

Rational roadmap overview

Giorgio Galli
Client Technical Professionals



Innovate2010

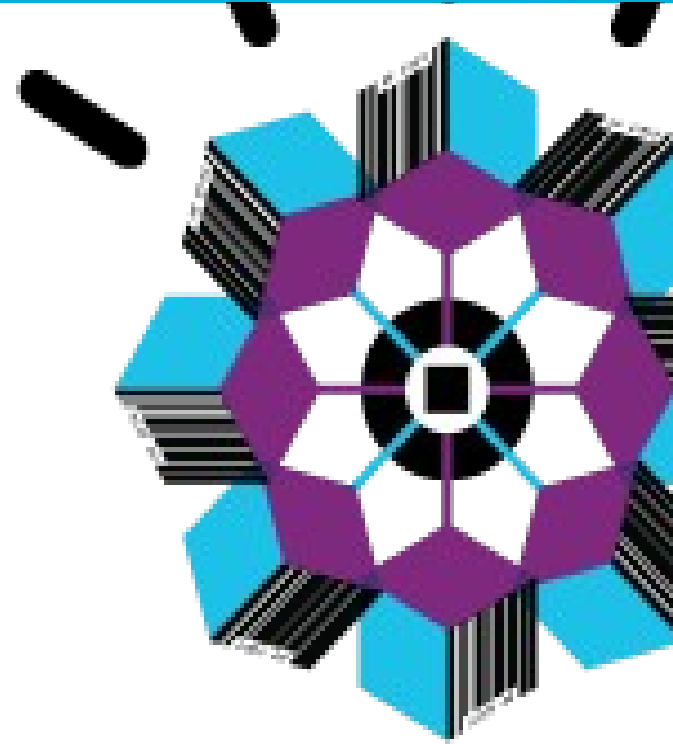
The Rational Software Conference

Let's **build** a smarter planet.



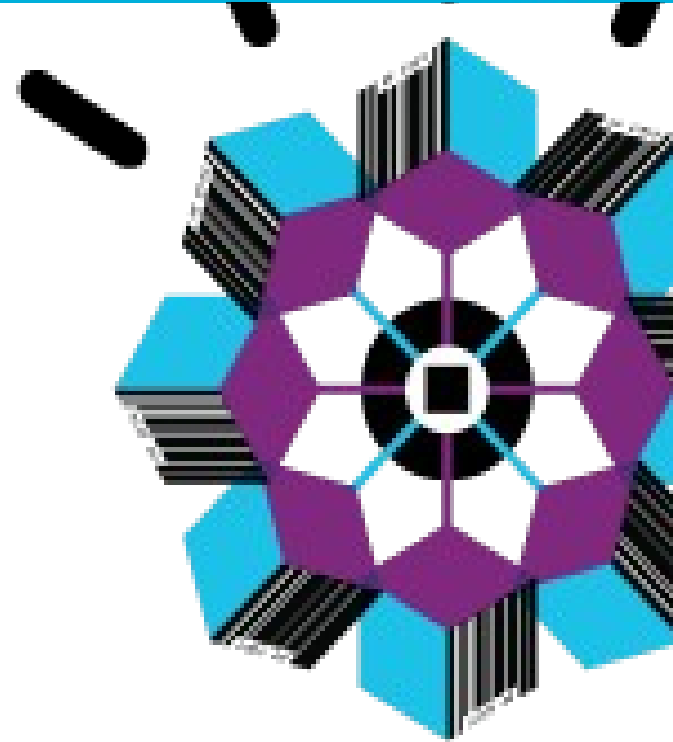
Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Product Direction
- In summary



Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Direction
- In summary



Software Delivery Challenges

Technologies

- Platforms (mobile, server, rich client...)
- Languages (existing & emerging...)
- Programming Model (existing & emerging...)
- Skill/background/target (Web, legacy, embedded...)

Productivity

- Planning Challenges (Business priorities change and project execution ...)
- Team Challenges (Diverse tools)
- Platform Process Challenges (Process is on paper - tools don't enact it)
- Platform Integration Challenges

Collaboration

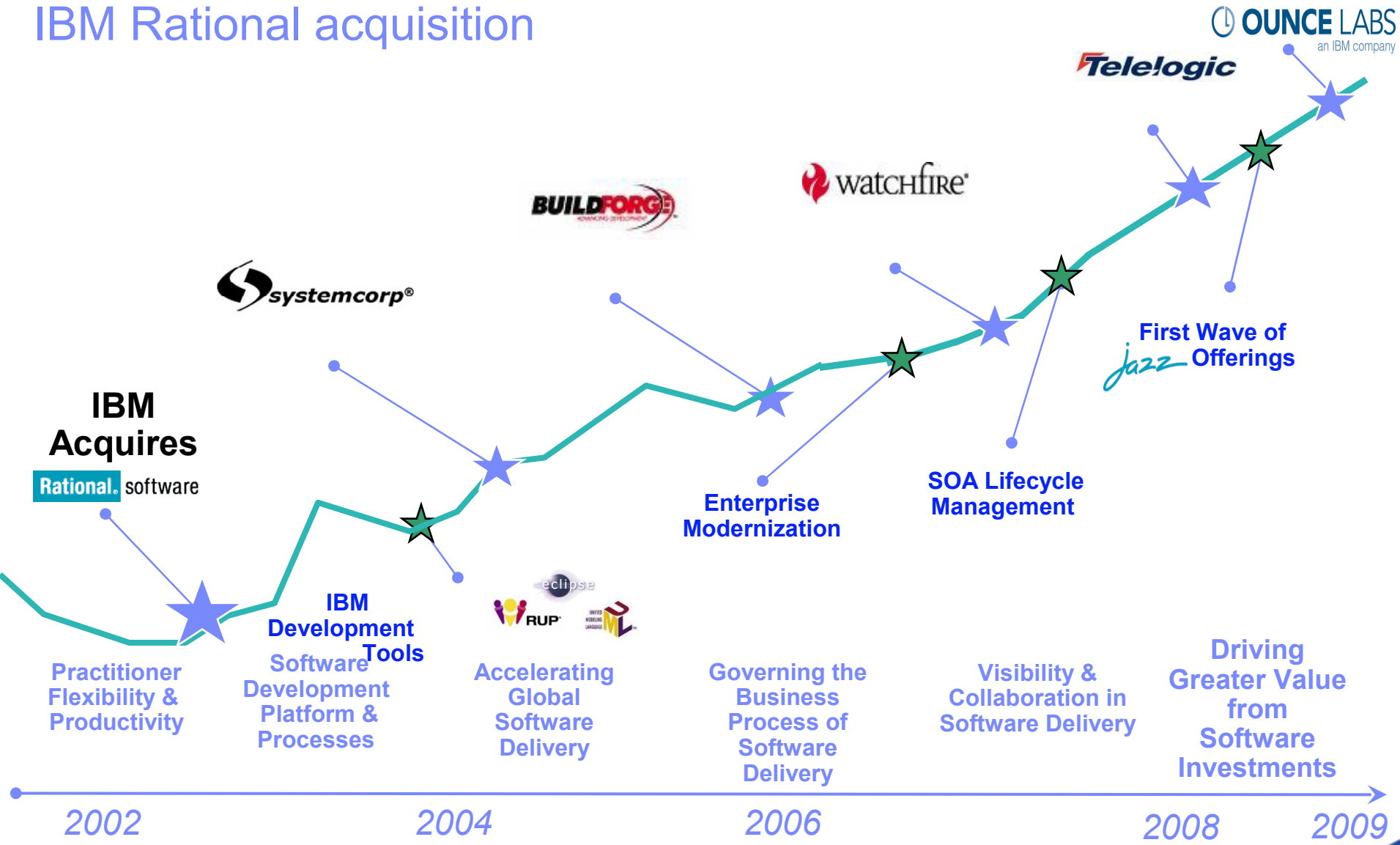
- Scenario (in-house, outsourced, auction contracted..)
- Teams (One site to multi-site..)
- Roles (Developer, Tester, Architect, Deployment/Production, customer/client...)
- Methodology (iterative, agile, MDD...)
- Crosscutting Concerns (Fidelity to requirements, process...)

Software Delivery & Business Domain



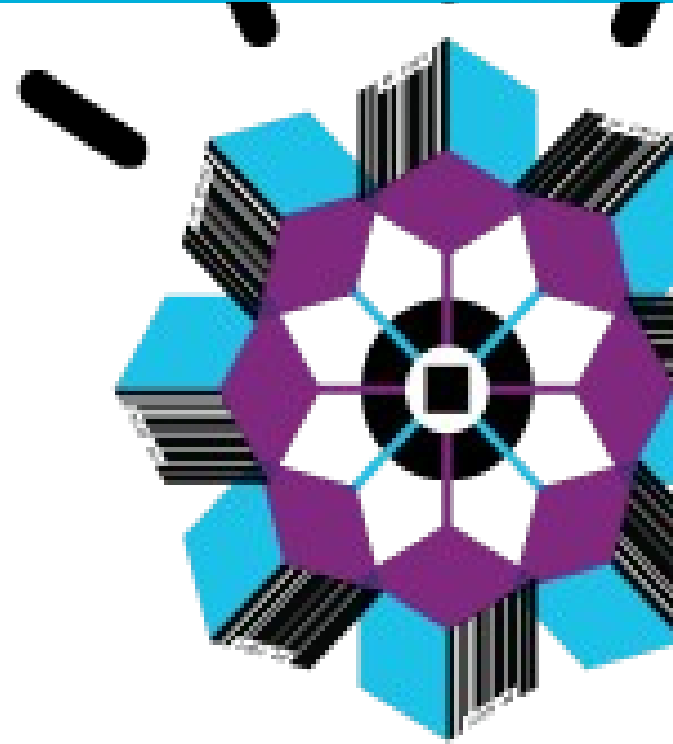
A common platform provides an advantage!

IBM Rational acquisition



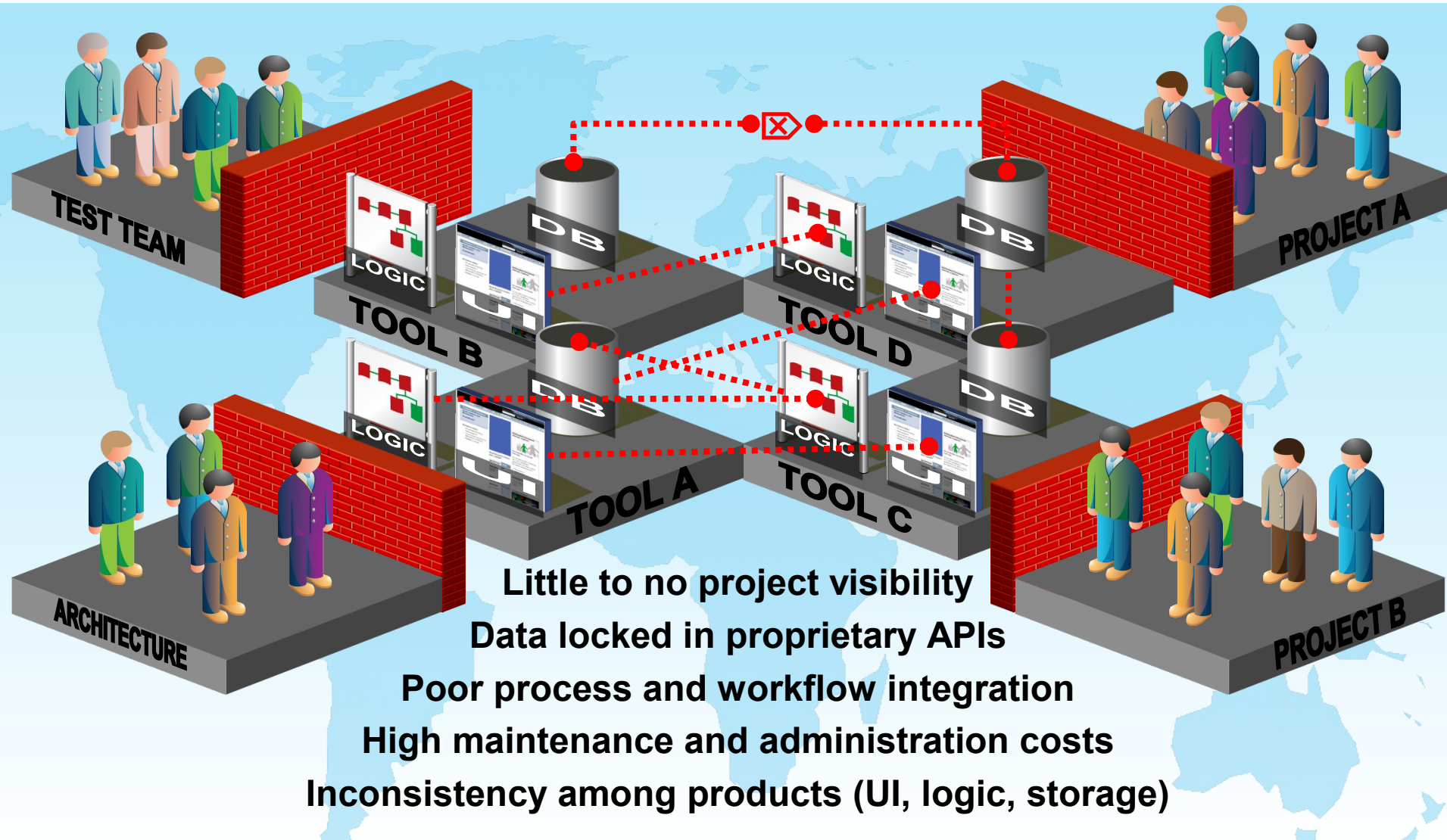
Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Key Product Direction
- In summary

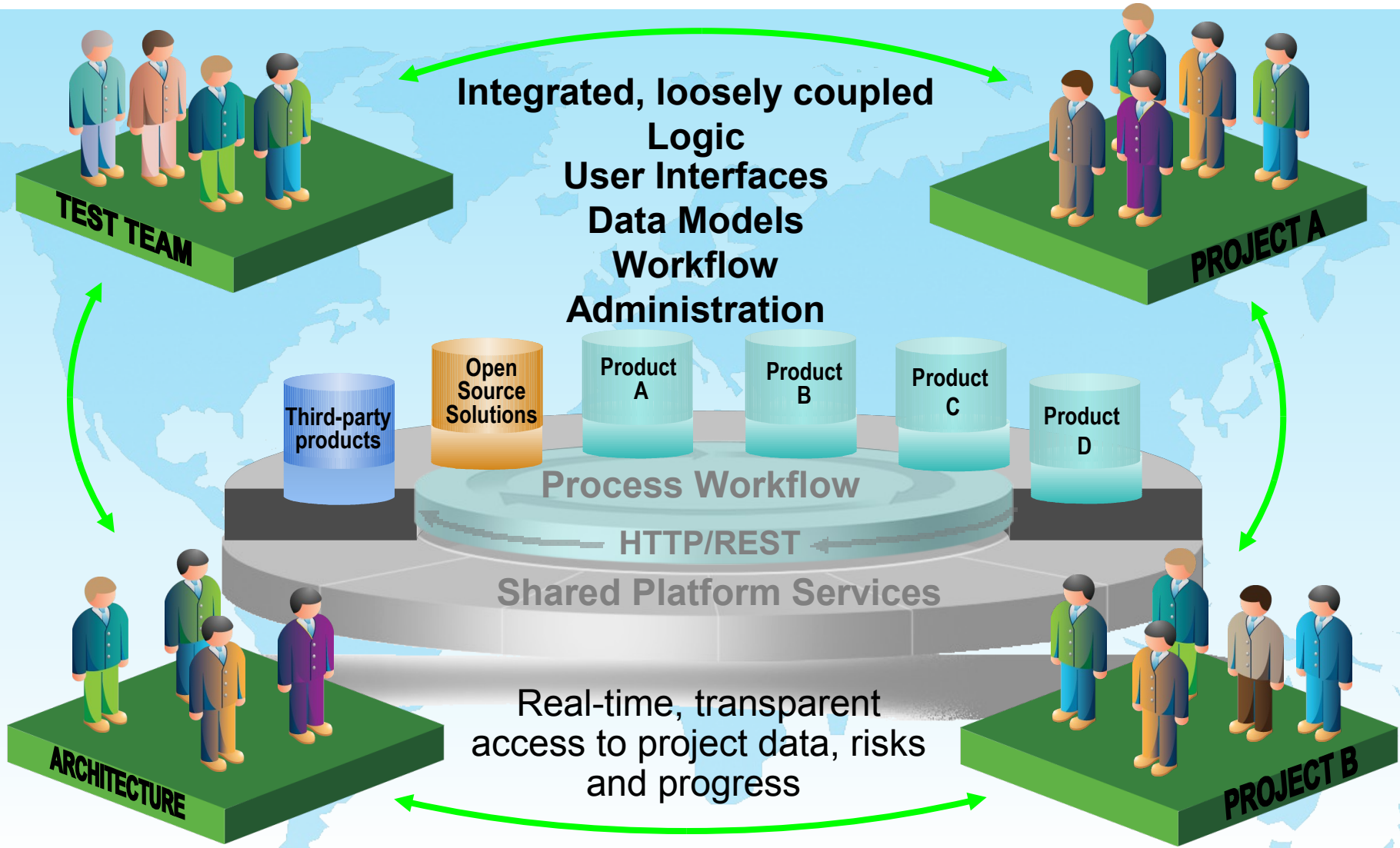


Traditional software delivery environments reinforce the problem

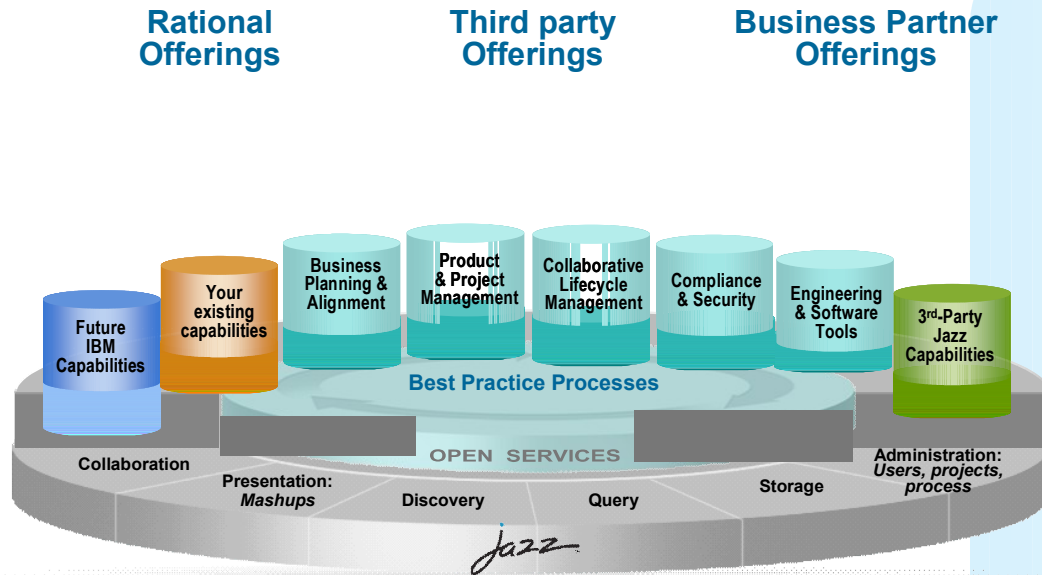
Heterogeneous environments, multiple teams, diverse systems



An open but integrated software delivery platform can break down organizational, functional and technology barriers



The IBM Jazz Platform – Delivering Agility



■ **Jazz is...**

- Open Services for Lifecycle Collaboration (OSLC)
- “Eclipse for the Team” on the web
- Integration of ANYTHING across the lifecycle
- Integration architecture enabling mashups and non-Jazz products to participate
- A community at Jazz.net where Jazz products are built

Collaborate, Automate, Report / Analyze



Open Services for Lifecycle Collaboration

Simplifying collaboration across the software delivery lifecycle



Open Services for Lifecycle Collaboration

Open interfaces. Open possibilities.

An industry initiative for making it easier to use software delivery tools in combination.

215+ registered community members.

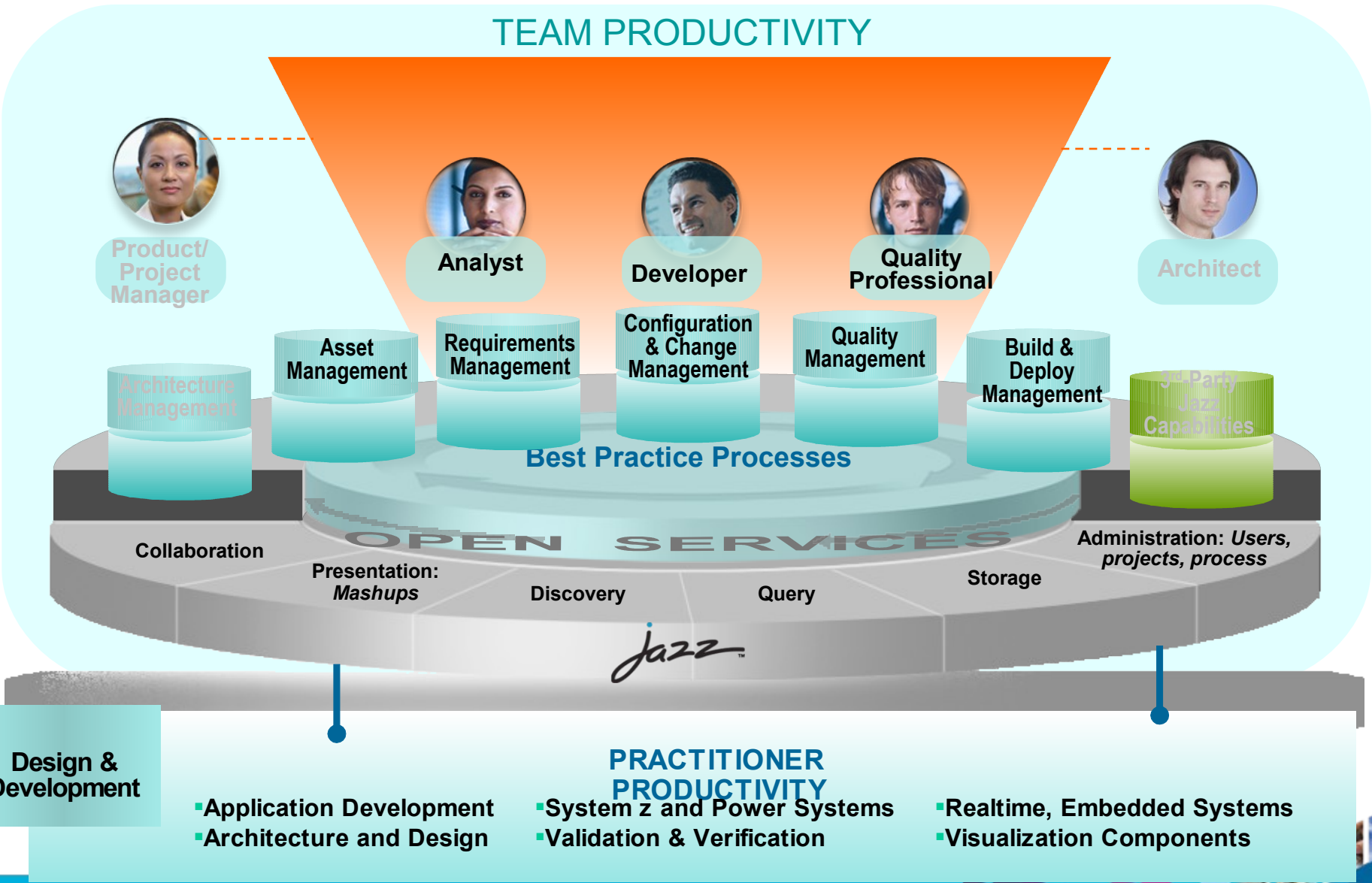
Barriers to sharing resources across the software lifecycle

- ▶ Multiple vendors, open source projects, and in-house tools
- ▶ Private vocabularies, formats and stores
- ▶ Entanglement of tools and data

▶ The Open Services initiative is

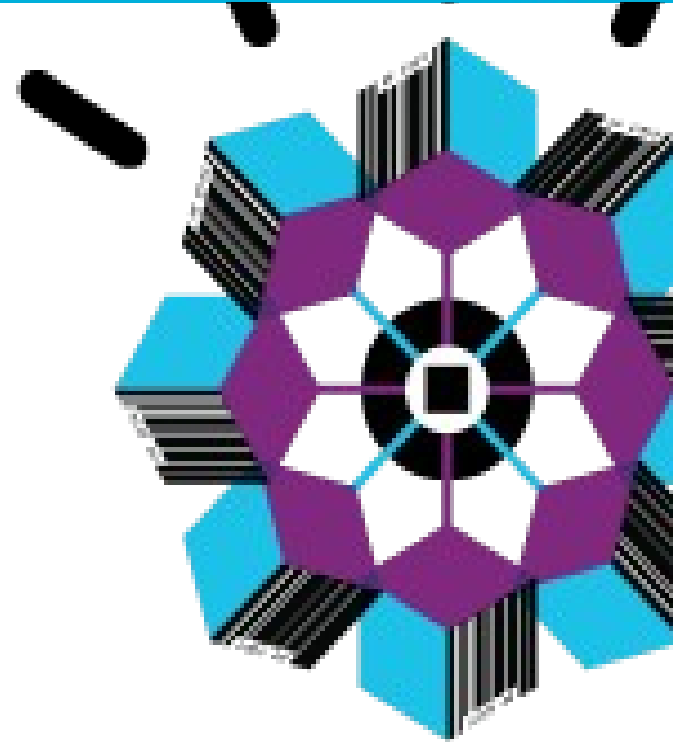
- **Building** a community of software vendors, open source projects, integrators, and corporate IT teams, operating at **open-services.net**
- **Creating** public specifications of resources and services for sharing the things that software teams rely on, like change requests, test cases, defects, requirements and user stories
- **Delivering** loosely coupled resource formats and services with “just enough” standardization

Today: Collaborative Lifecycle Management



Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Direction
 - ▶ Workbench for Collaborative Lifecycle Management
- In summary



Evolution of the Jazz platform

2008

2009

2010 and Beyond

1. Create the platform, introduce new products

Rational.

Team Concert 1.0.1

Revolutionary Collaborative Development Environment

Rational.

Quality Manager

Collaborative Business-driven Quality

Rational.

Requirements Composer

Business Expert Collaboration on requirements

Rational.

Open Services for Lifecycle Collaboration

Community specifications for integration

2. Surf the CLM Web

Rational.

CLM 2009

Team Concert 2.0.0.2
Quality Manager 2.0.0.1
Requirements Composer 2.0

Rational.

Project & Portfolio Management

Project Conductor, Insight, MCIF / Deployment Pkg

Rational.

Jazz Integrations

ClearQuest Bridge, ClearCase Bridge, Asset Manager, Build Forge

Rational.

Open Services for Lifecycle Collaboration

Change Management 1.0 specification & implementations

3. Scale and Simplicity

Rational.

CLM 2010

Collaborative ALM @ Scale

Rational.

Jazz Integrations

Expanded ecosystem extensions to Collaborative ALM @ Scale

Rational.

Foundation

Integration Services reducing total cost of ownership

Rational.

Jazz Modeling

Modeling, Jazz Automation Framework, Expanded EM capabilities

Next: Rational Workbench for Collaborative Lifecycle Management



Jazz Project

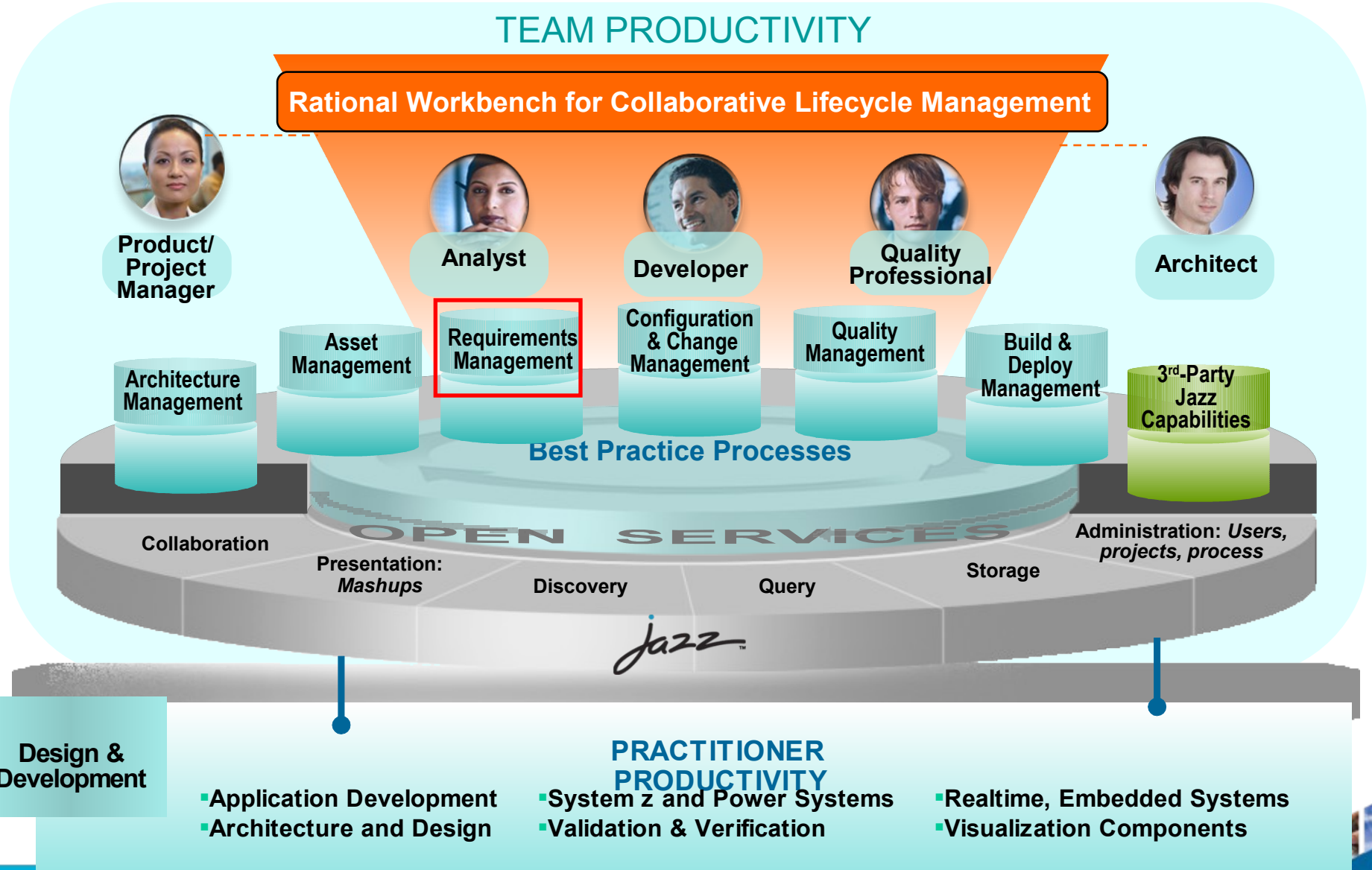
Rational Workbench for Collaborative LifeCycle Management

The screenshot shows a web browser window displaying the project page for the Rational Workbench for Collaborative Lifecycle Management. The page features a navigation menu at the top with options like Home, About Jazz, Roles, Projects, Downloads, Forums, Library, Development, and Community. The main content area includes a sidebar with links to Overview, Features, Product Versions, and Library. The central content area has a heading "Rational Workbench for Collaborative Lifecycle Management" with a sub-heading "Weave a 'web' of ALM resources". Below this is a paragraph describing the project's focus on coordinating software development activities. A section titled "What are we doing in 2010?" lists updates such as Rational Team Concert, Rational Quality Manager, and Rational DOORS Requirements Professional*. At the bottom of the main content area, there is a banner for the project with three circular profile pictures of team members labeled "Developer" and "Quality". On the right side, there are sections for "Join the conversation" (with links to forum and mailing lists), "Participate" (with links to bug reports, dashboard, iteration plans, and wiki), and "Learn" (with a list of recent library items).

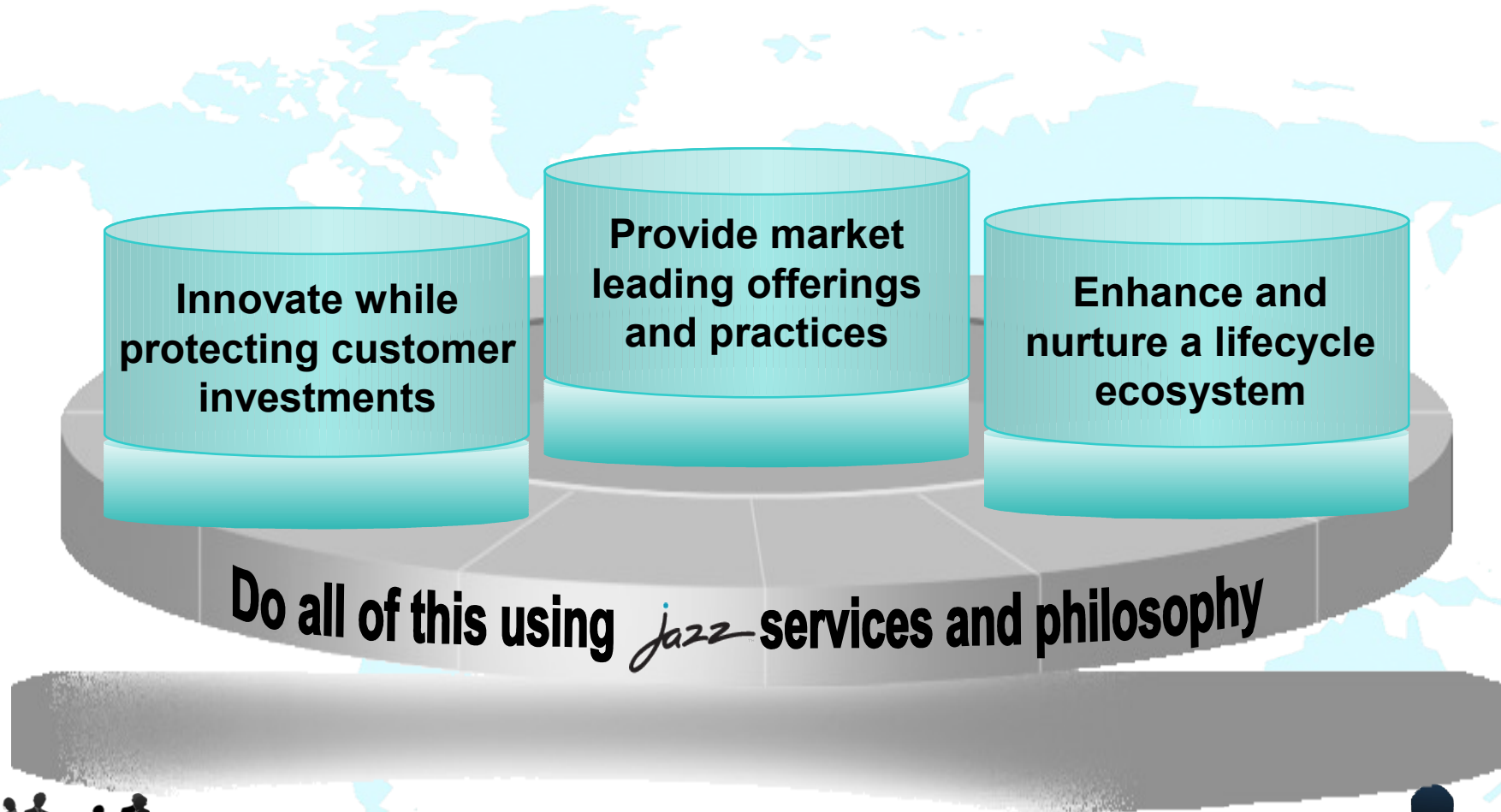
Jazz Project: Workbench for CLM (Beta)



Rational Requirement

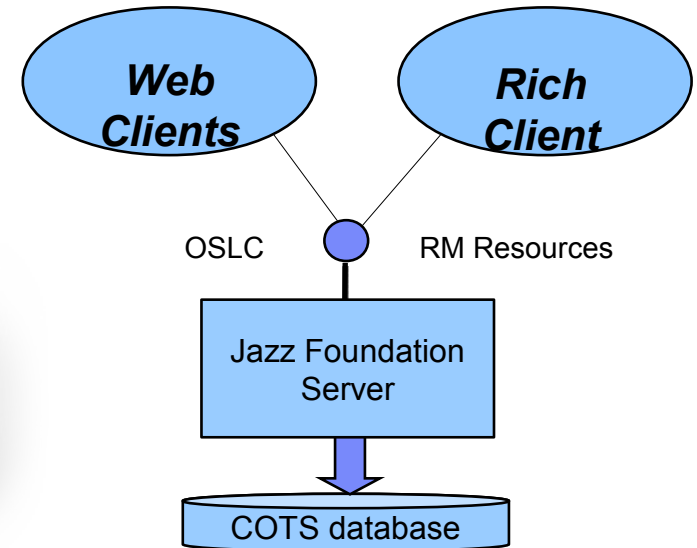
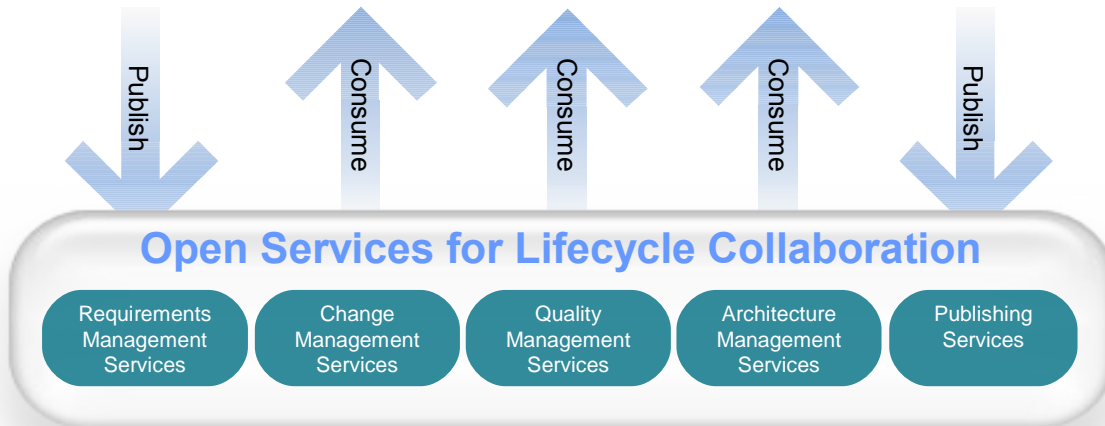


The IBM Rational requirements management strategy



Vision for requirements management solution architecture

- **Improve team productivity and ease of use** with integrated requirements definition and management
- **Choose the requirements capabilities** that fits your teams' needs
- **Clear migration paths** to the latest product versions
- **Seamless requirements visibility and traceability across the lifecycle** with an open integration architecture



Jazz Project: Rational Requirement

The screenshot shows a web browser window displaying the Rational Requirements Composer project page on the Jazz Community Site. The browser's address bar shows the URL <http://jazz.net/projects/rational-requirements-composer/>. The website header includes navigation links for Home, About Jazz, Roles, Projects, Downloads, Forums, Library, Development, and Community, along with Register and Log In options.

The main content area features the Rational Requirements Composer logo and the title "Rational Requirements Composer / Rational DOORS Requirements Professional". Below the title is the tagline "Collaborative requirements for the whole team". The text describes the platform as a complete next-generation requirements management environment for better requirements practices and to enable larger sets of stakeholders and team members to participate in a requirements driven development life cycle. It highlights that the platform is intended to deliver proven requirements management methods and capabilities for fast-paced, agile-at-scale project teams.

Key features and resources listed include:

- Download 2.0.0.2 iFix 1** (dated June 30, 2010)
- Download 3.0 RC0 (Beta 2)** (dated September 20, 2010), described as a pre-release Early Access Trial milestone.
- Join the conversation** options: Ask questions in the forum and Subscribe to mailing lists.
- Participate** section: We use our own tools to openly develop this project. Track our progress or get involved.

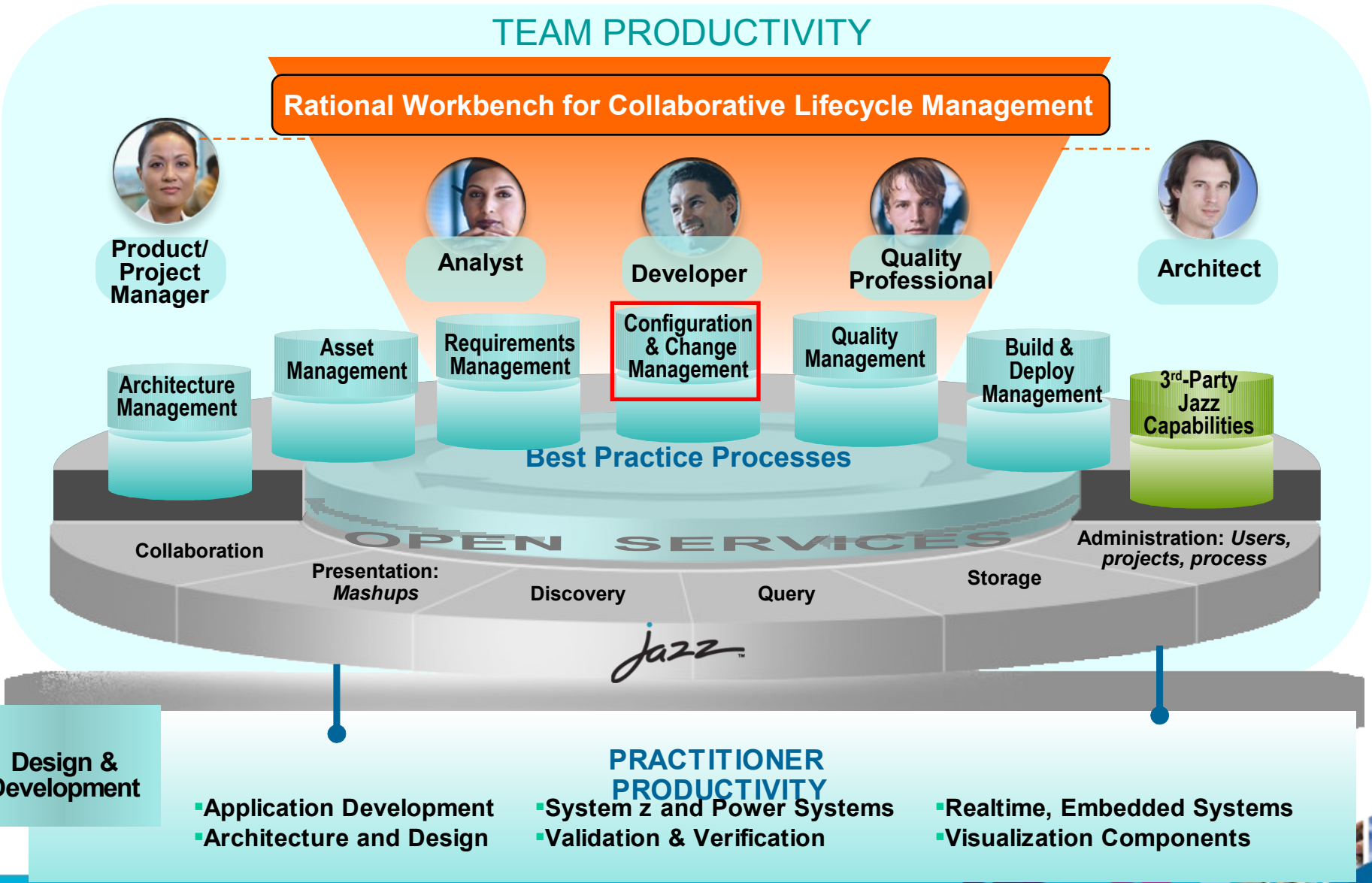
A sidebar on the left provides navigation for "Projects" (Overview, What's New, Integrations **New!**, Release Plan) and "Related Resources" (Rational Requirements Composer / Rational DOORS Requirements Professional 2.0.0.2 iFix 1 Online Help, Library).

A tooltip at the bottom of the page reads: "Slide Show from current slide (Shift+F5)".

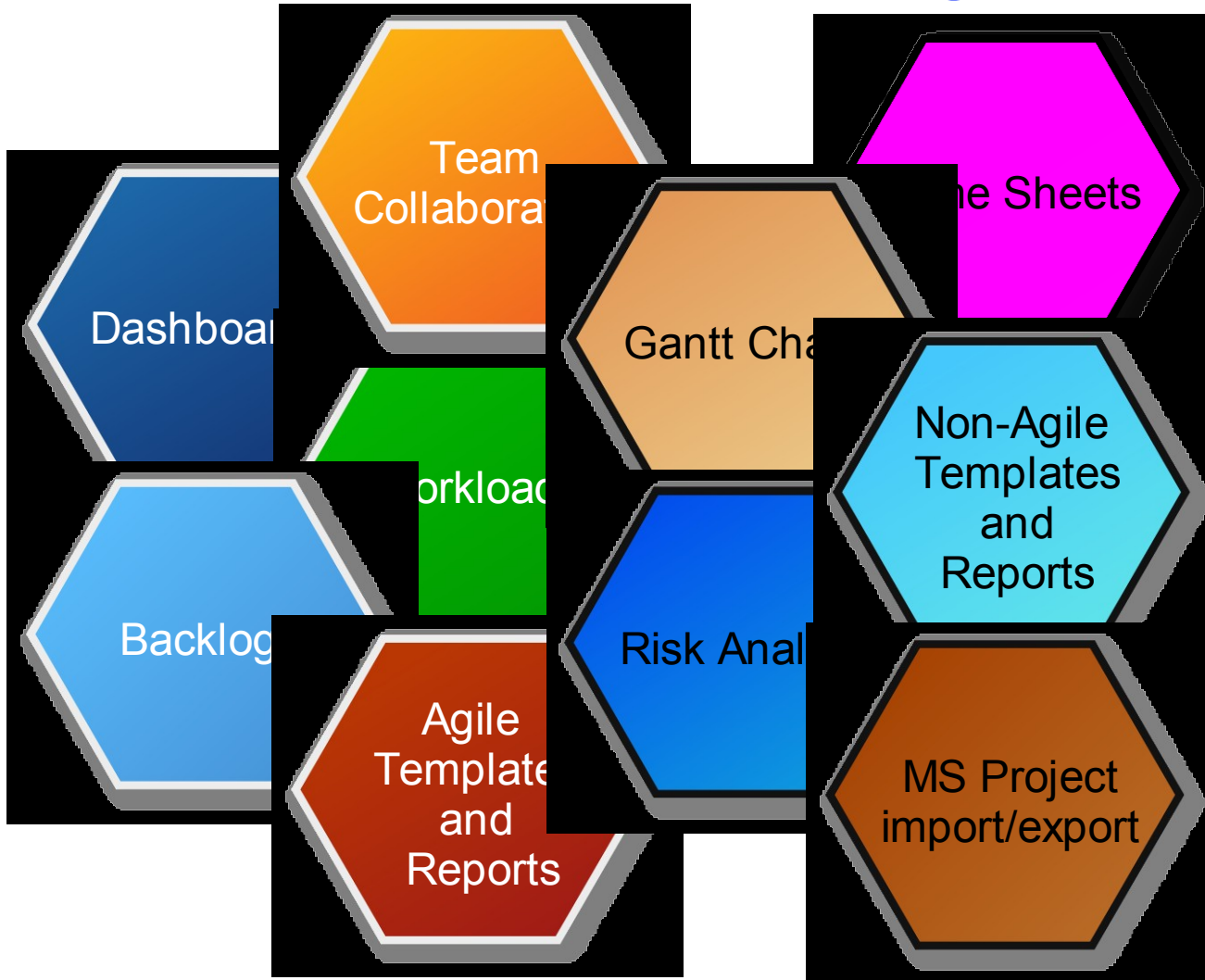
Jazz Project: Rational Requirement (Beta)



Rational Team Concert



Rational Team Concert 3.0 Planning enhancements



Allows Agile and Traditional teams to work together

Eases the introduction of Agile techniques

Supports Larger Teams



Jazz Project

Rational Team Concert v3.0

The screenshot shows a Mozilla Firefox browser window displaying the Rational Team Concert project page. The browser's address bar shows the URL <http://jazz.net/projects/rational-team-concert/>. The page features a navigation menu with options like Home, About Jazz, Roles, Projects, Downloads, Forums, Library, Development, and Community. The main content area is titled "Rational Team Concert" and includes a sidebar with navigation links such as Overview, Features, Learn More, What's New, Integrations (marked as New!), Assessing Your Needs, Supported Platforms (marked as New!), and Related Resources. The main text describes the product as "A collaborative software development environment" and includes a quote: "What if my team could collaborate on plans, tasks, code, and builds ... all in one place?". It also mentions that the product integrates work item tracking, source control management, continuous builds, iteration planning, and a highly configurable process support. A "What's coming in our next release?" section highlights changes for the next release, planned for 4Q 2010, including updates to the Rational Workbench for Collaborative Lifecycle Management and packaging/licensing changes. A "Buy at ibm.com" button is visible, along with a "Download 2.0.0.2 iFix 4" button dated September 9, 2010. Other download links include "Download 3.0 RC1" dated October 8, 2010, and a "Try it in the Sandbox" option. The page also includes a "Join the conversation" section with links to the forum, online help, and mailing lists, and a "Participate" section.

Jazz Project: Rational Team Concert v3.0 (Beta)



Rational for Collaborative Lifecycle Management

Track: La piattaforma IBM di Collaborative Lifecycle Management per lo sviluppo software multiplatforma: TivoliLAB
Speaker: Silvia Giaccone, Mariangela Orme

Rational for Collaborative Lifecycle Management



Design & Development

- Application Development
- Architecture and Design

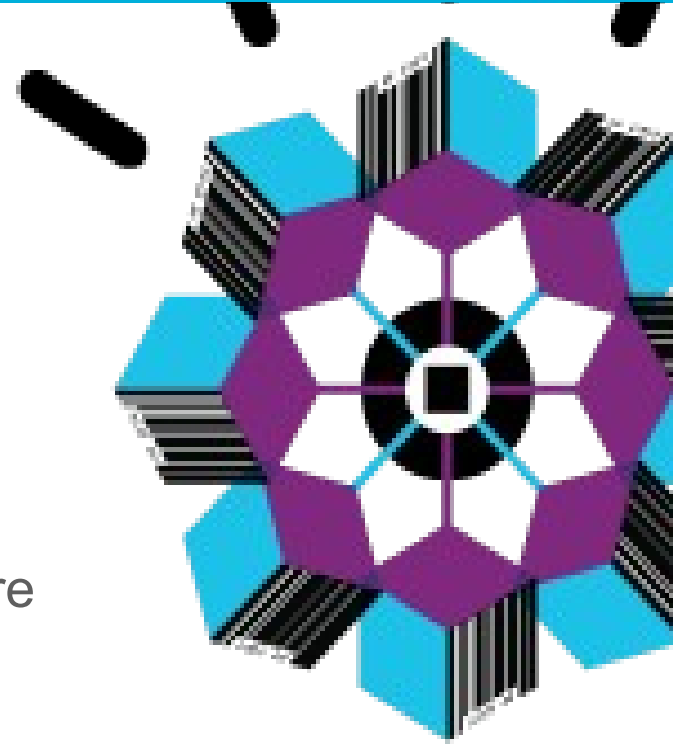
PRACTITIONER PRODUCTIVITY

- System z and Power Systems
- Validation & Verification

- Realtime, Embedded Systems
- Visualization Components

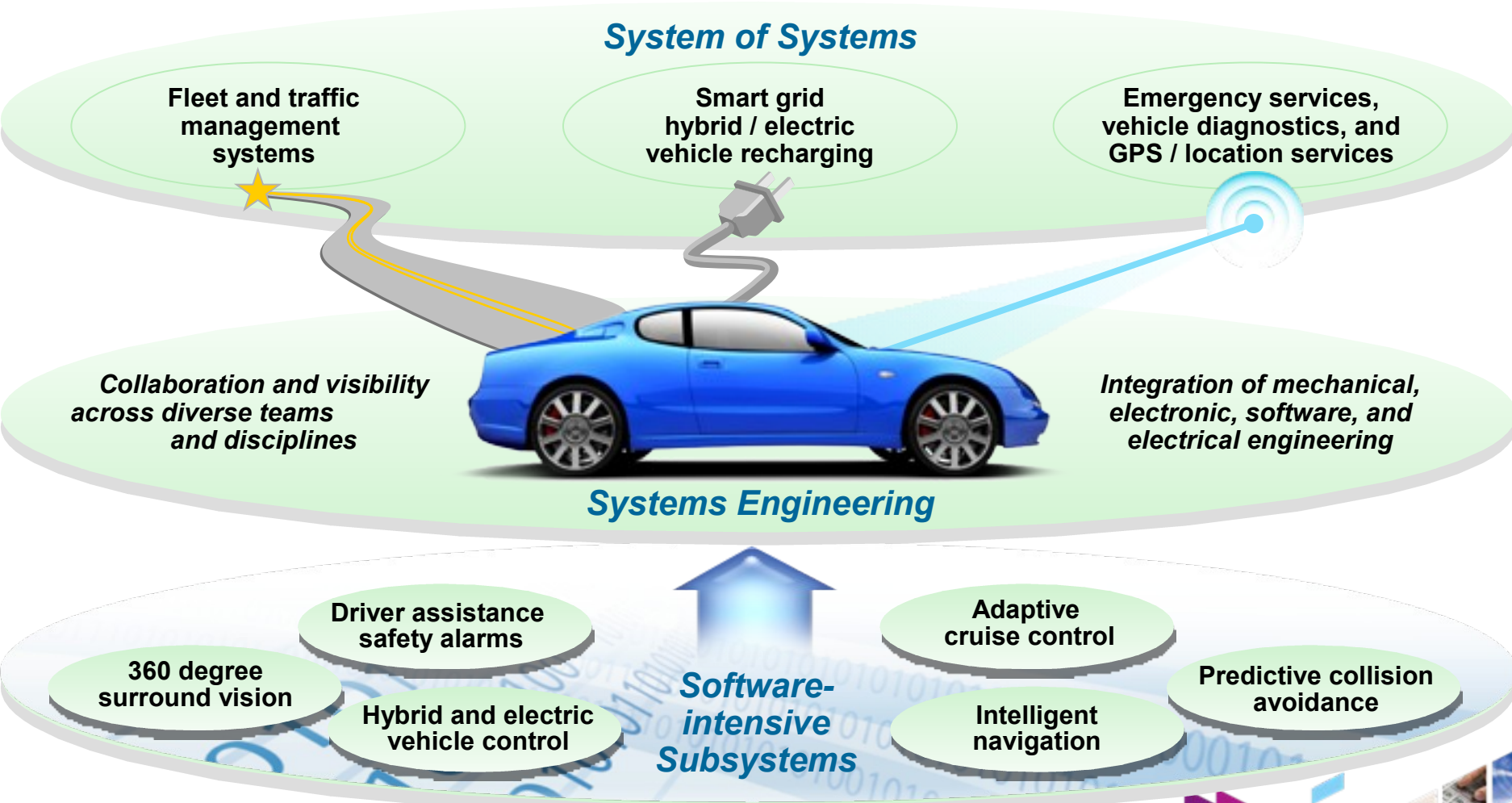
Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Direction
 - ▶ Rational Workbench for Systems and Software Engineering
- In summary



Smart products example

From sophisticated in-vehicle software, to complex “system of systems” ecosystems, cars are getting smarter every day



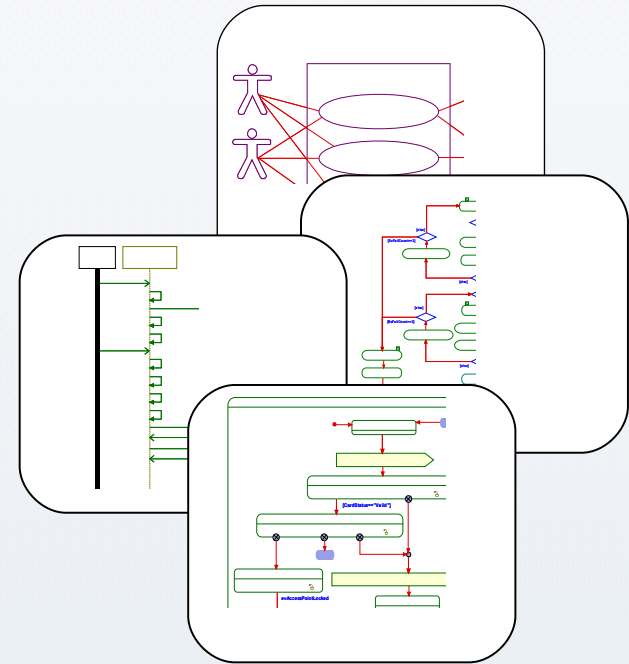
Modern Approaches for Describing Systems Are Evolving *To Better Manage Complexity and Reduce Time-to-market*

Past



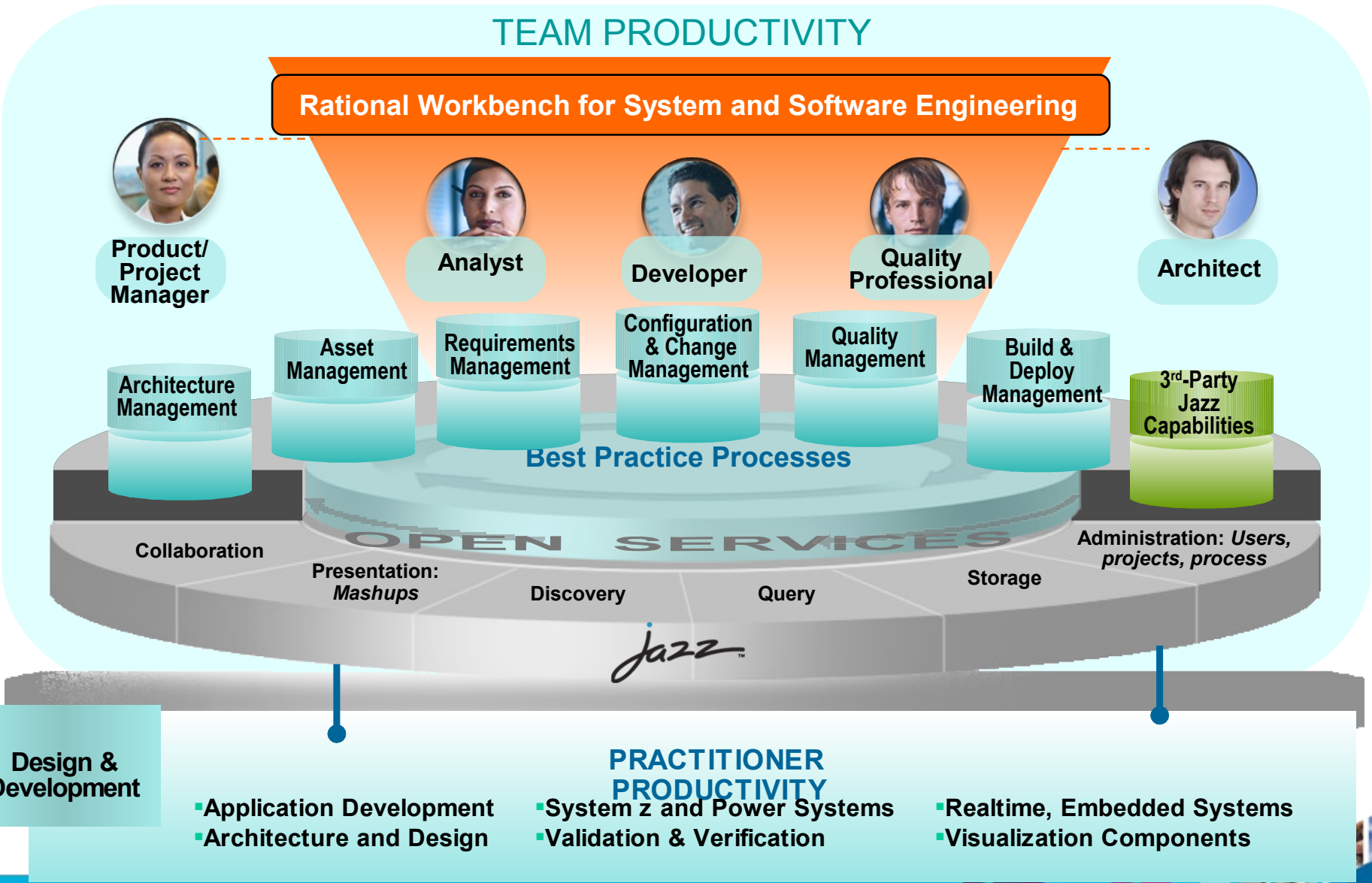
- Specifications
- Interface requirements
- System design
- Analysis & trade-off
- Test plans

Future



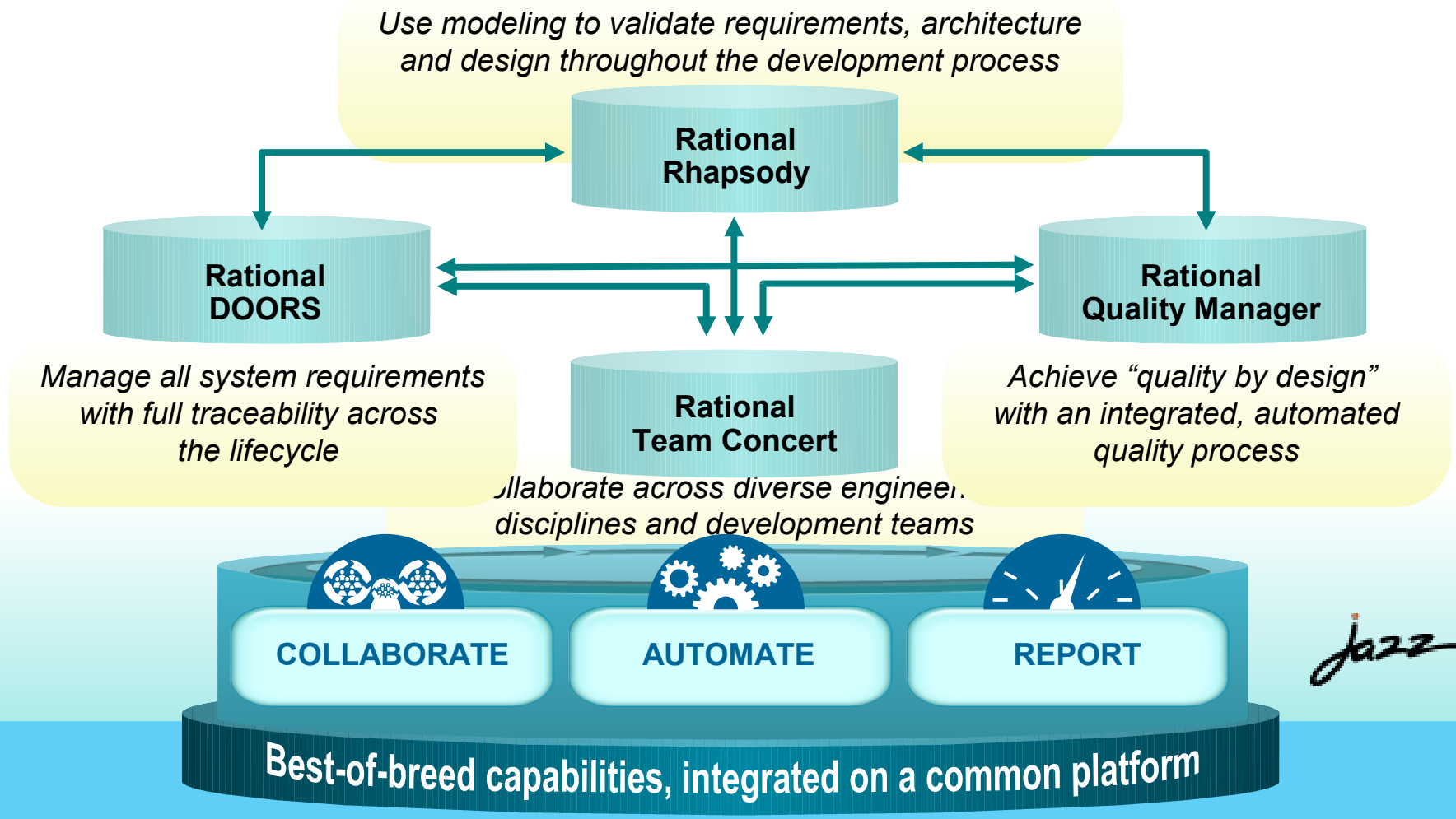
Moving from manual methods to a model-driven approach

Rational Workbench for Systems and Software Engineering



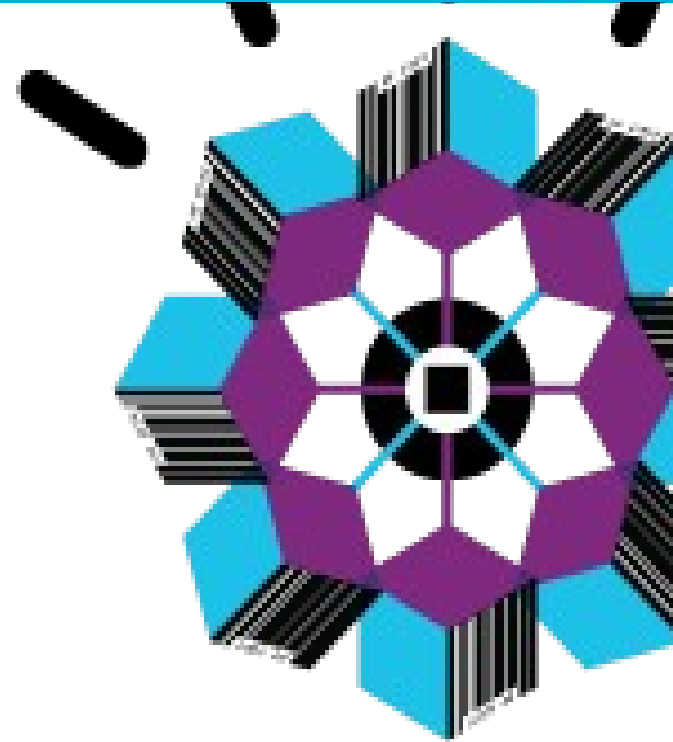
Rational Workbench for Systems and Software Engineering

Track: La piattaforma per lo sviluppo software nel mondo System
Speaker: Paolo Cravino – Alberto Sardini



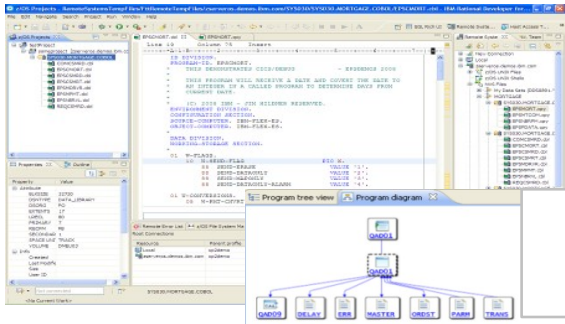
Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Direction
 - ▶ Collaborative Lifecycle Management Platform for Enterprise Modernization
- In summary



Enterprise Modernization

Rational Development Tools **support the lifecycle of application development** and maintenance for zEnterprise System applications on **z/OS, Linux for System z, AIX, and x86 Linux** using a combination of tools running on a variety of platforms to best suit the client's requirements



Rational Developer for System z and Power Systems

IDE for COBOL, PL/I, HLASM, C/C++, Java and JCL
 Increase productivity
 Reduce MIPS

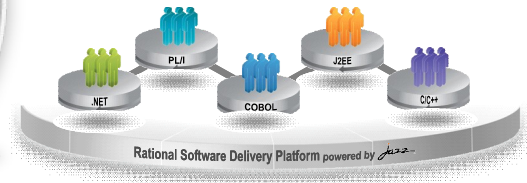
Rational Asset Analyzer

Dramatically improve productivity
 Better manage quality



Rational Developer for System z Unit Test

Rapidly prototype z/OS apps
 High fidelity Unit Test
 Unit test without consuming valuable production MIPS

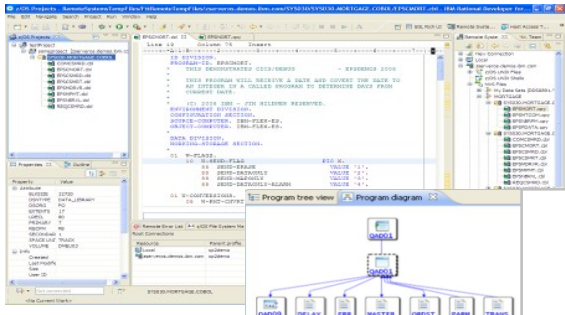


Rational Team Concert for System z

Common repository across multiple platforms & languages
 Different processes by developer type
 Integrated or loosely federated processes across all development groups

Enterprise Modernization

Rational Development Tools **support the lifecycle of application development** and maintenance for zEnterprise System applications on **z/OS, Linux for System z, AIX, and x86 Linux** using a combination of tools running on a variety of platforms to best suit the client's requirements

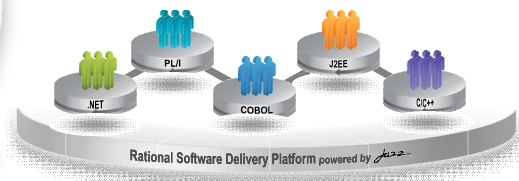


Rational Developer for System z and Power Systems
 IDE for COBOL, PL/I, HLASM, C/C++, Java and JCL
 Increase productivity
 Reduce MIPS

Rational Asset Analyzer
 Dramatically improve productivity
 Better manage quality



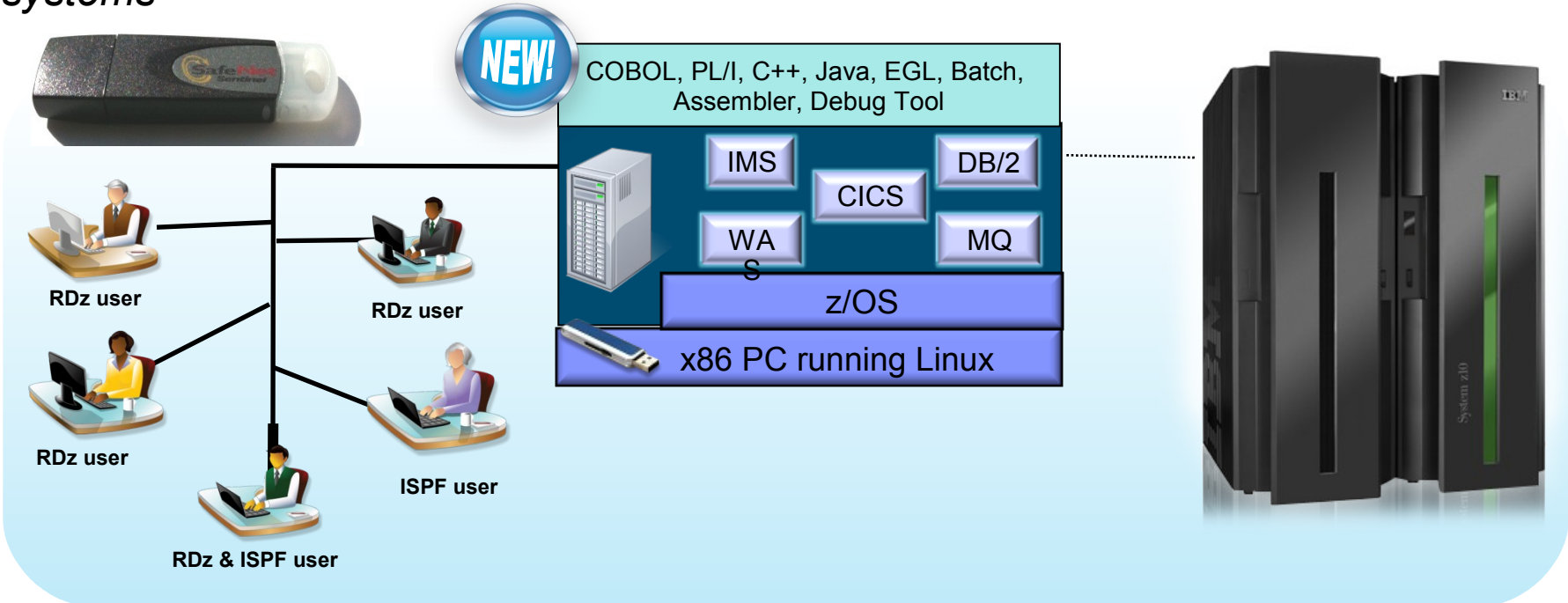
Rational Developer for System z Unit Test
 Rapidly prototype z/OS apps
 High fidelity Unit Test
 Unit test without consuming valuable production MIPS



Rational Team Concert for System z
 Common repository across multiple platforms & languages
 Different processes by developer type
 Integrated or loosely federated processes across all development groups

System z environment “on a stick”

Rational Developer for z Unit Test: A System z environment for testing on x86 Linux systems

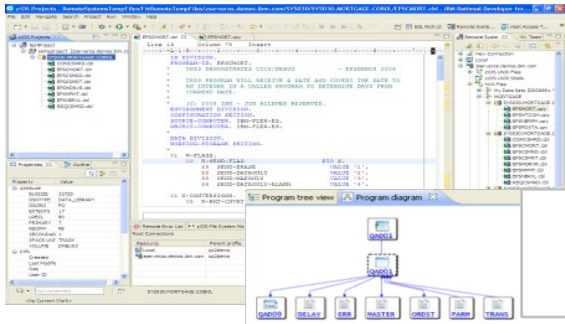


- Liberate developers to rapidly prototype new applications Develop and test System z applications anywhere, anytime
- Free up mainframe development MIPS for production capacity
- Eliminate costly delays by reducing dependencies on operations staff

Note: This Program is licensed only for development and test of applications that run on IBM z/OS. The Program may not be used to run production workloads of any kind, nor more robust development workloads including without limitation production module builds, pre-production testing, stress testing, or performance testing.

Enterprise Modernization

Rational Development Tools **support the lifecycle of application development** and maintenance for zEnterprise System applications on **z/OS, Linux for System z, AIX, and x86 Linux** using a combination of tools running on a variety of platforms to best suit the client's requirements



Rational Developer for System z and Power Systems

IDE for COBOL, PL/I, HLASM, C/C++, Java and JCL
 Increase productivity
 Reduce MIPS

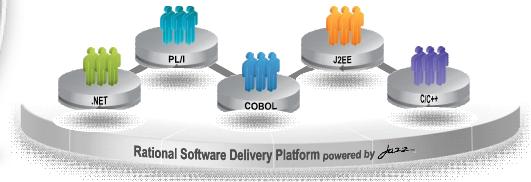
Rational Asset Analyzer

Dramatically improve productivity
 Better manage quality



Rational Developer for System z Unit Test

Rapidly prototype z/OS apps
 High fidelity Unit Test
 Unit test without consuming valuable production MIPS



Rational Team Concert for System z

Common repository across multiple platforms & languages
 Different processes by developer type
 Integrated or loosely federated processes across all development groups

Jazz Project: Rational Team Concert for System Z v3.0 (Beta)

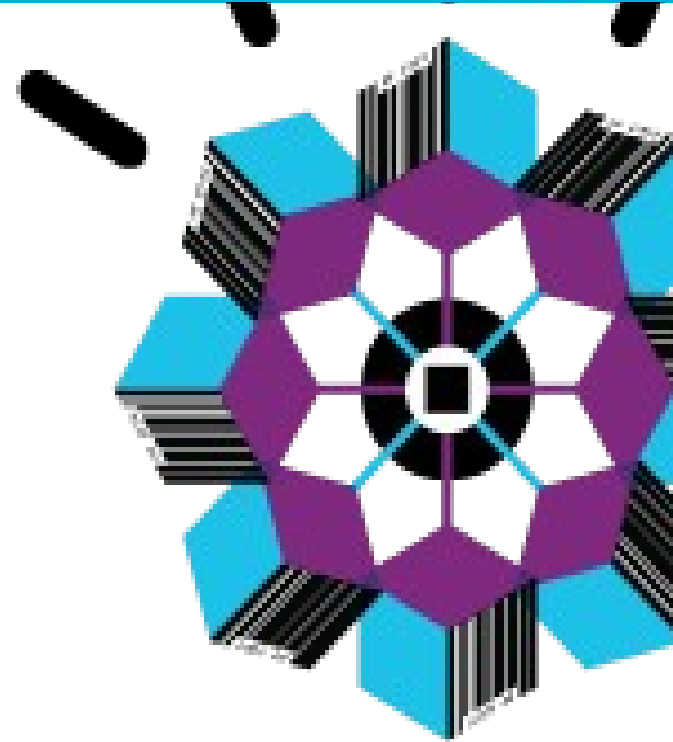
The screenshot shows the Rational Team Concert project page on the Jazz Community Site. The page is titled "Rational Team Concert - Projects - Jazz Community Site - Mozilla Firefox". The main content area features a quote: "What if my team could collaborate on plans, tasks, code, and builds ... all in one place?" and a description of the product's capabilities. There are also download links for version 2.0.0.2 iFix 4 and 3.0 RC1, and a "Buy at ibm.com" button.

- ISPF Client
 - ▶ SCM Client only
- Dependency Based Build
 - ▶ Build affected modules only
 - ▶ Based on dependency meta data collection
 - ▶ Uses Build Forge agent
 - ▶ Generic support for all platforms (direction – starting with z/OS and IBM i)
- Promotion and Deployment
 - ▶ Promotion scoped at component level
 - Focused on deployment for z/OS



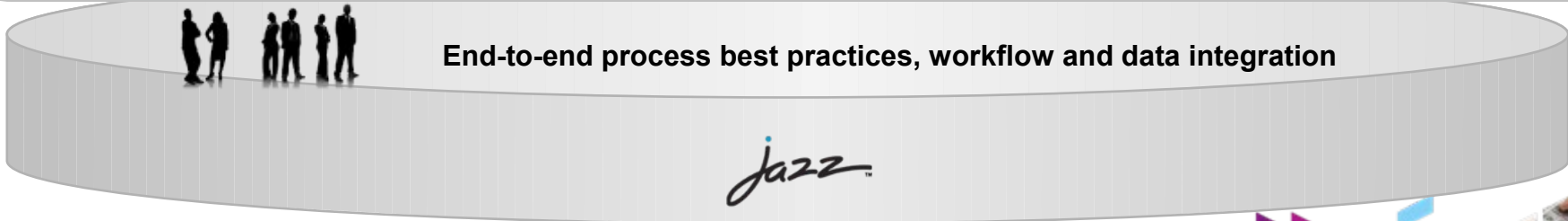
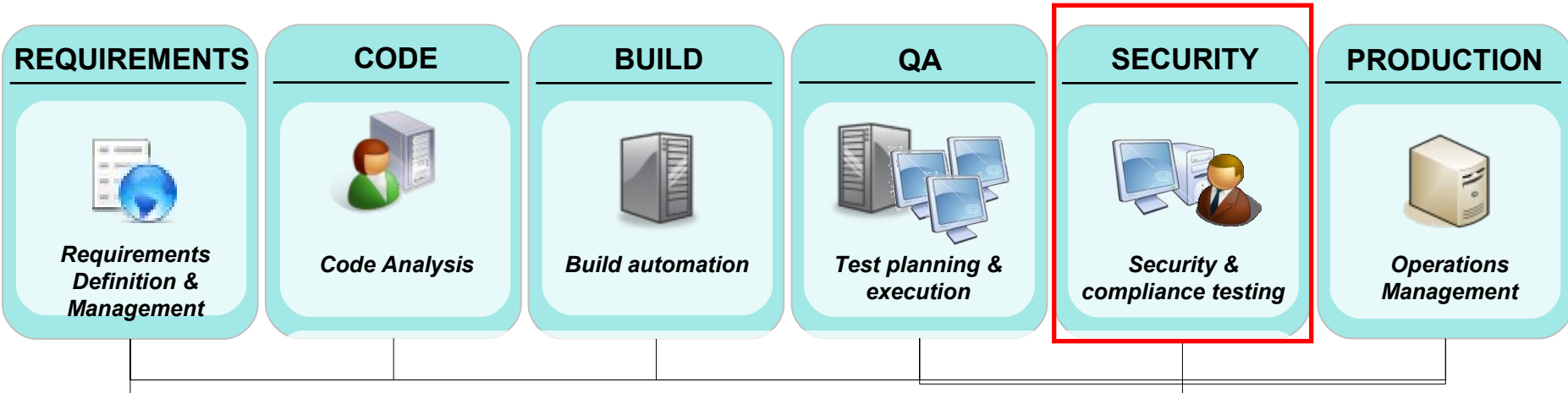
Agenda

- Software Delivery Challenges
- The Jazz Delivery Platform
- Direction
 - ▶ Security
- In summary

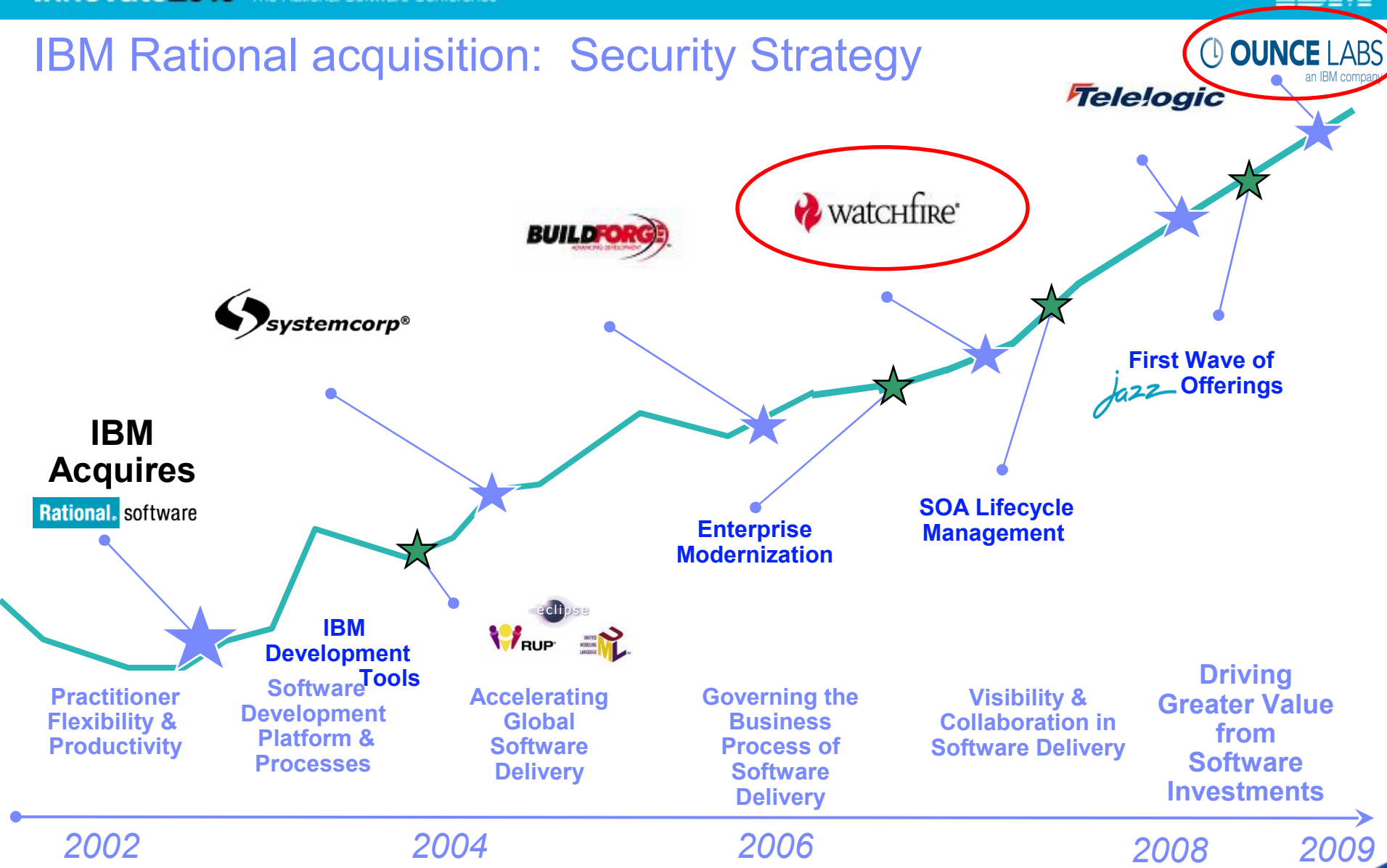


Quality in Software Delivery Lifecycle

End-to-end process

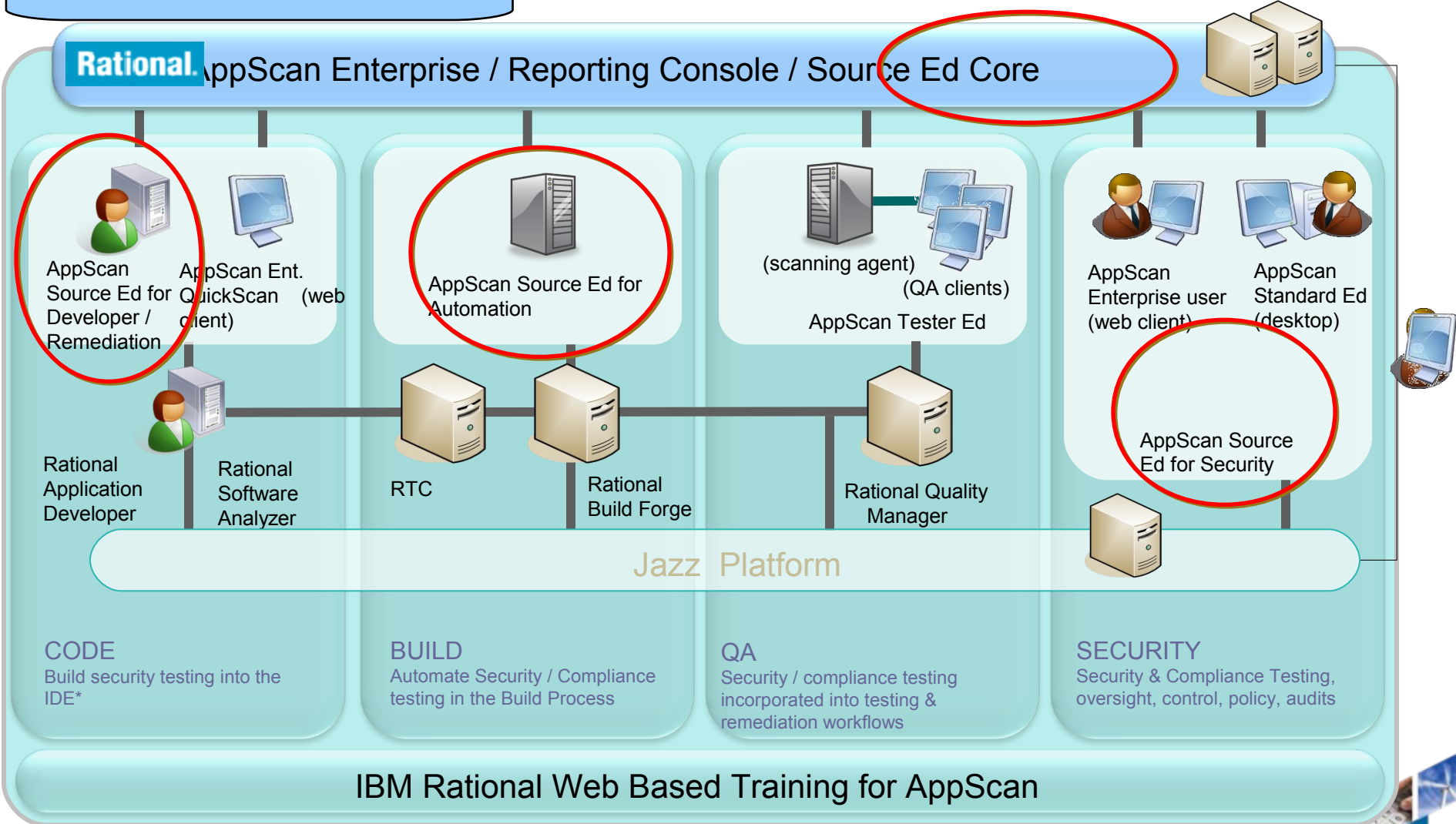


IBM Rational acquisition: Security Strategy



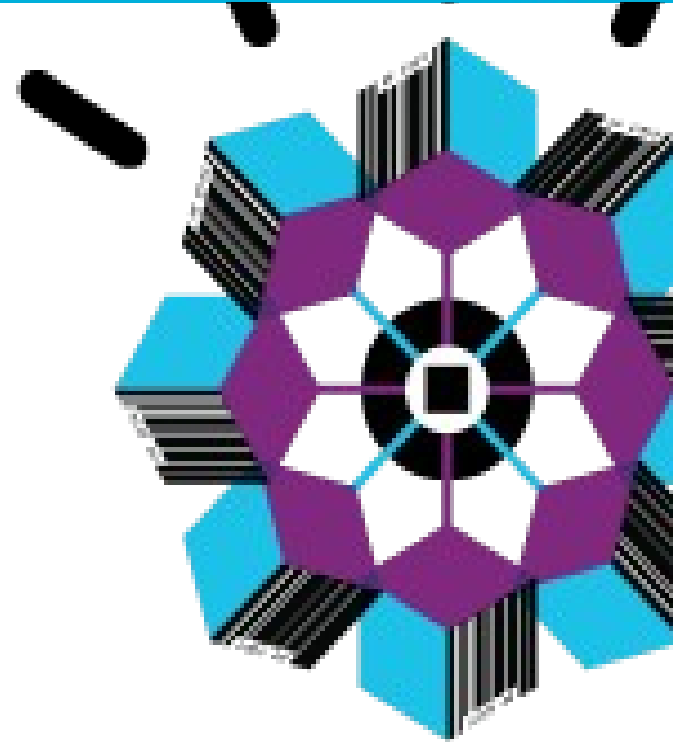
IBM Rational AppScan Ecosystem

Track: La soluzione end to end per la Sicurezza
 Speaker: Massimo Caprinali, Simone Riccetti



Agenda

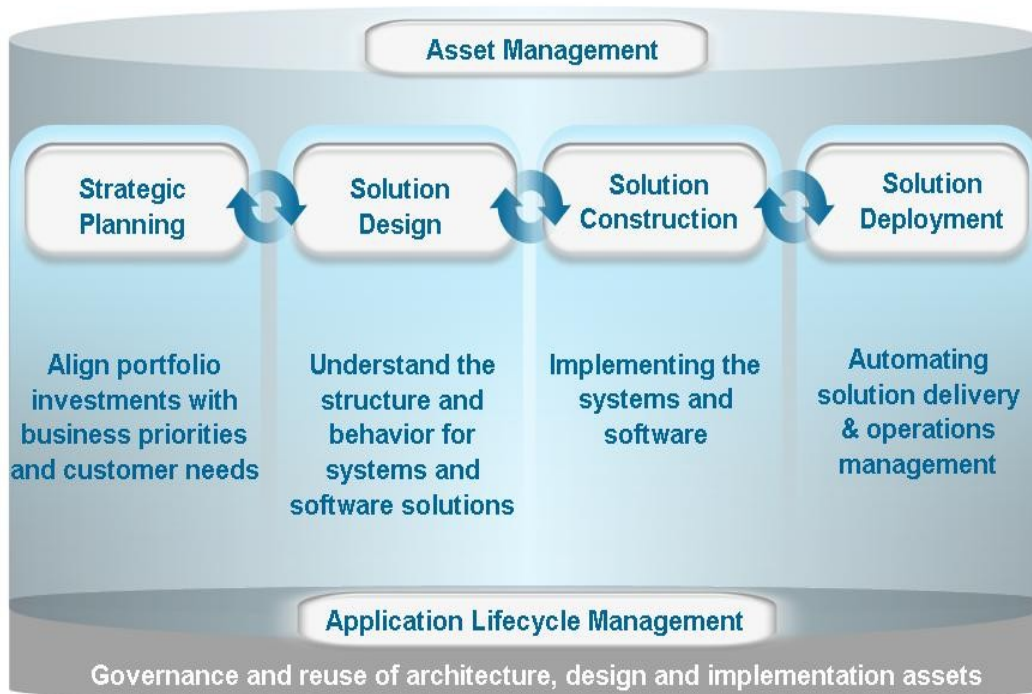
- Software Delivery Challenges
- The Jazz Delivery Platform
- Direction
 - ▶ Enterprise Architecture
- In summary



Achieving Business Results From Your Enterprise Architecture

Collaborative Planning through Deployment - risk management enabled through traceability

Collaborative Architecture Management



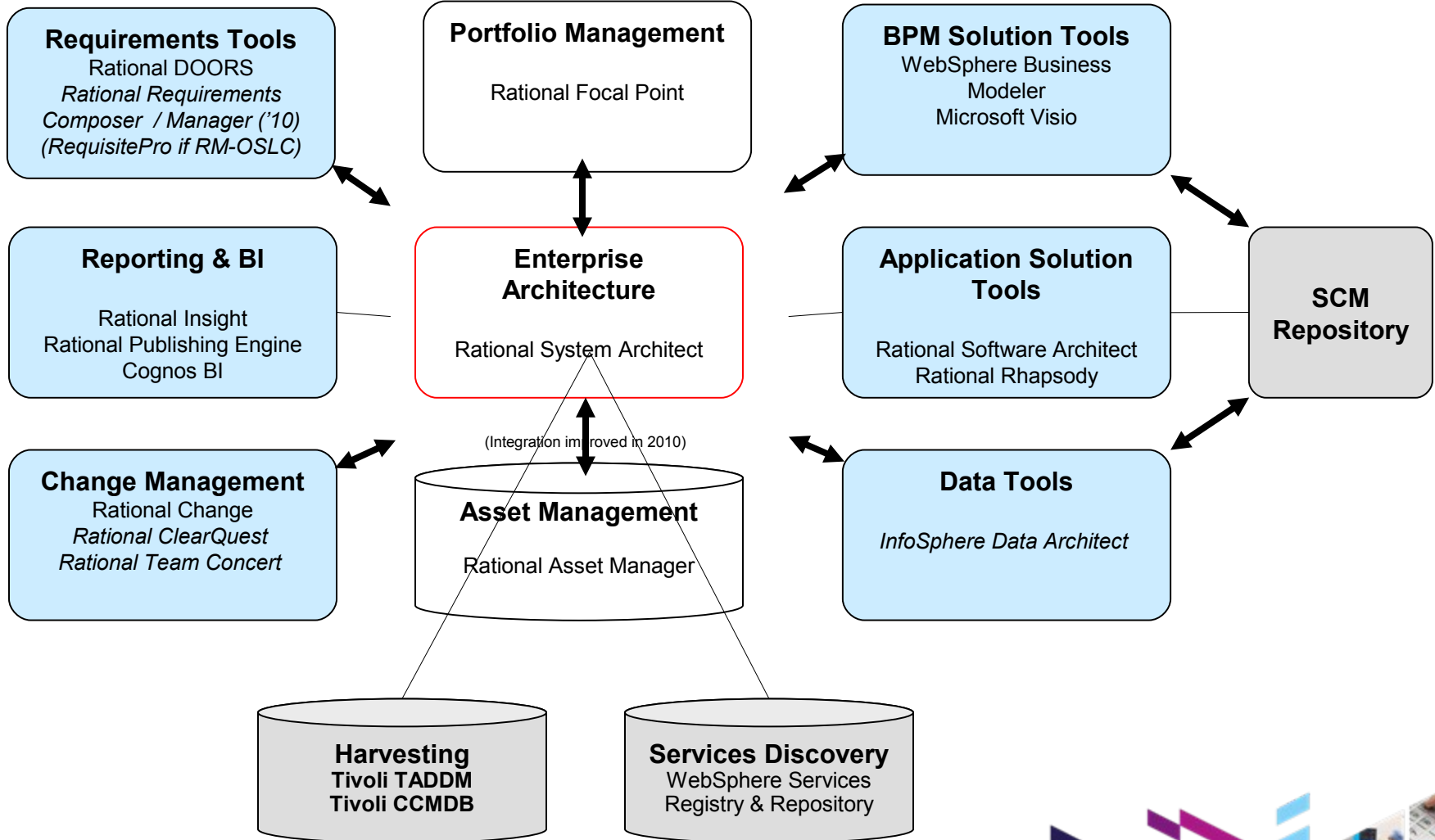
- Reduce time, effort, and cost to analyze and plan the project portfolio and technology investments
- Enables the ability to deliver a simplified and agile “Foundation for Execution”
- Improve efficiency of execution in alignment with Strategic goals and objectives (i.e. approve the “right” projects/programs)



Strategic planning & EA Management: today and tomorrow

Track: Enterprise Architecture come strumento di Business Planning, Alignment & Investment

Speaker: Marco Torrisi, Francesco Meloni



Software Delivery Challenges

Technologies

- Platforms (mobile, server, rich client...)
- Languages (existing & emerging...)
- Programming Model (existing & emerging...)
- Skill/background/target (Web, legacy, embedded...)

Productivity

- Planning Challenges (Business priorities change and project execution ...)
- Team Challenges (Diverse tools)
- Platform Process Challenges (Process is on paper - tools don't enact it)
- Platform Integration Challenges

Collaboration

- Scenario (in-house, outsourced, auction contracted..)
- Teams (One site to multi-site..)
- Roles (Developer, Tester, Architect, Deployment/Production, customer/client...)
- Methodology (iterative, agile, MDD...)
- Crosscutting Concerns (Fidelity to requirements, process...)

Rational Solution



A common platform provides an advantage!



Learn more at:

- IBM Rational software
- Rational launch announcements
- Rational Software Delivery Platform
- Accelerate change & delivery
- Deliver enduring quality
- Enable enterprise modernization
- Ensure Web security & compliance
- Improve project success
- Manage architecture
- Manage evolving requirements
- Small & mid-sized business
- Targeted solutions
- Rational trial downloads
- developerWorks Rational
- Leading Innovation
- IBM Rational TV
- IBM Business Partners
- IBM Rational Case Studies

© Copyright IBM Corporation 2010. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials 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 the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

Software Delivery Challenges—Programming technologies

Platforms

- Mobile
- Server
- Rich Client
- Embedded/specialized system

Languages

- Existing (Java, C/C++, COBOL, EGL, PHP, Ruby, Perl, Python, C#, etc)
- Emerging (X10, Thorn, LIME, UAL, UPC, OpenCL, etc)

Programming Models

- Existing (J2EE, CICS, IMS, SQL, .NET, Web 2.0, 4GLs, etc)
- Emerging (executable UAL, Map-Reduce/Hadoop, Streams, etc)

Skills/background/target

- Web
- Business or domain...”not a programmer”
- Legacy systems
- Systems/Embedded



Constant innovation driving substantial productivity gains

Software Delivery Challenges—Collaboration challenges

Teams

- All sizes from 1 to thousands
- One site to multi-site
- In-house, outsourced, auction contracted
- Project managed in-house or out

Roles

- Developer, Tester, Architect, spec author
- Project manager, CIO, LOB
- IT, Deployment/Production, customer/client

Development Methodology or Scenario

- Situational/agile
- Formal, MDD
- SW/HW co-design
- Incremental/maintenance
- Migration/modernization

Crosscutting Concerns

- Interlock, reuse, refinement/change
- Security, privacy, compliance/governance
- Scalability, high availability, reliability
- Fidelity to requirements, process
- Time to value, responsiveness to change



Enablement via rich services exposed as a common platform

Software Delivery Challenges—Productivity challenges

Planning Challenges

- Business priorities change and we need to prioritize our plans in an agile way?
- Large number of projects that release much more frequently.

“What if plans were visible to all and linked directly to development execution?”

Team Challenges

- Diverse tool requirements for the team requiring cross role collaboration

“What if teams could collaborate around artifacts using URI links and bring new team members on board anywhere in seconds?”

Platform Process Challenges

- Process is on paper - tools don't enact it
- Need for agility **at scale**

“What if all your tools followed a common role based permission and process model?”

Platform Integration Challenges

- Lack of standards impacts ability to collaborate, automate and report across teams and assumptions

“What if you could link tools, using standards and those integrations didn't break from release to release?”



A common platform provides an agile advantage!