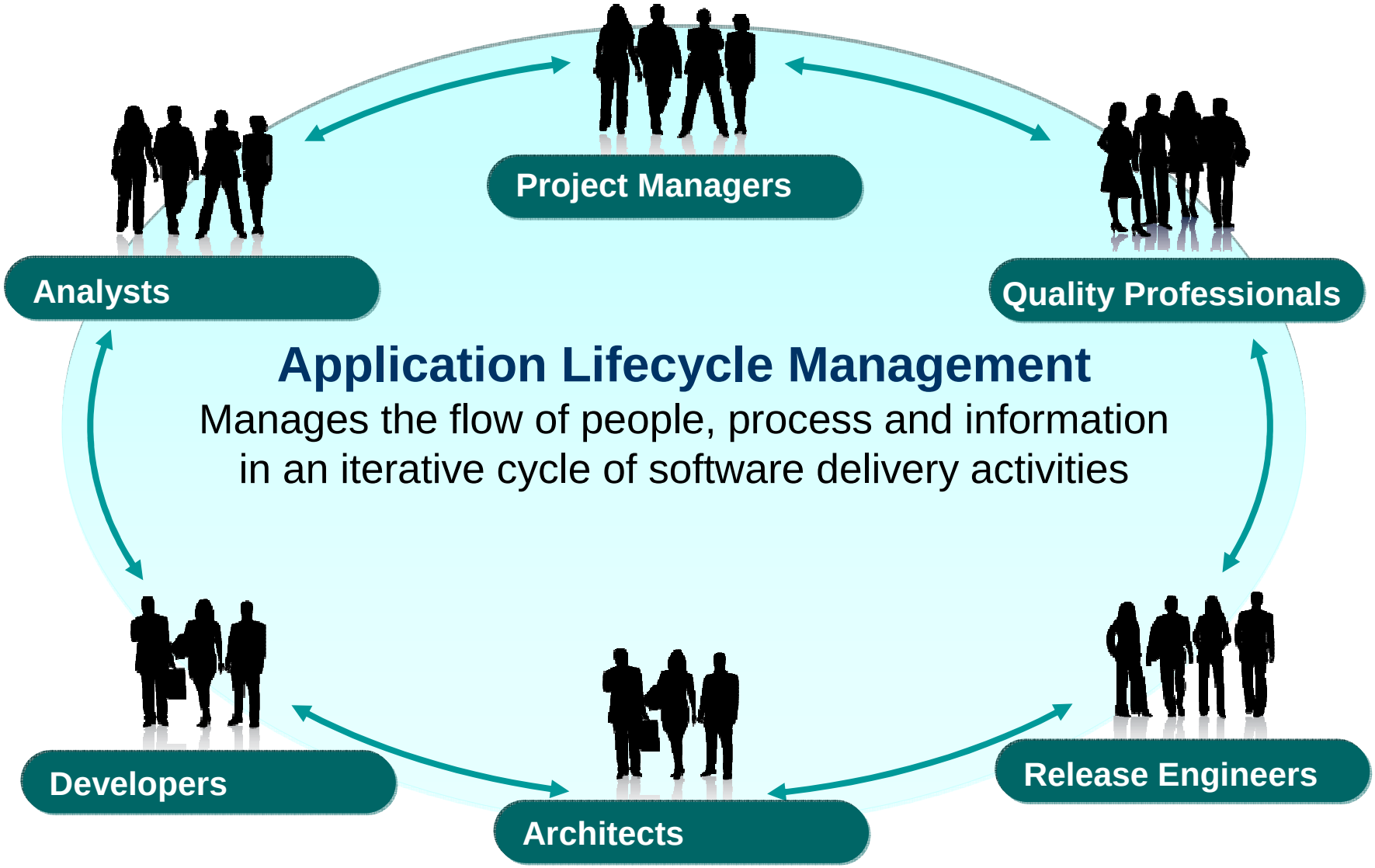
Five Imperatives for effective Application Lifecycle Management

Improve productivity with Application Lifecycle Management

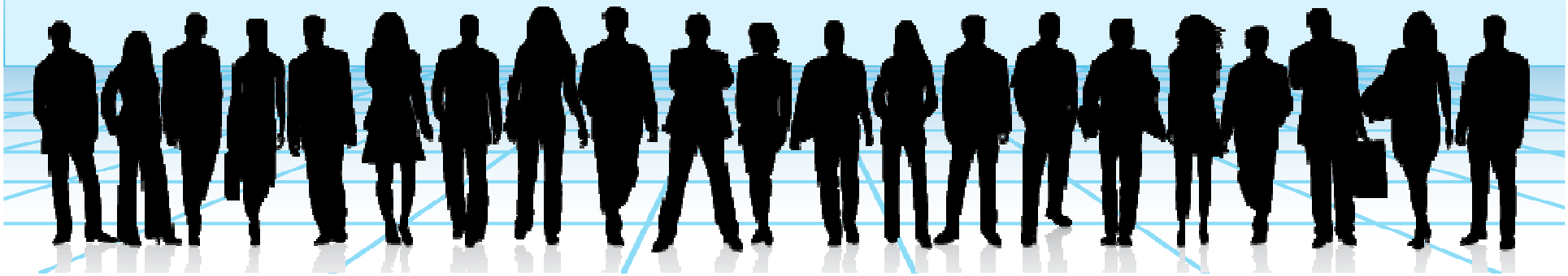


What do teams do to improve their productivity?

Five Imperatives for effective Application Lifecycle Management

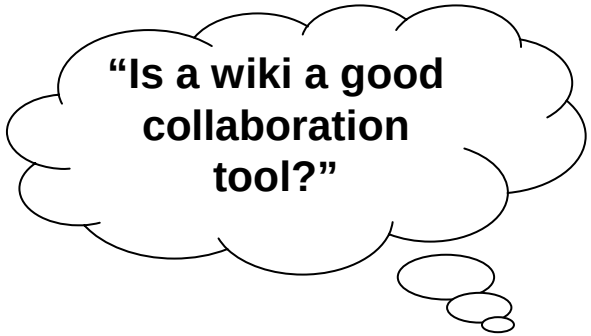
To improve organizational & team productivity

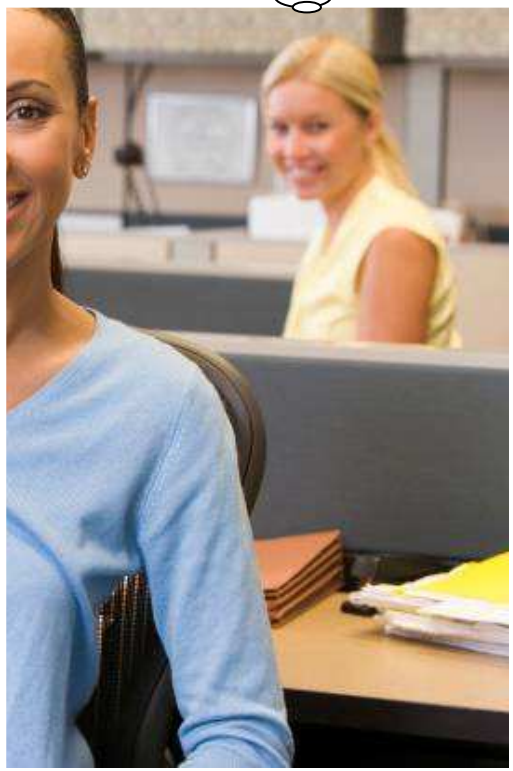
1. Maximize **product value** with **In-context Collaboration**
2. Accelerate **time to delivery** with **Real-time Planning**
3. Improve **quality** with **Lifecycle Traceability**
4. Refine **predictability** with **Development Intelligence**
5. Reduce **costs** with **Continuous Improvement**



Collaboration challenges

Can your team…

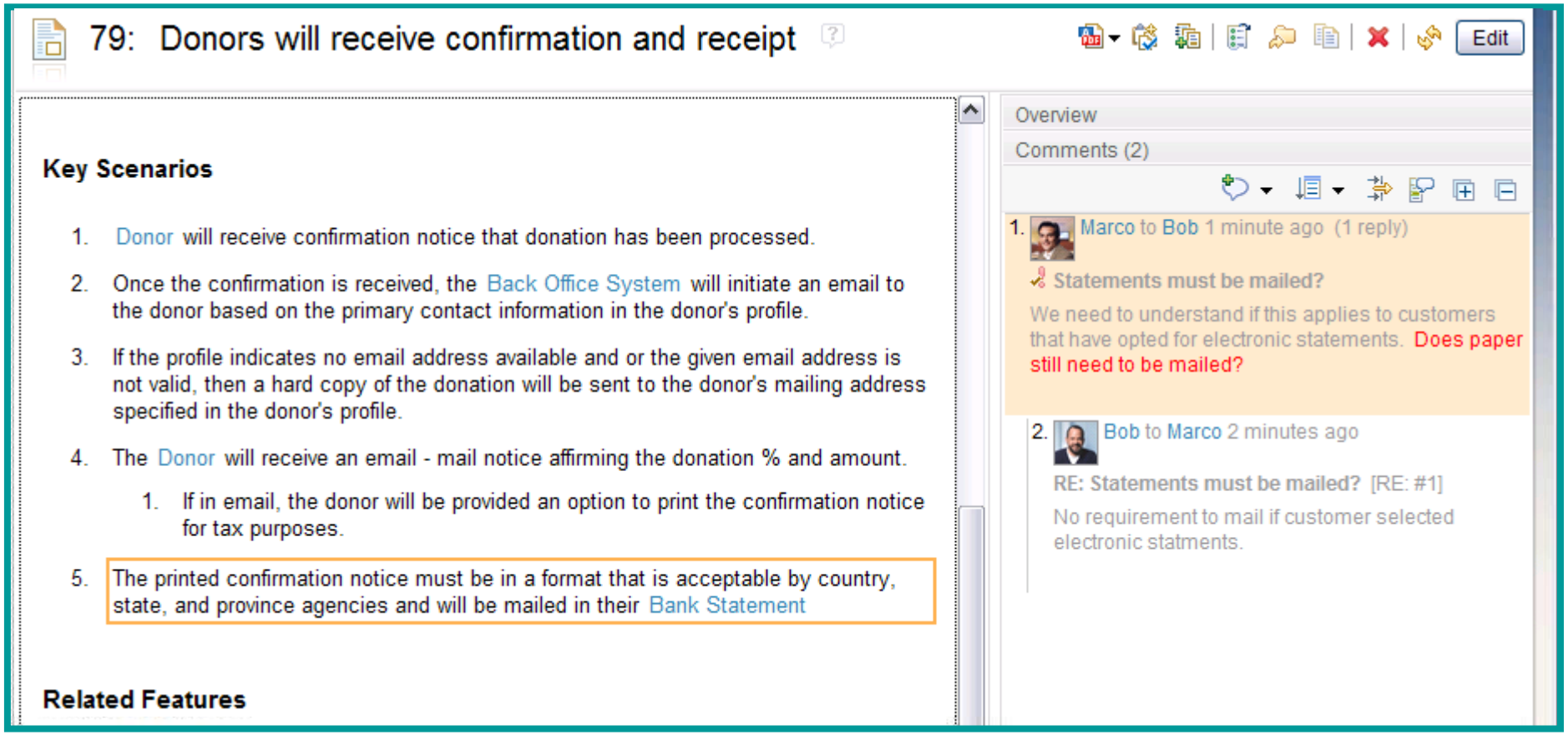




- Overcome the barriers of multiple languages and time zones when working with outsourced and distributed team members?
- Bring new team members up-to-speed quickly leveraging historical information in a single source of truth?
- Quickly grasp the “who, what, when and why” of team activities?
- Do you still view Wiki’s and email as advanced collaboration tools?

In-context Collaboration

Comments on are in-context of the artifact

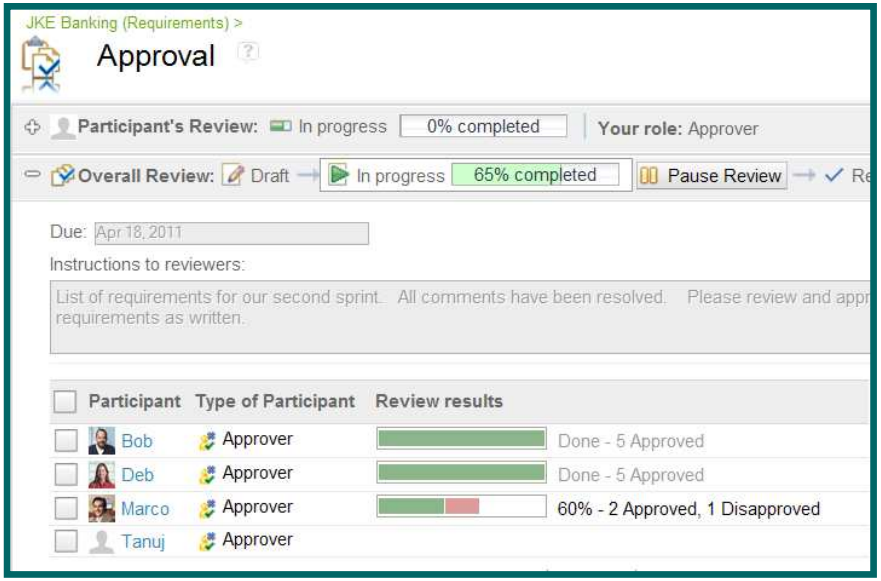


A click on the comment highlights the text under discussion.

Formal Reviews drive agreement and prevent re-work

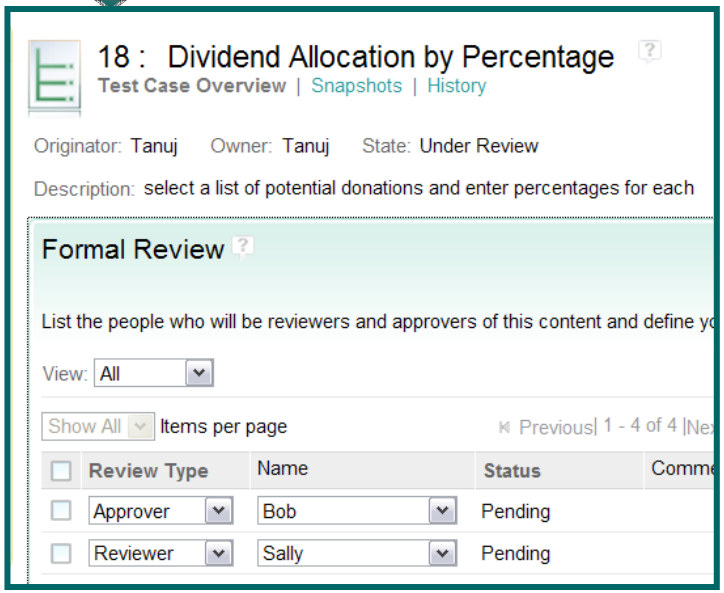
- ✓ Reviews & approvals insures artifacts are reviewed and/or approved by key team members and captures compliance requirements.

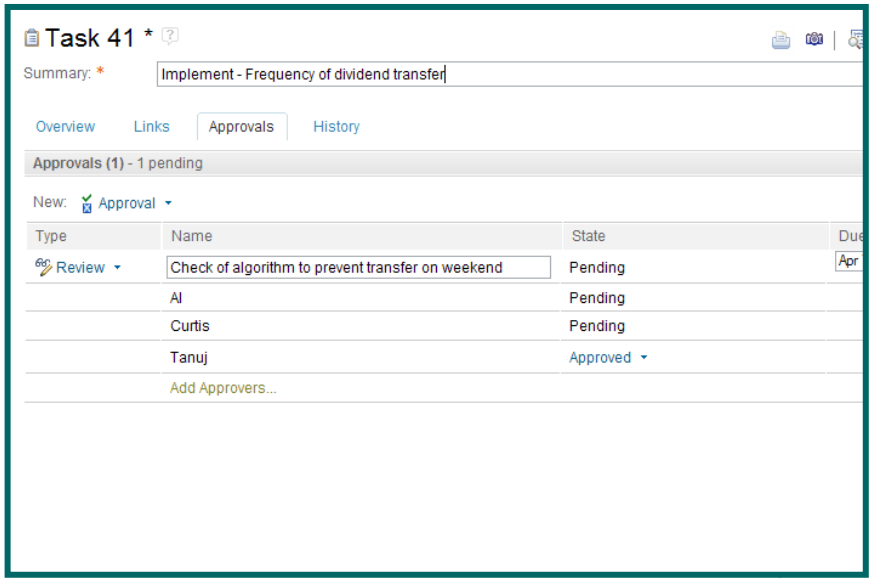
Requirements



Work Items

Test Artifacts



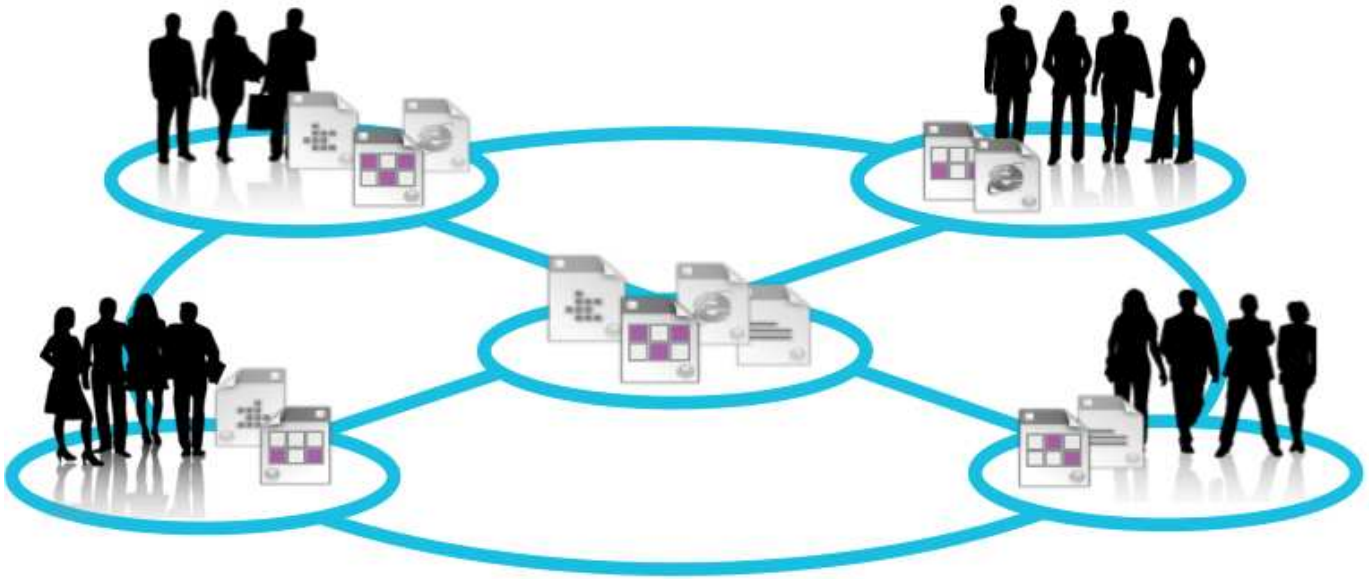


In-Context Collaboration improves product value by:

– Making **information <u>immediately</u> accessible** to all team members in the context of their work

– Empowering teams to collaborate on and review software development artifacts so they can **incorporate feedback early and often**

– Providing **single source of truth** hosted in a shared repository so that team members can collaborate effectively around the globe



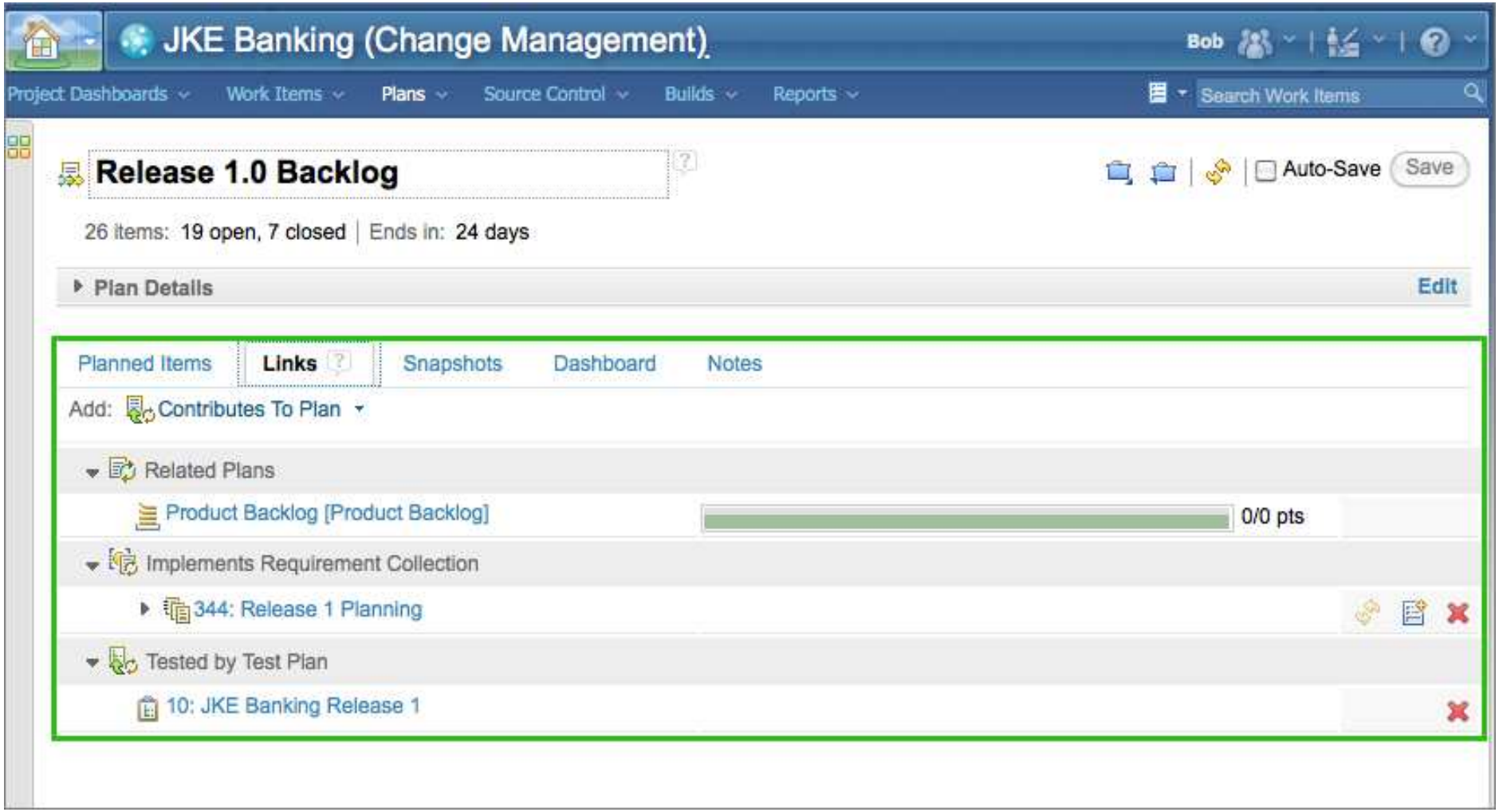
Planning challenges

Does your team…

- View “Plan” as a noun or a verb?
- Plan for improved security and vulnerability remediation?
- Plan across the entire team, involving all stakeholders/disciplines?
- Plan for waterfall, iterative and agile environments?
- Manually collect/report status?

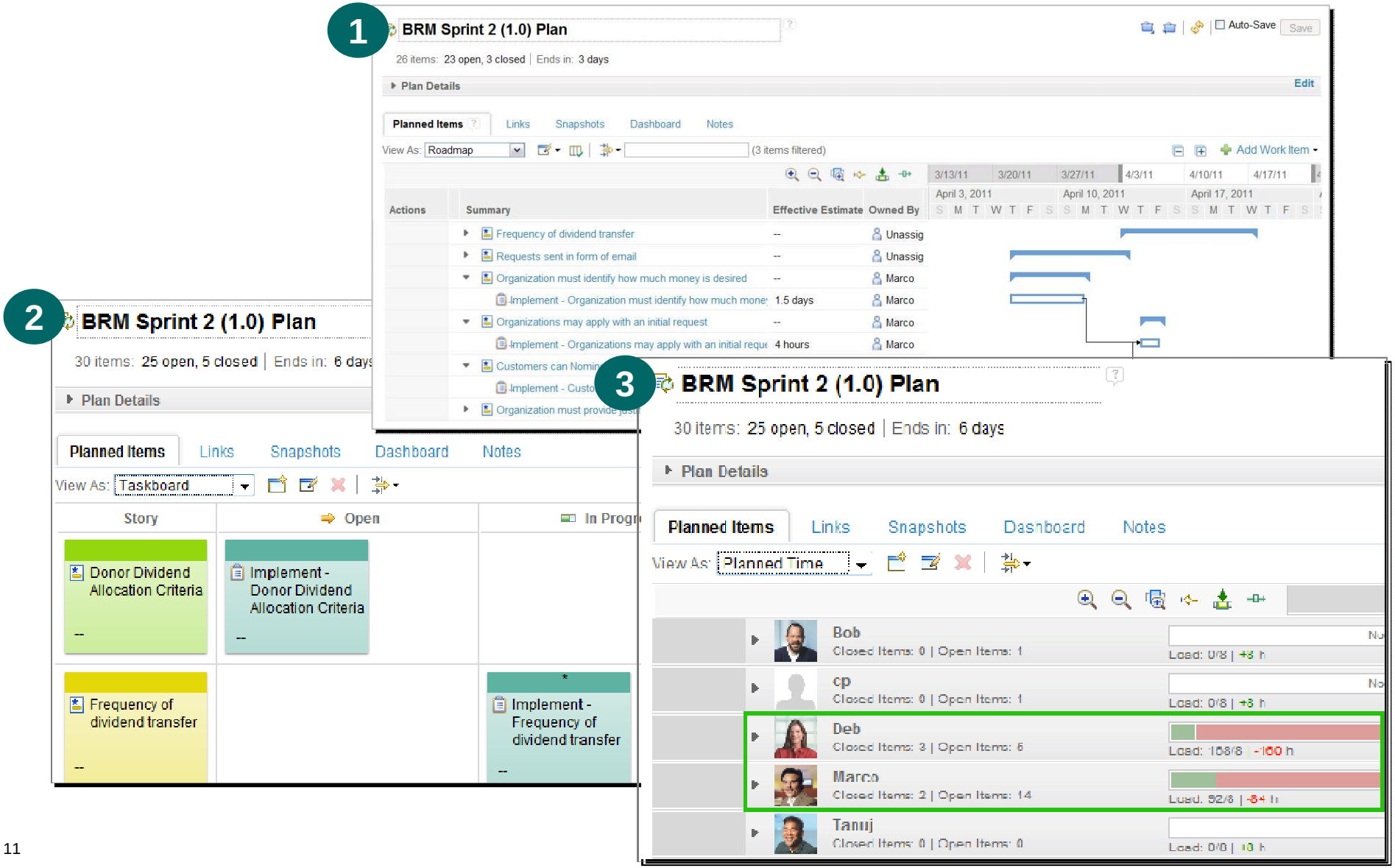
Real-time Planning

Ensure alignment across the business, development and test teams



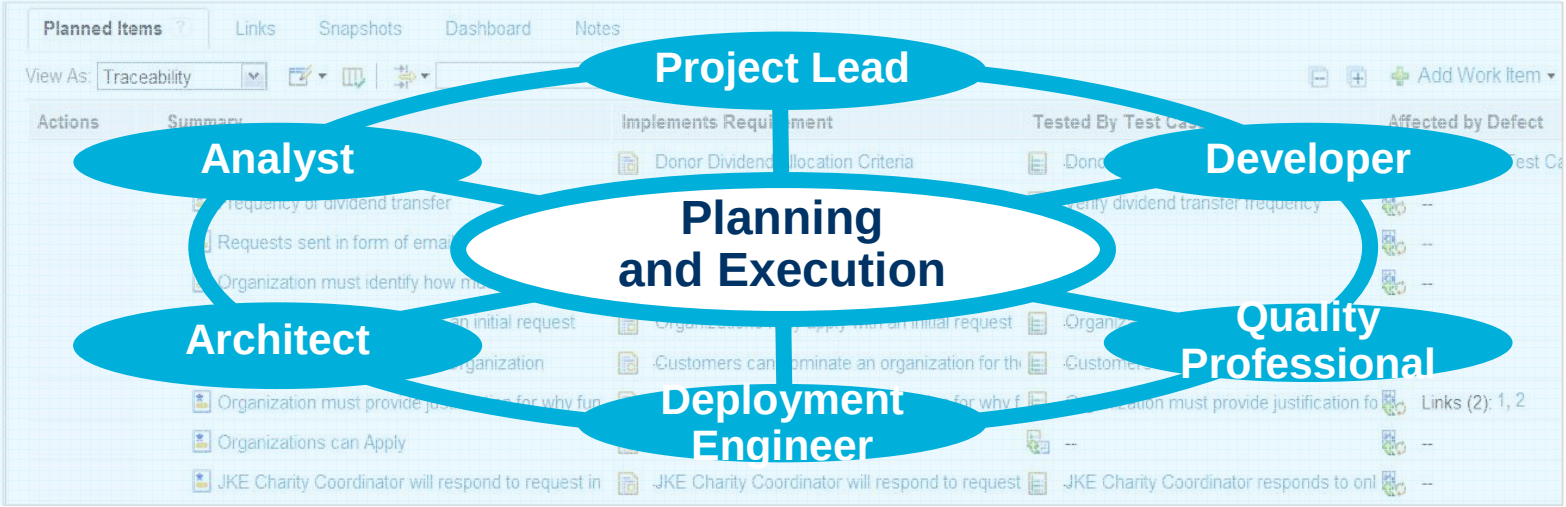
Real-time Planning

One plan with multiple views facilitate detailed analysis



Real-time Planning accelerates time to delivery by:

- Providing a **single plan** that spans requirements, development, and test, ensuring a team understanding of the overall scope of a project
- **Integrating planning with execution** ensuring the entire team understands the true project status
- Allowing **everyone to participate** in keeping the plan current and accurate
- Helping teams **respond to the unexpected** in a timely manner ensuring the team stays on schedule



Lifecycle Traceability challenges

Can your team answer…

Analyst

Which requirements are addressed in this iteration?

Are all of the requirements tested?

What defects are affecting which requirements?

Project Manager

Can we pass an audit?

Are we ready to release?

What defects were resolved in this release?

Are we aligned with the business?

Architect

Can we implement in a way that supports maintainability?

Which requirements impact the design?

Are there reusable components we can leverage?

Developer

What requirements am I implementing?

What test uncovered this defect, on which environment and what build?

What changes occurred overnight?

Release Engineer

How can I standardize when teams use different tools?

Where are the bottlenecks in our processes?

Are build times getting longer or shorter?

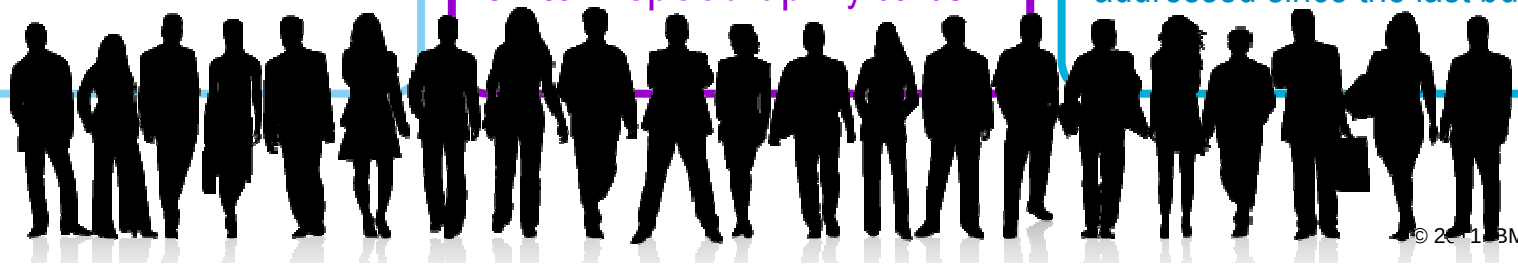
How can I speed up my builds?

Quality Professional

What is the quality of the build?

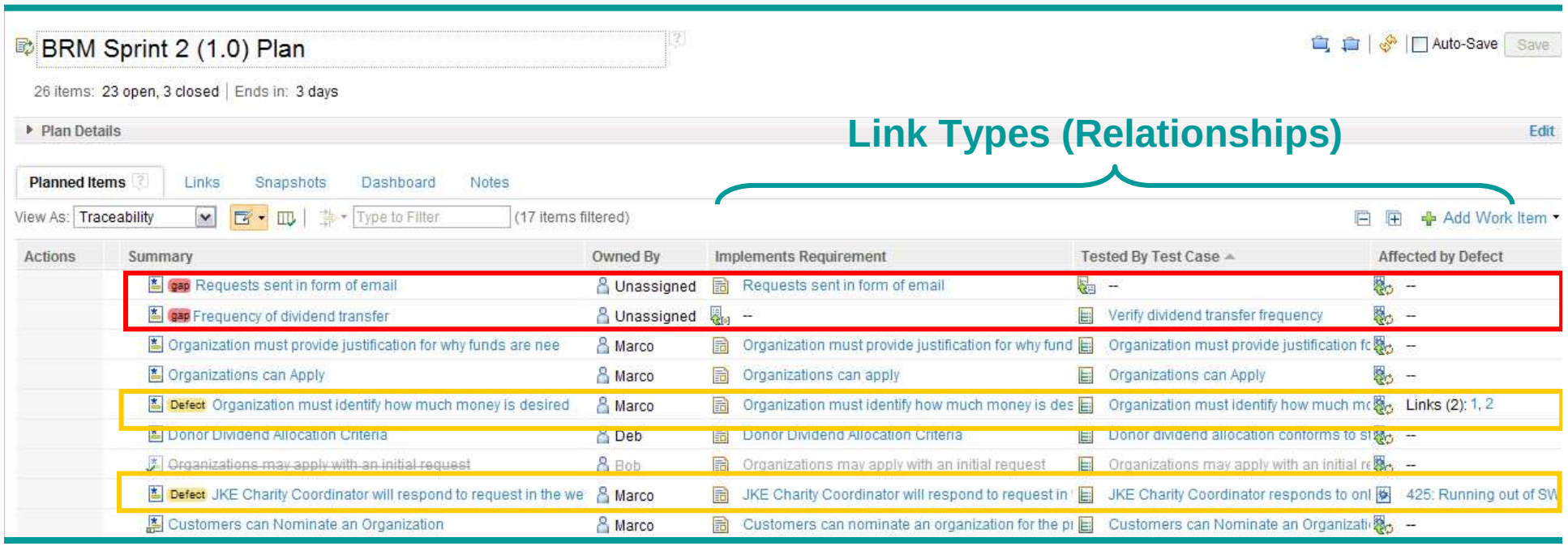
What is ready for me to test?

What defects have been addressed since the last build?



Team Leads have visibility into coverage & completeness

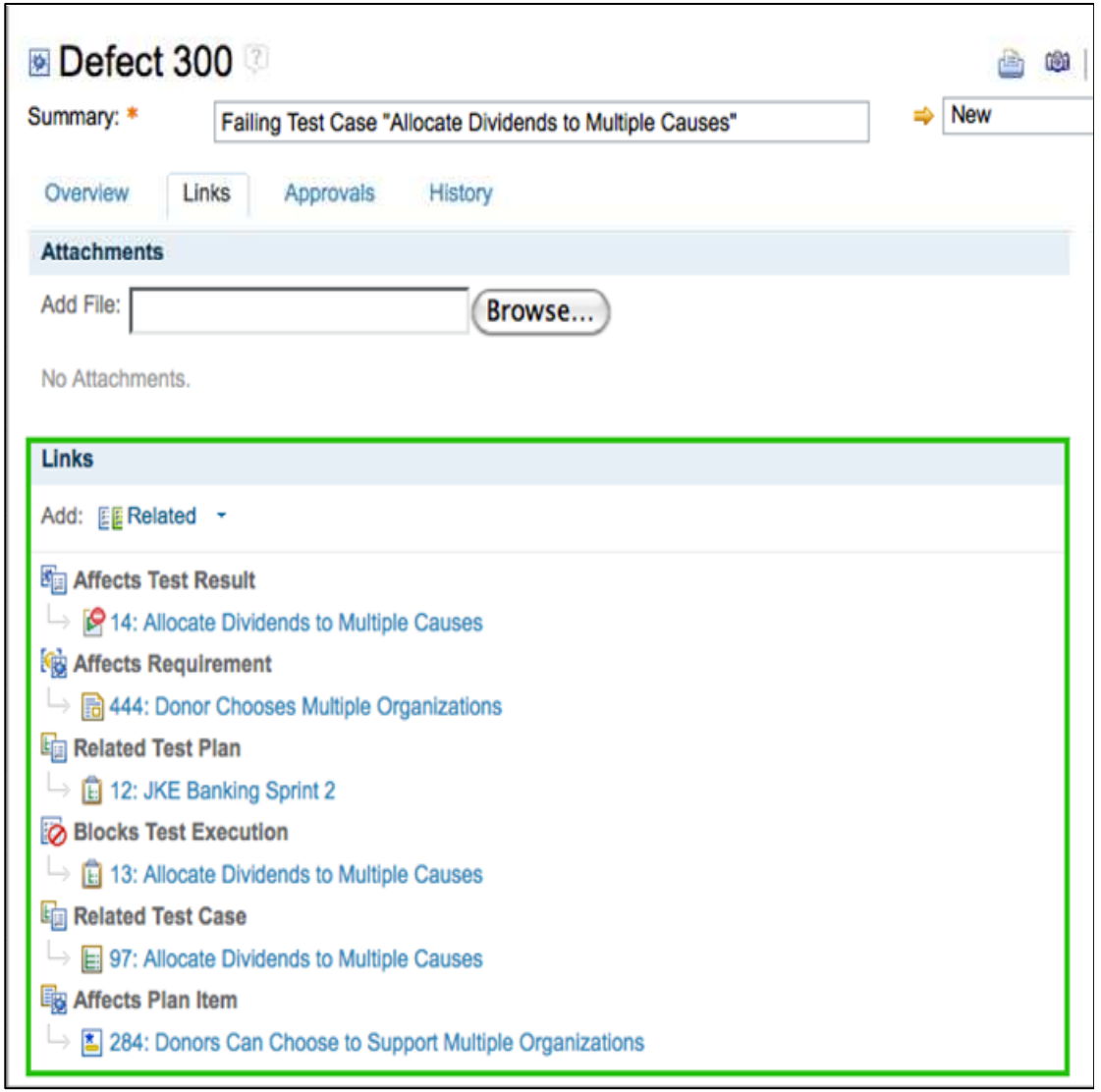
Tailor traceability View to highlight planning Gaps



Linked data is visible on all artifacts

- All roles can view the links on their artifacts.
- Each user has their own perspective and user interface for viewing links.
- Create, delete, navigate links on artifacts

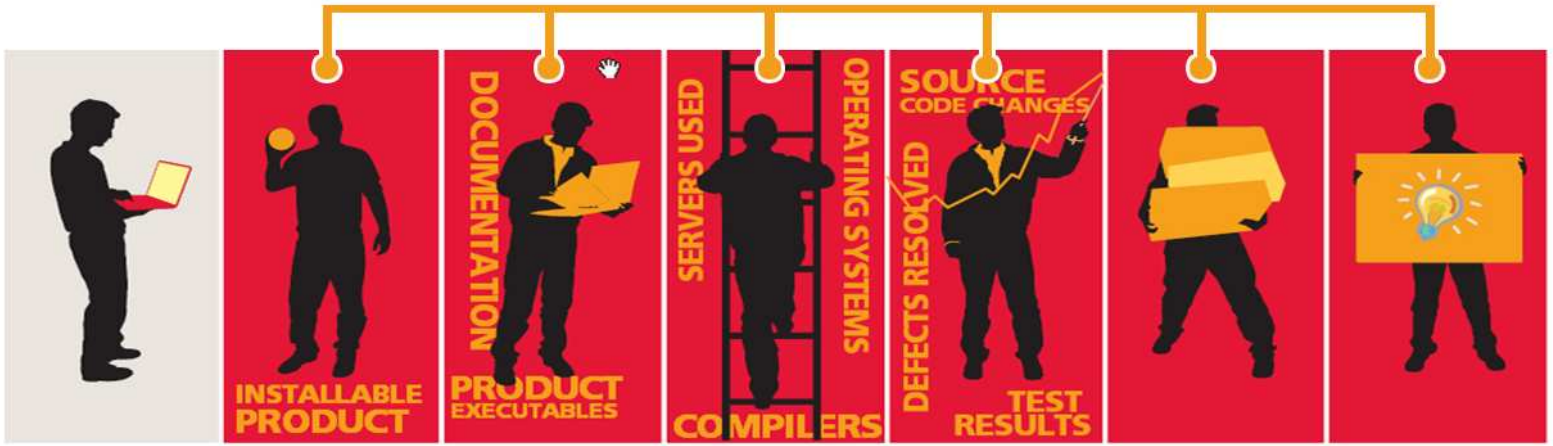
- Example: Use the relationships on a defect to determine it's impact
- Traceability links on defects are automatically created



Lifecycle Traceability improves quality by:

- Establishing **relationships between software artifacts**
- Helping you **identify and closing artifact gaps,** ensuring **coverage** across disciplines
- Provides visibility into the **completeness** of planned items by inspecting all related artifacts
- Provides easy **access to related artifacts** ensuring everyone shares the same view
- Delivers transparency which enables everyone to make **fully informed decisions** based business priorities

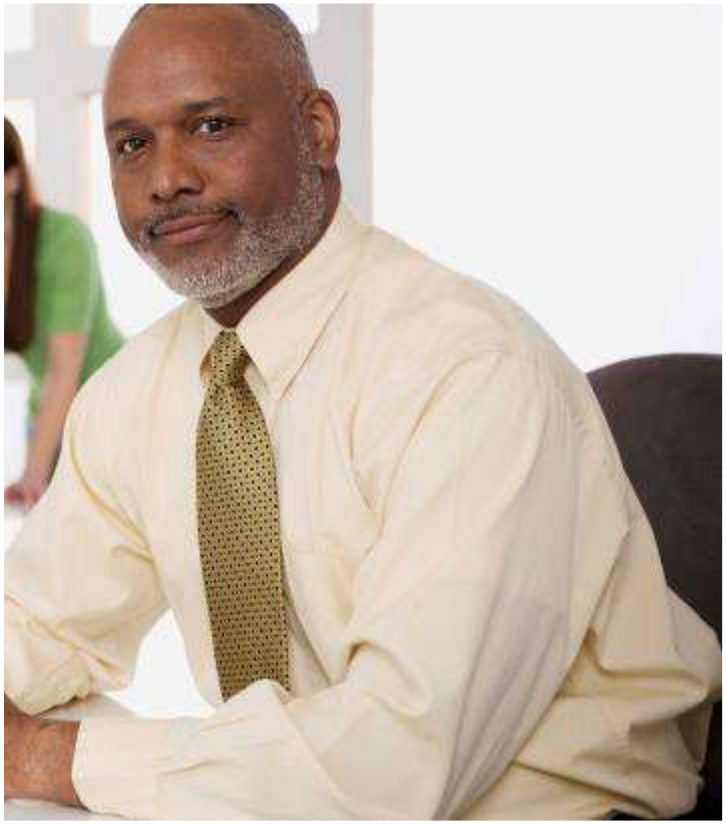
Instant access to details from any point in the development process



Measuring and steering challenges

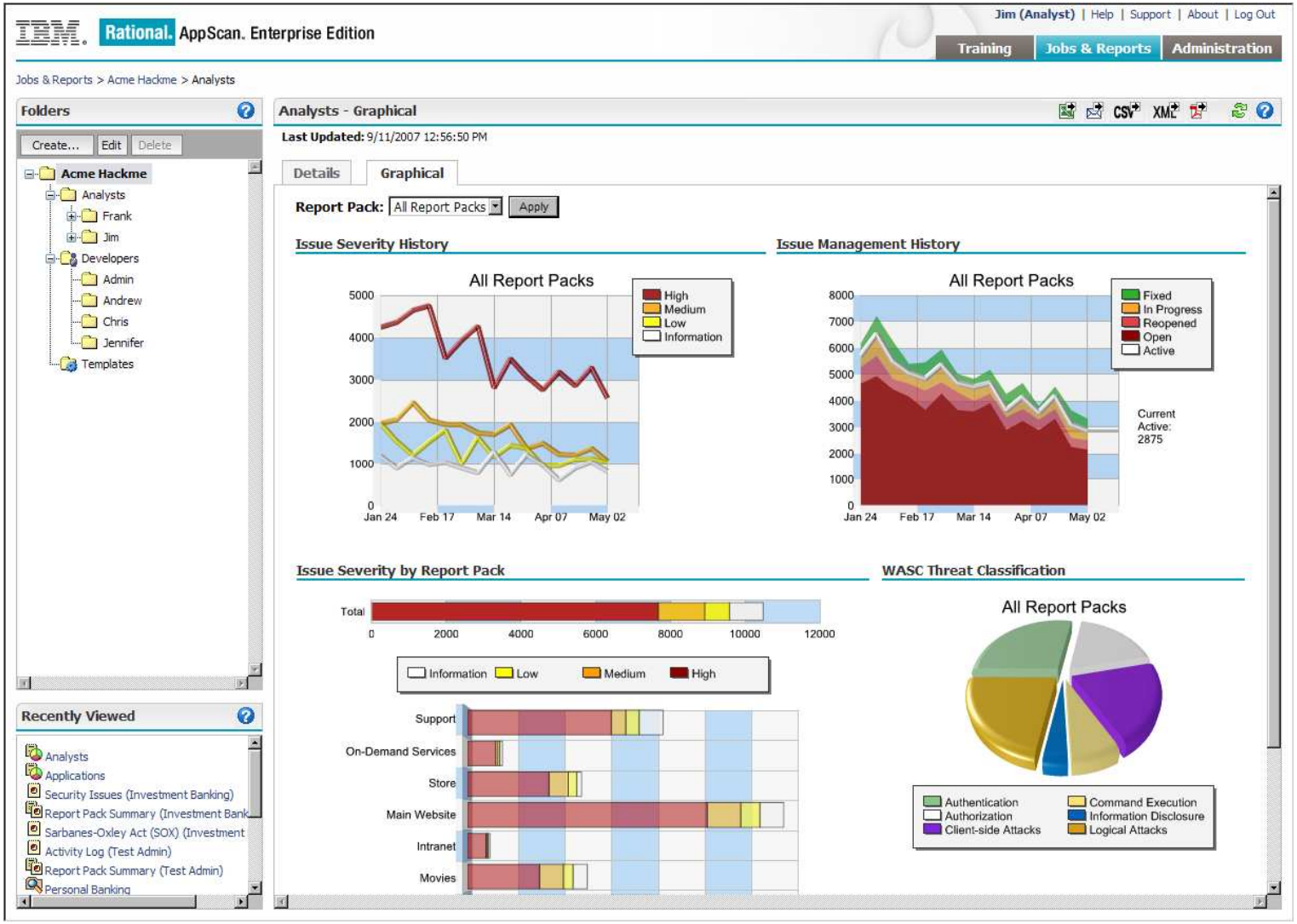
Can your team…

- Track desired improvements with metrics?
- Minimize administrative overhead by relying on your environment to capture metrics?
- Provide one view of project health at multiple levels?
- Make steering decisions based on facts rather guestimates?
- We call this Development Intelligence

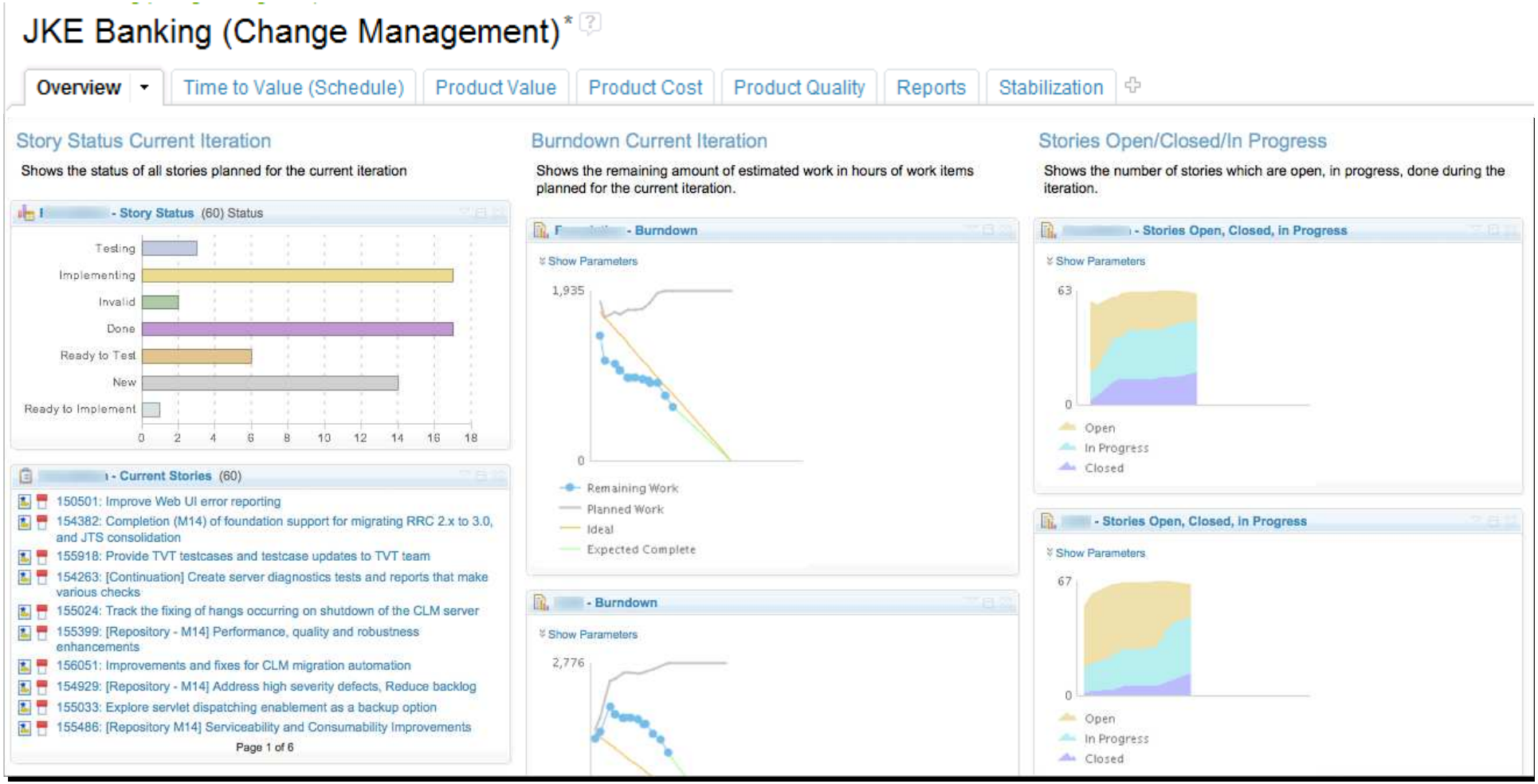


Development Intelligence

Track progress on vulnerability remediation progress

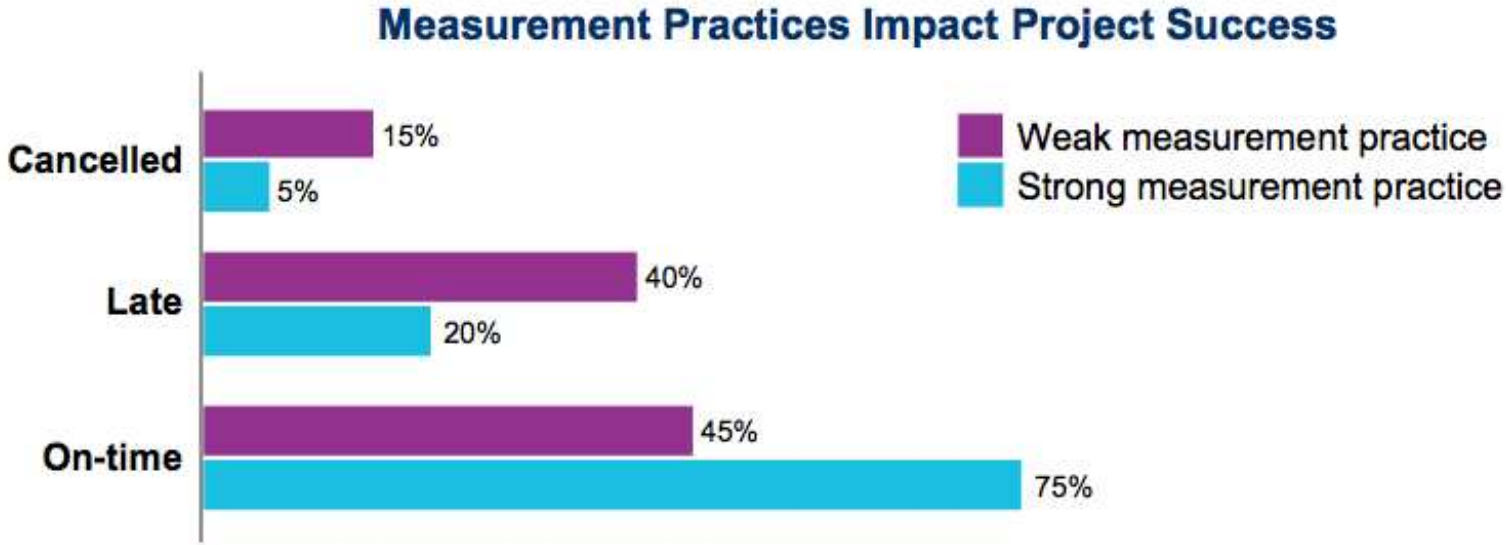


Information radiators provide a view of project health for all team members



Development Intelligence improves predictability by:

- Applying **Business Intelligence techniques** to software and systems development
- Enabling **fact-based decision making** (to communicate status, monitor progress, diagnose problems, identify corrective actions)
- **Steering projects and programs** to deliver on-time



Source: Capers Jones, Measurement, Metrics and Industry Leadership, 2009 and Software Engineering Best Practices, McGraw Hill, 2010.

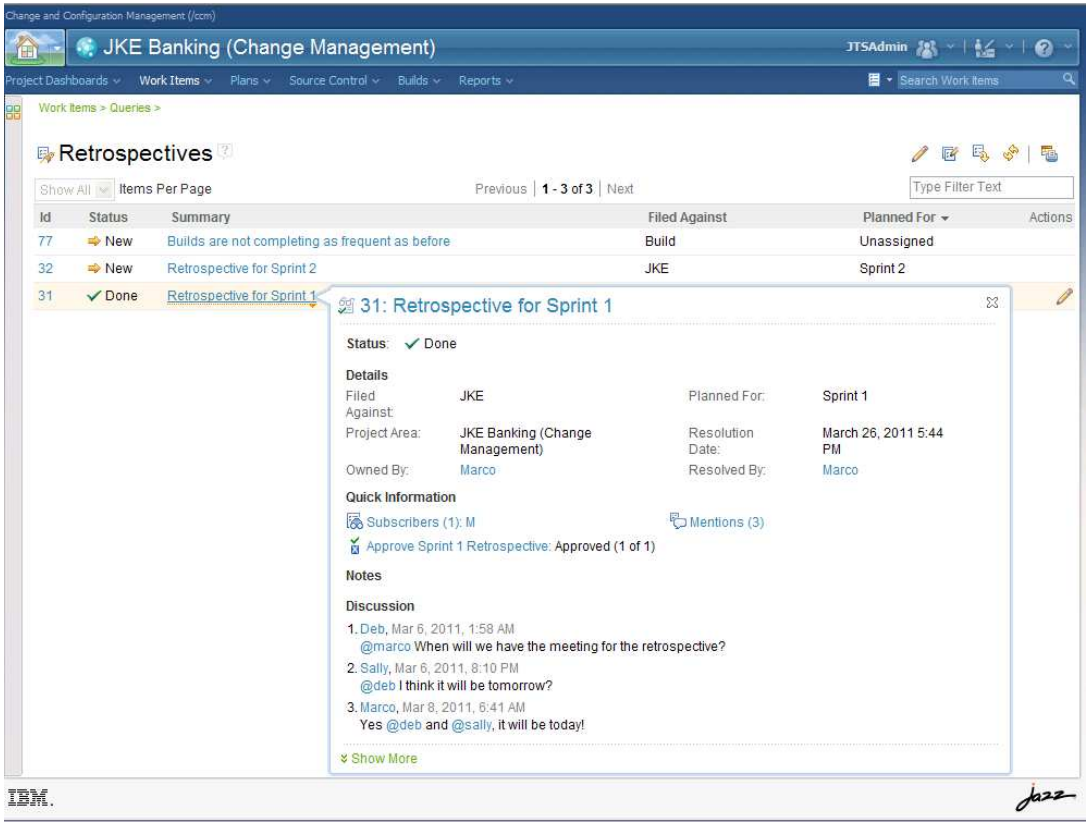
Continuous improvement challenges

Can your team…

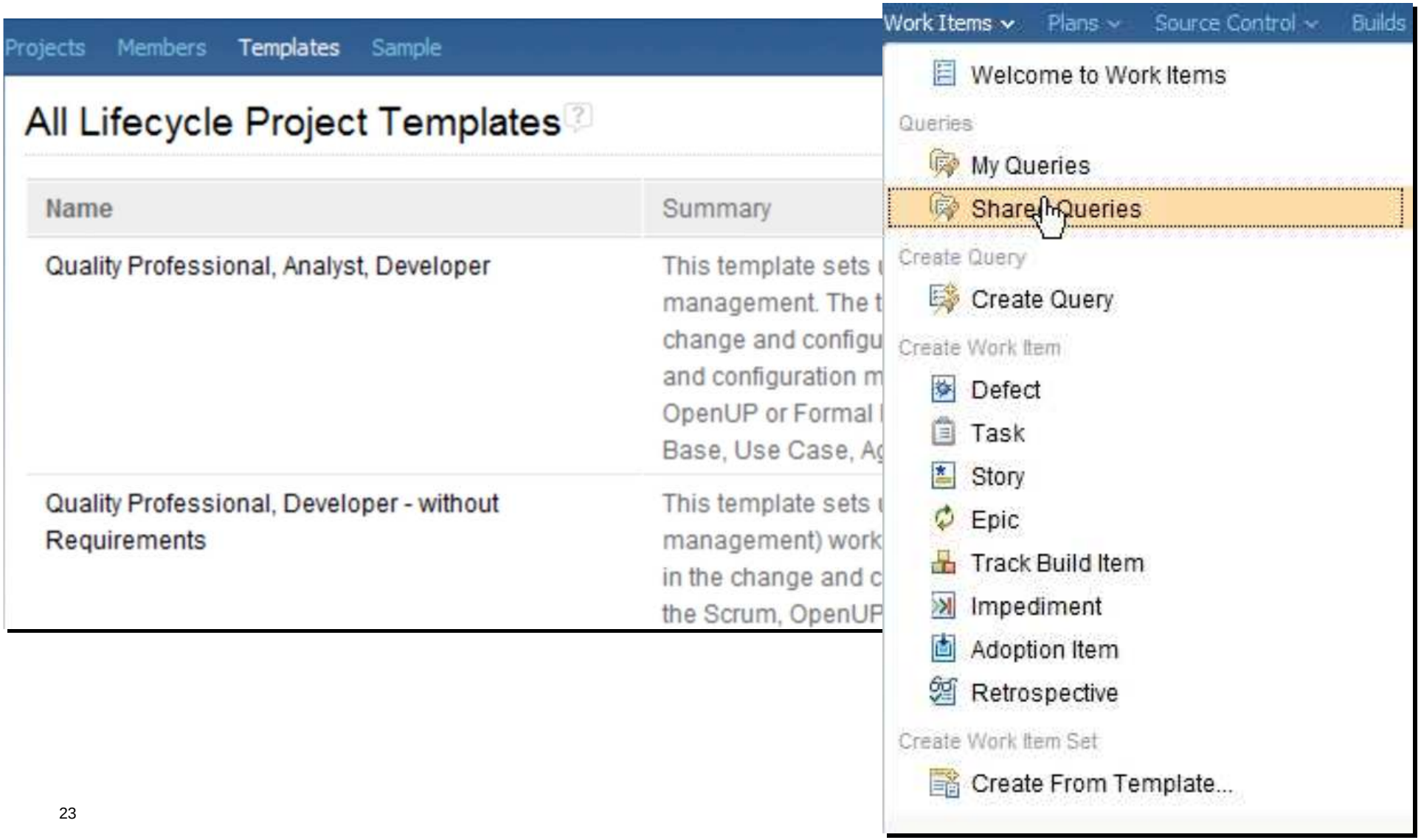
- Leverage out-of-the-box templates that support traditional and agile workflows?
- Change process “on the fly” as part of a continuous feedback loop?
- Allow team leads to determine how strict or lax the “rules of the road” should be?
- Modify process enforcement over the life of a project, to encourage early-stage experimentation and end-game stability?

Team Retrospectives help identify areas of improvement

- Tracking Retrospectives help team members to collaborate on improvements.
  - Ranking of process improvements help to focus on where the team feels the biggest pain.
  - Metrics where available help support process improvements and show results.



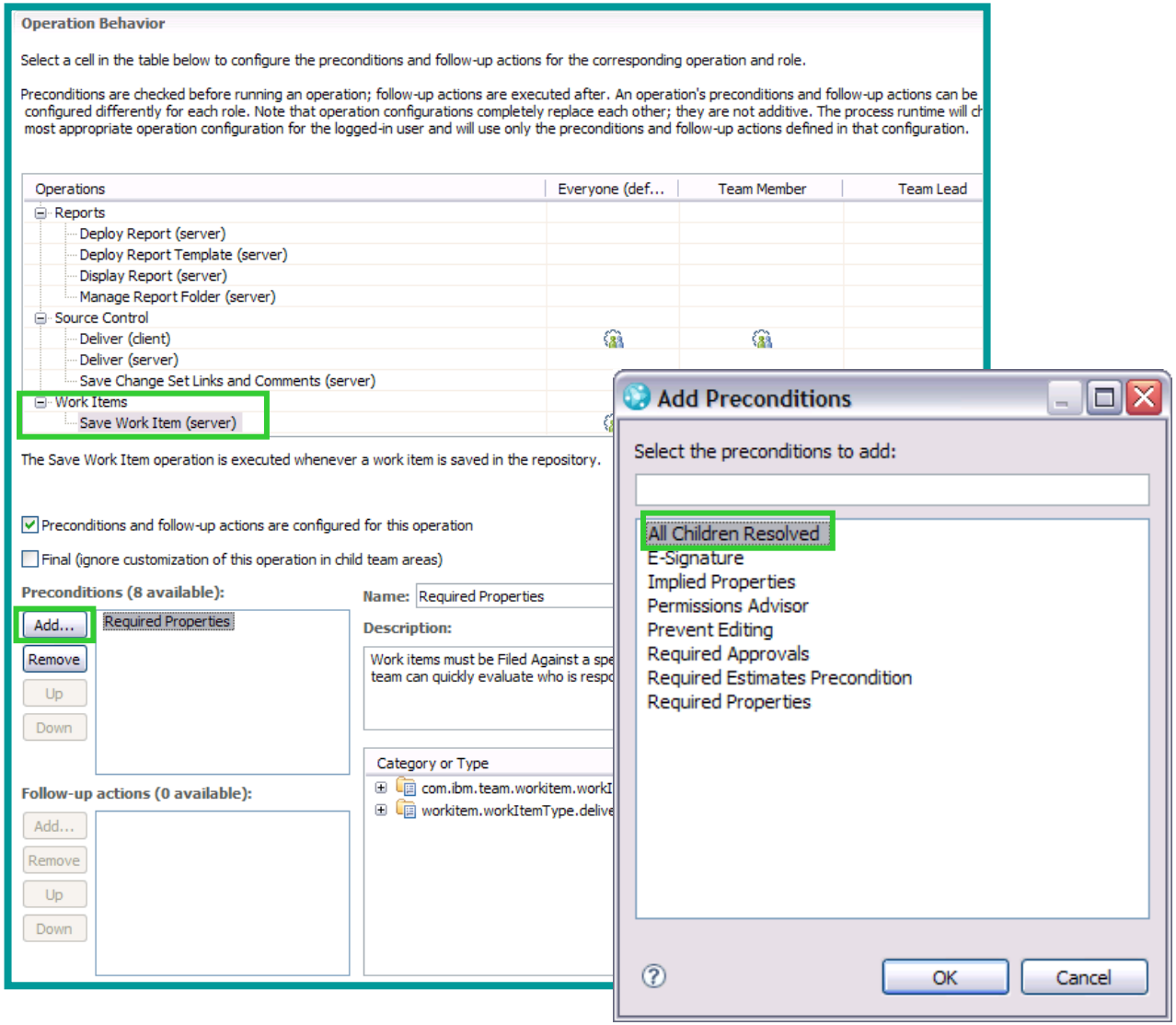
Instantiate best practices as templates for breakthrough improvement



Tweak process definition “in-flight” to correct team actions

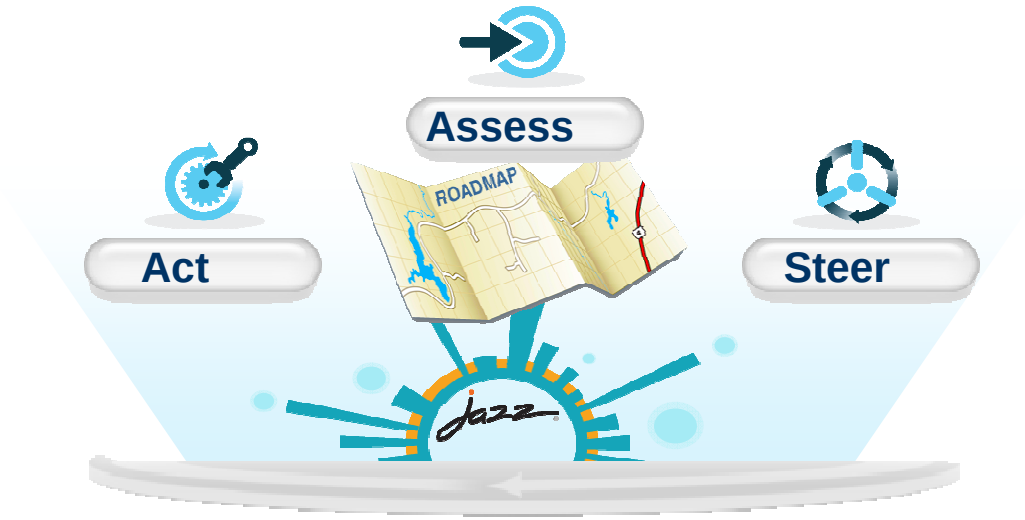
Problem: Story completed without all work completed

Solution: Require detailed tasks completed before completing parent



Continuous Improvement reduces cost by:

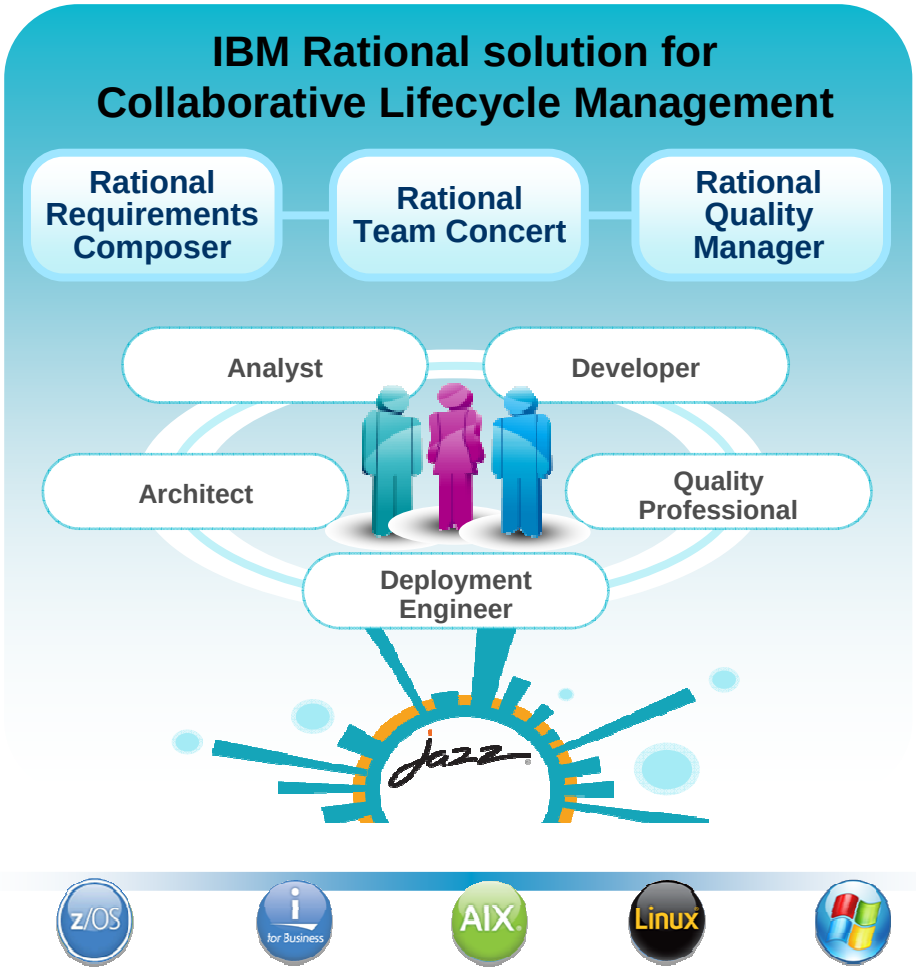
- Enabling breakthrough improvement through the ongoing **adoption of best practices and automation** to reduce manual, non-creative and error prone tasks
- Promoting **incremental improvement of a project** when needed through user interfaces that are easy to customize and 'in-flight' changes to process
- Allowing **everyone to participate** with easy to adopt best practices at your fingertips.



IBM Rational solution for Collaborative Lifecycle Management

Improve productivity with an integrated ALM solution

- **Optimize your team through support of the 5 ALM Imperatives**
  - Real-time planning
  - Lifecycle traceability
  - In-context collaboration
  - Development Intelligence
  - Continuous Improvement
- **Get up and running quickly**
- **Extend as your needs evolve with role-based licensing**
- **Support heterogeneous development across multiple platforms and technologies**



www.ibm.com/software/rational