Rational ALM Strategy and Directions

Mats Göthe (mats.gothe@se.ibm.com)

HSBC ALM Conference. Aug 2007

Rational Cross-Product Green Threads

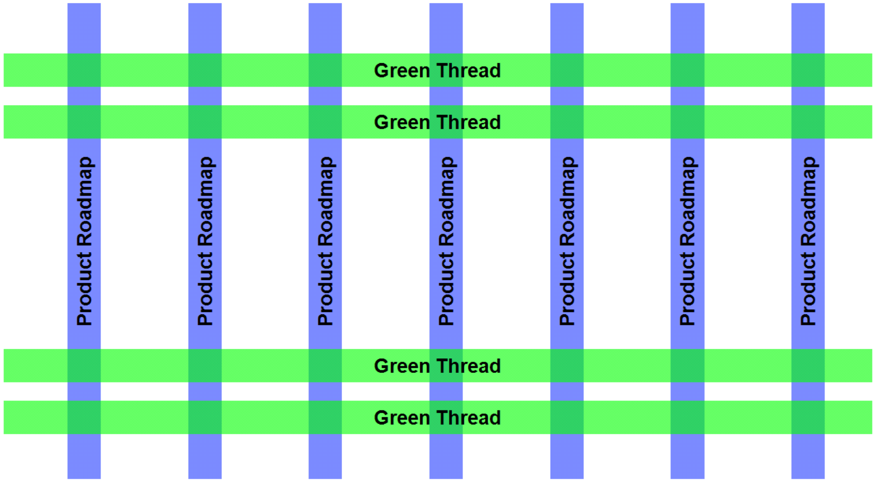
Definition

- Workflow across a set of products to solve a complex customer problem

Goals

- Move beyond point product thinking
- Deliver solutions that solve real-world customer problems
- Identify and prioritize key issues and roadblocks to product development
- Provide guidance and recommendations for customers to use

From Product-Centric to Solution-Centric

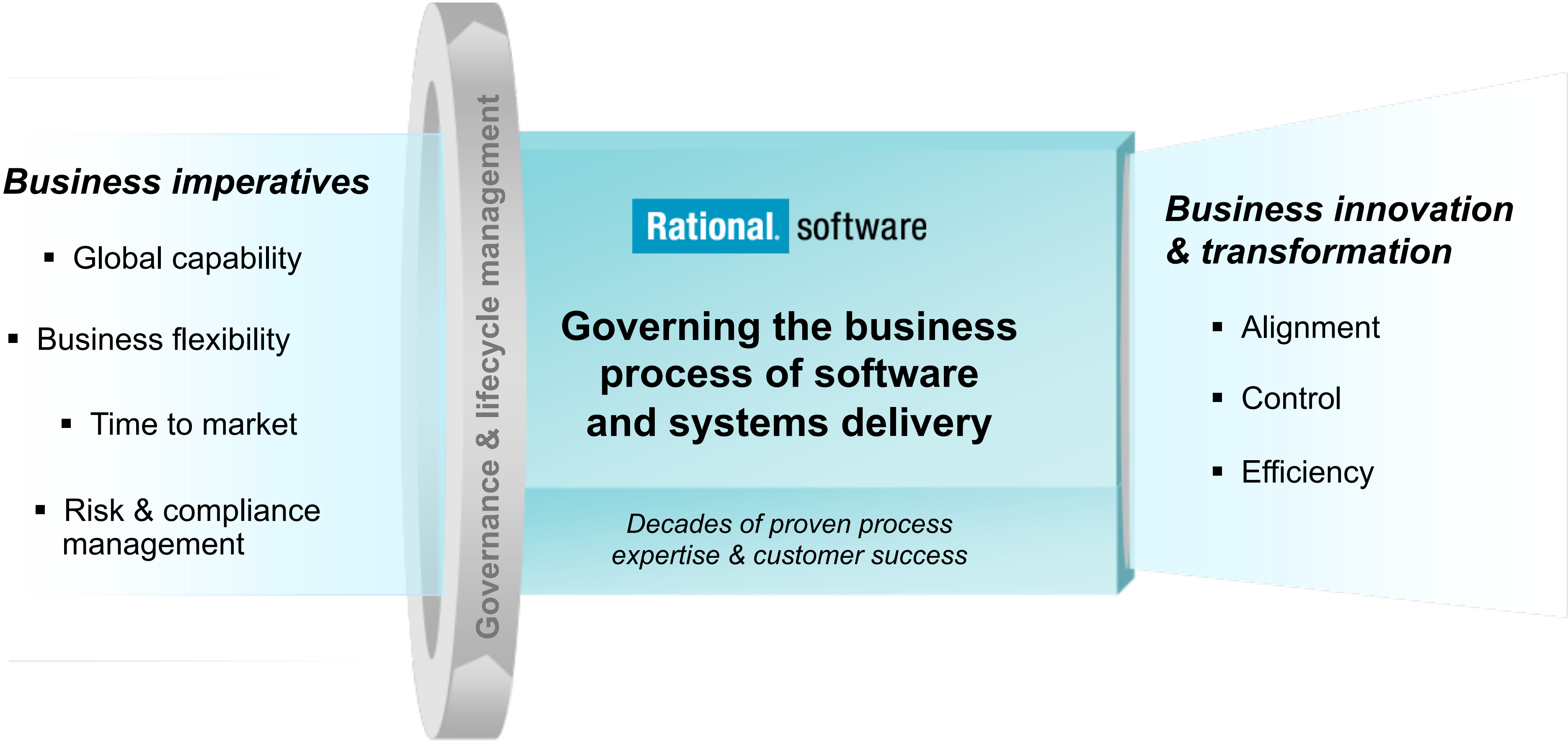


Organization

- Reporting to VP Development
- Collaborative effort with product and offering management, marketing, engineering, and field teams
- Focus on Application Lifecycle Management, Enterprise Architecture and Software Development Governance

IBM Rational software

A strategic partner in business innovation & transformation



Business imperatives

- Global capability
- Business flexibility
- Time to market
- Risk & compliance management

Governance & lifecycle management

Rational. software

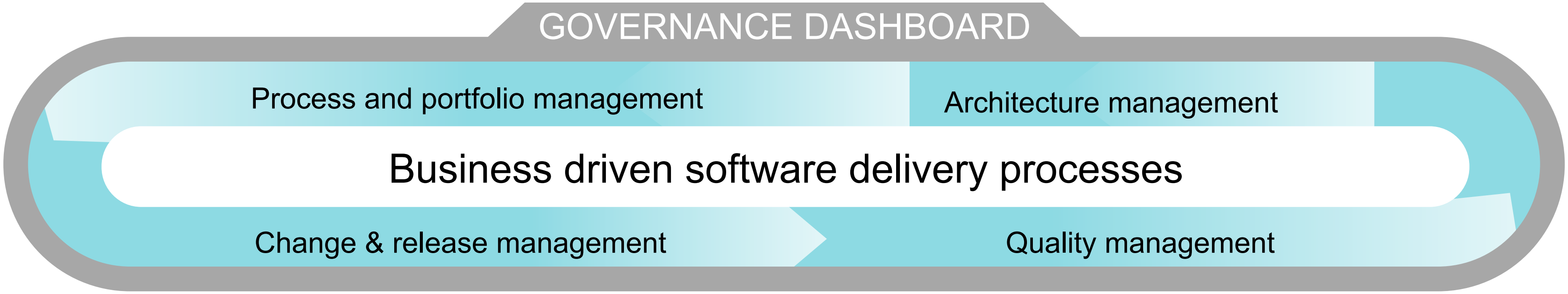
Governing the business process of software and systems delivery

Decades of proven process expertise & customer success

Business innovation & transformation

- Alignment
- Control
- Efficiency

Collaborative, Process-Driven Lifecycle Management



GOVERNANCE DASHBOARD

Process and portfolio management

Architecture management

Business driven software delivery processes

Change & release management

Quality management

Manage value

- Real-time analytics linking financial and software information
- Real-time resource management
- Comprehensive, flexible dashboards monitoring health of projects

Development flexibility

- Proven best practices
- Integrated requirements & architectural mgmt
- SOA design and construction capabilities
- Open, collaborative role-based team environment

Control risk and change

- Lifecycle change and asset management
- Built-in audit and status information on projects and assets
- Functional & compliance quality management
- Service-level monitoring

Pillars of ALM

- Governance
  - … govern projects on live metrics in context of established plans traceable to original business goals and strategies
- Collaboration
  - … establish role based, contextual collaboration among organizationally distributed networks of stakeholders and contributors
- Traceability
  - … reduce impact and risk of change – by understanding the relationships across versions, variants and releases of assets, where change to one impacts others
- Process Automation
  - … improve performance, compliance and quality by automating development process workflows and leaving an audit trial of what occurred
- Distribution and Integration
  - … support flexible and agile business sourcing decisions by integrating distributed roles, teams, workflows, and asset repositories into a responsive software delivery chain

ALM Philosophical Principles

- Development is not an island unto itself; rather, we provide a service to the business
- Solutions never die, they are refined & maintained for years
- Software Solutions are the product of many Conversations
- There are cycles within ALM; some of these cycles are ripe for automation
- Simplicity First

The development focus has shifted

from the Individual…

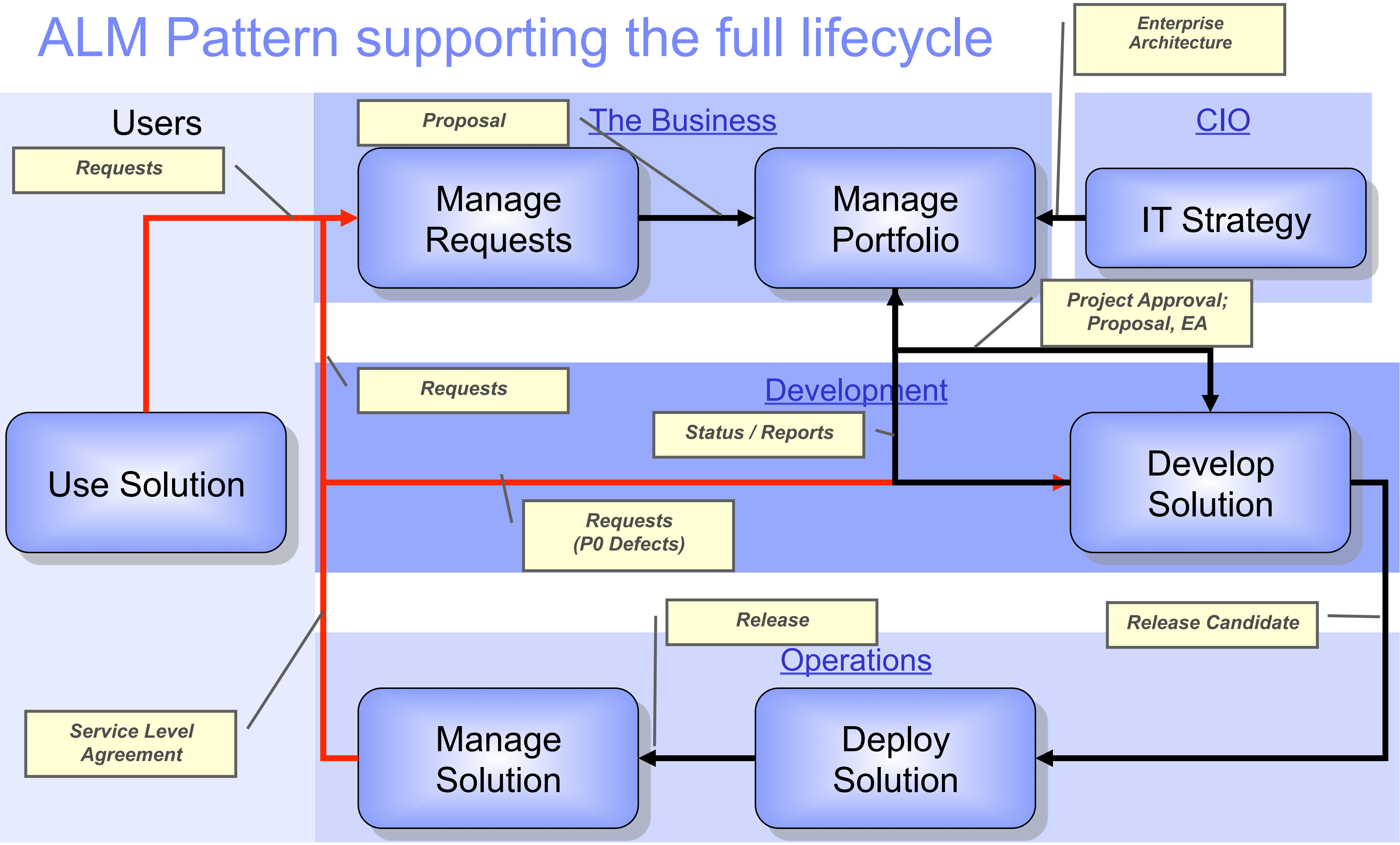
- Remember late 90’s
  - Focus on “point tools”
  - Who can build the best Java IDE, the best C IDE, the best Web tool, …
- When Eclipse was built
  - Focus a developer’s productivity
  - Seamless integration across a set of tools to improve the productivity of one developer

…to the Team

- Today, we must focus on the team’s productivity
  - Collaboration
  - Geographically distributed
  - Accelerated delivery demands
  - Agility with predictability
  - Innovation and repeatability
  - Increased need for transparency
- Plus, we want to improve total cost of ownership (TCO) of our solutions

And we are introducing Collaborative Development Environments with jazz

ALM Pattern supporting the full lifecycle



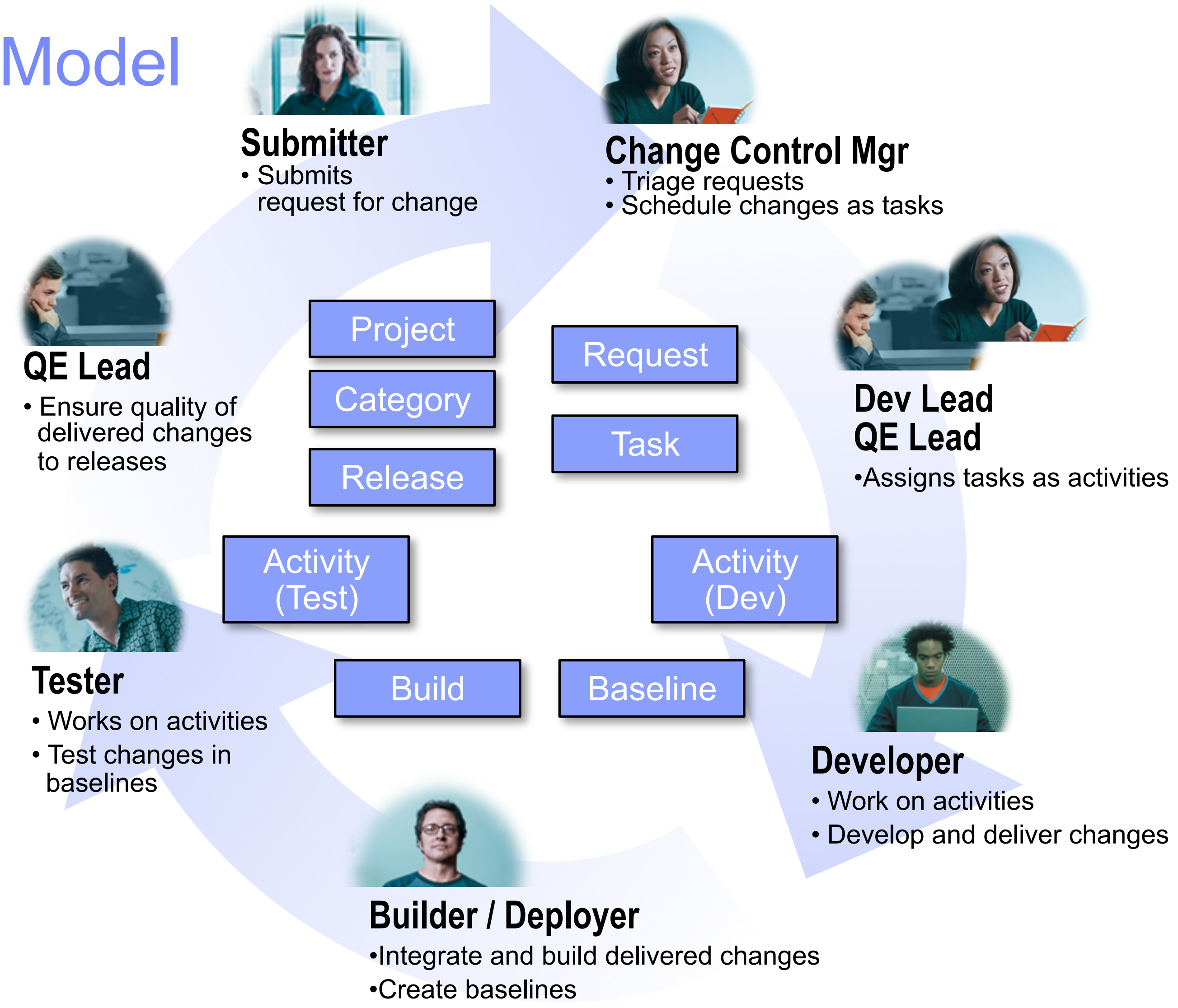
CQALM

- Objective
  - Improving ClearQuest ability to support the management of artifacts through the application lifecycle
  - Consolidate Analysis, Design, Test, Build, Deployment and Project Tracking schemas. Establish End-to-End traceability of software development artifacts
  - Leverage CQ as ALM information Hub
  - Out-of-the-box CQ ALM schema for SMB projects or add-ons to existing schemas
  - Provide Out-of-the-box Queries/Reports necessary for release decision making
  - Designed for Geographically Distributed Development

- Status
  - Progress demonstrated at RSDC 2007
  - Key use-cases and requirements identified
  - Targeting availability in 7.1 release 2H 2008

Enterprise ALM Model

- Enterprise process model focused on Request object submitted for a Release of a Project
- Workload can be balanced across Phases and Iterations of a Project
- Activity objects governing the work performed by the Roles in the teams



Submitter
- Submits request for change

Change Control Mgr
- Triage requests
- Schedule changes as tasks

QE Lead
- Ensure quality of delivered changes to releases

Dev Lead
QE Lead
•Assigns tasks as activities

Tester
- Works on activities
- Test changes in baselines

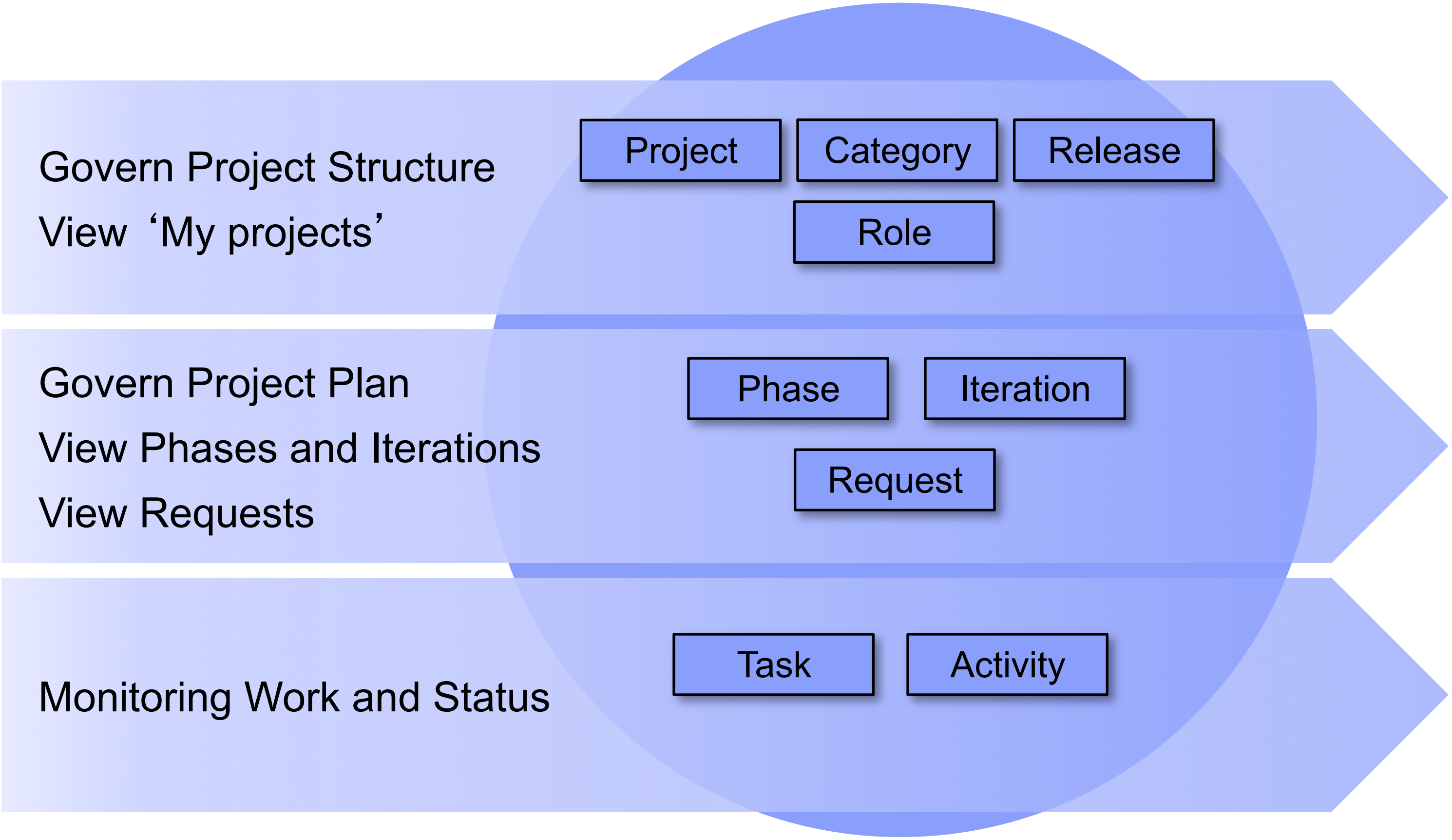
Developer
- Work on activities
- Develop and deliver changes

Builder / Deployer
•Integrate and build delivered changes
•Create baselines

Project Health - Project Manager View

- Patricia the Project Manager must assess the team’s progress / status before finalizing current project milestone
  - Did they complete all work they planned to complete?
  - What’s the Quality?
  - How can Patricia make this decision?

Patricia – Project Manager



Govern Project Structure

View ‘My projects’

Govern Project Plan

View Phases and Iterations

View Requests

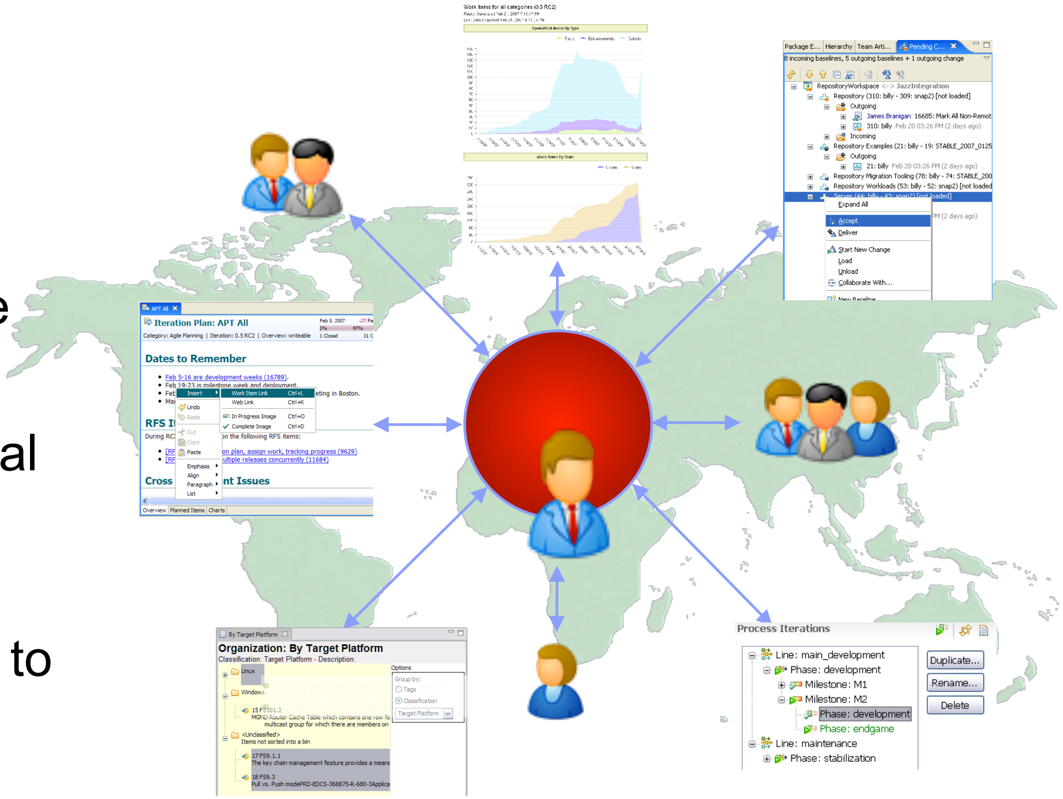
Monitoring Work and Status

What is jazz?

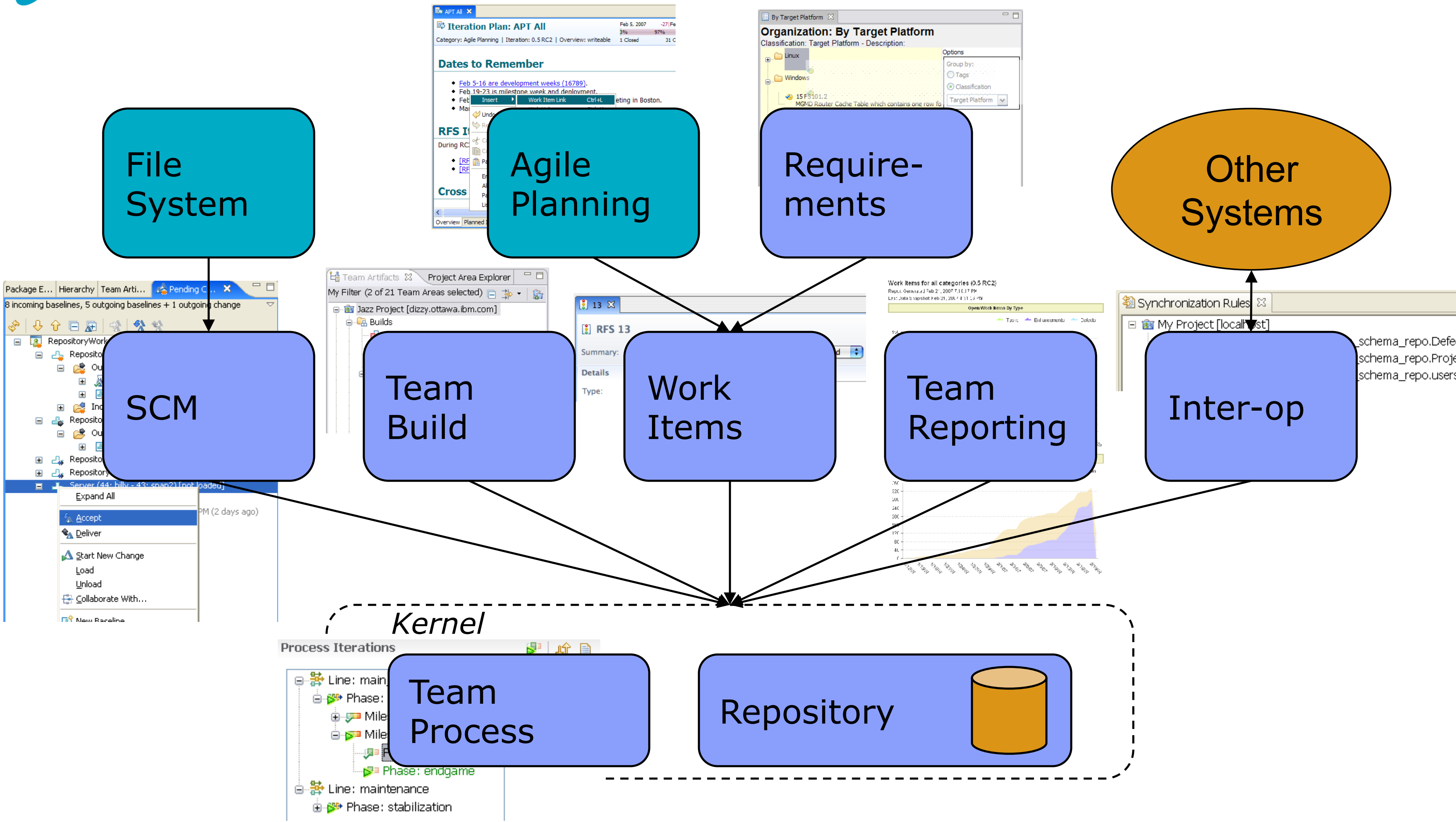
- Jazz leads the way to Collaborative Development Environments
- Jazz is open, scalable and extensible team collaboration technology for seamlessly integrating work across the development lifecycle
  - Open and extensible in both worlds (Client and Server)
    - Based on the IT standards (J2EE, Web Services, Ajax,…)
    - Uses the Eclipse principles (OSGi bundles, extension points, …)
    - Extensible by IBM, partners, customers, etc…
  - Scalable according to
    - Process complexity
    - Number of users: from small teams to large organizations

What can jazz do for your enterprise?

- Jazz enables projects to be run more effectively by providing accurate real-time project health information drawn directly from actual project work
- Jazz manages artifacts across the development lifecycle
- Jazz shifts thinking from "Individual First" beyond "Team First" to "Collaboration First"
- Jazz enables development teams to collaborate in real time in the context of the work that they are doing


Work Items for all categories (0.5 RC2)
Package E...
Hierarchy
Team Arti...
Pending C...
8 incoming baselines, 5 outgoing baselines + 1 outgoing change
RepositoryWorkspace <-> JazzIntegration
Repository (310: billy - 309: snap2) [not loaded]
Outgoing
James Branigan 16685: Mark All Non-Remot
310: billy Feb 20 03:26 PM (2 days ago)
Incoming
Repository Examples (21: billy - 19: STABLE_2007_0125
Outgoing
21: billy Feb 20 03:26 PM (2 days ago)
Repository Migration Tooling (78: billy - 74: STABLE_200
Repository Workloads (53: billy - 52: snap2) [not loaded
Server (44: billy - 43: snap2) [not loaded]
Expand All
Accept
Deliver
Start New Change
Load
Unload
Collaborate With...
New Baseline
APT All
Iteration Plan: APT All
Category: Agile Planning | Iteration: 0.5 RC2 | Overview: writeable
Feb 5, 2007
3%
97%
1 Closed
Dates to Remember
Feb 5-16 are development weeks (16789).
Feb 19-23 is milestone week and deployment.
eting in Boston.
Insert
Work Item Link Ctrl+L
Web Link Ctrl+K
In Progress Image Ctrl+O
Complete Image Ctrl+D
Undo
Redo
Cut
Copy
Paste
Emphasis
Align
Paragraph
List
RFS It
During RC2
on the following RFS items:
on plan, assign work, tracking progress (9629)
ultiple releases concurrently (11684)
Cross
nt Issues
Overview
Planned Items
Charts
By Target Platform
Organization: By Target Platform
Classification: Target Platform - Description:
Linux
Windows
15 FS101.2
MGMD Router Cache Table which contains one row fo multicast group for which there are members on
<Unclassified>
Items not sorted into a bin
17 FS9.1.1
The key chain management feature provides a means
18 FS9.3
Pull vs. Push modePRD-EDCS-368875-R-680-3Applica
Options
Group by:
○ Tags
● Classification
Target Platform
Process Iterations
Line: main_development
Phase: development
Milestone: M1
Milestone: M2
Phase: development
Phase: endgame
Line: maintenance
Phase: stabilization
Duplicate...
Rename...
Delete

jazz components overview



Questions

Thank You

Backup

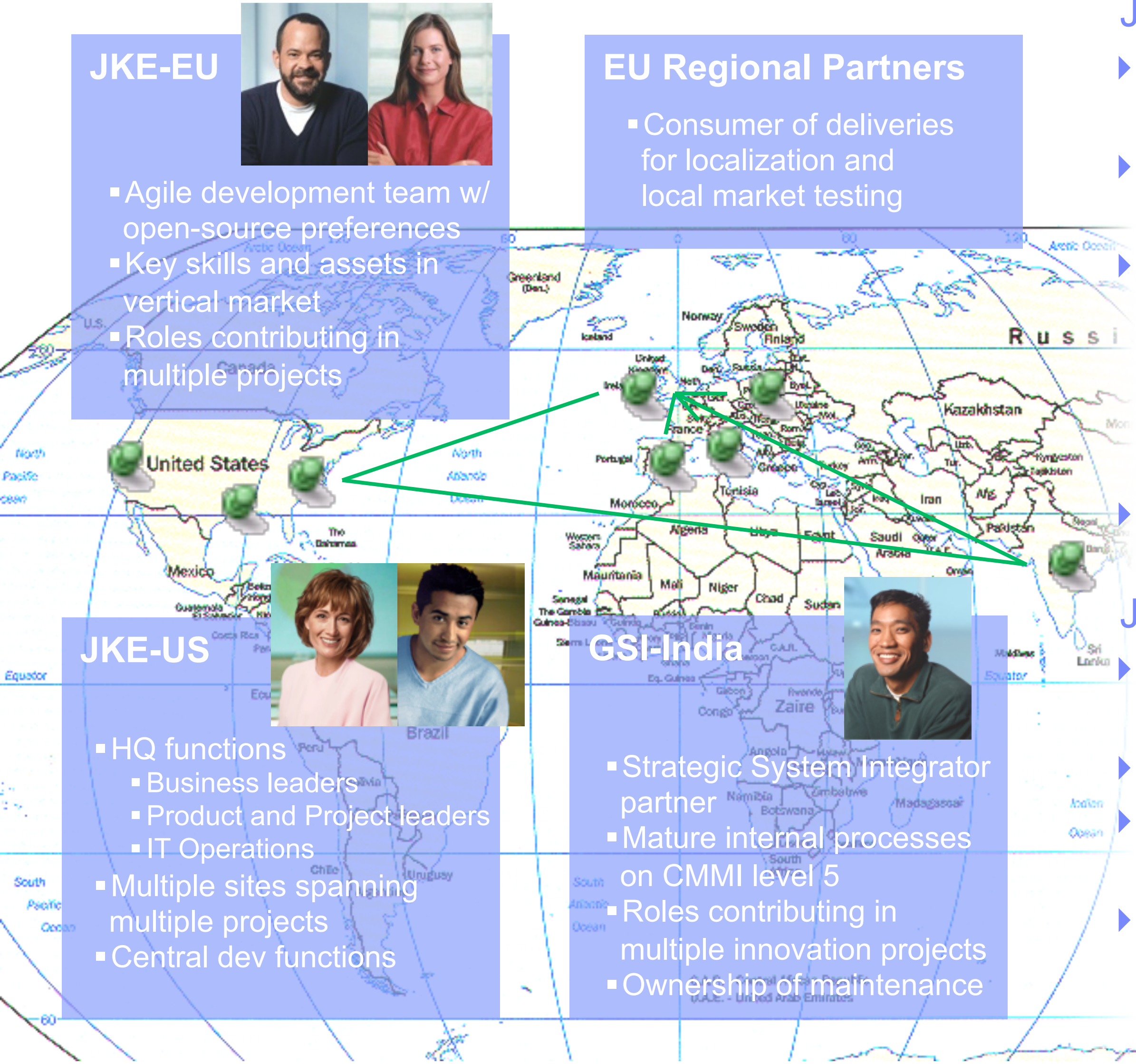
Backup

- Rational Cross-Product Green Threads
- IBM Rational Strategy
- Rational ALM Vision and Manifesto
- Ongoing ALM/2 Investments
  - CQALM
  - Jazz and Rational Team Concert

Cross-Product Green Threads

Business-Driven Development	The identification, analysis, design, implementation, test, deployment and monitoring of a process change.
System In Trouble	The detection, diagnose, fix, test and redeployment of an encountered problem in a system in production.
Deployment	The deployment of a system into test or deployment.
Software Development Governance	The creation and establishment of a governance framework, including definition and management of policies and audits, in a software development organization throughout the development lifecycle.
Geographically-Distributed Development	The collaboration on a single application release, which must also integrate fixes from an outsourcer who handles the application maintenance.
Systems Development	Challenges of software development in the context of systems development
Application Lifecycle Management	Explores traceability across the development lifecycle, following a single requirement from initial creation to deployed release.
SOA for Financial Services Sector	Challenges of use, reuse, and management of IFW/IAA models, SWG tools and methodologies

JK Enterprises – GDALM Development & Delivery Scenario



JKE-EU

- Agile development team w/ open-source preferences
- Key skills and assets in vertical market
- Roles contributing in multiple projects

EU Regional Partners

- Consumer of deliveries for localization and local market testing

JKE-US

- HQ functions
  - Business leaders
  - Product and Project leaders
  - IT Operations
- Multiple sites spanning multiple projects
- Central dev functions

GSI-India

- Strategic System Integrator partner
- Mature internal processes on CMMI level 5
- Roles contributing in multiple innovation projects
- Ownership of maintenance

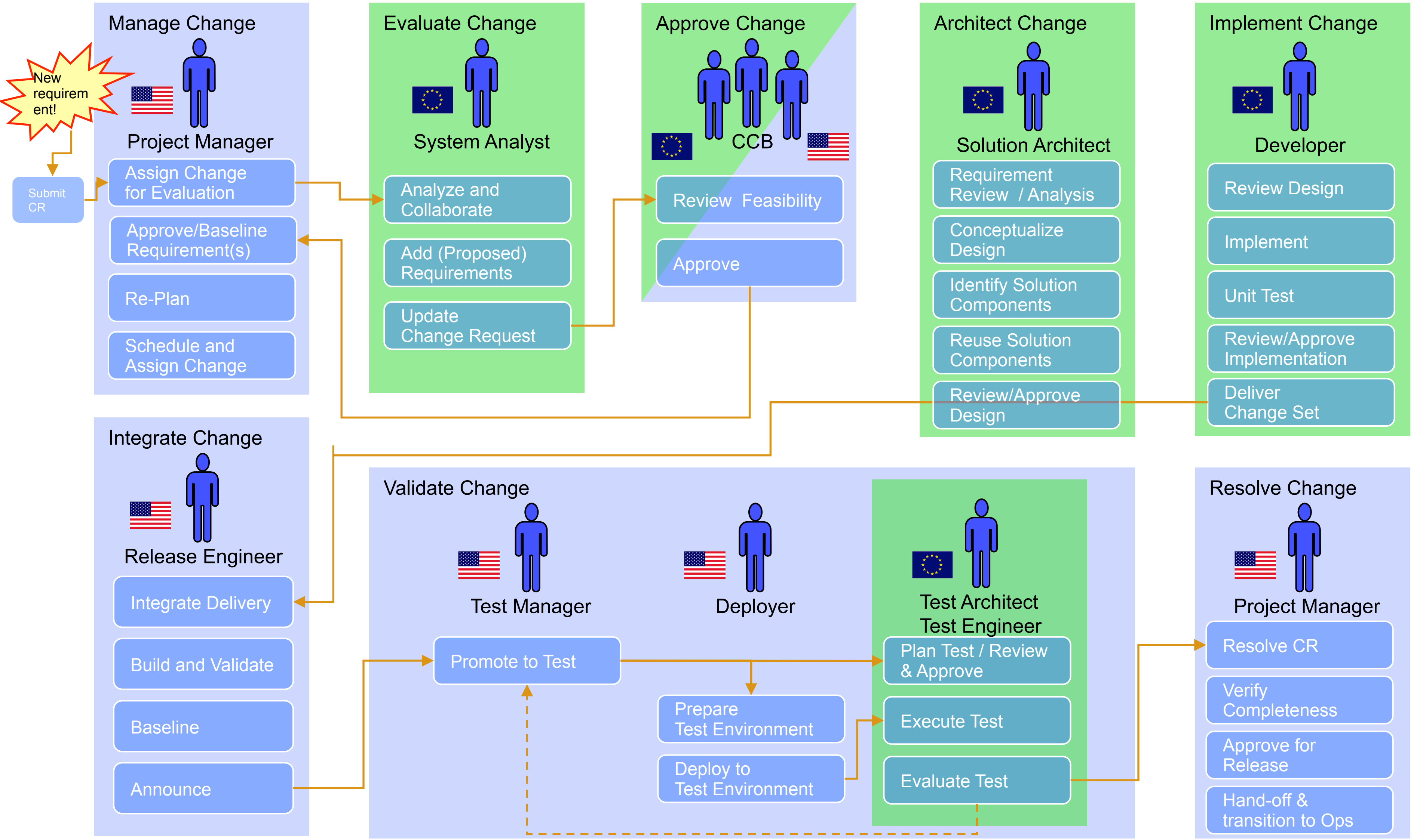
JKE Scenario

- JK Enterprise is a financial sector company
- JKE IT is tasked to deliver business optimization project(s)
- JKE R&D Conducting Organizational Distributed Development
  - New innovation projects run in US, EU, and India
  - Maintenance in India
  - Partner network for application localization
- Increasing dependency on off-shore capacity

JKE ALM Challenges

- Extending tools to a Global Governance and Delivery Platform
- Enable team collaboration
- Improve IT delivery efficiency and capacity
- Manage repository and asset integration and security

GDALM - New Development Thread



New requirement!

Submit CR

Manage Change

Project Manager

- Assign Change for Evaluation
- Approve/Baseline Requirement(s)
- Re-Plan
- Schedule and Assign Change

Evaluate Change

System Analyst

- Analyze and Collaborate
- Add (Proposed) Requirements
- Update Change Request

Approve Change

CCB

- Review Feasibility
- Approve

Architect Change

Solution Architect

- Requirement Review / Analysis
- Conceptualize Design
- Identify Solution Components
- Reuse Solution Components
- Review/Approve Design

Implement Change

Developer

- Review Design
- Implement
- Unit Test
- Review/Approve Implementation
- Deliver Change Set

Integrate Change

Release Engineer

- Integrate Delivery
- Build and Validate
- Baseline
- Announce

Validate Change

Test Manager

- Promote to Test

Deployer

- Prepare Test Environment
- Deploy to Test Environment

Test Architect
Test Engineer

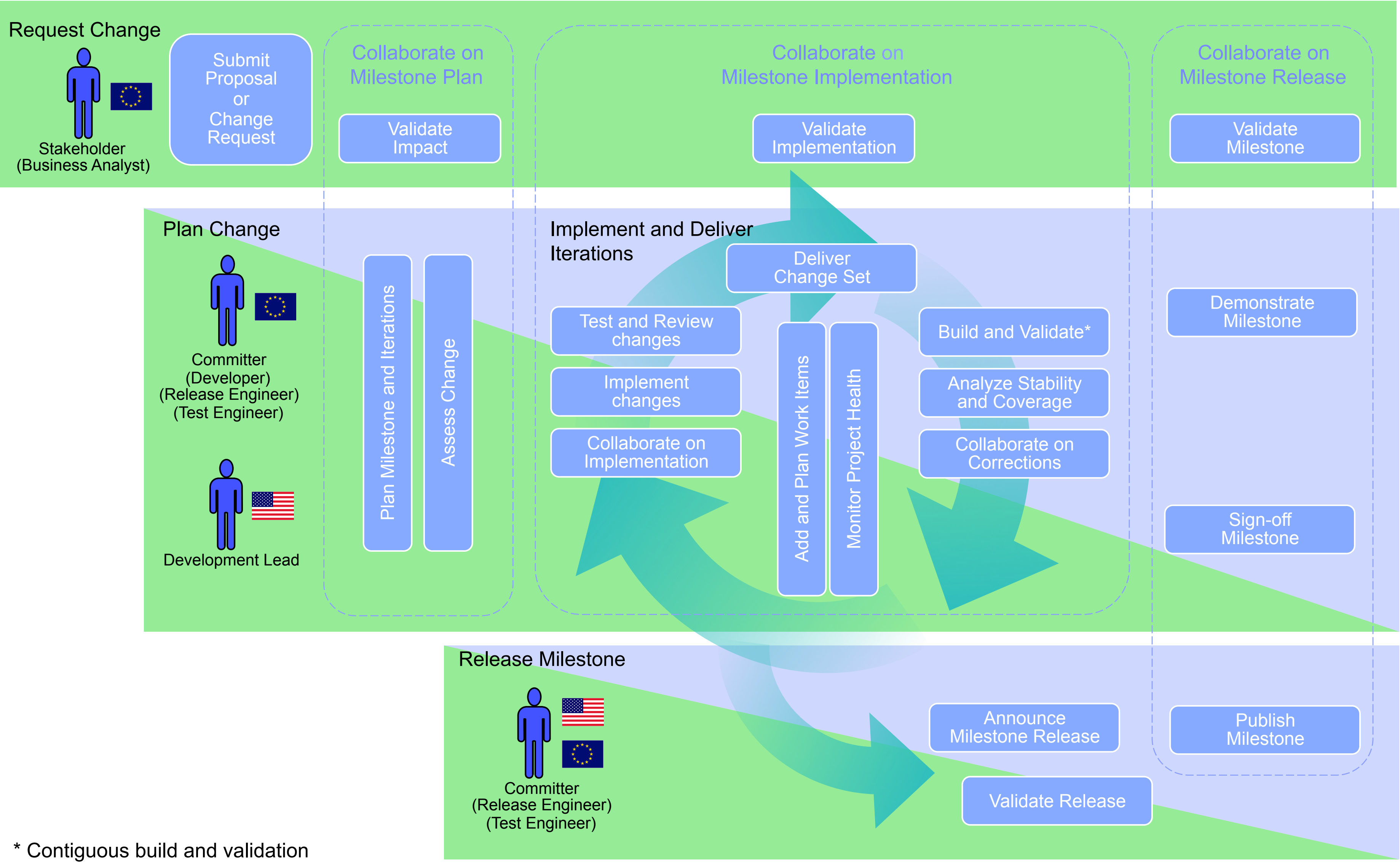
- Plan Test / Review & Approve
- Execute Test
- Evaluate Test

Resolve Change

Project Manager

- Resolve CR
- Verify Completeness
- Approve for Release
- Hand-off & transition to Ops

RTC Develop and Deliver Scenario



Request Change

Stakeholder
(Business Analyst)

Submit Proposal or Change Request

Collaborate on Milestone Plan

Validate Impact

Collaborate on Milestone Implementation

Validate Implementation

Collaborate on Milestone Release

Validate Milestone

Plan Change

Committer
(Developer)
(Release Engineer)
(Test Engineer)

Development Lead

Plan Milestone and Iterations

Assess Change

Implement and Deliver Iterations

Deliver Change Set

Test and Review changes

Implement changes

Collaborate on Implementation

Add and Plan Work Items

Monitor Project Health

Build and Validate*

Analyze Stability and Coverage

Collaborate on Corrections

Demonstrate Milestone

Sign-off Milestone

Release Milestone

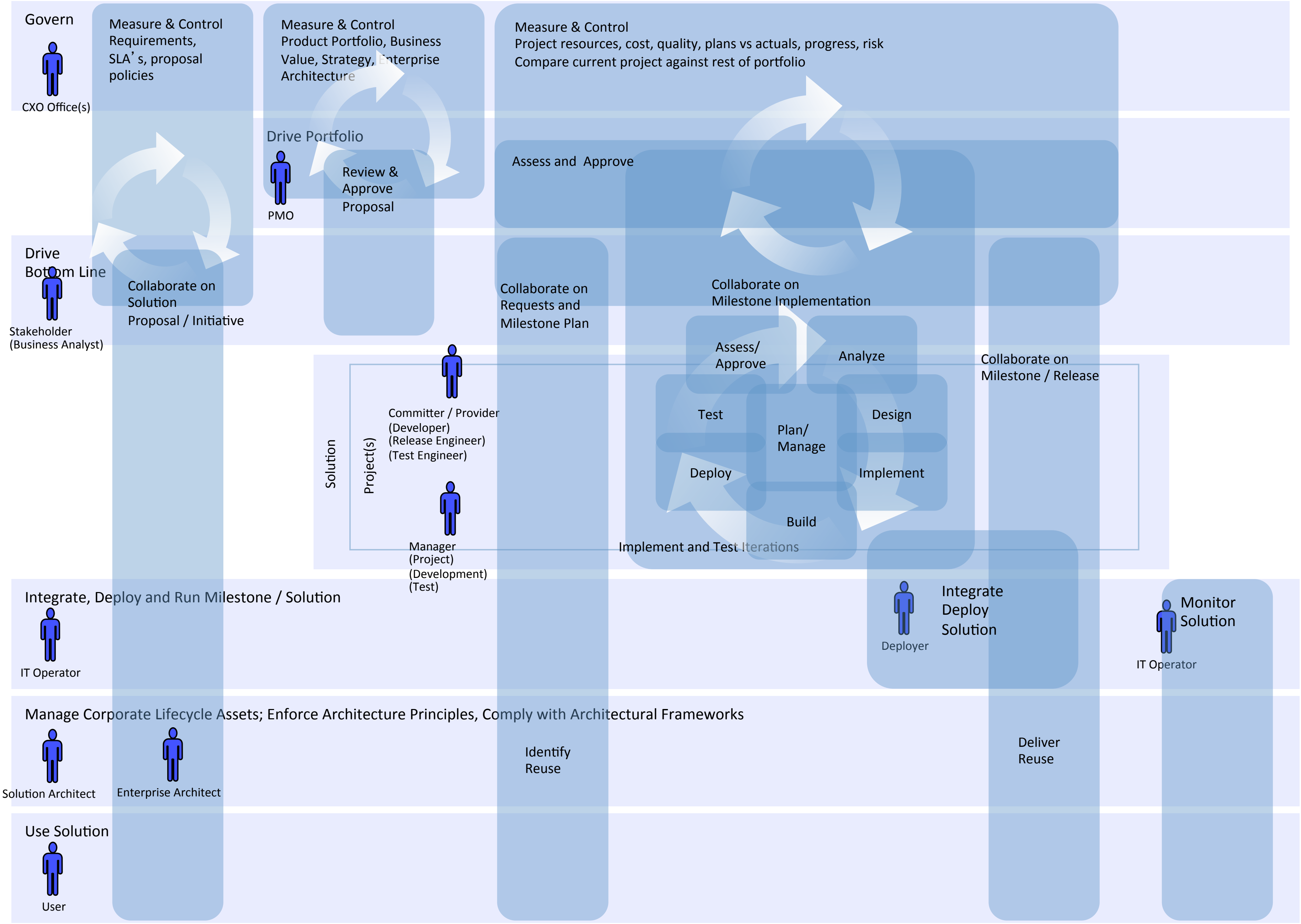
Committer
(Release Engineer)
(Test Engineer)

Announce Milestone Release

Validate Release

Publish Milestone

* Contiguous build and validation



Govern

CXO Office(s)

Measure & Control
Requirements,
SLA's, proposal
policies

Measure & Control
Product Portfolio, Business
Value, Strategy, Enterprise
Architecture

Measure & Control
Project resources, cost, quality, plans vs actuals, progress, risk
Compare current project against rest of portfolio

Drive Portfolio

PMO

Review &
Approve
Proposal

Assess and Approve

Drive
Bottom Line

Stakeholder
(Business Analyst)

Collaborate on
Solution
Proposal / Initiative

Collaborate on
Requests and
Milestone Plan

Collaborate on
Milestone Implementation

Assess/
Approve

Analyze

Test

Plan/
Manage

Design

Deploy

Implement

Build

Implement and Test Iterations

Collaborate on
Milestone / Release

Solution

Project(s)

Committer / Provider
(Developer)
(Release Engineer)
(Test Engineer)

Manager
(Project)
(Development)
(Test)

Integrate, Deploy and Run Milestone / Solution

IT Operator

Deployer

Integrate
Deploy
Solution

Monitor
Solution

IT Operator

Manage Corporate Lifecycle Assets; Enforce Architecture Principles, Comply with Architectural Frameworks

Solution Architect

Enterprise Architect

Identify
Reuse

Deliver
Reuse

Use Solution

User

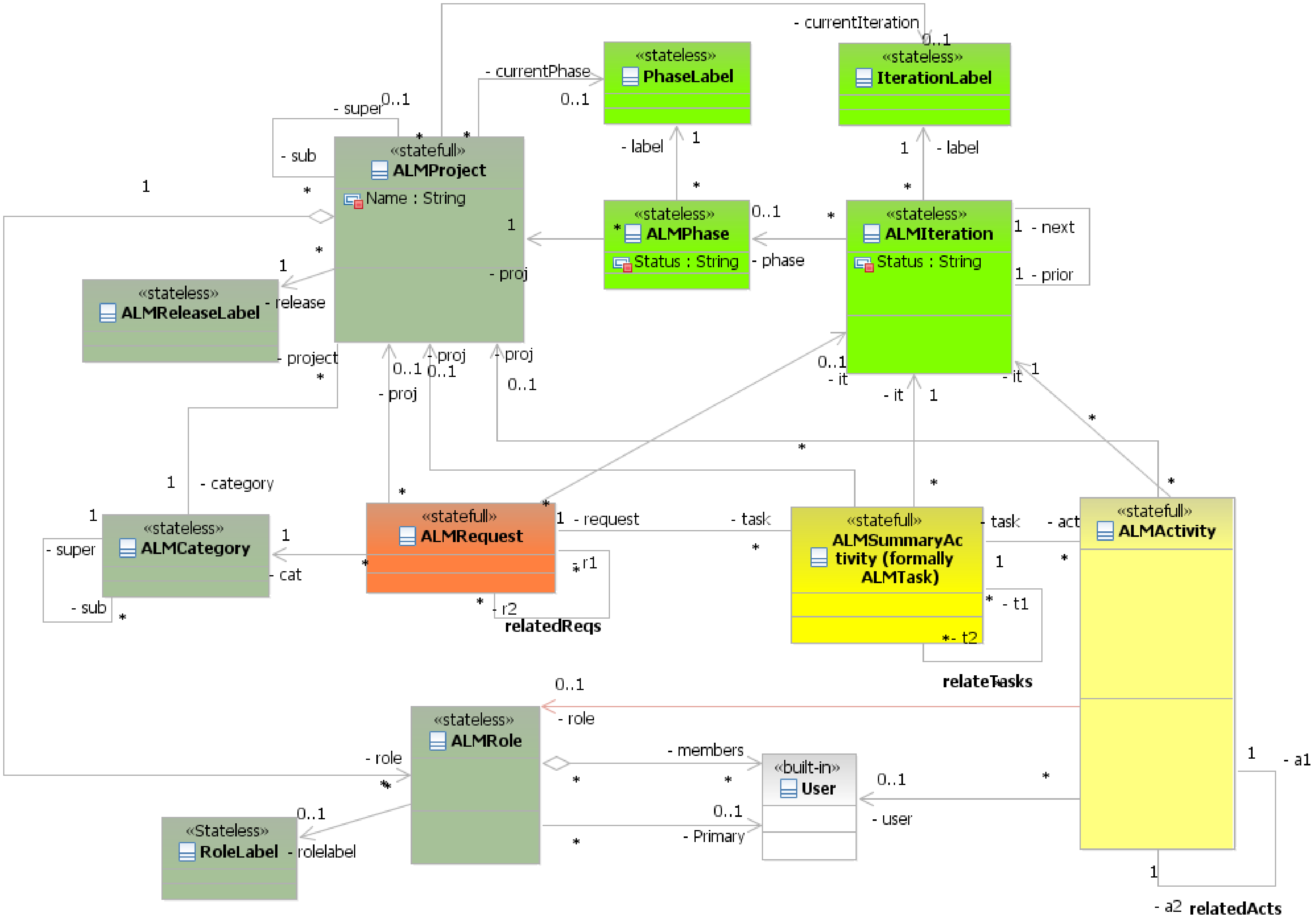
- Business driven software development and delivery supported by
  - Governance
  - Collaboration
  - Traceability
  - Process automation
  - Integration and Distribution

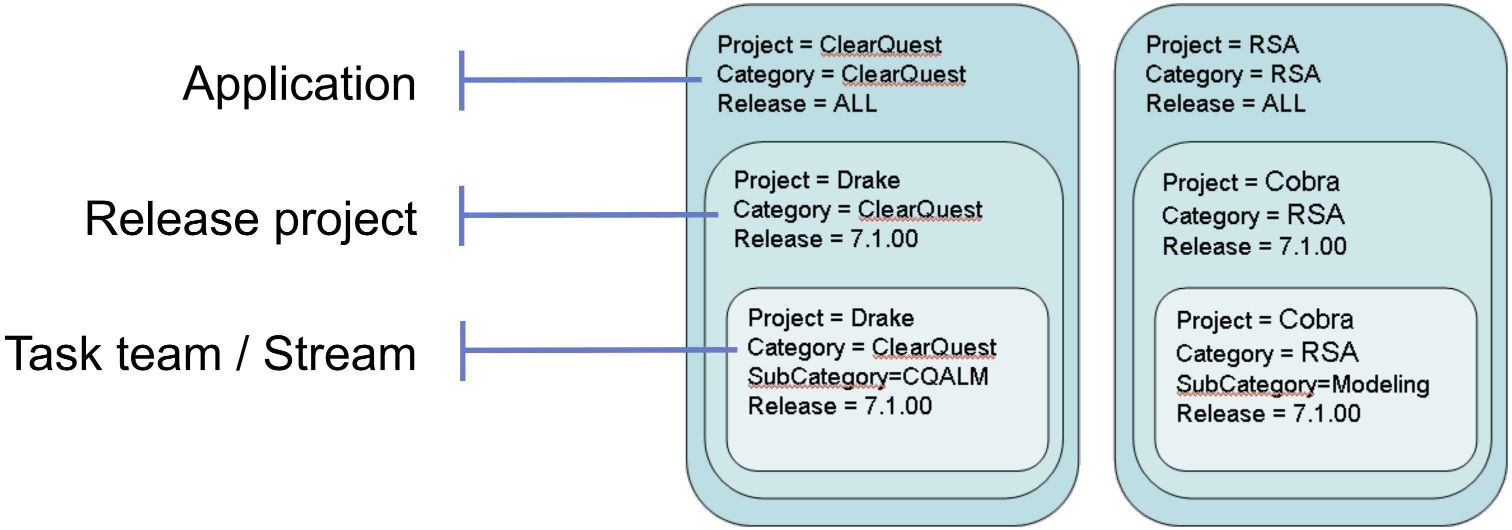
Philosophical Principles

- Development is not an island unto itself; rather, we provide a service to the business.
  - The development team innovates within the context of business goals, enterprise architectures and operational constraints.
  - Teams around the world interact to produce software solutions (Business, Operations, Development, Partners)
- Software Solutions are the product of many Conversations
  - ALM involves many conversations which result in artifacts which ultimately result in deployable software.
  - Support people whoever they are, where ever they are, and whatever tools they use to create artifacts
  - Relationships Matter. There are People clouds, Artifact clouds, Tag clouds. Public/Shared Bookmarks.
    - People collaborate to produce artifacts
    - Artifacts are linked and traceable regardless of where they are located
    - A change to one artifact impacts other artifacts and the people who own them
- Solutions never die, they are refined & maintained for years.
  - All artifacts are Investments and should be treated as such.
  - Most project artifacts move forward and are re-used for the next "release."
  - Solutions can be assembled from parts of other solutions (asset & artifact re-use)
- There are cycles within ALM; some of these cycles are ripe for automation
  - Business priorities drive software project selection while Operations priorities drive maintenance cycles & hotfixes.
  - Project schedules drive development software projects
  - Quality drives development cycles within iterations; there are repeated patterns of activity
- Simplicity First
  - It’s easy to grow from something simple. It’s difficult to simplify something that is complex.
  - Customers look to Rational for a reference implementation to start with, but expect to customize it to suit their needs

What is jazz?

- Jazz leads the way to Collaborative Development Environments
- Jazz is open, scalable and extensible team collaboration technology for seamlessly integrating work across the development lifecycle
  - Open and extensible in both worlds (Client and Server)
    - Based on the IT standards (J2EE, Web Services, Ajax,…)
    - Uses the Eclipse principles (OSGi bundles, extension points, …)
    - Extensible by IBM, partners, customers, etc…
  - Scalable according to
    - Process complexity
    - Number of users: from small teams to large organizations


- currentIteration
0..1
«stateless»
PhaseLabel
«stateless»
IterationLabel
- currentPhase
0..1
- super
0..1
- sub
«statefull»
ALMProject
Name : String
1
*
*
*
- label
1
*
1
- label
*
«stateless»
ALMPhase
Status : String
0..1
- phase
*
«stateless»
ALMIteration
Status : String
1 - next
1 - prior
1
*
«stateless»
ALMReleaseLabel
1
- release
- proj
- project
*
0..1
0..1
- proj
- proj
- proj
0..1
0..1
- it
- it
1
- it
1
*
*
*
1
- category
1
«stateless»
ALMCategory
- super
- sub
*
1
- cat
*
«statefull»
ALMRequest
*
1 - request
- r1
- r2
relatedReqs
- task
*
«statefull»
ALMSummaryActivity (formally ALMTask)
- task
1
*
- t1
*- t2
relateTasks
- act
*
«statefull»
ALMActivity
0..1
- role
«stateless»
ALMRole
- role
*
- members
*
*
«built-in»
User
0..1
*
- user
0..1
- Primary
*
0..1
«Stateless»
RoleLabel
- rolelabel
1
- a1
1
- a2
relatedActs



Application

Release project

Task team / Stream

Project = ClearQuest
Category = ClearQuest
Release = ALL

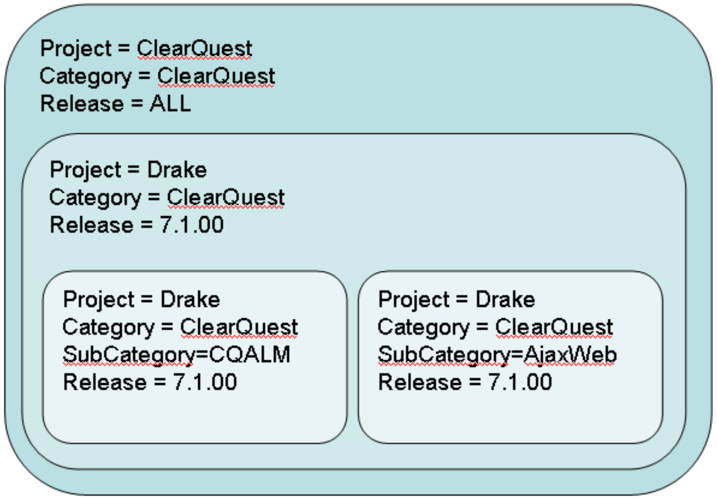
Project = Drake
Category = ClearQuest
Release = 7.1.00

Project = Drake
Category = ClearQuest
SubCategory=CQALM
Release = 7.1.00

Project = RSA
Category = RSA
Release = ALL

Project = Cobra
Category = RSA
Release = 7.1.00

Project = Cobra
Category = RSA
SubCategory=Modeling
Release = 7.1.00



Project = ClearQuest
Category = ClearQuest
Release = ALL

Project = Drake
Category = ClearQuest
Release = 7.1.00

Project = Drake
Category = ClearQuest
SubCategory=CQALM
Release = 7.1.00

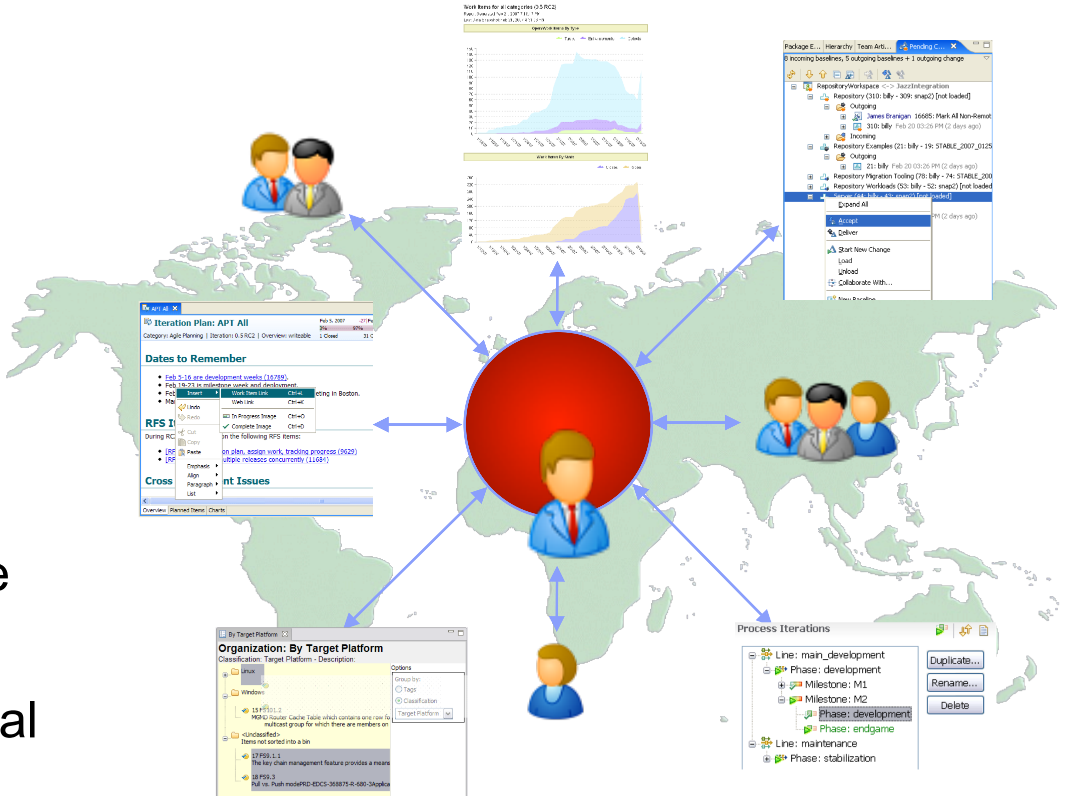
Project = Drake
Category = ClearQuest
SubCategory=AjaxWeb
Release = 7.1.00

CQALM Use-Cases

- Project Manager
  - Organize project
    - Create Project
    - Establish Roles, Team Members, Categories, Phases and Iterations
    - Establish access security
  - Track Project
    - Identify any outstanding requests
    - Identify un-assigned work (Request/Task)
    - Project load balancing - Who is working on what
    - Task completeness - How much is left to go
    - Project status against plan - Are we done
- Any Role
  - Submit Request, Task, Activity
  - Deliver Activity
  - Report on assigned Task, Activity - What do I have to do?
  - Report on assignment Status - What is the state of work I've submitted?
  - Report on health - What's our status?
- Release Engineer
  - Identify Build status and contents
- Tester
  - Determine what to test

What can Jazz do for your enterprise?

- Jazz enables development teams to collaborate in real time in the context of the work that they are doing
- Jazz enables projects to be run more effectively by providing accurate real-time project health information drawn directly from actual project work
- Jazz manages artifacts across the development lifecycle
- Jazz shifts thinking from "Individual First" beyond "Team First" to "Collaboration First"



Work Items for all categories (0.5 RC2)
Report Generated Feb 21, 2007 7:18:17 PM
Last Data Snapshot Feb 21, 2007 4:51:59 PM

Package E... | Hierarchy | Team Arti... | Pending C...

8 incoming baselines, 5 outgoing baselines + 1 outgoing change

- RepositoryWorkspace <-> JazzIntegration
  - Repository (310: billy - 309: snap2) [not loaded]
    - Outgoing
      - James Branigan 16685: Mark All Non-Remot
      - 310: billy Feb 20 03:26 PM (2 days ago)
    - Incoming
  - Repository Examples (21: billy - 19: STABLE_2007_0125
    - Outgoing
      - 21: billy Feb 20 03:26 PM (2 days ago)
  - Repository Migration Tooling (78: billy - 74: STABLE_200
  - Repository Workloads (53: billy - 52: snap2) [not loaded
  - Server (44: billy - 43: snap2) [not loaded]

APT All

Iteration Plan: APT All

Category: Agile Planning | Iteration: 0.5 RC2 | Overview: writeable

Feb 5, 2007 3% 97% 1 Closed 31 C

Dates to Remember

- Feb 5-16 are development weeks (16789).
- Feb 19-23 is milestone week and deployment.
- Feb ... eting in Boston.
- Mar

RFS It...

During RC2 ... on the following RFS items:

- [RF... on plan, assign work, tracking progress (9629)
- [RF... ultiple releases concurrently (11684)

Cross ... nt Issues

By Target Platform

Organization: By Target Platform

Classification: Target Platform - Description:

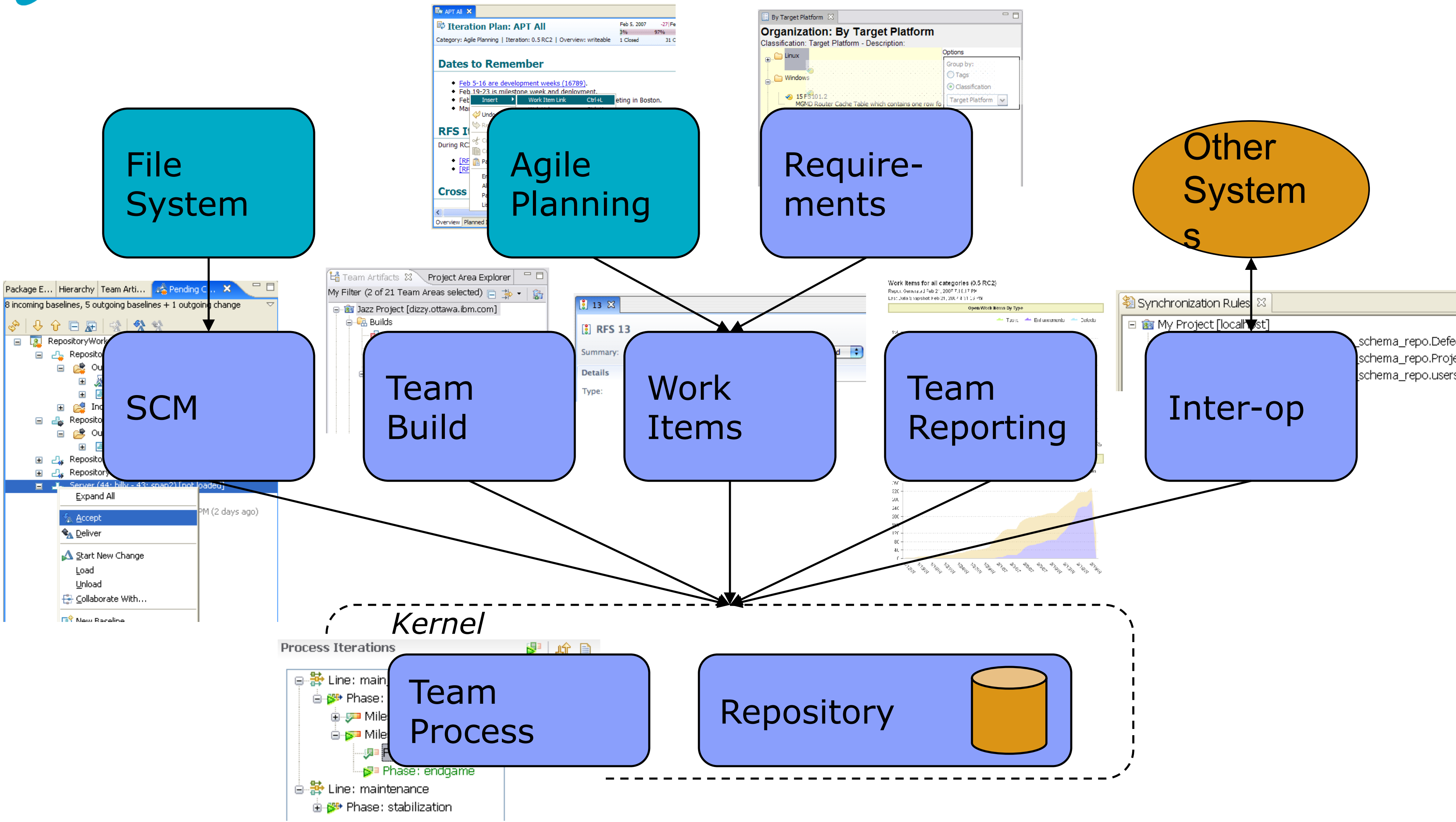
- Linux
- Windows
  - 15 FS101.2
    MGMD Router Cache Table which contains one row fo multicast group for which there are members on
- <Unclassified>
  Items not sorted into a bin
  - 17 FS9.1.1
    The key chain management feature provides a means
  - 18 FS9.3
    Pull vs. Push modePRD-EDCS-368875-R-680-3Applica

Options
Group by:
○ Tags
● Classification
Target Platform

Process Iterations

- Line: main_development
  - Phase: development
    - Milestone: M1
    - Milestone: M2
      - Phase: development
      - Phase: endgame
- Line: maintenance
  - Phase: stabilization

jazz components overview



Iteration Plan: APT All

Category: Agile Planning | Iteration: 0.5 RC2 | Overview: writeable

Dates to Remember

- Feb 5-16 are development weeks (16789).
- Feb 19-23 is milestone week and deployment.

Organization: By Target Platform

Classification: Target Platform - Description:

Options

Group by:

○ Tags

● Classification

8 incoming baselines, 5 outgoing baselines + 1 outgoing change

Expand All

Accept

Deliver

Start New Change

Load

Unload

Collaborate With...

Team Artifacts | Project Area Explorer

My Filter (2 of 21 Team Areas selected)

Jazz Project [dizzy.ottawa.ibm.com]

Builds

RFS 13

Summary:

Details

Type:

Synchronization Rules

My Project [localhost]

schema_repo.Defe

schema_repo.Proje

schema_repo.users

Process Iterations

Line: main

Phase:

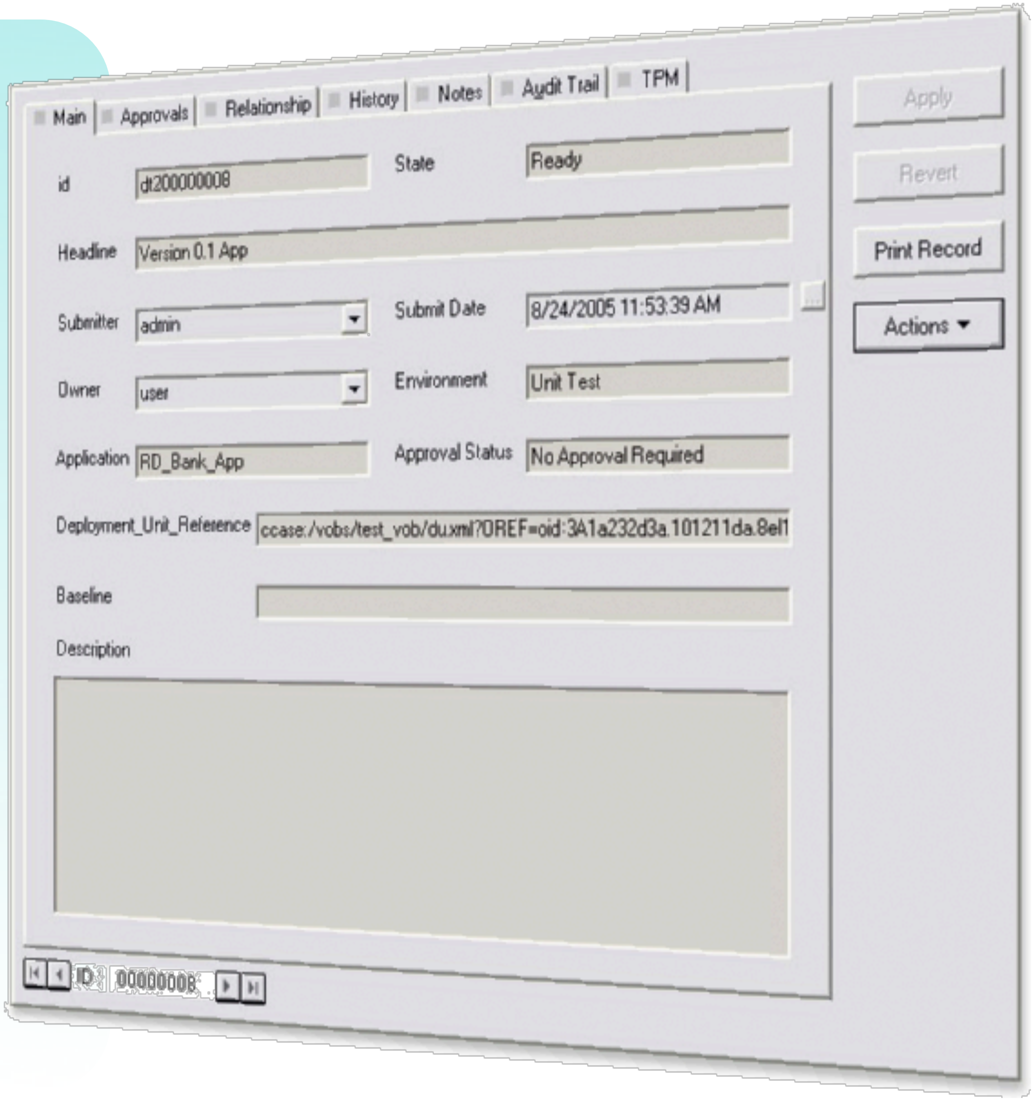
Phase: endgame

Line: maintenance

Phase: stabilization

Build and Deployment Tracking with ClearQuest

- **Track Builds and Deployments into test environments**
  - Track which build is to be used for testing
  - Define and sequence test environments specific to your organization
  - Establish approval gates before deploying to an environment
- **Pass regulatory audits**
  - Associate builds with deployments
  - Capture electronic signatures when needed
  - Maintain build artifacts under version control in UCM
- **Add Build Automation**
  - Integrate 3rd party build automation packages
  - Automatically create and update build records
- **Add Deployment Automation**
  - Optionally link to Tivoli Provisioning Manager to automate the provisioning of servers with latest build
  - Deploy “approved” build files directly from source control



id: dt200000008

State: Ready

Headline: Version 0.1 App

Submitter: admin

Submit Date: 8/24/2005 11:53:39 AM

Owner: user

Environment: Unit Test

Application: RD_Bank_App

Approval Status: No Approval Required

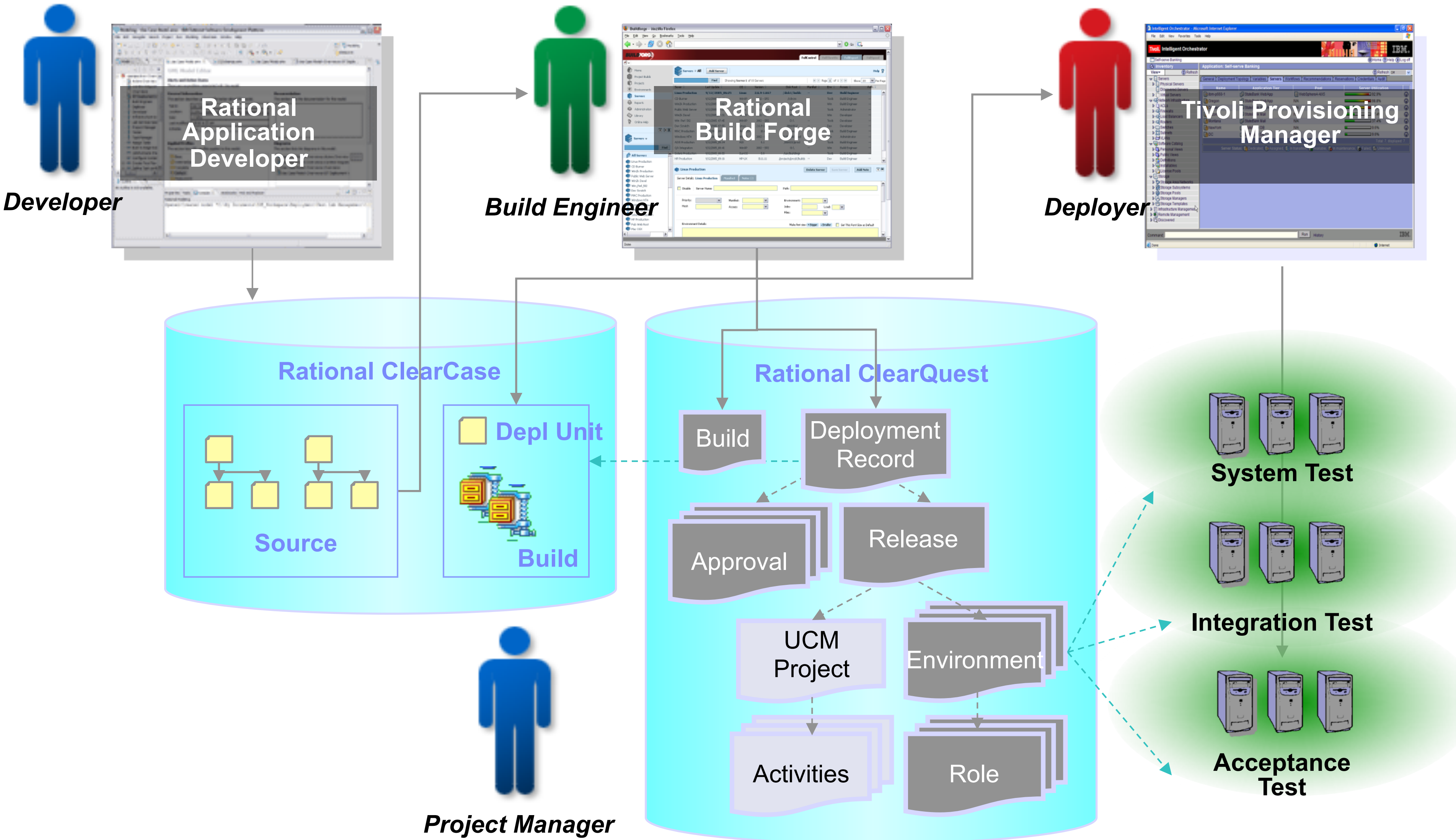
Deployment_Unit_Reference: ccase:/vobs/test_vob/du.xml?OREF=oid:3A1a232d3a.101211da.8el1

Baseline:

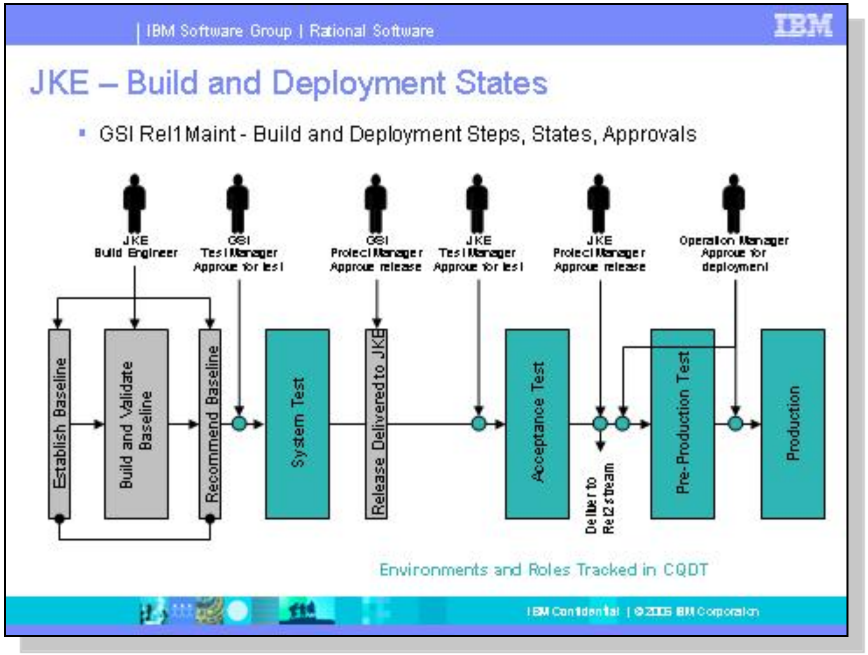
Description:

ID: 00000008

Build and Deployment Tracking



Build and Deployment Workflows



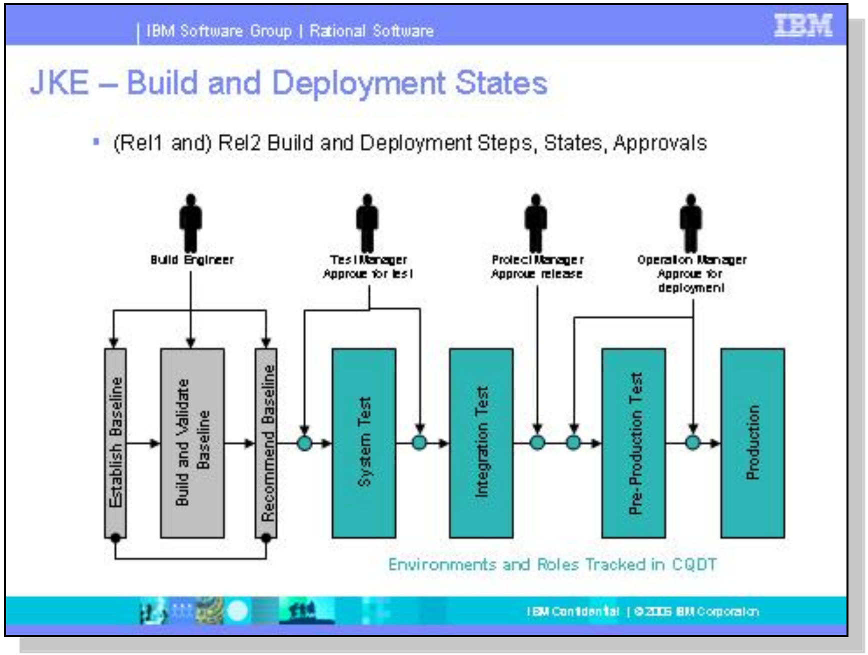
IBM Software Group | Rational Software

JKE – Build and Deployment States

- GSI Rel1 Maint - Build and Deployment Steps, States, Approvals

Environments and Roles Tracked in CQDT

IBM Confidential | © 2006 IBM Corporation



IBM Software Group | Rational Software

JKE – Build and Deployment States

- (Rel1 and) Rel2 Build and Deployment Steps, States, Approvals

Environments and Roles Tracked in CQDT

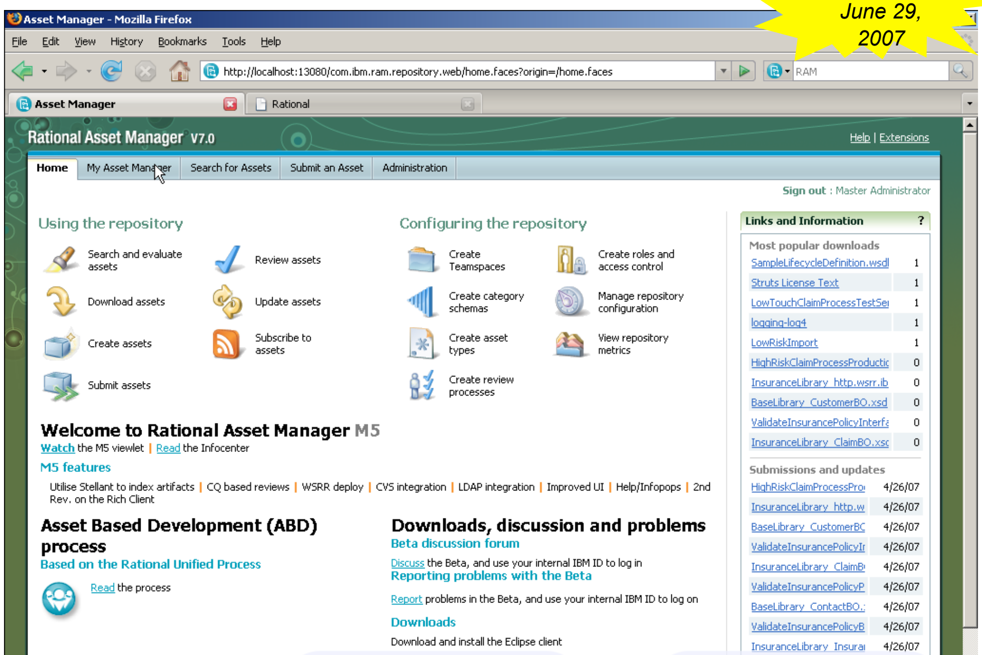
IBM Confidential | © 2006 IBM Corporation

Introducing IBM Rational Asset Manager

New!

Available June 29, 2007

- Collaborative asset management to identify and manage assets & ROI best practices
- Manages assets across their lifecycle from design/creation to consumption/change
- Manages service creation & reuse across service oriented architectures (SOA) projects
- Leverages an extensive library of process best practices for asset creation & reuse in Rational Method Composer (ABS, SOA, GDD, etc.)


Asset Manager - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:13080/com.ibm.ram.repository.web/home.faces?origin=/home.faces
RAM
Asset Manager
Rational
Rational Asset Manager V7.0
Help | Extensions
Home
My Asset Manager
Search for Assets
Submit an Asset
Administration
Sign out : Master Administrator

Using the repository

Search and evaluate assets
Review assets
Download assets
Update assets
Create assets
Subscribe to assets
Submit assets

Configuring the repository

Create Teamspaces
Create roles and access control
Create category schemas
Manage repository configuration
Create asset types
View repository metrics
Create review processes

Welcome to Rational Asset Manager M5

Watch the M5 viewlet | Read the Infocenter

M5 features

Utilise Stellant to index artifacts | CQ based reviews | WSRR deploy | CVS integration | LDAP integration | Improved UI | Help/Infopops | 2nd Rev. on the Rich Client

Asset Based Development (ABD) process

Based on the Rational Unified Process

Read the process

Downloads, discussion and problems

Beta discussion forum

Discuss the Beta, and use your internal IBM ID to log in

Reporting problems with the Beta

Report problems in the Beta, and use your internal IBM ID to log on

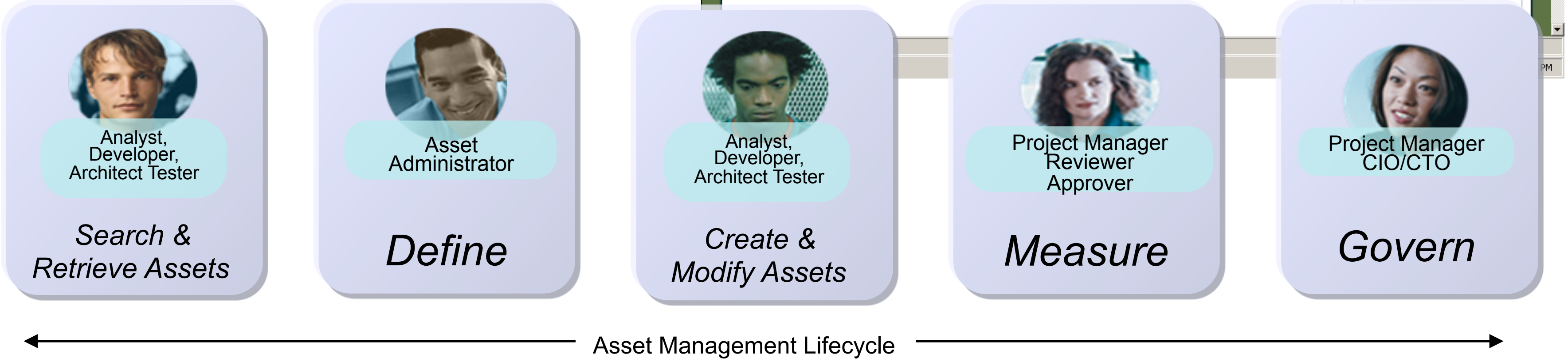
Downloads

Download and install the Eclipse client

Links and Information ?

Most popular downloads	
SampleLifecycleDefinition.wsdl	1
Struts License Text	1
LowTouchClaimProcessTestSe	1
logging-log4	1
LowRiskImport	1
HighRiskClaimProcessProductic	0
InsuranceLibrary_http.wsrr.ib	0
BaseLibrary_CustomerBO.xsd	0
ValidateInsurancePolicyInterfa	0
InsuranceLibrary_ClaimBO.xsc	0

Submissions and updates	
HighRiskClaimProcessPro	4/26/07
InsuranceLibrary_http.w	4/26/07
BaseLibrary_CustomerBC	4/26/07
ValidateInsurancePolicyIr	4/26/07
InsuranceLibrary_ClaimB	4/26/07
ValidateInsurancePolicyP	4/26/07
BaseLibrary_ContactBO.:	4/26/07
ValidateInsurancePolicyB	4/26/07
InsuranceLibrary_Insura	4/26/07



Analyst, Developer, Architect Tester — *Search & Retrieve Assets*

Asset Administrator — *Define*

Analyst, Developer, Architect Tester — *Create & Modify Assets*

Project Manager Reviewer Approver — *Measure*

Project Manager CIO/CTO — *Govern*

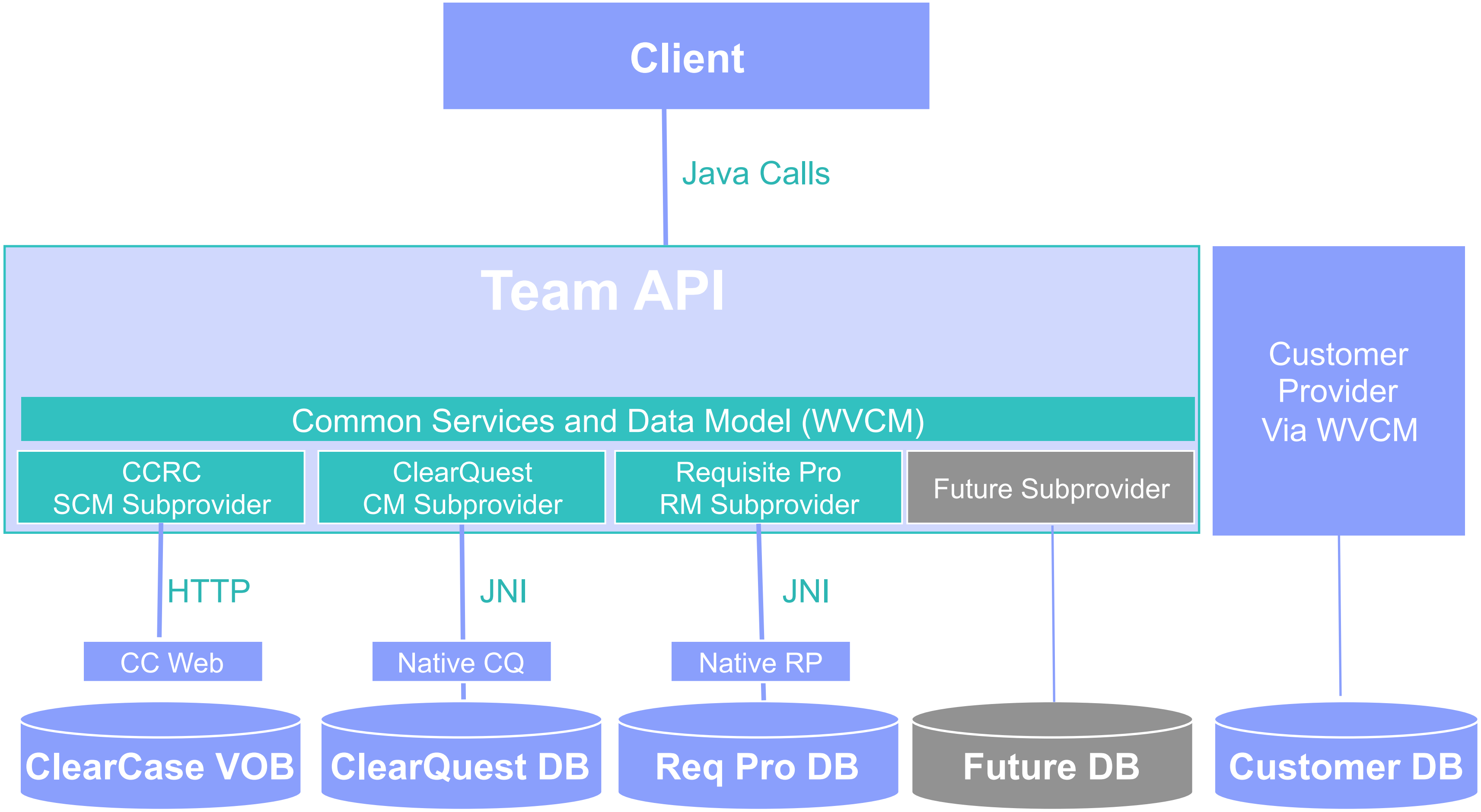
Asset Management Lifecycle

IBM Rational Team API - Platform Support

Tech Preview

- Team API
  - A unified client-side Java API for access to RequisitePro, ClearCase and ClearQuest artifacts.
  - Based on the JSR 147 proposed standard: WVCM
  - OS and platform independent
  - Provides a common team data model and domain specific services
  - Supports the full lifecycle of software product development
  - Supports customer and partner developer ecosystem
- Platforms
  - ClearCase APIs
    - Windows, Linux, AIX, Solaris, HP-UX
  - ClearQuest APIs
    - Windows, Linux, AIX, Solaris, HP-UX
  - Requisite Pro APIs
    - Windows
- Documentation & sample code available on DeveloperWorks

IBM Rational Team API



Client

Java Calls

Team API

Common Services and Data Model (WVCM)

CCRC
SCM Subprovider

ClearQuest
CM Subprovider

Requisite Pro
RM Subprovider

Future Subprovider

Customer
Provider
Via WVCM

HTTP

JNI

JNI

CC Web

Native CQ

Native RP

ClearCase VOB

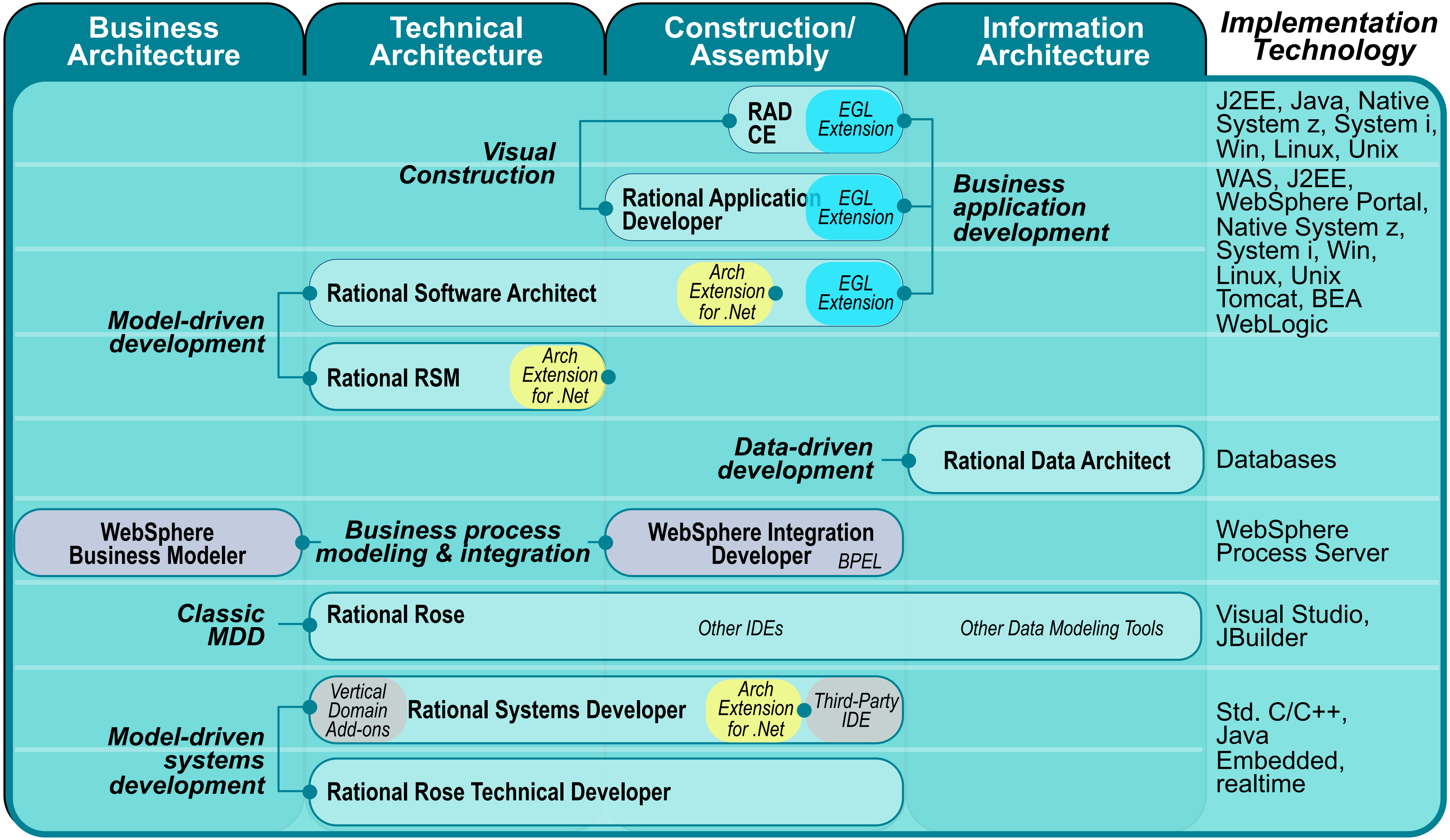
ClearQuest DB

Req Pro DB

Future DB

Customer DB

Current ADC product portfolio



Business Architecture

Technical Architecture

Construction/ Assembly

Information Architecture

Implementation Technology

Visual Construction

RAD CE — *EGL Extension*

Rational Application Developer — *EGL Extension*

Business application development

J2EE, Java, Native System z, System i, Win, Linux, Unix

WAS, J2EE, WebSphere Portal, Native System z, System i, Win, Linux, Unix Tomcat, BEA WebLogic

Model-driven development

Rational Software Architect — *Arch Extension for .Net* — *EGL Extension*

Rational RSM — *Arch Extension for .Net*

Data-driven development

Rational Data Architect

Databases

WebSphere Business Modeler

Business process modeling & integration

WebSphere Integration Developer — *BPEL*

WebSphere Process Server

Classic MDD

Rational Rose — *Other IDEs* — *Other Data Modeling Tools*

Visual Studio, JBuilder

Model-driven systems development

Vertical Domain Add-ons — Rational Systems Developer — *Arch Extension for .Net* — *Third-Party IDE*

Rational Rose Technical Developer

Std. C/C++, Java Embedded, realtime