



 Drive Results.

# RTC for Product Development

Peer Törngren, Software Architect  
Cognos Controller



**IBM**

**RUC2014**  
Rational User Conference

The "badge" is the unofficial logo of Scrum team "Conso" – the team that is focused on the core business logic in our consolidation engine.

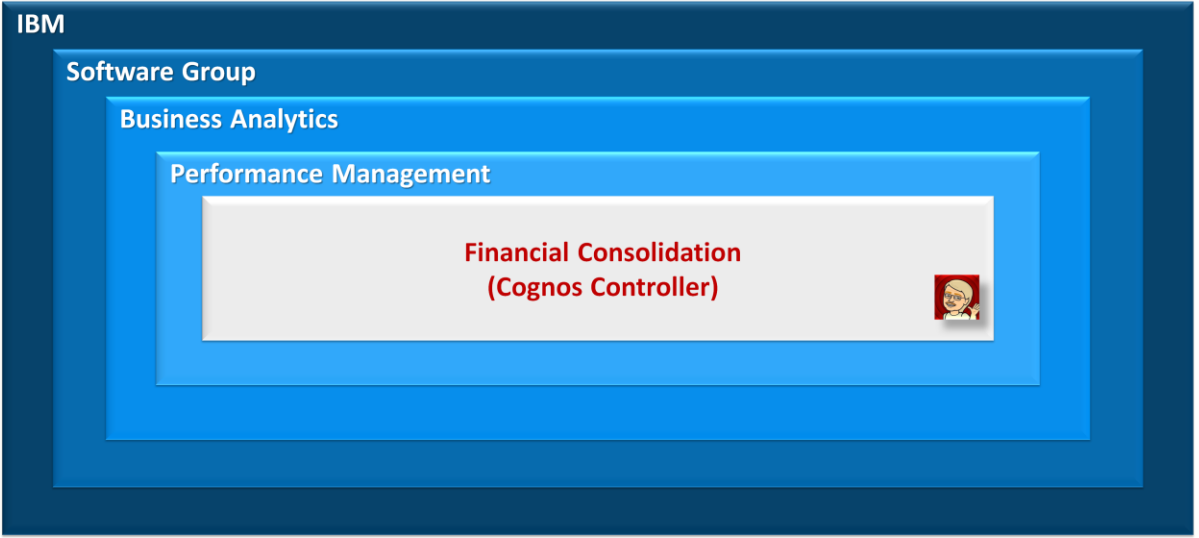


## Upscale Agile – Can Elephants Dance?

- Doing SCRUM (agile)
- Working for IBM (large)
- Using RTC (new)
  
- Lessons learned – what to expect



# Who We Are



# Our Mission



## Complete Coverage of Financial Governance Processes



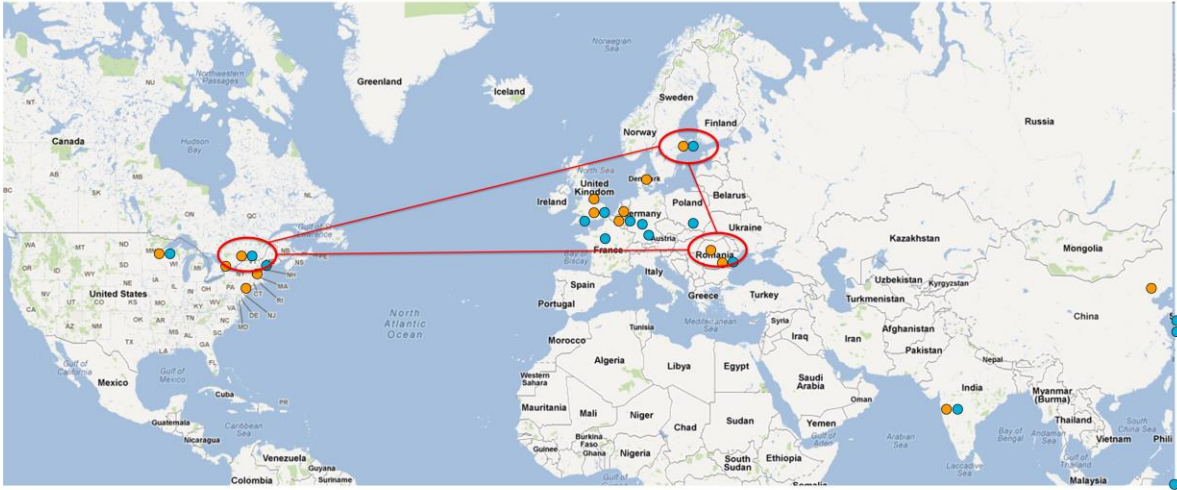
RUC2014



GAAP: Generally Accepted Accounting Principles (usually within a country)  
Mgmt Reporting: Financial performance reports for business management



# Our Team



Development: Stockholm, Copenhagen, York, Staines, Weert, Brussels, Bucharest, Cluj, Ottawa, Toronto, Princeton, Littleton, Pune, Beijing  
Support: Ottawa, Minneapolis, Boston, Rochester, Chicago, San Paulo, London, Farnborough, Stockholm, Paris, Brussels, Frankfurt, Munich, Bucharest, Krakow, Sydney, Tokyo, Seoul, Beijing, Pune

RUC2014



Most of the Controller development is done in Stockholm, Bucharest and Ottawa.



# Our Business

- Consolidation
- ~1000 customers (WW)
- ~5-500 users/customers

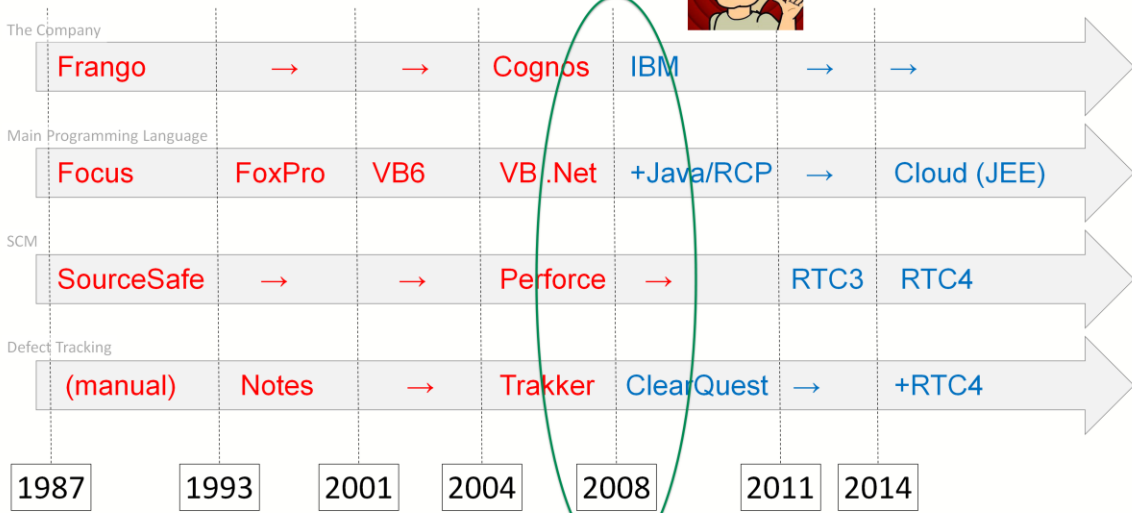


RUC2014





# Our Technology (So Far)





## About Me



**Software Architect/Developer**

(hands-on Java EE)

**Agile Coach/Methodologist**

(aka Process Fascist)

**Commercial Software Products**

(never seen a real user)





# Upscale Agile – Can Elephants Dance?

- Large organizations struggle with Agile methods
  - Distributed teams (vs co-located)
  - Conflicting projects (vs 100% focus)
  - Unplannable interruptions (vs sprint heartbeat)
  - Specialist roles (vs multi-talent team)
  - Specialist teams (vs cross-functional teams)
  - Project phases (vs potentially shippable increment)
- Practitioner's report
  - Does RTC help or hinder?
  - How?

RUC2014



## Specialist roles:

- UX (User Experience)
- DB (design, vendors)
- Domain Experts and specialized developers (Intercompany Elimination, Closing process, Currency Translation ...)

## Specialist teams:

- UX & Graphics
- QE (Test automation, load testing)
- Information Development (documentation, online help)
- Translation (contractors)
- Legal (lawyers, licensing, export restrictions)
- Release Management

## Phases:

- Feasibility and rough estimates
- Development: Sprints (2 weeks), Milestones (2-4 with 6-8 sprints each), Beta releases
- End Game: Regression Testing, Translation, Documentation, Legal Clearance, COO (Certificate Of Originality)



 Drive Results.

Can large corporations or development organizations be agile?

Does SCRUM work for product development?

# AGILE

Do you need any special tools?

RTC?



**IBM**

**RUC2014**  
Rational User Conference



# Agile – Sprint Planning Perspective

- Context
  - Product backlog produced by Product Management
  - Project (release) backlog produced by Feature Owner
  - Fairly standard Scrum sprint planning (planning poker)
  - Non-Standard Scrum team (QE, ID, UX, i18N, DBA)
  - ScrumBan workflow (3LS interruptions)
  
- How RTC Helps
  - Backlog per product, project, milestone, sprint and team
  - Plan and load per release, milestone, sprint and team
  - Prioritized features, stories and defects
  - Work breakdown on features, sprints, teams
  - Kanban view (no experience yet)

**FCM: Granite Release Plan**  
 Owner: Granite (FCM) | Iteration: FCM 1.0 | **76 Closed** | **52 Open**

**FCM 1.0 Sprint 12**  
 Closed items: 5 | Open items: 11

**91/157pts | 1/1hrs** **76%**

**12/22pts** **Estimated: 73%**

**RUC2014** **IBM**



# Agile – Daily Perspective

- Context
  - Need prioritized list of stuff to do
    - Sometimes blocked or distracted
    - Sometimes waiting for someone/something
  - Need up-to-date status of defects/stories
  - Not everyone in office
  - ScrumMaster works in team (no separate "manager")
- How RTC helps
  - "My Work" with current work
  - Future (unplanned) work with priorities
  - Change status and reassign from within Eclipse
  - Supports remote teams and home office workers (indispensable)
  - Assign stories and defects to team, not to person



The screenshot shows the Eclipse IDE interface for a project named "FCM: Granite Release Plan - FCM 1.0". The main workspace displays a tree view of the project structure under "Financial Close Management".

- Inbox (Financial Close Management)**: Contains a sub-section for **Current Work (Financial Close Management)** with a badge indicating **(5 unread items)**. Below this, a summary box shows **Load: 0 / 34 | +34 h** and **Today (7 items)**. A red circle highlights the **Load: 0 / 34 | +34 h** text.
- Future Work (Financial Close Management)**: Contains a sub-section for **Future Work** with a badge indicating **(6 unread items)**. A red circle highlights the **Future Work** text.

Below the tree view, a list of tasks is visible, including:

- Forms that use a monthly actuality should not be possible to open for a weekly actuality. The
- Verify incoming requests according to security requirements
- 'From others' sometimes computed differently

At the bottom of the IDE, a search bar shows "Financial Close Man..." and a status bar indicates "8186: Structures are not being u ..." and "223 Hits".



# Agile – Monitoring Perspective

- Context
  - Multiple simultaneous projects
  - Report progress to Executive Management
  - Identify disruptions, handle resource conflicts and shortage
  - Ensure quality targets are met
  - Plan and synchronize teams (UX, doc, i18N, legal)
- How RTC Helps
  - All information stored in one place and accessible by everyone.
  - Workload on teams and team members
  - Status per epic, feature, folder, story and defect
  - Progress and projection per epic, feature and team
  - Tracking by story points, velocity and burn down graphs
  - Everyone has the same view of the truth
  - Force common “Definition of Done”

The screenshot displays the IBM Project Center interface for the "FCM: Granite Release Plan". The main header shows the project name and owner: "FCM: Granite Release Plan" and "Owner: Granite (FCM) | Iteration: FCM 1.0 | 76 Closed | 52 Open".

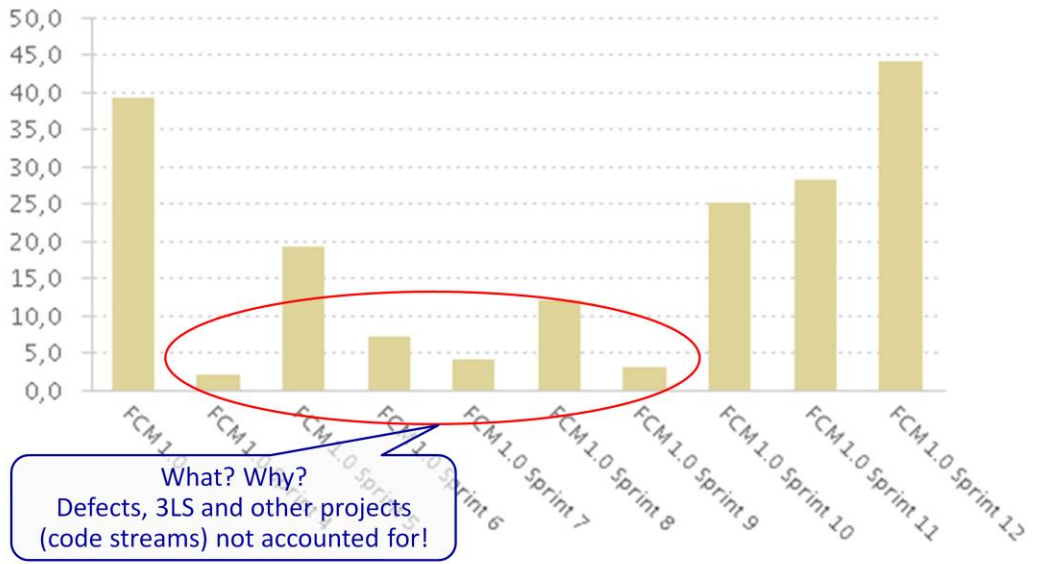
The interface is divided into several sections:

- Task List:** A central table listing tasks with columns for priority, name, status, and progress. For example, "FCM 1.0 Sprint 12" is highlighted with a red circle, showing 12/22pts completed and 73% estimated.
- Summary Cards:** Overlaid cards provide quick snapshots of task counts. One card for "FCM 1.0 Sprint 12" shows "Closed items: 5 | Open items: 11".
- Progress Indicators:** Horizontal bars at the top and right of the task list show completion percentages, such as 76% for the overall plan and 73% for the selected sprint.
- Navigation and Tools:** The top navigation bar includes options like "Work Breakdown", "Edit | Copy", and "Exclude".

Red circles in the image highlight the project name, a task's progress bar, and a milestone's progress bar.

Same screen used for planning and monitoring. Good!





FCM: Granite Release Plan [FCM 1.0] | FCM 1.0 Release Plan [FCM 1.0]

Owner: Financial Close Management | Iteration: FCM 1.0 | 194 Closed | 229 Open

222/413pts | 1/1hrs | 64%

### Burndown

Hours

2014-03-30 | 2014-06-30 | 2014-09-30

- Remaining Work
- Planned Work
- Ideal

Notes | Planned Items | Charts | Links

Search for ID or Text | Financial Close Man... | 194 Hits

**Progress Report**

Story Points Done:	222 of 413 (54%)
Hours Done:	1 of 1 (100%)
Items with Story Points:	30%
Items with 'Hours':	98%
Items Completed:	194 of 423 (46%)

What? Why?  
1 hour estimated, that's what we see ☹️  
We want to track story points!

RUC2014 IBM



 Drive Results.

# SCM

Software Configuration Management  
Source Code Control  
Revision Control  
Version Control  
RTC?



**IBM**

**RUC2014**  
Rational User Conference



# SCM – developer perspective

- Context
  - Several branches (main, FPx, IFy, production-x)
  - Lots of code, lots of people, lots of changes
  - Backtrack changes to find cause of defect (some by [gustav.vasa@mora.svearike](mailto:gustav.vasa@mora.svearike))
  - Sometimes collaborate on larger change
  - Current work often interrupted
- How RTC helps
  - Merge changes between branches
  - Link change sets with defects and stories
  - Restore code to previous baseline
  - Share changes with team member (before delivery)
  - Review changes (before delivery)
  - Suspend/resume change sets



# SCM - development lead perspective

- Context
  - Separate code streams (main, fixpack, prototyping ...)
  - Testing on stable code
  - Backtrack root cause of defect
  - Monitor changes in code (per defect, feature, team, developer)
- How RTC Helps
  - Create multiple branches
  - Set baselines
  - Link changes to defects or features
  - Provide metrics for analysis



 Drive Results.

Are we happy, agile and productive?  
Because or despite RTC?  
Could we do better?

# FINDINGS



**IBM**

**RUC2014**  
Rational User Conference



## The GOOD

- Well integrated
  - Tight integration with Eclipse (plugin)
  - Defects and stories in one system (RTC)
  - Assign stories/defects to category, not person
- SCM is fast and flexible
  - suspend/resume change set
  - fast (no server roundtrip for checkout)
  - link stories and defects with change sets
  - decent diff/merge support (in Eclipse)
- Agile support for developer's daily chores
  - "My Work" showing current and upcoming sprints and milestones
  - Tracking of change sets in stories and defects
  - Ability to share links (email, chat, etc)





## The BAD

- Cannot use whiteboard
- Speed (RTC server in Canada)
- SCM "Time machine"
  - restore to baseline, not arbitrary point in time
- SCM history lost in migration from v3 to v4 (probably our fault)
- Agile sprint planning
  - Views, layouts and filtering too rigid (CQ is more flexible)
- Scrum artifacts and estimates
  - Not an ideal fit for ScrumBan
  - "Hours" obscure vision (for us)
  - Hard to aggregate status/hours/story points to folders/features





## The UGLY (wishlist)

- Integration with whiteboard (how?)
- Better support for Lean/ScrumBan (how?)
- Better support for sprint planning
  - Fully configurable views (in many cases, Excel does a better job)
- Custom artifacts and estimates
  - configure what to use
  - custom aggregation of status/hours/story points to folders/features
- Better "time machine" to diff/restore code (arbitrary point in time)
- Collaborate on change w/o "Deliver" or "Set Complete"
- Local server with replication to Master
- Utilize RTC build engine, integrate with Jenkins and uDeploy
- Better automation and tracking of build ids, branches and baselines





 Drive Results.



# SUMMARY



**IBM**

**RUC2014**  
Rational User Conference



## Does RTC help or hinder?

- SCM works well
- Developers and Managers well supported
- Planning is a challenge
- Explore
  - Lean and Kanban, not just SCRUM
  - DevOps, uDeploy, build engine



# THANK YOU!

People who say it can't be done should  
not interrupt those who are doing it.



# References

- Me and my World
  - [ibm.com/software/products/category/business-analytics](https://ibm.com/software/products/category/business-analytics)
  - [ibm.com/software/products/category/performance-management](https://ibm.com/software/products/category/performance-management)
  - [ibm.com/software/products/cognos-controller](https://ibm.com/software/products/cognos-controller)
  - [peer.torngren@se.ibm.com](mailto:peer.torngren@se.ibm.com)
  - [linkedin.com/in/peertorngren](https://linkedin.com/in/peertorngren)
- Other
  - CM Crossroads: [cmcrossroads.com](https://cmcrossroads.com)
  - Agile: [agilemanifesto.org](https://agilemanifesto.org)
  - Lean: [linkedin.com/groups?qid=78197](https://linkedin.com/groups?qid=78197), [ibm.com/developerworks/community/blogs/ambler/entry/principles\\_lean\\_software\\_development](https://ibm.com/developerworks/community/blogs/ambler/entry/principles_lean_software_development),
  - SCRUM: [scrumalliance.org](https://scrumalliance.org), [linkedin.com/groups?qid=52030](https://linkedin.com/groups?qid=52030)
  - Kanban: [everydaykanban.com/what-is-kanban](https://everydaykanban.com/what-is-kanban), [linkedin.com/groups/IT-Kanban-2908737](https://linkedin.com/groups/IT-Kanban-2908737)
  - Disciplined Agile Delivery: [disciplinedagiledelivery.com](https://disciplinedagiledelivery.com), [linkedin.com/groups/Disciplined-Agile-Delivery-4685263](https://linkedin.com/groups/Disciplined-Agile-Delivery-4685263)
  - Scaled Agile Framework: [scaledagileframework.com](https://scaledagileframework.com), [linkedin.com/groups/Scaled-Agile-Framework-4189072](https://linkedin.com/groups/Scaled-Agile-Framework-4189072)
  - ScrumBan: [leansoftwareengineering.com/ksse/scrum-ban](https://leansoftwareengineering.com/ksse/scrum-ban), [linkedin.com/groups/Lean-Scrumban-8138149](https://linkedin.com/groups/Lean-Scrumban-8138149)
  - ScrumBut: [scrum.org/ScrumBut](https://scrum.org/ScrumBut), [linkedin.com/pulse/article/20140910082716-14110601-the-scrumbut-challenge](https://linkedin.com/pulse/article/20140910082716-14110601-the-scrumbut-challenge)
  - Software Craftsmanship: [manifesto.softwarecraftsmanship.org](https://manifesto.softwarecraftsmanship.org), [linkedin.com/groups?qid=1845268](https://linkedin.com/groups?qid=1845268)
  - The Dark Side of Agile: [darkagilemanifesto.org](https://darkagilemanifesto.org), [linkedin.com/groups/Dark-Side-Agile-4688911](https://linkedin.com/groups/Dark-Side-Agile-4688911)
  - RTC (Rational Team Concert): [ibm.com/software/products/rtc](https://ibm.com/software/products/rtc)
  - Boken om IT-arkitektur: [thearchitctbook.com](https://thearchitctbook.com)





# QUESTIONS?

([peer.torngren@se.ibm.com](mailto:peer.torngren@se.ibm.com))