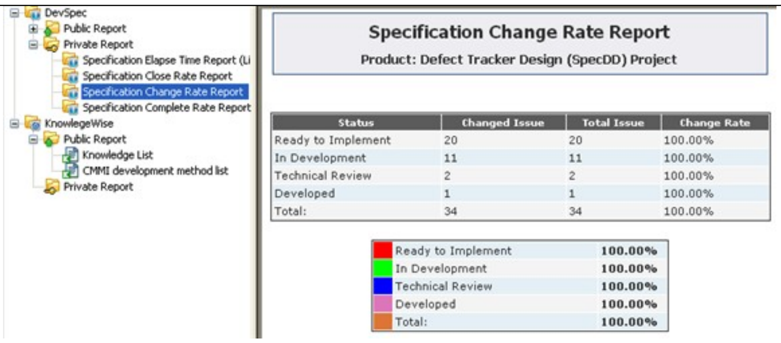
Spec Type

☐ Requirement
☑ Specification

9. The button at the bottom of the report can be used to change the report properties.

10. The button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

11. The ··· button at the bottom of the report can be used to select a specific branch of thespecification folder tree that will be used to fetch the report data.

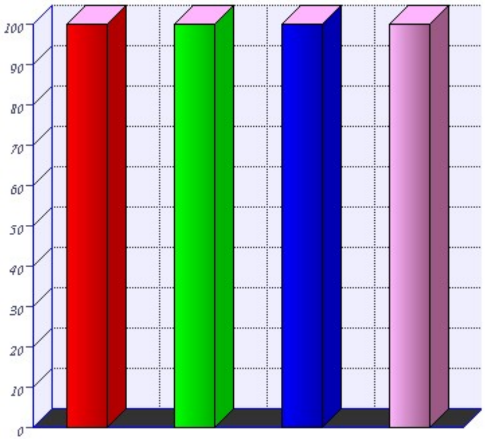

DevSpec
Public Report
Private Report
Specification Elapse Time Report (Li
Specification Close Rate Report
Specification Change Rate Report
Specification Complete Rate Report
KnowlegeWise
Public Report
Knowledge List
CMMI development method list
Private Report

Specification Change Rate Report

Product: Defect Tracker Design (SpecDD) Project

Status	Changed Issue	Total Issue	Change Rate
Ready to Implement	20	20	100.00%
In Development	11	11	100.00%
Technical Review	2	2	100.00%
Developed	1	1	100.00%
Total:	34	34	100.00%

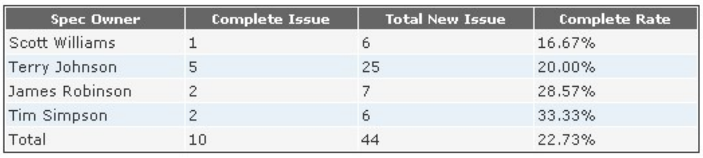
🟥	Ready to Implement	100.00%
🟩	In Development	100.00%
🟦	Technical Review	100.00%
🟪	Developed	100.00%
🟧	Total:	100.00%



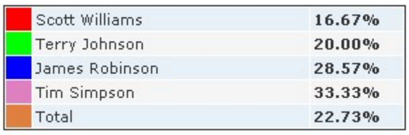
2.7 Complete Rate Report

In DevSpec, a close rate report shows the number of work items that have been completed within a defined period of time.

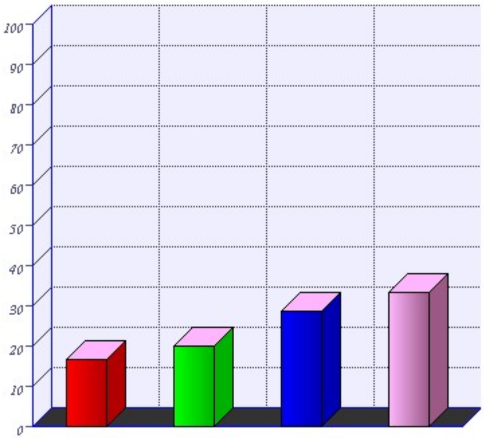
Specification Complete Rate Report



Spec Owner	Complete Issue	Total New Issue	Complete Rate
Scott Williams	1	6	16.67%
Terry Johnson	5	25	20.00%
James Robinson	2	7	28.57%
Tim Simpson	2	6	33.33%
Total	10	44	22.73%



🟥	Scott Williams	**16.67%**
🟩	Terry Johnson	**20.00%**
🟦	James Robinson	**28.57%**
🟪	Tim Simpson	**33.33%**
🟧	Total	**22.73%**



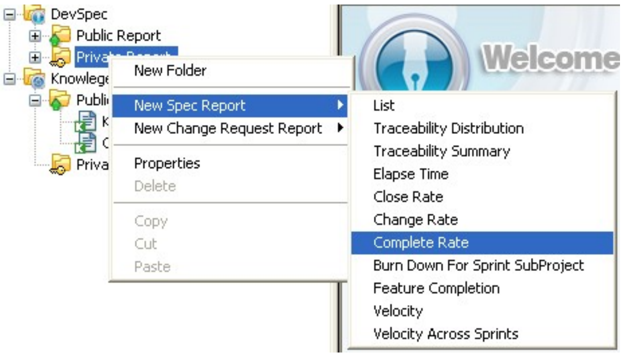
Complete rate report data may be grouped by seven different work item properties. The*Group By*control defines the x-axis in the column chart. The y-axis represents the number of work items completed within the report time period.

Creating a specification complete rate report

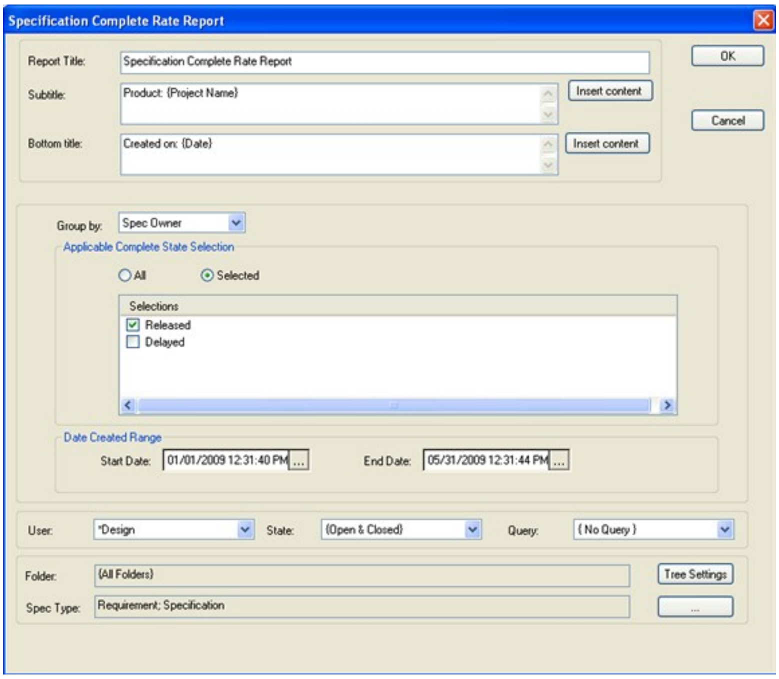
To create a specification complete ratereport:

1. Go to the report view by clicking the button on the tool bar.

2. In the tree panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report>Complete Rate*.



Note:Only users with the required privileges are able to create reports in the*Public Report*folder. Public reports are accessible to all DevSpec users, while private reports are only available to the report author. For more information, please see section 1, at the beginning of this chapter.



Specification Complete Rate Report

Report Title: Specification Complete Rate Report

Subtitle: Product: {Project Name}

Bottom title: Created on: {Date}

Group by: Spec Owner

Applicable Complete State Selection

○ All ● Selected

Selections

☑ Released

☐ Delayed

Date Created Range

Start Date: 01/01/2009 12:31:40 PM

End Date: 05/31/2009 12:31:44 PM

User: *Design

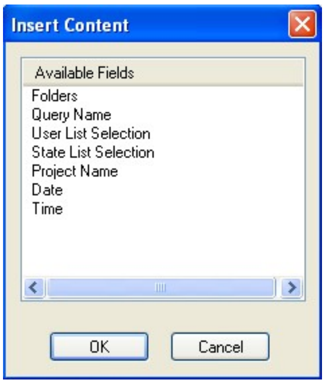
State: (Open & Closed)

Query: (No Query)

Folder: (All Folders)

Spec Type: Requirement; Specification

3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.



Insert Content

Available Fields

Folders

Query Name

User List Selection

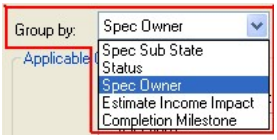
State List Selection

Project Name

Date

Time

4. Users can select a field in the*Group By*dropdown list to define the grouping of data in the report.



Group by: Spec Owner

Spec Sub State

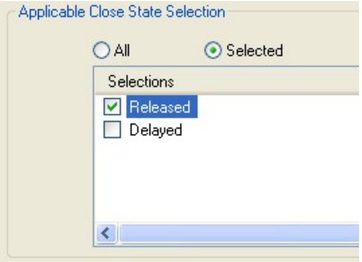
Status

Spec Owner

Estimate Income Impact

Completion Milestone

5. DevSpec administrators can define a specification workflow so that there are multiple closed states in the specification lifecycle. Users generating reports can choose to include closed specifications in only certain closed states or all closed states.



Applicable Close State Selection

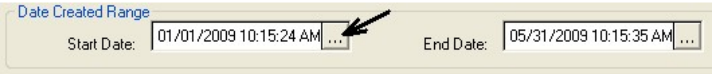
○ All ● Selected

Selections

☑ Released

☐ Delayed

6. Define the date range. Only those specifications submitted in this date range will be included in the report.

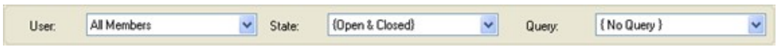


Date Created Range

Start Date: 01/01/2009 10:15:24 AM ...

End Date: 05/31/2009 10:15:35 AM ...

7. The*User, State*and*Query*dropdown list fields allow users to filter the data fetched in the report

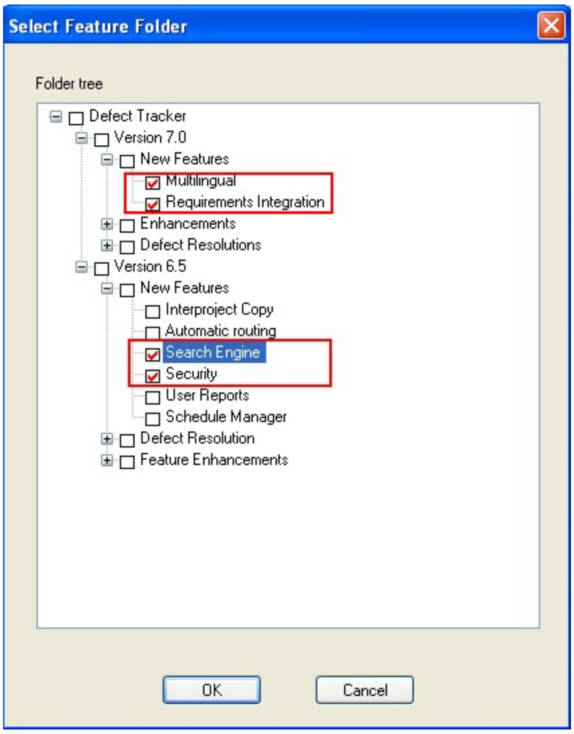


User: All Members

State: (Open & Closed)

Query: (No Query)

8. Click the*Tree Settings*button to select a specific branch of the specification folder tree that will be used to fetch the report data.

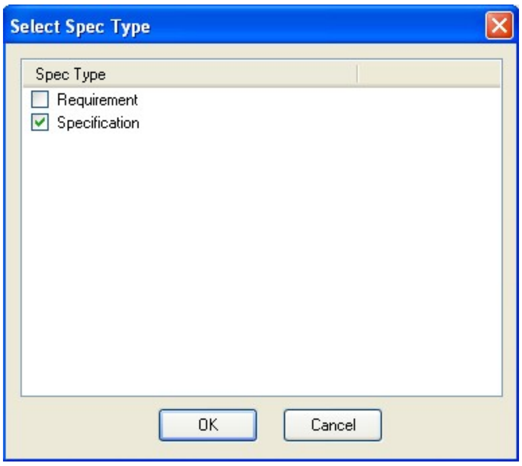


Select Feature Folder

Folder tree

- □ Defect Tracker
  - □ Version 7.0
    - □ New Features
      - ☑ Multilingual
      - ☑ Requirements Integration
    - □ Enhancements
    - □ Defect Resolutions
  - □ Version 6.5
    - □ New Features
      - □ Interproject Copy
      - □ Automatic routing
      - ☑ Search Engine
      - ☑ Security
      - □ User Reports
      - □ Schedule Manager
    - □ Defect Resolution
    - □ Feature Enhancements

9. Click theellipses (...)button to select the item type: specification, requirement, or both.



Select Spec Type

Spec Type

□ Requirement

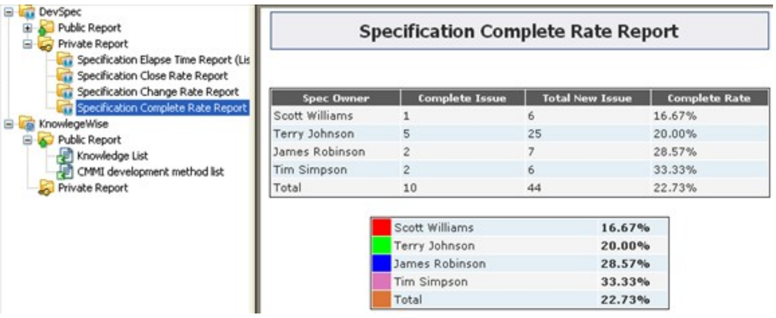
☑ Specification

10. The button at the bottom of the report can be used to change the report properties.

11. The button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

12. The ... button at the bottom of the report can be used to select a specific branch of the specification folder tree that will be

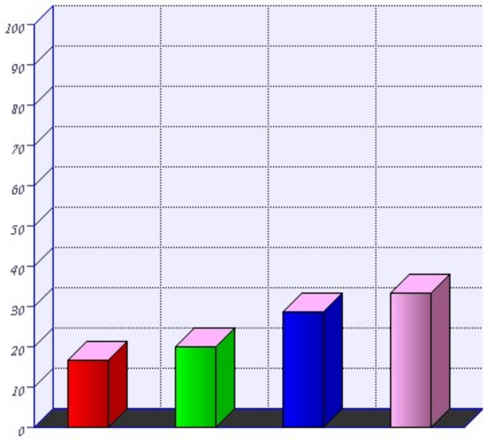
used to fetch the report data.


DevSpec
Public Report
Private Report
Specification Elapse Time Report (Lis
Specification Close Rate Report
Specification Change Rate Report
Specification Complete Rate Report
KnowlegeWise
Public Report
Knowledge List
CMMI development method list
Private Report

Specification Complete Rate Report

Spec Owner	Complete Issue	Total New Issue	Complete Rate
Scott Williams	1	6	16.67%
Terry Johnson	5	25	20.00%
James Robinson	2	7	28.57%
Tim Simpson	2	6	33.33%
Total	10	44	22.73%

🟥	Scott Williams	16.67%
🟩	Terry Johnson	20.00%
🟦	James Robinson	28.57%
🟪	Tim Simpson	33.33%
🟧	Total	22.73%



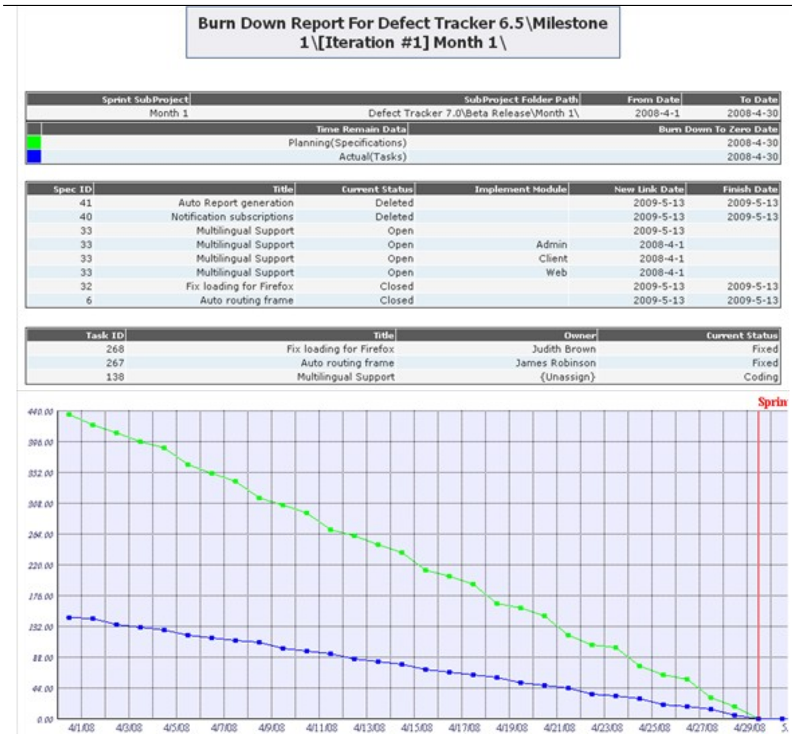
2.8 Burndown Report

In DevSpec, the burndown report shows the estimated work remaining to be done within an iterative subproject, or sprint. The amount of work remaining is based on the time estimates made when a specification is linked to an iterative subproject. The burndown compares this time estimate to the time remaining in the iteration (usually 30 days).

Burndown reports enable product and project managers to monitor the progress of a product/module development. This report is mostly applicable to agile or iterative development methodologies. Once all specifications in a subproject/iteration are scheduled and linked to a development task, all resources are allocated, and all time lines are assigned, the burndown report will displays a graph with a green line indicating the steady standard progress path from start date to finish date. The time remaining to finish all development tasks is deducted in each interval in the burndown chart.

This chart also displays a graph with a blue line representing the actual progress through each interval from start date to current date. Based on the previous progress pace, the chart then plots a graph with a dotted blue line to predict the future progress pace leading to the end of the project, indicating the projected finish date. This projected finish date may fall behind or ahead of the planned finish date. Based on this information and other supporting data, managers can compare the planned progress path against the actual progress path and take appropriate measures to adjust the project plan so that the deadline is met.

An example of a burndown report is depicted below.



Burn Down Report For Defect Tracker 6.5\Milestone 1\[Iteration #1] Month 1\

Sprint SubProject	SubProject Folder Path	From Date	To Date
Month 1	Defect Tracker 7.0\Beta Release\Month 1\	2008-4-1	2008-4-30

	Time Remain Data	Burn Down To Zero Date
🟩	Planning(Specifications)	2008-4-30
🟦	Actual(Tasks)	2008-4-30

Spec ID	Title	Current Status	Implement Module	New Link Date	Finish Date
41	Auto Report generation	Deleted		2009-5-13	2009-5-13
40	Notification subscriptions	Deleted		2009-5-13	2009-5-13
33	Multilingual Support	Open		2009-5-13	
33	Multilingual Support	Open	Admin	2008-4-1	
33	Multilingual Support	Open	Client	2008-4-1	
33	Multilingual Support	Open	Web	2008-4-1	
32	Fix loading for Firefox	Closed		2009-5-13	2009-5-13
6	Auto routing frame	Closed		2009-5-13	2009-5-13

Task ID	Title	Owner	Current Status
268	Fix loading for Firefox	Judith Brown	Fixed
267	Auto routing frame	James Robinson	Fixed
138	Multilingual Support	{Unassign}	Coding

The DevSpec burndown report is divided into three primary sections:
The specification list report

The task list report

The burndown chart

The specification list report

The specification list report shows the ID number, title, current status, implementation module, link date, and close date for every specification linked to the selected iterative subproject.

The task list report

The task list report shows the ID number, title, owner, and current status for every development issue managed within the iterative subproject.

The burndown chart

The burndown chart is a graphical representatiaon of the work that needs to be done in an iterative subproject, or sprint, over time.

In the burndown report the y-axis represents the backlog - the estimated number of hours required to implement a set of specification designs. The x-axis represents development time-the time between the date the specification was initially linked to the development task until the end of the sprint.

If multiple iterative subprojects are selected, distinct burndown charts, specification lists, and task lists are displayed for each subproject selected in the subproject tree.

Report authors may choose to display the task burndown, specification burndown, or both the task burndown and the specification burndown. Specifications are represents by a solid green line; development issues by a solid blue line.

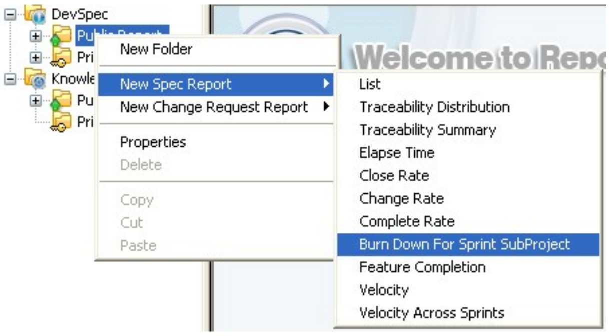
Creating a burndown list report

To create a burndown list report:

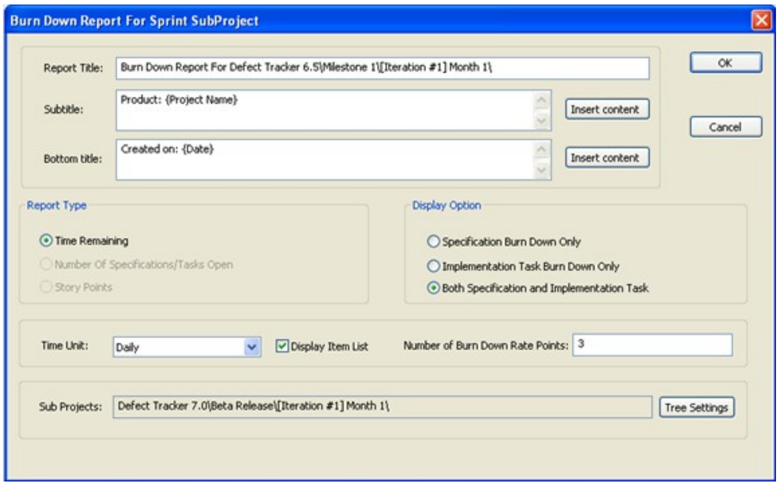
1. Go to the report view by clicking the button in the tool bar.

2. In the tree panel, under the DevSpec folder, right-click on*Private Report*or*Public Report*folder and select*New Spec*

Report>Burn Down For Sprint SubProject.



Note:Only users with the required privileges are able to create reports in the*Public Report*folder. Public reports are accessible to all other DevSpec users, while private reports are only available to the report author. For more information, please see section 1, at the beginning of this chapter.



Burn Down Report For Sprint SubProject

Report Title: Burn Down Report For Defect Tracker 6.5\Milestone 1\[Iteration #1] Month 1\

Subtitle: Product: {Project Name}

Bottom title: Created on: {Date}

Report Type

● Time Remaining

○ Number Of Specifications/Tasks Open

○ Story Points

Display Option

○ Specification Burn Down Only

○ Implementation Task Burn Down Only

● Both Specification and Implementation Task

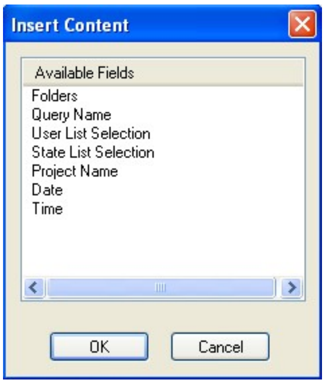
Time Unit: Daily

☑ Display Item List

Number of Burn Down Rate Points: 3

Sub Projects: Defect Tracker 7.0\Beta Release\[Iteration #1] Month 1\

3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.

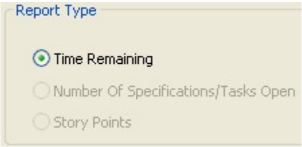


Insert Content

Available Fields

Folders
Query Name
User List Selection
State List Selection
Project Name
Date
Time

4. Users can choose a report type option.



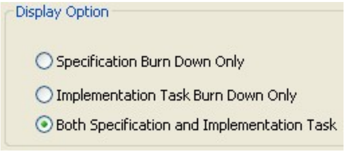
Report Type

● Time Remaining

○ Number Of Specifications/Tasks Open

○ Story Points

5. Users can also choose the*Display Option*to specify which type of data will be included in the burndown report.



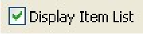
Display Option

○ Specification Burn Down Only

○ Implementation Task Burn Down Only

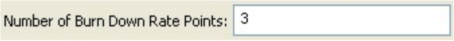
● Both Specification and Implementation Task

6. Users can check the*Display Item List*check box to include the specification list and linked DevTrack development tasks.



☑ Display Item List

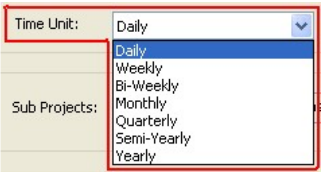
7. User can set the number of burndown rate points.



Number of Burn Down Rate Points: 3

Time Unit: The x-axis of the burndown chart represents the duration of the iterative subproject, or sprint, from the time that the specificaiton was assigned to the iterative subproject to the end date. Using controls in the report manager, report authors may define the time intervals used in the report.

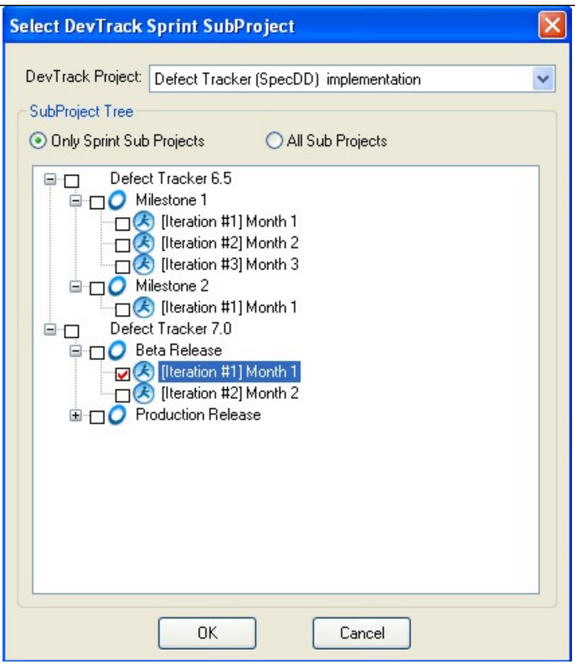
Daily, Weekly, Bi-Weekly, Monthly, Semi-Yearly, Yearly



Time Unit: Daily

Sub Projects:

8. Click the*Tree Settings*button to select a specific branch of the DevTrack subproject tree that will be used to fetch the report data.



Select DevTrack Sprint SubProject

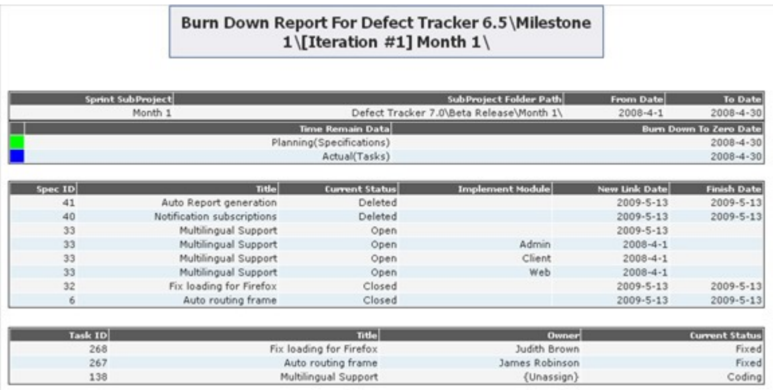
DevTrack Project: Defect Tracker (SpecDD) implementation

SubProject Tree

● Only Sprint Sub Projects ○ All Sub Projects

- □ Defect Tracker 6.5
  - □ Milestone 1
    - □ [Iteration #1] Month 1
    - □ [Iteration #2] Month 2
    - □ [Iteration #3] Month 3
  - □ Milestone 2
    - □ [Iteration #1] Month 1
- □ Defect Tracker 7.0
  - □ Beta Release
    - ☑ [Iteration #1] Month 1
    - □ [Iteration #2] Month 2
  - □ Production Release

9. Click*OK*button



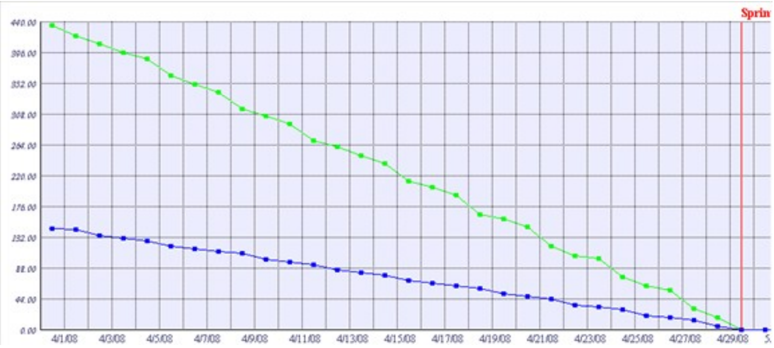
Burn Down Report For Defect Tracker 6.5\Milestone 1\[Iteration #1] Month 1\

Sprint SubProject	SubProject Folder Path	From Date	To Date
Month 1	Defect Tracker 7.0\Beta Release\Month 1\	2008-4-1	2008-4-30

	Time Remain Data	Burn Down To Zero Date
🟩	Planning(Specifications)	2008-4-30
🟦	Actual(Tasks)	2008-4-30

Spec ID	Title	Current Status	Implement Module	New Link Date	Finish Date
41	Auto Report generation	Deleted		2009-5-13	2009-5-13
40	Notification subscriptions	Deleted		2009-5-13	2009-5-13
33	Multilingual Support	Open		2009-5-13	
33	Multilingual Support	Open	Admin	2008-4-1	
33	Multilingual Support	Open	Client	2008-4-1	
33	Multilingual Support	Open	Web	2008-4-1	
32	Fix loading for Firefox	Closed		2009-5-13	2009-5-13
6	Auto routing frame	Closed		2009-5-13	2009-5-13

Task ID	Title	Owner	Current Status
268	Fix loading for Firefox	Judith Brown	Fixed
267	Auto routing frame	James Robinson	Fixed
138	Multilingual Support	{Unassign}	Coding



10. The button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

11. The button at the bottom of the report can be used to change the report properties.

12. The ··· button at the bottom of the report can be used to select a specific branch of the DevTrack subproject tree that will

be used to fetch the report data.

2.9 Feature Completion Report

The DevSpec feature completion reports show the implementation of specifications, within an iteration.

In the velocity chart the y-axis represents the backlog-the estimated number of hours required to implement a set of specification designs. The x-axis represents the development time-the time between the date the specification was initially linked to the development task until the end of the sprint.

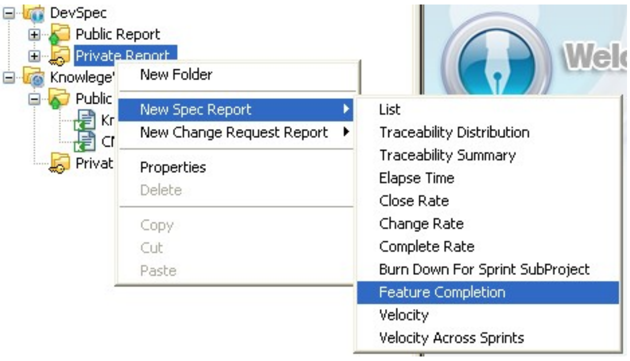
If multiple iterative subprojects are selected, distinct burndown charts, specification lists, and task lists are displayed for each subproject selected in the subproject tree.

Creating a feature completion report

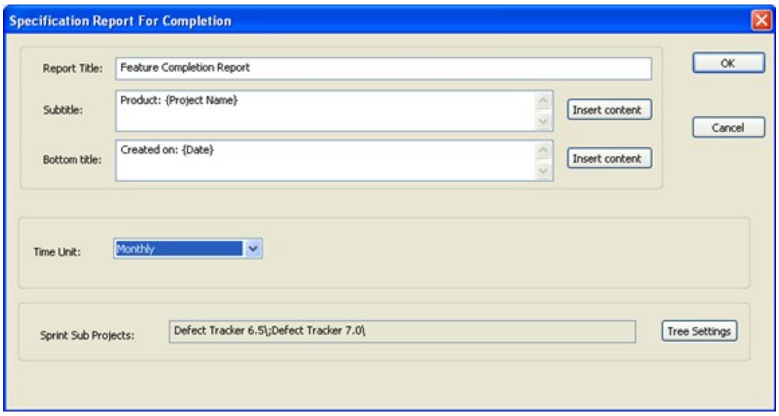
To create a feature completionreport:

1. Go to the report view by clicking the button in the tool bar.

2. In the tree panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report*>*Feature Completion*.



> ***Note:***Only users with the required privileges are able to create reports in the*Public Report*folder. Public reports are accessible to all other DevSpec users, while private reports are only available to the report author. For more information, please see section 1, at the beginning of this chapter.



Report Title: Feature Completion Report

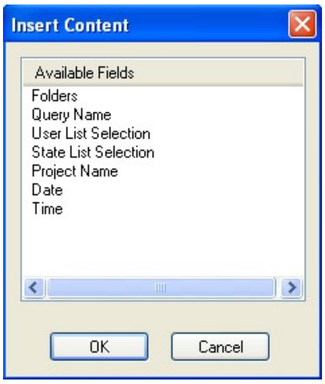
Subtitle: Product: {Project Name}

Bottom title: Created on: {Date}

Time Unit: Monthly

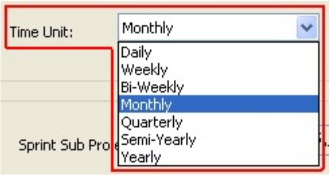
Sprint Sub Projects: Defect Tracker 6.5\;Defect Tracker 7.0\

3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.



Time Unit: The x-axis of the burndown chart represents the duration of the iterative subproject, or sprint, from the time the specification was assigned to the iterative subporject to the end date. Using controls in the report manager, report authors may define the time intervals used in the report.

Daily, Weekly, Bi-Weekly, Monthly, Semi-Yearly, Yearly


Time Unit: Monthly

4. The button at the bottom of the report can be used to change the report properties.

5. The button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

6. The ... button at the bottom of the report can be used to select a specific branch of the specification folder tree that will be used to fetch the report data.

2.10 Velocity Report

The term velocity is used in agile development to indicate the amount of work a team can handle in one iteration. The DevSpec velocity report shows the implementation of specifications, within an iteration.

In the velocity chart the y-axis represents the backlog-the estimated number of hours required to implement a set of specification designs. The x-axis represents development time-the time between the date that the specification was initially linked to the development task until the end of the sprint.

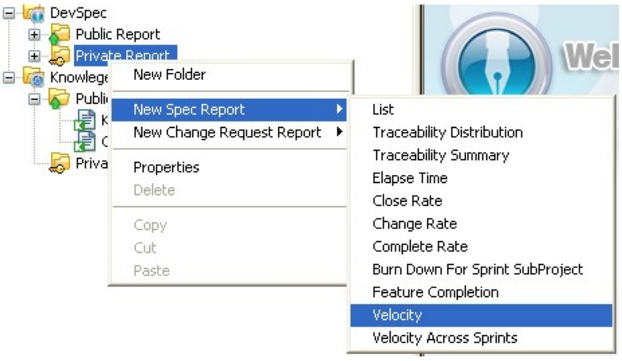
If multiple iterative subprojects are selected, distinct burndown charts, specification lists, and task lists are displayed for each subproject selected in the subproject tree.

Creating a velocity report

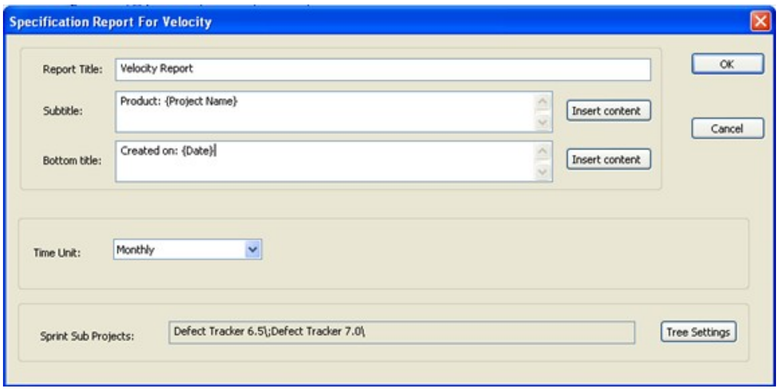
To create a velocityreport:

1. Go to the report view by clicking the button in the tool bar.

2. In the treepanel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report>Velocity*.



Note:Only users with the required privileges are able to create reports in the*Public Report*folder. Public reports are accessible to all DevSpec users, while private reports are only available to the report author. For more information, please see section 1, at the beginning of this chapter.



Report Title: Velocity Report

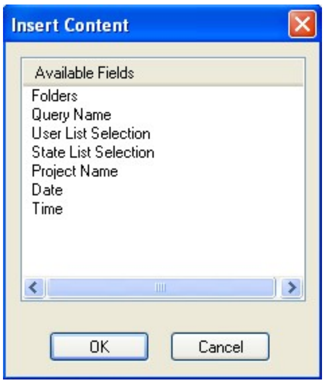
Subtitle: Product: {Project Name}

Bottom title: Created on: {Date}

Time Unit: Monthly

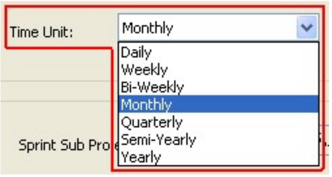
Sprint Sub Projects: Defect Tracker 6.5\;Defect Tracker 7.0\

3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.



Time Unit: The x-axis of the burndown chart represents the duration of the iterative subproject (sprint)-from the time the specification was assigned to the iterative subproject to the end date. Using controls in the report manager, report authors may define the time intervals used in the report.

Daily, Weekly, Bi-Weekly, Monthly, Semi-Yearly, Yearly



4. The button at the bottom of the report can be used to change the report properties.

5. The button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

6. The ··· button at the bottom of the report can be used to select a specific branch of the specification folder tree that will be used to fetch the report data.

2.11 Velocity Across Sprints Report

The term velocity is used in agile development to indicate the amount of work a team can handle in one iteration. The DevSpec velocity report across sprints report shows the implementation of specifications, within multiple iterations.

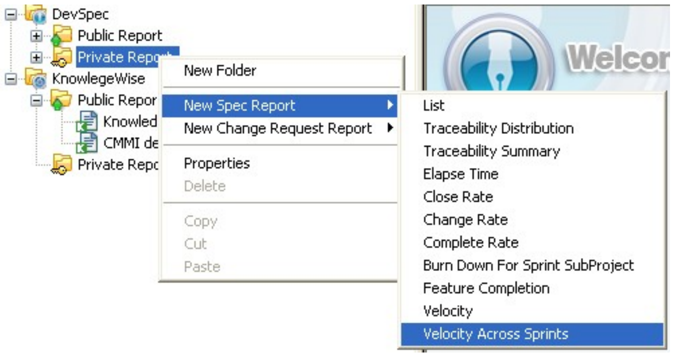
In a velocity across sprints report, the y-axis in the chart represents the backlog-the estimated number of hours required to implement a set of specification designs. The x-axis represents development time-the time between the date each specification was initially linked to a development task until the end of the sprints.

Creating a velocity across sprint report

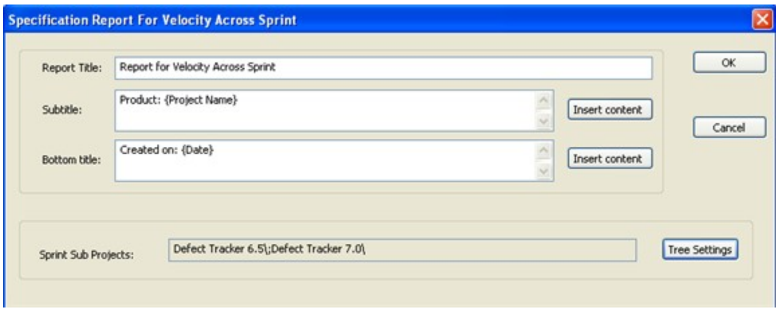
To create a velocity across sprintreport:

1. Go to the report view by clicking the button in the tool bar.

2. In the tree panel, under the DevSpec folder, right-click on the*Private Report*or*Public Report*folder, and select*New Spec Report>Velocity Across Sprints*.



Note: Only users with the required privileges are able to create reports in the *Public Report* folder. Public reports are accessible to all other DevSpec users, while private reports are only available to the report author. For more information, please see section 1, at the beginning of this chapter.



Specification Report For Velocity Across Sprint

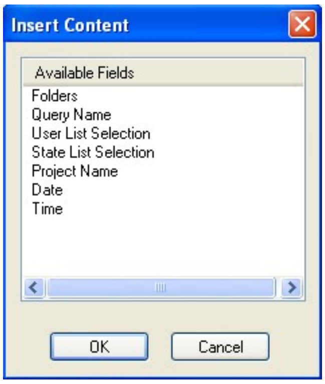
Report Title: Report for Velocity Across Sprint

Subtitle: Product: {Project Name}

Bottom title: Created on: {Date}

Sprint Sub Projects: Defect Tracker 6.5\;Defect Tracker 7.0\

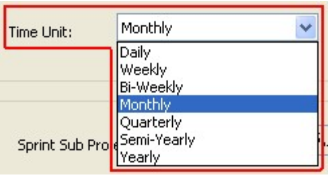
3. Define the report title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the *Insert content* button.



Insert Content

Time Unit: The x-axis of the burndown chart represents the duration of the iterative subproject (sprint)-from the time that the specification was assigned to the iterative subproject to the end date. Using controls in the report manager, report authors may define the time intervals used in the report.

Daily, Weekly, Bi-Weekly, Monthly, Semi-Yearly, Yearly



Time Unit: Monthly

4. The button at the bottom of the report can be used to change the report properties.

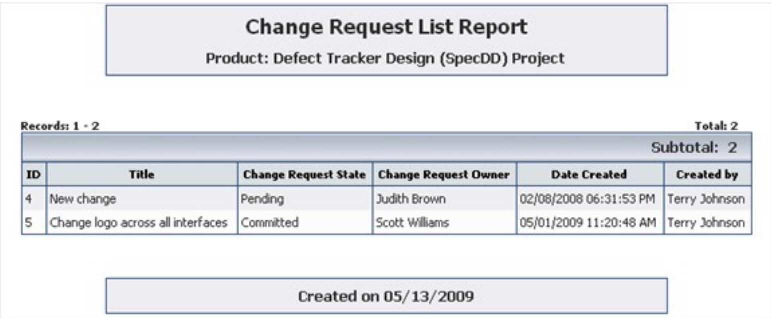
5. The button at the bottom of the report facilitates exporting to an Excel, Word or CSV format.

6. The ··· button at the bottom of the report can be used to select a specific branch of the specification folder tree that will be

used to fetch the report data.

2.12 Change Request List Report

The change request list report displays information tracked under the change request view in DevSpec. Any field or attribute used to manage a change request can be included in thisreport.



Change Request List Report

Product: Defect Tracker Design (SpecDD) Project

Records: 1 - 2

Total: 2

Subtotal: 2

ID	Title	Change Request State	Change Request Owner	Date Created	Created by
4	New change	Pending	Judith Brown	02/08/2008 06:31:53 PM	Terry Johnson
5	Change logo across all interfaces	Committed	Scott Williams	05/01/2009 11:20:48 AM	Terry Johnson

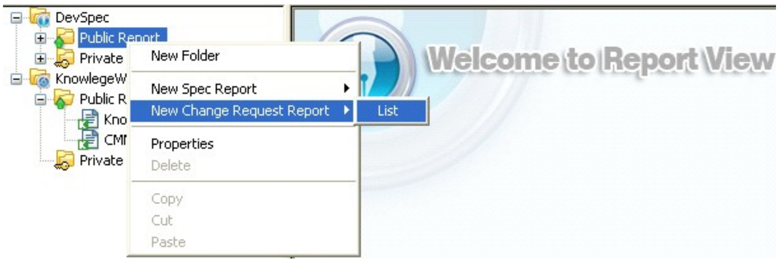
Created on 05/13/2009

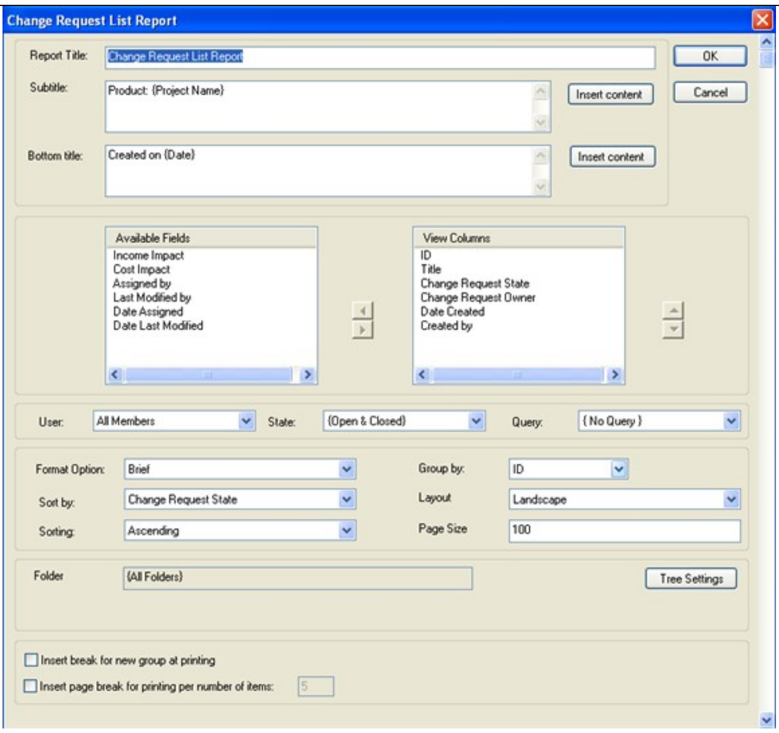
Creating a change request list report

To create a change request list report:

1. Go to the report view by clicking the button in the tool bar.

2. In the tree panel, under the DevSpec folder, right-click the on*Private Report*or*Public Report*folder, and select*NewChange Request Report>List*.





Change Request List Report

Report Title: Change Request List Report

Subtitle: Product: (Project Name)

Bottom title: Created on (Date)

Available Fields	View Columns
Income Impact	ID
Cost Impact	Title
Assigned by	Change Request State
Last Modified by	Change Request Owner
Date Assigned	Date Created
Date Last Modified	Created by

User: All Members State: (Open & Closed) Query: (No Query)

Format Option: Brief Group by: ID

Sort by: Change Request State Layout: Landscape

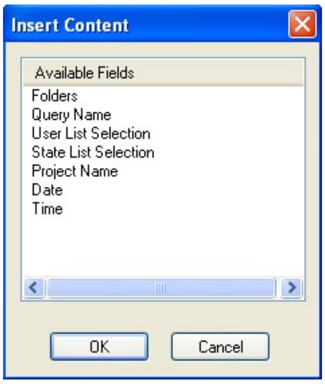
Sorting: Ascending Page Size: 100

Folder (All Folders)

☐ Insert break for new group at printing

☐ Insert page break for printing per number of items: 5

3. Define the project title, subtitle, and bottom title. Users can also insert certain values from the database, by clicking the*Insert content*button.



Insert Content

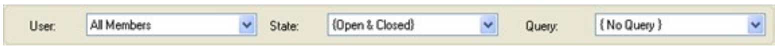
Available Fields

- Folders
- Query Name
- User List Selection
- State List Selection
- Project Name
- Date
- Time

4. Users can click the◀▶ buttons to add or remove fields from the*Available Fields*list to the*View Columns*list, and vice versa.

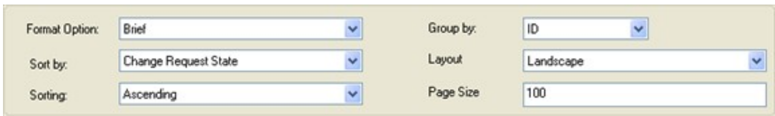
Users can also set the field order by using the▲▼ buttons.

5. The*User, State*and*Query*dropdown lists allow users to filter the data fetched in the report.



User: All Members State: (Open & Closed) Query: (No Query)

6. Users can also define the report to be brief or detailed under the*Format Option*control. A detailed report allows inclusion of more fields than the brief report does.

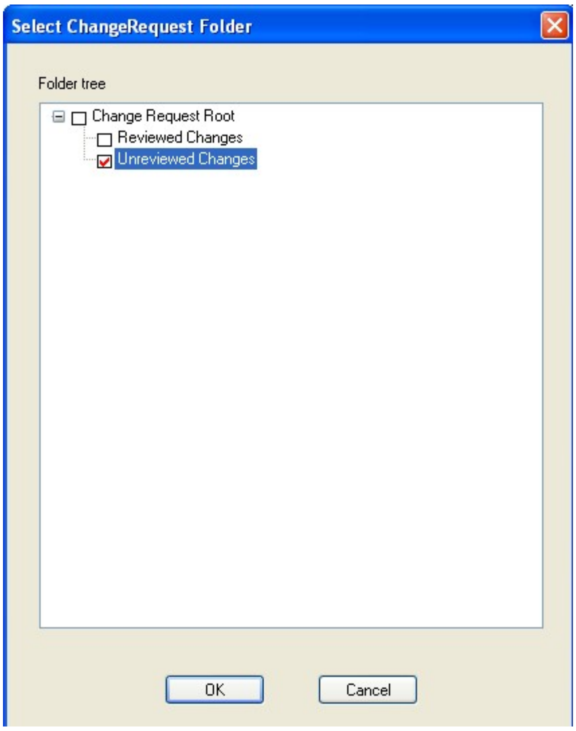


Format Option: Brief
Sort by: Change Request State
Sorting: Ascending
Group by: ID
Layout: Landscape
Page Size: 100

7. Users can further customize the report by configuring the*Group By, Sort By, Layout*and*Sort*options.

8. The*Page Size*value allows users to restrict the number of records displayed per page.

9. Click the*Tree Settings*button to select a specific branch of the change request folder tree that will be used to fetch the report data.

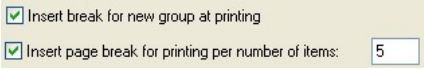


Select ChangeRequest Folder

Folder tree

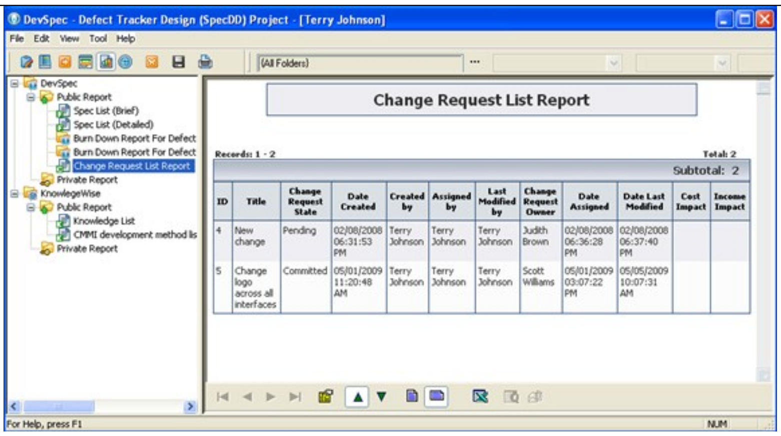
- □ Change Request Root
  - □ Reviewed Changes
  - ☑ Unreviewed Changes

10. Users can choose to insert a page break after every grouping of records when printing the report. Page breaks can also be inserted based on the number of records. This can be configured by selecting the check boxes, as shown below:



☑ Insert break for new group at printing
☑ Insert page break for printing per number of items: 5

11. Click the*OK*button.



Change Request List Report

Records: 1 - 2 Total: 2

Subtotal: 2

ID	Title	Change Request State	Date Created	Created by	Assigned by	Last Modified by	Change Request Owner	Date Assigned	Date Last Modified	Cost Impact	Income Impact
4	New change	Pending	02/08/2008 06:31:53 PM	Terry Johnson	Terry Johnson	Terry Johnson	Judith Brown	02/08/2008 06:36:28 PM	02/08/2008 06:37:40 PM		
5	Change logo across all interfaces	Committed	05/01/2009 11:20:48 AM	Terry Johnson	Terry Johnson	Terry Johnson	Scott Williams	05/01/2009 03:07:22 PM	05/05/2009 10:07:31 AM		

Tip: Once a report is created, users can use the filter controls in the report bar to dynamically customize the report. For more information on using these filter controls in the report bar, please see section 1,*Report Basics*, earlier in this chapter.

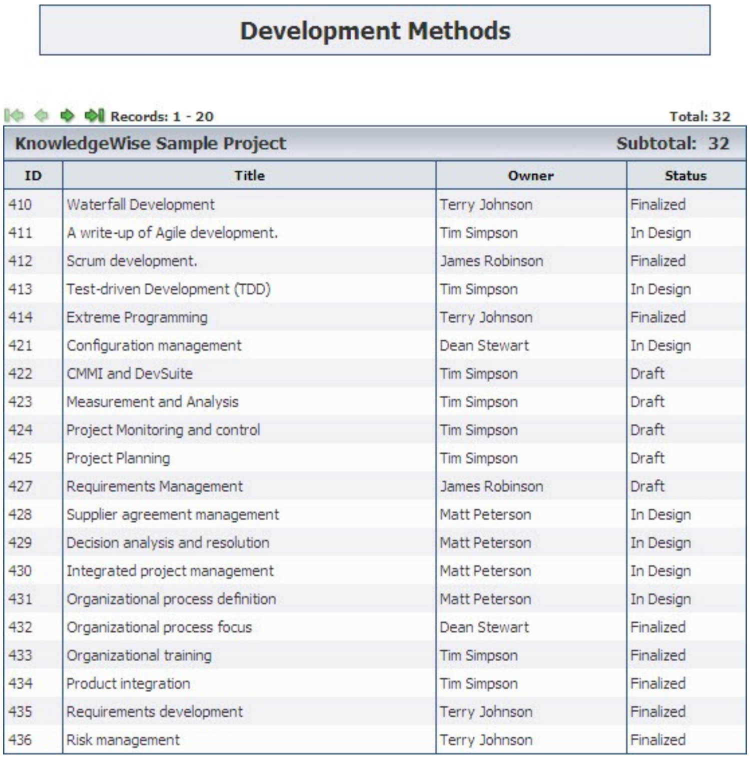
3 KnowledgeWise Reports

KnowledgeWise reports can be accessed in the DevSpec report view.We currently support one report style for Knowledge items--the knowledge list report.

3.1 Knowledge List Reports

The knowledge list report displays information tracked under the knowledge view in DevSpec. Any field or attribute related to a knowledge item can be included in the report.

Below is an example of a knowledge list report:



Development Methods

Records: 1 - 20 Total: 32

KnowledgeWise Sample Project Subtotal: 32

ID	Title	Owner	Status
410	Waterfall Development	Terry Johnson	Finalized
411	A write-up of Agile development.	Tim Simpson	In Design
412	Scrum development.	James Robinson	Finalized
413	Test-driven Development (TDD)	Tim Simpson	In Design
414	Extreme Programming	Terry Johnson	Finalized
421	Configuration management	Dean Stewart	In Design
422	CMMI and DevSuite	Tim Simpson	Draft
423	Measurement and Analysis	Tim Simpson	Draft
424	Project Monitoring and control	Tim Simpson	Draft
425	Project Planning	Tim Simpson	Draft
427	Requirements Management	James Robinson	Draft
428	Supplier agreement management	Matt Peterson	In Design
429	Decision analysis and resolution	Matt Peterson	In Design
430	Integrated project management	Matt Peterson	In Design
431	Organizational process definition	Matt Peterson	In Design
432	Organizational process focus	Dean Stewart	Finalized
433	Organizational training	Tim Simpson	Finalized
434	Product integration	Tim Simpson	Finalized
435	Requirements development	Terry Johnson	Finalized
436	Risk management	Terry Johnson	Finalized

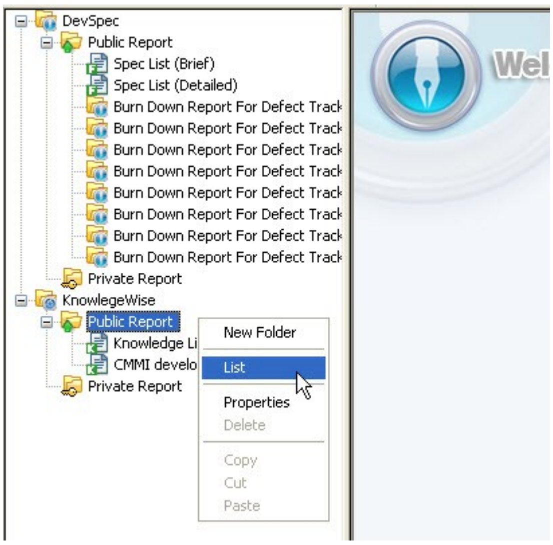
Creating a knowledge list report

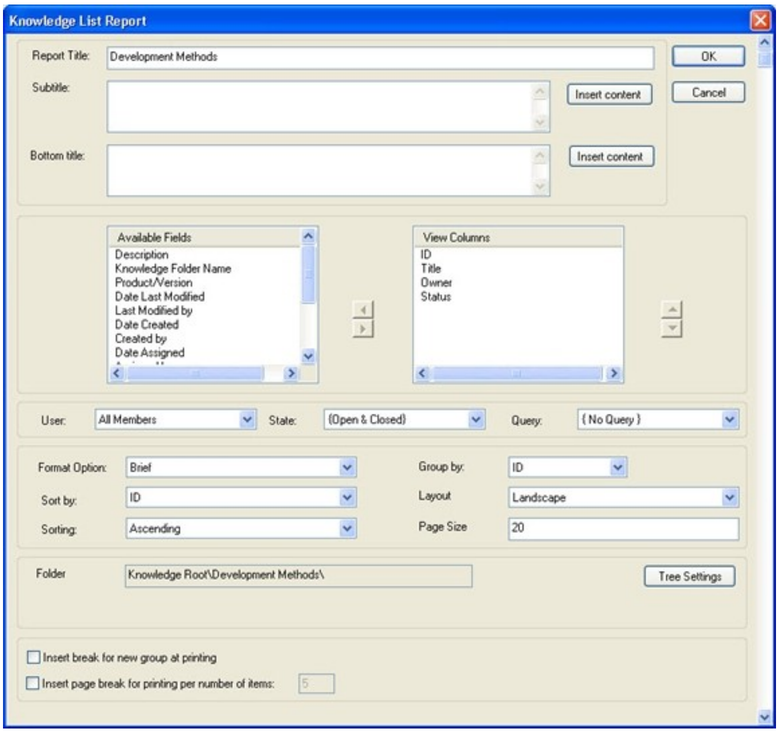
To create a knowledge list report:

1. Go to the*Report*view by clicking the button on the tool bar.

2. In the*Tree*panel, under the KnowledgeWise folder, right-click on the*Private Report*or*Public Report*folder (or any subfolder

underneath them), and select*List*. The*Knowledge List Report*dialog appears.


DevSpec
Public Report
Spec List (Brief)
Spec List (Detailed)
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Burn Down Report For Defect Track
Private Report
KnowlegeWise
Public Report
Knowledge Li
CMMI develo
Private Report
New Folder
List
Properties
Delete
Copy
Cut
Paste
Wel



Report Title: Development Methods

Subtitle:

Bottom title:

Available Fields: Description, Knowledge Folder Name, Product/Version, Date Last Modified, Last Modified by, Date Created, Created by, Date Assigned

View Columns: ID, Title, Owner, Status

User: All Members State: (Open & Closed) Query: (No Query)

Format Option: Brief Group by: ID

Sort by: ID Layout: Landscape

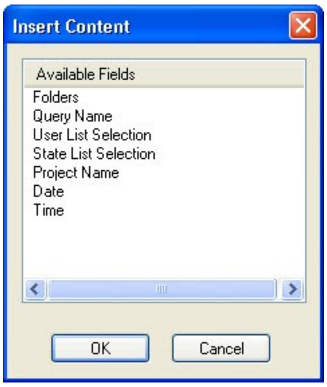
Sorting: Ascending Page Size: 20

Folder: Knowledge Root\Development Methods\

☐ Insert break for new group at printing

☐ Insert page break for printing per number of items: 5

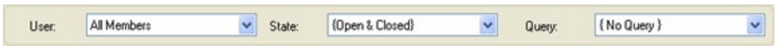
3. In the*Knowledge List Report*dialog, define the report title, subtitle, and bottom title. Users can also insert pre-defined system fields by clicking the*Insert content*button. Highlight the fields that are to be added to the subtitle or bottom title, and click the*OK*button.



4. Define the report columns. Users can click the ◀▶ buttons to add or remove fields between the*Available Fields*and*View*

*Columns*lists. Users can also define the column order by using the ▲▼ buttons.

5. The*User, State*,and*Query*dropdown lists allow the user to filter the report data. These three filters can also be found in the tool bar in the report view.

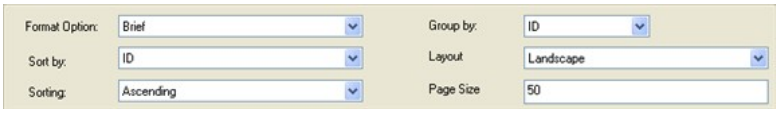


User: All Members State: (Open & Closed) Query: (No Query)

6. Users can also define the report as brief or detailed by using the*Format Option*dropdown list. More columns can be included in a detailed report than a brief one.

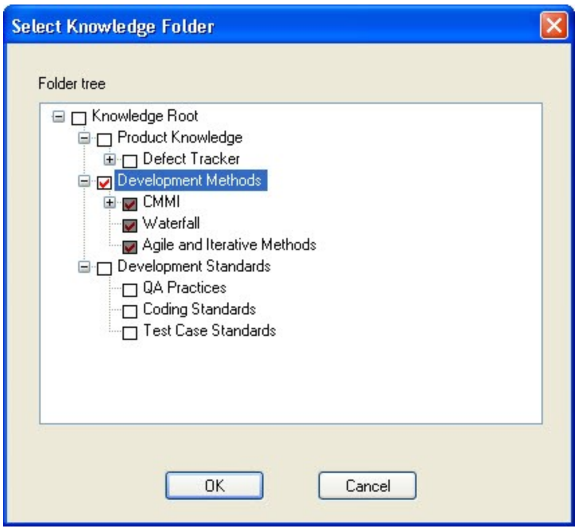
7. Define how the knowledge list is displayed. Users can customize the report by configuring the*Group By*,*Sort By*,*Layout*, and*Sorting*options.

8. The*Page Size*value allows users to determine the number of records displayed per page.



Format Option: Brief
Group by: ID
Sort by: ID
Layout: Landscape
Sorting: Ascending
Page Size: 50

9. Define the report scope. Click the*Tree Settings*button to select a specific branch of the knowledge folder tree. When a parent folder is selected, all the descendent child folders will be selected automatically.

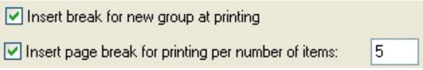


Select Knowledge Folder

Folder tree

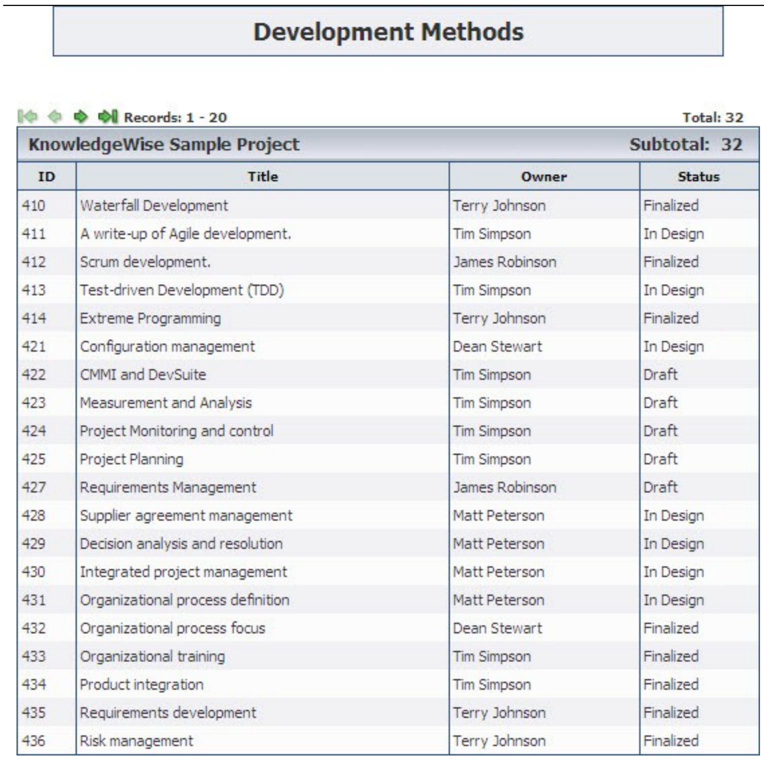
- □ Knowledge Root
  - □ Product Knowledge
    - □ Defect Tracker
  - ☑ Development Methods
    - ☑ CMMI
    - ☑ Waterfall
    - ☑ Agile and Iterative Methods
  - □ Development Standards
    - □ QA Practices
    - □ Coding Standards
    - □ Test Case Standards

10. Users can choose to insert a page break after every grouping of records when printing the report. Page breaks can also be inserted based on the number of records. This can be configured by selecting the checkboxes, as shown below:



☑ Insert break for new group at printing
☑ Insert page break for printing per number of items: 5

11. Click the*OK*button.



Development Methods

Records: 1 - 20 Total: 32

KnowledgeWise Sample Project Subtotal: 32

ID	Title	Owner	Status
410	Waterfall Development	Terry Johnson	Finalized
411	A write-up of Agile development.	Tim Simpson	In Design
412	Scrum development.	James Robinson	Finalized
413	Test-driven Development (TDD)	Tim Simpson	In Design
414	Extreme Programming	Terry Johnson	Finalized
421	Configuration management	Dean Stewart	In Design
422	CMMI and DevSuite	Tim Simpson	Draft
423	Measurement and Analysis	Tim Simpson	Draft
424	Project Monitoring and control	Tim Simpson	Draft
425	Project Planning	Tim Simpson	Draft
427	Requirements Management	James Robinson	Draft
428	Supplier agreement management	Matt Peterson	In Design
429	Decision analysis and resolution	Matt Peterson	In Design
430	Integrated project management	Matt Peterson	In Design
431	Organizational process definition	Matt Peterson	In Design
432	Organizational process focus	Dean Stewart	Finalized
433	Organizational training	Tim Simpson	Finalized
434	Product integration	Tim Simpson	Finalized
435	Requirements development	Terry Johnson	Finalized
436	Risk management	Terry Johnson	Finalized

In the screenshot above, a knowledge list report, called*Development Methods*, is created. The list is sorted by knowledge item ID, and is displayed in an ascending order. This report allows KnowledgeWise project members to view all knowledge items in the*Development Methods*folder. Project members are also able to tell from this report the knowledge items’ current owner and status.

Tip: Once a report is created, users can use the filter controls in the report bar to dynamically customize the report. For more information on using these filter controls in the report bar, please see section 1,*Report Basics*, earlier in this chapter.

Chapter 10 - Knowledge Management

In DevSpec, and throughout all DevSuite components, the core of any product development initiative is knowledge management: from ideas, to formal specifications, to competitive information, to issue resolution and customer insight. The DevSuite knowledge-centric strategy enables improved communication, ensures users are up-to-date on changes, and reduces the development cycles, so that businesses may deliver the right products for the right markets in the shortest possible time.

1 Knowledge Management in DevSpec

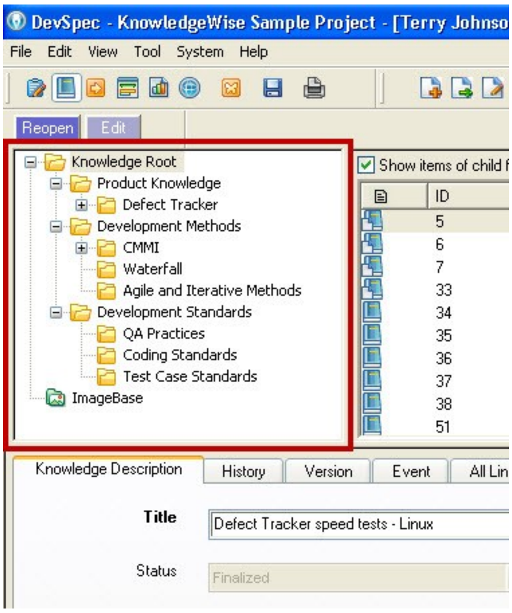
In the knowledge view users can organize knowledge gathered from a wide variety of sources. Knowledge may include enhancement requests from customers, feature requests from product marketing, design improvements from the development team, or even new product ideas from any employee within the company. This collection of knowledge/ideas is the genesis of the final delivered product.

In DevSpec, knowledge items may be entered in different forms: e.g. documents, images, HTML links, and other digital assets. Some knowledge items will be discarded, many will be consolidated and improved, and others will be accepted as is. Knowledge items are completely tracked through workflow.

Knowledge is not unique just to the current DevSpec project. Multiple DevSpec projects can be managed under a parent KnowledgeWise project. Thus, users may see the same knowledge folders and items when viewing different projects.

2 Knowledge Folder Tree

In DevSpec, all knowledge items are well organized via the knowledge folder tree. Users can create knowledge categories and sub-categories in the knowledge folder tree panel (top left).

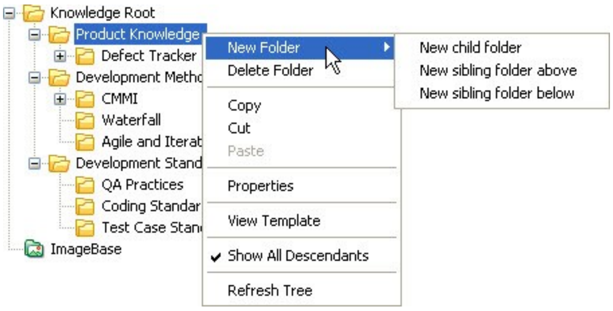


Title: Defect Tracker speed tests - Linux

Status: Finalized

Creating a New Folder

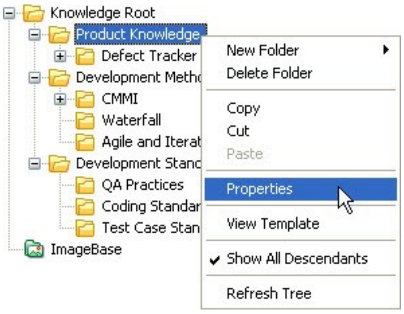
To create a new folder in the knowledge folder tree, right-click on an existing folder, where the new folder is to be created, and select*New Folder*.



Select*New child folder*to create a subfolder underneath the current folder.
Select*New sibling folder above*to create a folder in the same level but above the current folder.
Select*New sibling folder below*to create a folder in the same level but below the current folder.
* Please see the*Folder Properties*section for more information on defining folder properties.

Editing a Folder

To edit the properties of an existing folder in the knowledge folder tree, right-click on an existing folder, and select*Properties*.



* Please see the*Folder Properties*section for more information on defining folder properties.

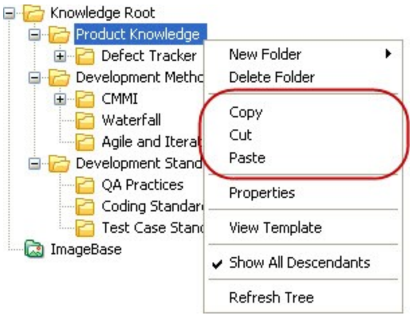
Copying/Moving Folders

Users can create an identical set of folders and its knowledge items to another folder from the current selected folder.

1. Right-click on a folder and select*Copy*.
2. Right-click on a destination folder and select*Paste*.

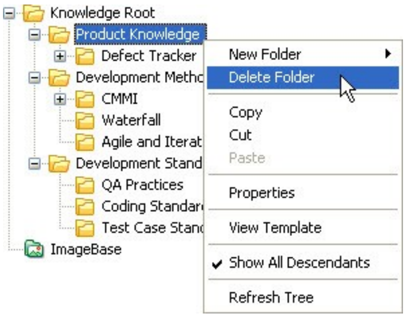
Users can also choose to move a folder and its contents to a different directory.

1. Right-click on a folder and select*Cut*.
2. Right-click on a destination folder and select*Paste*.



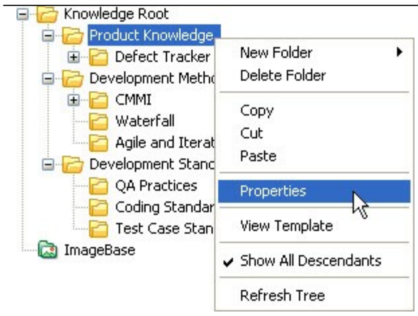
Deleting a Folder

For a folder to be deleted, it must be empty (i.e. not containing any knowledge items), and must not be currently used by a project. To delete a folder in the knowledge folder tree, right-click on an existing folder, and select*Delete*.



2.1 Knowledge Folder Properties

To access folder properties, right-click on the folder and select*Properties*.

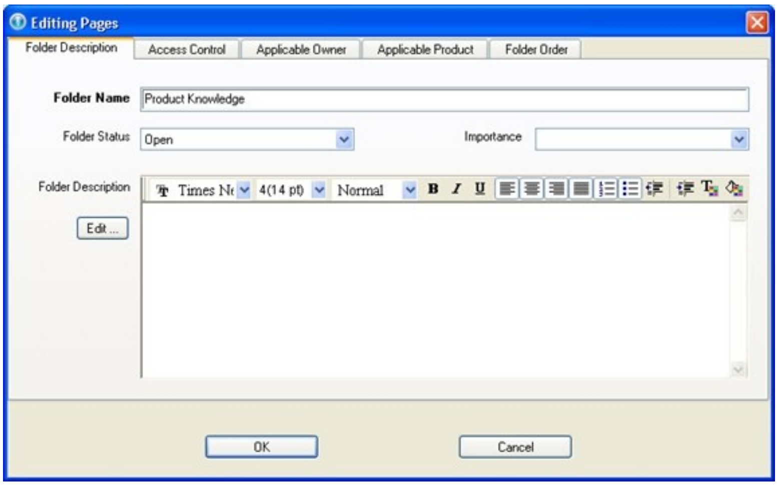


Folder Description

This section allows users to accurately describe the details of a folder.

1. Provide an accurate folder name so that work items can be easily filtered.
2. Provide a folder status to indicate whether or not this folder should still be in use.
3. Provide an importance value.
4. Provide a description.

*Administrators can add custom fields and pages to track additional details.



Folder Name: Product Knowledge

Folder Status: Open

Importance:

Folder Description:

Access Control

This section allows users with sufficient privileges to secure the contents within a knowledge folder and the folder itself.

Users can select 1 out of 3 different folder access types. All folder access types are split into two panes. This allows the manager to view the permissions defined by the administrator for folders and knowledge items within the folders.

Public Folder:
    A set of account types defined in the Admin.
No Access: users will not be able to see existing folder/knowledge items

Read-Only: users cannot update existing folder/ knowledge items

Can Edit: users can only update existing folder/ knowledge items

Can Create and Edit: users can submit new folder/ knowledge items

Can Delete, Create and Edit: users can submit new folder/ knowledge items as well as delete existing items
Private Folder:
    A second set of privileges defined in the Admin.
No Access: users will not be able to see existing folder/ knowledge items

Read-Only : users cannot update existing folder/ knowledge items

Can Edit : users can only update existing folder/ knowledge items

Can Create and Edit : users can submit new folder/ knowledge items

Can Delete, Create and Edit: users can submit new folder/ knowledge items as well as delete existing items

Secured Folder:

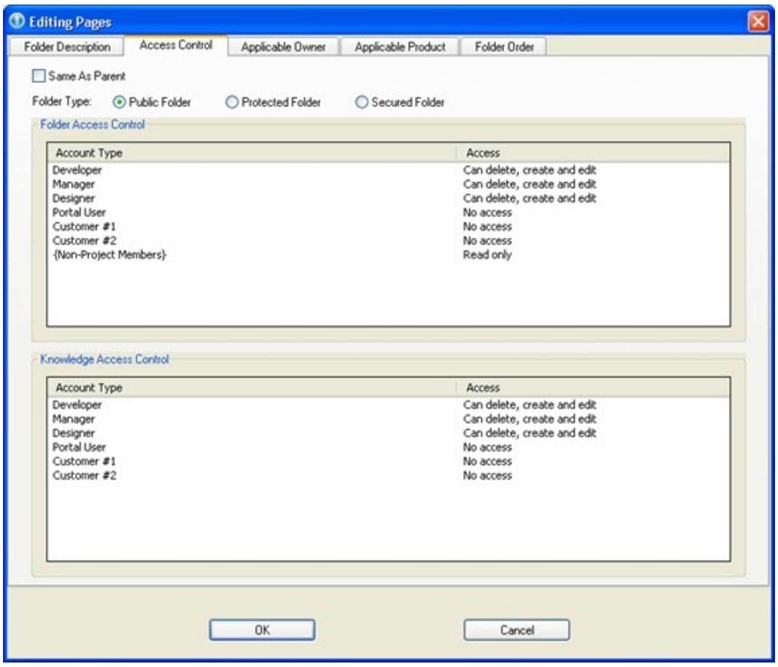
This folder access type is used if the public/private folder access types are not sufficient. Administrators can define different sets of custom access levels for account types and team groups that can be applied to any folder. This is beneficial if privileges may need to be changed later.

To view privileges for each access type, click*view access type*.

In addition, individual users can also be added as an exception to the account type and team group privileges defined in the access level:

1. To add a user, click*Add User*, select the user(s), and give applicable privileges.
2. To remove user(s) from the access level, click*Remove User*.

*Check off*same as parent*to inherit the access control from the parent folder



□ Same As Parent

Folder Type: ● Public Folder ○ Protected Folder ○ Secured Folder

Folder Access Control

Account Type	Access
Developer	Can delete, create and edit
Manager	Can delete, create and edit
Designer	Can delete, create and edit
Portal User	No access
Customer #1	No access
Customer #2	No access
{Non-Project Members}	Read only

Knowledge Access Control

Account Type	Access
Developer	Can delete, create and edit
Manager	Can delete, create and edit
Designer	Can delete, create and edit
Portal User	No access
Customer #1	No access
Customer #2	No access

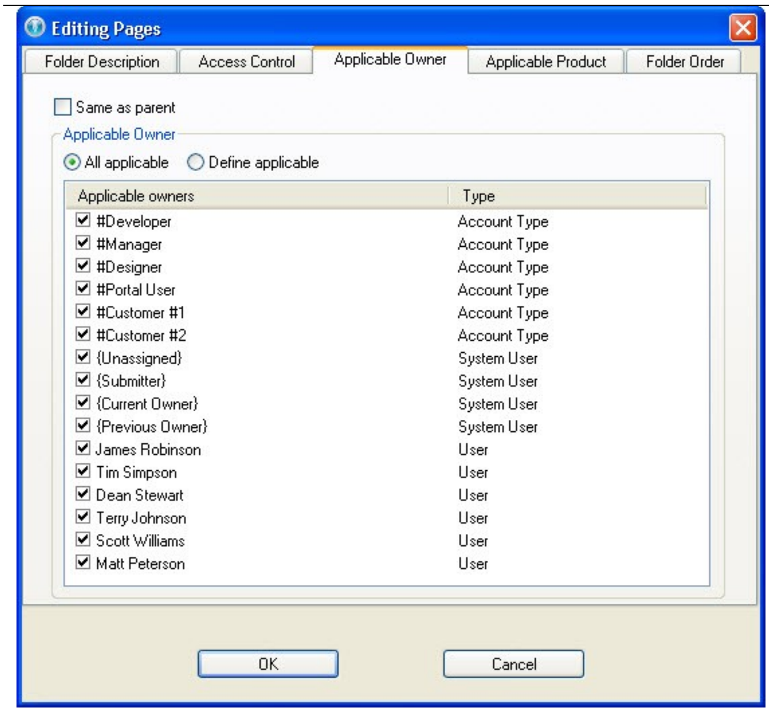
Applicable Owner

This section allows managers to define account types, groups and individuals that can own a knowledge item in the folder. Users that do not belong in the account types or groups defined here cannot be selected as a knowledge item owner, even though the workflow permits them.

Select*All Applicable*to quickly allow all DevSpec users to be able to own knowledge items if the workflow permits them.

Select*Define Applicable*to define specific account types, groups and users to be able to own knowledge items if the workflow permits them.

*Check off*same as parent*to inherit the access control from the parent folder



□ Same as parent

Applicable Owner

● All applicable ○ Define applicable

Applicable owners	Type
☑ #Developer	Account Type
☑ #Manager	Account Type
☑ #Designer	Account Type
☑ #Portal User	Account Type
☑ #Customer #1	Account Type
☑ #Customer #2	Account Type
☑ {Unassigned}	System User
☑ {Submitter}	System User
☑ {Current Owner}	System User
☑ {Previous Owner}	System User
☑ James Robinson	User
☑ Tim Simpson	User
☑ Dean Stewart	User
☑ Terry Johnson	User
☑ Scott Williams	User
☑ Matt Peterson	User

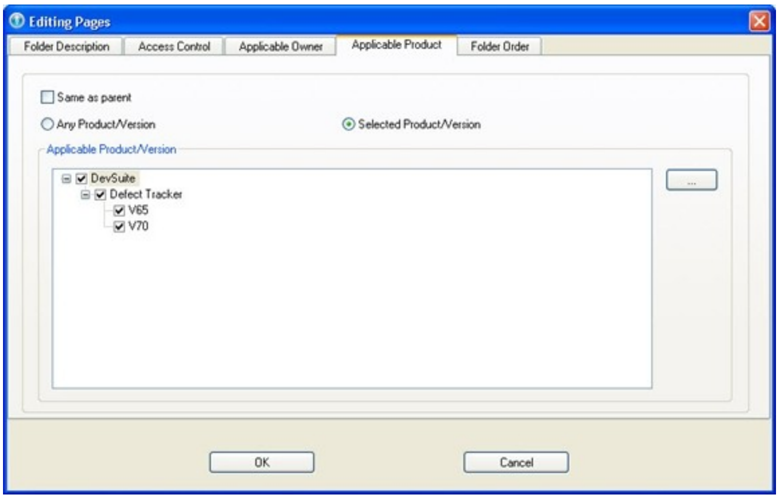
Applicable Product

Knowledge items can be defined by the product/version property-a subset of products, versions, and builds applicable to a knowledge item. In this section, users can set the default value of the product/version property for any item created in the current knowledge item folder. Setting this folder property on an existing folder will not change the product/version property values of any existing knowledge items.

Select*Any Product/Version*to quickly select all products, versions, and builds as the default value of the product/version property for new knowledge items created in the current folder.

Select*Define Applicable*to define specific products, versions, and builds as the default value of the product/version property for new knowledge items created in the current folder.

* Check off*same as parent*to inherit the access control from the parent folder



□ Same as parent

○ Any Product/Version ● Selected Product/Version

Applicable Product/Version

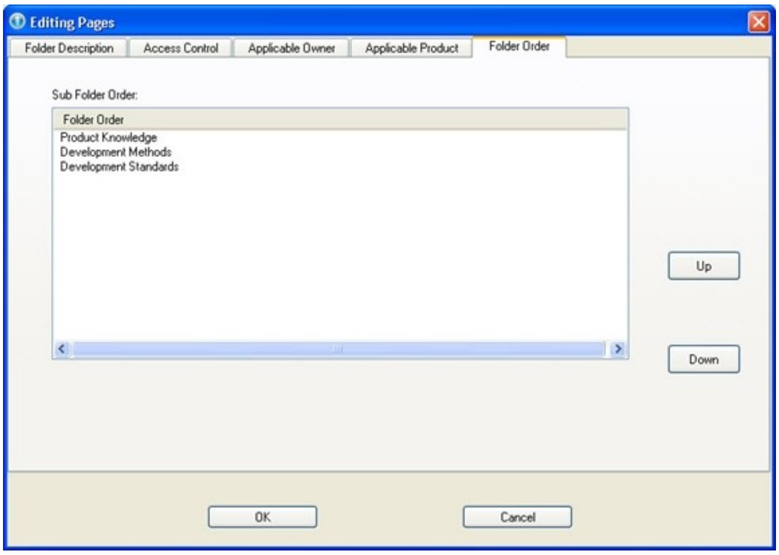
- ☑ DevSuite
  - ☑ Defect Tracker
    - ☑ V65
    - ☑ V70

Folder Order

This section allows managers to be able to sort the subfolders underneath the selected folder.

Click the up button to move a subfolder higher in the tree.

Click the down button to move a subfolder lower in the tree.



Sub Folder Order:

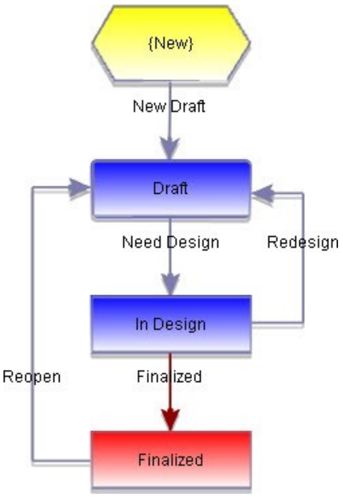
Folder Order
Product Knowledge
Development Methods
Development Standards

3 Knowledge Items

In DevSpec, just as with specifications, knowledge items are completely tracked through a workflow, which defines how knowledge items are created, managed, and tracked.

An administrator-defined workflow determines the sequence of workflow states-how and when a knowledge item may pass from one workflow state to the next. Each state is also privileged controlled-who may submit, forward, edit, or delete a knowledge item at each stage of its lifecycle.

The follow diagram is a possible workflow. A knowledge item workflow is typically simpler than a specification workflow. After a new knowledge item is created, it will start in the*Draft*state. Then it will move on to the*In Design*state (via the*Need Design*transition), and finally to the*Finalized*state (via the*Finalized*transition).

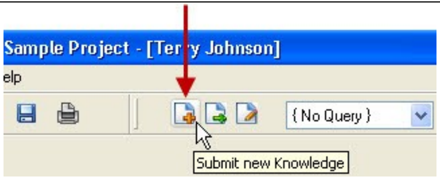


3.1 Submitting a New Knowledge Item

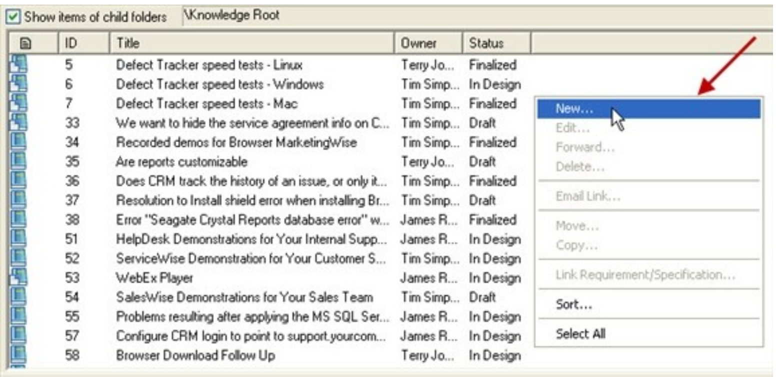
To submit a new knowledge item:

1. Open the submission dialog by one of the following commands:

Click on the*Submit new Knowledge*button in the tool bar

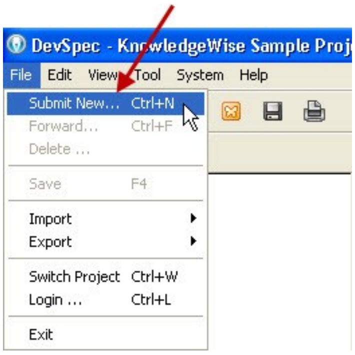


Right-click in the list panel and select*New...*


☑ Show items of child folders \Knowledge Root

	ID	Title	Owner	Status
	5	Defect Tracker speed tests - Linux	Terry Jo...	Finalized
	6	Defect Tracker speed tests - Windows	Tim Simp...	In Design
	7	Defect Tracker speed tests - Mac	Tim Simp...	Finalized
	33	We want to hide the service agreement info on C...	Tim Simp...	Draft
	34	Recorded demos for Browser MarketingWise	Tim Simp...	Finalized
	35	Are reports customizable	Terry Jo...	Draft
	36	Does CRM track the history of an issue, or only it...	Tim Simp...	Finalized
	37	Resolution to Install shield error when installing Br...	Tim Simp...	Draft
	38	Error "Seagate Crystal Reports database error" w...	James R...	Finalized
	51	HelpDesk Demonstrations for Your Internal Supp...	James R...	In Design
	52	ServiceWise Demonstration for Your Customer S...	Tim Simp...	In Design
	53	WebEx Player	James R...	In Design
	54	SalesWise Demonstrations for Your Sales Team	Tim Simp...	Draft
	55	Problems resulting after applying the MS SQL Ser...	James R...	In Design
	57	Configure CRM login to point to support.yourcom...	James R...	In Design
	58	Browser Download Follow Up	Terry Jo...	In Design

Select*File>Submit New...*in the menu bar



Press Ctrl + N
* To auto-populate the value of the folder field in the submission dialog, click on a folder in the knowledge item folder tree prior to performing any of the above actions.

2. Optional: Select a knowledge item template. For more information, please see the section,*Creating a Template*, later in this section.

3. In the submission dialog, define the knowledge item properties. For more detailed information on knowledge item properties, please see the section,*Knowledge Properties*, later in this chapter.

4.To close the dialog upon submission, check the*Close submission dialog after a Knowledge is submitted*checkbox. To keep the dialog open to submit another new knowledge item, leave the box unchecked.

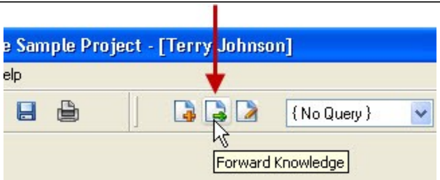
5. Click the*OK*button.

3.2 Forwarding a Knowledge Item

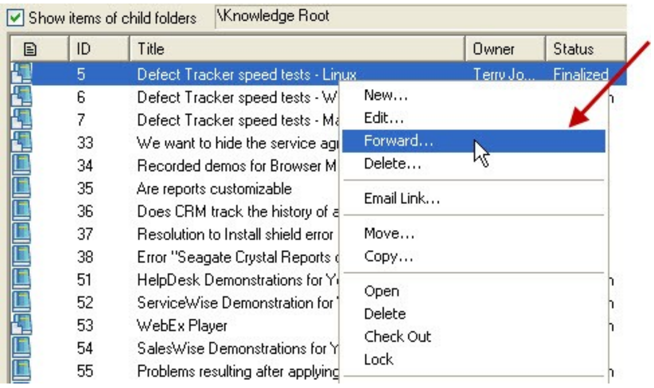
Forwarding is primarily done to change the owner of a knowledge item, but can also be used to change the state of the knowledge item, as well. To forward a knowledge item:

1. Open the forward dialog by one of the follow commands:

Highlight a knowledge item in the list panel and press the*Forward Knowledge*button in the tool bar

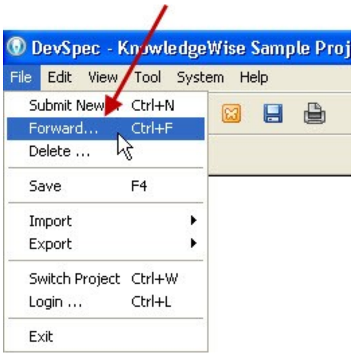


Right-click a knowledge item in the list panel and click*Forward...*

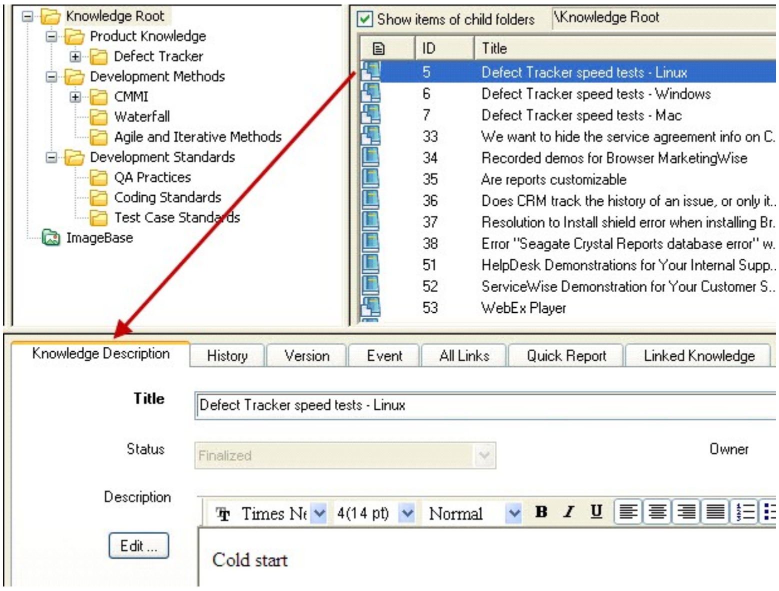


ID	Title	Owner	Status
5	Defect Tracker speed tests - Linux	Terry Jo...	Finalized
6	Defect Tracker speed tests - W		
7	Defect Tracker speed tests - Ma		
33	We want to hide the service agr		
34	Recorded demos for Browser M		
35	Are reports customizable		
36	Does CRM track the history of a		
37	Resolution to Install shield error		
38	Error "Seagate Crystal Reports c		
51	HelpDesk Demonstrations for Yo		
52	ServiceWise Demonstration for Y		
53	WebEx Player		
54	SalesWise Demonstrations for Y		
55	Problems resulting after applying		

Highlight a knowledge item in the list panel and from the menu bar, select*File>Forward...*



Highlight a knowledge item in the list panel and press Ctrl + F
Highlight a knowledge item in the list panel and go to the*Knowledge Description*tab in the detail panel (no window opens, but rather the knowledge item is forwarded directly from the detail panel)



ID	Title
5	Defect Tracker speed tests - Linux
6	Defect Tracker speed tests - Windows
7	Defect Tracker speed tests - Mac
33	We want to hide the service agreement info on C.
34	Recorded demos for Browser MarketingWise
35	Are reports customizable
36	Does CRM track the history of an issue, or only it..
37	Resolution to Install shield error when installing Br.
38	Error "Seagate Crystal Reports database error" w.
51	HelpDesk Demonstrations for Your Internal Supp..
52	ServiceWise Demonstration for Your Customer S..
53	WebEx Player

Title: Defect Tracker speed tests - Linux
Status: Finalized
Owner:
Description: Cold start

2. In the forward dialog, define the new knowledge item owner, to whom the knowledge item will be forwarded.

3. Make any other necessary changes to the knowledge item properties. For more detailed information on knowledge item properties, please see the section,*Knowledge Properties*, later in this chapter.

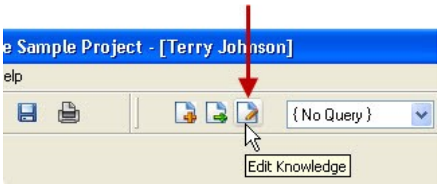
4. Click the*OK*button to forward the knowledge item to the next owner and/or next state.

3.3 Editing a Knowledge Item

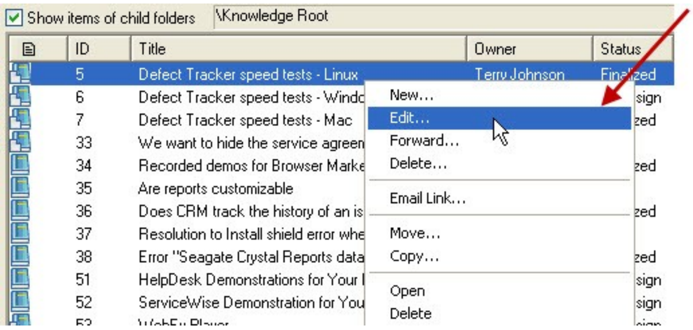
To edit a knowledge item:

1. Open the edit dialog by one of the following commands:

Highlight a knowledge item in the list panel and press the*Edit Knowledge*button in the tool bar

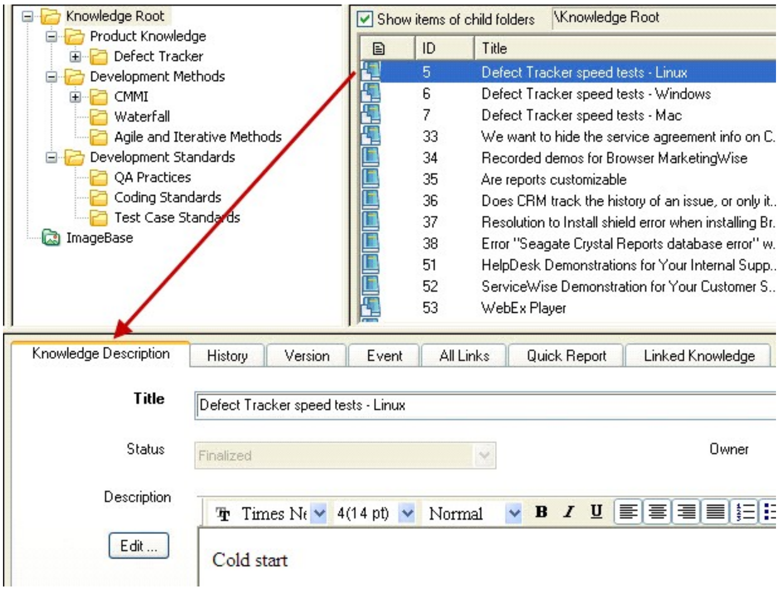


Right-click a knowledge item in the list panel and click*Edit...*


☑ Show items of child folders \Knowledge Root

	ID	Title	Owner	Status
	5	Defect Tracker speed tests - Linux	Terry Johnson	Finalized
	6	Defect Tracker speed tests - Windc		sign
	7	Defect Tracker speed tests - Mac		zed
	33	We want to hide the service agreen		
	34	Recorded demos for Browser Marke		zed
	35	Are reports customizable		
	36	Does CRM track the history of an is		zed
	37	Resolution to Install shield error whe		
	38	Error "Seagate Crystal Reports data		zed
	51	HelpDesk Demonstrations for Your I		sign
	52	ServiceWise Demonstration for You		sign

Highlight a knowledge item in the list panel and click the*KnowledgeDescription*tab in the detail panel (no window opens, but rather the knowledge item is forwarded directly from the detail panel)



☑ Show items of child folders \Knowledge Root

	ID	Title
	5	Defect Tracker speed tests - Linux
	6	Defect Tracker speed tests - Windows
	7	Defect Tracker speed tests - Mac
	33	We want to hide the service agreement info on C.
	34	Recorded demos for Browser MarketingWise
	35	Are reports customizable
	36	Does CRM track the history of an issue, or only it..
	37	Resolution to Install shield error when installing Br.
	38	Error "Seagate Crystal Reports database error" w.
	51	HelpDesk Demonstrations for Your Internal Supp..
	52	ServiceWise Demonstration for Your Customer S..
	53	WebEx Player

Title Defect Tracker speed tests - Linux

Status Finalized

Owner

Description

Cold start

2. Make the desired changes to the knowledge item properties. For more detailed information on knowledge item properties, please see the section,*Knowledge Properties*, later in this chapter.

3. Click the*OK*button to save the knowledge item changes.

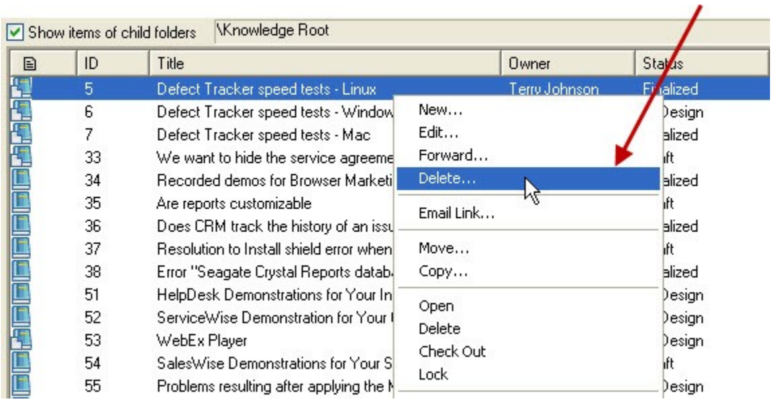
3.4 Deleting a Knowledge Item

Warning: Deleting a knowledge item is a non-reversible action!

To delete a knowledge item:

1. Do one of the following delete commands:

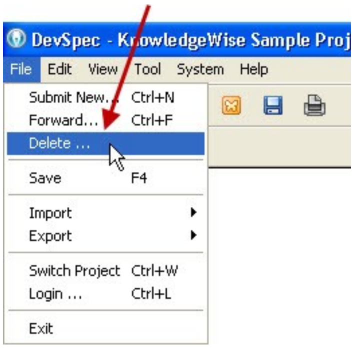
Right-click on a knowledge item in the list view and click*Delete...*



☑ Show items of child folders \Knowledge Root

	ID	Title	Owner	Status
	5	Defect Tracker speed tests - Linux	Terry Johnson	Fi alized
	6	Defect Tracker speed tests - Window		esign
	7	Defect Tracker speed tests - Mac		alized
	33	We want to hide the service agreeme		ft
	34	Recorded demos for Browser Marketi		alized
	35	Are reports customizable		ft
	36	Does CRM track the history of an issu		alized
	37	Resolution to Install shield error when		ft
	38	Error "Seagate Crystal Reports datab		alized
	51	HelpDesk Demonstrations for Your In		esign
	52	ServiceWise Demonstration for Your I		esign
	53	WebEx Player		esign
	54	SalesWise Demonstrations for Your S		ft
	55	Problems resulting after applying the I		esign

Highlight a knowledge item in the list view and from the menu bar, select*File*>*Delete…*

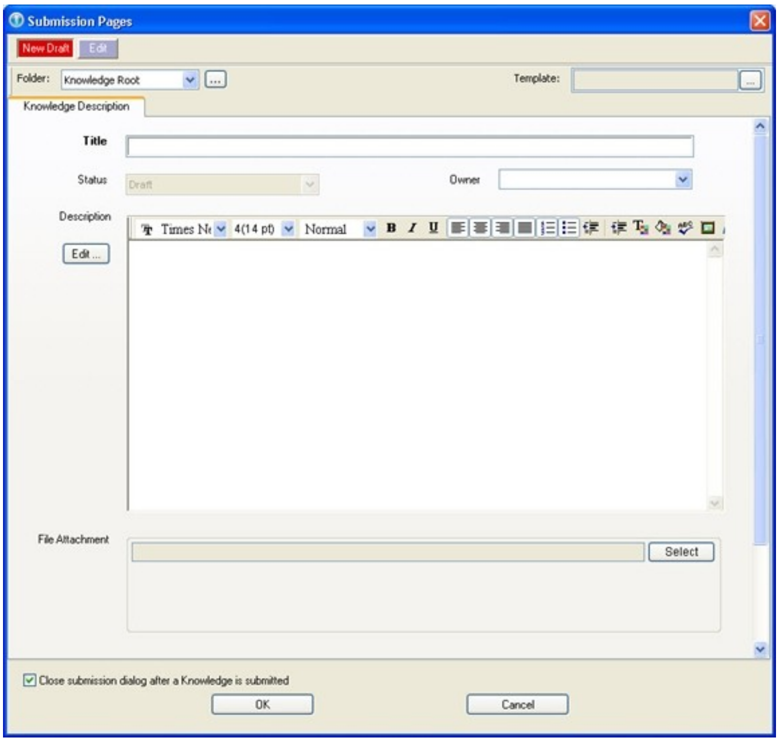


DevSpec - KnowledgeWise Sample Proj

2. In the confirmation dialog, click the*OK*button.

3.5 Knowledge Properties

This section covers the different properties that can be defined when creating, forwarding, or editing a knowledge item. More properties than those just mentioned here may be used to define knowledge items, when set up by the DevSpec administrator.



Submission Pages

Folder: Knowledge Root

Template:

Status: Draft

Owner:

☑ Close submission dialog after a Knowledge is submitted

Title

Title of the knowledge item. Cannot be left blank, and must be unique.

Status

Status of the knowledge item represents the current workflow state.

Owner

Owner of the knowledge item.

Description

Description of the knowledge item.

File Attachment

The original knowledge file can be attached to a knowledge item, and then later downloaded by other users. For more detailed information on managing file attachments, please see section 5,*Knowledge Item Attachments*, later in this chapter.

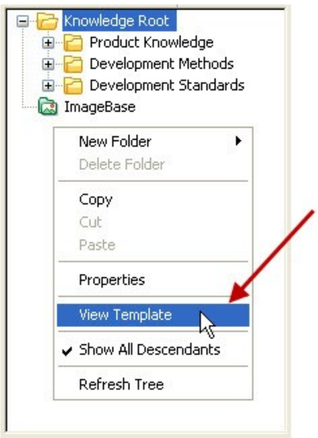
Folder

Folder of the knowledge item folder tree, under which the knowledge item will be saved.

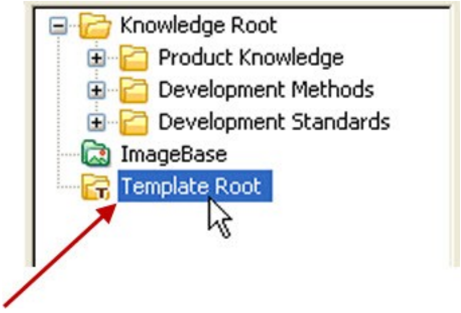
3.6 Creating a Knowledge Template

A template is a predefined collection of definitions that may be used to submit a new knowledge item, quickly and easily. To create a new template:

1. Right-click in the folder tree panel and click*View Template*.



2. Click on the newly appeared*Template Root*.



3. Create a new knowledge item template just as if an actual knowledge item were being created (please see the previous section,*Submitting a New Knowledge Item*). A template is not actually a knowledge item, although it may appear so in the list panel.

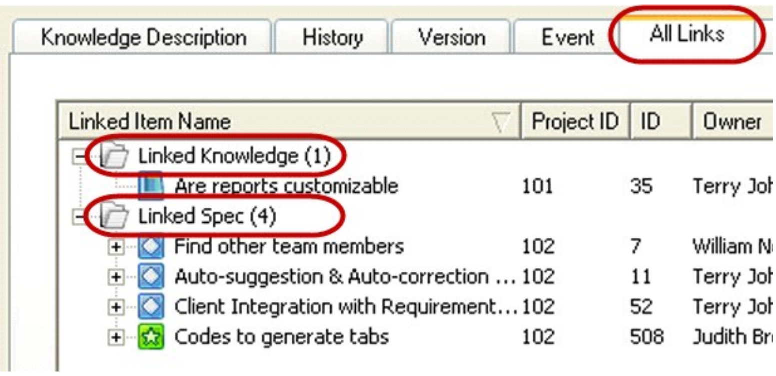
4 Linking Knowledge Items

Often there is a need to link a knowledge item with another knowledge item, specification, or requirement, because of a relationship or dependency amongst them. To link a knowledge item with other item(s):

1. Highlight a knowledge item in the list panel.

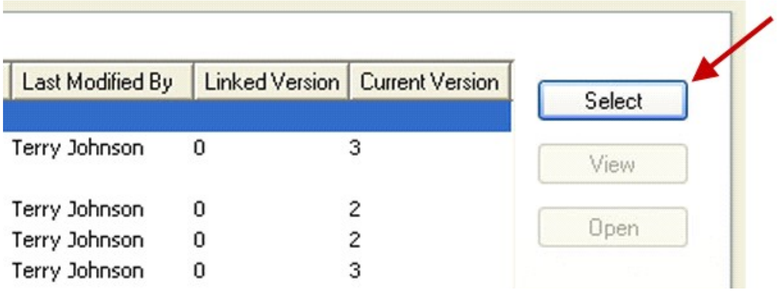
2. Go to the*All Links*tab in the detail panel, and highlight*Linked Knowledge*when linking a knowledge item, or

highlight*Linked Spec*when linking a specification or requirement.



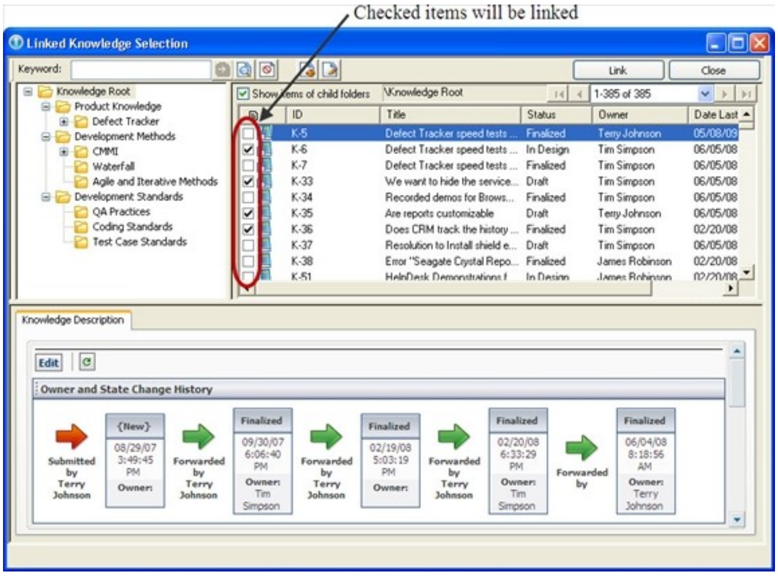
Linked Item Name	Project ID	ID	Owner
Linked Knowledge (1)			
Are reports customizable	101	35	Terry Joh
Linked Spec (4)			
Find other team members	102	7	William N
Auto-suggestion & Auto-correction …	102	11	Terry Joh
Client Integration with Requirement…	102	52	Terry Joh
Codes to generate tabs	102	508	Judith Br

3. Click the*Select*button in the top-right corner of the*All Links*tab.



Last Modified By	Linked Version	Current Version
Terry Johnson	0	3
Terry Johnson	0	2
Terry Johnson	0	2
Terry Johnson	0	3

4. In the newly opened dialog, check the items that are to be linked to the current knowledge item.


Checked items will be linked

	ID	Title	Status	Owner	Date Last
□	K-5	Defect Tracker speed tests …	Finalized	Terry Johnson	05/08/09
☑	K-6	Defect Tracker speed tests …	In Design	Tim Simpson	06/05/08
□	K-7	Defect Tracker speed tests …	Finalized	Tim Simpson	06/05/08
☑	K-33	We want to hide the service…	Draft	Tim Simpson	06/05/08
□	K-34	Recorded demos for Brows…	Finalized	Tim Simpson	06/05/08
☑	K-35	Are reports customizable	Draft	Terry Johnson	06/05/08
☑	K-36	Does CRM track the history …	Finalized	Tim Simpson	02/20/08
□	K-37	Resolution to Install shield e…	Draft	Tim Simpson	06/05/08
□	K-38	Error "Seagate Crystal Repo…	Finalized	James Robinson	02/20/08
□	K-51	HelpDesk Demonstrations f…	In Design	James Robinson	02/20/08

5. Click the*Link*button in the top-right corner to link the selected items, and close the dialog.

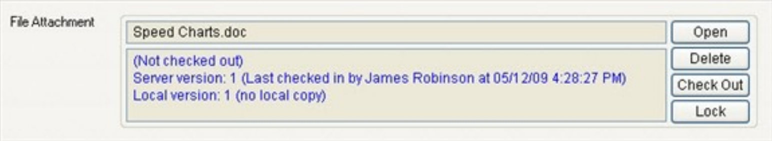
The*All Links*tab can also be used to:
View a linked item by selecting a linked item, and clicking the*View*button.
Open the attachment of a linked item by selecting a linked item and clicking the*Open*button.

5 Knowledge Item Attachments

In DevSpec, the need for a knowledge item typically comes from a file, such as a document, image, or HTML link. That original file can be attached to the new knowledge item. Other users can then download the file, make changes, and upload the revised version. Only one file at a time can be attached to a knowledge item.

All actions performed on file attachments are done using the*File Attachment*section-found in any area of the knowledge view where knowledge item properties can be defined, such as the submission, forward, or edit dialogs, or the*Knowledge Description*tab on the detail panel.

The*File Attachment*section also displays information about the file on the document server, the file on the current user's local computer, and whether the file has been checked out. The text is also color coded based on the local file: blue when the user has not yet downloaded a local copy; red when the user has downloaded a local copy, and it is currently being editing or has been edited; and black otherwise.



Speed Charts.doc

(Not checked out)
Server version: 1 (Last checked in by James Robinson at 05/12/09 4:28:27 PM)
Local version: 1 (no local copy)

In the screenshot above, the file information tells us that:
The current user has not yet downloaded the file to his local computer (hence, the text is blue).
The file is currently not checked out.
The server file is still on its first version, from when James Robinson first attached the file.

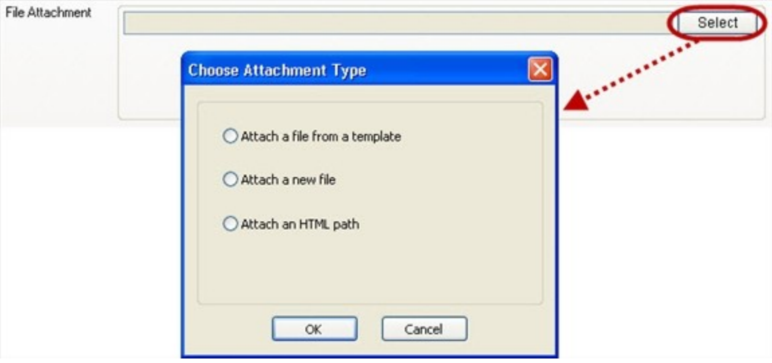
Icons in the list panel will denote which knowledge items have a file attachment:

Knowledge item**with**a file attachment.

Knowledge item**without**a file attachment.

5.1 Attaching a File

To attach a file to a knowledge item, in the*File Attachment*section, click the*Select*button. In the newly opened*Choose Attachment Type*dialog, the user can then choose a file from three possible sources:



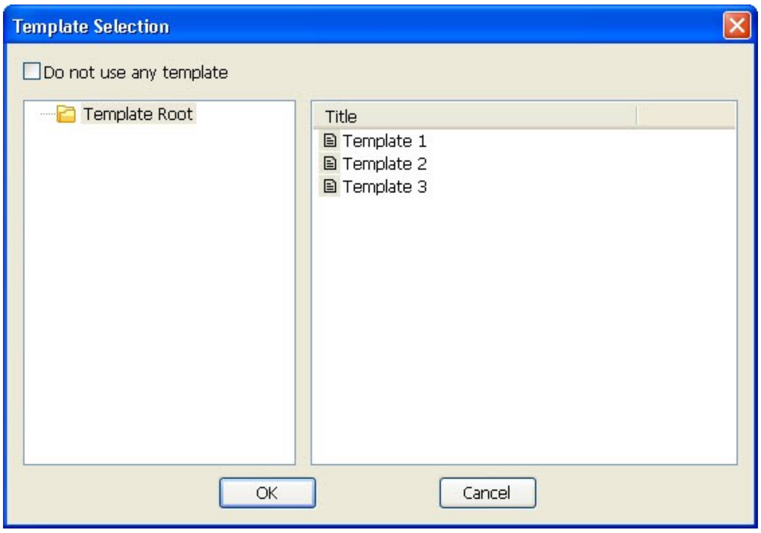
○ Attach a file from a template

○ Attach a new file

○ Attach an HTML path

Attaching a file from a template:

Users can attach a file from a knowledge item template. Since the file only needs to be duplicated within the document server, no upload time is needed, and the file is quickly attached. Select the*Attach a file from a template*radio button, and click the*OK*button to open the*Template Selection*dialog. Choose a template with an attachment, and click the*OK*button.

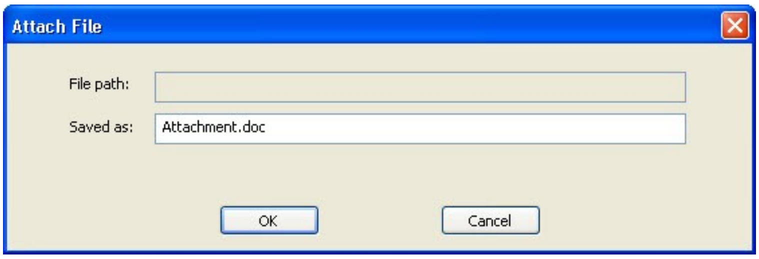


□ Do not use any template

Template Root

Title
Template 1
Template 2
Template 3

In the newly opened*Attach File*dialog, define the file name in the*Saved as*field, and click the*OK*button. A copy will be saved to the document server.

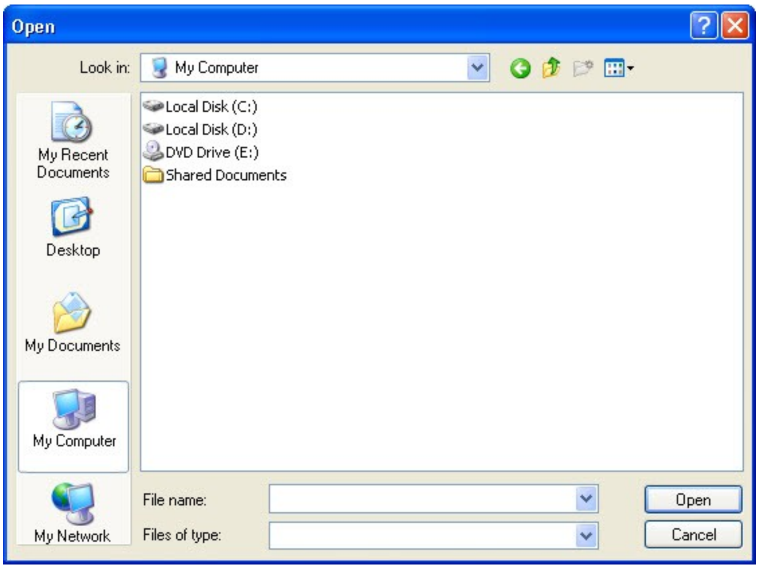


File path:

Saved as: Attachment.doc

Attaching a new file:

Users can attach a file from their local computer. Select the*Attach a new file*radio button, and click the*OK*button to open the*Open*dialog. Navigate to find the file on the local computer, and click the*Open*button.



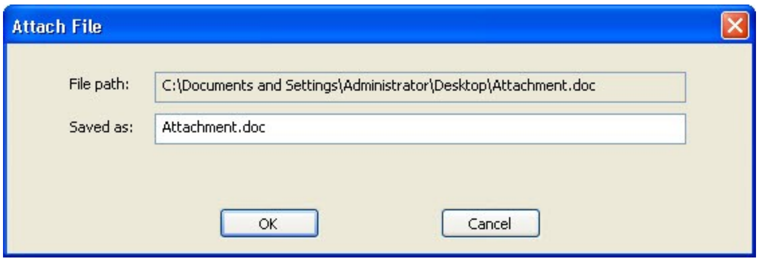
Look in: My Computer

Local Disk (C:)
Local Disk (D:)
DVD Drive (E:)
Shared Documents

File name:

Files of type:

In the newly opened*Attach File*dialog, define the file name in the*Saved as*field, and click the*OK*button. A copy will be saved to the document server.

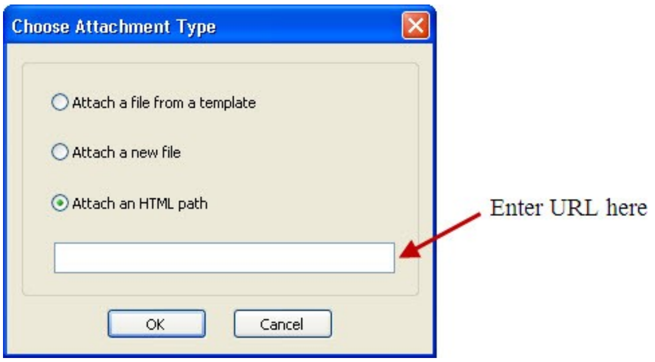


File path: C:\Documents and Settings\Administrator\Desktop\Attachment.doc

Saved as: Attachment.doc

Attaching an HTML path:

Users can save a URL to link to a file on a different server, such as a local-area network or the internet. The file itself is not saved on the DevSpec document server, rather the URL serves as a reference to the file. Select the*Attach an HTML path*radio button, enter the URL, andclick the*OK*button.



○ Attach a file from a template

○ Attach a new file

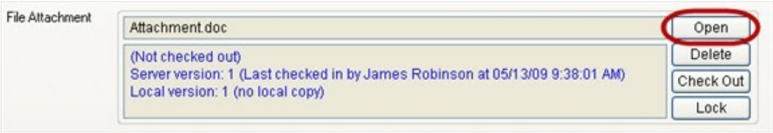
● Attach an HTML path

Enter URL here

5.2 Downloading and Opening a File

Once a file has been attached to a knowledge item, it can be downloaded to and/or opened on the user’s local machine. To download and/or open an attached file:

1. In the*File Attachment*section, click the*Open*button.

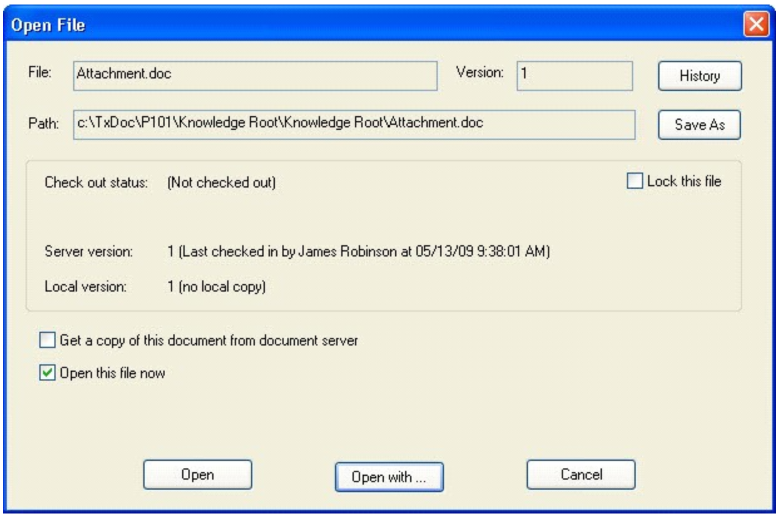


File Attachment

Attachment.doc

(Not checked out)
Server version: 1 (Last checked in by James Robinson at 05/13/09 9:38:01 AM)
Local version: 1 (no local copy)

2. The*Open File*dialog opens.



Open File

File: Attachment.doc
Version: 1

Path: c:\TxDoc\P101\Knowledge Root\Knowledge Root\Attachment.doc

Check out status: (Not checked out)

□ Lock this file

Server version: 1 (Last checked in by James Robinson at 05/13/09 9:38:01 AM)

Local version: 1 (no local copy)

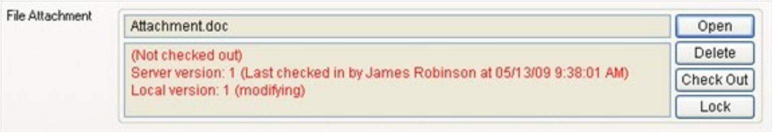
□ Get a copy of this document from document server

☑ Open this file now

The*Path*control displays where the file will be saved on the user’s local machine. Click the*Save As*button to choose a new directory and/or file name. To define a default path, please see chapter 2, section 3,*User Preferences*.
Check the*Get a copy of this document from document server*checkbox to download the file from the document server. After the first time a file has been downloaded, unless this checkbox is checked, the file on the local computer will be opened.
Check the*Open this file now*to open the file after the download is complete.

3. Click the*Open*button to download and/or open the file, OR click the*Open with...*button to choose a program, with which the file will be opened.

4. Press F5 to update the file information in the*File Attachment*section.



File Attachment

Attachment.doc

(Not checked out)
Server version: 1 (Last checked in by James Robinson at 05/13/09 9:38:01 AM)
Local version: 1 (modifying)

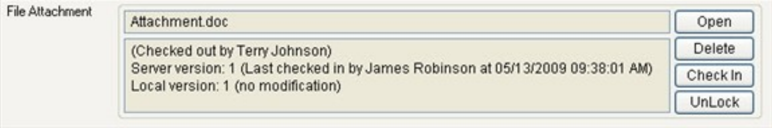
The file attachment information in the screenshot above states that the current user has downloaded the first version of their local copy, and is modifying it (i.e. the file is opened on the current user’s computer).

5.3 Checking Out and Locking a File

After a user downloads a file, they may wish to make some changes and then update the attachment with their modified file. A user making any changes to a file can “check out” and “lock” a file from being modified by other users. This prevents other users’ changes in the meanwhile from being overridden.

To check out and lock a file simply denotes which user has the control to modify the server file (other users can still download the latest file in the meanwhile). Unless given the required privileges, users cannot check out a file until it has been unlocked by the user who originally checked it out.

* Checking out and locking are events that always happen together (i.e. when a user checks out a file, it is automatically locked as well, and vise versa).



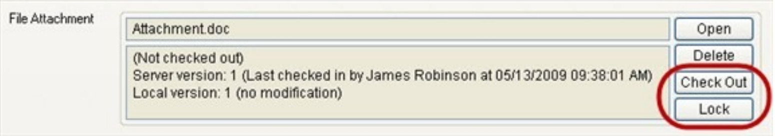
Attachment.doc

(Checked out by Terry Johnson)
Server version: 1 (Last checked in by James Robinson at 05/13/2009 09:38:01 AM)
Local version: 1 (no modification)

The above screenshot is an example when a file is checked out. The file was originally attached by James Robinson, and is currently checked out by Terry Johnson.

To check out a file:

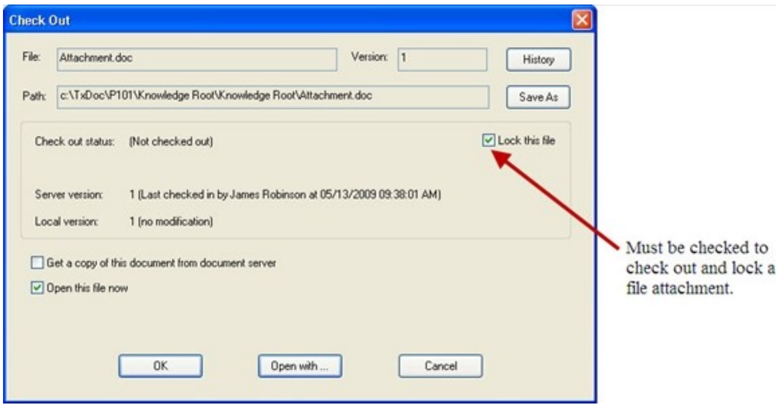
1. In the*File Attachment*section, click the*Check Out*button, OR the*Lock*button.



Attachment.doc

(Not checked out)
Server version: 1 (Last checked in by James Robinson at 05/13/2009 09:38:01 AM)
Local version: 1 (no modification)

2. A new dialog opens (similar to the dialog when downloading and/or opening a file).



File: Attachment.doc
Version: 1
Path: c:\TxDoc\P101\Knowledge Root\Knowledge Root\Attachment.doc

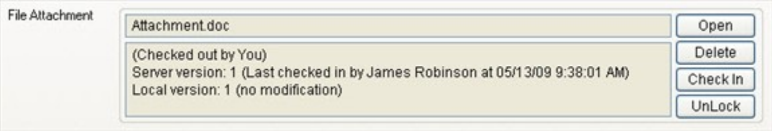
Check out status: (Not checked out)
☑ Lock this file
Server version: 1 (Last checked in by James Robinson at 05/13/2009 09:38:01 AM)
Local version: 1 (no modification)

☐ Get a copy of this document from document server
☑ Open this file now

Must be checked to check out and lock a file attachment.

The*Lock this file*checkbox must be checked. If not, the file will not be checked out and locked, rather only downloaded.
Click the*History*button for more information on the file’s modification history.
Check the*Get a copy of this document from document server*checkbox to download the file from the document server. After the first time a file has been downloaded, unless this checkbox is checked, the file on the local computer will be opened.
Check the*Open this file now*to open the file after the download is complete.
3. Click the*Open*button to download and/or open the file, OR click the*Open with...*button to choose a program, with which the file will be opened. The file will be checked out and locked.

4. Press F5 to update the file information in the*File Attachment*section.



Attachment.doc

(Checked out by You)
Server version: 1 (Last checked in by James Robinson at 05/13/09 9:38:01 AM)
Local version: 1 (no modification)

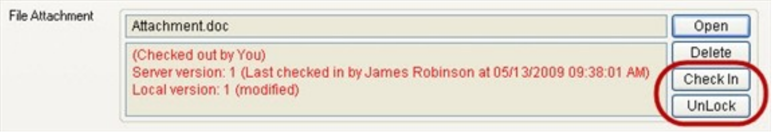
Note: The user, to whom the file is checked out, is different from the owner of the corresponding knowledge item.

5.4 Checking In and Unlocking a File

When a user who has checked out a file is ready to update the server file with their modified local file, they can “check in” and/or “unlock” the file.

To check in a file:

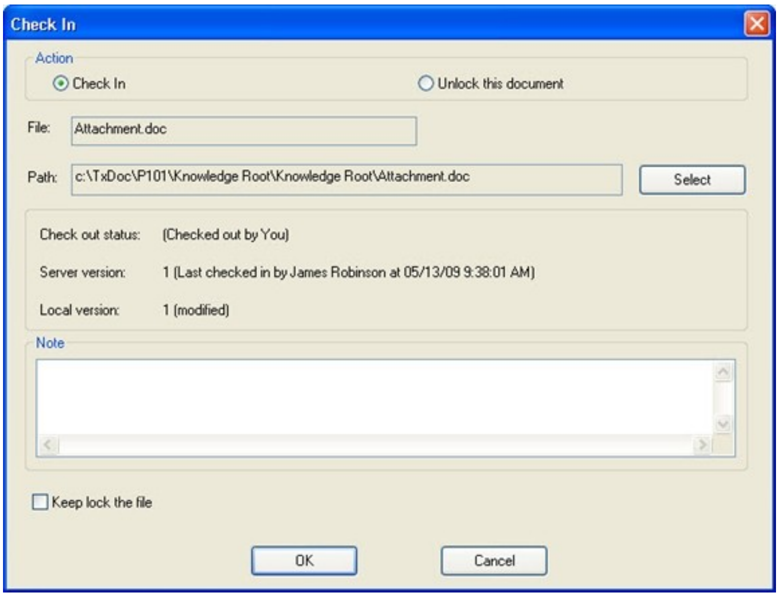
1. In the*File Attachment*section, click the*Check In*button, OR the*Unlock*button.



Attachment.doc

(Checked out by You)
Server version: 1 (Last checked in by James Robinson at 05/13/2009 09:38:01 AM)
Local version: 1 (modified)

2. A new dialog opens.



● Check In ○ Unlock this document

File: Attachment.doc

Path: c:\TxDoc\P101\Knowledge Root\Knowledge Root\Attachment.doc

Check out status: (Checked out by You)

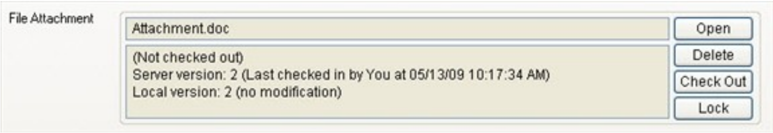
Server version: 1 (Last checked in by James Robinson at 05/13/09 9:38:01 AM)

Local version: 1 (modified)

□ Keep lock the file

To unlock the file, without replacing the server file, click the *Unlock this document*radio button at the top. All other controls become inactive, or “grayed-out”.
No changes will be made to the server file.
The file will be unlocked-allowing other users to check out the file.
To replace the server file with the local file, but keep the file locked, click the*Check In*radio button at the top, and check the*Keep lock the file*checkbox at the bottom.
The server file will be replaced with the current user’s local file.
The file will remain checked out-allowing the current user to make further modifications to the server file later.
To replace the server file with the local file, and unlock the file at the same time, click the*Check In*radio button at the top, and uncheck the*Keep lock the file*checkbox at the bottom.
The server file will be replaced with the current user’s local file.
The file will be unlocked-allowing other users to check out the file.
3. Click the*OK*button.

4. Press F5 to update the file information in the*File Attachment*section.



Attachment.doc

(Not checked out)
Server version: 2 (Last checked in by You at 05/13/09 10:17:34 AM)
Local version: 2 (no modification)

In the above screenshot, the file was checked in and unlocked.