



Agile – Scrum Introduction



Scrum in 100 words

- Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.
- It allows us to rapidly and repeatedly inspect actual working software (every two weeks to one month).
- The business sets the priorities. Teams self-organize to determine the best way to deliver the highest priority features.
- Every two weeks to a month anyone can see real working software and decide to release it as is or continue to enhance it for another sprint.



The Agile: Scrum Framework at a glance

Inputs from Executives,
Team, Stakeholders,
Customers, Users



Every
24 Hours

2 Week
Sprints



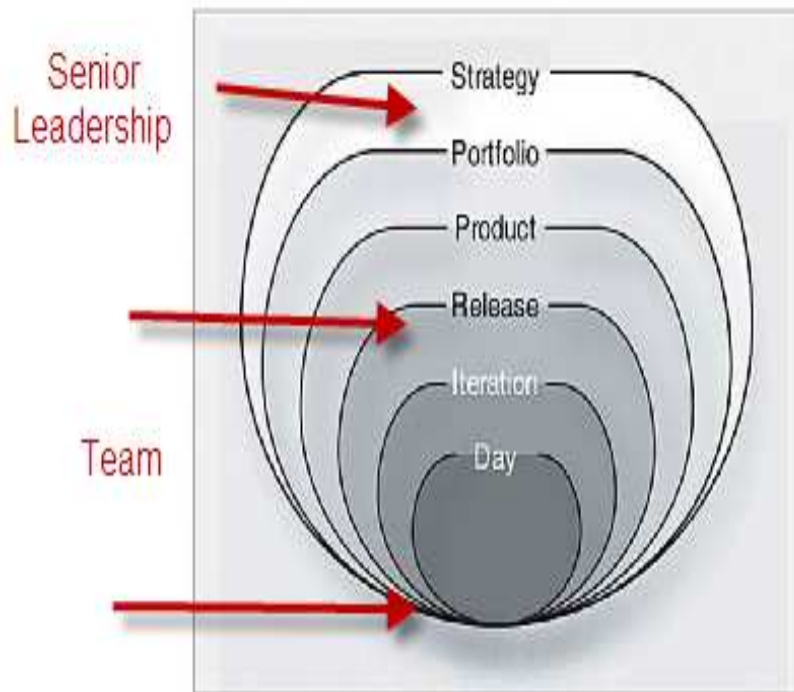
Sprint Backlog

Sprint end date and
team deliverable
do not change



Multiple Levels of Planning

PLANNING ONION



- Iteration planning considers the tasks that are needed to transform a Feature request into working, tested software
- Iteration Planning occurs at the start of each iteration

- Product, Portfolio and Strategic Planning are outside the scope of most agile teams

- Agile Teams achieve progressive elaboration of the plan at three distinct horizons, viz. Release, Iteration and Current Day.

- Release planning considers the user stories or themes that will be developed for a new release of a product or system
- Release Planning occurs at the start of the project
- It is updated throughout the project

- Product Planning involves a product owner looking ahead than the immediate release and planning for the evolution of released system.
- Portfolio planning involves the selection of the products that will best implement a vision established through an organization's strategic planning.

TYPICAL 2-WEEK SPRINT CALENDAR

	Monday	Tuesday	Wednesday	Thursday	Friday
Iteration 1	Sprint Planning Part 1 (Up to 2 Hours) Sprint Planning Part 2 (Up to 2 Hours)	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)
	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)	Sprint Review (up to 2 hours)
	Product Backlog Refinement (4 - 8 hours)				Sprint Retrospective (up to 1.5 hours)
Iteration 2	Sprint Planning Part 1 (Up to 2 Hours) Sprint Planning Part 2 (Up to 2 Hours)	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)
	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)	Daily Scrum (15m)	Sprint Review (up to 2 hours)
	Product Backlog Refinement (4 - 8 hours)				Sprint Retrospective (up to 1.5 hours)



Roles

- Product Owner
- ScrumMaster
- Team

Artifacts

- Product Backlog
- Sprint Backlog
- Task Board
- Burn-down Chart
- Working Software

Activities

- Release Planning
- Sprint Planning/Sprint
- Daily Stand-up
- Sprint Review/Demo
- Sprint Retrospective



Roles



Roles

Product Owner

Inputs from Executives,
Team, Stakeholders,
Customers, Users



Product Owner

- ❑ Defines and prioritize the product features
- ❑ Decides release date and content of the release
- ❑ Adjust features and priority every iteration, as needed
- ❑ Accept/Reject work products





- ❑ Responsible for enacting Scrum values and practices
- ❑ Represents management to the project
- ❑ Removes impediments
- ❑ Ensure that the team is fully functional and productive
- ❑ Shield the team from external interferences

Roles

Team

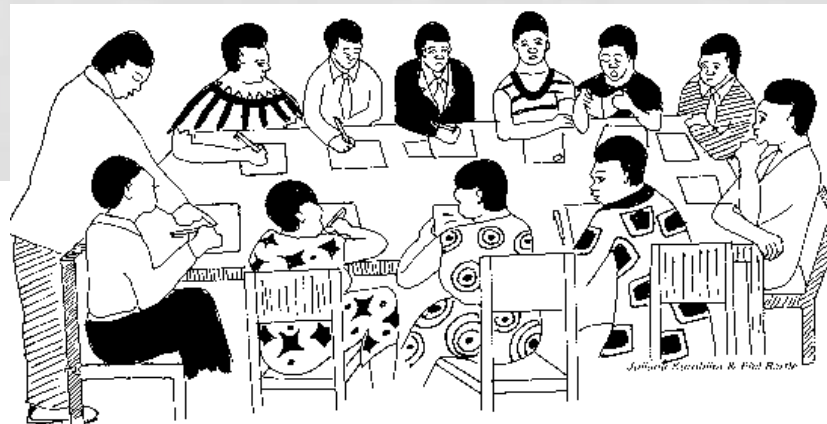


- ❑ Typically 5-9 individuals (but core team is full-time)
- ❑ Best if co-located in the same area
- ❑ Self-Organizing
- ❑ Responsible for development of the end product.
- ❑ Cross functional: analysts, designers/developers, QA, architects, etc.
- ❑ Team structure can change as the product is built out.



PARTICIPATORY DECISION MAKING

- Decider protocol
 - Concise and Actionable Proposal
 - Everyone Votes simultaneously, in one of 3 ways
 - **Thumbs Up** – Vote *thumbs up* if you like the proposal and will actively support it (i.e. be willing to champion the proposal).
 - **Thumbs Sideways** - Vote *thumbs sideways* if you will go along with the proposal as currently stated. The expectation is that if accepted, you will fully commit to this proposal.
 - **Thumbs Down** - Vote *thumbs down* if you can not support the proposal as currently stated. You must be willing to amend the proposal to make it acceptable. Those who voted Thumbs Down have a responsibility to amend the proposal as needed to reach consensus. They must make a new [counter-] proposal to the group that they would be willing to support. When a counter proposal is being made, everyone should listen to the counter proposal rather than trying to persuade the audience one way or another. The group votes on the new revised proposal.
 - Benefits:
 - Full participation
 - Quick, fast, simple
 - Scalable



Artifacts



Product Backlog

The Product Backlog is a prioritized features list, containing short descriptions of all functionality desired in the product.



Product Backlog

- ❑ Ranking is based on business value, technical value, risk management.
- ❑ Large items are decomposed.
- ❑ Backlog is reprioritized at the start of each Sprint
- ❑ Effort is measured in ‘Story Point’
- ❑ Managed by Product Owner

Sprint Backlog

The main purpose of the Sprint Backlog is to let the team to organize all the activities involved in the Sprint.



- List of tasks to be completed and assigned to individuals
- Updated daily by team members to reflect remaining work.
- Effort measured in 'Hours'
(compared to 'Story Points' in Product Backlog.)

Artifacts Task Board

Feature	Tasks		
	Waiting	In Progress	Done
A			6
B	1	3	1
C	5		
D	4		
E	5		

- ❑ Makes the Sprint Backlog visible
- ❑ What is:
 - Waiting (To Do)
 - In Progress
 - Done
- ❑ Used every day to track progress, assign work, run daily meetings and update tasks.



Burn Up/Down Chart

Burn Up/Down charts provide daily progress for a sprint over the sprint's duration.

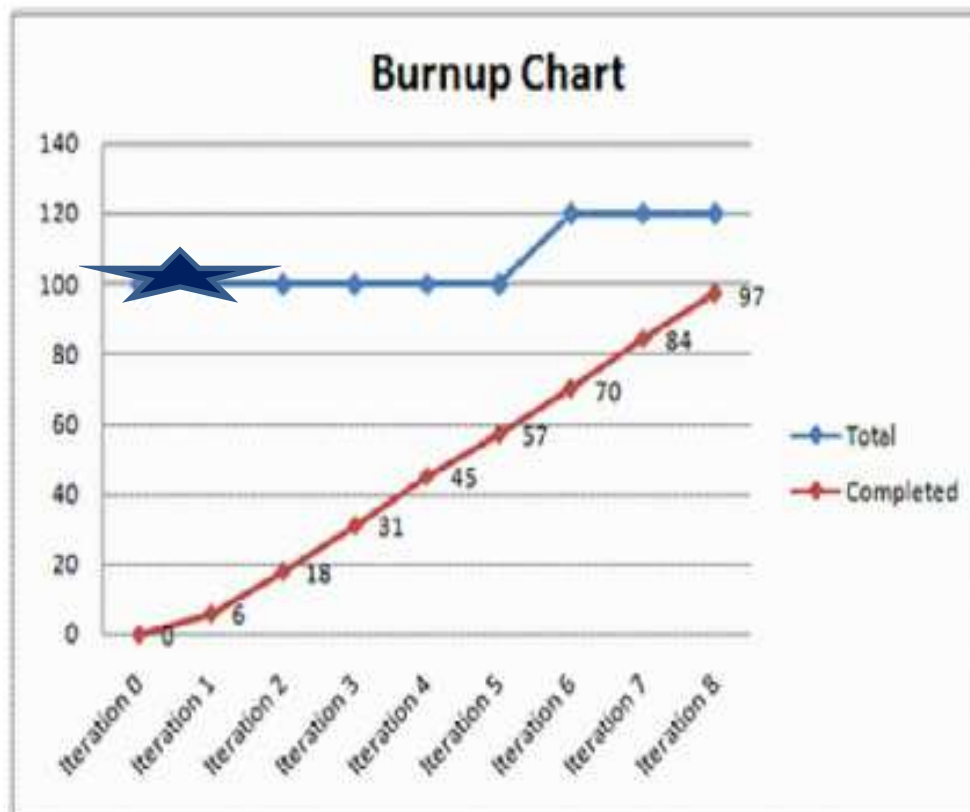


**Burndown/up
Charts**

- ❑ Burn Down Chart – Reflects remaining work in Sprint.
- ❑ Burn Up Chart – Reflects increment work accomplished in Sprint.

Burn Up Charts

The Burn Up chart, in addition to showing how much work is completed also shows the work in the project (read scope)



- The blue line shows that scope has increased around the 5th iteration

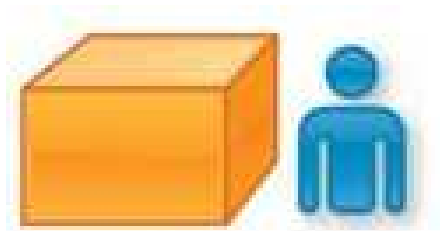
- Known as 'Feature Complete Graph' in Feature Driven Development (FDD)

Artifacts

Working Software

Close of each Sprint is potentially shippable product.

- ❑ Complete & Working
- ❑ Can be incremental a piece of the overall product.



Finished Work

Activities



Release Planning

- ❑ Provides a common vision about what needs to be achieved, and when.
- ❑ It guides the team to make decisions during detailed planning.
- ❑ It helps prioritize the stories.
- ❑ It resolves conflicts and guides the team toward the right balance on trade-offs.
- ❑ Uncovers an opportunity for enhanced functionality.

Example : User Story Card (Formal)

"As a <role>,
I want <goal/desire>
so that <benefit>"

As a
Who wants this piece of functionality

I want
What the user wants

So that
Why the user wants it

As an employee, I want to purchase a parking pass so that I can park my car safely in the basement and save money.

Priority : 8

Estimate : 4

Very clear acceptance criteria to validate the story



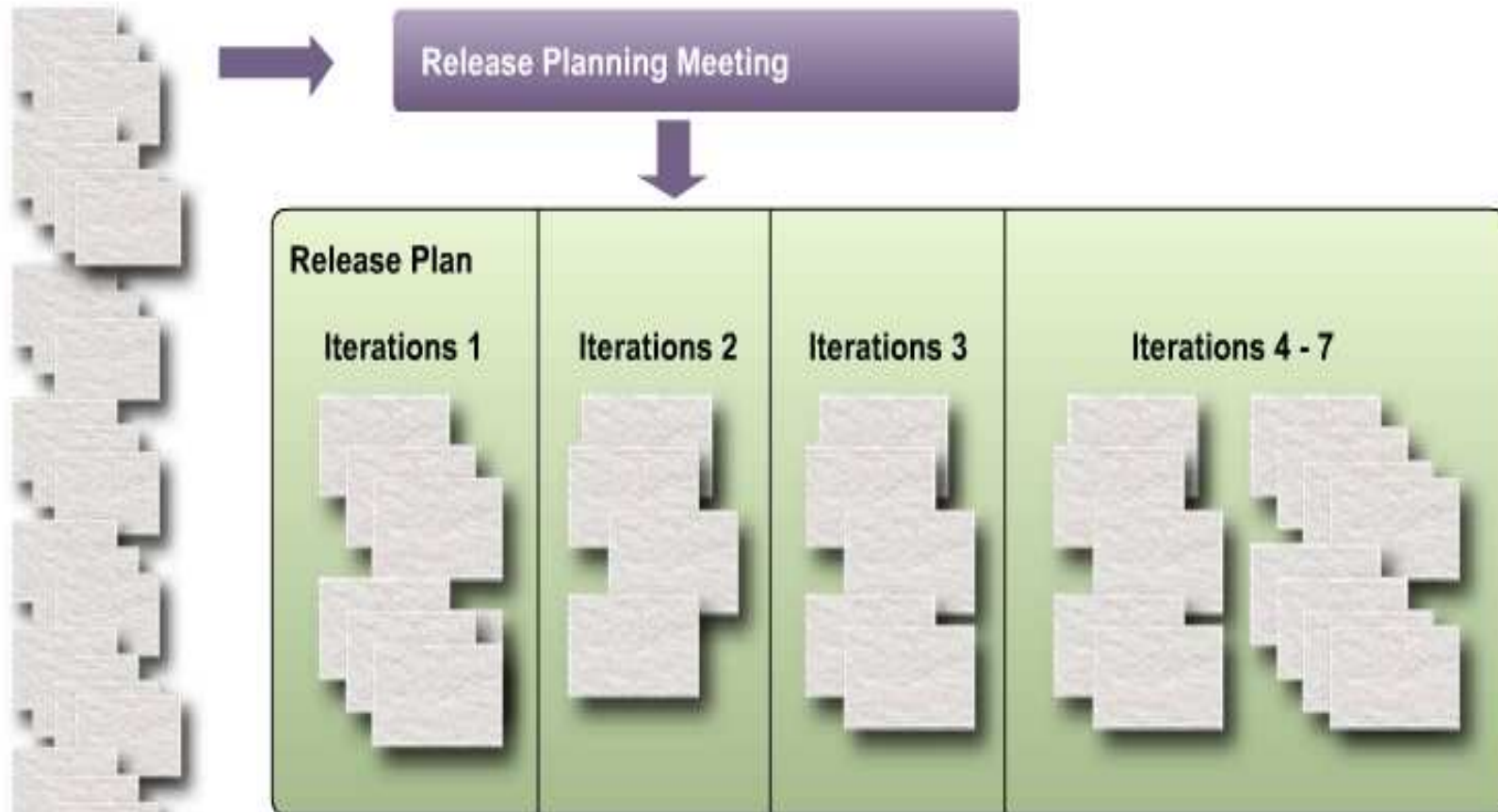
Planning poker—an example



Estimator	Round 1	Round 2
Erik	3	5
Martine	8	5
Lnga	2	5
Tor	5	8

Team can go for **5 or 8 story points**

Release planning



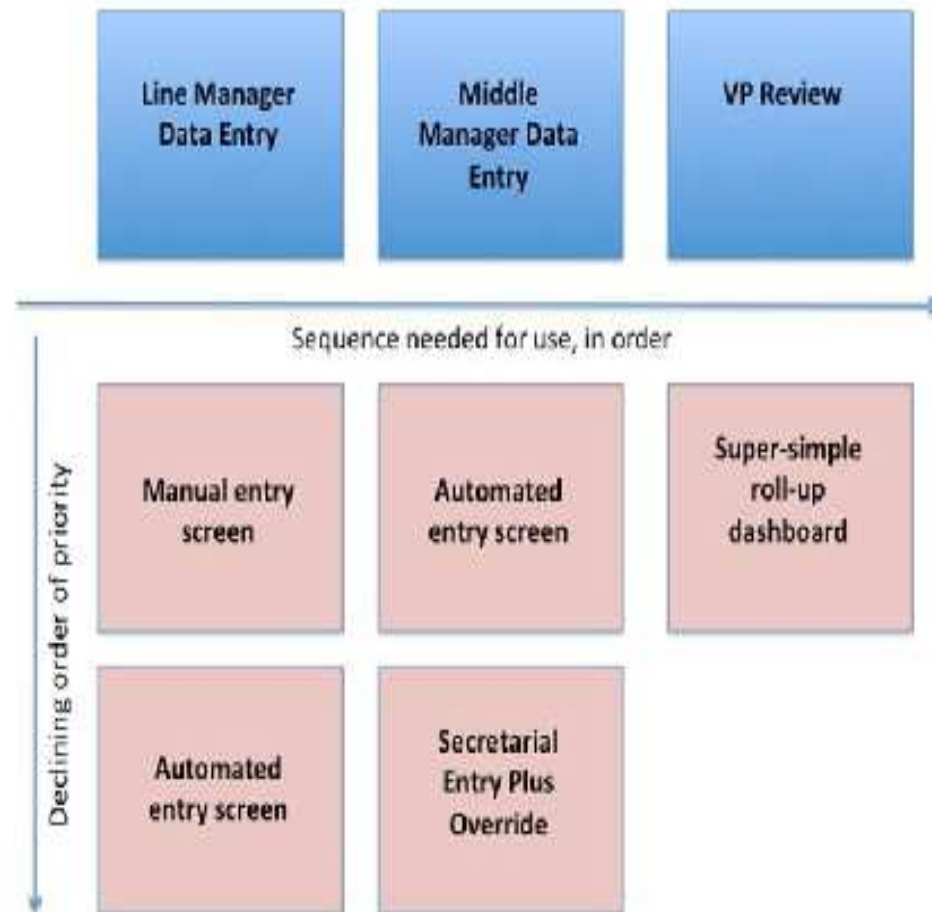
The purpose of Release Planning is to define the contents of a release or a specific shippable product increment.

Iteration 0 – ZERO story points
To setup Team, and work
environment

Iteration H – Hardening Iteration
before Product Launch, ZERO Story
Point

Agile story maps

A story map is an arrangement of the story cards which represents the sequence in which the stories will be needed by the business on the horizontal axis, and the priority of the stories on the vertical axis. The horizontal axis should represent the minimum set of features needed by the system to allow it to be released.



Sprint Planning

At the beginning of each Sprint, the Sprint Planning Meeting takes place. It is divided into **two** distinct parts.

Team selects starting at top as much as it can commit to deliver by end of Sprint

Sprint Planning Meeting

Part #1 - Focuses on understanding what the Product Owner wants.

Part #2 - Focuses on detailed task planning for how to implement the items that the team decides to take on.

As a vacation planner, I want to see photos of the hotels.

Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)

Sprint

A Sprint is the basic unit of development in Scrum that is **timeboxed**.



- ❑ The duration is **fixed in advance** for each sprint and is normally between one week and one month.
- ❑ During each sprint, the team creates finished portions of a product.
- ❑ The set of features that go into a sprint come from the product *backlog*, which is an ordered list of **requirements**.

Daily Standup

A Daily Standup meeting is a short (**15 minutes or less**) meeting that happens every workday at an appointed time and place.

Each member of the team reports 3 things to the other members of the team:



Q1 - What they were able to get done since the last meeting

Q2 - What they are planning to finish by the next meeting

Q3 - Any roadblocks or impediments that are in the way

Daily Scrum is not a status meeting to report to a manager

Sprint Review/Demo

Meeting where the Product Owner, Scrum Team and stakeholder review:

- ❑ What was done during the Sprint
- ❑ Discuss the deliverables
- ❑ Figure out what to do next.
 - ✓ Not a PowerPoint presentation
 - ✓ Demonstrates real working software.



Sprint Retrospective

The sprint retrospective is usually the last thing done in a sprint.

- ❑ Learn what works and what does not work and make adjustments for the next Sprint.



- ❑ Transform identified improvements to concrete action points

Team Velocity

Over a number of sprints your team will determine the number of story points the team can achieve in a Sprint – the “velocity”

- ❑ The number is derived by adding all the story points from the last sprint's stories/features.
- ❑ It enables a team to predict delivery times based on number of story points outstanding.



Team Effort Hours calculation

Team Member		% Avail	Hrs Off	Hours	
Jason Archer		25%		8.00	
Don Bass		38%		12.16	
Cheryl Lu		75%		24.00	
Harry Smith		75%		24.00	
Terry Pin		75%		24.00	
Jack Bean		50%	16	8.00	
Roy Jones		75%		24.00	
Person-Days Available:	15.5	Net Team Resources:	3.10	Team Hours:	124.16



Definition of Done

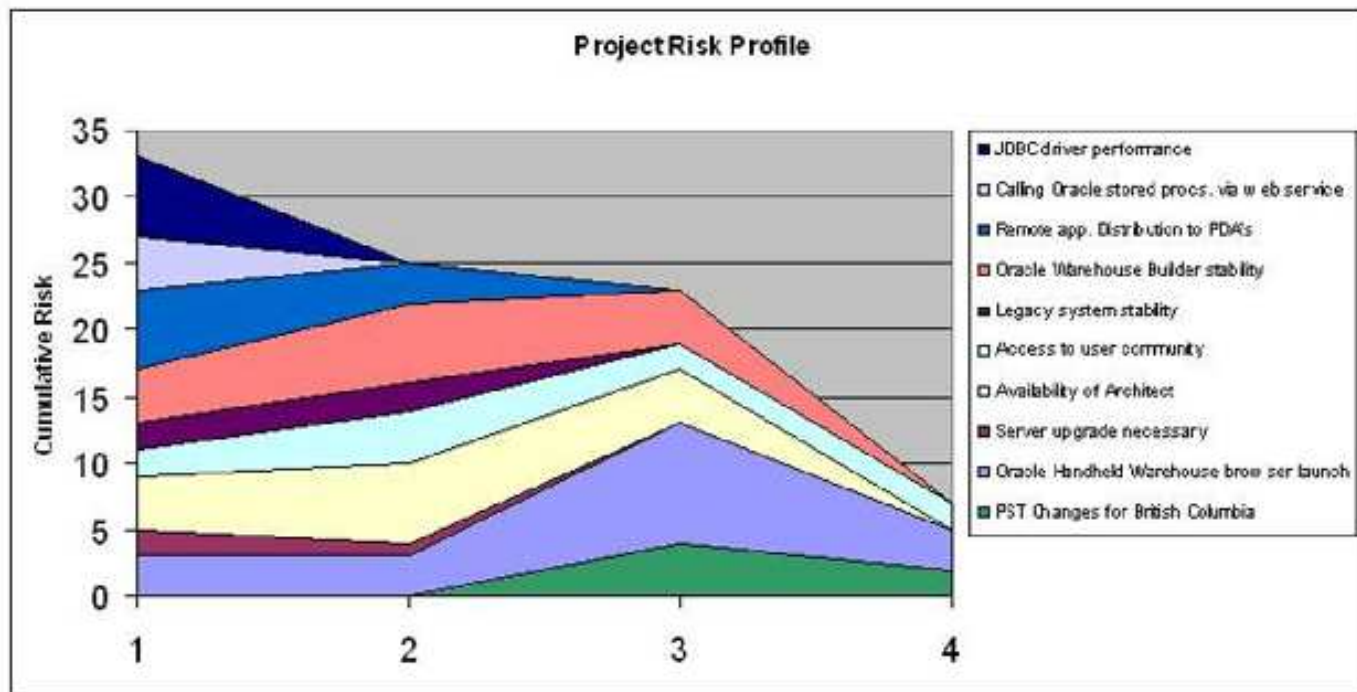
- ✓ Task (Code) completed
- ✓ Unit test completed
- ✓ Task (Code) reviewed
- ✓ Code I checked-in
- ✓ Necessary document is complete
- ✓ QA activity completed



Agile Risk Management



Risk Profile Graphs



Risk Profile Graphs quickly inform stakeholders if the risks are moving in the right direction (downwards) or if issues and concerns are escalating.



Scaling through Scrum of Scrums

