Development productivity in an agile world – past, present, and future

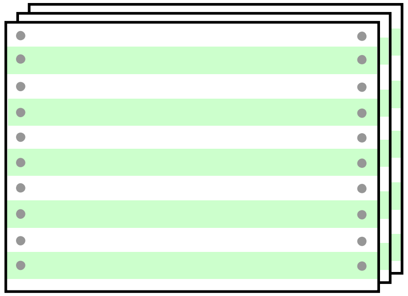
Maria Ericsson

Executive IT Specialist

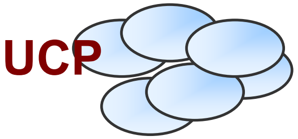
IBM Rational Tiger Team

GOTO Prague 2011

Historical Perspective on Productivity



- SLOC was easy to measure … still is:
  - There is a large pool of SLOC based productivity statistics
  - These statistics are still the most complete set of data existing
- Function Points & Use Case Points were developed to solve shortcomings, however:
  - Different programmers code differently
  - Not all code adds value
  - Functions or Use Cases can have different “sizes”



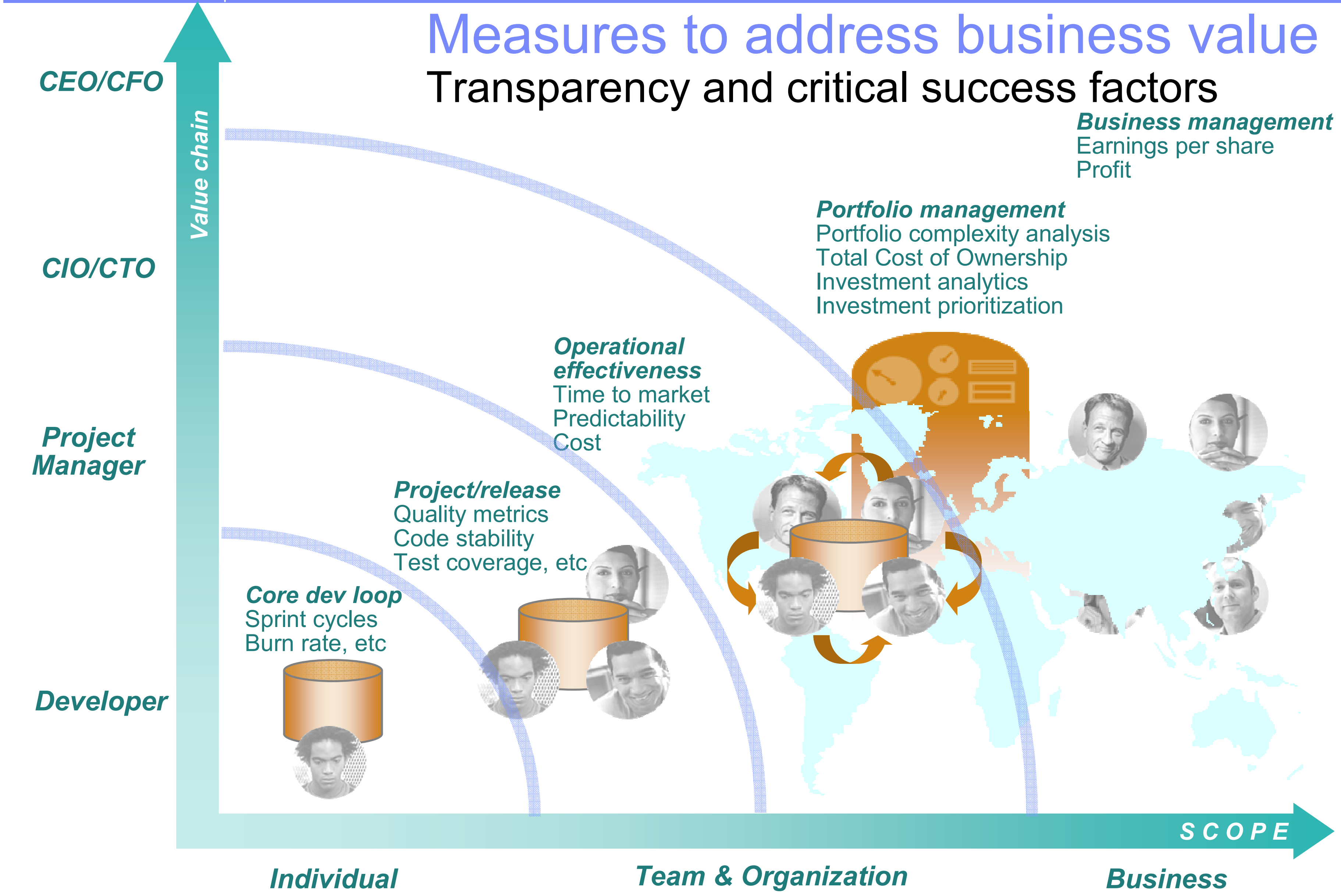
- Experience indicates that utility of these measures is limited:
  - Intra-team, it can be used (*carefully*!) to calibrate productivity within the team
  - Across organizations & platforms, comparisons break down

Modern Perspective on Productivity

- Many development organizations have sophisticated dashboards
  - Richer measurement set
  - Much less human intervention
- However, productivity is not only about asset volume – –
  Productivity is about *business value produced*
  - Challenge: determining a consistent meaning for business value that is useful to the entire organization
  - Challenge: making the measures meaningful, transparent, real-time € € € €
- Conclusion: a productivity strategy should include
  - Transparency at all levels in an organization
  - Consistent and useful leverage of a carefully crafted chain of related measures
  - Using modern development technologies that include measurement facilities to foster continuing relevance, accuracy, and real-time results

Measures to address business value

Transparency and critical success factors



Business management
Earnings per share
Profit

Portfolio management
Portfolio complexity analysis
Total Cost of Ownership
Investment analytics
Investment prioritization

Operational effectiveness
Time to market
Predictability
Cost

Project/release
Quality metrics
Code stability
Test coverage, etc

Core dev loop
Sprint cycles
Burn rate, etc

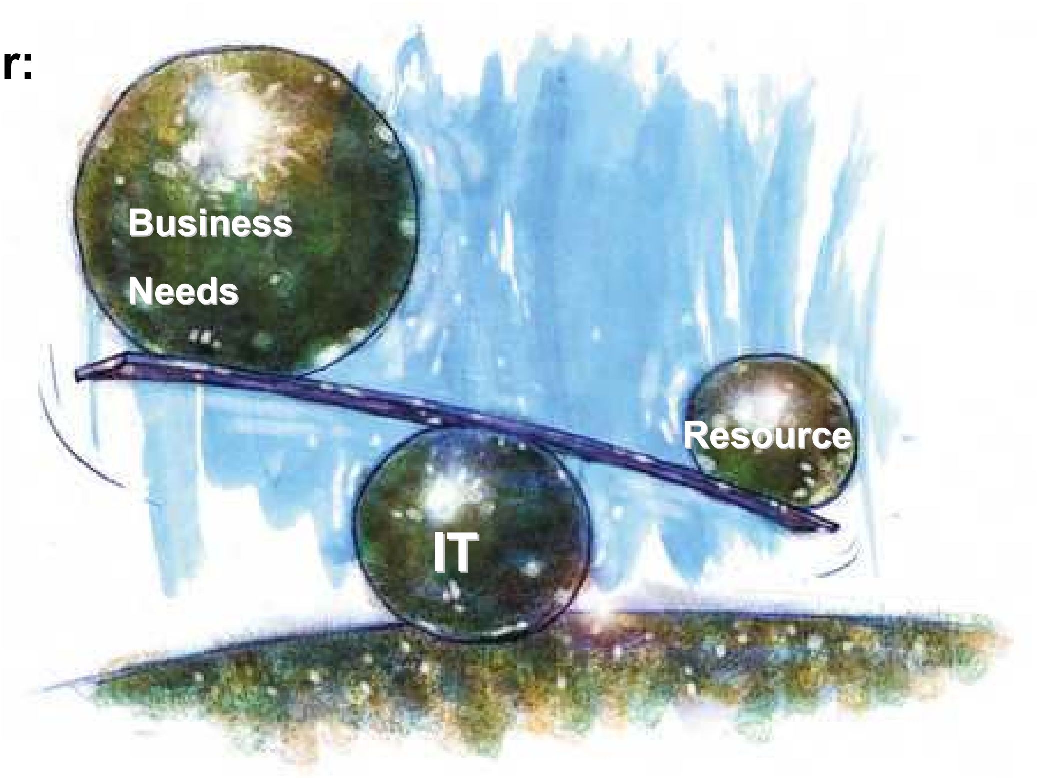
Software Development That Supports IT Agility

— **Software Development Must Deliver:**

- ✓More business solutions ↑
- ✓More flexibility ↑
- ✓More quality ↑
- ✓More business value ↑

-yet-

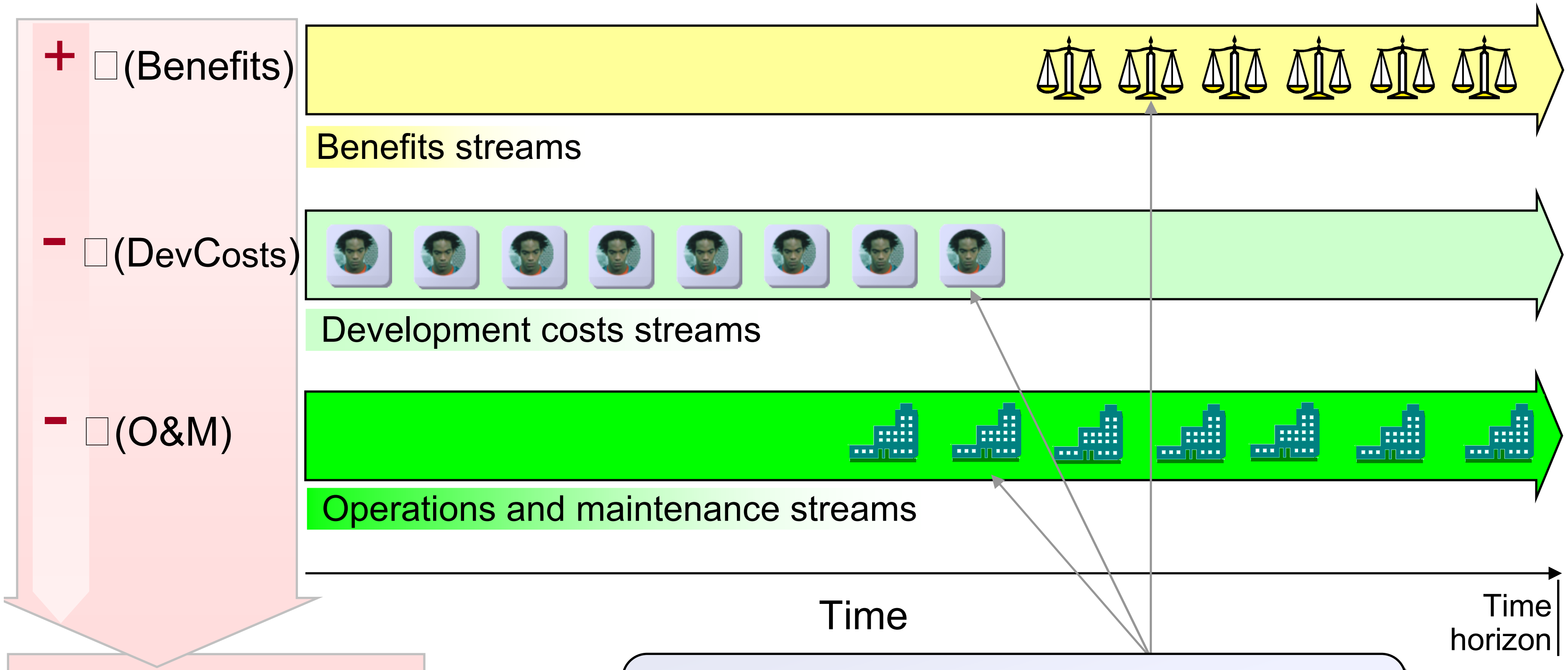
- ✓Less risk ↓
- ✓Less costs ↓
- ✓Less or flat resources ↓



Effective software development promotes a delicate balance of resources, reducing costs while delivering more, higher-quality business solutions.

In the end, it is about return on investments made

Investment Analytics: Estimating streams of value accrued over time



+ □(Benefits)

Benefits streams

- □(DevCosts)

Development costs streams

- □(O&M)

Operations and maintenance streams

Time

Time horizon

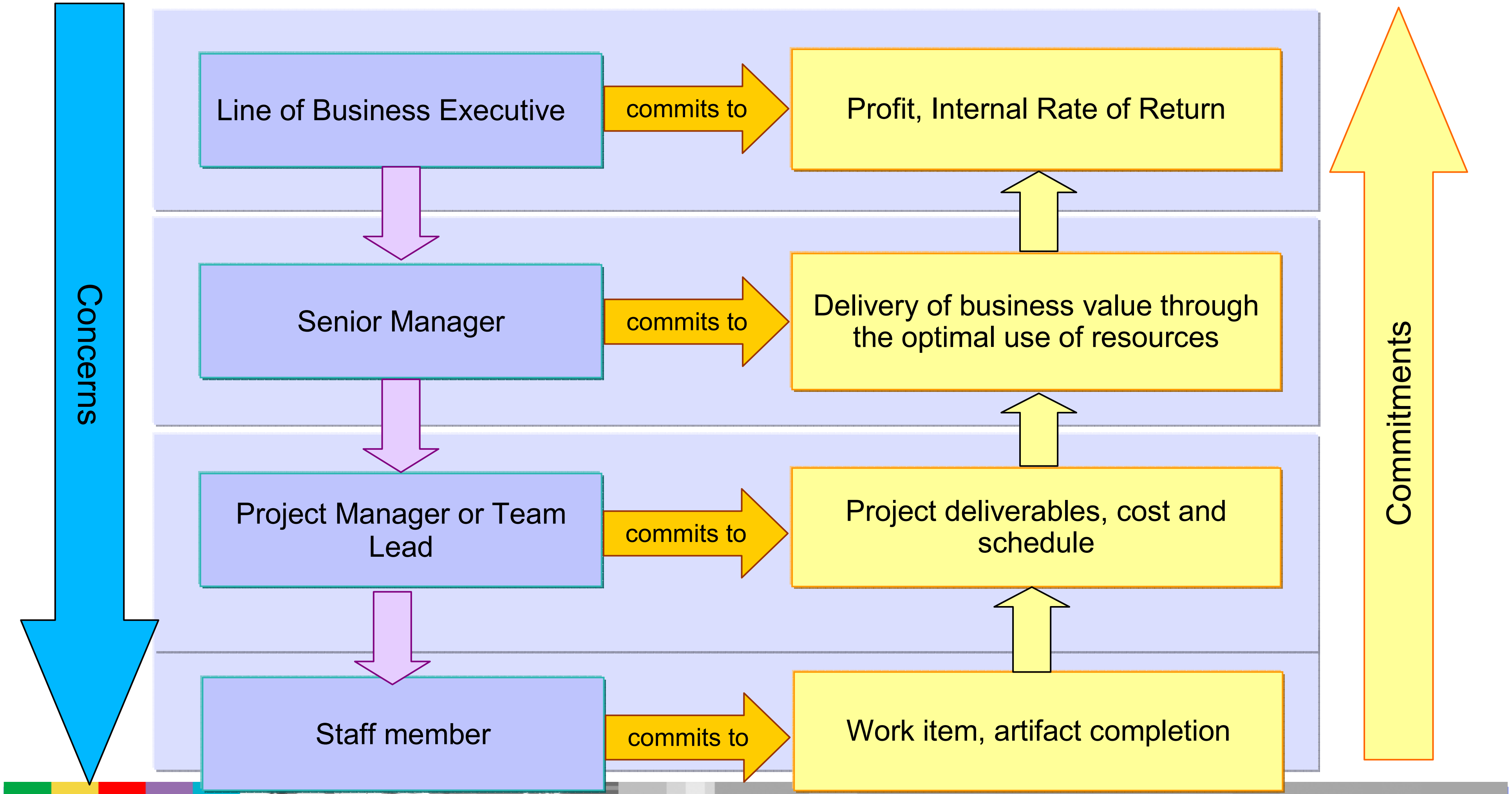
Investment Value expressed as NPV

Capability provided by Investment Analyzer, a component of Rational FocalPoint

- Each symbol represents a *variable* indicating an uncertain monetary amount
- Each symbol's amount is appropriately discounted; all are summed (□) to obtain NPV

Organization Dynamics

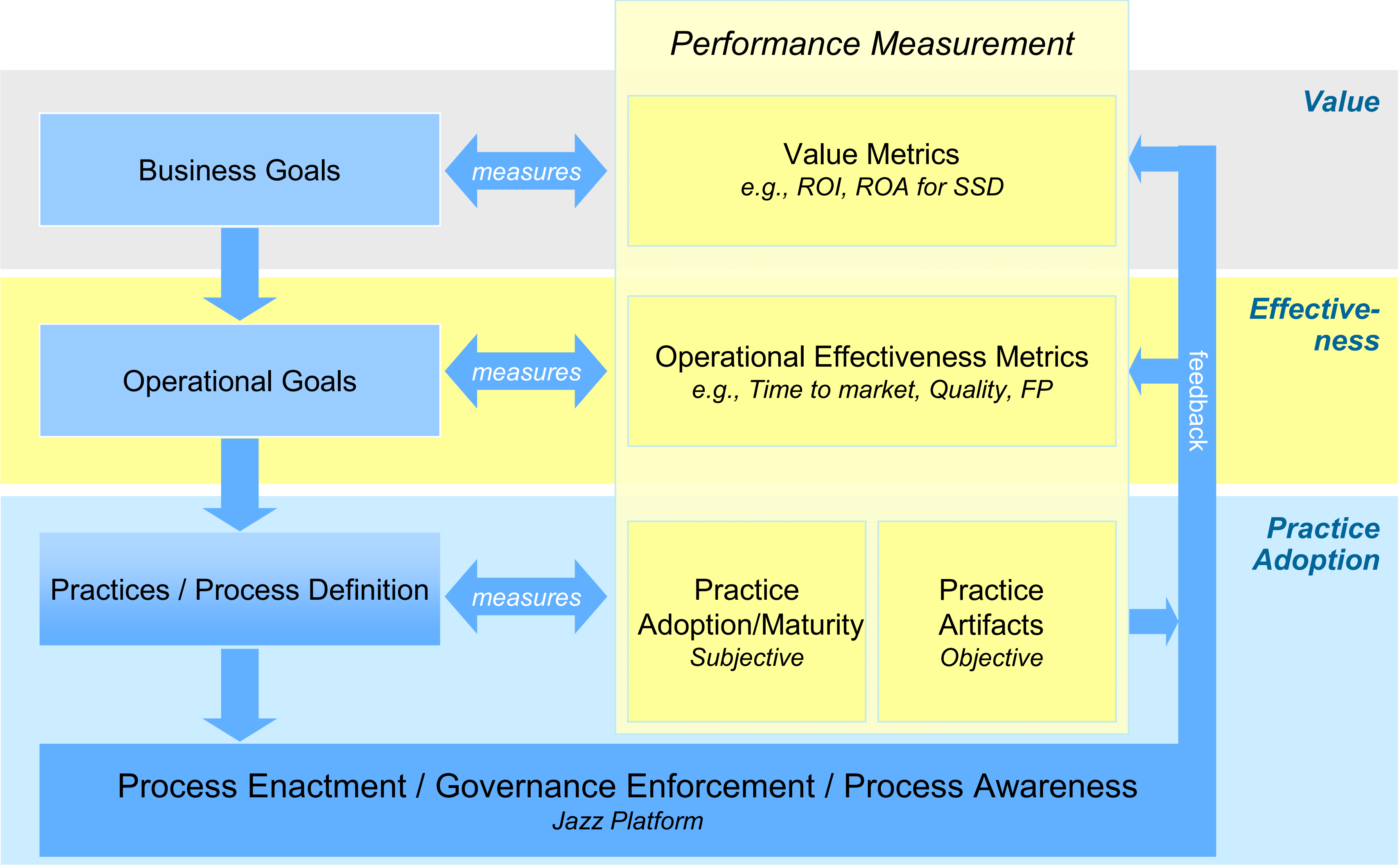
- Concerns flow down the organization while measures (and data) flow up
- We need tools to plan, track, and deliver on our commitments at every level
- Productivity needs to be measured at every level, and it is measured differently



Concerns
Line of Business Executive
commits to
Profit, Internal Rate of Return
Senior Manager
commits to
Delivery of business value through the optimal use of resources
Project Manager or Team Lead
commits to
Project deliverables, cost and schedule
Staff member
commits to
Work item, artifact completion
Commitments

To be practical, a metrics framework is needed

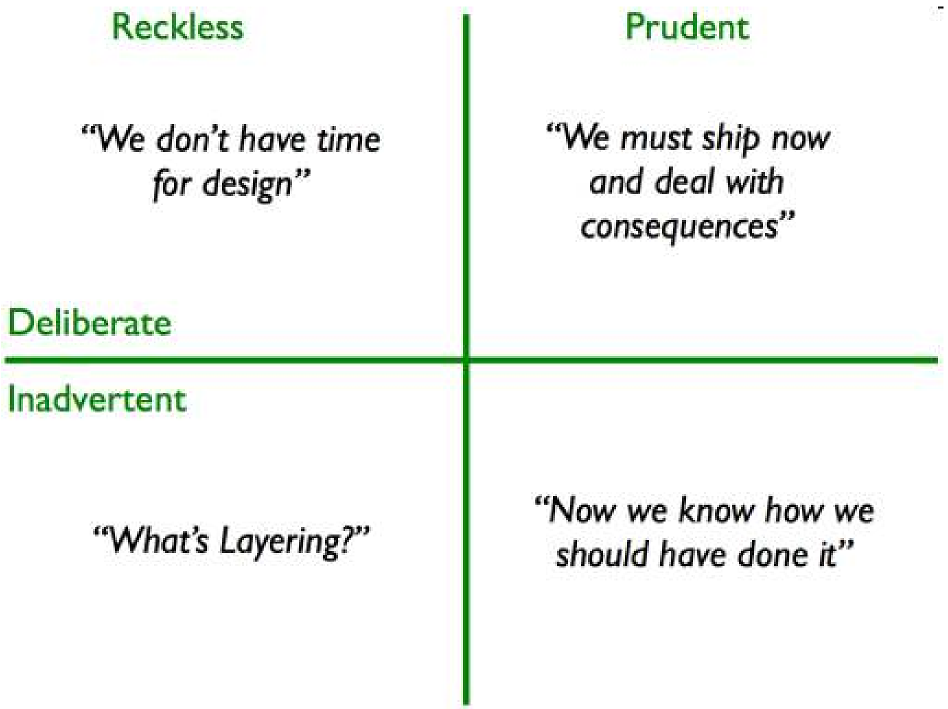
Measured Capability Improvement Framework



Accruing Technical Debt

Activities that a development team or a team member choose not to do well now, which will impede future development if left undone

- A financial liability
- Technical debt incurs interest payments:
  - Extra effort spent in future development tasks due to less effective design – translates to labor hours.
- We can choose to continue to pay that interest, or pay down the principal by investing labor in refactoring the design to something more effective.
- Some are deliberate decisions, e.g. to meet deadlines
  - If known, you can attempt an assessment of the interest cost vs. refactoring cost
- Others are inadvertent and not known



Reckless

Prudent

"We don't have time for design"

"We must ship now and deal with consequences"

Deliberate

Inadvertent

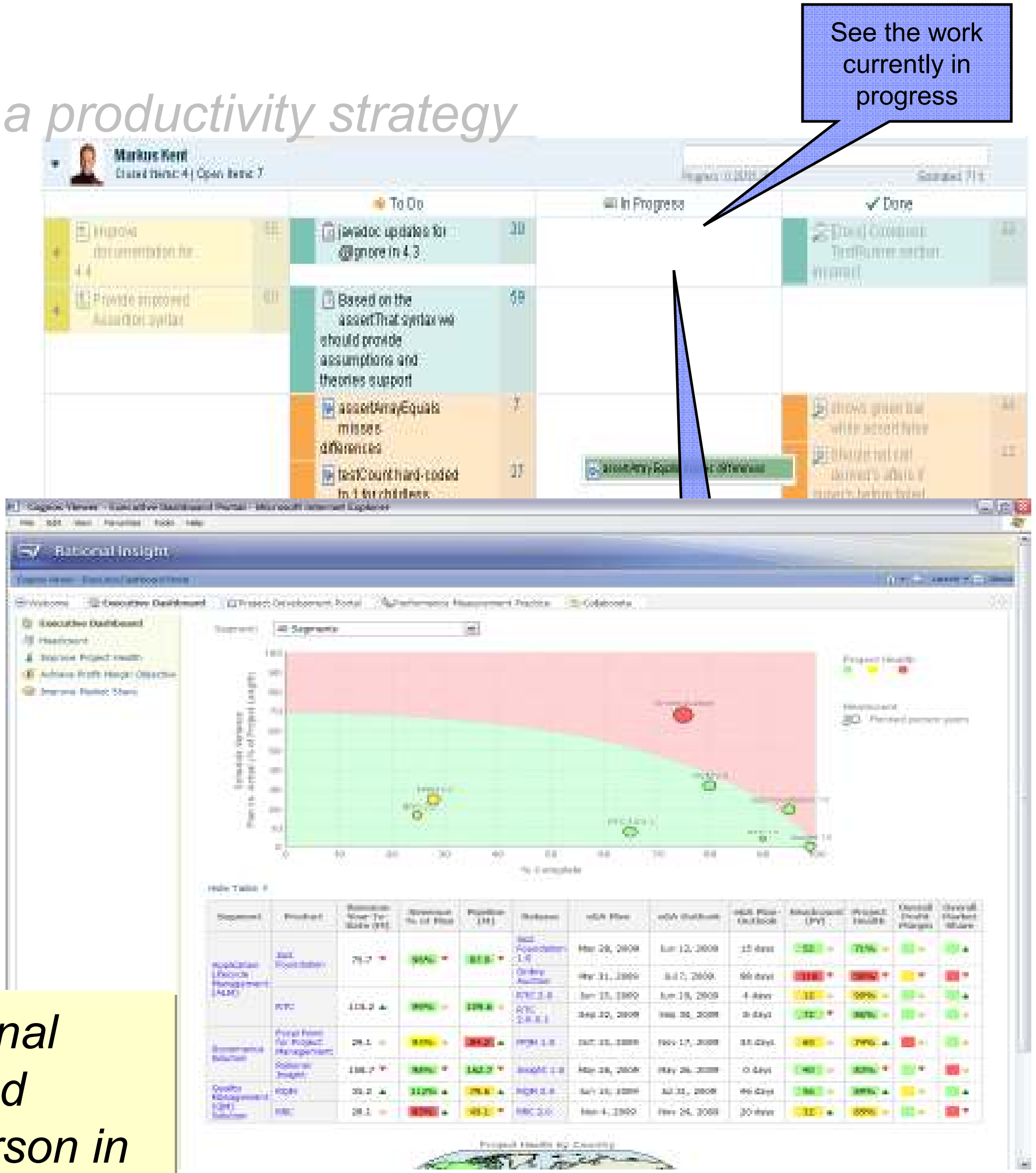
"What's Layering?"

"Now we know how we should have done it"

Ward Cunningham, Oopsla 1992: http://c2.com/doc/oopsla92.html
Martin Fowler: TechnicalDebt, TechnicalDebtQuadrant
Chris Sterling: Managing Software Debt: Building for Inevitable Change

Questions we hear ...

Evidence of the emerging need for a productivity strategy


See the work currently in progress
Markus Kent
Closed Items: 4 | Open Items: 7
To Do
In Progress
Done
Improve documentation for 4.4
Provide improved Assertion syntax
javadoc updates for @Ignore in 4.3
Based on the assertThat syntax we should provide assumptions and theories support
assertArrayEquals misses differences
testCount hard-coded to 1 for childless
[Docs] Combine TestRunner section in install
shows green bar when assert false
Rational Insight
Executive Dashboard
% Complete

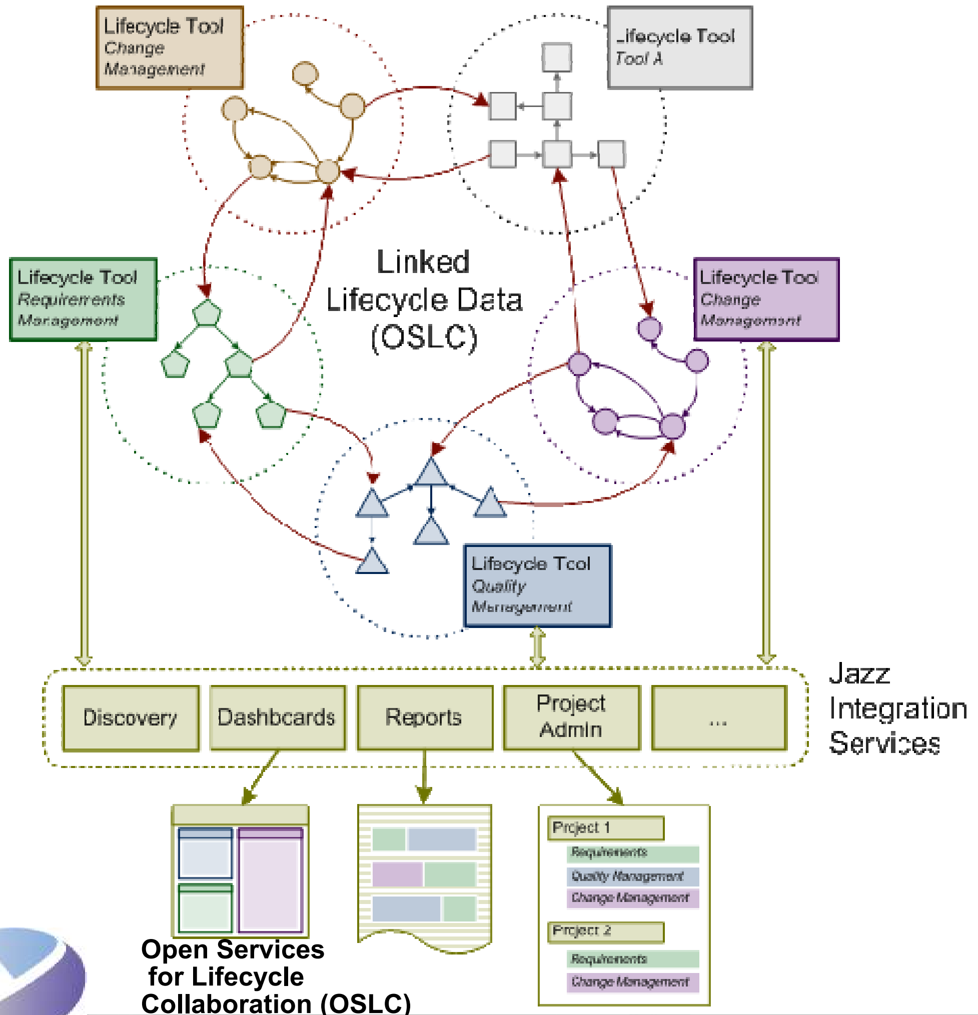
- How can we support team collaboration better?
  - Taskboards (developer cock pit) track progress
- How can we make status of projects more transparent?
  - Dashboards can provide insight
- How can we achieve improved operational efficiency?
  - Executive level dashboards to support prioritization
  - Portfolio management
  - Portfolio analytics (emerging)

The observation is that a foundational challenge is to provide accurate and relevant information to the right person in real time – in other words: transparency

The Jazz Architecture: An open architecture for lifecycle tool integration

Focusing on integrating the information not the development tools

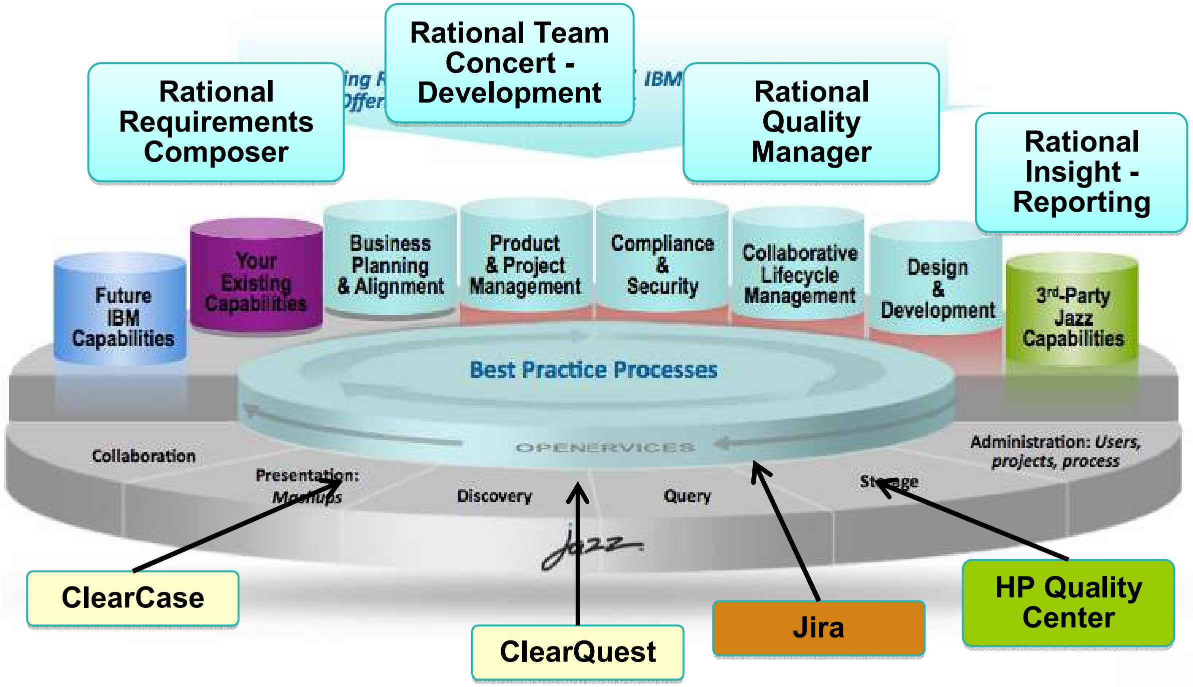
- **Built for the 21st century**: designed using Web architectural principles, implemented with Web technologies
- **Realistic**: recognizes that customers will not replace their current investments wholesale
- **Pragmatic**: allows tools and services to be upgraded independently, without sacrificing rich integration
- **Open**: supports the requirement to have a variety of tools from different sources – 3rd party as well as open source/cheap ware



Open Services for Lifecycle Collaboration (OSLC)

Envisioning a platform that can transform software delivery

Transparency: Dashboards, taskboards, reporting, seamless integration


Rational Requirements Composer
Rational Team Concert - Development
Rational Quality Manager
Rational Insight - Reporting
Future IBM Capabilities
Your Existing Capabilities
Business Planning & Alignment
Product & Project Management
Compliance & Security
Collaborative Lifecycle Management
Design & Development
3rd-Party Jazz Capabilities
Best Practice Processes
OPENSERVICES
Collaboration
Presentation: Mashups
Discovery
Query
Storage
Administration: Users, projects, process
Jazz
ClearCase
ClearQuest
Jira
HP Quality Center

Jazz is a project and platform for *transforming how people work together* to deliver greater value and performance from their software investments.

- A scalable, extensible team collaboration platform
- A community at Jazz.net where you can see Jazz-based products being built
- Our vision of the future of systems and software delivery, supporting globally distributed teams
- An integration architecture enabling mashups and non-Jazz based products to participate
- An evolution of our portfolio which will evolve to leverage Jazz technology over time

Examples

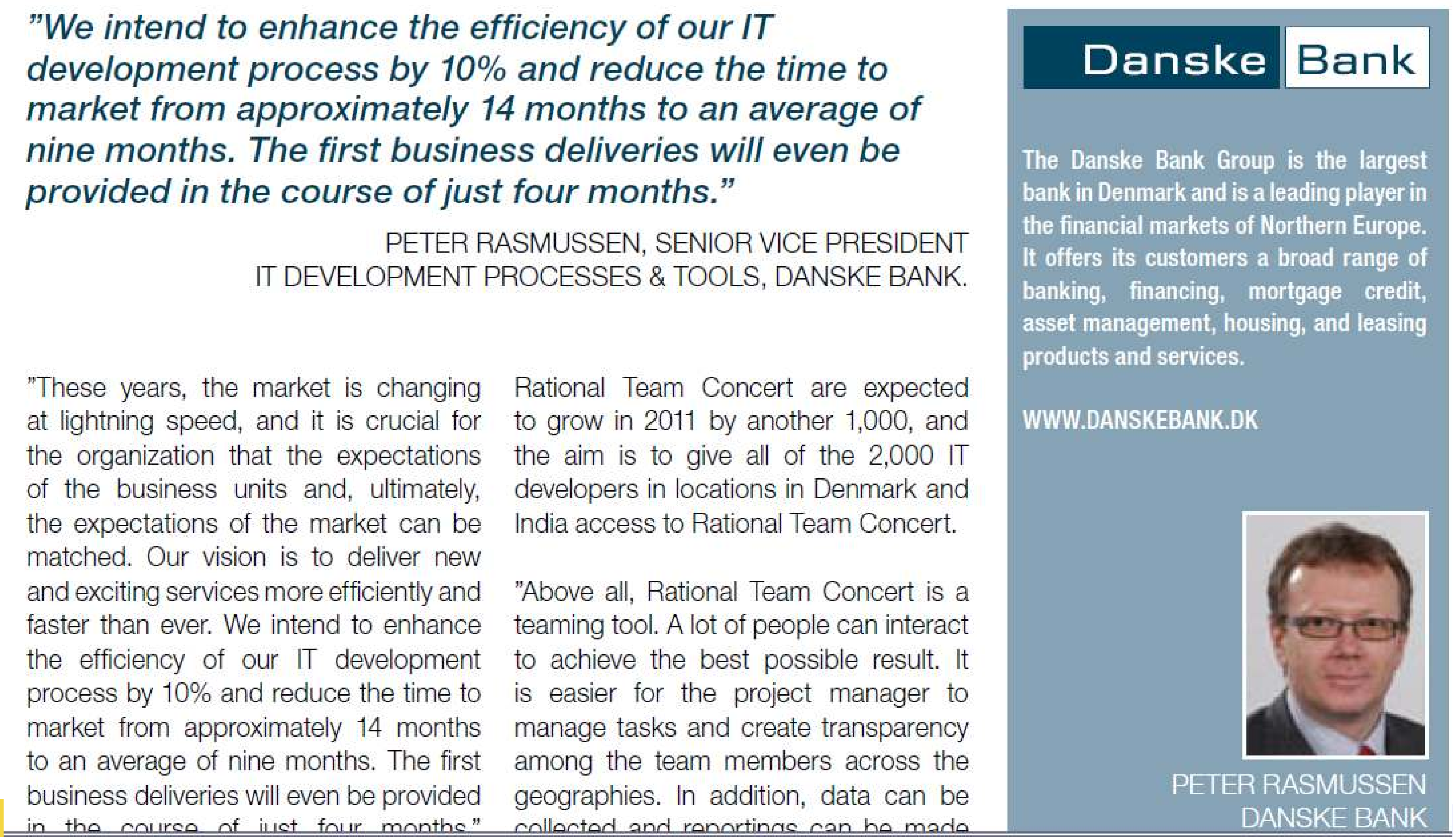
Danske Bank

IBM Development

Danske Bank: Measuring a Large-scale Agile Process Improvement Effort

- 2000+ developers
- 6 business units
- Development teams are often geographically distributed

Danske Bank Results



"We intend to enhance the efficiency of our IT development process by 10% and reduce the time to market from approximately 14 months to an average of nine months. The first business deliveries will even be provided in the course of just four months."

PETER RASMUSSEN, SENIOR VICE PRESIDENT
IT DEVELOPMENT PROCESSES & TOOLS, DANSKE BANK.

"These years, the market is changing at lightning speed, and it is crucial for the organization that the expectations of the business units and, ultimately, the expectations of the market can be matched. Our vision is to deliver new and exciting services more efficiently and faster than ever. We intend to enhance the efficiency of our IT development process by 10% and reduce the time to market from approximately 14 months to an average of nine months. The first business deliveries will even be provided in the course of just four months."

Rational Team Concert are expected to grow in 2011 by another 1,000, and the aim is to give all of the 2,000 IT developers in locations in Denmark and India access to Rational Team Concert.

"Above all, Rational Team Concert is a teaming tool. A lot of people can interact to achieve the best possible result. It is easier for the project manager to manage tasks and create transparency among the team members across the geographies. In addition, data can be collected and reportings can be made

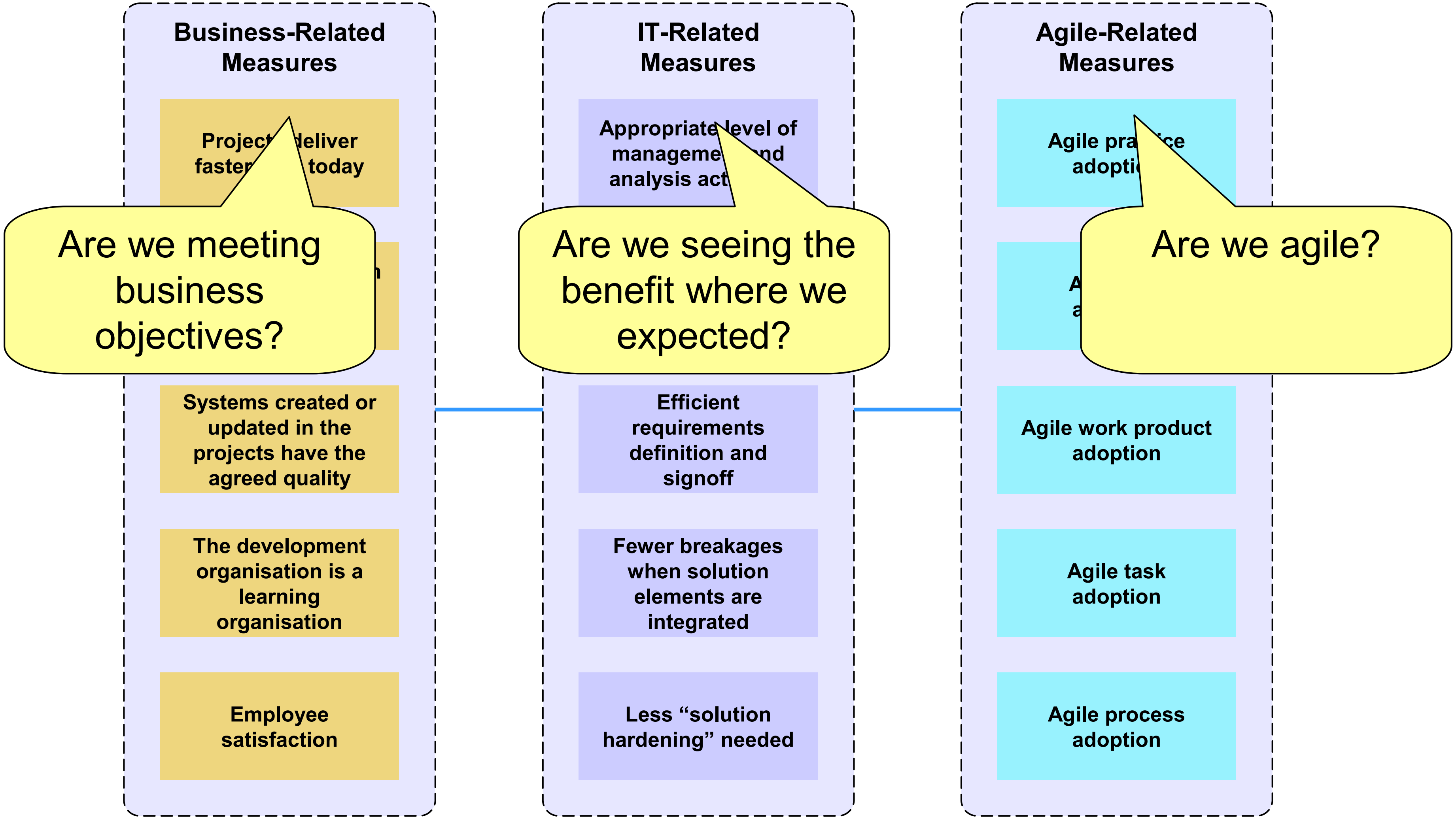
Danske Bank

The Danske Bank Group is the largest bank in Denmark and is a leading player in the financial markets of Northern Europe. It offers its customers a broad range of banking, financing, mortgage credit, asset management, housing, and leasing products and services.

WWW.DANSKEBANK.DK

PETER RASMUSSEN
DANSKE BANK

Measures help answer key questions



Business-Related Measures
Projec deliver
faste today
n
Systems created or updated in the projects have the agreed quality
The development organisation is a learning organisation
Employee satisfaction
IT-Related Measures
Appropriate level of manageme nd analysis act
Efficient requirements definition and signoff
Fewer breakages when solution elements are integrated
Less “solution hardening” needed
Agile-Related Measures
Agile pra ice adopti
A a
Agile work product adoption
Agile task adoption
Agile process adoption
Are we meeting business objectives?
Are we seeing the benefit where we expected?
Are we agile?

The business case for a more efficient development organisation

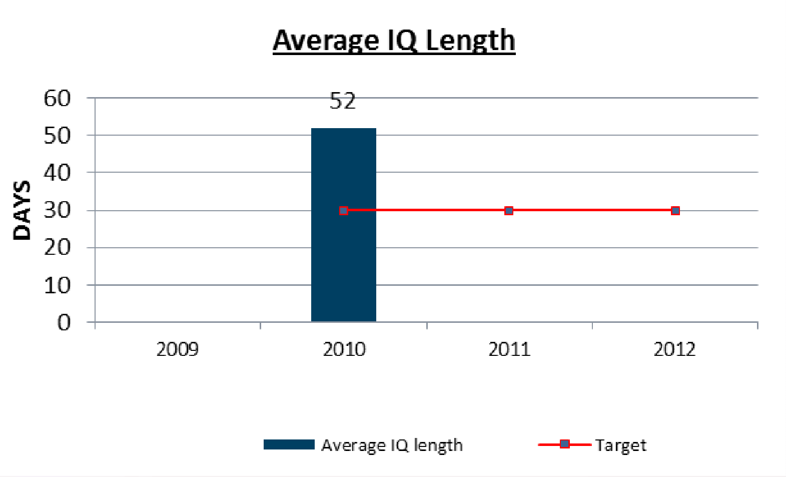
The target: 10% increase in efficiency over a 3 year period

- Reduced time-to-market (faster)
- Reduced cost (cheaper)
- The value of the solution (the best solution)
- The quality of the solution (the right quality)

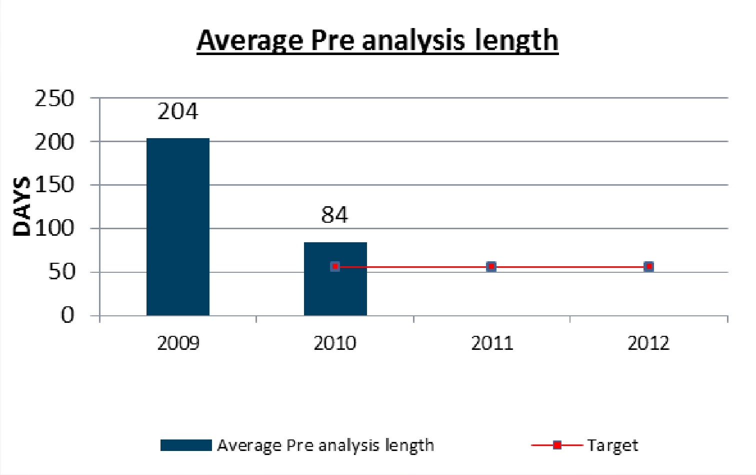
The measurements :

- Idea Qualification phase (IQ) duration
- Pre-analysis phase duration
- Average Time to Market for first business release
- Average CMMI Level-3 Score
- Average (Effort/Function Point) Productivity

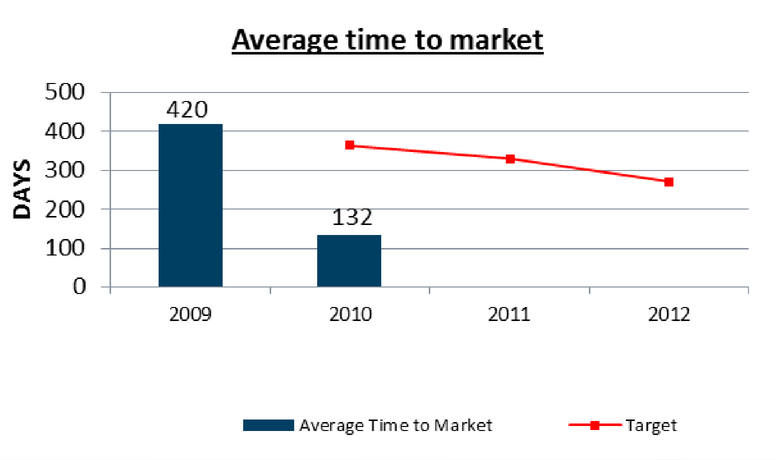
The results – so far



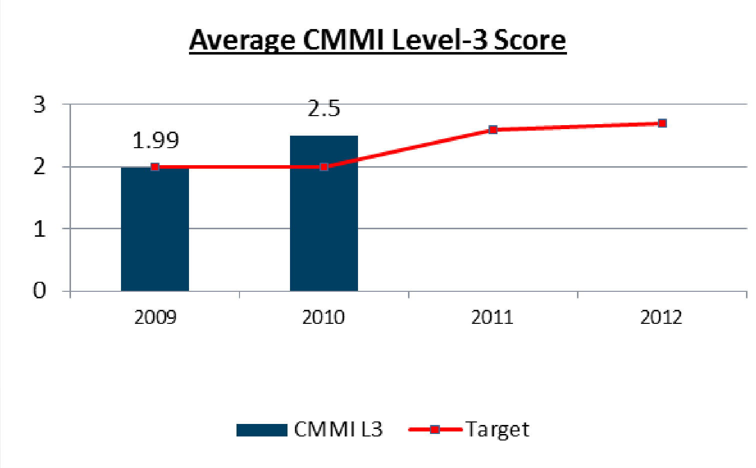
Average IQ Length



Average Pre analysis length



Average time to market



Average CMMI Level-3 Score

Conclusions

Change of focus to results, not documents

- Moved from planning activities to planning deliveries
- Still overly precise plans and requirements

Faster time to market

- IQ phase forces early collaboration on design
- Big uncertainties are addressed earlier

Improved collaboration among stakeholders

- Honest measures to identify the right improvements.
- Moving past cherry-picking

The IBM Story: Five Years Ago - Our Pain Points…

- ✓ joining a team
- ✓ get my environment configured to be productive
- ✓ what is happening in my team
- ✓ collecting progress status
- ✓ following the team's process
- ✓ ad hoc collaboration/sharing of changes
- ✓ starting an ad hoc team

Team awareness

- ✓ is the fix in the build?
- ✓ run a personal build
- ✓ tracking a broken build
- ✓ why is this change in the build?
- ✓ reconstructing a context for a bug/build failure

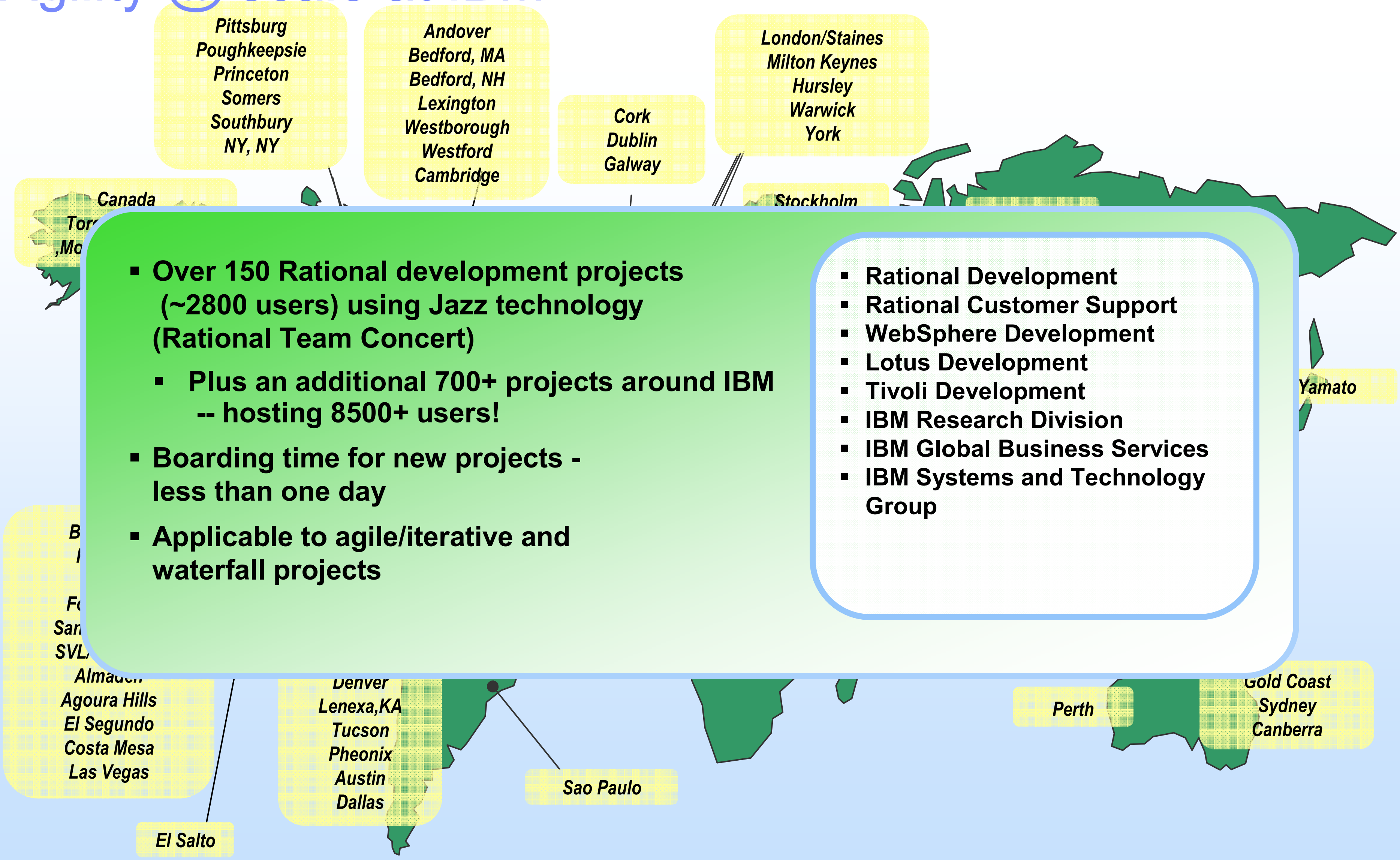
Build awareness

- ✓ interrupting development due to a high priority bug fix
- ✓ working on multiple releases concurrently
- ✓ tracking the code review of a fix
- ✓ referencing team artifacts in discussions
- ✓ how healthy is a component?
- ✓ collecting project data/metrics?
- ✓ keeping plans up to date

Project awareness

Boring and painful

Agility @ scale at IBM



- **Over 150 Rational development projects (~2800 users) using Jazz technology (Rational Team Concert)**
  - **Plus an additional 700+ projects around IBM -- hosting 8500+ users!**
- **Boarding time for new projects - less than one day**
- **Applicable to agile/iterative and waterfall projects**

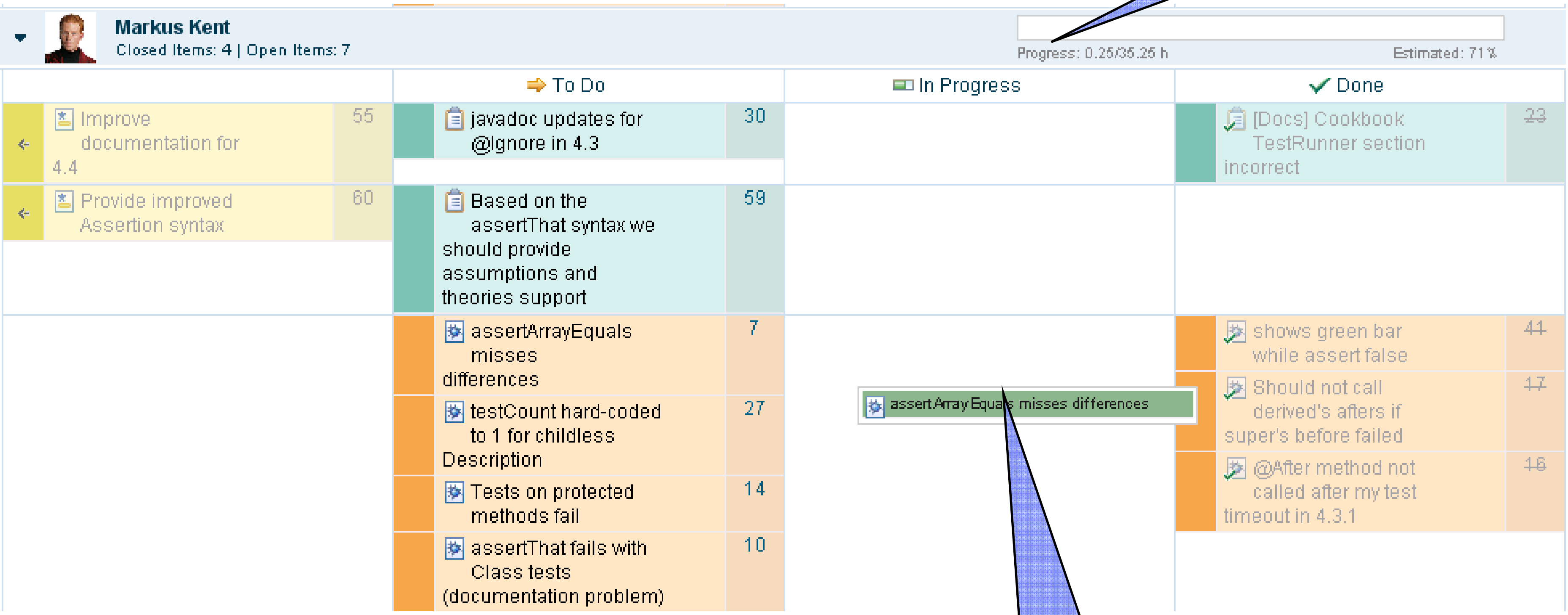
- **Rational Development**
- **Rational Customer Support**
- **WebSphere Development**
- **Lotus Development**
- **Tivoli Development**
- **IBM Research Division**
- **IBM Global Business Services**
- **IBM Systems and Technology Group**

21

Team Awareness: Taskboards track Work in Progress

Accelerate stand-up meetings, increased transparency

See the work currently in progress



Markus Kent
Closed Items: 4 | Open Items: 7

Progress: 0.25/35.25 h Estimated: 71 %

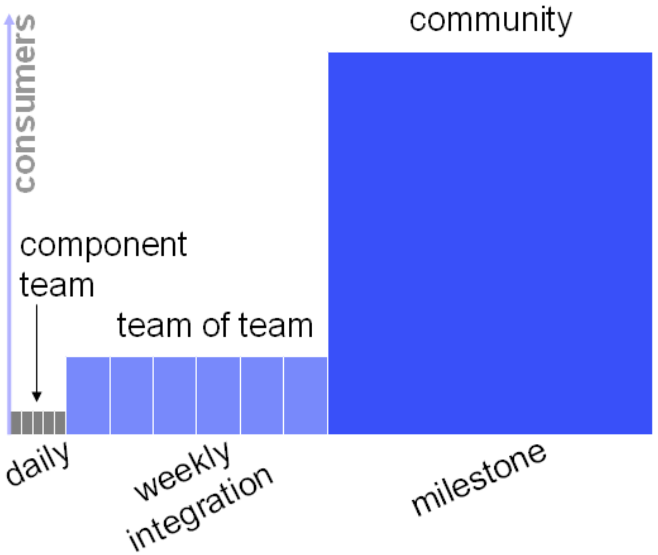
	To Do	In Progress	Done
Improve documentation for 4.4 55	javadoc updates for @Ignore in 4.3 30		[Docs] Cookbook TestRunner section incorrect ~~23~~
Provide improved Assertion syntax 60	Based on the assertThat syntax we should provide assumptions and theories support 59		
	assertArrayEquals misses differences 7 testCount hard-coded to 1 for childless Description 27 Tests on protected methods fail 14 assertThat fails with Class tests (documentation problem) 10	assertArrayEquals misses differences	shows green bar while assert false ~~44~~ Should not call derived's afters if super's before failed ~~17~~ @After method not called after my test timeout in 4.3.1 ~~16~~

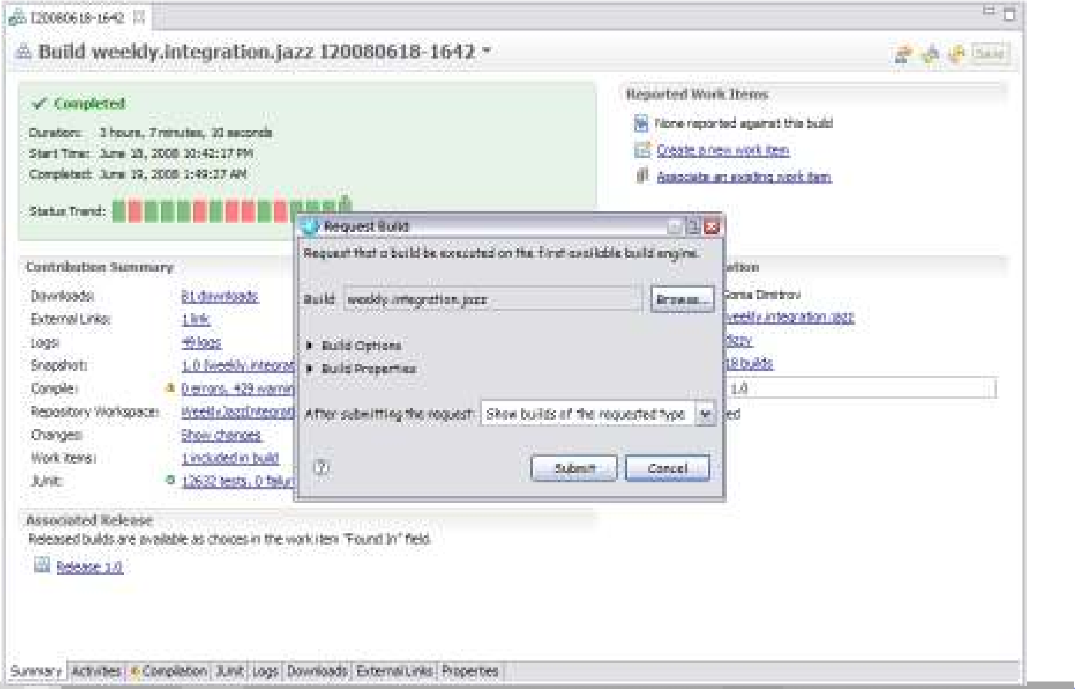
Drag and drop work items to change their state.

22

Build Awareness

- Team of Team
  - Scheduled weekly integration build
    - Stabilized until green
  - Continuous integration stream
    - Share changes, rarely green
- Each Team
  - Continuous local builds
  - Always green
- A Developer
  - Personal builds




I20080618-1642

Build weekly.integration.jazz I20080618-1642

✓ Completed

Duration: 3 hours, 7 minutes, 10 seconds
Start Time: June 18, 2008 10:42:17 PM
Completed: June 19, 2008 1:49:27 AM

Status Trend: 🟩🟥🟩🟩🟩🟥🟩🟥🟥🟩🟥🟩🟩🟩🟩

Reported Work Items

None reported against this build
Create a new work item
Associate an existing work item

Contribution Summary

Downloads:	81 downloads
External Links:	1 link
Logs:	49 logs
Snapshot:	1.0 (weekly.integra
Compile:	0 errors, 429 warnin
Repository Workspace:	WeeklyJazzIntegrat
Changes:	Show changes
Work Items:	1 included in build
JUnit:	12632 tests, 0 failu

Request Build

Request that a build be executed on the first available build engine.

Build: weekly.integration.jazz

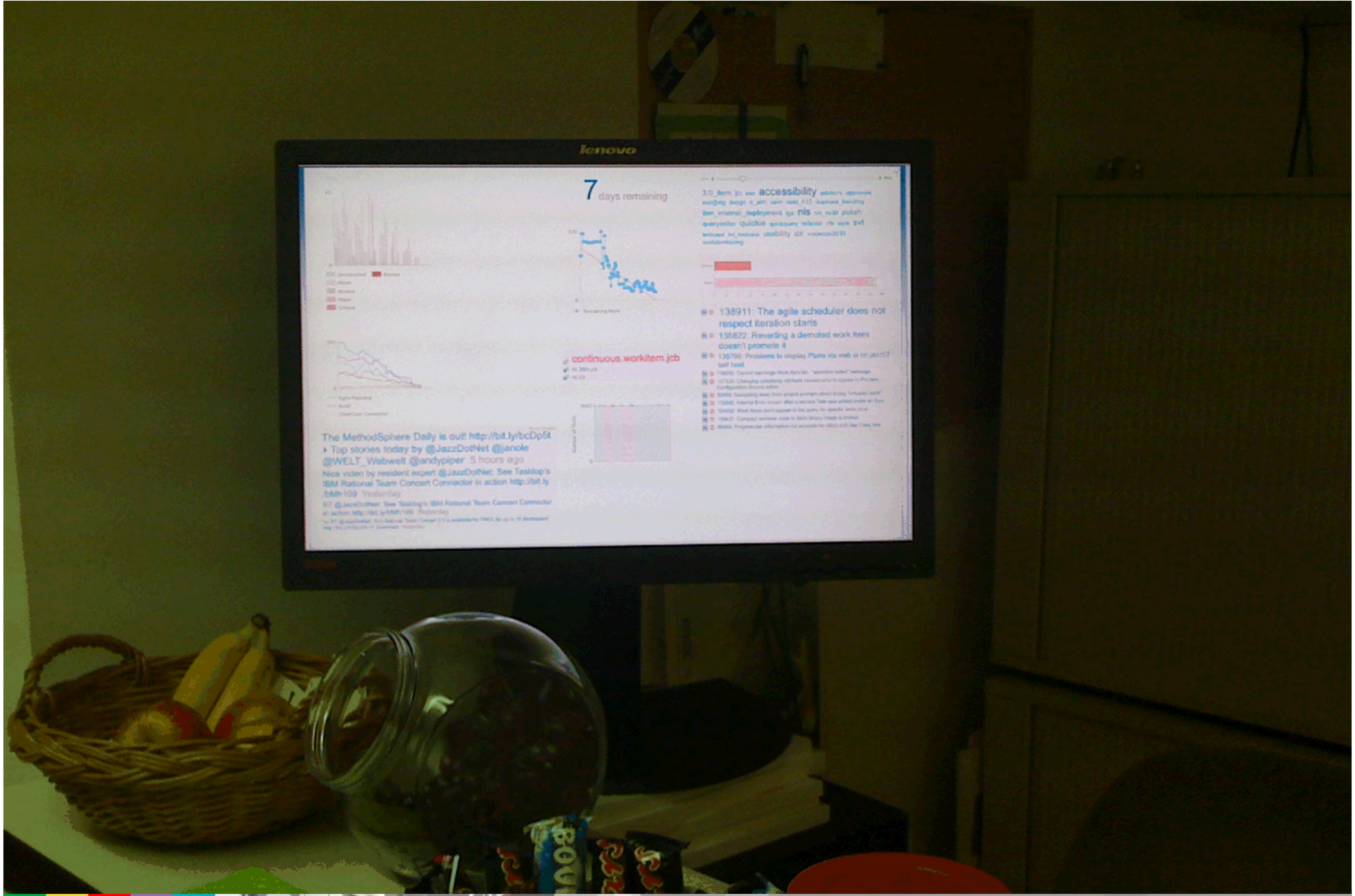
After submitting the request: Show builds of the requested type

Sonia Dimitrov
weekly.integration.jazz
1.0

Associated Release

Released builds are available as choices in the work item "Found In" field.

Release 1.0



138911: The agile scheduler does not respect iteration starts

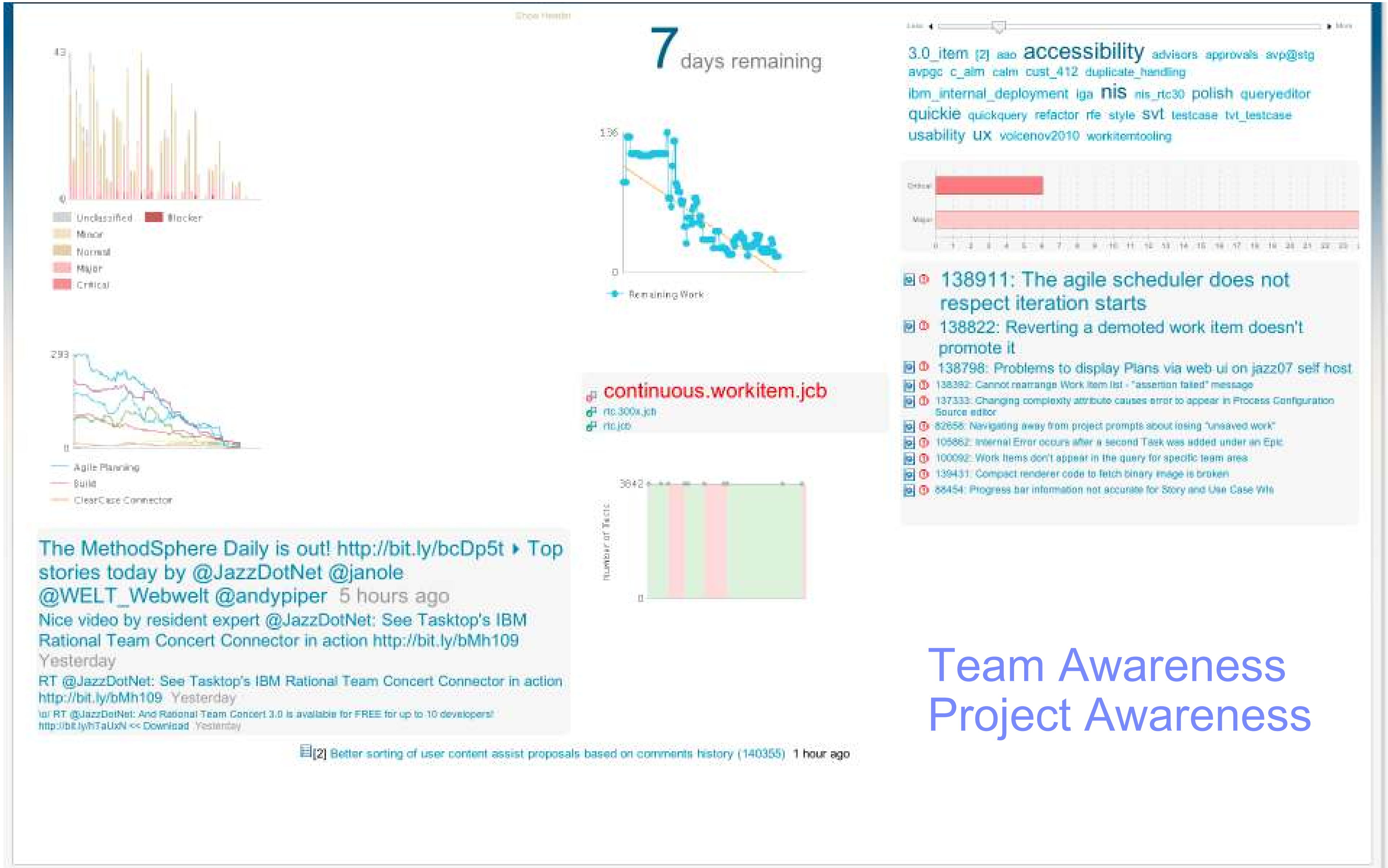
138822: Reverting a demoted work item doesn't promote it

138798: Problems to display Plans via web ui on jazz47 self host

The MethodSphere Daily is out! http://bit.ly/bcDp5t ▸ Top stories today by @JazzDotNet @janole @WELT_Webwelt @andypiper 5 hours ago

Nice video by resident expert @JazzDotNet: See Tasktop's IBM Rational Team Concert Connector in action http://bit.ly /bMh109 Yesterday

24



3.0_item [2] aao accessibility advisors approvals avp@stg avpgc c_alm calm cust_412 duplicate_handling ibm_internal_deployment iga nis nis_rtc30 polish queryeditor quickie quickquery refactor rfe style svt testcase tvt_testcase usability ux voicenov2010 workitemtooling

- 138911: The agile scheduler does not respect iteration starts
- 138822: Reverting a demoted work item doesn't promote it
- 138798: Problems to display Plans via web ui on jazz07 self host
- 138392: Cannot rearrange Work item list - "assertion failed" message
- 137333: Changing complexity attribute causes error to appear in Process Configuration Source editor
- 82656: Navigating away from project prompts about losing "unsaved work"
- 105862: Internal Error occurs after a second Task was added under an Epic
- 100092: Work items don't appear in the query for specific team area
- 139431: Compact renderer code to fetch binary image is broken
- 88454: Progress bar information not accurate for Story and Use Case WIs

continuous.workitem.jcb
rtc.300x.jcb
rtc.jcb

The MethodSphere Daily is out! http://bit.ly/bcDp5t ▸ Top stories today by @JazzDotNet @janole @WELT_Webwelt @andypiper 5 hours ago

Nice video by resident expert @JazzDotNet: See Tasktop's IBM Rational Team Concert Connector in action http://bit.ly/bMh109 Yesterday

RT @JazzDotNet: See Tasktop's IBM Rational Team Concert Connector in action http://bit.ly/bMh109 Yesterday

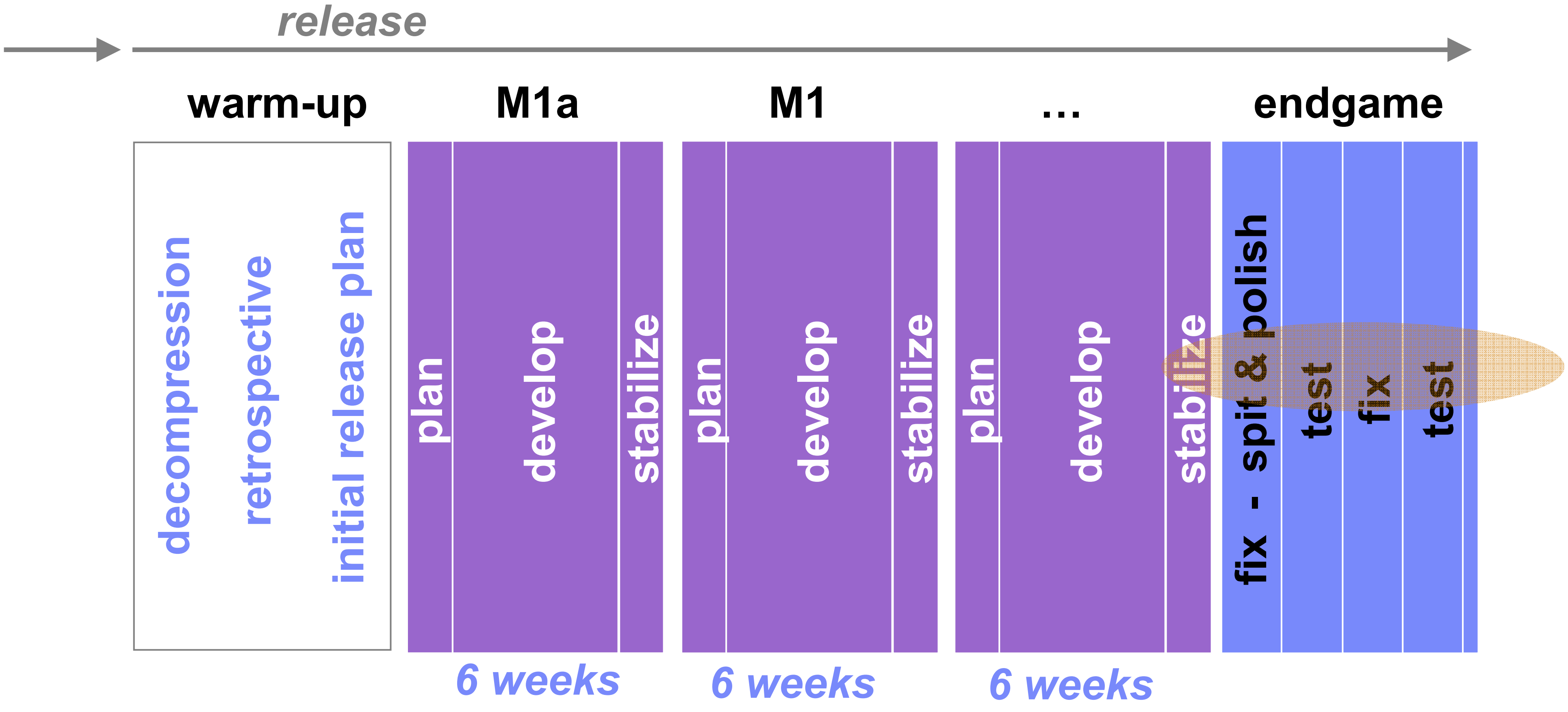
\o/ RT @JazzDotNet: And Rational Team Concert 3.0 is available for FREE for up to 10 developers! http://bit.ly/hTaUxN << Download Yesterday

[2] Better sorting of user content assist proposals based on comments history (140355) 1 hour ago

Team Awareness
Project Awareness

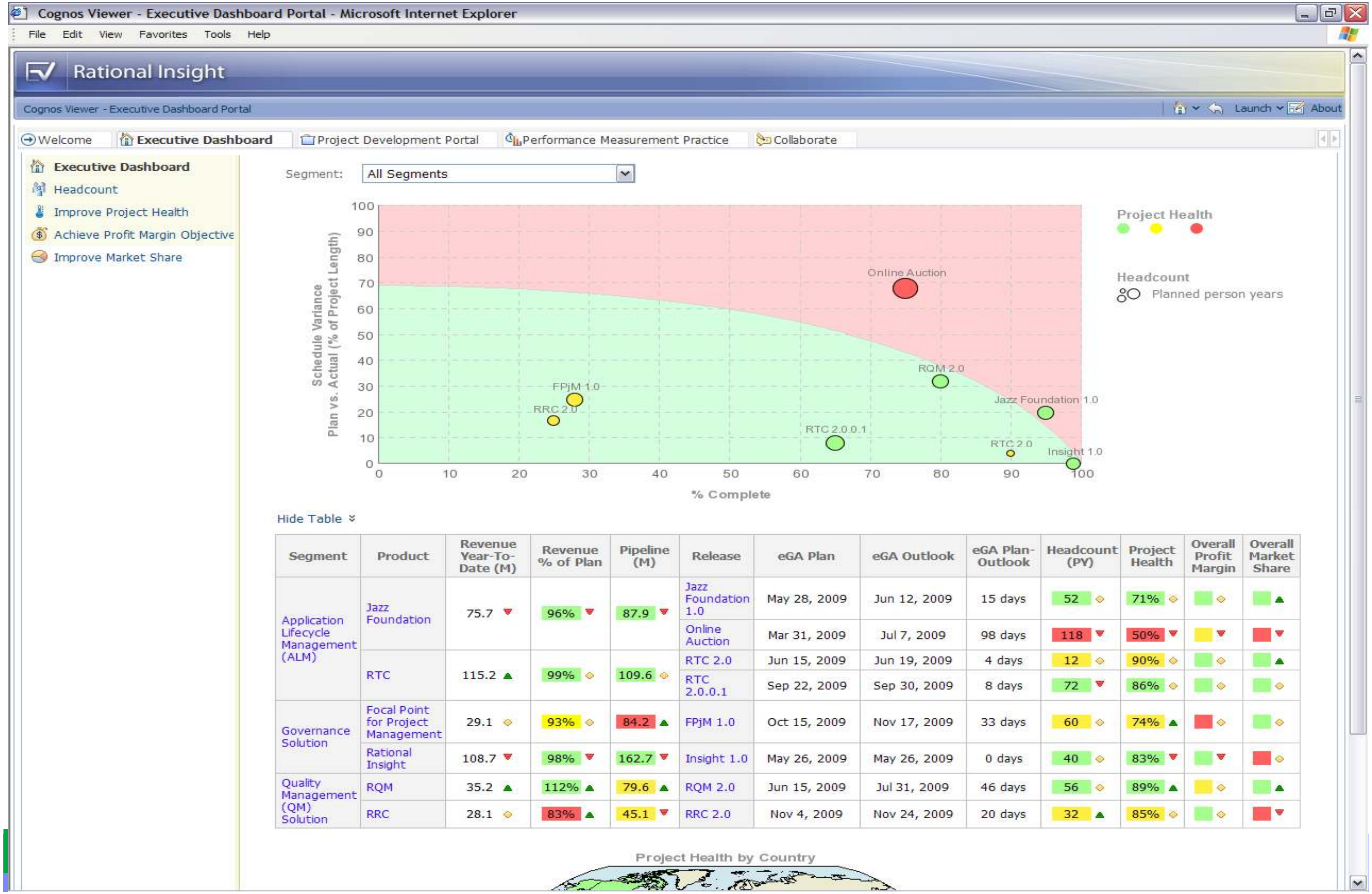
Project Awareness

Endgame Focused and Disciplined



26

Executive Dashboard



Segment	Product	Revenue Year-To-Date (M)	Revenue % of Plan	Pipeline (M)	Release	eGA Plan	eGA Outlook	eGA Plan-Outlook	Headcount (PY)	Project Health	Overall Profit Margin	Overall Market Share
Application Lifecycle Management (ALM)	Jazz Foundation	75.7 ▼	96% ▼	87.9 ▼	Jazz Foundation 1.0	May 28, 2009	Jun 12, 2009	15 days	52 ◇	71% ◇	🟩 ◇	🟩 ▲
					Online Auction	Mar 31, 2009	Jul 7, 2009	98 days	118 ▼	50% ▼	🟨 ▼	🟥 ▼
	RTC	115.2 ▲	99% ◇	109.6 ◇	RTC 2.0	Jun 15, 2009	Jun 19, 2009	4 days	12 ◇	90% ◇	🟩 ◇	🟩 ▲
					RTC 2.0.0.1	Sep 22, 2009	Sep 30, 2009	8 days	72 ▼	86% ◇	🟩 ◇	🟩 ◇
Governance Solution	Focal Point for Project Management	29.1 ◇	93% ◇	84.2 ▲	FPjM 1.0	Oct 15, 2009	Nov 17, 2009	33 days	60 ◇	74% ▲	🟥 ◇	🟩 ◇
	Rational Insight	108.7 ▼	98% ▼	162.7 ▼	Insight 1.0	May 26, 2009	May 26, 2009	0 days	40 ◇	83% ▼	🟩 ▼	🟥 ◇
Quality Management (QM) Solution	RQM	35.2 ▲	112% ▲	79.6 ▲	RQM 2.0	Jun 15, 2009	Jul 31, 2009	46 days	56 ◇	89% ▲	🟨 ◇	🟩 ▲
	RRC	28.1 ◇	83% ▲	45.1 ▼	RRC 2.0	Nov 4, 2009	Nov 24, 2009	20 days	32 ▲	85% ◇	🟩 ◇	🟥 ▼

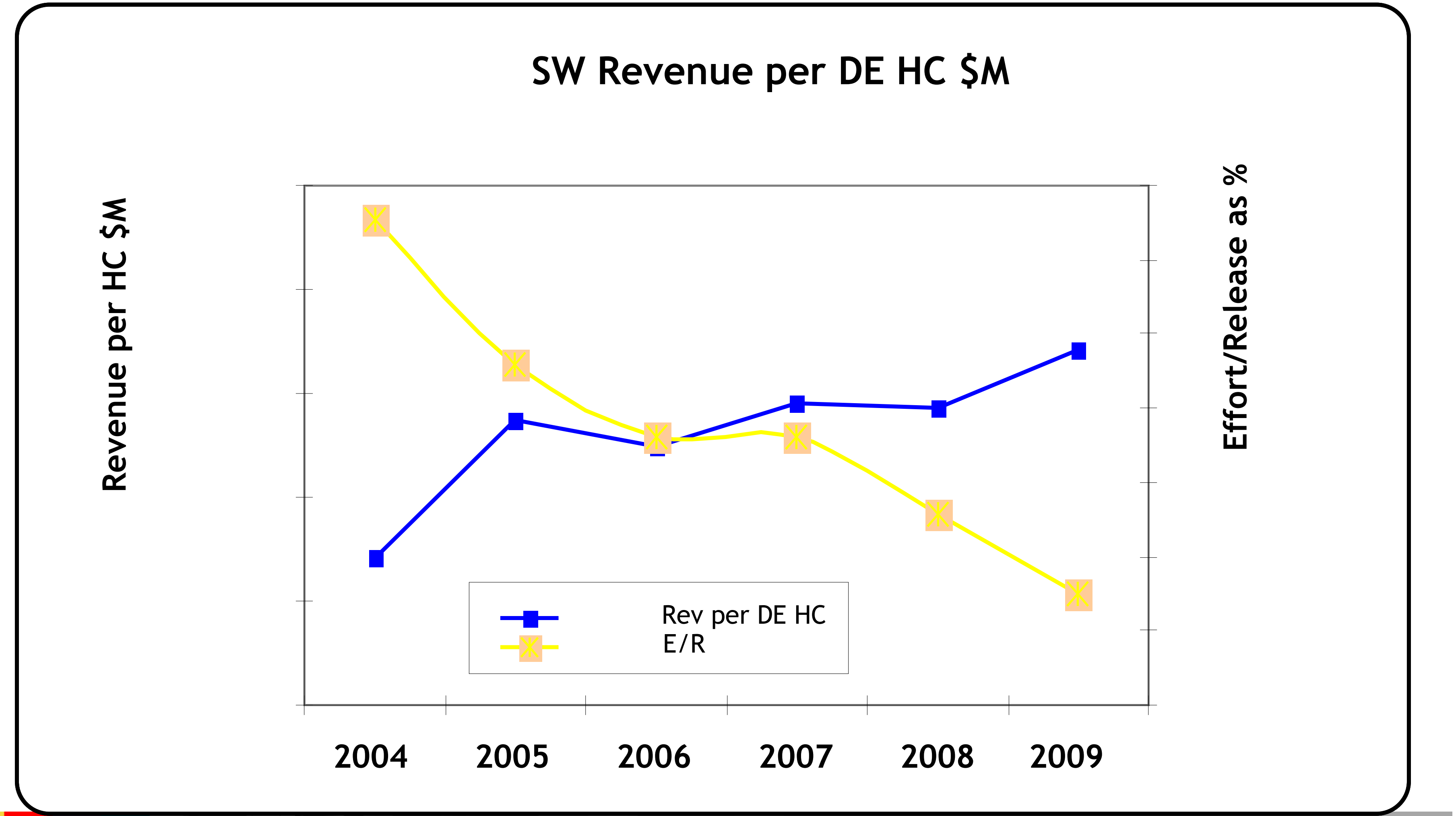
Tech Debt Measurements

Quality Focused

Metric	Goal	2006 Measurement	2010 Measurement
Maintenance / Innovation	50/50	42% / 58%	31% / 69%
Time to Market (Major)	12 Months	18 + Months	12.5 Months
Customer Calls	-5% YoY	~ 135,000	~100,000 (-19% since 2009)
Customer Defect Arrivals	-5% YoY	~ 6,900	~2200
On Time Delivery	65%	47%	92%
Defect Backlog	3 Months	9+ Months	3 months
Customer Sat Index	88%	83%	88%

***Note:** Goals are either internal IBM statistics or industry benchmarks.*

Improving Bottom-Line Growth



Conclusions

- In the end, development productivity is about return on development assets
- Technology is emerging that help us provide real-time feedback at all levels in an organization
- Transparency is key to leaps in productivity improvement

Thank
You