

Planning is Agile,
Planning is Traditional,
Planning is Hybrid,
Planning is Planning

Sandeep Somavarapu
RTC Planning & Tracking
sandeep.s@in.ibm.com

IBM Software

Innovate2011

The Premier Event for Software and Systems Innovation



Software. Everywhere.

August 9-11, Bangalore | August 11, Delhi



Please Note:

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Outline

- Basics: RTC Planning
- Agile, Traditional, Hybrid
- Compare, Mix & Match

Basics: RTC Planning

Team Concert's Planning Support

- Team Concert's Planning support has the following key characteristics:
- Process neutral (works with Scrum, OpenUp, Formal Project Management, ...), but assumes
 - ▶ supports two level planning: plan items and execution items
 - ▶ supports monitoring releases, phases, iterations/milestones
- No separation between planning/implementing of features and bug fixing
 - ▶ Both planning and defect management share a common data model
 - ▶ They are supported in the same tool and are highly integrated
 - ▶ Plans are in fact a query for work items
- Supports:
 - ▶ Top down planning (project manager, product owner, team leads, ...) **AND**
 - ▶ Bottom Up (team members) **AND**
 - ▶ Plan consolidation (Sprint planning meetings, stand-ups, ...)

Lingo: Project Areas, Timelines, Iterations, and Categories

JUnit Project
A Team Concert example project.
This example project area illustrates the use of work items, source control, builds, and change management.

Defined Timelines

- development [Project Timeline]
 - 4.4 [4/9/2011 - 8/7/2011]
 - 4.4 s1 [4/9/2011 - 5/9/2011]
 - 4.4 s2 [5/9/2011 - 6/7/2011]
 - 4.4 s3
 - 4.4 s4
 - Backlog

Categories

Categories	Associated Project/Team Area
Unassigned <Root Category>	JUnit Project [Project Area]
JUnit	JUnit Project [Project Area] [inherited]
Doc	JUnit Project [Project Area] [inherited]
Framework	JUnit Project [Project Area] [inherited]
IDE Integration	JUnit Project [Project Area] [inherited]
Eclipse	IDE Integration [inherited]
IDEA	IDE Integration [inherited]
Tests	JUnit Project [Project Area] [inherited]

Defect 92
Summary: * timeout doesn't work properly for >=2 cases in junit4.3?
Type: Defect
Filed Against: * JUnitFramework
Severity: Normal
Owned By: Bill Cassavelli
Priority: Unassigned
Planned For: 4.4 s2
Estimate: 2 h
Time Remaining: Corrective
Due Date:

Annotations:
- "defines Timelines & Iterations" points to the Defined Timelines window.
- "defines Categories" points to the Categories table.
- "Project Setup Work Items" points to the Defect 92 details.
- "Project Setup Work Items" also points to the JUnit Project area in the Categories table.

Plans: Creating & Populating

- Plans are owned by a Team or Project and are associated with an Iteration
- Plan content is derived from this:
 - ▶ All work items which have a Filed Against value set to a category associated with the plan's owner and which are planned for the plan's iteration.
 - ▶ If the plan's project area / team area has child-team areas work items of the child-team areas are included as well.
 - ▶ If the iteration has child-iterations work items of the child-iterations are included as well.

Planning 3.0.1 Release ?

632 items: 3 open, 629 closed | Ends in: 16 days

▼ Plan Details

Owner: Planning

Iteration: 3.0.1 [11/15/10 - 6/3/11]

Plan Type: Release

Include All Items

Burndown

Velocity
Progress
Load

535.0

0.0

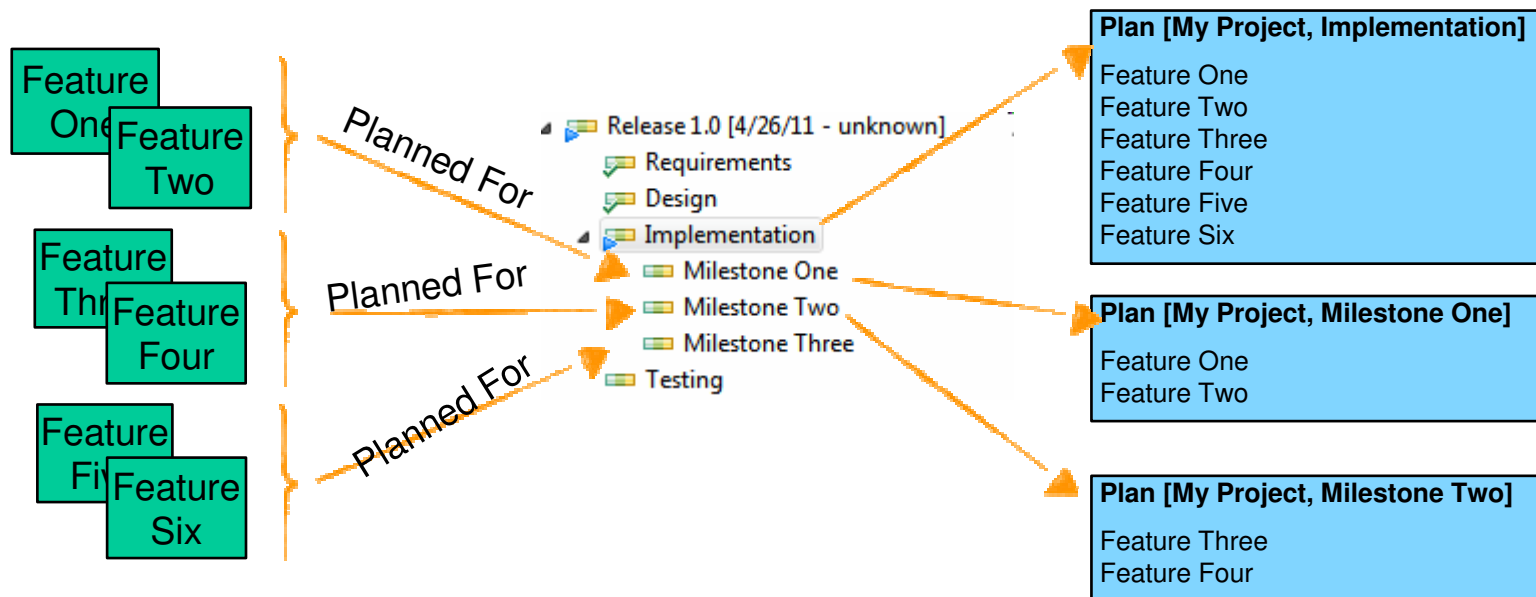
3.0.1 3.0.1 M13 3.0.1 RC1 3.0.1 RC3 Now

Owner & Iteration

Plan Slicing

- Plans include items from child-team areas and child-iterations. This allows to slice larger plans into smaller ones.

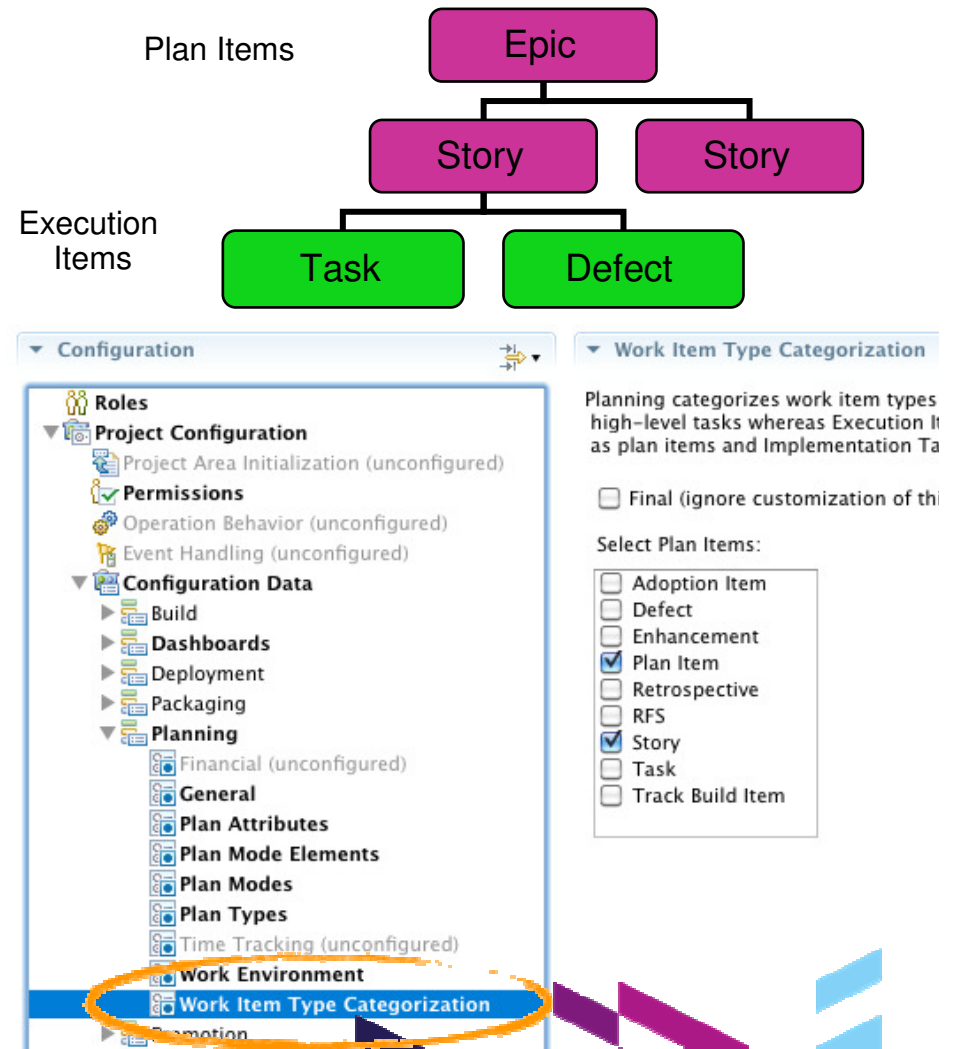
My Project – Work Items



Plan and Execution items

- Work items have two flavors:
 - ▶ Execution Items - Work items which have work assigned. In Scrum these are Task and Defect
 - ▶ Plan Items - Work items that are relevant for the planning. In Scrum these are Epics and Stories

- RTC allows to customize which work item types are plan items and which types are execution items.



Agile Estimation and Sizing

- Planning separates **estimating the size** from **estimating the effort**
- The **size** is often estimated in relative values. Scrum uses **Story Points**
- The **effort** is typically estimated in **Ideal Hours/Days**
- **Velocity** measures a teams rate of progress
 - ▶ Completing 2 Stories of 5 story points in one iteration gives a velocity of 10
 - ▶ Good guess is to assume the same velocity for the next iteration
 - ▶ team velocity is what matters
- It is not so different for traditional planning. They usually size in man days (ideal effort). RTC Planning component can use man days for sizing as well.

One Plan, Many Views

- Plans offer multiple views depending on what you're doing
 - ▶ Ranked List aka Backlog
 - ▶ Work Break Down Structure (WBS)
 - ▶ Roadmap / Gantt-Chart
 - ▶ Schedule Variance
 - ▶ Task Board

Characterize: Traditional & Agile

Traditional: PMBOK Guide

- Traditional project management is based on the principals described in the PMBOK Guide⁽¹⁾

Five Process Groups:

- Initiating
- Planning*
- Executing*
- Monitoring and Controlling*
- Closing

Nine Knowledge Areas:

- Integration Management
- Scope Management*
- Time Management*
- Cost Management
- Quality Management*
- Human Resource Management
- Communications Management
- Risk Management*
- Procurement Management

(1) A Guide to the Project Management Body of Knowledge (PMBOK Guide) - Project Management Institute

Agile: The Manifesto

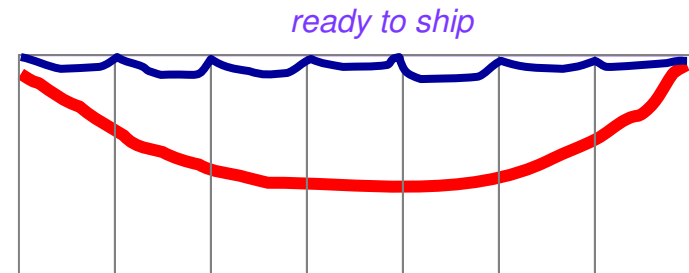
Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

(see <http://agilemanifesto.org/>)

Short iterations – the main driver

- Allows feedback / quality checks any n weeks
- Each milestone is a miniature development cycle
 - plan, execute, test, ship, retrospective
- The iteration outcome (library, product, ...) must be shippable / consumed
 - Other teams, betas, demos, ...

∅ Short iterations reduce stress !!



Agile Methods

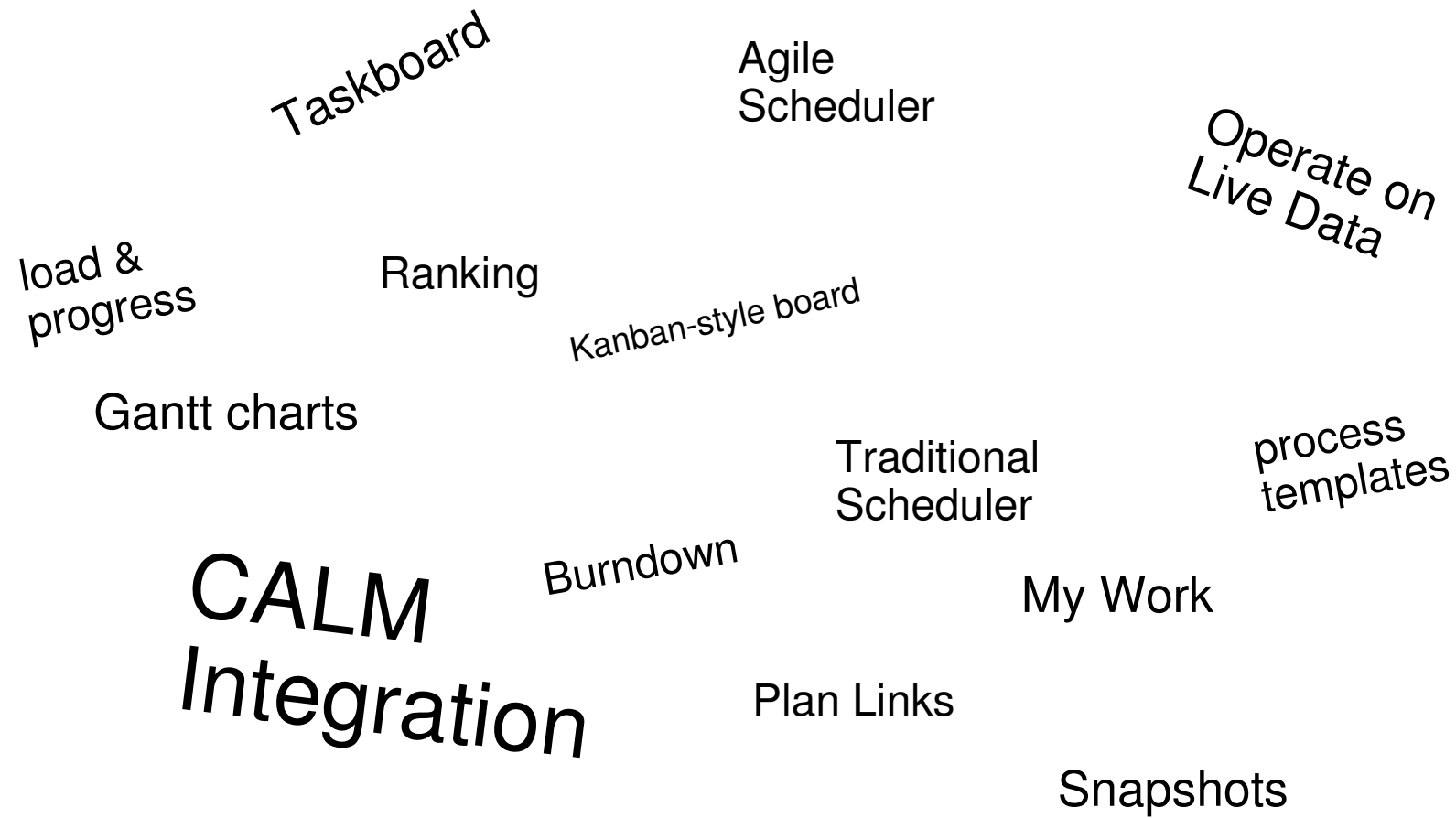
- There are several Agile Methods available, however from a bird's eye view they share a common set of tooling relevant properties:
 - Short development cycles (1 – 6 weeks)
 - Self organizing development teams
 - ▶ Joint planning meetings
 - ▶ Stand-up meetings
 - High personal responsibility of team members
 - ▶ Developers “manage” their work
 - ▶ Developers / Teams estimate
 - Progress tracking
 - Ongoing customer involvement
 - Retrospectives
 - Use of historical data to improve planning

Mix & Match - Planning

“We do Scrum, but...”

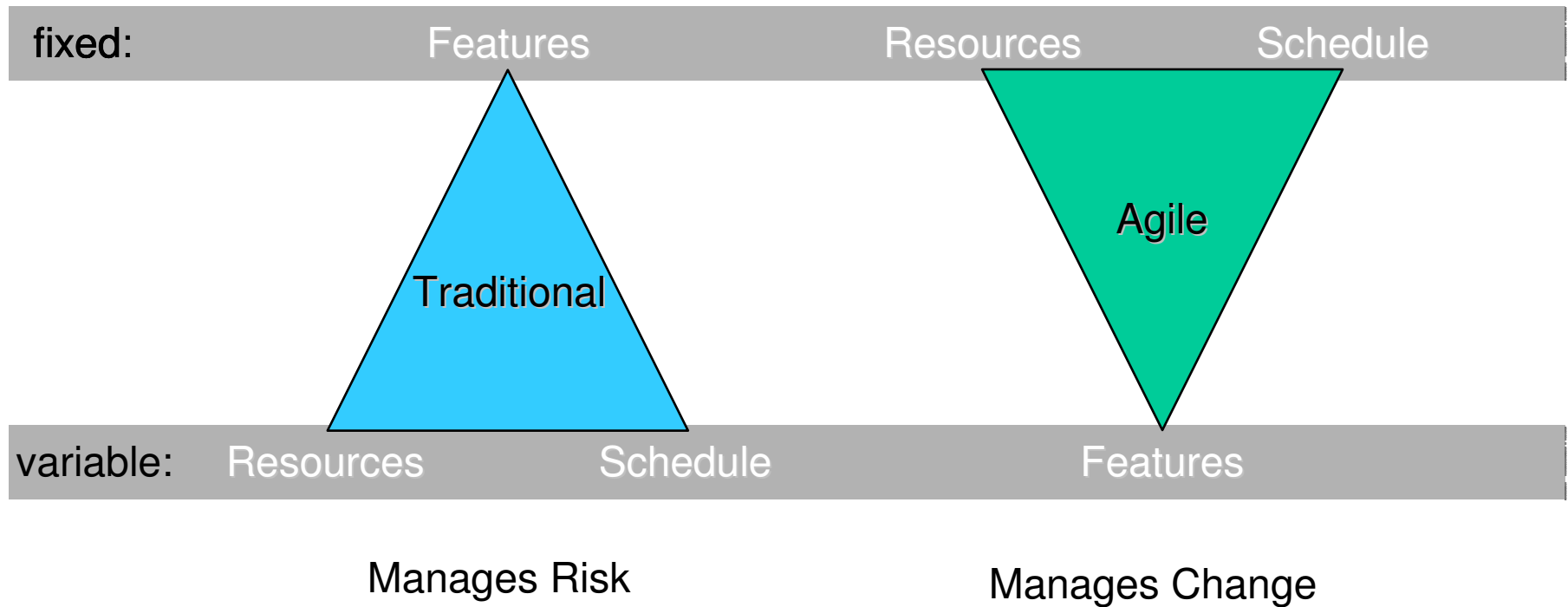
- Most of the RTC customers are neither 100% agile nor 100% traditional (even IBM isn't 😊)
- Goal is to support projects that mix agile and traditional concepts.
- The overall process is traditional the development execution happens agile.
 - ▶ Need for an overview Gantt – which feature is delivered in which iteration
 - ▶ Visualization of the dependencies between features
- The overall process is agile but the execution of a single iteration happens traditional
 - ▶ Need for a traditional scheduler (schedule dependencies & constraints)
 - ▶ Need for traditional resource management

RTC/Planning is a tool, not a methodology



Compare: Traditional & Agile

Comparison between Traditional & Agile

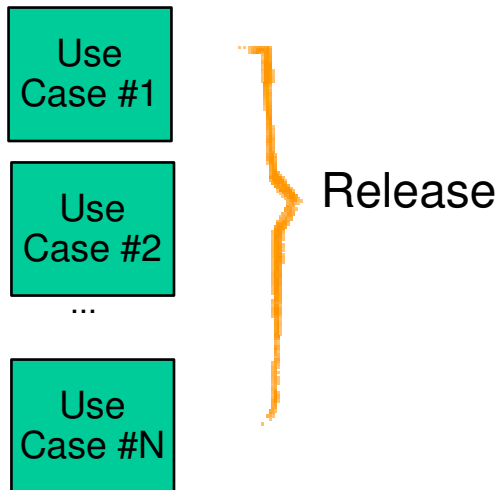


(1) The Software Project Manager's Bridge to Agility

Different Management of Features

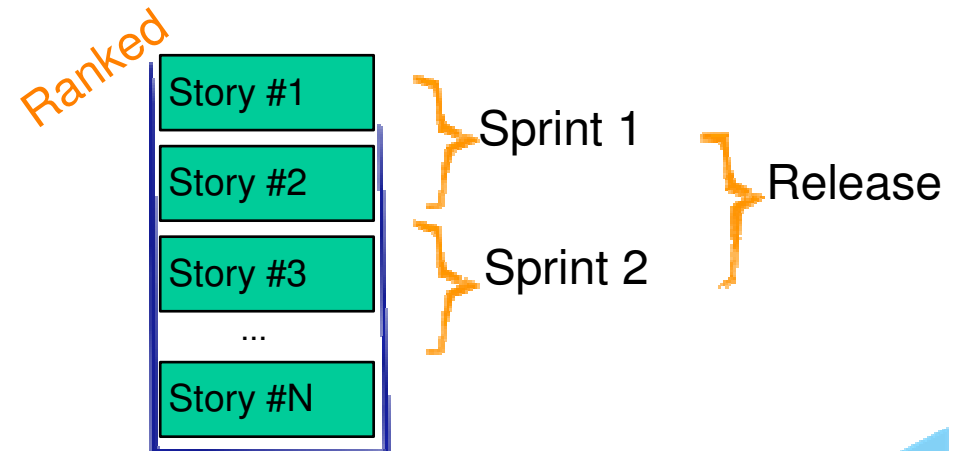
- Traditional

- ▶ Manage a single release plan
- ▶ Requirements (committed)
- ▶ Won't Change
- ▶ Sized in Hours, Man-Days



- Agile

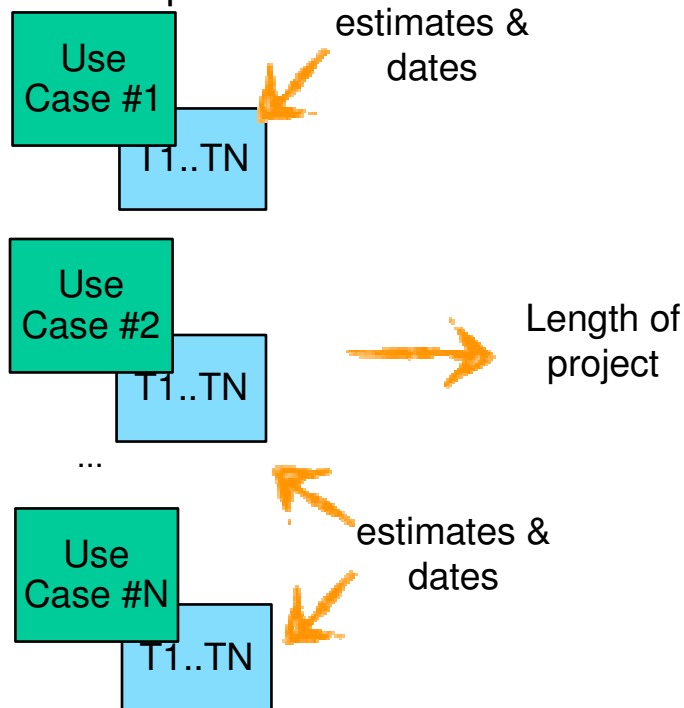
- Ranked Backlog (wish list, stack ranked)
- Tentative Release Backlog/View
- Sprint Backlog (commitment)
- Sprint is immutable, Backlog isn't
- Sized in Complexity Unit
- Two Level Planning



Differences in Scheduling (high level)

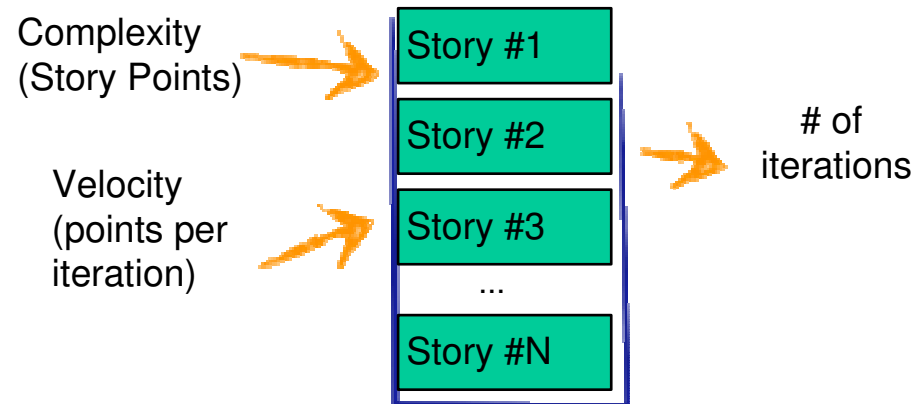
- Traditional

- ▶ Requirements & Use Cases
- ▶ Phases
- ▶ Estimated in Hours & Computed Dates



- Agile

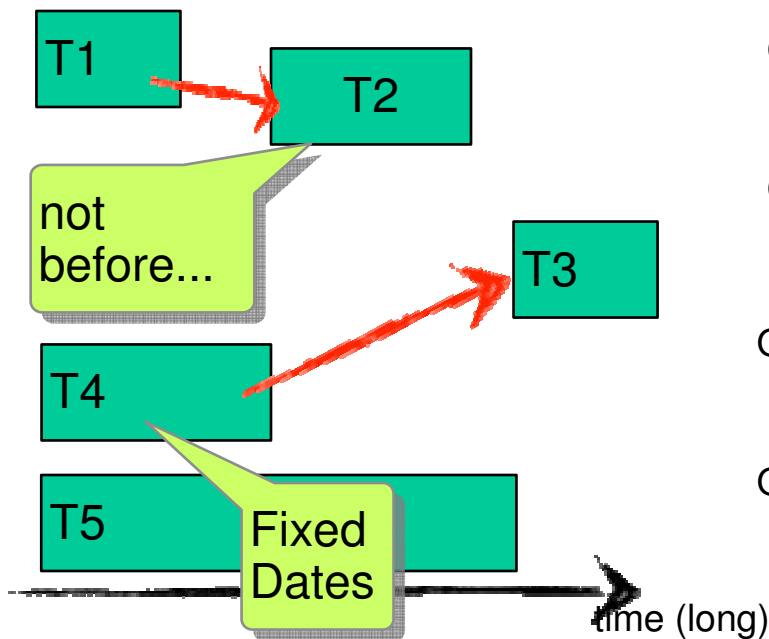
- Ranked Backlog
- Complexity & Velocity
- Release Iteration (optional, as view)
- Many Sprints



Differences in Scheduling (intra iteration)

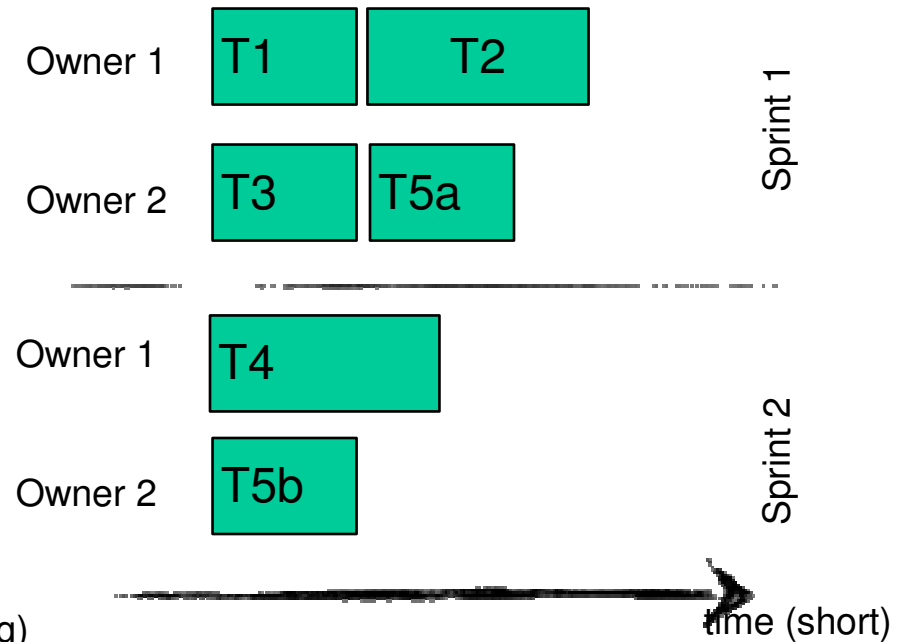
- Traditional

- ▶ Dates (start/end)
- ▶ Dependencies
- ▶ Long Phase
- ▶ Long Tasks



- Agile

- Sequences
- Dependencies (lightweight)
- Short Sprints
- Short Task (<1day)



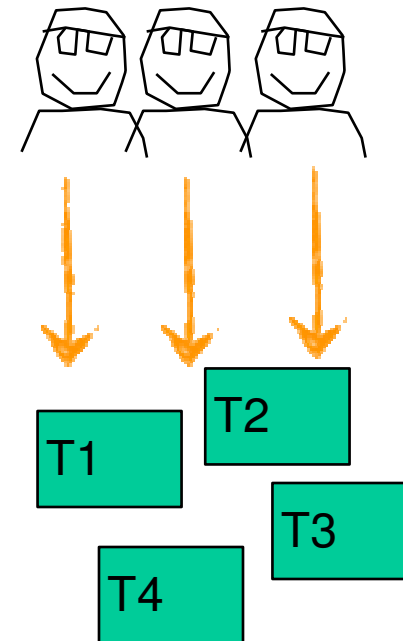
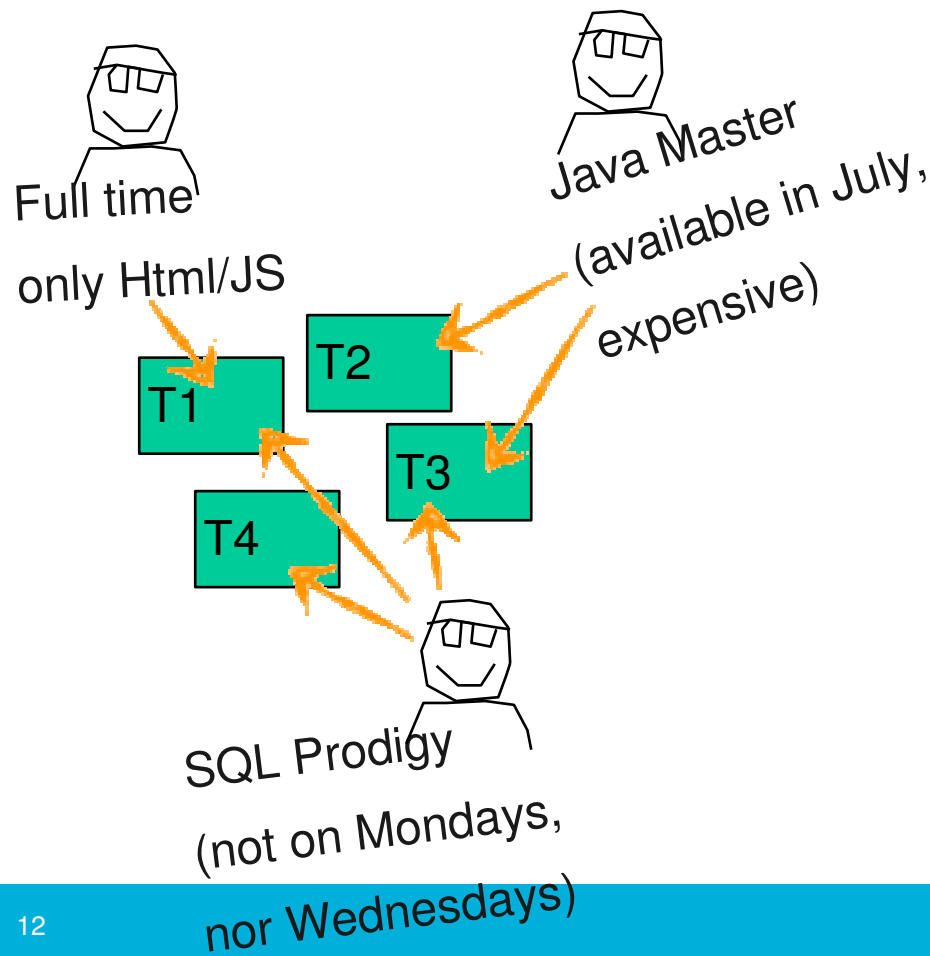
Different Handling of Resources

- Traditional

- Specialists & Contractors
- Time & Price constraints

- Agile

- Fixed team of all-rounders
- Load & Progress



Mix & Match: Tracking Progress

- Taskboard
 - Lets you know what's currently being worked on
- Progress Bars
- Burndown charts
- Progress on Items

The screenshot displays a project management interface with several key components:

- Summary Panel:** Shows 'Story Points: 13 pts' and 'Progress: 40 / 44 h' with a green progress bar. Below this, a table lists items:

Summary	Progress	Status	Estimated
Frequency of dividend transfer	0/13 pts 40/44 h	In Progress	100%
- Taskboard:** A grid view showing tasks in different stages:
 - Planned Items:** Donor Dividend Allocation Criteria, Frequency of dividend transfer.
 - In Progress:** Implement - Frequency of dividend transfer, Implement - Organization must identify how, Implement - Requests sent in form of email.
 - Completed:** Implem Donor Allocat.
- Progress and Burndown Metrics:**
 - Progress:** Progress: 5/36 pts, Estimated: 100%
 - Load:** Progress: 28/162 | -4 h, Estimated: 95%

Mix & Match: Track Change

- Plan Snapshots
 - Capture the state of a plan
 - Compare the plan state
- Planned Time
 - Planned Time Variance

Planned Items Links **Snapshots** ? Dashboard Notes

[Back to Snapshots](#)

	Comparison - May 10, 2011 4:28:27 PM	Current Plan - May
Overview		
Iteration Start Date	0	Feb 7, 2011
Iteration End Date	0	Feb 25, 2011
Scheduled Start Date	0	Feb 3, 2011
Scheduled End Date	6 weeks 4 days 23 hours	Feb 25, 2011
Work Items Total	27	90
New	-48	0
In Progress	-4	0
Closed	79	90
Not Accessible	--	--

Planned Items Links **Snapshots** ? Dashboard Notes

To perform a comparison, select two items from the list

	Name	Type	Modified Date	Created By
<input checked="" type="checkbox"/>	Current Plan	--	--	--
<input checked="" type="checkbox"/>	Almost the beginning of the Sprint	Regular	Feb 8, 2011	Johannes Rieken

Sounds traditional but helps Agile projects to answer 'How stable is my sprint?'

Mix & Match: Highlights & Task-oriented work

- View Modes
 - ‘View As’ your current tasks demands
 - Taskboard, Ranked List, Gantt, .
- Colors
 - Expression-based coloring
 - Color risks

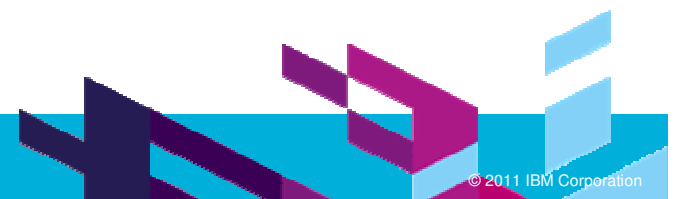
The screenshot displays a 'Planned Items' interface. At the top, there are tabs for 'Planned Items', 'Links', 'Snapshots', 'Dashboard', and 'Notes'. Below these, a 'View As' dropdown is set to 'Verification'. A search bar contains 'Type to Filter' and indicates '(2 items filtered)'. The interface includes sections for 'Options' and 'Columns'. A 'Filter' dropdown is set to 'Expression', and the filter value is 'Istate:Resolved'. A 'Colors' dropdown is set to 'verify me', and the color expression is 'severity>Minor'. Below these settings is a table with columns 'Actions', 'Summary', and 'Owned By'. The table contains three rows of tasks, each with a 'verify me' icon and a summary description.

Actions	Summary	Owned By
verify me	Summary field can't be edited in IE8	Sharoon She
verify me	Stop script warning on adding huge number of work items to the-	Sharoon She
verify me	fix formatting around copyright	Sonia Dimitro

QUESTIONS



www.ibm/software/rational





www.ibm/software/rational

© Copyright IBM Corporation 2011. 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.