Software Engineering is still improving…

- You won't develop software:
  - as it was done 15-30 years ago
  - the same way in 15-30 years from now

Scrum + Engineering Practices: Experiences of Three Microsoft Teams

- Williams_ESEM2011[…].pdf presents the experiences of three Microsoft teams with Scrum
  - Laurie Williams is a Software Engineering researcher at NCSU and has worked for years on agile methodologies
  - The other authors work at Microsoft

Scrum + Engineering Practices: Experiences of Three Microsoft Teams

- Thesis: Scrum, when augmented with engineering practices, improves:
  - quality
  - productivity and
  - estimation accuracy

Scrum is Popular

- 2008 Industry Survey of Agile Methodologies
  - 49% of respondents using agile are using Scrum
  - 22% of respondents are using Scrum with Extreme Programming (XP)

Extreme Programming (XP)

Extreme Programming (XP)

- Agile development methodology, on:
  - Adaptable to customer changes
  - Focusing on software quality

https://en.wikipedia.org/wiki/Extreme_programming

Extreme Programming (XP)

- Practices:
  - Pair Programming (PP)
  - Test-Driven Development (TDD)
  - Continuous Integration (CI)
  - Refactoring
  - Coding Standards
  - Collective/Team/Shared Code Ownership
  - Sustainable Pace
  - etc.

https://en.wikipedia.org/wiki/Extreme_programming

Scrum is Popular

- 2008 Industry Survey of Agile Methodologies
  - 49% of respondents using agile are using Scrum
  - 22% of respondents are using Scrum with Extreme Programming
  - What do you think the other respondents are using?

Scrum is Popular

- 2008 Industry Survey of Agile Methodologies
  - 49% of respondents using agile are using Scrum
  - 22% of respondents are using Scrum with Extreme Programming
  - What do you think the other respondents are using?
- 60% of respondents in a Microsoft Survey use Scrum
- Current popularity may be even higher!

What Scrum Practices were used at Microsoft?

- Product Owner creates and prioritizes requirements
- Incremental development ("iterations" in 2011Williams) through sprints
- Spikes (prototypes)
- Poker Planning
- Sprint Planning Meeting
- "Quality Gates" (aka Definition of Done)
- Time-boxing
- Daily Scrum Meeting
- Sprint Review
- Sprint Retrospective

Notice Some Differences with the "Sims book" Scrum?

- Roles: Project Manager, "Tester" and Management also attend Sprint Planning Meeting. Why?

Notice Some Differences with the "Sims book" Scrum?

- Roles: Project Manager, "Tester" and Management also attend Sprint Planning Meeting. Why?

- What type of language can users stories be written in if teams are building "internal" tools for other engineers?

Notice Some Differences with the "Sims book" Scrum?

- Roles: Project Manager, "Tester" and Management also attend Sprint Planning Meeting. Why?

- What type of language can users stories be written in if teams are building "internal" tools for other engineers?

- Microsoft augmented Scrum with engineering practices

Why Augment Scrum with Engineering Practices?

- Scrum doesn't describe how to build software
- Consequently… it's arguably not a software lifecycle!
- The original Scrum article almost didn't mention software!

What Engineering Practices are in Use at the Microsoft Scrum Teams?

- Planning Poker
- Continuous integration
- Unit Test-Driven development
- "Quality Gates" (aka Definition of Done)
- Source control (TFS)
- Code coverage
- Static Analysis Tools
- Peer Review
- Code Documentation (XML annotations)

Sample Definition of Done for Microsoft Teams

- All unit tests must pass
- Unit test code coverage must be at least 80% (for all teams except Team B)
- All public methods must have documentation
- All non-unit test code must not have any static analysis errors or warnings (see Section IV.9)
- Build must compile with no errors or warnings on the highest level

Team Productivity

(assumption 1 month = 20 work days)

- Team A:

4 engineers … 24,952LOC … 14 months

~…LOC engineer/day

- Team B:

3 engineers … 8,826LOC … 11 months

~…LOC engineer/day

- Team C:

19 engineers … 31,399LOC … 18 months

~…LOC engineer/day

Team Productivity

(assumption 1 month = 20 work days)

- Team A:

4 engineers … 24,952LOC … 14 months

~22LOC engineer/day

- Team B:

3 engineers … 8,826LOC … 11 months

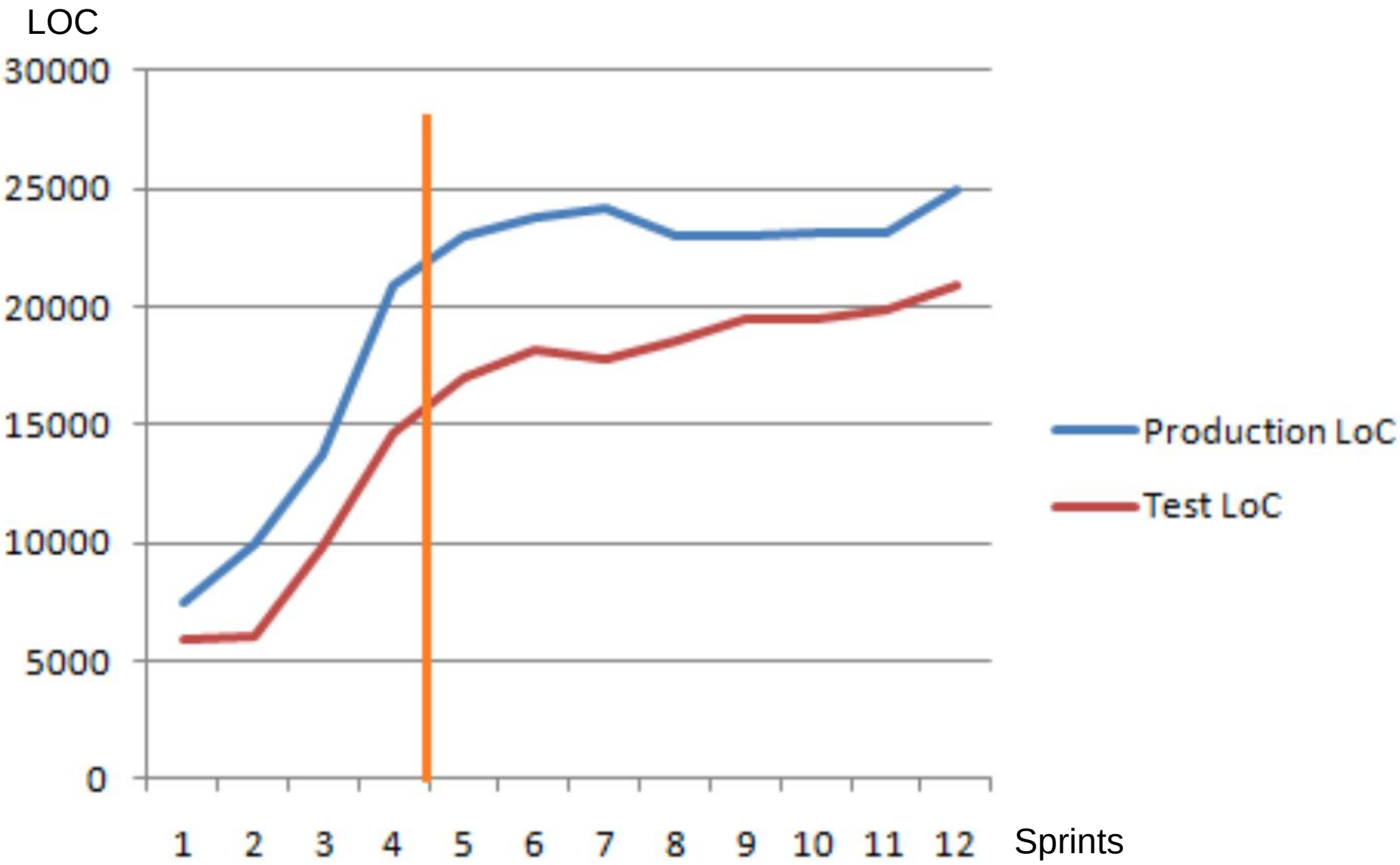
~13LOC engineer/day

- Team C:

19 engineers … 31,399LOC … 18 months

~4.6LOC engineer/day

Team A Cumulative LOC Produced



Results

	Team-A	Team-B	Team-C
Product LoC	24,952	8,826	31,399
Test LoC	20,912	4,031	26,283
$\frac{\text{Test}}{\text{Product}}$ LoC Ratio			
Test Coverage			
Project Duration			
Continuous Integration			
Delivered Defects			
Defect Density (Defects/KLOC)			

Results

	Team-A	Team-B	Team-C
Product LoC	24,952	8,826	31,399
Test LoC	20,912	4,031	26,283
$\frac{\text{Test}}{\text{Product}}$ LoC Ratio	0.84	0.46	0.84
Test Coverage			
Project Duration			
Continuous Integration			
Delivered Defects			
Defect Density (Defects/KLOC)			

Results

	Team-A	Team-B	Team-C
Product LoC	24,952	8,826	31,399
Test LoC	20,912	4,031	26,283
$\frac{\text{Test}}{\text{Product}}$ LoC Ratio	0.84	0.46	0.84
Test Coverage	82%	53%	N/A
Project Duration			
Continuous Integration			
Delivered Defects			
Defect Density (Defects/KLOC)			

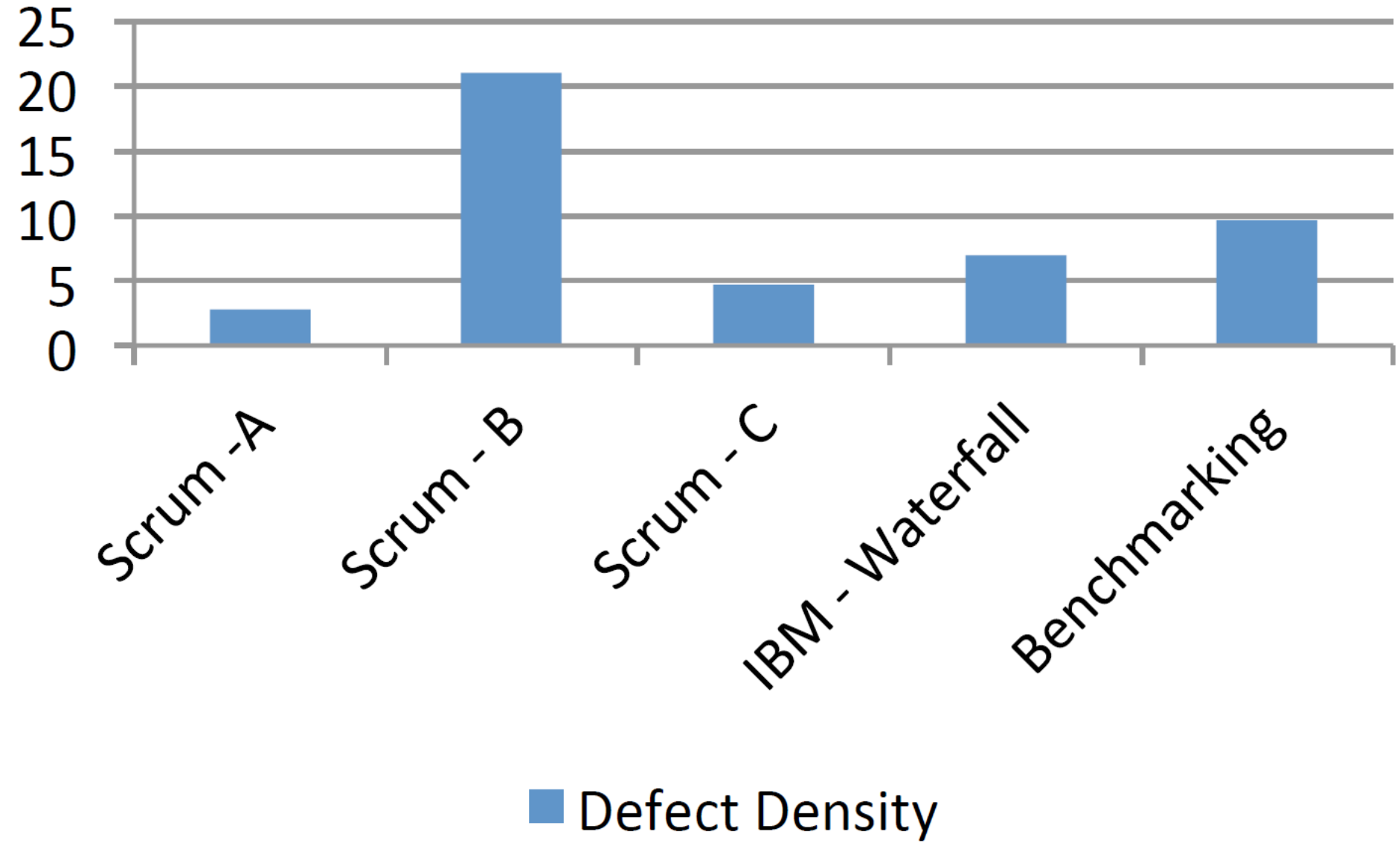
Results

	Team-A	Team-B	Team-C
Product LoC	24,952	8,826	31,399
Test LoC	20,912	4,031	26,283
$\frac{\text{Test}}{\text{Product}}$ LoC Ratio	0.84	0.46	0.84
Test Coverage	82%	53%	N/A
Project Duration	14 months	11 months	18 months
Continuous Integration	yes	yes	yes
Delivered Defects	70	187	149
Defect Density (Defects/KLOC)			

Results

	Team-A	Team-B	Team-C
Product LoC	24,952	8,826	31,399
Test LoC	20,912	4,031	26,283
$\frac{\text{Test}}{\text{Product}}$ LoC Ratio	0.84	0.46	0.84
Test Coverage	82%	53%	N/A
Project Duration	14 months	11 months	18 months
Continuous Integration	yes	yes	yes
Delivered Defects	70	187	149
Defect Density (Defects/KLOC)	2.8	21	4.75

Scrum vs. non-Scrum Teams: Defect Density Comparison



Learnings

- Scrum with sound Engineering Practices delivered higher quality (about 3-5 defects/KLOC) than non-scrum teams (about 7 defects/KLOC)
- Scrum with weak Engineering Practices (testing) failed
- Planning Poker improved estimation accuracy

What's on the Scrum Quiz? Texts and Lectures

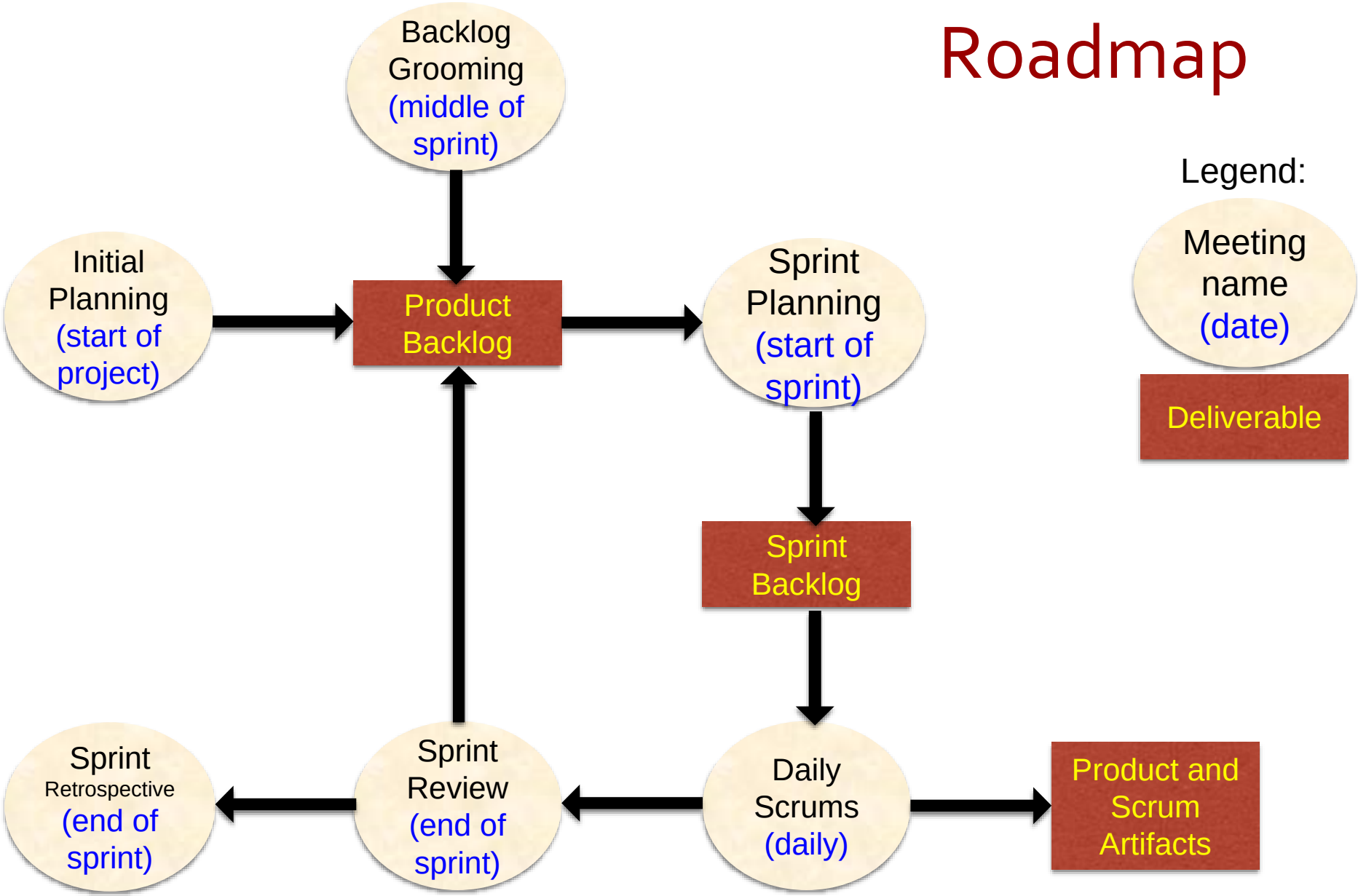
- "The Elements of Scrum"
- Scrum lectures
- Williams_ESEM2011_Scrum_3_Microsoft_Teams.pdf on Piazza (Additional Reading)
  - Williams, Brown, Meltzer and Nagappan. "Scrum + Engineering Practices: Experiences of Three Microsoft Teams." *Proceedings of the 2011 International Symposium on Empirical Software Engineering and Measurement*. IEEE Computer Society. 2011.

Scrum Quiz

- Take-home
- Open book, open notes
- Assigned via Blackboard
- Assigned ~Monday (2/12) due Monday (2/19)
- Designed to take ~20-30 minutes
- Time limit not enforced

Scrum Review

Scrum Process Roadmap



Scrum: Sprint Meetings / Activities

- Initial Planning (start of the project)
- Sprint Planning Meeting (start of Sprint)
- Daily Scrum (daily)
- Backlog Grooming (~ middle of sprint)
- Sprint Review (end of Sprint)
- Sprint Retrospective (end of Sprint)

What's on the Quiz?
Product Backlog

- User Stories
  - Who writes them?
  - Template for writing them (Role-Goal-Benefit)
  - Prioritized (Who prioritizes them?)
  - Estimates (Who estimates them? How?)
  - Epics vs. detailed stories

- Acceptance Criteria
  - Template for writing them (Given-When-Then)

What's on the Quiz?
Sprint Backlog

- Tasks
  - Engineering Language
  - Owner
  - Estimated (by who?)

Expect Questions

- Which test your knowledge of Scrum concepts
- Test your in-depth understanding of Scrum (WHY do we do things this way)
- Test your ability to apply Scrum

Class Exercise: New Team Member

- Your scrum team is developing a classroom response (i.e. clicker) service and mobile apps

- Your team will begin Sprint 5 (of 10 expected) on Monday

Class Exercise: New Team Member

- Management promoted Alice, the developer who did your database table design, to lead another project

- Alice hired Bob, a recent graduate with 400-level database coursework and database intern experience, to replace her

- What does Bob need to know to become productive?

Class Exercise: New Team Member

- Management promoted Alice, the developer who did your database table design, to lead another project

- Alice hired Bob, a recent graduate with 400-level database coursework and database intern experience, to replace her

- What does Bob need to know to become productive?
- How can Scrum help introduce Bob to your project?