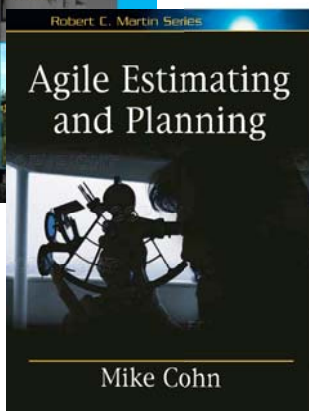# Selecting a Development Process:
Choosing Among the Leading Alternatives

Mike Cohn
Mountain Goat Software
mike@mountaingoatsoftware.com

1

Mike Cohn - background

The Addison-Wesley Signature Series

USER STORIES APPLIED
FOR AGILE SOFTWARE DEVELOPMENT

MIKE COHN
Foreword by Kent Beck

Robert C. Martin Series

Agile Estimating and Planning

Mike Cohn

Consultant, author, and speaker

O Founding member and director of Agile Alliance, Scrum Alliance, and Agile Project Leadership Network

O Founder of Mountain Goat Software

2

Today's agenda

- ☐ Considerations
- ☐ Team Software Process
- ☐ Scrum
- ☐ Extreme Programming
- ☐ OpenUP/Basic
- ☐ Rational Unified Process

Ceremony

- The amount of formalism in a process

  - Documentation, method weight, reviews

Few documents	Many documents
Few steps	Formal steps

Cycles

- Number and length of iterations

Sequential

Few documents
Few steps

Many documents
Formal steps

Many short iterations
(5 days)

Placing the processes

Sequential

Rational Unified Process

Few documents
Few steps

Team Software
OpenUP

Many documents
Formal steps

Scrum

Extreme
Programming

Many short iterations
(5 days)

Today's agenda

- ☑ Considerations
- ☐ Team Software Process
- ☐ Scrum
- ☐ Extreme Programming
- ☐ OpenUP/Basic
- ☐ Rational Unified Process

Team Software Process (TSP)

- Created by Watts Humphrey
  - Of Software Engineering Institute and Capability Maturity Model (CMM)
  - Builds on his Personal Software Process
- High discipline, highly defined
- A "cyclic development strategy"
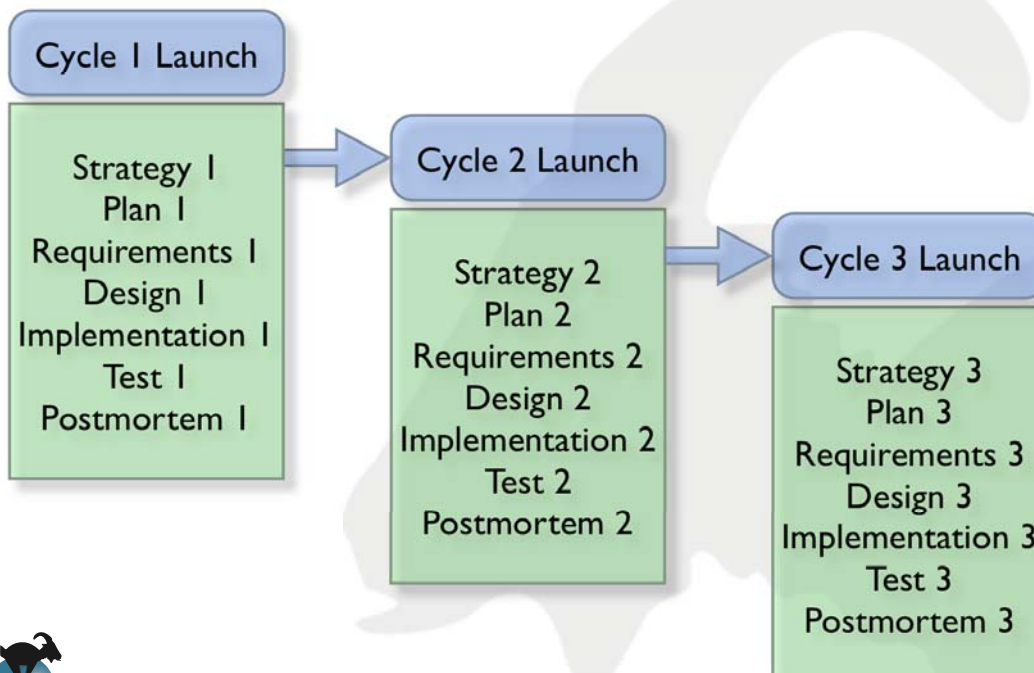 - Another way of saying "iterative and incremental"

Goals of the TSP

1. Build on the Personal Software Process
2. Develop products in cycles
3. Establish standard measures for quality and performance
4. Provide precise measures
5. Use role and team evaluations
6. Require process discipline
7. Provide guidance on teamwork problems

Team Software Process (TSP)

Cycle 1 Launch

Strategy 1
Plan 1
Requirements 1
Design 1
Implementation 1
Test 1
Postmortem 1

Cycle 2 Launch

Strategy 2
Plan 2
Requirements 2
Design 2
Implementation 2
Test 2
Postmortem 2

Cycle 3 Launch

Strategy 3
Plan 3
Requirements 3
Design 3
Implementation 3
Test 3
Postmortem 3

Cyclic development

- A *testable version* is produced at the end of each cycle

  - Purpose of testing is to assess the product, not fix it

  - Defects should be removed during the cycle

11

Designed to solve team problems

"The success or failure of a project is seldom due to technical issues. You almost never find yourself asking 'has the state of the art advanced far enough so that this program can be written?' Of course it has. If the project goes down the tubes, it will be non-technical, human interaction problems that do it in. The team will fail to bond, or the developers will fail to gain rapport with the users, or people will fight interminably over meaningless methodological issues."

Tom DeMarco

12

Teams, members, roles and goals

- A team has one or more members

- Each member fills one or more roles

Team leader	Quality/process manager
Development manager	Support manager
Planning manager	

- Goals are assigned throughout this hierarchy

  - Team, member, role

Goals and goals and...

Team goals

- Produce a quality product
- Run a productive and well-managed project
- Finish on time

Team member goals

- Be a cooperative and effective team member
- Do consistently disciplined personal work
- Plan and track all your personal work
- Produce quality products

...more goals

Dev manager goals

- Produce a superior product
- Fully use the team members' skills and abilities

Some of these goals point out some underlying beliefs of TSP, such as:
- You can't control what you can't measure
- Managers "use the team members' skills"

15

Scripts and forms

- TSP uses scripts to define the steps of building a software system

  - 21 activity scripts

  - 10 role scripts

- Often quite detailed

- Scripts prescribe completing 21 supporting forms

16

The inspection script

Purpose	To help engineers produce quality products	
Entry criteria	A completed and reviewed product with available materials	
Step	Activities	Description
1	Plan the inspection	The developer • Arranges with the quality/process manager or some other qualified team member to be the moderator • Handles the mechanics of setting up and running…
2	Hold the inspection	The moderator describes the inspection process The producer familiar the inspection team with the product.
…	…	…
Exit criteria		INS and LOGD forms completed and filed in the …

Choose TSP if...

- …you have the discipline to follow all of its specific steps

- …are already using the complementary Personal Software Process (PSP)

- …team members need this level of guidance

- …you do not have emergent requirements

Today's agenda

- ☑ Considerations
- ☑ Team Software Process
- ☐ Scrum
- ☐ Extreme Programming
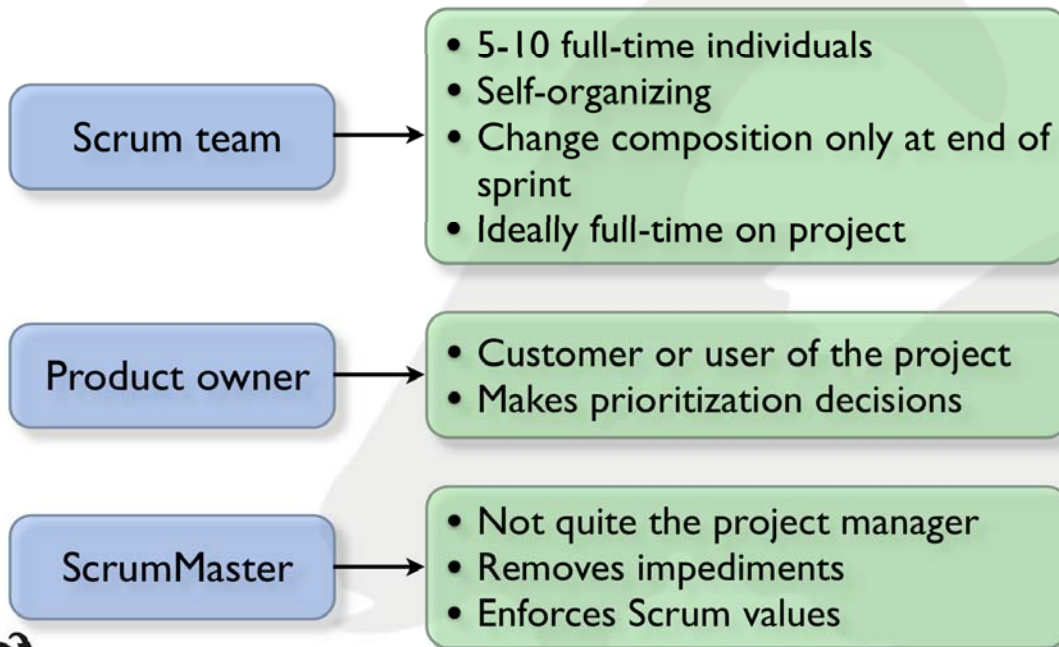- ☐ OpenUP/Basic
- ☐ Rational Unified Process

19

Scrum characteristics

- 30-day iterations called "sprints"

- Self-organizing teams

- No specific engineering practices prescribed

  - But many Scrum teams are adopting much of XP

- Uses generative rules to create an agile environment for delivering projects

20

Scrum roles

Scrum team →
- 5-10 full-time individuals
- Self-organizing
- Change composition only at end of sprint
- Ideally full-time on project

Product owner →
- Customer or user of the project
- Makes prioritization decisions

ScrumMaster →
- Not quite the project manager
- Removes impediments
- Enforces Scrum values

21

Product backlog

Return
Gift wrap
Cancel

- A "to-do list" for the entire project
- Often written as *user stories*
- Prioritized by the product owner
- Estimated in *story points* to track progress

22

Sprints

Copyright Mountain Goat Software, LLC
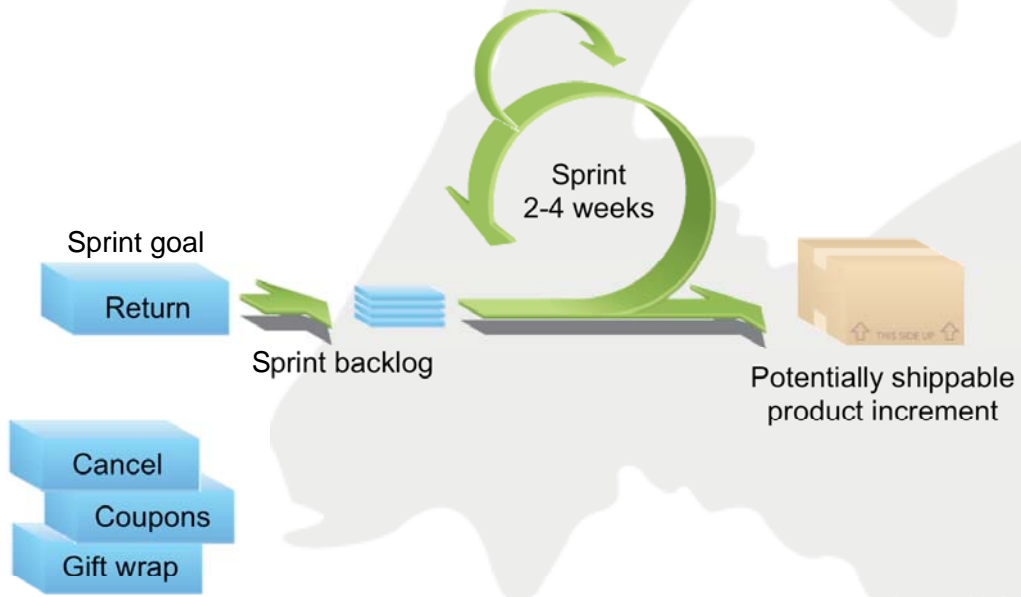
Review and planning

Copyright Mountain Goat Software, LLC
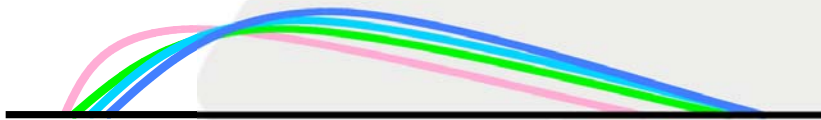
The daily scrum
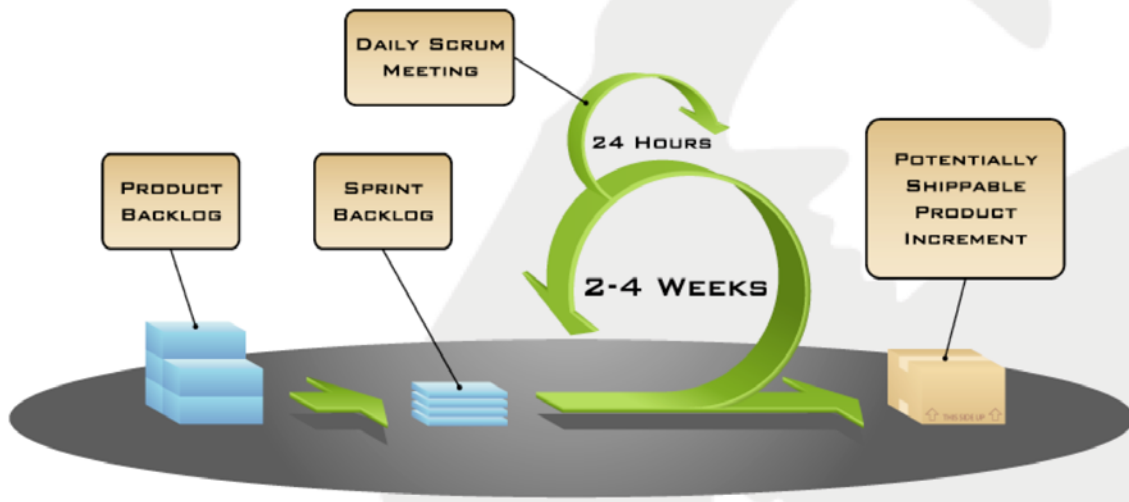
Scrum

Sequential vs. overlapping development

Rather than doing all of one thing at a time ——— Scrum teams do a little of everything all the time.

Source: "The New New Product Development Game", Hirotaka Takeuchi and Ikujiro Nonaka, *Harvard Business Review*, January 1986.

27

Scrum meetings

Sprint planning
- First half includes Product Owner who explains high priority needs
- Second half is team only
- Create sprint backlog

Daily scrum
- What did you do yesterday?
- What will you do today?
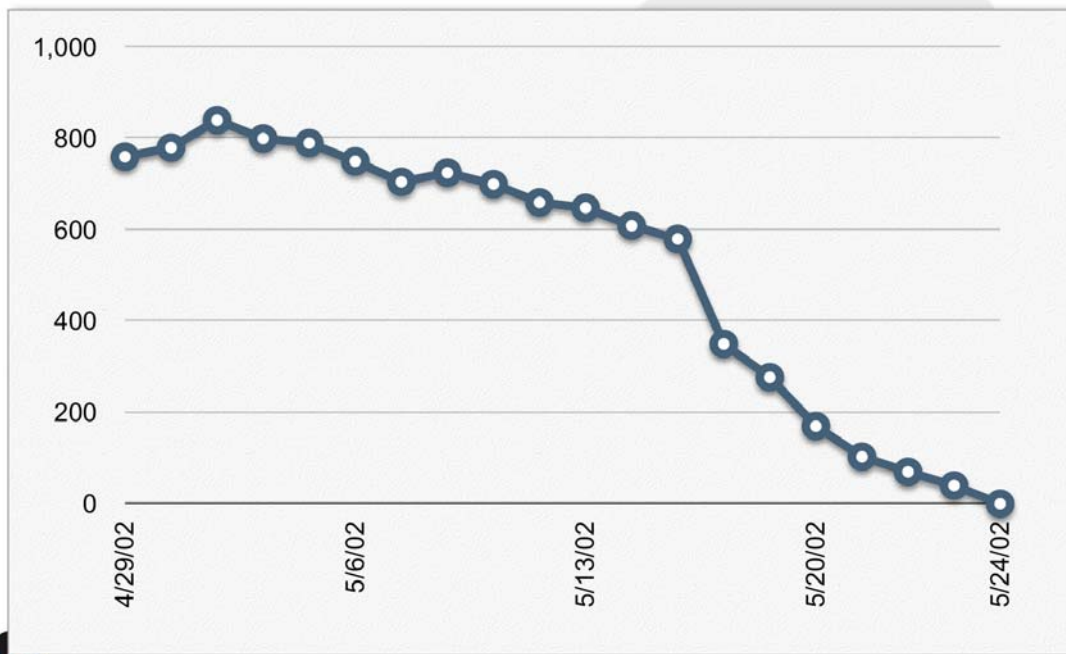- What's in your way?

Sprint review
- Team demos what was accomplished
- 2-hour prep time rule
- No PowerPoint!
- Anyone can attend

28

A sprint burndown chart

29

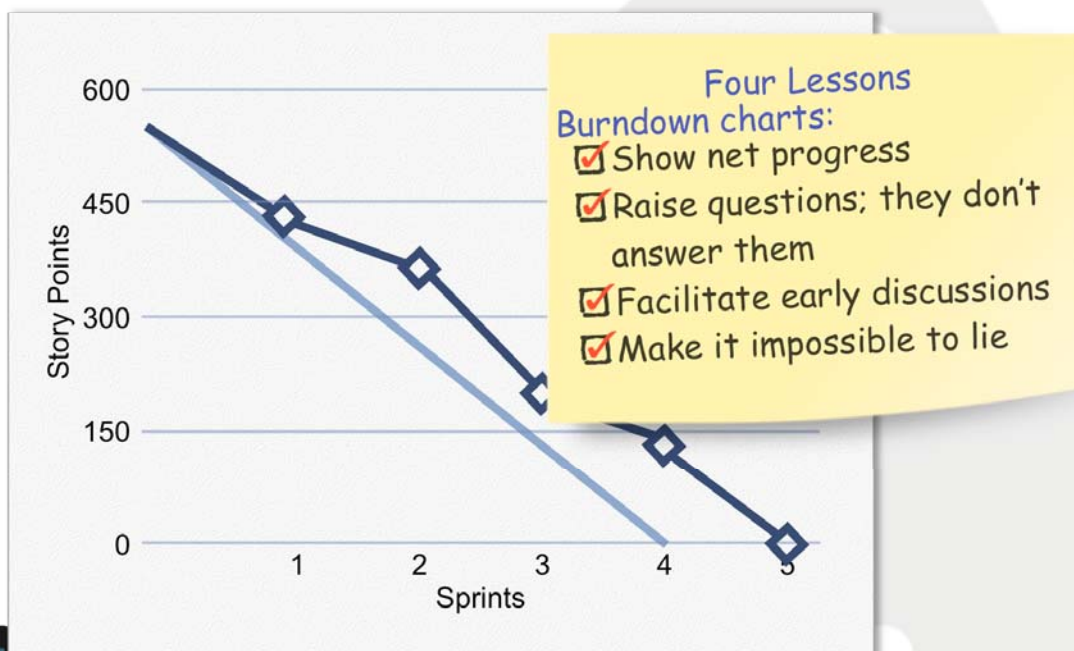# When will this project be released?

Four Lessons
Burndown charts:
☑ Show net progress
☑ Raise questions; they don't answer them
☑ Facilitate early discussions
☑ Make it impossible to lie

30

Scaling with a Scrum of Scrums
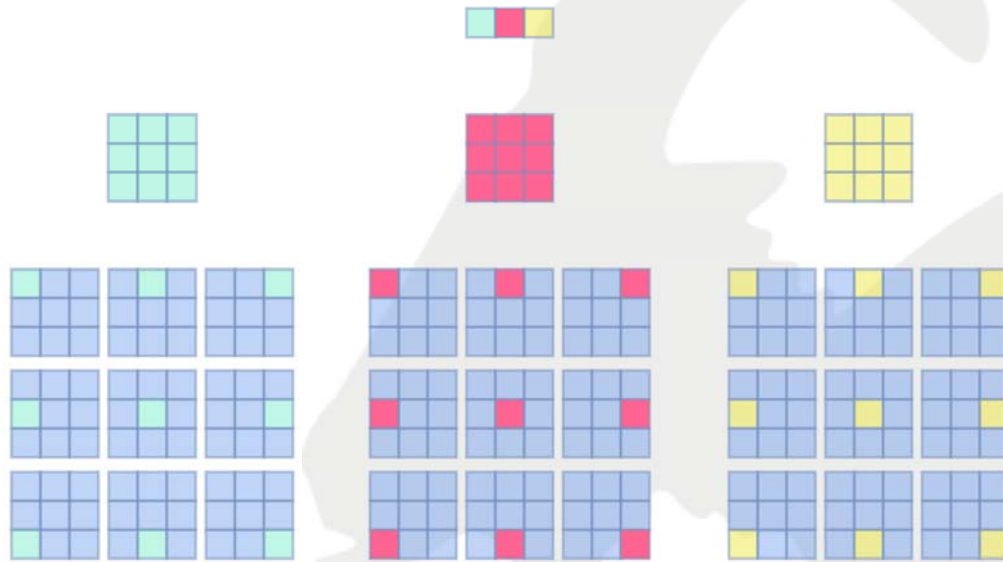
Choose Scrum if...

- …requirements are changing or emergent

- …you're willing to let the team self-organize

- …you need a management framework more than a set of engineering practices

- …you want to better manage risk

- …you need a proven, scalable agile process

Today's agenda

- ☑ Considerations
- ☑ Team Software Process
- ☑ Scrum
- ☐ Extreme Programming
- ☐ OpenUP/Basic
- ☐ Rational Unified Process

Extreme Programming

- Adds specific engineering practices to Scrum's project management approach

- A tough target to hit but highly productive if you do

XP's customer practices

On-site customer

Small releases

35

XP's quality practices

Metaphor

Testing

Simple design

Refactoring

Pair programming

36

XP's quality practices

The Planning Game

Sustainable pace

Collective ownership

Coding standards

Continuous integration

37

Choose Extreme Programming if...

- …you have loosely-defined or volatile requirements

- …you have or can develop strong engineering skills and practices

- …customers can be involved on a daily (hourly) basis

- …it's important you hit the bulls eye right off and you think you can

38

Today's agenda

☑ Considerations

☑ Team Software Process

☑ Scrum

☑ Extreme Programming

☐ OpenUP/Basic

☐ Rational Unified Process

OpenUP/Basic

- A subset of the Open Unified Process

- Delivered through the Eclipse Process Framework

- Derived from the Basic Unified Process

  - (If they're unified why are there so many of them?)

- Targeted at teams of 3-6 people

- Targeted at projects 3-6 months long

Comparison with RUP

	OpenUP/Basic	RUP
Pages	190	~2000
Roles	6	~45
Tasks	23	~150
Artifacts	17	~80

Source: *What is the Eclipse Process Framework*,
Per Kroll and Scott Ambler,
IBM Rational Software Development Conference 2006.

OpenUP roles

- Analyst

- Architect

- Developer

- Tester

- Project manager

Roles, tasks, and artifacts

Plan the project — Assess results — Prioritize work — Monitor and control project — Initiate Iteration — Plan Iteration

Project manager

Performs

Responsible for

Project plan — Status assessment — Iteration plan — Work items list

---
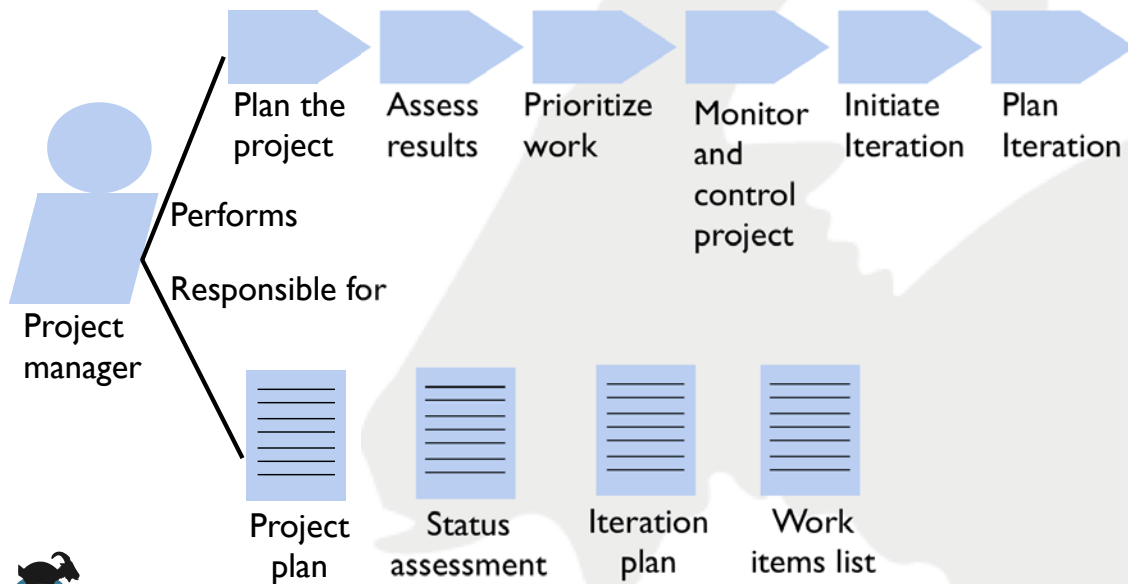
Open UP phases

Inception

- Establish scope and produce business case
- Mitigate business risks
- Get buy-in and decision to undertake project

Elaboration

- Reduce major risks to enable cost and schedule estimates
- Mitigate major technical risks
- Implement a baseline architecture
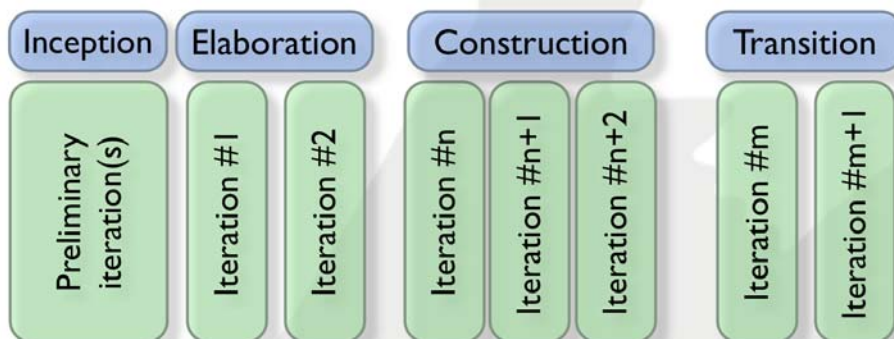- Implement key capabilities

Construction

- Majority of implementation
- Ends with deployment of a beta version

Transition

- Ensure software meets users' needs
- Prepare for release
- Solicit user feedback and make minor adjustments

45

Iterate within each phase

Inception	Elaboration		Construction			Transition	
Preliminary iteration(s)	Iteration #1	Iteration #2	Iteration #n	Iteration #n+1	Iteration #n+2	Iteration #m	Iteration #m+1

46

Choose OpenUP/Basic if...

- You like RUP but want something lighter

  - You want an open source version of RUP

- You have a small project (3-6 people; 3-6 months)

- You want a semi-agile process based on role, task and artifact guidance

47

Today's agenda

- ☑ Considerations
- ☑ Team Software Process
- ☑ Scrum
- ☑ Extreme Programming
- ☑ OpenUP/Basic
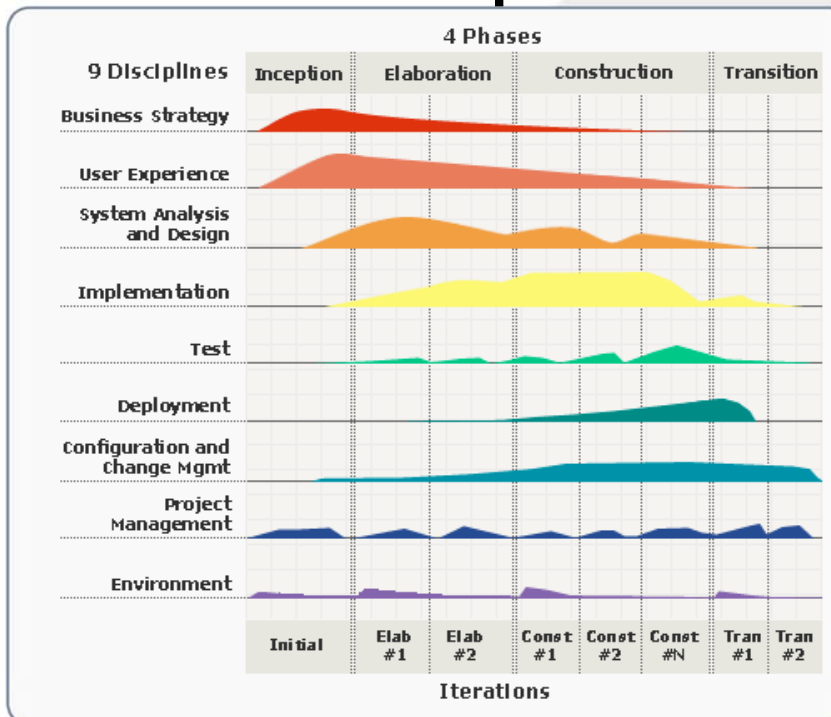- ☐ Rational Unified Process

48

The Rational Unified Process

- Proprietary, sold as a product from IBM

- Use case-driven

- Architecture-centric

- Iterative and incremental

- Lots of role, task and artifact guidance

- Strongly oriented toward OO languages

49

Phases and disciplines

50

Some common RUP challenges

- It's meant to be a cookbook but is often used like a buffet

- Phases are mistakenly treated like a waterfall

- Over-modeling is common

- Iterations should be weeks not months

Choose the RUP if...

- …you need a process framework that has everything ever possible already in it

- …you can resist the temptation to adopt too much of it

- …you want well-defined roles

- …it's important to have a well-documented process that new hires may be familiar with

- …you do not have emergent requirements

Where to go next

Team Software Process	• *Introduction to the Team Software Process* by Watts S. Humphrey
Scrum	• www.mountaingoatsoftware.com/scrum • *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle • *Agile Project Management with Scrum* by Ken Schwaber
Extreme Programming	• www.xprogramming.com • c2.com/cgi/wiki?ExtremeProgramming Roadmap • extremeprogramming@yahoogroups.com

Where to go next

OpenUP	• www.eclipse.org/epf • *Agility and Discipline Made Easy: Practices from OpenUP and RUP* by Per Kroll and Bruce MacIsaac
RUP	• *The Rational Unified Process Made Easy* by Per Kroll and Philippe Kruchten
Agile in General	• www.agilealliance.org • especially Articles and Roadmap sections • www.mountaingoatsoftware.com

Upcoming public classes

| Date	What	Where
September 26-27	Certified ScrumMaster	London
September 28	Agile Estimating & Planning	London
November 7-8	Certified ScrumMaster	Santa Clara
November 9	Agile Estimating & Planning	Santa Clara
November 29-30	Certified Product Owner (with Ken Schwaber)	Boulder
January 16-17	Certified ScrumMaster (with Ken Schwaber)	Orlando

Register at
www.mountaingoatsoftware.com

55

Mike Cohn contact info
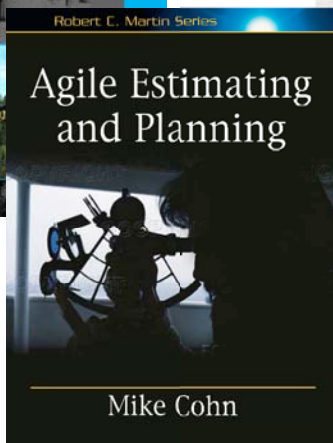
mike@mountaingoatsoftware.com

www.mountaingoatsoftware.com

(720) 890-6110 (office)

(303) 810-2190 (mobile)

MOUNTAIN GOAT
SOFTWARE

56