

USC



agile software
development
mark duvall, april 2017



Innovation center of Excellence

what we do...

- multiple lines of business
 - SVOD
 - POD
 - digital greetings
- special research projects



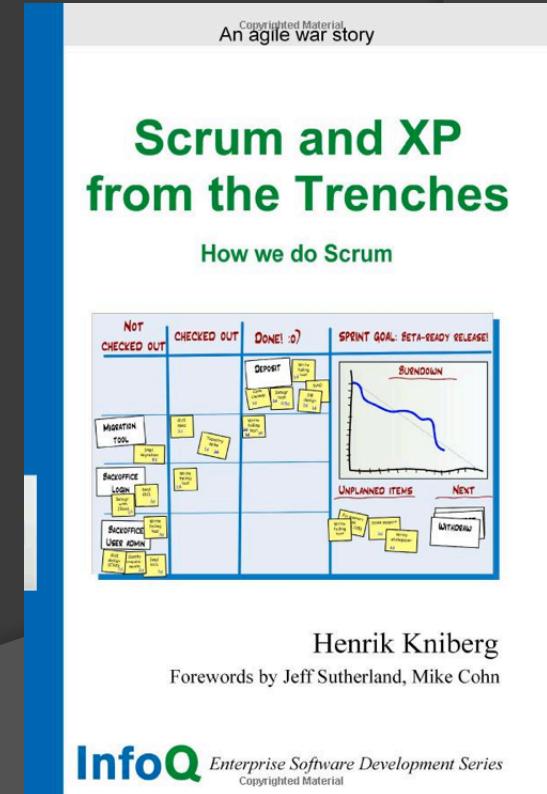
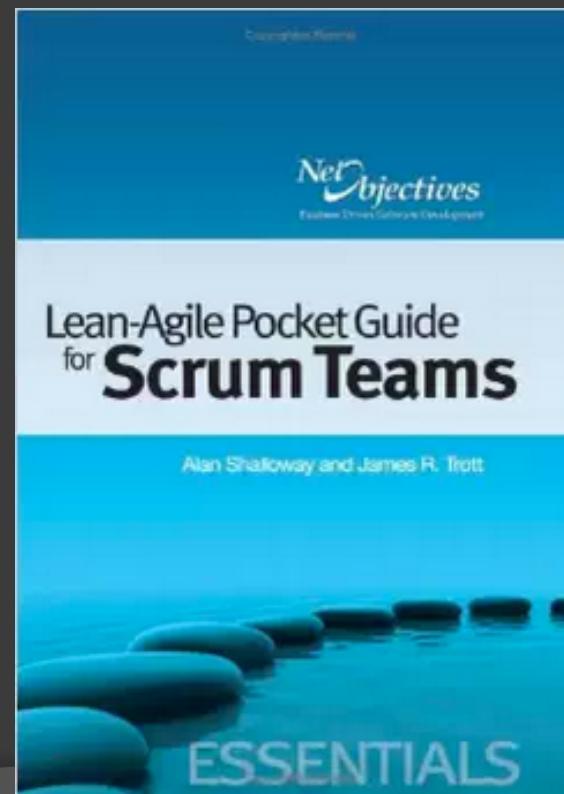
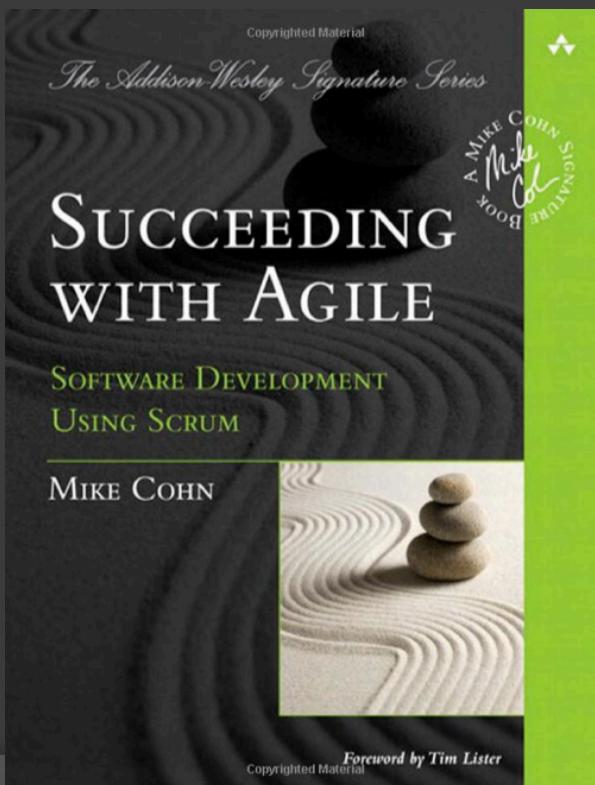
Mark DuVall Agile Bio

Technology Executive

- practicing agile development since 2002
- ken schwaber/mike cohn disciple
- csm x 4, cspo x 2
- transitioned 7 companies
- taught agile to 100s
- delivered numerous of projects w/agile
- been through 1000s of sprints
- agile alliance, acm, ieee

references

- mike cohn, <http://www.mountaingoatsoftware.com>
- agile alliance, <http://www.agilealliance.org>
- scrum alliance, <http://scrumalliance.org>
- scrum.org, <http://www.scrum.org>
- crisp blog, <http://blog.crisp.se>



observations



social activity



encourages collaboration



transparency provides accountability



peer accountability drives performance



teams will self optimize

software development

- notoriously difficult to deliver
 - on-time
 - on-budget
- quest for determinism
- estimation models
- formalized processes
- organizations to manage process
- documentation driven

scrum

- oopsla 1995, schwaber & southerland
- lightweight process framework
- inspection and adaptation - error correction
- iterate and refactor
- ability to incorporate best practices
- time-boxes
- collocated teams
- quality never sacrificed for dates
- shippable product quality at each iteration (aka sprint)
- definition of “done”

agile manifesto 2001

- individuals and interactions over processes and tools
 - working software over comprehensive documentation
 - customer collaboration over contract negotiation
 - responding to change over following a plan
-
- <http://agilemanifesto.org/principles.html>

differences

non-agile development

command & control

document driven

serial in nature

big upfront design

test late

bureaucratic

heavy, rigid process

agile development

self managing team

high bandwidth communication

iterative

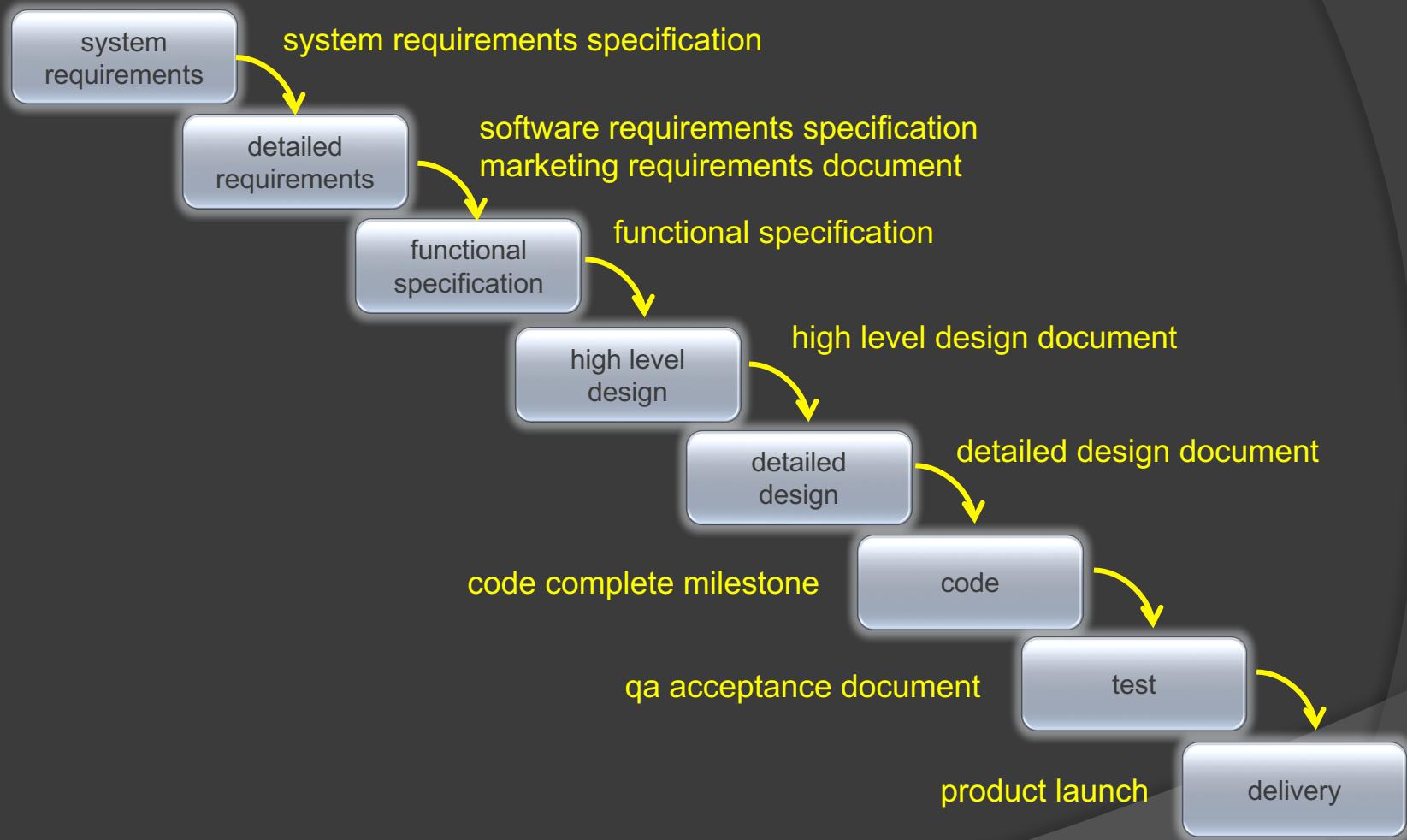
evolutionary/emerging design

test early & continuously

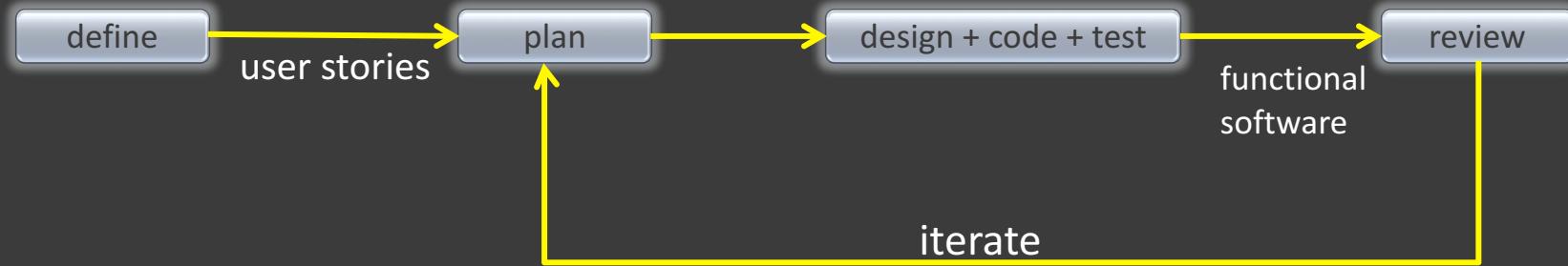
collaborative

lightweight, tailor-able process

waterfall



scrum



agile tenants

- create software incrementally
- deliver value quickly
- commitment driven
- self organizing and managing teams
- continuous improvement
- high bandwidth communications
- evolutionary/emerging design
- customer intimacy
- progress is measured by functioning software
- maintain a sustainable pace

roles

- ◎ **product owner**
 - classically the product manager, system or business analyst
 - maximizes roi of each sprint
 - manages the product backlog
- ◎ **scrum master (agile champion)**
 - development manager or project manager
 - enforces process, insures process adherence
 - impediment removal
 - facilitates communications
- ◎ **scrum team (pigs)**
 - cross functional team (sm, po, dev, qa, ux) that builds the product
 - team size: 3-12 pigs
 - maintains sprint backlog
- ◎ **observers (chickens)**
 - people that do not have deliverables in the sprint
 - stakeholders & non-stakeholders

user stories

- ◎ a feature description
- ◎ as a <role>, i want <goal> so that <reason>
- ◎ invest
 - independent
 - negotiable
 - valuable to users
 - estimate-able
 - small
 - testable

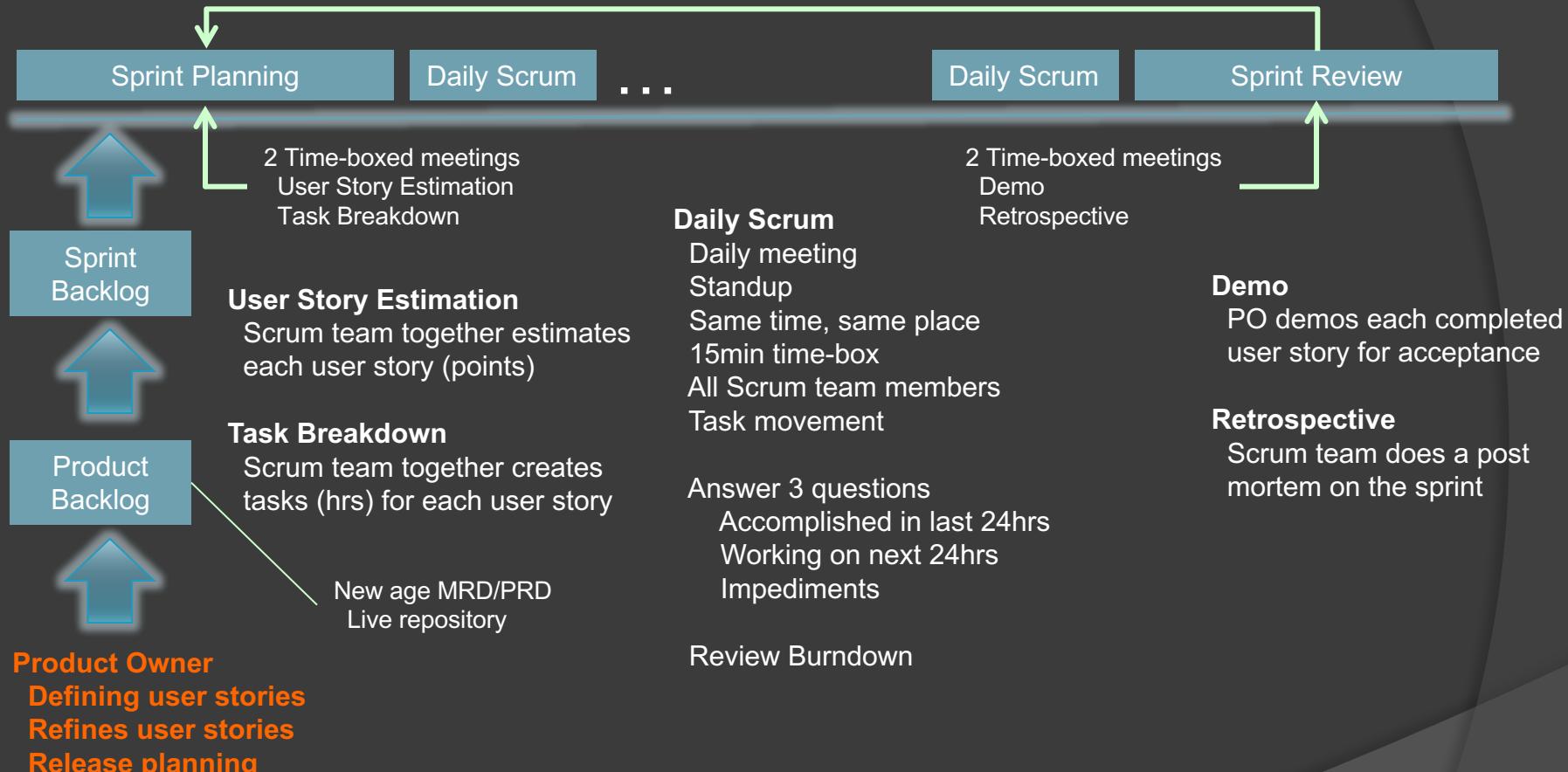
product backlog

- the new age requirement specification/mrd/prd
- live repository of all project work
 - user stories
 - defects
- contributed to by anyone
- managed by product owner
- groomed and prioritized by po
- estimated
 - by scrum team
 - story points
- subset moved to sprint backlog; every sprint

done

- code complete
- all code checked into source code control (TFS)
- all tests checked into source code control
- passes coding standards (stylecop)
- passes peer code review (team review)
- passes static code analysis (ndepend)
- builds on build server (tfss)
- builds with no warnings in pre-build area (shelf set)
- all unit test & test automation executes in dev/qa/pre-prod environments
- all unit test pass
- unit test code coverage is 65%
- no priority 1 defects
- no profiling issues (dotTrace)
- documented
- accepted by product owner

scrum basic mechanics



sprint planning

- first day of the sprint
 - product backlog - n hour time-boxed
 - backlog review, user story by user story
 - estimate and prioritize stories
 - sprint backlog - n hour time-boxed
 - task breakdown: tasks for implementing stories
 - assign hours and resources

estimation

◎ story points

- relativistic educated guesstimate of effort
- not directly correlated to time
- non-linear numbers like Fibonacci sequence
- t-shirt sizing (xs,x,m,l,xl)
- used for sprint planning (capacity)

◎ epic

- story too large to fit in a sprint
- broken down into smaller stories

estimation process

- product backlog estimation
 - discuss of each story
 - planning poker - team
 - until backlog empty or time-box expires
 - prioritize – by po
- assign stories to sprint backlog
 - based on team's *velocity*
 - assume initial velocity as 65%
- assign tasks to sprint stories
 - <8 hour tasks
 - *assign owners
- adjustments
 - move stories back if velocity is exceeded

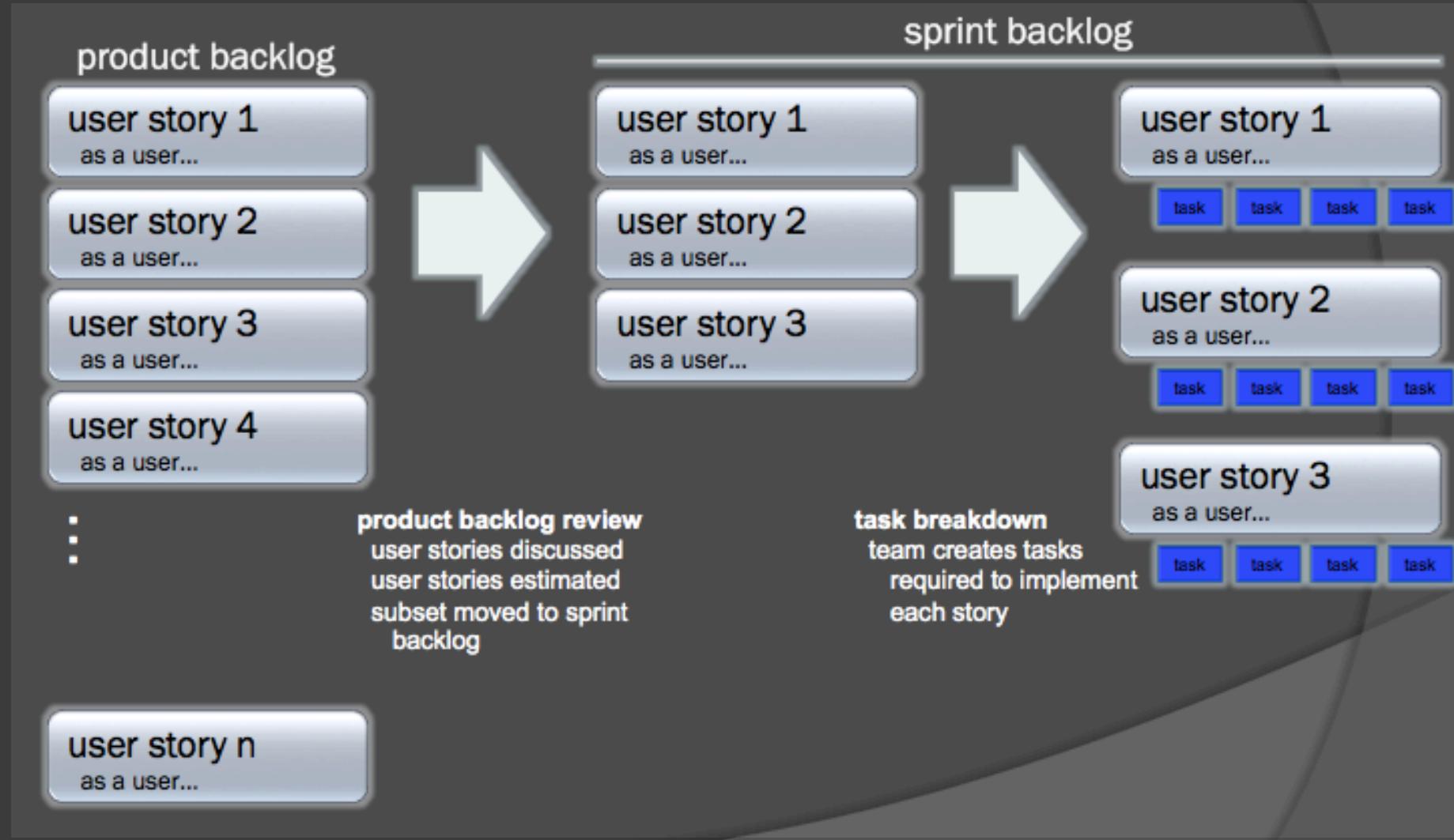
planning poker

- iterative approach
 - adapted from delphi
 - entire scrum team participates
 - everyone has a deck of story point cards
-
- user story discussion
 - everyone selects a card privately
 - cards are revealed simultaneously
 - discuss outliers
 - re-estimate till convergence or timebox
-
- <http://www.planningpoker.com>

sprint backlog

- repository of work for each sprint
 - user stories
 - broken down into tasks
 - task granularity less than 8 hrs
- managed by scrum team
 - updated daily, in real-time

flow



composition



daily scrum

- daily meeting
- same place, same time
- time-boxed 15 minutes
- standup
- \$1 for being late
- no one can interrupt the current speaker
- details go into the parking lot
- 3 questions
 - what did you accomplish in the last 24 hrs?
 - what are you going to work on in the next 24hrs?
 - any impediments?
- review burndowns post scrum

electronic task board

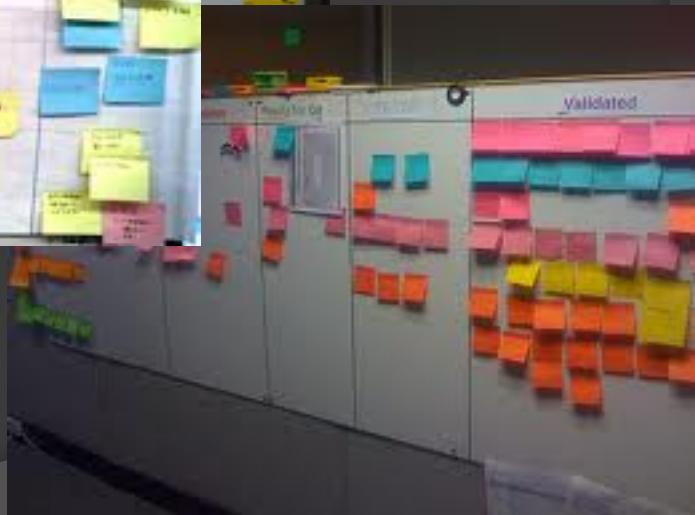
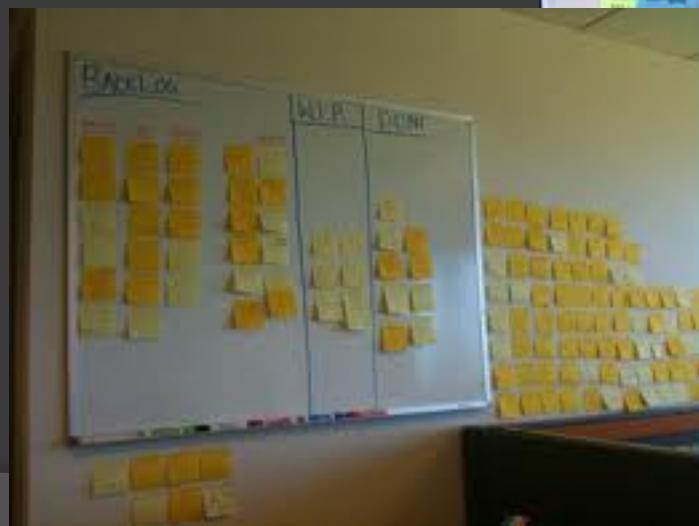
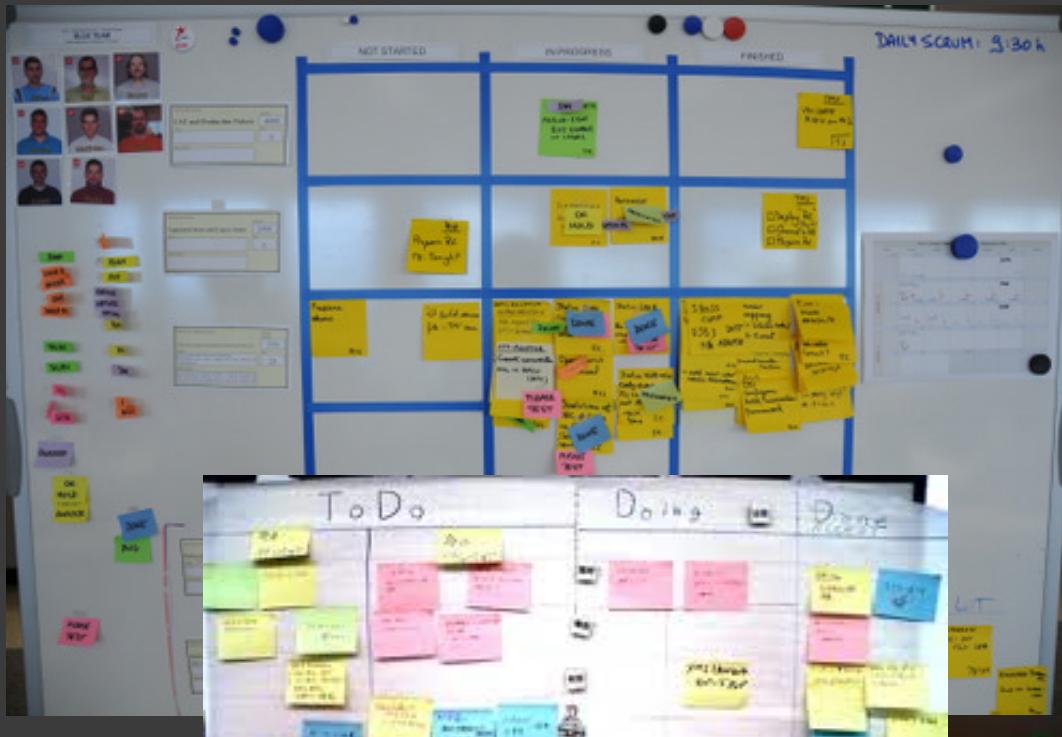
Task Board

Sprint 4
Monday April 11, 2011

Change iteration: Sprint 4 | Show work owned by: All Team Members | Hide accepted work:

	Defined	In-Progress	Completed												
<p>D P C A</p> <p>US7462 Research SPIKE - Data Replication</p> <p>Phillip</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Actual</td></tr><tr><td>2</td><td>2</td><td>0</td></tr></table>	Est	To Do	Actual	2	2	0	<p>TA14348 Research SPIKE</p> <p>Christopher</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Act</td></tr><tr><td>2</td><td>2</td><td>-</td></tr></table>	Est	To Do	Act	2	2	-		
Est	To Do	Actual													
2	2	0													
Est	To Do	Act													
2	2	-													
<p>D P C A</p> <p>DE2702 Error found in US7236: Create Feed Handler</p> <p>Sean</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Actual</td></tr><tr><td>1</td><td>0</td><td>3</td></tr></table>	Est	To Do	Actual	1	0	3			<p>TA14347 Fix it!</p> <p>Sean</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Act</td></tr><tr><td>1</td><td>0</td><td>3</td></tr></table>	Est	To Do	Act	1	0	3
Est	To Do	Actual													
1	0	3													
Est	To Do	Act													
1	0	3													
<p>D P C A</p> <p>DE2699 Error found in US7236: Create Feed Handler</p> <p>Sean</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Actual</td></tr><tr><td>1</td><td>0</td><td>1</td></tr></table>	Est	To Do	Actual	1	0	1			<p>TA14342 Fix It!</p> <p>Sean</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Act</td></tr><tr><td>1</td><td>0</td><td>1</td></tr></table>	Est	To Do	Act	1	0	1
Est	To Do	Actual													
1	0	1													
Est	To Do	Act													
1	0	1													
<p>D P C A</p> <p>US7241 B2S Unit Tests</p> <p>Phillip</p> <table border="1"><tr><td>Est</td><td>To</td></tr><tr><td>8</td><td></td></tr></table>	Est	To	8			<p>TA14118 Create Unit Tests</p> <p>Sean</p> <table border="1"><tr><td>Est</td><td>To Do</td><td>Act</td></tr><tr><td>8</td><td>6</td><td>2</td></tr></table>	Est	To Do	Act	8	6	2			
Est	To														
8															
Est	To Do	Act													
8	6	2													
															

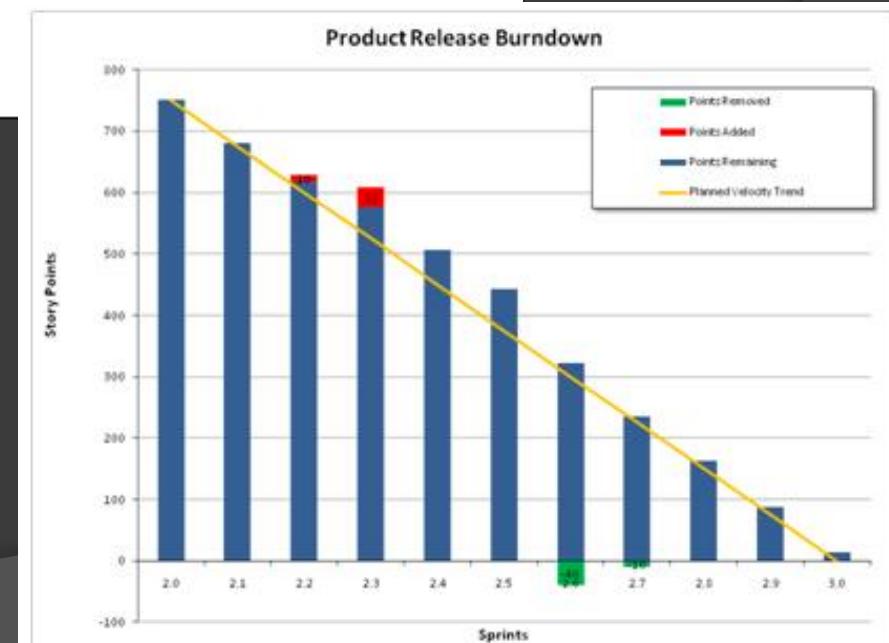
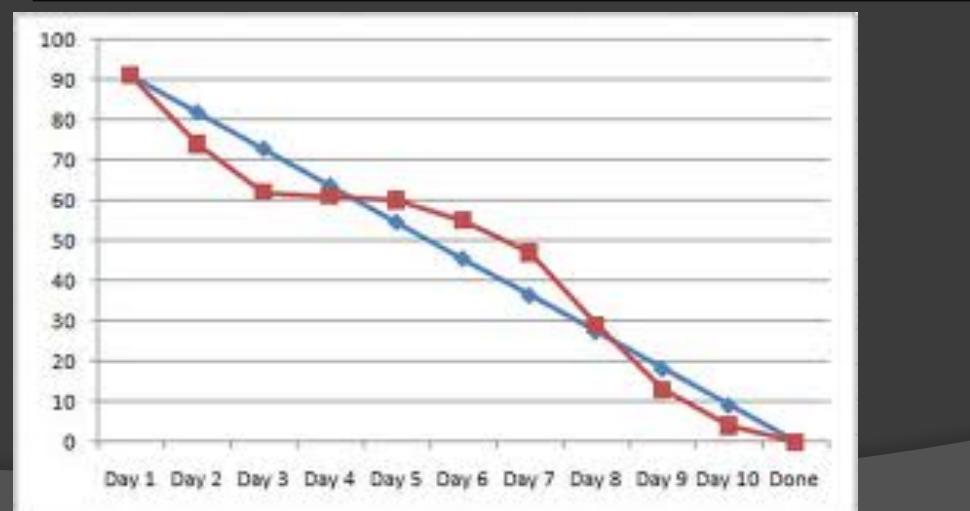
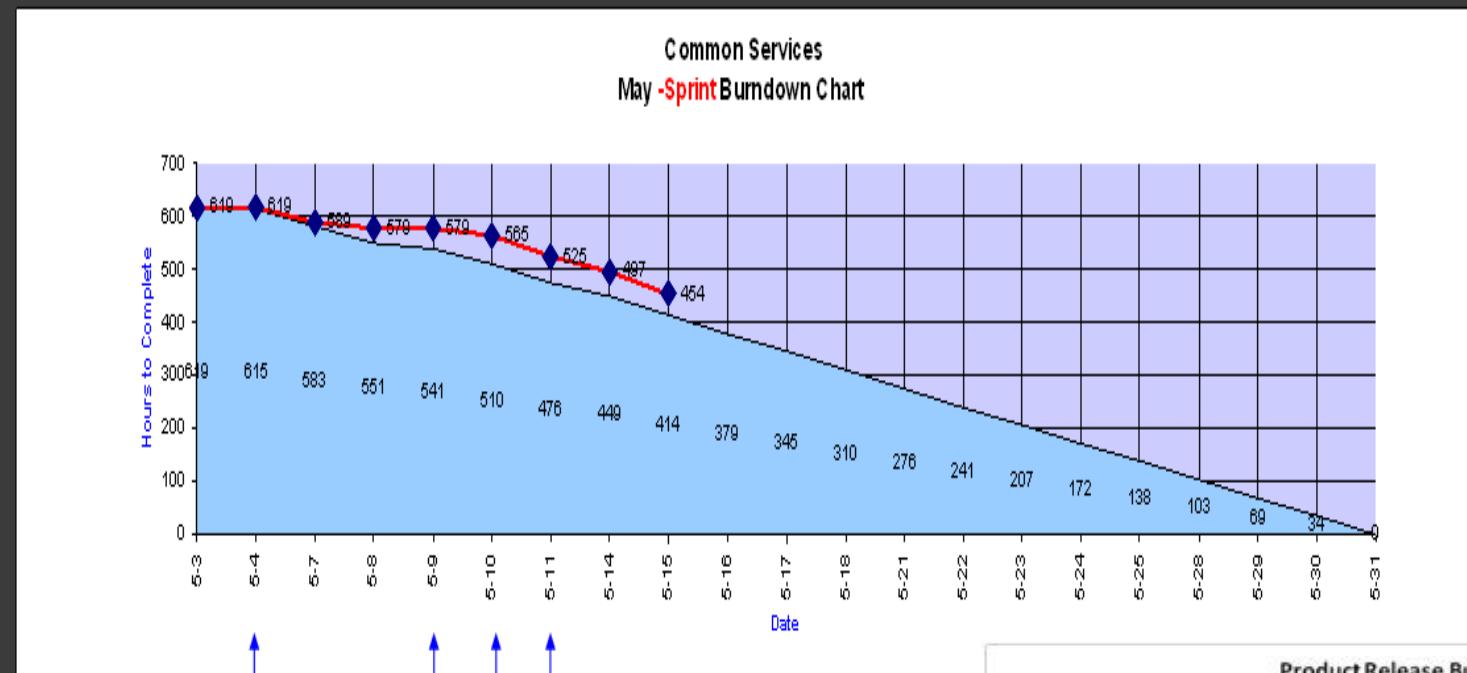
physical task board



daily scrum



burndown charts



sprint review

- last day of the sprint
 - demo - n hour time-boxed
 - demonstrate sprint accomplishments
 - story acceptance walk through
 - log defects for next sprint
 - retrospective - n hour time-boxed
 - what went right?
 - what went wrong?
 - what can we do to improve?

best practices

- evolutionary / emerging design
- vertical slices
- continuous integration / continuous deployment
- shorter sprints
- fail quickly - test early, test often
- high flow / eliminate waste
- sprint start/end mid-week
- test driven development
- pair programming / mobbing
- peer code review
- promote transparency - information radiators

scrum anti-patterns

- “manager” estimates work
- developers not doing their own task breakdowns
- untrackable sprints
- no customer value in sprint
- partially complete sprints
- story or task creep in sprint backlog
- no definition of “done”
- working on tasks not in the sprint backlog
- qa under utilized
- testable software completed late in the sprint
- lack of peer accountability or commitment
- impediments not immediately addressed
- floating timeboxes
- back loaded effort: slow starts – quick finishes

continuous integration & deployment

○ build automation system

- builds are triggered by source code check-ins
- unit tests are executed as part of each build
- tests are executed in the context of a profiler
- static code analysis
- each build is versioned
- each build is deployed (dev, qa, staging, prod)
- test automation scripts execute

questions