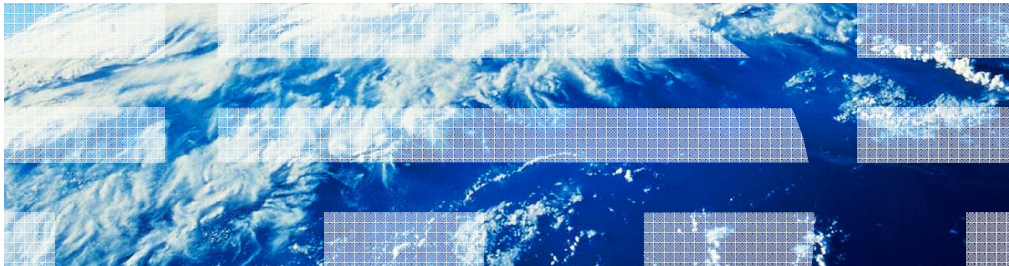


IBM Rational change and release management

Key features and improvements in ClearQuest application life cycle management



This presentation covers the key features and improvements in the application life cycle management (ALM) packages for ClearQuest®.

ALM key features (1 of 4)

- **Migration in place**

- If you are using ClearQuest, you may want to modernize your ClearQuest schema
- ClearQuest application life cycle management packages can be added to your existing schema with immediate improvements
 - Performance
 - Process flexibility
- Earlier ClearQuest data is still available

With ALM for ClearQuest, you have the option of migration in place. An example of this would be taking a database of a flat file record and adding the ALM packages on to it. You then make the existing data read only, and have end users to submit ALM requests. This would be done along side so they had their old data available.

ALM key features (2 of 4)

- **Diverse workflows**
- ClearQuest ALM is a third generation approach to ClearQuest schema design.
 - Generation 1: Monolithic records
 - Generation 2: MultiSited monolithic records
 - Generation 3: MultiSited parallel role-based records
- With ClearQuest ALM, bringing all areas of your organization into one process no longer means everyone must follow the same workflow
 - Workflows (Practices) can vary widely within a Project and across multiple Projects without losing general accountability of progress
 - Component Based development, even tracking in combined Hardware and its embedded software with their differing workflows is supported

ClearQuest ALM operates on the idea of diverse workflows, which is a third-generation approach for ClearQuest schema design. Monolithic records was the first generation. An example of this is a Defect record in the DefectTracking schema. Monolithic records across multiple sites was a second generation. Multisited, parallel, roll-based records are what ClearQuest ALM is about. You can bring all the areas of your organization to one process, meaning that not everyone must follow the same workflow. You will notice that there is a significant difference in the philosophy of bringing people together for a particular process.

ALM key features (3 of 4)

- **Incremental Adoption of Jazz™-based technology**

- ClearQuest ALM architecture was used as a model for the design of the Rational Team Concert™ Bridge to ClearQuest
- Now with ClearQuest ALM and OSLC, you can take advantage of the power and flexibility of ClearQuest using your embedded knowledge base while supporting the early adaptors of Rational Team Concert and Rational Quality Manager
 - All without losing the ability to track development in:
 - ClearQuest
 - Rational Team Concert
 - Rational Quality Manager
- The decision no longer needs to be an 'All or nothing' adoption of Jazz-based technology

Also implemented is the incremental adoption of Jazz-based technology and tasks as a planning mechanism. With Open Services for Life cycle Collaboration (OSLC), you can generate work items into Rational Team Concert. An example of this would be the transferring of Defects from Rational Quality Manager. Previously, you might have relied on a synchronized implementation before availability of OSLC. The goal here is to avoid having to take an “All or Nothing” approach to adopting Jazz technology. You can move forward with the early adoption of Jazz technology such as Rational Team Concert. You do not have to lose the rest of your base or go through a cathartic change in order to do it.

ALM key features (4 of 4)

- **Scalable agility**
- Three dimensions of scalability:
 - Functional scalability: ALM can be integrated with Rational Team Concert, Rational Quality Manager, DOORS® (using OSLC), ClearCase® UCM, and Build Forge®
 - User and record volume scalability: ALM is designed to support Large and small scale ClearQuest MultiSite for geographically distributed development and web-based ClearQuest single site
 - Process scalability: Agile, normal and well-governed ClearQuest ALM models are supported

ClearQuest ALM supports three dimensions of scalability: functional scalability, user or record volume scalability, and process scalability.

Functional scalability means you can use loosely coupled applications. You can use the applications and you can grow, in that you can be doing more than just tracking items. You can introduce requirements, testing, and content management.

Then there is user and record volume scalability. ClearQuest ALM is designed to work as small records. It divided functionality instead of putting all on one record. For example, with the traditional change request record, there was numerous, advanced, behind-the-scenes hook code. This was designed and constantly conditioned to take on all possibilities for all types of users at any given time. With ClearQuest ALM, that is now broken into different record types. Now, you are able to do parallel functionality. Additionally, this can reduce the amount of time it takes to work with records because of reduced resource strain. This is especially noticeable when working with records on ClearQuest sites across long distances.

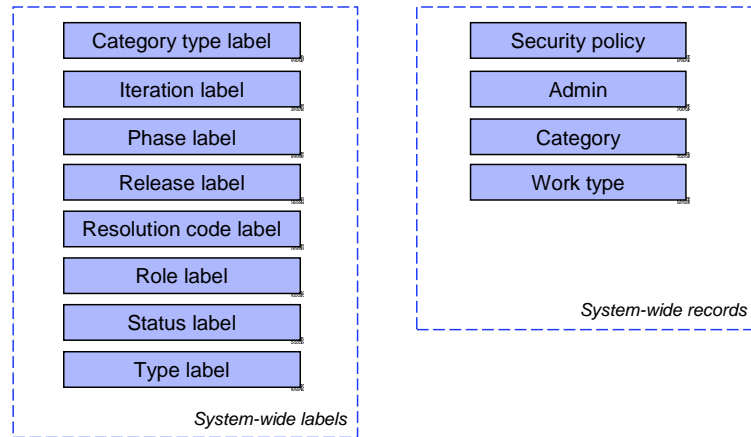
In regards to process scalability, ClearQuest ALM introduces “upper and lower tiers” and “1 and 2 record models”. See slide 9 for more information on this that process.

Core concepts for ClearQuest ALM

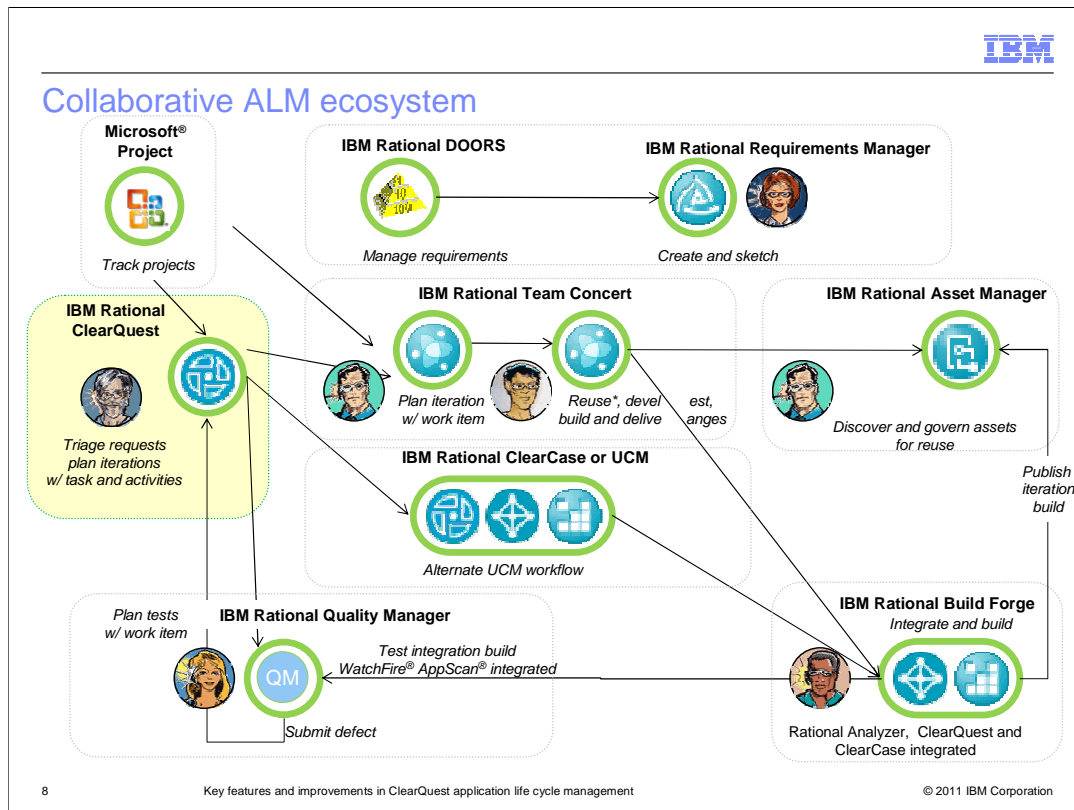
- Projects provide security, roles, and a context
- Work is completed in the context of a project
- There are system-wide settings that you can modify without having to change the schema

ALM Projects provide security roles and context for work. Work is completed in that context. There are system-wide settings that you can modify without having to change the schema.

System-wide settings

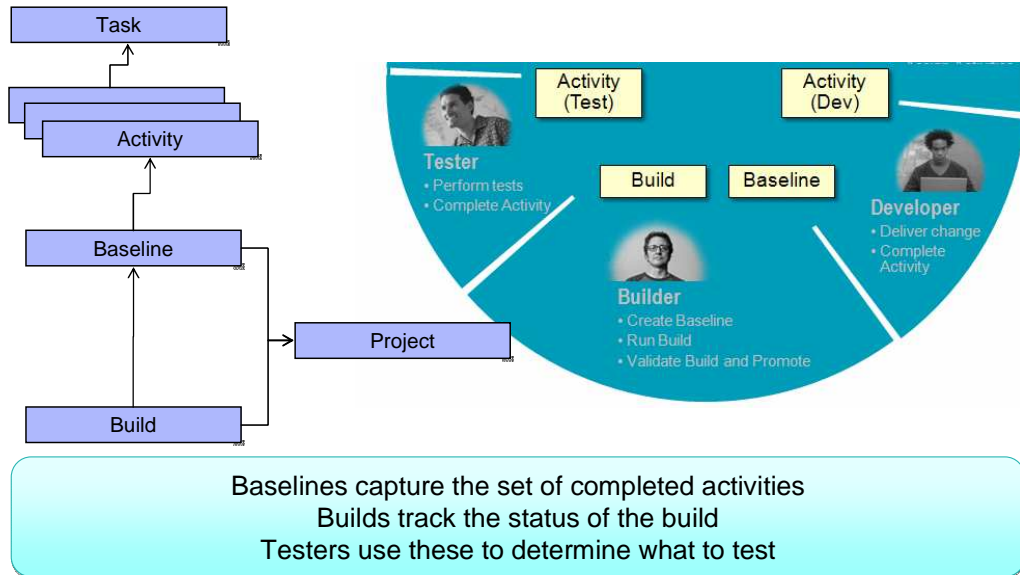


Labels let you use your vocabulary across the enterprise
System-wide records facilitate re-use and consistency across projects



This image is an overview of some of the functionally interactions that is supported with ClearQuest in a collaborative, ALM ecosystem. Requirements are developed and managed by DOORS and Requirements Manager. Defects and enhancement requests are managed by ClearQuest and Rational Team Concert. Testing is managed by Rational Quality Manager. BuildForge integrates with ClearQuest ALM and Rational Team Concert to automatically create Builds. The connection lines represent OSLC Links.

Summary: Activities, builds, and baselines enable team collaboration and governance



9

Key features and improvements in ClearQuest application life cycle management

© 2011 IBM Corporation

This is an graphical representation of the upper and lower tier, one and two record model. With the one record model, you can perform all your functionality with just the ALMAActivity. The ALMProject record has a default ALMRequest and a default ALMTask. The ApprovedActions for the ALMAActivity roles include *Submit* for developers.

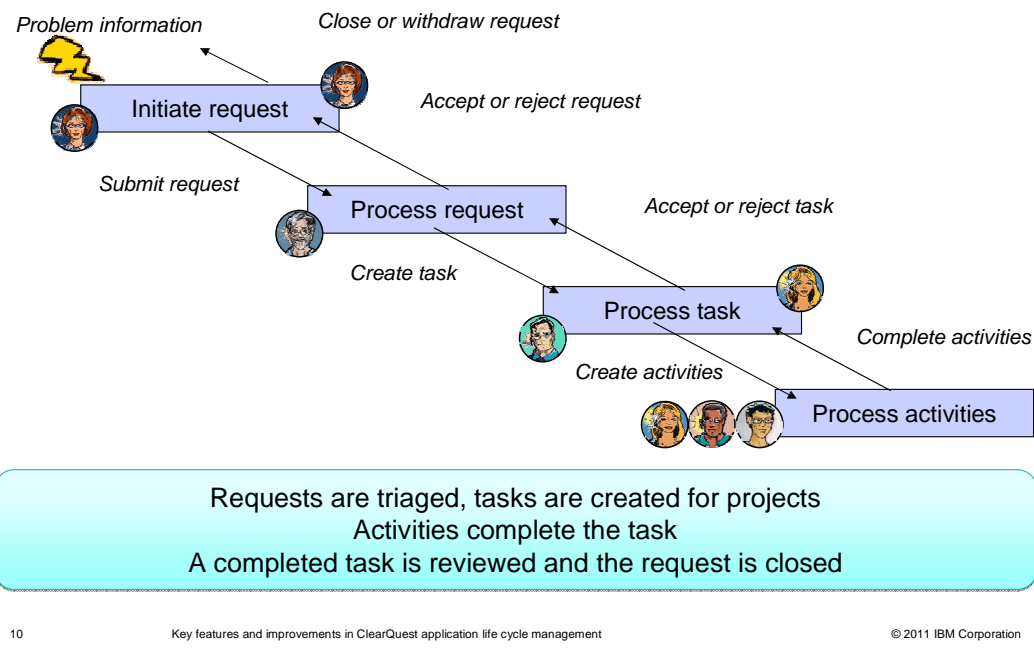
For example: After clicking the SetDefault Utility for the ALMProject, which you can do once per session, developers perform a ClearQuest Submit ALMAActivity on the fly. This happens when they perform their own checkout, add to source control, or perform other UCM activities.

With the lower tier, two record model, you might invoke a Change Control Board for increased governance as you get further into a project. The ALMProject record has a default ALMRequest, but no default ALMTask. Additionally, the ApprovedActions for ALMAActivity roles do not include *Submit* for Developers.

For Example: After clicking the SetDefault Utility for the ALMProject, which you can do once per session, developer leads Submit ALMTasks. Then they perform a *CreateActivity* to create ALMAActivity records. Developers can only modify or transition the state of the activities that have been assigned to them.

Baseline and Build records are supported. That functionality exists to help shadow the reality of creating a baseline and build. The baseline might be just a gathering of code for those who are not using the UCM integration. The UCM integration is often a better solution for identifying fixes and when they are provided. UCM is supported with ALM, but it is not required. Either situation is supported. Support also exists for composite baselines. You can create a composite baseline that includes elements baselines developed outside or inside of the UCM integration.

The flow of request, task, and activity



This is an graphical representation of the flow for requests, tasks, and activities in ClearQuest ALM. The Request is processed by creating one or more Tasks. The Tasks are processed by creating one or more Activities.

The Activities are eventually completed. The Task is completed when sufficient Activities have been completed. The Request is accepted as complete when the Tasks have been approved.

Updates with ClearQuest ALM records starting in ALM 1.1 (1 of 2)

Items that ALMAdmins should be aware of when upgrading to ALM 1.1.

1. The **CreateTask**, **Reject_Request**, **Unreproducible**, and **WorksAsDesigned** actions for **Request** records must be added to an **ALMRole** and **ALMRoleLabel**
2. For the **ALMProject** record, **SetDefault** is no longer needed in any **ALMRole**
3. All **ALMWorkConfigurations** must have a **PrimaryRole** specified before using the **CopyProject** action for the **ALMProject**
4. At least one **ALMProject** must be specified with its **AssignedToAllRejectedRequests** check box checked

Here are some changes that ALMAdmins should be aware of when upgrading to ALM 1.1 or later versions.

The **CreateTask**, **Reject_Request**, **Unreproducible**, and **WorksAsDesigned** actions for **Request** records must now be specifically added to an **ALMRole** and **ALMRoleLabel** for anyone expected to be able to perform those commands.

For the **ALMProject** record, **SetDefault** is no longer needed in any **ALMRole**. Everyone has that by default. There is no need to remove it from the **ApprovedActions** list for **ALMRole** and **ALMRoleLabel**, but also no need to add it either.

All **ALMWorkConfigurations** must have a **PrimaryRole** specified before using the **CopyProject** action for the **ALMProject** record.

At least one **ALMProject** must be specified with its **AssignedToAllRejectedRequests** check box checked. You cannot use **CopyProject** on this **Project** without first un-checking this check box. After the **CopyProject**, you can recheck it.

Updates with ClearQuest ALM records starting in ALM 1.1 (2 of 2)

5. If you have marked the 'ALL' **Project->Category** Obsolete, you will need to uncheck **Obsolete?** in **ALMCategory** before modifying the **ALMProject**
 - This is the result of how obsolescing works starting in ALM 1.1
6. Viewing **Priority** choices for **ALMRequest** and **ALMActivity** records require new **ALMType** records
7. All **WorkConfigurations** must have a **PrimaryRole** for all **Task** and **Activity** types for the **CopyProject** action

If you have marked your project's "ALL" category as Obsolete, you will need to uncheck the Obsolete check box for the ALMCategory record before modifying your ALMProject. Otherwise, the "ALL" Category is not seen as an option on that Project and therefore you cannot set the AssignedToAllRejectedRequests check box. This is because of how Obsolescing works starting in ALM 1.1.

You can update your "ALL" Project to be in the AssignedToAllRejectedRequests state. After doing this you can once again modify the Obsolete check box in ALMCategory so that the "ALL" Category is not a nuisance choice to your users. The "ALL" Project will work properly so long as you do not modify it or transition its state. If you do need to change the "ALL" Project, repeat this process. In order to see Priority choices for the ALMRequest and ALMActivity records, you need to add new ALMType records. Additionally, all WorkConfigurations must have a PrimaryRole for all Task and Activity types if you intend to perform a CopyProject action.

Changing the project of a task (1 of 3)

- You want to change a task when:
 - Work does not always get done in the release for which it was targeted
 - Activities might or might not be works in progress, with and without Change Sets
- Starting with ALM 1.1 you can change the **Project** to which your **Task** is associated, even if Activities already exist

You have the ability to change which project a task is associated with starting in ALM 1.1. There is functionality provided that allows you to get the project from the task, so that only one click is necessary to update the project functionality by refreshing the task.

Changing the project of a task (2 of 3)

- **Activities** associated with a **Task** whose **Project** has changed have four main scenarios.
 - If the **Activity->Project->UCMProject** value is NULL
 - **First scenario:** Following **Task->Project** change, Modify the record and then click **Activity-> Get Project From Task**. The **Activity->Project** is changed to match **Task->Project**
- Note:** Know if your database is replicated. Multisite sync replication might first need to occur if **Task** is modified on a different ClearQuest site than the **Activity**

There are some restrictions to changing the Project of a Task.

If there is a UCM project involved, in which something has already been delivered to an integration stream, you cannot change a Project on the Task or Activity. That is because it can cause a cognitive dissonance flow problem. This restriction exists starting in ClearQuest ALM 1.1.

Changing the project of a task (3 of 3)

- If **Activity->Project->UCMProject** value exists, and is different from the **Task->Project->UCMProject**
 - **Second scenario:** If there is no **CC ChangeSet** on the **Activity**, you can click **Activity-> Get Project From Task**. **Activity->Project** will change to match **Task->Project**
 - **Third scenario:** If there is **Activity->ucm_project** that is different from the **Activity->Project->UCMProject** you cannot save **Activity** changes
 - **Note:** You can change the **Activity->ucm_project** to match. You can then save the **Activity**
 - **Fourth scenario:** If there is a **CC ChangeSet** on the **Activity**, you cannot change the **Activity->Project**

For any of these other cases, you are able to change the tasks. When integrated, it is advised to complete all Activities with ClearCase ChangeSets before changing the UCMProject for an ALMProject.

Before and after: Changing the project for a task on non-completed activity

Activity | Related Records | Resolution | Baseline and Build | History | Comments | Attachments | Keywords | Unifi

Headline:

Project:

ID	Name	Category	Release	Obsolete
CQALM000...	Project_T...	Category_ToDo	ReleaseLabel0...	No

Activity | Related Records | Resolution | Baseline and Build | History | Comments | Attachments | Keywords

Headline:

Project:

ID	Name	Category	Release	Obsolete
CQALM000...	ProjectAlt...	Category_ToDo		No

Here is a before-and-after view of the changing of the Project of a task for a non completed activity. Note that you cannot change the task if the activity has been completed.

Role support for requests (1 of 2)

- **Owner**
 - User responsible for the **Request** and only person other than **ALMAdmins** exempt from **Role** restrictions for this **Request**
- **Submitter**
 - User who created the **Request** (for example, Scribe in a brainstorming session)
- **Originator**
 - User who initiated the **Request**

ALM has a variety of role support options for Requests starting in ALM 1.1. Roles are optional on a WorkConfiguration for a Request.

The Owner of the Request does not need to be defined in the role for a Request he or she owns. The ALMAdmins record type can be used to define users that are exempt from role restrictions. In any situation, no one can avoid base ClearQuest security context restrictions.

The Submitter is someone who created the request. The Originator is someone who logically initiated the request, but did not technically create it in ClearQuest ALM. The Submitter value is automatically set, and cannot be changed manually.

Role support for requests (2 of 2)

- No **WorkConfiguration->PrimaryRole** able to be set because Owner is set to **Submitter** by default and is able to be changed
- **Role->ApprovedActions** now needed on **Request Roles** for non-exempt users
 - *CreateTask*
 - *Reject_Request*
 - *Unreproducible*
 - *WorksAsDesigned*

The Owner is someone who is responsible for processing the Request. The Owner and Originator for the record can be manually set, and can be the same person or different persons as needed.

Role approved Actions are needed for non exempt users. For example, to create a task for the Request, you need to be approved for the CreateTask action.

SecurityPolicy is set for all BTBuild records

- If **BTBuild** records from Deployment Tracking package are used outside of ALM, they need a **SecurityPolicy** to be viewable by non-security admins
- ClearQuest ALM automatically sets **BtBuild->SecurityPolicy** to “Everyone”
- All Actions automatically approved for **BtBuild** if **BTBuild->ALMProject** field has no value
- In ClearQuest 7.1.0.2 and later, **SecurityPolicies** with **ratl_context_groups** can work correctly when referencing the “Everyone” ClearQuest group

You might have used the BTBuild record from the ClearQuest Deployment Tracking package before you applied the ClearQuest ALM record to the BTBuild record. In that case, your BTBuild records will become invisible to all but ClearQuest Security Admins, since ALM adds SecurityContext to the BTBuild record.

Starting in ALM 1.1, BTBuild records are automatically set with a SecurityPolicy group of “Everyone”. With the group set, everyone can see the BTBuild records, as long as they are not associated with ClearQuest ALM. Having an “Everyone” group for security policy is a requirement for ALM.

Category drilling for requests (1 of 2)

- **ALMRequest** Drill Down button will now stop at bottom of tree instead of drilling past the bottom
- The bottom-most **Category** is not added to the **CategoryPath** control (Sorted by sequence of **Category->Name**)
- The bottom-most **Category** is not added to the **CategoryStringPath** control (Sorted by sequence of Drill Down selection)
 - Queries will need to use the OR operator to look in the Category or CategoryPath fields to cover references to any given **Category**.
- The Drill Back and Clear Path buttons can drill back to the top

Starting in ALM 1.1, the Category Drill down is updated. Using the Drill Down button for a Request will navigate down the tree and stop at the bottom of it. The bottom-most category is not added to the CategoryPath, but it is available in the Category field. You will have to use an OR operator if you want to query for it in the CategoryPath or in the Category.

For Example: Where CategoryPath Contains 'xyz' **OR** Category Contains 'xyz'.

There is also a drill back button you can use to drill back to the top one layer at a time. The Clear Path button is still available, which drills all the way to the top.

Category drilling for requests (2 of 2)

Request | Project | Related Records | History | Comments | Attachments | Keywords | Subscriber List

Request's Found-in Category Context:

Category Type (Filter):

Category: SubSubCategory

Category Path (Ref):

Name	Category T...
AfterSubCateg...	SubCategory
Category_ToDo	Offering
SubCategory_...	SubCategory

Category Path (String):

[Category_ToDo]->[Offering]
 [SubCategory_ToDo]->[SubCategory]
 [AfterSubCategory]->[SubCategory]

There are two fields available for the categories. There is a Category Path reference, and a Category Path string. The string field shows you the logical sequence of the category drill down, instead of the alphanumeric sorting of the categories in the reference field. With the string field, as you drill up or back you will see the layer removed from the last choice, not from the last alphabetic choice. Note that Sub-Sub Category is not listed in either of the CategoryPath fields.

Batch copy scripts provided (1 of 2)

- Batch CopyEntity script has been provided to clone multiple whole Projects in one executable.
 <common directory>\Packages\ALMWork\1.1\pkg_util\alm_copyproject.pl
- Partial Batch Copy script provided:
 <common directory>\Packages\ALMWork\1.1\pkg_util\alm_syncprojects.pl
 - You can script to propagate updates to **Projects** that have been cloned from the changed **Project**. Specify the Source and Target **Project** one at a time

Batch copy scripts are provided with ClearQuest ALM. A batch copy entity script is included with the Packages directory of your product installation. You can copy and paste everything into a batch file of some kind, and use that to create several copies of a source project at one time. There is also a partial batch copy script provided called ALM_SyncProjects. If you add a Phase, Role, or Work Configuration to a source project from which you have cloned a target project, you can run that utility. If you have cloned projects, you can script to propagate updates to Projects that have been cloned from the changed Project. Specify the Source and Target Project, and update one Project at a time.

Batch copy scripts provided (2 of 2)

- Partial batch copy script (continued):
 - A batch file can reference the sync script to facilitate making changes to multiple **Projects** at a time
 - Changes to the **ALMPhase**, **ALMRole**, or **ALMWorkConfiguration** for a Project are cloned to each specified **Project**
 - Single location to make the change, it is propagated from there

Example:

```
ratlperl alm_syncprojects.pl -projadmin ProjectAdmin_ToDo -pw "" -dbname  
CQALM -dbset CQMS.ALMEML.Boston -fromprojid CQALM00000095 -  
toprojid CQALM00000130
```

It is possible to compile a batch file that references the script with the source and target projects multiple times to facilitate making changes to multiple Projects in one execution. If there is a change to a setup in the Project, for instance a new ALMPhase, ALMRole or ALMWorkConfiguration, these changes are cloned to each specified Project. There will now only be a single location to make the change. Changes are propagated from there.

Extended component base development support (1 of 2)

- You can take the **CurrentProject** value of a **Category** from the lowest level of **Category** rather than the highest
- The previous approach with ClearQuest ALM:
 - Only an **ALMCategory** with no **SuperCategories** is chosen as the **Category** for the **ALMRequest** record.
 - The **CurrentProject** is automatically set as the **Project** for **ALMRequest**

Component based development support is extended starting in ALM 1.1. You can have a Request with one or more Tasks.

An archival example of this: Someone reports a problem with a product offering. Upon triage of the problem, it is determined that the problem is not in the offering but it is in the component consumed by the offering. This results in the generation of a task for that component. Additionally, you generate another task for the next release of that product offering. That second task exists because once the component has been fixed, at that point it can be included in the new version the product offering.

Previously in ALM, you can only choose a CurrentProject from a Category that was at the top of the category tree.

For example: If you have a request record, you can drill down into various categories. If the Category has a current project, it will automatically set that project when you select that Category. Before ClearQuest ALM 1.1, you had to be at the top of the tree in order to take advantage of that functionality.

Extended component base development support (2 of 2)

- The **CurrentProject** of the **Category** is used to set the **Project** for the **ALMRequest**
 - If using the **Drill Down** button, a **SubCategory** is chosen, and the **Category** has a **CurrentProject**, it overrides the **CurrentProject** for the **SuperCategory**
 - The more specific the **Category** is, the more specific that **Category->CurrentProject** will be
 - The most specific, smallest set of **Members** and **ALMRole->Primary Member** values is set. This provides better support for Component Based development

There is another change in behavior starting in ALM 1.1 as you drill down. If any of the sub categories or sub-sub categories have a current project, it will replace the current project set by a Category higher in the Category hierarchy. This happens regardless of whether a Category has any SuperCategories.

The advantage of this is that roles and other records are associated with projects. So, the further you drill down, the more specific you get. The more specific you get, the more you can have a role associated with a more appropriate number of users. For example, you might have a role associated with a component development team of a dozen users, opposed to having a role end up with hundreds of associated users.

Also, there is now an **PrimaryRole** field for **ALMWorkConfiguration** that supports designating the Role whose **PrimaryMember** is used to set the Owner for a Task or Activity automatically. Before ALM 1.1, if an **ALMWorkConfiguration** referred to more than one **ALMRole**, the Role which sorted first alphabetically was used to determine **PrimaryMember** and thence, default Owner.

Categories can have a **CurrentProject** or not. If a Category is drilled-down, and it has no **CurrentProject**, the **Project** is not set. If the Category does have a **CurrentProject**, that is used to set the **Project**.

Type fields for priority and severity are project dependent (1 of 2)

- Contained as Reference_list fields on the **Project** record's **WorkConfiguration** Tab
- The **Request->Severity** and **Task->Priority** choice lists are derived from the **Project's Requests_Severity_Choices** and **Tasks_Priority_Choices** fields
- If the **Requests_Severity_Choices** and **TasksPriority_Choices** fields are not set, the system defaults to the method of pulling all **ALMTypes** in the system
 - **Note:** These settings do not pertain to the new **ALMRequest->Priority** or the **ALMActivity->Priority**

Starting in ALM 1.1, Severities and Priorities can be set on the ALMProject record. Before this release, these two fields applied to all Projects. Now you can override the values used by setting them on the ALMProject records. The Severity field on the Request record and the Priority field on the Task record type are choice lists that are derived from the Project's Requests_Severity_Choices and Tasks_Priority_Choices fields.

If these fields are not set, the behavior from the original ALM release will take effect.

Type fields for priority and severity are project dependent (2 of 2)

Project | Plans | Team Members | Related Projects | ALMWork Configurations | Workflow Automation | History

Default Parents :

Default Request:

Project ID	Request ID	Type	State	Owner	Headline
CQALM000...	CQALM000...	Defect	Completed	admin	Default Request

Default Task:

Project ID	Task ID	Type	State	Owner	Headline
CQALM000...	CQALM000...	Defect	Completed	DevLead_T...	Default Task

Work Configurations:

Record Type	Type Label	Obsolete
ALMActivity	Dev	NO
ALMActivity	Doc Assess	NO
ALMActivity	Test	NO
ALMActivity	Sizing	NO
ALMActivity	Planning	NO
ALMRequest	Defect	NO
ALMRequest	Enhancement	NO
ALMRequest	New Functionality	NO
ALMTask	Defect	NO
ALMTask	Enhancement	NO
ALMTask	New Functionality	NO

Choice Lists Configurations

Requests Severity:

- ALMRequest Sev2 - High Impact Severi
- ALMRequest Sev4 - Low Impact Severil

Tasks Priority:

- ALMTask Unassigned Priority
- ALMTask P3 - Moderate Priority

27 Key features and improvements in ClearQuest application life cycle management © 2011 IBM Corporation

Here is the Work Configuration Tab for the project. Notice the Severity and Priority settings in “Choice Lists Configurations” area.

Miscellaneous updates (1 of 3)

- Description boxes on **ALMRequest** now have vertical scroll bars
- The SetDefault action for ALMProject is allowed
- More information is displayed on the **Comment** form:
 - Record identifier for record being referenced (ID or DBID)
 - **Author**
 - **Subject**
 - **To** and **From**

The description field on the ALMRequest record has vertical scroll bars for improved usability.

You can now set the default action for your project. With that capability you can choose a product and choose default action. The project will then be automatically set. If you have set the default previously, you will now be able to open a new project and put it into to default mode. From there all the projects associated with the default will appear, leaving you with only the headline to enter.

Commenting is supported throughout the ClearQuest ALM system. Anyone can click Utility for the Request, Task, or Activity, and select the QuestionOrComment Action. This allows you to add a Comment to that record. Anyone can also click the Respond Utility action that Comments to respond to a question or comment.

The comment form is more informative by putting the identifiers for various records, whether it is a request, a task, an activity, or a response to another comment. You can also view author, subject, to, and from information in the comments.

Miscellaneous updates (2 of 3)

dbid: 33557279 *Security Policy: [dropdown]

To: [Request] [Task] [Activity] [Comment]

Author: [dropdown]

*Subject: [text box]

Response Requested?

Question/Comment [text area]

id	Headline

Add Remove

Template: [dropdown] Load

OK Cancel

Here is an example of the comment record.

Miscellaneous updates (3 of 3)

- New fields in the **ALMActivity** record type:
 - **Phase**
 - **Iteration**
 - **EstimatedEffort** (int type)
 - **ActualEffort** (int type)
- The **TypeLabel** choice list for **ALMWorkConfiguration** is sorted.
- **Keywords**, a reference list field, has been added.

Here is a list of additional changes that require at least ALM version 1.1.

The ALMActivity record has new fields: Phase, Iteration, EstimatedEffort, and ActualEffort. These fields aid in Agile development. The effort fields use integers that you provide. There is no unit specification, so make sure you understand the unit you are specifying. A universal understanding is important to avoid users setting different meanings or different time measurements.

The TypeLabel for ALM WorkConfiguration records are now sorted. A Keywords field is also available. It is a reference field that you can use to create Keywords and reference them on your records.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_RCQ_ALM.ppt

This module is also available in PDF format at: [./RCQ_ALM.pdf](http://RCQ_ALM.pdf)

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, AppScan, Build Forge, ClearCase, ClearQuest, DOORS, Jazz, Rational, Rational Team Concert, and WatchFire are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2011. All rights reserved.