

An Introduction to Scrum

Premek Brada, Lukas Holy
<http://esecc.zcu.cz/>
5.11.2013

OVERVIEW OF SCRUM

What to Expect from Scrum

- Quality?
- Speed?
- Cost?
- Agility?

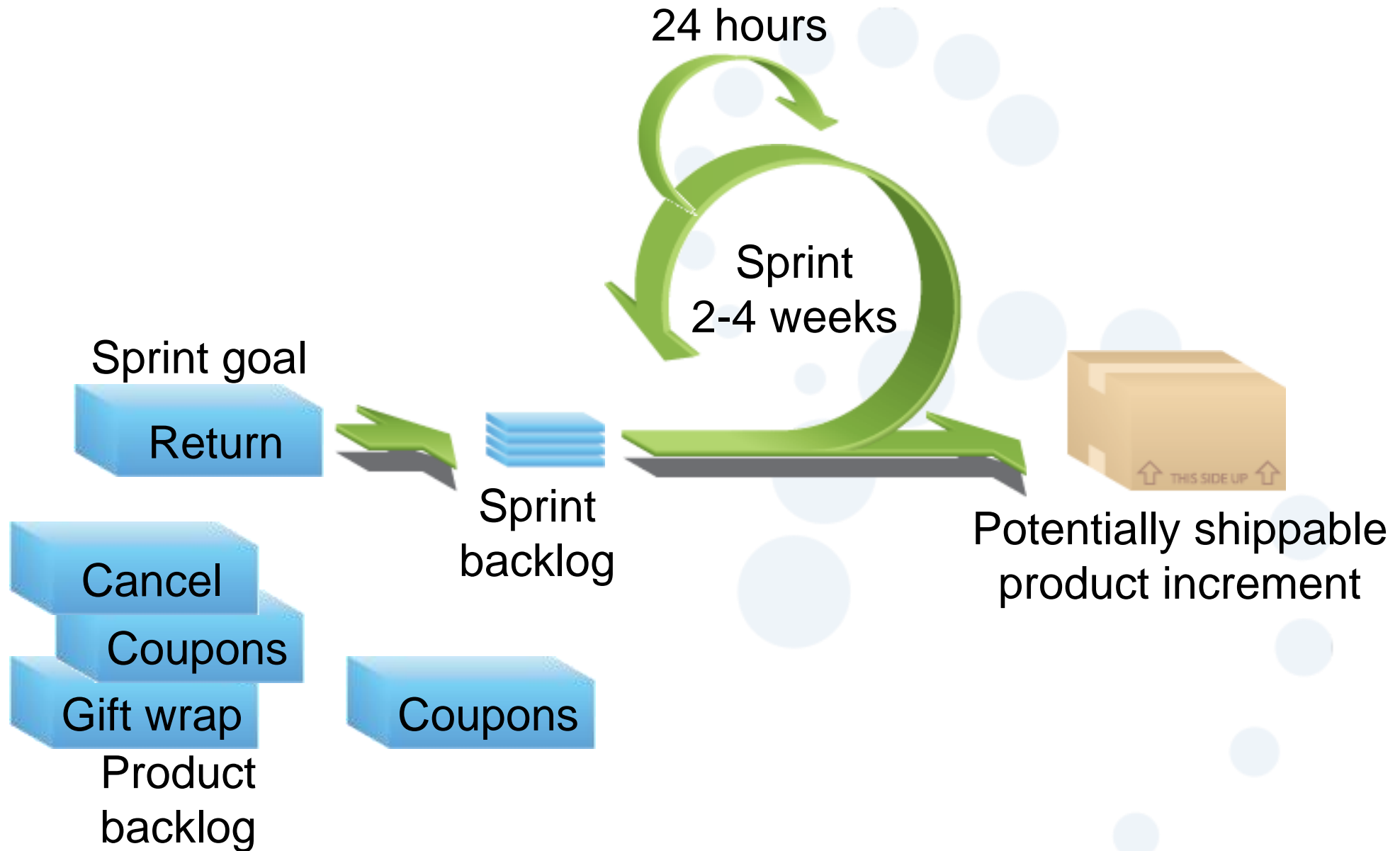
Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

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

That is, while there is value in the items on the right, we value the items on the left more.

Scrum



Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

DETAILS ON PROCESS PARTS

Scrum framework

Roles

- Team
- Product owner
- ScrumMaster

Ceremonies

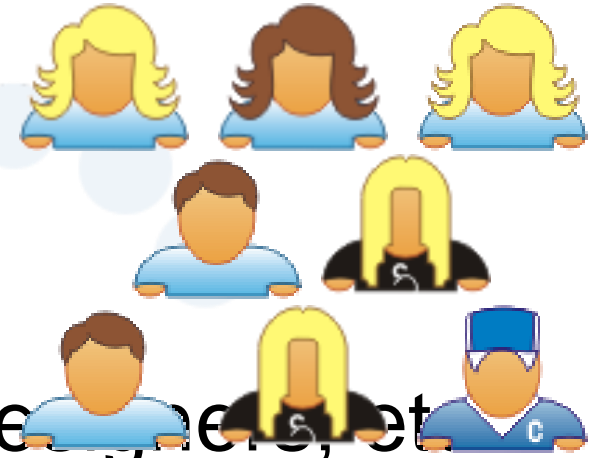
- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

The team

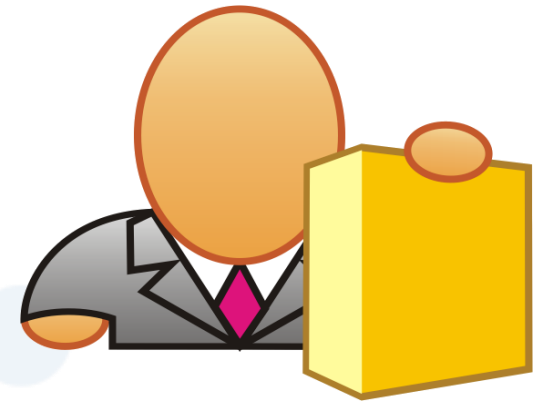
- Cross-functional
 - Programmers, testers, UX designers, etc.
 - Ideally “no titles”
- Self-organizing
- Empowered + responsible
- Ideally full-time members
 - Strive for long-lived teams



The Team



Product owner



- Responsible for the profitability of the product (ROI)
- Define the features of the product
- Decide on release date and content
- Prioritize features according to market value
- Adjust features and priority every iteration, as needed
- Accept or reject work results

The ScrumMaster



- Responsible for enacting Scrum values and practices
- Represents management to the project
- Removes **impediments**
- Ensure that the team is fully functional and **productive**
- Enable close **cooperation** across all roles
- **Shield** the team from external interferences

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

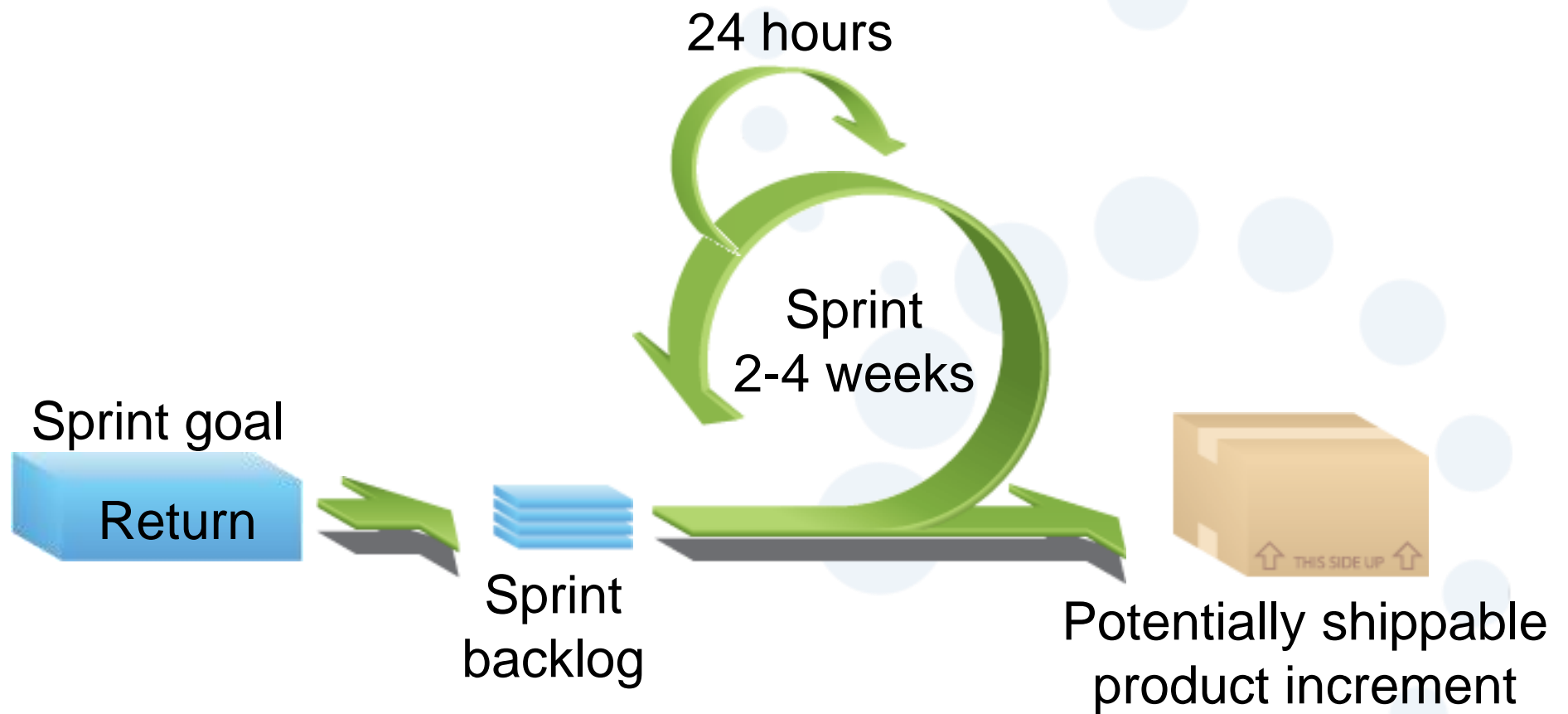
Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

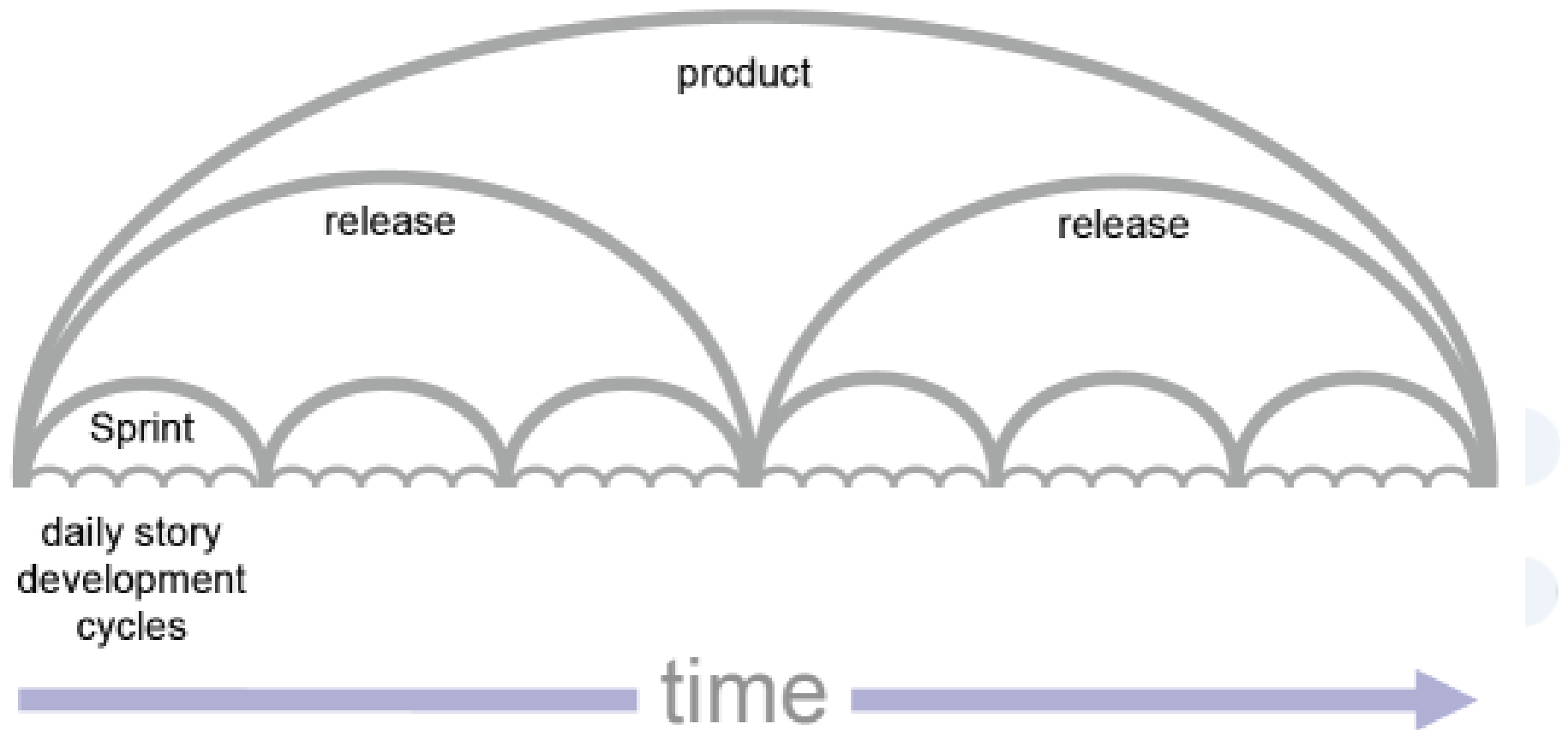
Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Sprints



Product Planning and Scrum



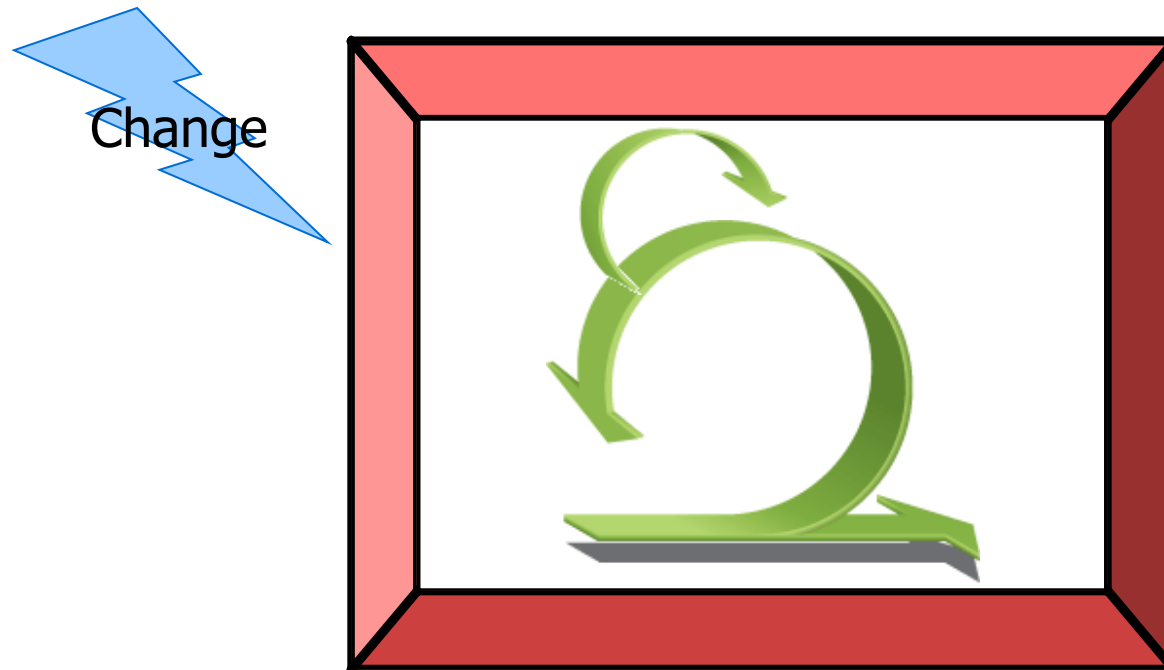
Sprint planning

- **PO explains** product backlog items + Team discusses (**collaboratively** => “Planning game”)
- Sprint backlog is created
 - Items from the product backlog which **Team can commit to** completing
 - **Tasks** are identified, each is estimated (1-16 hours)
- High-level design is considered
- Definition of DONE is needed

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)

No changes during a sprint



- Plan sprint durations around how long you can commit to keeping change out of the sprint

The Daily Scrum

- Parameters

- Daily, 15-minutes
- Stand-up

1 What did you do yesterday?

2 What will you do today?

3 Is anything in your way?

- Not reports to SM

- Synchronization info for peers
- Commitment in front of peers
- Whole world is invited BUT only Team members (plus ScrumMaster, Product Owner) can talk
- Not for problem solving

The Daily Scrum



The Sprint Review

- **Team presents** what it accomplished during the sprint (demo of new features)
- **PO accepts** – Definition of Done, Sprint Goal
- Informal
 - 2-hour prep time rule, No slides
- Whole team participates
- Invite the world

Sprint Review



Sprint retrospective

- Periodically take a look back at **what is and is not working**
- After every sprint
- Whole team
 - ScrumMaster
 - Team
 - + Product owner
 - + Possibly customers and others

Start doing

Stop doing

Continue doing

This is just one
of many ways to
do a sprint
retrospective.

Sprint Retrospective

DIDN'T WORK

- working with documentation - Word as ~~first~~ traceability to dev progress,
 change not dissemination
 sales, marketing etc don't know how (and why) to use Jira+Confluence
 e.g. icons
- planning meetings for DEV redundant - flow of work clear, cycles of (re)work handled regularly in the flow (even on closed tasks, needed on box/worklot/analyst level)
- closing tasks looks needless at first sight → real issue = traceability to reqs
 (mentor: each change request on reqs level should generate 2hr tasks)
 + assign them to all affected reqs + reopen these + some form of regression testing
- vertical slicing & modular development don't work too a large part due to front-end framework (AEM) + SDK x box incompatibilities
- people sync/planning issues ⇒ e.g. complicated reviews
 (exacerbated by multiple projects running in parallel)
 plan ahead according to schedule
 detailed reqs (GUTS mockups)
 ?? many app owner
- estimates not used and verified
 no upper mg't/sales
- UMC → originally expected that deadline will be given and achieved
 very starting + finish

app layers too interrelated ⇒ very large change impact ⇒ how to handle
 - planning
 - testing
 - work parallelism

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

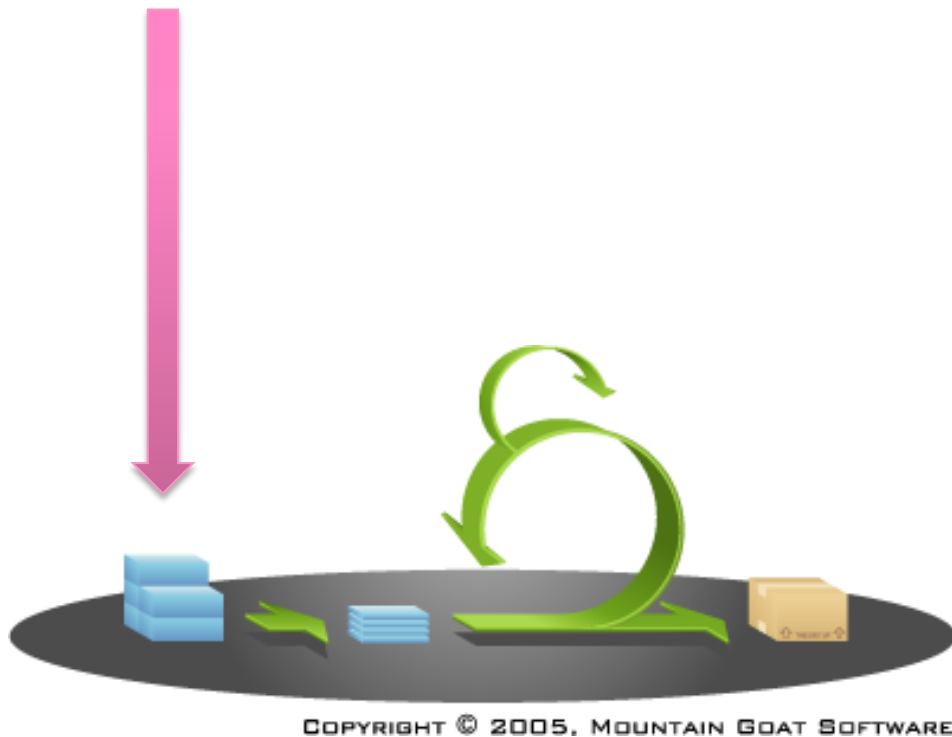
Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts











































- Product backlog
- Sprint backlog
- Burndown charts

Product backlog



- The **requirements**
- A list of all desired work on the project
- Ideally expressed such that **each item has value to the users** or customers of the product
- **Prioritized** by the product owner
- Reprioritized at the start of each sprint

A sample product backlog

Backlog			Create Sprint
82 issues			
	 SWEE-12891 SDK - SIS Booking - DAG API	2 Versions	2d
	 SWEE-9189 [Server / Web] Check for record changes before action (validation) - part 2	Rel-4.0	2w 1d
	 SWEE-12885 SDK - SIS Booking - Business Service API	2 Versions	1w
	 SWEE-11243 [MWB - Web] - Use server function for counting of deposit price for some article	Rel-4.0	1w
	 SWEE-11052 Web: Result set restrictions in overview functions	Rel-4.0	2w 4d
	 SWEE-12825 [MM] - ... - Email start summary - inventory master data export	Rel-4.0	
	 SWEE-10696 Portal - ... - save of positions into count list + small label improvement	Rel-4.0	
	 SWEE-12786 [Server - Report Items without stock] items must be loaded based on ...	Rel-4.0	
	 SWEE-10816 Server - configurable prefixes for ...	Rel-4.0	4d 4h
	 SWEE-2104 Support Main Currency in all Processes and Store Device Control	Rel-4.0	
	 SWEE-6085 Web: Configurable UI - layout 2/2 - Concept	Rel-4.0	
	 SWEE-11732 ...	Rel-4.0	1w
	 SWEE-7228 International Master Data on one system - Control Points	Rel-4.0	
	 SWEE-6539 [Portal] Validation Property (NullQuantityAllowed) hangs up save, 0 is defined as valid (d	4.0.0	1w
	 SWEE-12760 [MWB] Support Sybase Database (ASE)	Rel-4.0	
	 SWEE-10744 [SIS] Properties in Storeman - ... Standard view ... be Displayed in Standard Wa	Rel-4.0	1d
	 SWEE-10746 [SIS] ...	Rel-4.0	1d 4h
	 SWEE-10606 SDK - API JavaDoc description improvement	Rel-4.0	7w 1d
	 SWEE-11772 [Server/Web/Mobile] Support of Storage Locations - new function Stock Transfer	Rel-4.0	4w
	 SWEE-10778 Inventory - Additive registration by scan in quantity field	Rel-4.0	1w
	 SWEE-6192 GUI Portal Tables improvement - Details Viewer (item + supplier)	Rel-4.0	2w



The sprint goal

- A short statement of what the work will be focused on during the sprint

Database Application

Make the application run on SQL Server in addition to Oracle.

Life Sciences

Support features necessary for population genetics studies.

Financial services

Support more technical indicators than company ABC with real-time, streaming data.



The Sprint Backlog

- Plan of the work to be finished during sprint (result of sprint planning)

SWEE Plan Board - MWB 1

SPRINT: Sprint 20 - MWB 1 | QUICK FILTERS: Req only | Server | Mobile | Web | Unresolved | Only My Issues | Recently Updated

In Preparation	Ready for Development	In Progress	Resolved	Tested
SWEE-12314 Basis test - sprint20 2d	SWEE-12634 Sales history error on 0m	SWEE-10670 [MM] Store Number - External Store Number - 0m	SWEE-12282 [Server + Portal] export - display items must be decomposed (property) 0m	SWEE-11021 [MOBS - Update] New configuration for eaninterpreter properties 1h
		SWEE-11515 [Server/Web/Mobile] Support of Storage Locations - solution mode! 5h 34m	SWEE-12812 [SDK] mapping interceptor to ManualPrintRequest wanted 2h 20m	SWEE-10642 [MM] Store Number vs External Store Number in Import - Inventory 3h
		SWEE-12992 SDK - DAO API 1d	SWEE-11551 [MOBS] Support delete flags in 2h	SWEE-10666 [MM] Store Number vs External Store Number - Inventory Export 2m
		SWEE-12887 SDK - Business Service API 1w	SWEE-11550 Support for deleted data (Customer, WorkGroup) 0m	SWEE-11879 [MWB-UI/preddefined inventory] filter - combo box "HK" is senseless 0m
		SWEE-12194 SDK - 1d 5h 21m	SWEE-8029 [SIS] Leading Zeros in SIS Export Configurable 2h	SWEE-12770 Missing SequenceNumber from ShelfMaintenance 1h
		SWEE-12195 SDK - statistics - Business API 3d	SWEE-12248 [MOBS WriteOff] Not possible to choose reason 2h	SWEE-12848 Caches are not called after sync 0m
		SWEE-11514 [Web, Server] New Function Dashboard - new web function 4d 2h 10m	SWEE-11917 SDK - SIS Common - DAO API 0m	SWEE-12862 CLONE - Caches are not called after sync 0m
			SWEE-12908 [MOE] Barcode Service - Improvement 16m	SWEE-11920 [Server - Item Check] error in mwb log for sale history from SIS 0m
				SWEE-12388 [MWB, MOBILE] SalesHistory - missing effectiveTimestamp when loading via SIS 0m

Managing the sprint backlog (1)

- Individuals sign up for work of their own choosing
- Estimated work remaining is updated daily



Managing the sprint backlog (2)

- Any team member can add, delete or change the sprint backlog
 - Work for the sprint emerges
- If work is unclear
 - Define a sprint backlog item with a larger amount of time and break it down later
 - Update work remaining as more becomes known

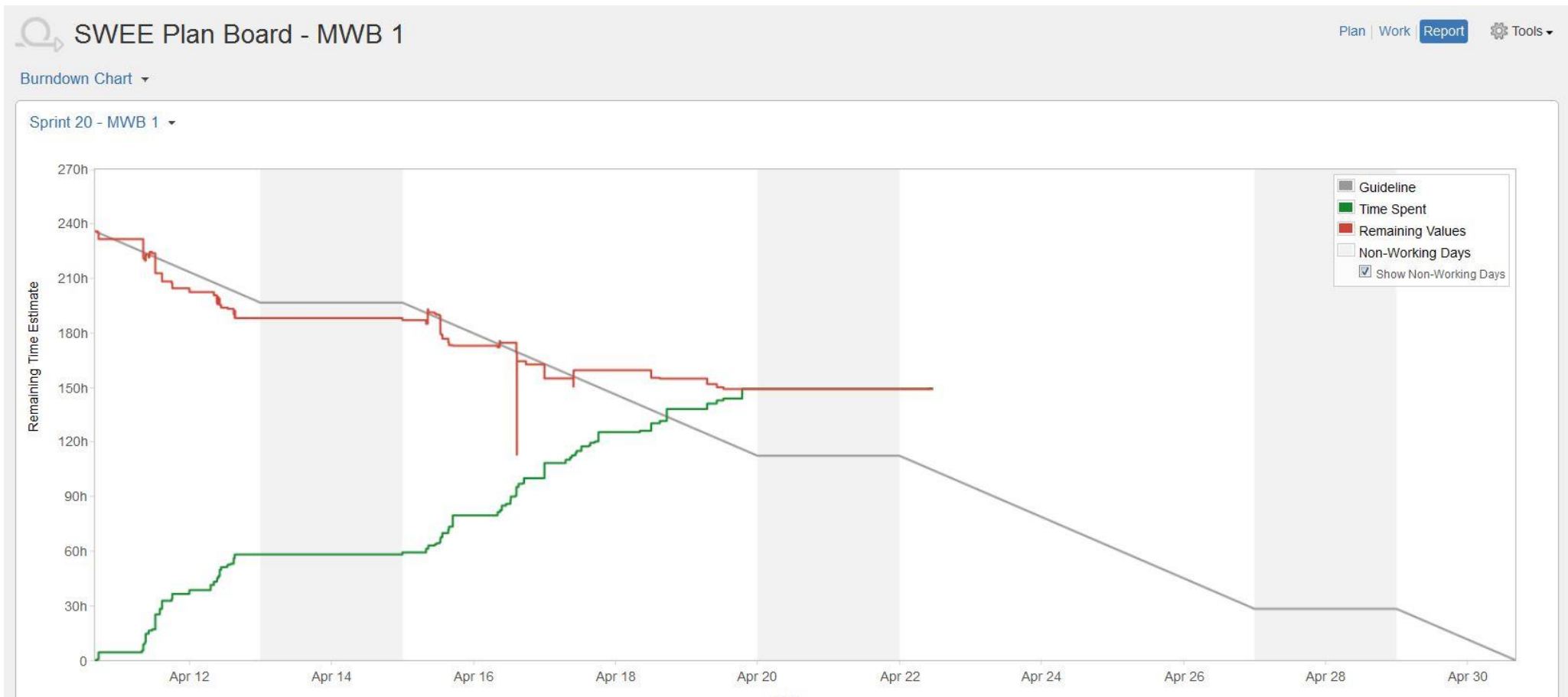


Tracking progress: Burndown

- Simple visual information on work remaining vs time available
- Information radiator

A sprint burndown chart

- Work **remaining vs available** time – simple visual presentation (“information radiator”)



SUMMARY

SCRUM DEVELOPMENT PROCESS

SOFTHOUSE

<http://labs.openviewpartners.com/what-is-scrum-infographics/4/>

1 Product Backlog

- List of requirements and issues
- Owned by Product Owner
- Anybody can add items
- Only Product Owner prioritizes

Backlog Estimation

- Product Owner describes new items
- All discuss
- Team estimates

Product Owner updates **Product Backlog**

2 Sprint Planning

- Declare Sprint Goal
- Select highest priority items from **Product Backlog** that supports the Sprint Goal
- Team turns selected items into **Sprint Backlog** and estimates them

3 Sprint Backlog

- List of tasks
- Owned by Scrum Team
- Only Scrum Team modifies it



4 Daily Scrum

Same time every day

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

Team updates **Sprint Backlog**
Scrum Master updates Impediment List & Sprint Burndown Chart



5 Sprint Review

Sprint Demonstration

- Team demos **Increment**
- All discuss
- Announce next **Sprint Planning Meeting**

Sprint Retrospect

Team discusses the sprint to answer the following:

- What should we keep doing?
- What should we stop doing?
- What should we start doing?

6 Increment

- Version of the product
- Potentially shippable (tested, documented etc)

Product Owner

Sets priorities and manages the product backlog



Scrum Master

Manages the process and removes impediments



Scrum Team

Develops the product and self-organizes



FURTHER TOPICS

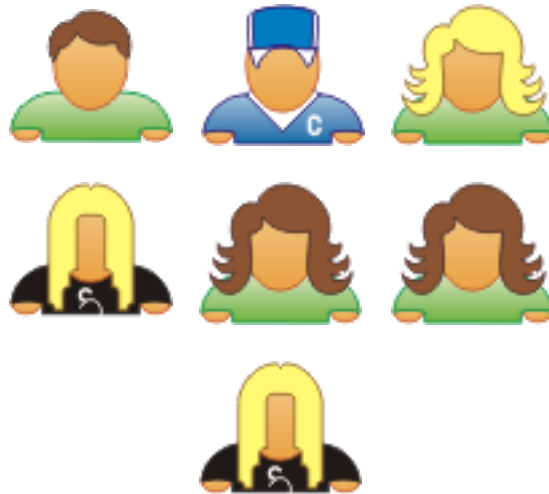
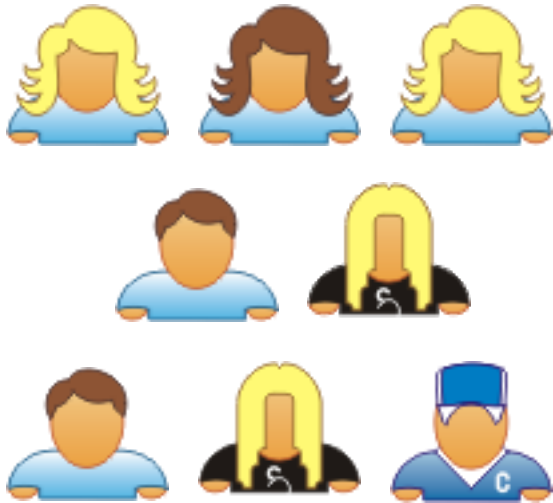
Distributed Teams

- Not ideal
- But possible
- Backlog in online tools
- Meetings via voice/video conference
- Communicate a lot (include “small talk”)
- Document more

Scalability

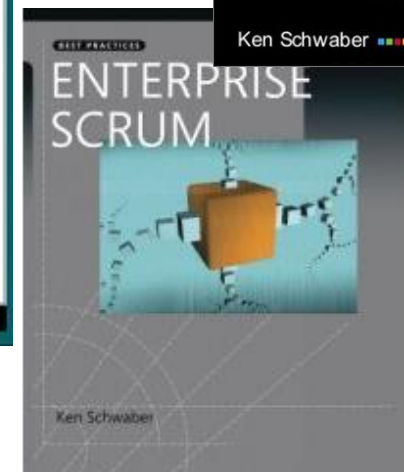
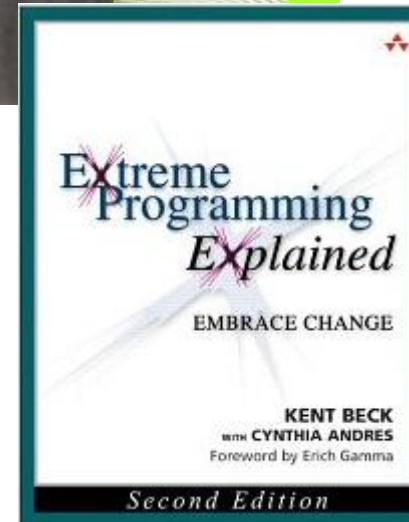
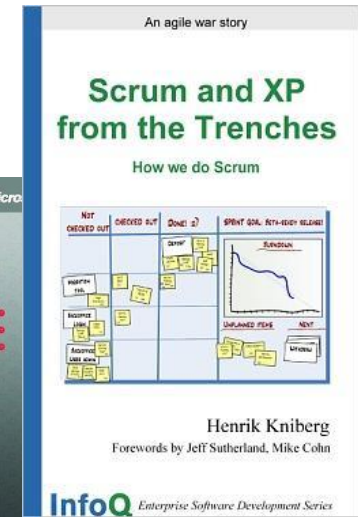
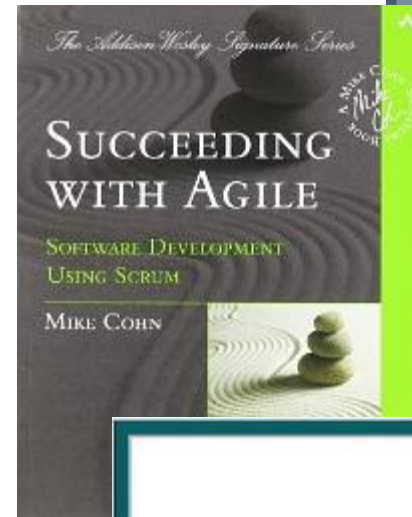
- Typical individual team is 7 ± 2 people
 - Scalability comes from **teams of teams**
- Factors in scaling
 - Type of application
 - Team size
 - Team dispersion
 - Project duration
- Scrum has been used on multiple 500+ person projects

Scaling through the Scrum of scrums



Names and Books

- Jeff Sutherland
- Ken Schwaber
- Mike Cohn
- Craig Larman
- Henrik Kniberg



Thank You

Presentation by:

Mike Cohn <mike@mountaingoatsoftware.com>

Premek Brada <brada@kiv.zcu.cz>