An Oracle White Paper
September 2011

Oracle Team Productivity Center Overview

Introduction

Oracle Team Productivity Center is an Application Lifecycle Management (ALM) tool that enables software development teams to collaborate and work productively together when developing applications using JDeveloper.

Application Lifecycle Management can be described as the management of the lifecycle development practices (requirements, build, test, change control, defect management etc) integrated together through application of process, reporting, traceability and collaboration. The better this integration, the better the software. However, in the past many of these practices worked in isolation - in functional silos that did not lend themselves to collaboration. However, collaboration has become increasingly necessary in the software development world where teams work across organizations, time zones, cultures and business functions.

Oracle Team Productivity Center provides a framework to enable third party Application Lifecycle Management tools to be integrated into Oracle JDeveloper. These repositories include task and project management, version control, document management, software bug reporting, and build and management systems. The integration of repositories in JDeveloper enables users to directly interact with existing ALM artifacts whilst working in their IDE. Additionally, Oracle TPC provides configurable team management facilities designed to improve productivity and communication among team members.

Oracle JDeveloper supports the complete development lifecycle with integrated features for modeling, coding, debugging, testing, profiling, tuning and deploying applications. With the addition of Oracle Team Productivity Center developer productivity is improved and functional silos are minimized through flexible user and team maintenance; query and update of third party ALM repositories; linking of disparate ALM artifacts through extensible tagging and relationship and context management facilities; content management at artifact and team level; associating SCM versioned files with artifacts; enhanced developer collaboration through integrated chat client

Installation

Oracle Team Productivity Center server install is downloadable from Oracle Technology Network (OTN) as a platform-independent JAR file. Full details are contained in the Oracle Fusion Middleware Installation Guide for Oracle Team Productivity Center Server

The client model is a JDeveloper extension shipped with JDeveloper. It is accessed through the View –> Teams menu

Connectors to enable the integration of ALM repositories require both a server install (as detailed in the installation guide) and installation in JDeveloper as extensions. The Oracle JDeveloper Help Center contains detailed help on installing connectors and using Oracle Team Productivity Center. This help is available once the client software has been installed

Architecture

The basic architecture of the Team Productivity Center product is comprised of three main components illustrated in Figure 1

1. JDeveloper Client Model
2. Team Productivity Center Server
3. Team Productivity Center Connectors

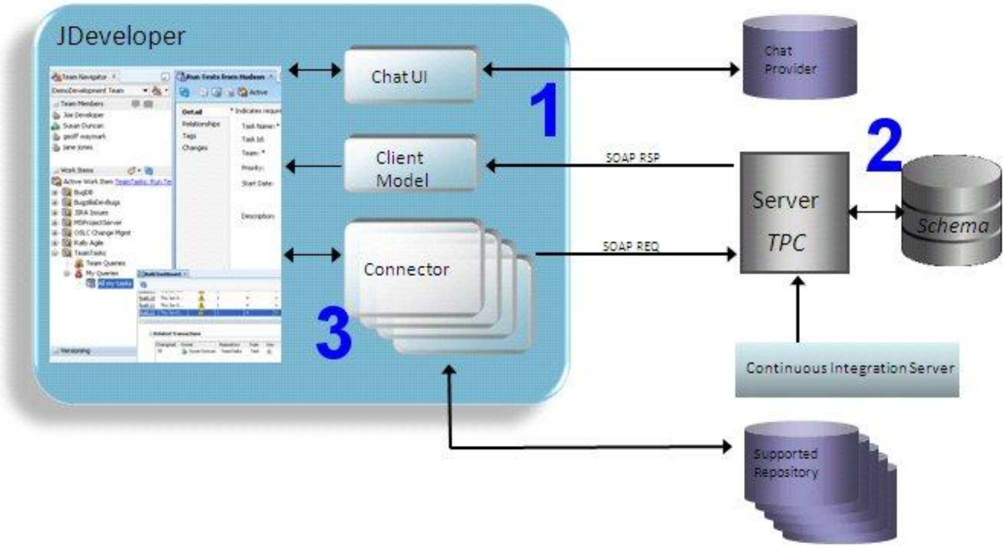


Figure 1. Oracle Team Productivity Center Architecture

JDeveloper Client Model

The JDeveloper Team Productivity Center extension acts as the TPC client framework. The Team Navigator drives all team related features. The TPC extension includes a chat interface allowing JDeveloper users to collaborate with others directly in JDeveloper without having to leave their IDE.

Team Productivity Center Server

The TPC team server allows JDeveloper clients to connect and retrieve team related information. The TPC server also has a password vault for each user to enter credentials in order to seamlessly connect to their 3rd party repositories.

The team server stores centralized information (such as tags, relationships, team information) in a database schema along with query criteria against connector repositories. This database also provides storage of the built-in Task repository that teams can choose to install.

Build and test results are stored against the related work items and SCM check-ins to provide the seamless end-to-end integration of ALM artifacts.

Team Productivity Connectors

The TPC connector infrastructure allows ALM repositories to be instantiated and a UI exposed directly inside of JDeveloper. Through the published interface built on open standards, the connector can expose repository fields using a meta-driven user interface and support basic create/update/delete operations on those objects. Each connector can support a number of artifact object types each with multiple user interface definitions. The connector also defines what fields in each artifact type are queryable by the user and how to query them. By implementing the connector interface and supplying some XML metadata to define a connector, a developer can quickly integrate their own repository into the Team Productivity Center framework, allowing any 3rd party repository to easily be exposed inside of JDeveloper.

Connector Framework

The connector framework is used both by Oracle and by partners and others to create connectors.

The Oracle Team Productivity Center Developer Guide provides details on how to develop a connector to integrate a third-party ALM repository with JDeveloper. The TPC connector infrastructure allows ALM repositories to be instantiated and a UI exposed directly inside of JDeveloper. Through the published interface built on open standards, the connector can expose repository fields using a meta-driven user interface and support basic create/update/delete operations on those objects. Each connector can support a number of artifact object types each with multiple user interface definitions. The connector also defines what fields in each artifact type are queryable by the user and how to query them. By implementing the connector interface and supplying some XML metadata to define a connector, a developer can quickly integrate their own repository into the Team Productivity Center framework, allowing any 3rd party repository to easily be exposed inside of JDeveloper. This developer guide is available in the JDeveloper Help Center. To facilitate developing a connector, a sample connector together with the source code is available for download

A Special Case – The Task Connector

This connector is supplied by Oracle to access the task repository that is shipped with Team Productivity Center. It provides for teams to create named tasks with description and assign them to users with priority, start and end date and status. Users login to the Task repository using the Team Productivity Center logon. The tasks are stored on the Team Server database.

Team Navigator

The Team Navigator is accessed from the View -> Teams menu. The Team Navigator is the access point for Oracle Team Productivity Center in JDeveloper. The first action of a TPC user/admin is to logon to TPC. The server installation includes setting up the TPC administrator who then logs in here to set up additional users (see server installation guide for details on accessing the administrator user).

Once logged in, at the top of the Team Navigator is the team drop down menu. This provides access to the user's assigned teams. In Figure 2 the Business Services team is currently open. Selecting a team changes the access information in each of the accordions: Team Members, Work Items and Versioning. These display the members of the team, the ALM repositories that the team has access to and access to the versioning repositories for the team.

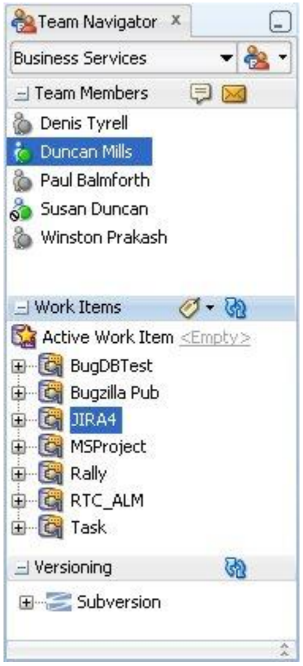
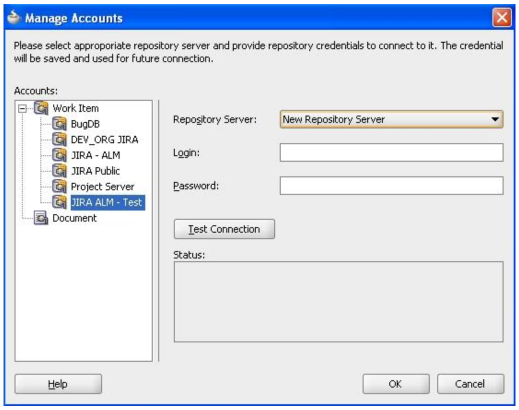


Figure 2. Team Navigator

To the right of the team selector is the Team Productivity Center administration menu. This menu provides access to the administration functions such as team and repository management (see Administration section of this whitepaper for more detail). In addition users can reset their password from this menu and manage their individual logins to the applicable third party ALM repositories. Figure 3 shows the Manage Accounts dialog. In this example the user would enter their login and password to the JIRA ALM – Test repository and then test the connection



Manage Accounts

Please select approporiate repository server and provide repository credentials to connect to it. The credential will be saved and used for future connection.

Figure 3. Manage Accounts dialog

Team Members

This accordion shows each member of the currently selected team. The icons to the left of the team member name indicate their TPC and chat status. Hovering the mouse over a team member gives additional status information. Double-clicking on a team member or clicking the chat icon in the Team Members accordion header starts a chat session. Clicking the mail icon in the header creates a new mail to the selected team member.

Chat Dialog

The chat window (Figure 4) is invoked from the View menu and is used to connect to the chat server, manage contacts and edit chat settings. It displays all contacts – not just those who belong to the currently selected team. Chat can be used regardless of whether the user is logged into the Team Server or not. The chat functionality is provided as part of the Oracle Team Productivity Center client

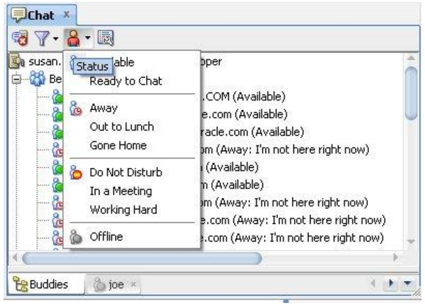


Figure 4. Chat Dialog

Work Items

In Oracle Team Productivity Center the term 'Work Item' is applied to any artifact from a connected ALM repository, for instance a JIRA issue.

The Work Items accordion shows all the ALM repositories that are used by the team. In Figure 5 there are six repositories listed for the Business Services team: Bug, Bugzilla, JIRA, Microsoft Project, Rally and Task.

Taking the JIRA repository as an example: There are two standard nodes that appear for each repository. Team Queries contains queries created by a team member with team administration rights. Any member of the team can run a team-level work item query. An individual team member can also save queries under the My Queries node. These queries are only visible to that team member.

It should be noted that the work items returned by a query are not stored by Team Productivity Center. Instead, the query criteria are saved and the query is performed against the relevant repository when the query is run.

The Active Work Item allows the user to specify the ALM repository artifact that she is currently working on. It provides a quick access link to open the work item. For other usages of the Active Work Item see Changes section of this whitepaper.

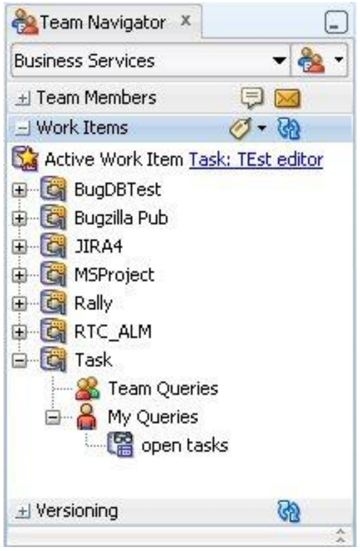


Figure 5. Work Item Accordion

Versioning

The versioning accordion allows a user to login and interact with their team's versioning system using the Team Navigator. The functionality provided mirrors that of the standalone Versioning Navigator. At the team level, the team administrator can specify the repository location of the team's versioned items. One advantage of this is that a new team member can be provided with their Team Productivity Center login and the Team Navigator will provide them with the information they need to review the team's repositories and documents from within JDeveloper.

Managing Queries

The integration of Application Lifecycle Repositories into Oracle JDeveloper is the lifeblood of Oracle Team Productivity Center. This integration enables JDeveloper users to work more productively with these systems and take advantage of the additional centralized services that Team Productivity Center provides to link work items from disparate repositories together.

Queries are the mechanism by which Team Productivity Center interacts with work items. As stated earlier, a work item is the generic term that is given to an artifact from an integrated repository. Search filters are saved by Team Productivity Center as either team level or user queries, depending on the role of the user (discussed later) and can be run or edited at any time.

Running and Editing A Query

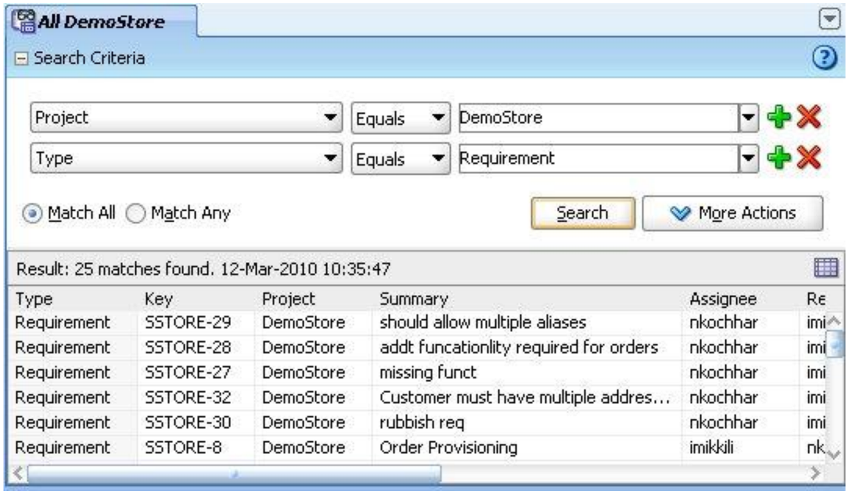
There are a number of ways to run/edit a new query

The context menu for a repository node in the Work Items accordion provides three ways to run/edit a query:

- Query by ID
    - Use this if you know the ID of the work item that you wish to query.
- Query By Tag
    - Tagging allows team members to identify and relate groups of work items with arbitrary keywords and is covered later in this paper.
- New Query
    - Use the declarative dialog (Figure 6) to build search criteria on the selected repository

Search Criteria

Figure 6 shows the result of running the ‘AllDemoStore’ team query. The top portion of the tab panel shows the search criteria that were saved for the query. Each queryable field is presented in the drop-down list and, as appropriate, both an operator and valid values are selectable. Notice in this example that the valid values for both fields have been selected from a drop-down list. If valid values are not necessary then a write-in field is available. For date fields, a date picker is provided.



Project ▾ | Equals ▾ | DemoStore ▾

Type ▾ | Equals ▾ | Requirement ▾

● Match All ○ Match Any

Result: 25 matches found. 12-Mar-2010 10:35:47

Type	Key	Project	Summary	Assignee	Re
Requirement	SSTORE-29	DemoStore	should allow multiple aliases	nkochhar	imi
Requirement	SSTORE-28	DemoStore	addt funcationlity required for orders	nkochhar	imi
Requirement	SSTORE-27	DemoStore	missing funct	nkochhar	imi
Requirement	SSTORE-32	DemoStore	Customer must have multiple addres...	nkochhar	imi
Requirement	SSTORE-30	DemoStore	rubbish req	nkochhar	imi
Requirement	SSTORE-8	DemoStore	Order Provisioning	imikkili	nk

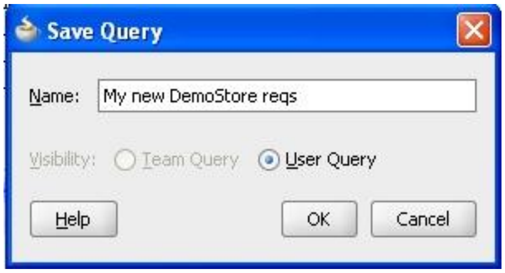
Figure 6. AllDemoStore Requirements query

Saving the Query

The bottom portion of the tab panel shows the results returned from running this query, using the Search button. As is standard with JDeveloper, the columns that are displayed are customizable and can be reordered using normal drag and drop functionality

The query can be saved using the standard JDeveloper Save menu/toolbar options or through the More Actions button available in this panel. The button also provides Save As to allow search criteria to be edited and saved as a new query. Figure 7 shows the Save Query dialog.

Note that although the original DemoStoreRequirements query was a team query, the user that ran the query has edited the search criteria and is creating a new user query. This user does not have the team administration role necessary to save the query as a Team Query.



Name: My new DemoStore reqs

Visibility: ○ Team Query ● User Query

Figure 7. Save Query dialog

Creating New Work Items

The user can use the context menu on the ALM repository node to create a new artifact in that underlying repository provided that she has the correct privileges on the repository and that the connector allows for the creation of artifacts

Working with Work Items

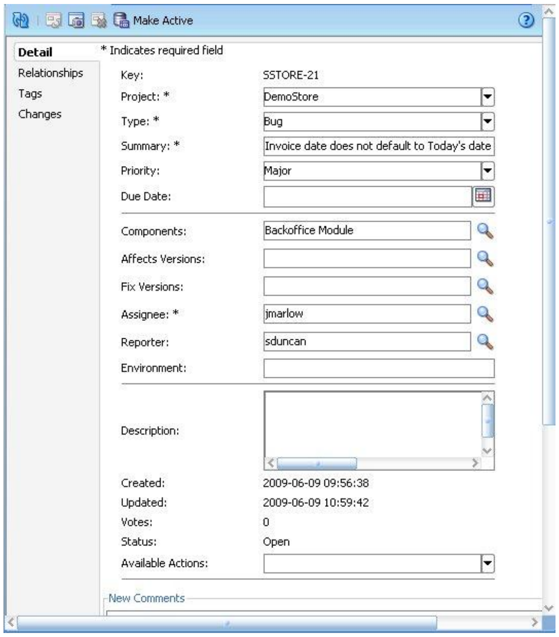
Oracle Team Productivity Center allows users to work with individual work items from within the JDeveloper IDE. Figure 8 shows one work item. The window consists of up to 5 finger tabs: Detail, Relationships, Tags, Attachments, Changes

For each work item type (i.e. for each repository type: JIRA, Task, Bugzilla, etc) the fields that are displayed, which fields are updateable, the type of validation that is carried out, and the interactions available with the repository are all determined by that repository's API. For instance, the current JIRA API does not provide for uploading files against issues. Therefore the Attachments finger tab does not appear for a work item from a JIRA repository (as in Figure 8).

The detail tab displays all the available detail information for the work item. Fields may be updateable if allowed by the repository API. Wherever possible declarative declaration is incorporated. Taking the example in Figure 8:

- Key – read-only field
- Project – dropdown select list, and denoted mandatory by the '*'
- Due Date – date picker
- Components – write-in
- Assignee – list of values
- Description – multi-line text with scrollbars

Updates to a work item are saved using the standard Save functionality in JDeveloper.



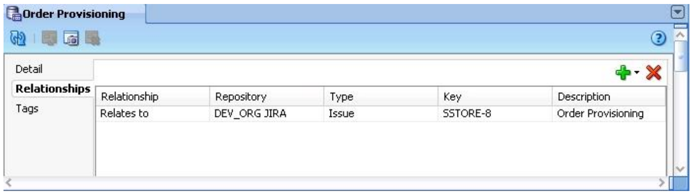
* Indicates required field

Key:	SSTORE-21
Project: *	DemoStore
Type: *	Bug
Summary: *	Invoice date does not default to Today's date
Priority:	Major
Due Date:	
Components:	Backoffice Module
Affects Versions:	
Fix Versions:	
Assignee: *	jmarlow
Reporter:	sduncan
Environment:	
Description:	
Created:	2009-06-09 09:56:38
Updated:	2009-06-09 10:59:42
Votes:	0
Status:	Open
Available Actions:	

Figure 8. Work item detail tab

Relationships

Oracle Team Productivity Center provides a mechanism for creating relationships between work items from the same or different repositories. For instance a relationship between a requirement in a JIRA repository and the bugs in a Bugzilla repository project can be created. This allows any user browsing either end of the relationship to open the associated work item(s), provided they have access to that repository. This is one of the major benefits of Oracle Team Productivity Center – the ability to integrate disparate repositories together. Figure 9 shows the Relationship tab of an MS Project task – and the associated JIRA issue. Relationships between work items are visible across teams and across multiple repositories.

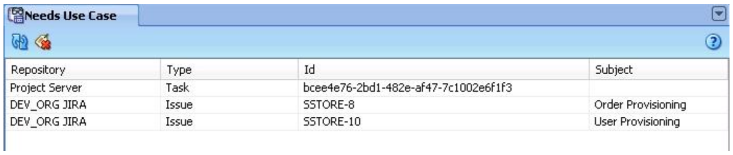


Relationship	Repository	Type	Key	Description
Relates to	DEV_ORG JIRA	Issue	SSTORE-8	Order Provisioning

Figure 9. Relationships finger tab

Tags

A less formal method grouping work items together is to use tags. Tags differ from relationships in that they apply only within a team. Figure 10 shows the work items tagged with 'Needs Use Case'. Note that the list contains two JIRA issues and a project task. It's possible to filter a list of tagged items by repository by querying by tag on a specific repository. The team server stores the tag list details. Double-clicking on one of the work items in a tag list queries that work item from its containing repository.

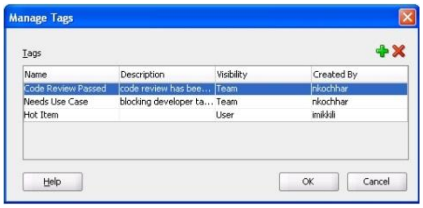


Repository	Type	Id	Subject
Project Server	Task	bcee4e76-2bd1-482e-af47-7c1002e6f1f3	
DEV_ORG JIRA	Issue	SSTORE-8	Order Provisioning
DEV_ORG JIRA	Issue	SSTORE-10	User Provisioning

Figure 10. Tagged work items

Tag Visibility and Maintenance

Team Administrators can create tags at the team level for use by all members of the team members. For instance, in Figure 11 the 'Needs Use Case' tag can be applied to any work item by any member of the team. All team members can query work items using the tag. Conversely, the Hot Items tag has a visibility of User, and because this tag was created by imikkili, and only she will be able to apply that tag to a work item and query work items using that tag.

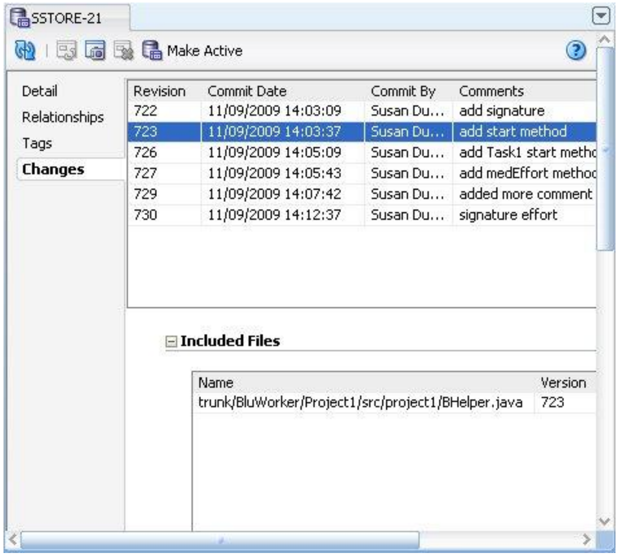


Name	Description	Visibility	Created By
Code Review Passed	code review has bee...	Team	nkochhar
Needs Use Case	blocking developer ta...	Team	nkochhar
Hot Item		User	imikkili

Figure 11. Manage Tags

Changes

The Changes finger tab displays associations of files committed to the versioning repository and any build/test results with work items. (see Figure 12). Double-clicking on a file opens a read-only version in the editor. See the section on Build & Test Integration further down in this paper.

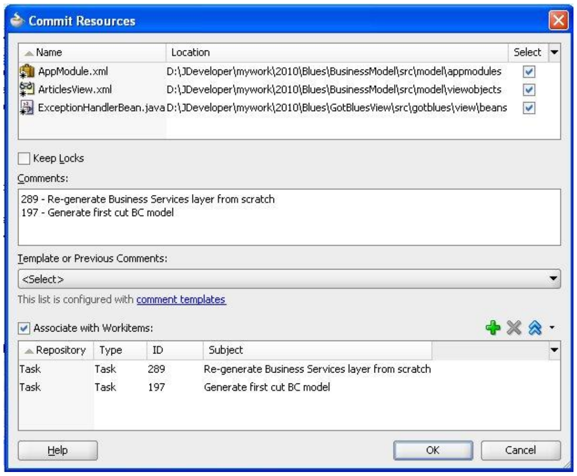


Revision	Commit Date	Commit By	Comments
722	11/09/2009 14:03:09	Susan Du...	add signature
723	11/09/2009 14:03:37	Susan Du...	add start method
726	11/09/2009 14:05:09	Susan Du...	add Task1 start methc
727	11/09/2009 14:05:43	Susan Du...	add medEffort methoc
729	11/09/2009 14:07:42	Susan Du...	added more comment
730	11/09/2009 14:12:37	Susan Du...	signature effort

Name	Version
trunk/BluWorker/Project1/src/project1/BHelper.java	723

Figure 12: Changes tab

When performing a commit to the SCM repository the user is able to associate the Active Work Item and/or select work items from a Tag she has created or any other work item that she has open in JDeveloper.



Name	Location	Select
AppModule.xml	D:\JDeveloper\mywork\2010\Blues\BusinessModel\src\model\appmodules	☑
ArticlesView.xml	D:\JDeveloper\mywork\2010\Blues\BusinessModel\src\model\viewobjects	☑
ExceptionHandlerBean.java	D:\JDeveloper\mywork\2010\Blues\GotBluesView\src\gotblues\view\beans	☑

□ Keep Locks

Comments:

289 - Re-generate Business Services layer from scratch
197 - Generate first cut BC model

Template or Previous Comments:

<Select>

This list is configured with comment templates

☑ Associate with Workitems:

Repository	Type	ID	Subject
Task	Task	289	Re-generate Business Services layer from scratch
Task	Task	197	Generate first cut BC model

Figure 13: Associate commit with work items dialog

Attachments

Attachments provide a mechanism for uploading either documents or references (URL) to a pre-specified content management system associated with the work item repository. This functionality is dependent on the relevant API being available for the repository type. For instance this API is not available for MS Project Server and so the attachments finger tab does not appear on the work item detail panel

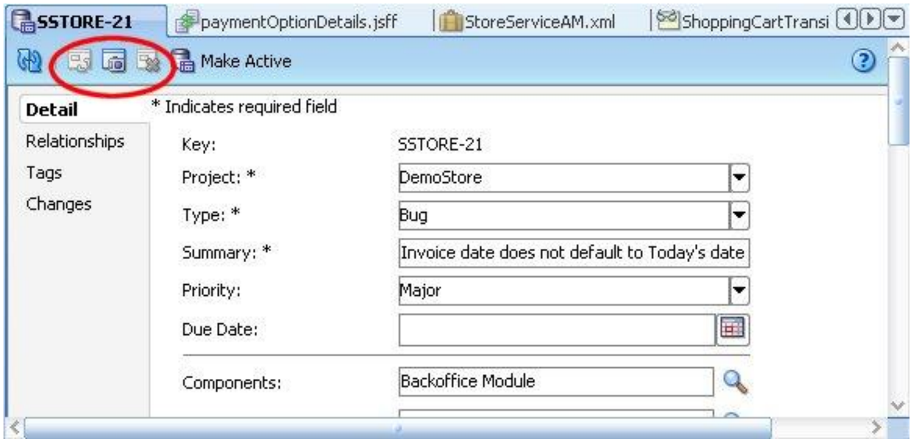
Active Work Item

As explained earlier, the Active Work Item allows the user to specify one work item that she is currently focus on. Figure 14 shows the Make Active icon – ready for this work item to be set as active. On activation, the icon changes to an 'Active' state. There are advantages to using this shortcut to a specific work item:

- Quickly open that work item from the Navigator
- Automatically associate versioned files with this workitem

Context

Context is the method by which a user can save the current state of their JDeveloper IDE (window layout, open files) against a specific work item within a team. This allows the user to be more productive when having to switch between multiple tasks or bug fixes, for example, and quickly restore their IDE when ready to resume work on the task against which context has been stored. Figure 14 shows an extract of the IDE. The user is about to save context against work item SSTORE-21. In this extract it appears the user wants to ensure that when she resumes work on this work item she can restore context and the paymentOptionsDetails.jsff and other currently open files will be re-opened



* Indicates required field

Key: SSTORE-21

Project: * DemoStore

Type: * Bug

Summary: * Invoice date does not default to Today's date

Priority: Major

Due Date:

Components: Backoffice Module

Figure 14. Save Context

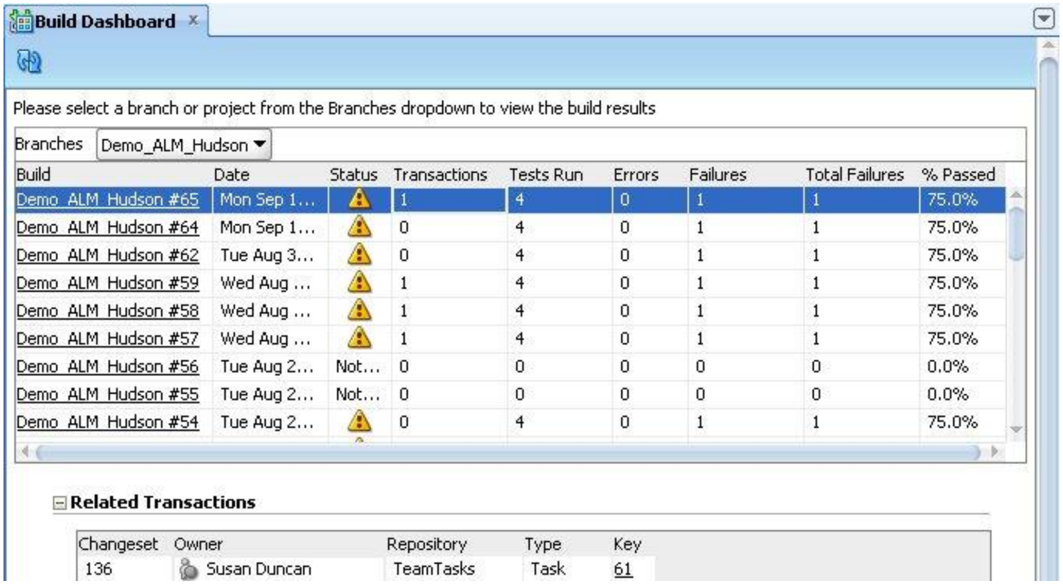
Build and Test Integration

New in the 11.1.2.0 version of Team Productivity Center is the ability to incorporate build and test results from both the Hudson and CruiseControl $_{tm}$ continuous integration tools. Results are returned from these systems using plugins deployed to them. To install the plugin use the Team Productivity Server Installer available from Oracle Technology Network.

Build Dashboard

Available from the team menu, this dashboard displays all the results being returned from your continuous integration system. It summarizes the transactions and any test runs that the CI server has been configured to return.

In the Related Transactions area of the dashboard the build information is tied to the applicable work item that was associated with the SCN check-in. Figure 15 shows that Build #65 from a Hudson server had one transaction and included 4 Tests, of which 1 failed.



Please select a branch or project from the Branches dropdown to view the build results

Branches: Demo_ALM_Hudson

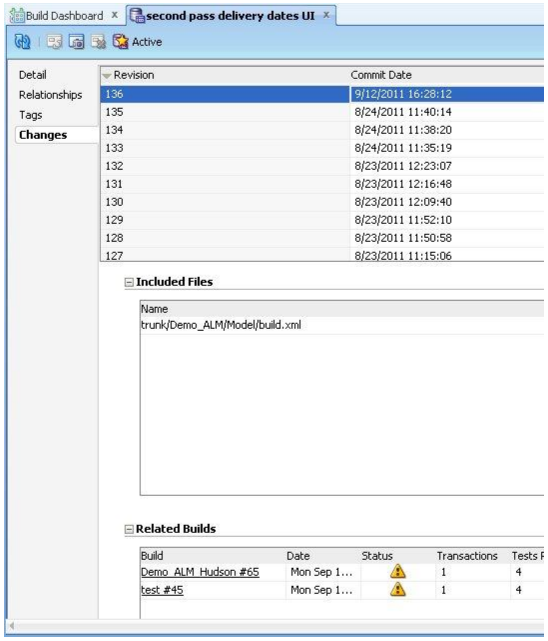
Build	Date	Status	Transactions	Tests Run	Errors	Failures	Total Failures	% Passed
Demo_ALM_Hudson #65	Mon Sep 1…	⚠	1	4	0	1	1	75.0%
Demo_ALM_Hudson #64	Mon Sep 1…	⚠	0	4	0	1	1	75.0%
Demo_ALM_Hudson #62	Tue Aug 3…	⚠	0	4	0	1	1	75.0%
Demo_ALM_Hudson #59	Wed Aug …	⚠	1	4	0	1	1	75.0%
Demo_ALM_Hudson #58	Wed Aug …	⚠	1	4	0	1	1	75.0%
Demo_ALM_Hudson #57	Wed Aug …	⚠	1	4	0	1	1	75.0%
Demo_ALM_Hudson #56	Tue Aug 2…	Not…	0	0	0	0	0	0.0%
Demo_ALM_Hudson #55	Tue Aug 2…	Not…	0	0	0	0	0	0.0%
Demo_ALM_Hudson #54	Tue Aug 2…	⚠	0	4	0	1	1	75.0%

Related Transactions

Changeset	Owner	Repository	Type	Key
136	Susan Duncan	TeamTasks	Task	61

Figure 15: Build Dashboard

In Related Transactions you can see that the changeset related to this build was 136. Clicking on the Key field takes you to the related work item(s), in this example a Task 61, as shown in Figure 16.



Revision	Commit Date
136	9/12/2011 16:28:12
135	8/24/2011 11:40:14
134	8/24/2011 11:38:20
133	8/24/2011 11:35:19
132	8/23/2011 12:23:07
131	8/23/2011 12:16:48
130	8/23/2011 12:09:40
129	8/23/2011 11:52:10
128	8/23/2011 11:50:58
127	8/23/2011 11:15:06

Included Files

Name
trunk/Demo_ALM/Model/build.xml

Related Builds

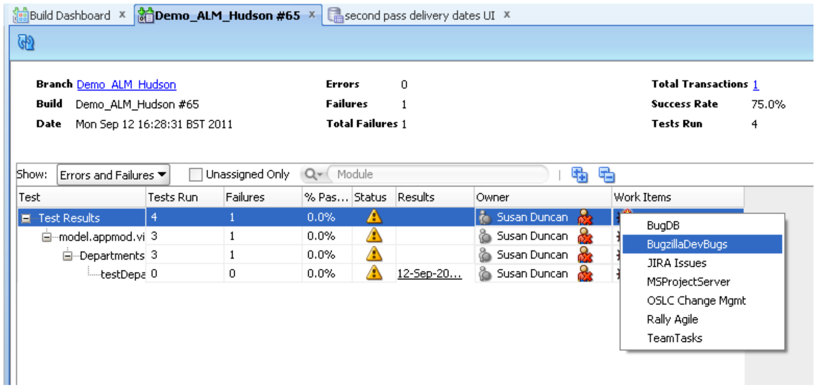
Build	Date	Status	Transactions	Tests R
Demo_ALM_Hudson #65	Mon Sep 1…	⚠	1	4
test #45	Mon Sep 1…	⚠	1	4

Figure 16. SCM and Build information integrated on the Changes tab of the work item

Finally, as seen in Figure 16, the end-to-end nature of TPC's SCM/Build integration is revealed on the Changes tab of a work item. The checkin, the related files and build/test results are integrated in one place.

Build Summary

From any entry in the build dashboard you can take a closer looks at the results. From the Build Summary you can assign yourself as the owner of tests and create a new work item to track the work you may have to do as a result of those failures. Figure 17 shows the work items repositories available to the user as a result of the test failures for Hudson build #65.



Branch Demo_ALM_Hudson
Build Demo_ALM_Hudson #65
Date Mon Sep 12 16:28:31 BST 2011

Errors 0
Failures 1
Total Failures 1

Total Transactions 1
Success Rate 75.0%
Tests Run 4

Test	Tests Run	Failures	% Pas…	Status	Results	Owner	Work Items
⊟ Test Results	4	1	0.0%	⚠		Susan Duncan	
⊟ model.appmod.vi	3	1	0.0%	⚠		Susan Duncan	
⊟ Departments	3	1	0.0%	⚠		Susan Duncan	
testDepa	0	0	0.0%	⚠	12-Sep-20…	Susan Duncan	

Figure 17. Build Summary

Administration

After installation of the server administration of Oracle Team Productivity Center is done from the administration dialog. This is found in the Administration menu as stated at the beginning of this paper.

The dialog consists of four tabs:

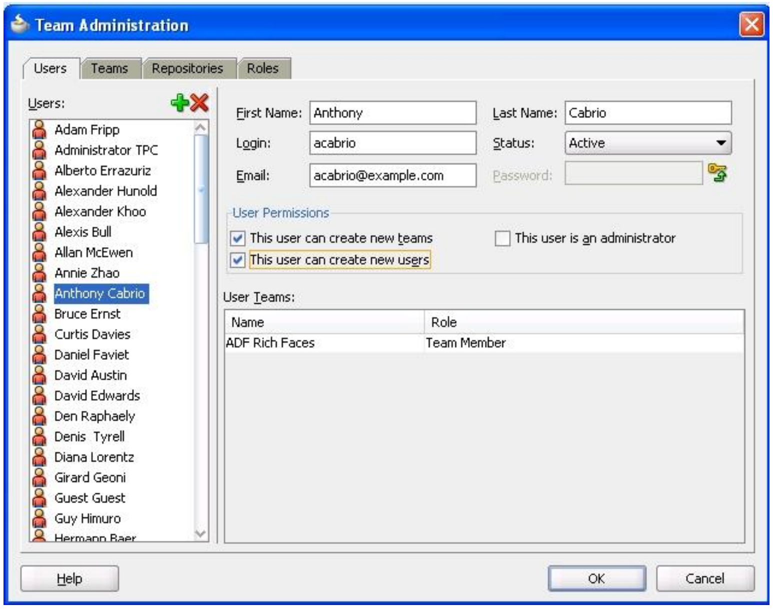
- Users – where user details are recorded
- Teams – where teams, team members and team repositories are configured
- Repositories – where ALM repository connections are setup and administered
- Roles – where roles and privileges are defined

A user's ability to access one or more these administration tabs is governed by their administrative privileges or team roles as discussed below.

Users

This tab maintains details on Team Productivity Center users. There is one user - Administrator TPC. This is the default administrator created on installation. This user will populate the first users after installation. Other users can be made full TPC administrators by checking 'user is an administrator' as shown in Figure 18

Any user who has either the 'create new users' or 'Administrator' permission can add new users to the system. The administrator may not want other users to have full administrator privileges for the system, but by implementing administrator permissions when choosing the 'Create New Users' and/or 'Create New Teams' a full TPC administrator can delegate additional administration tasks to other users.



Team Administration

Users: Adam Fripp, Administrator TPC, Alberto Errazuriz, Alexander Hunold, Alexander Khoo, Alexis Bull, Allan McEwen, Annie Zhao, Anthony Cabrio, Bruce Ernst, Curtis Davies, Daniel Faviet, David Austin, David Edwards, Den Raphaely, Denis Tyrell, Diana Lorentz, Girard Geoni, Guest Guest, Guy Himuro, Hermann Baer

First Name: Anthony
Last Name: Cabrio
Login: acabrio
Status: Active
Email: acabrio@example.com
Password:

User Permissions

☑ This user can create new teams
☐ This user is an administrator
☑ This user can create new users

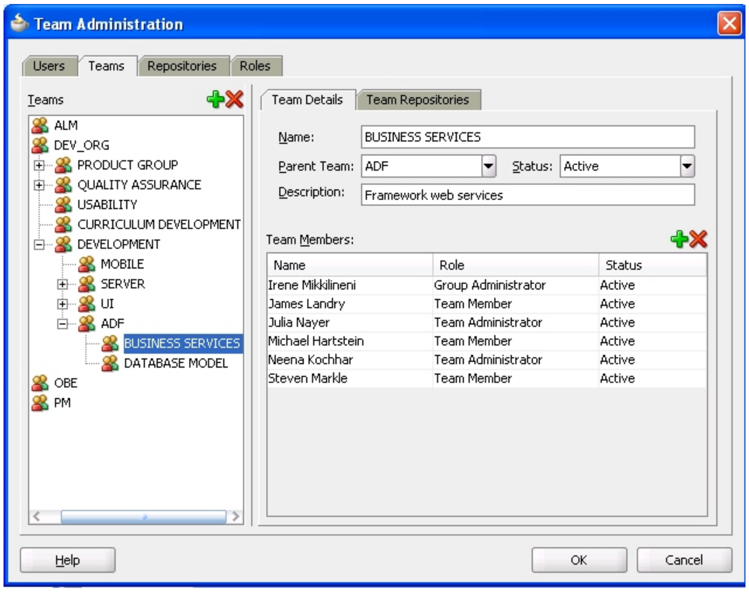
User Teams:

Name	Role
ADF Rich Faces	Team Member

Figure 18. User Administration

Teams

Oracle Team Productivity Center allows a hierarchical structure of teams to be maintained. Users can belong to one or more teams and hold a different role in each of those teams. Administrators and users granted the ‘create new teams’ permission can access the Teams administration tab (Figure 19)



Name: BUSINESS SERVICES

Parent Team: ADF Status: Active

Description: Framework web services

Team Members:

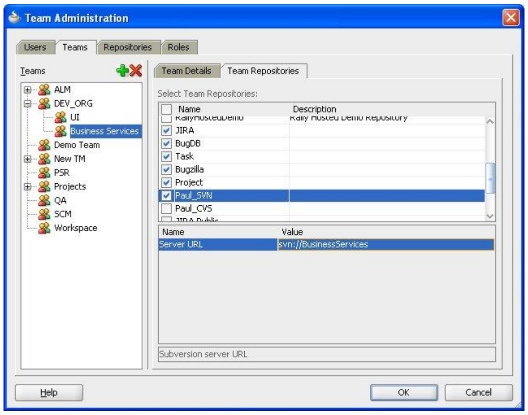
Name	Role	Status
Irene Mikkilineni	Group Administrator	Active
James Landry	Team Member	Active
Julia Nayer	Team Administrator	Active
Michael Hartstein	Team Member	Active
Neena Kochhar	Team Administrator	Active
Steven Markle	Team Member	Active

Figure 19: Team Details

The Team Repositories tab is where the repositories available to a specific team are selected. Figure 20 shows the repositories that are used by the Business Services Team.

In this example the Server URL for the team’s Subversion repository is specified. This allows teams to filter their versioning repository view in the versioning accordion (as explained earlier in this document) as required.

For work item repositories it is not necessary to specify parameters at the team level. Filtering is achieved through creating work item queries using search criteria (for example – by project, component, assignee etc.)



Select Team Repositories:

□	Name	Description
□	RallyHostedDemo	Rally Hosted Demo Repository
☑	JIRA	
☑	BugDB	
☑	Task	
☑	Bugzilla	
☑	Project	
☑	Paul_SVN	
□	Paul_CVS	
□	JIRA Public	

Name	Value
Server URL	svn://BusinessServices

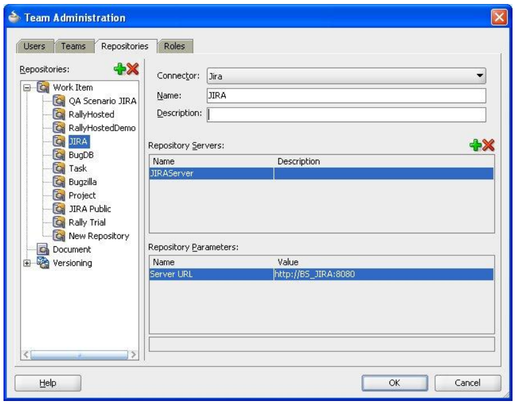
Subversion server URL

Figure 20. Team Repositories

Repositories Tab

Viewing and editing of repository connection details is restricted to full TPC administrators. On this tab, connections to ALM repositories are set up for all teams to use under the Work Items node. Figure 21 shows the connection to the JIRA repository. Confusingly, the administrator has named the connection JIRA – it would make more sense for each of these repositories to reflect perhaps their location or usage in their name, if there were multiple repositories of 'type' JIRA teams would have difficulty in choosing the correct one for their team. Oracle Team Productivity Center provides a connector that integrates a specific ALM repository type using its API. In this case the JIRA connector specifies the fields that will be contained on work items, their display type, whether they can be updated etc.

Versioning repository connections are set up in a similar way to work item repositories. But as noted above, the Server URL is not added through this repositories tab but specified at the team level.



Connector: Jira

Name: JIRA

Description:

Repository Servers:

Name	Description
JIRAServer	

Repository Parameters:

Name	Value
Server URL	http://BS_JIRA:8080

Figure 21. Repositories

Roles

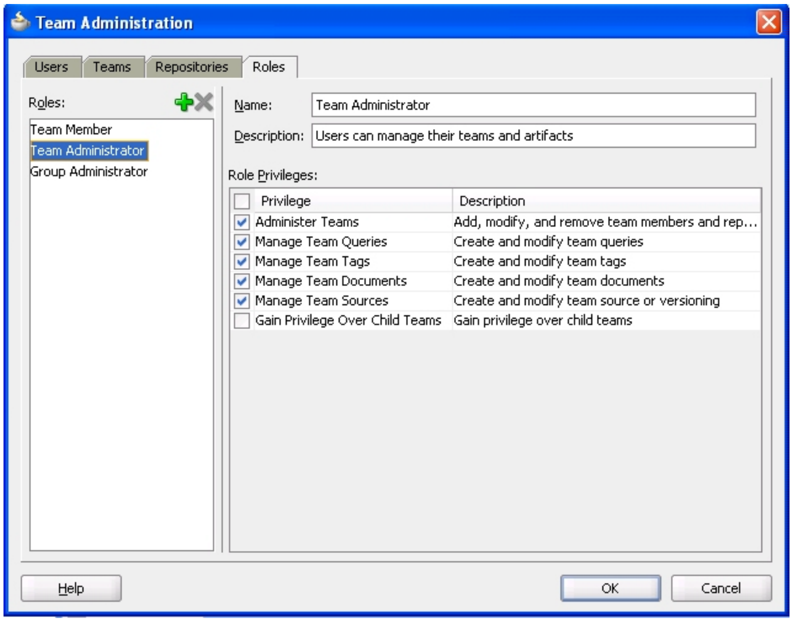
Oracle Team Productivity Center has a set of team privileges that dictate what team administration privileges a user has within their team. For instance – can a user create team level queries or tags? Privileges are applied to a team member through a role. A user may be granted a different role in different teams.

TPC ships with 3 default team roles:

- Team Member – no team level privileges
- Team Administrator - can manage all team level activities
- Group Administrator – can manage team level activities for specific team and any child teams.

New roles can be created with differing groupings of privileges according to the requirements of the TPC Administrator.

Figure 22 highlights the default privileges attached to the Team Administrator role. The Team Administrator is able to carry out administrative duties for their team including creating and managing team level queries and tags. A Group Administrator can, in addition, administer child teams under their control. By default a team member has no administrative privileges and will not be able to invoke the Team Administration dialog.



Roles:

Team Member
Team Administrator
Group Administrator

Name: Team Administrator

Description: Users can manage their teams and artifacts

Role Privileges:

□ Privilege	Description
☑ Administer Teams	Add, modify, and remove team members and rep...
☑ Manage Team Queries	Create and modify team queries
☑ Manage Team Tags	Create and modify team tags
☑ Manage Team Documents	Create and modify team documents
☑ Manage Team Sources	Create and modify team source or versioning
□ Gain Privilege Over Child Teams	Gain privilege over child teams

Figure 22. Roles

Conclusion

Oracle Team Productivity Center is an Application Lifecycle Management tool that enables software development teams to collaborate and work productively together when developing applications using JDeveloper.

Team working is enhanced through the introduction of the Team Navigator providing integration with multiple ALM repositories, allowing users to query and update work items in those repositories from within JDeveloper. Additional services such as collaboration and contextual linking of work items through tags and relationships enhance the productivity of users. Context information about development files can be saved against a work item and restored by the user.

End to end a developer can see the relationship between their work items, the SCM checkins and related files and the associated builds and tests run by their CI server in one place.

TPC maximizes investment in skills, processes and resources of teams and technologies through:

- Providing unified access to, and increasing existing value in, existing ALM tools in JDeveloper
- Maintaining quick and flexible hierarchies of teams and users leading to greater collaboration and reduced functional silos
- Integrating ALM artifacts from disparate sources resulting in better information, process flow and linkages between artifacts.

Oracle Team Productivity Center Overview
September 2011
Author: Susan Duncan

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Oracle is committed to developing practices and products that help protect the environment