26 May 2010 CQAA Lunch & Learn
Paul I. Pazderski (CSM/CSP, OD-CM, CSQA)
spcinc13@yahoo.com Cell: 224-595-8846

SOFTWARE
PROCESS
CONSULTANT INC.

AGILE THROUGH SCRUM

AGENDA & LEARNING POINTS

1. Open
2. Agile Overview
3. Scrum Basics   *Learning Points*
4. How-to trial Scrum
5. Close

Present   *Discuss*

AGILE MANIFESTO -- **4 AGILE VALUES**

We are uncovering better ways of developing

software by doing it and helping others do it.

Through this work we have come to <u>value</u>:

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

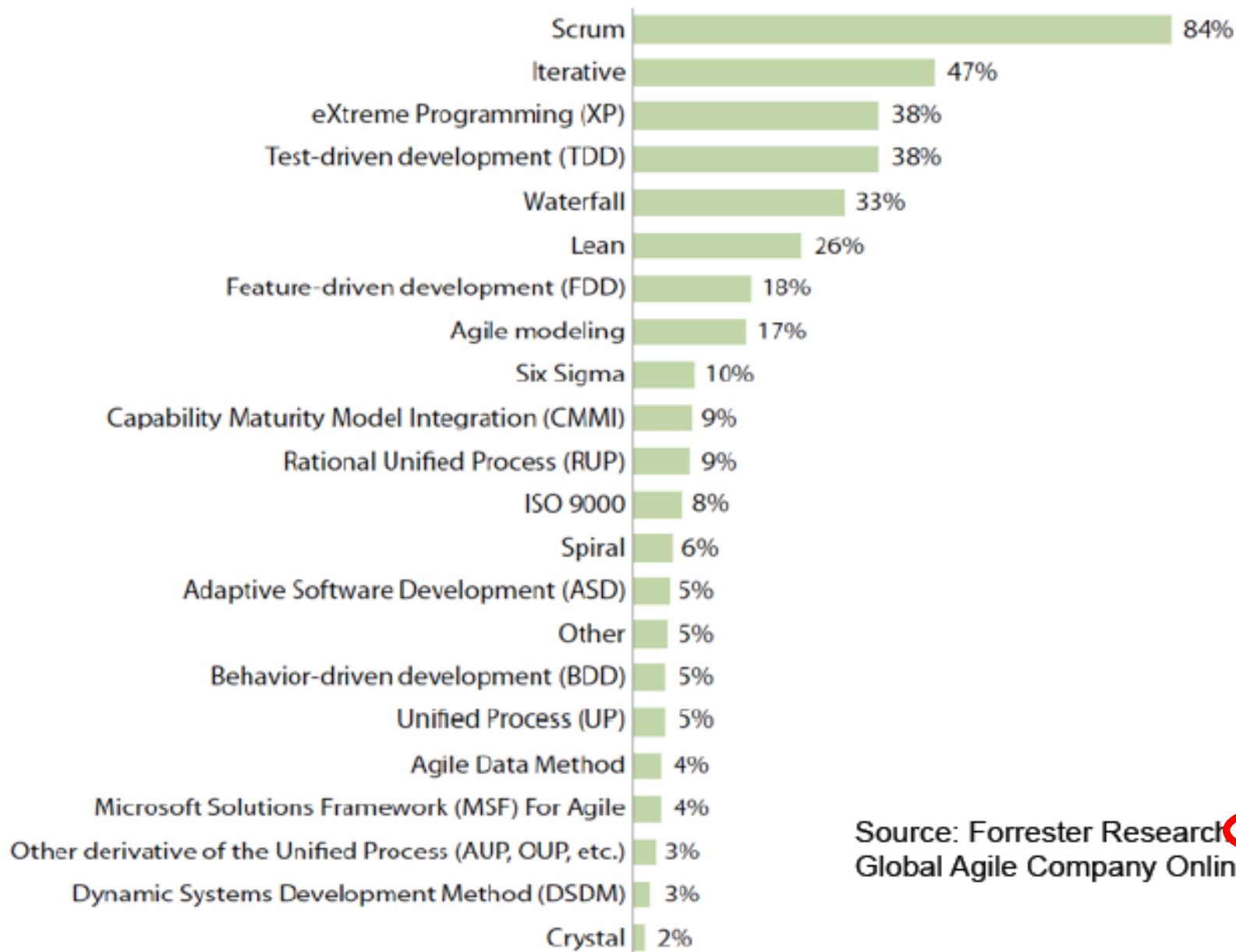the right, we value the items on the left more.

(12) PRINCIPLES BEHIND THE AGILE MANIFESTO

1. Our highest priority is to **satisfy the customer** through **early and continuous** delivery of valuable software.
2. **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
3. **Deliver** working software **frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. **Business** people and **developers** must **work together daily** throughout the project.
5. Build projects around **motivated individuals**. Give them the environment and support they need, and **trust** them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
7. **Working software** is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a **constant pace indefinitely**.
9. Continuous attention to **technical excellence** and **good design** enhances agility.
10. **Simplicity-**-the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from **self-organizing** teams.
12. At regular intervals, the **team reflects** on how to become more effective, then **tunes and adjusts** its behavior accordingly.
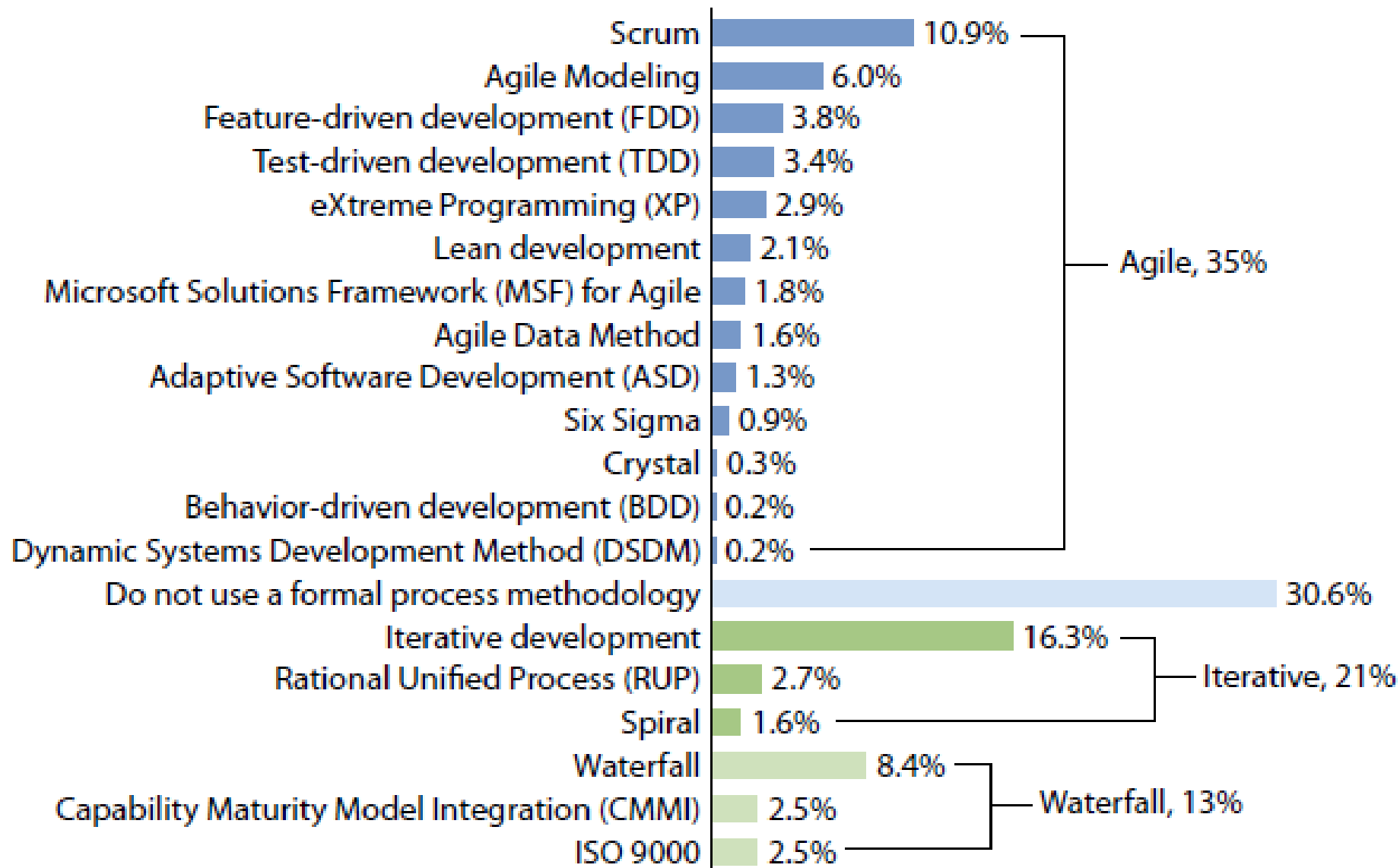
ARE YOU A AGILE MANIFASTO SIGNATORY?

* 8826 Signatories as of: 21 April 2010

* From all over the world

* Anyone can add their info

* Please join at: http://agilemanifesto.org/

* … its free!

Techniques or Methodologies Used

Technique	Percentage
Scrum	84%
Iterative	47%
eXtreme Programming (XP)	38%
Test-driven development (TDD)	38%
Waterfall	33%
Lean	26%
Feature-driven development (FDD)	18%
Agile modeling	17%
Six Sigma	10%
Capability Maturity Model Integration (CMMI)	9%
Rational Unified Process (RUP)	9%
ISO 9000	8%
Spiral	6%
Adaptive Software Development (ASD)	5%
Other	5%
Behavior-driven development (BDD)	5%
Unified Process (UP)	5%
Agile Data Method	4%
Microsoft Solutions Framework (MSF) For Agile	4%
Other derivative of the Unified Process (AUP, OUP, etc.)	3%
Dynamic Systems Development Method (DSDM)	3%
Crystal	2%

Source: Forrester Research December 2008
Global Agile Company Online Survey

Base: 241 technology industry professionals in a variety of roles, including but not limited to development (numbers have been rounded)

"Please select the methodology that most closely reflects the development process you are currently using."
(select only one)

Methodology	%
Scrum	10.9%
Agile Modeling	6.0%
Feature-driven development (FDD)	3.8%
Test-driven development (TDD)	3.4%
eXtreme Programming (XP)	2.9%
Lean development	2.1%
Microsoft Solutions Framework (MSF) for Agile	1.8%
Agile Data Method	1.6%
Adaptive Software Development (ASD)	1.3%
Six Sigma	0.9%
Crystal	0.3%
Behavior-driven development (BDD)	0.2%
Dynamic Systems Development Method (DSDM)	0.2%

Agile, 35%

Do not use a formal process methodology	30.6%

Methodology	%
Iterative development	16.3%
Rational Unified Process (RUP)	2.7%
Spiral	1.6%

Iterative, 21%

Methodology	%
Waterfall	8.4%
Capability Maturity Model Integration (CMMI)	2.5%
ISO 9000	2.5%

Waterfall, 13%

Base: 1,298 IT professionals

Source: Forrester/Dr. Dobb's Global Developer Technographics® Survey, Q3 2009

24 h

30 days

Product Backlog   Sprint Backlog   Sprint   Working increment of the software
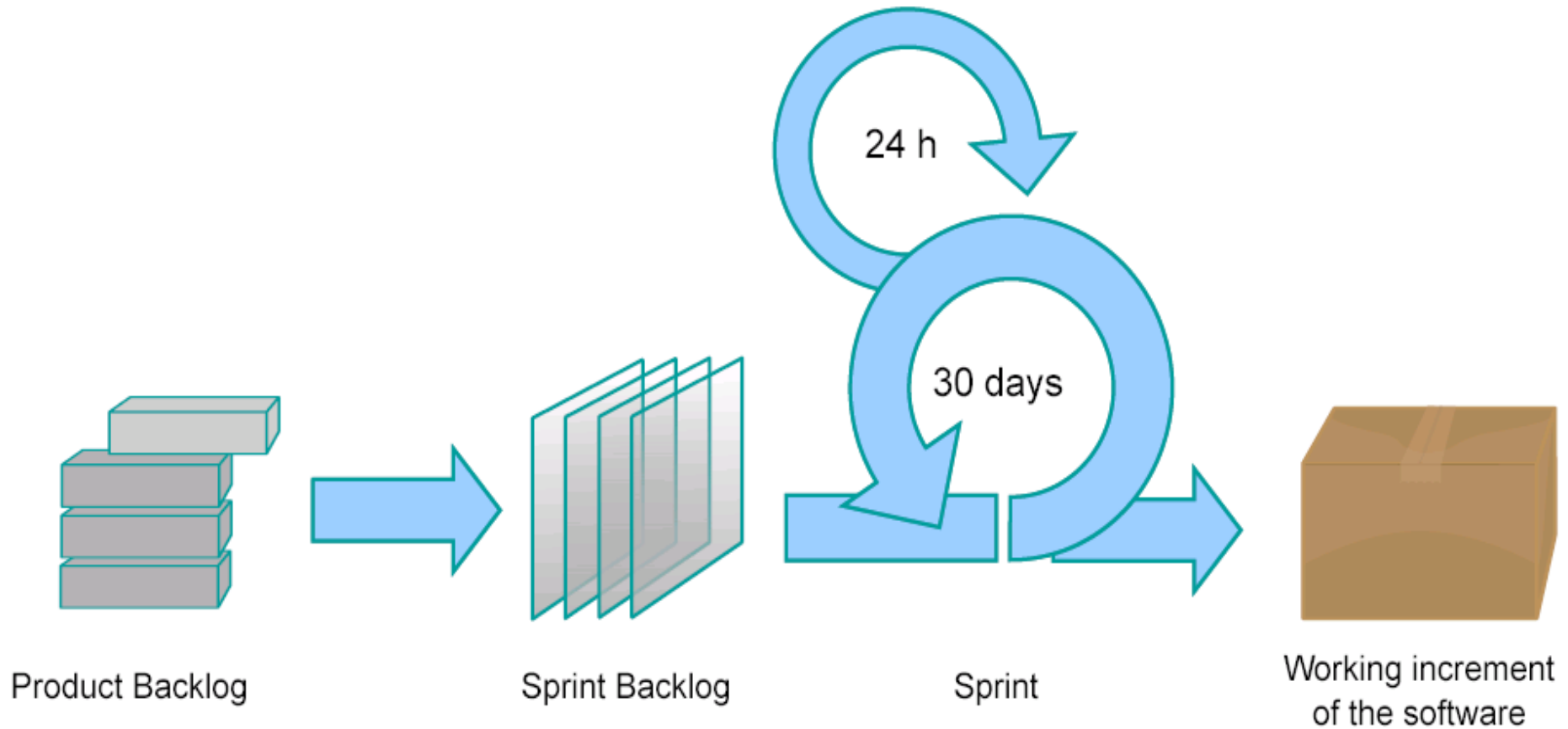
SCRUM BASICS – 3 X 3

✖ **3 Roles**

+ Product Owner
+ Scrum Master
+ Scrum Team

✖ **3 Artifacts**

+ Product Backlog
+ Sprint Backlog
+ Burndown Chart(s)

✖ **3 Activities**

+ Sprint Planning
+ Daily Scrums
+ Sprint Review

Product Backlog → Sprint Backlog → Sprint (24 h, 30 days) → Working increment of the software
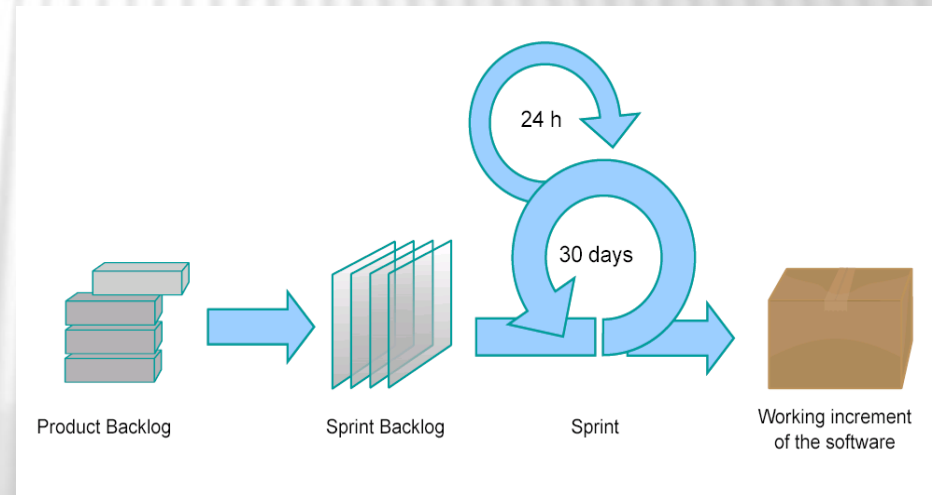
EXAMPLE – **MOVING**

Project: Family moving
	from one home to another
- **Product Owner**: Mother
- **Scrum Master**: Daughter
- **Scrum Team**: entire Family
- **Product Backlog**: all Items
	1. Prep & Organize
		a. *Subject matter research*
		b. *Option selection*
		c. *Move Budget*
	2. Buy new Home
	3. Sell old Home
	4. Move from old to new

Items: Activities & Deliverables
Sprints: 10 at 2 weeks each

Product Owner Sprint Planning Product Backlog
Scrum Master   Daily Scrums   Sprint Backlog
Scrum Team   Sprint Review   Burndown Chart(s)

24 h

30 days

Product Backlog   Sprint Backlog   Sprint   Working increment of the software
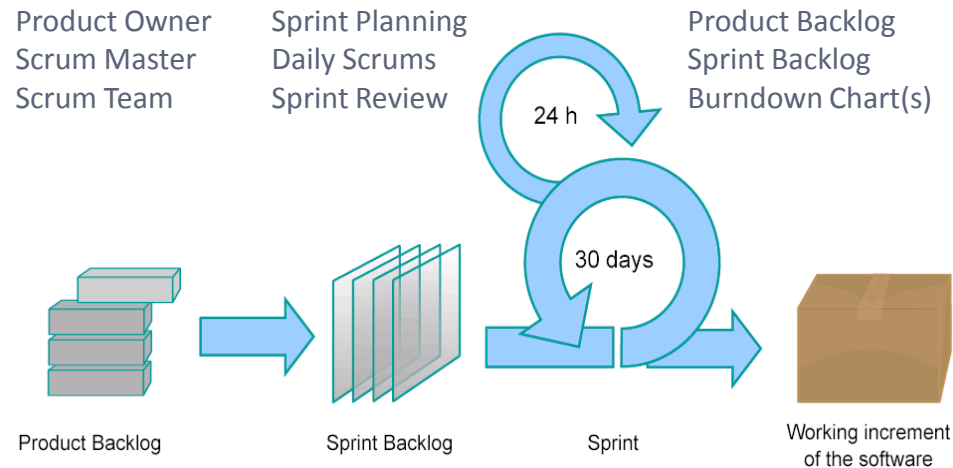
- **Sprint Backlog**: subset of Items
For example "Options" Tasks :
	1) Identify Options for Buying
	2) Identify Options for Selling
	3) Identify Options for Moving
	4) Estimate, compare, & choose best options

EXAMPLE – **MOVING**

- **Sprint Planning**:
Pre-Planning: Sprint-0 vs. Sprint-1
Id Product Backlog, Items, Sprints
Roles … all the stuff here!
Regularly: estimate, prioritize,
allocate & adjust.

Product Owner   Sprint Planning   Product Backlog
Scrum Master   Daily Scrums   Sprint Backlog
Scrum Team   Sprint Review   Burndown Chart(s)

24 h

30 days

Product Backlog   Sprint Backlog   Sprint   Working increment of the software
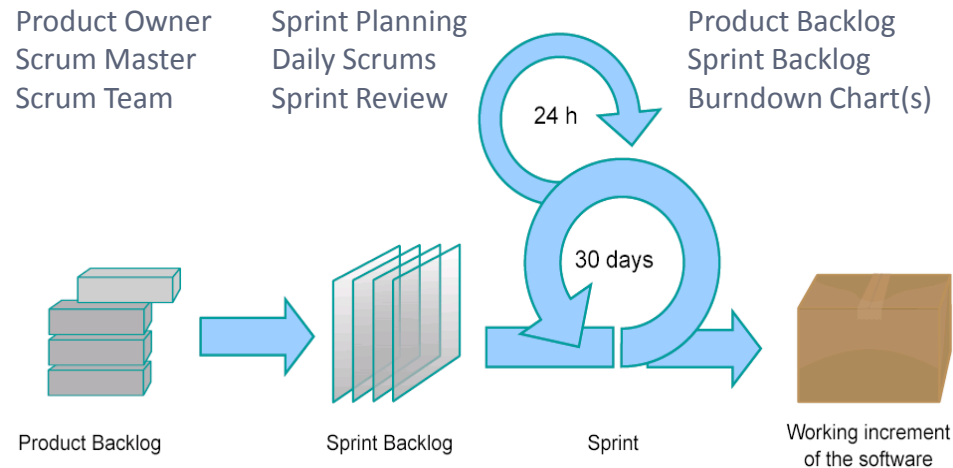
- **Daily Scrums**: at breakfast
Ask 3 Scrum Questions of everyone:
  1. *What did you do yesterday?*
  2. *What will you do today?*
  3. *What issues do you face?*
Each person takes Tasks for next day.
Scrum Master updates Burndown
Chart.

- **Burndown Chart**: track Tasks
Estimate 100 total.
10 per Sprint.
2 lines: estimated & actual

- **Sprint Review**: See how's it going?
Review done Items & their Tasks.
Decide changes on next Sprint.

EXAMPLE – **BURNDOWN CHART**
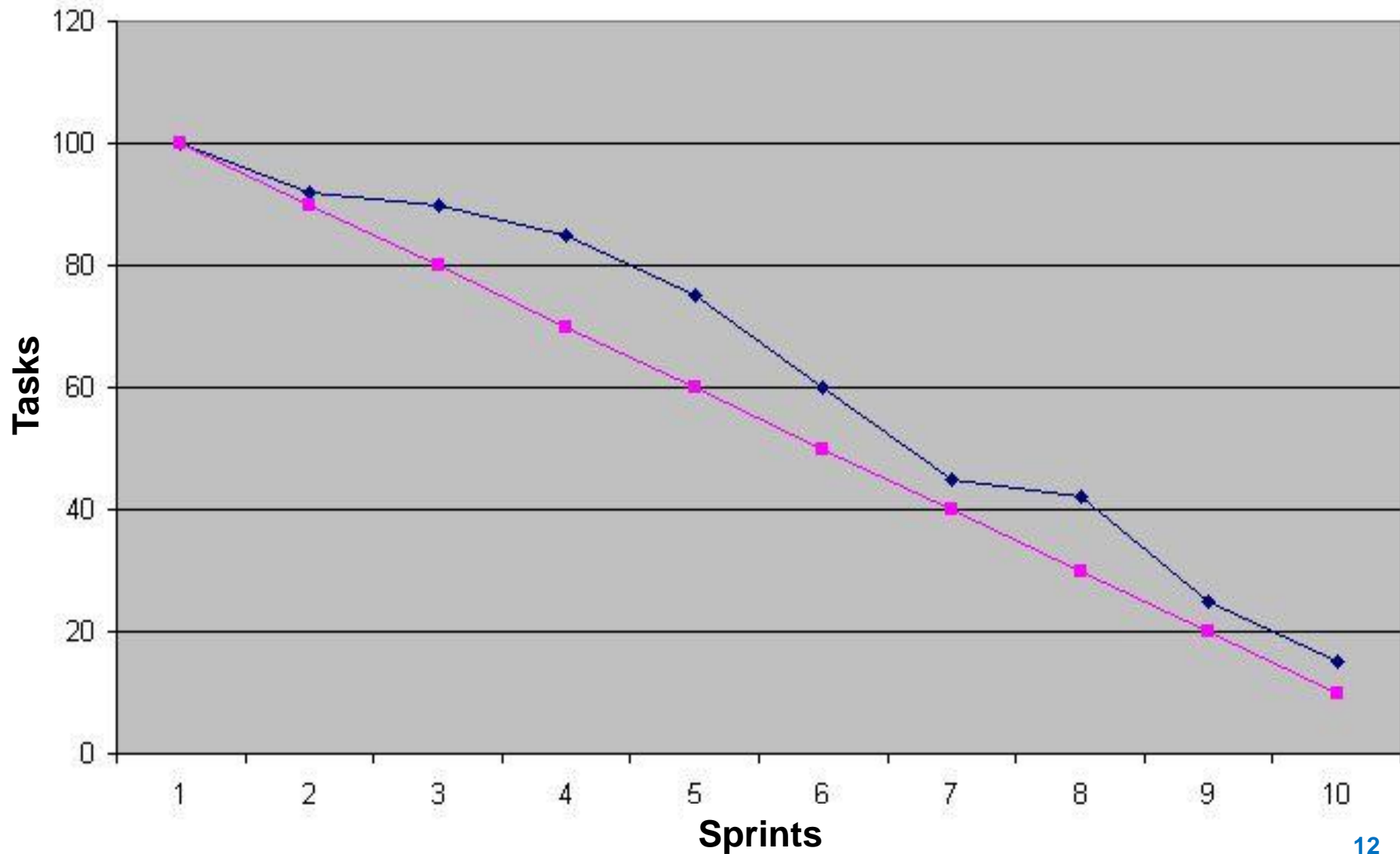
Sprints

SCRUM – GLOSSARY (MORE TERMS)

Impediment: Issue, Problem, Block

Work Increment: Sprint Goal
Working part of end-product usable
by Customer; software, manuals,
configuration, process, etc.

Scrum Board: Location to manage
Project info (physical or electronic)

Velocity: features (tasks or points)
per sprint

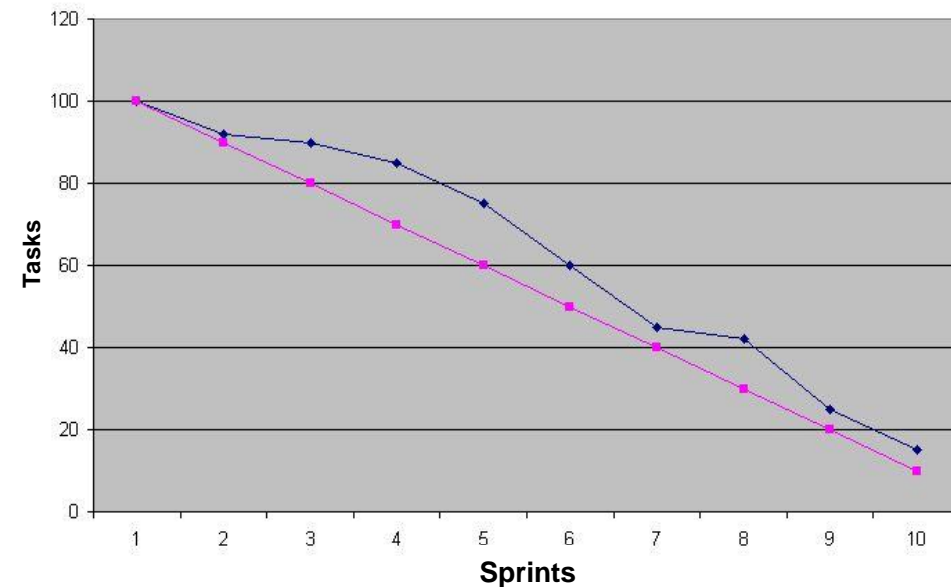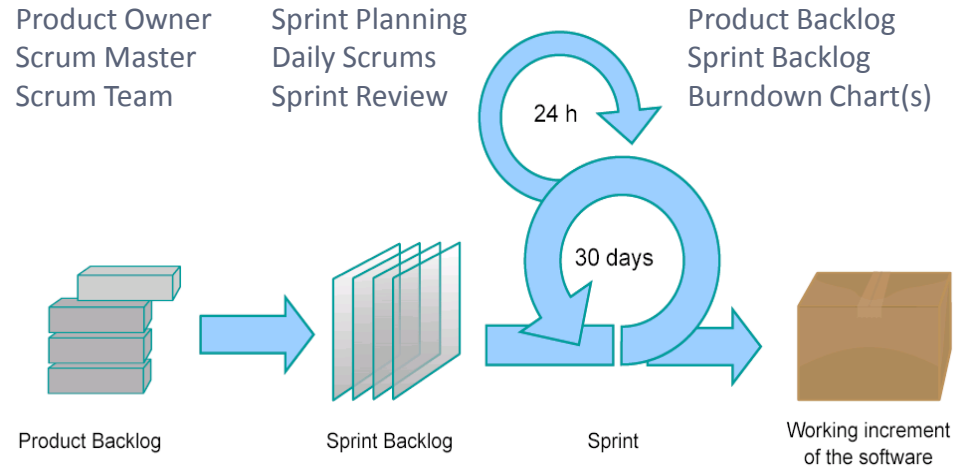**Abnormal Termination**: cancel
Sprint if unable to deliver & re-plan

Guidelines: improve, adapt, &
change practices as needed.

Product Owner	Sprint Planning	Product Backlog
Scrum Master	Daily Scrums	Sprint Backlog
Scrum Team	Sprint Review	Burndown Chart(s)

24 h

30 days

Product Backlog   Sprint Backlog   Sprint   Working increment of the software
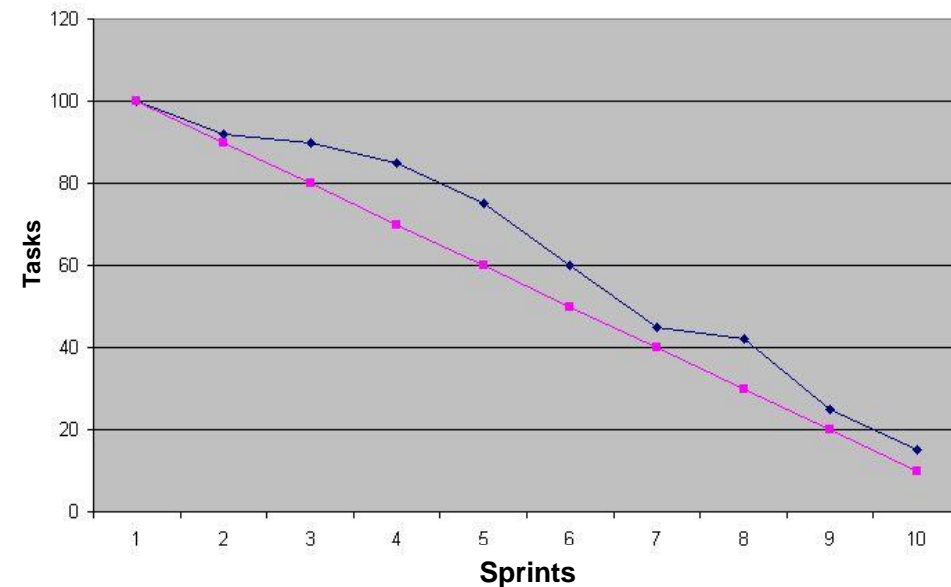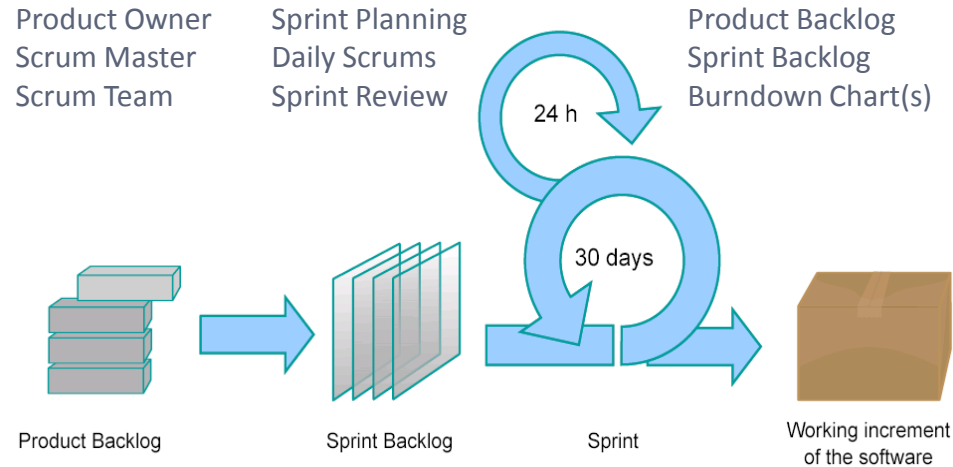
SCRUM – GLOSSARY (MORE TERMS)

Product Owner   Sprint Planning   Product Backlog
Scrum Master   Daily Scrums   Sprint Backlog
Scrum Team   Sprint Review   Burndown Chart(s)

- Product Burndown Chart
- Sprint Burndown Chart
- Release Burndown Chart
- Estimation – Planning Poker
- Release Plan
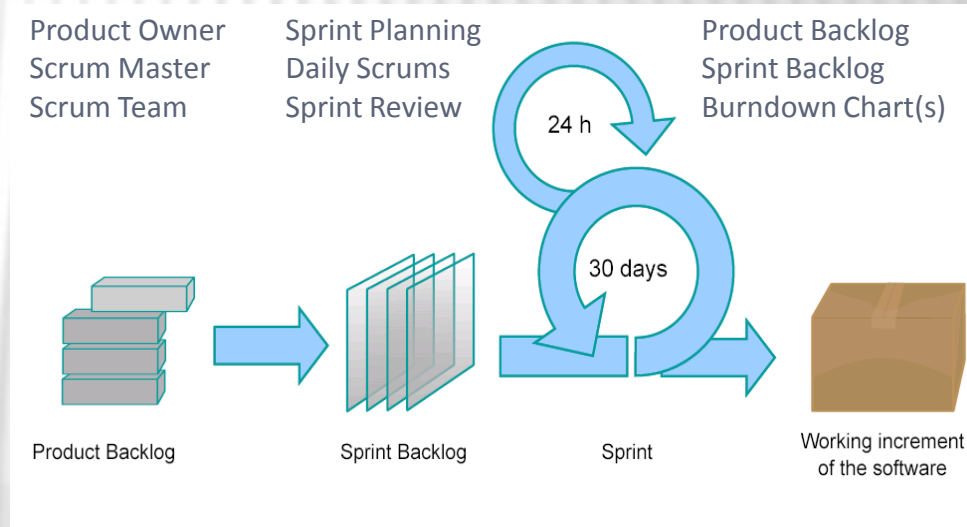- "Done" criteria
- Story Cards
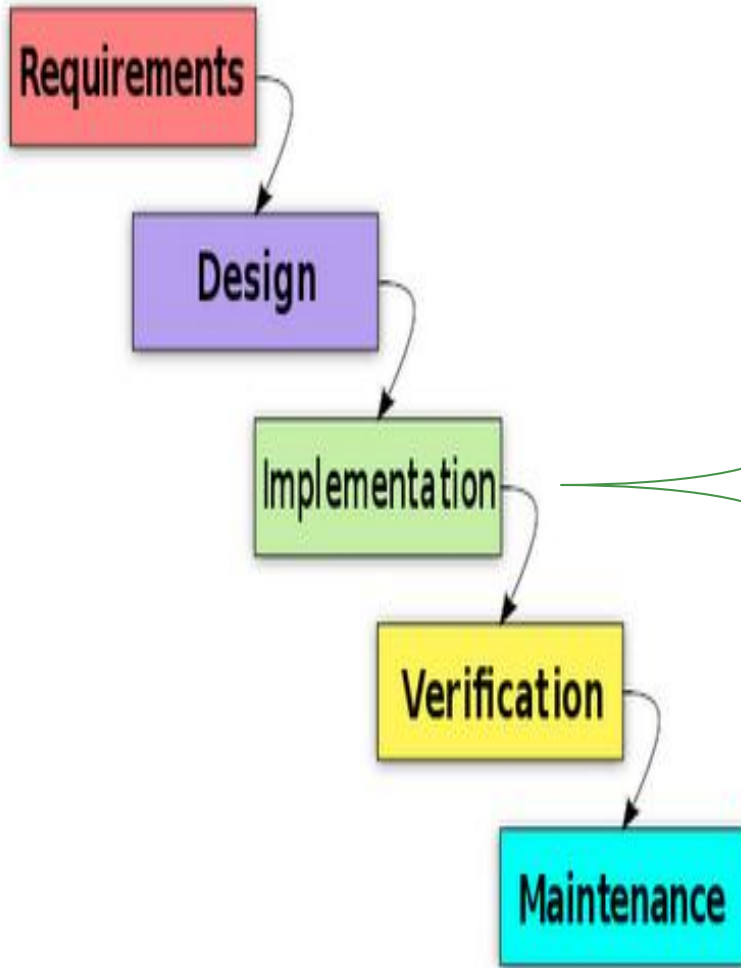- Spikes
- Epics
- Just enough

… what's your favorite?

Product Backlog   Sprint Backlog   Sprint   Working increment of the software

24 h

30 days

Tasks (y-axis: 0, 20, 40, 60, 80, 100, 120)
Sprints (x-axis: 1–10)

TRY SCRUM – **SELECT PHASE**
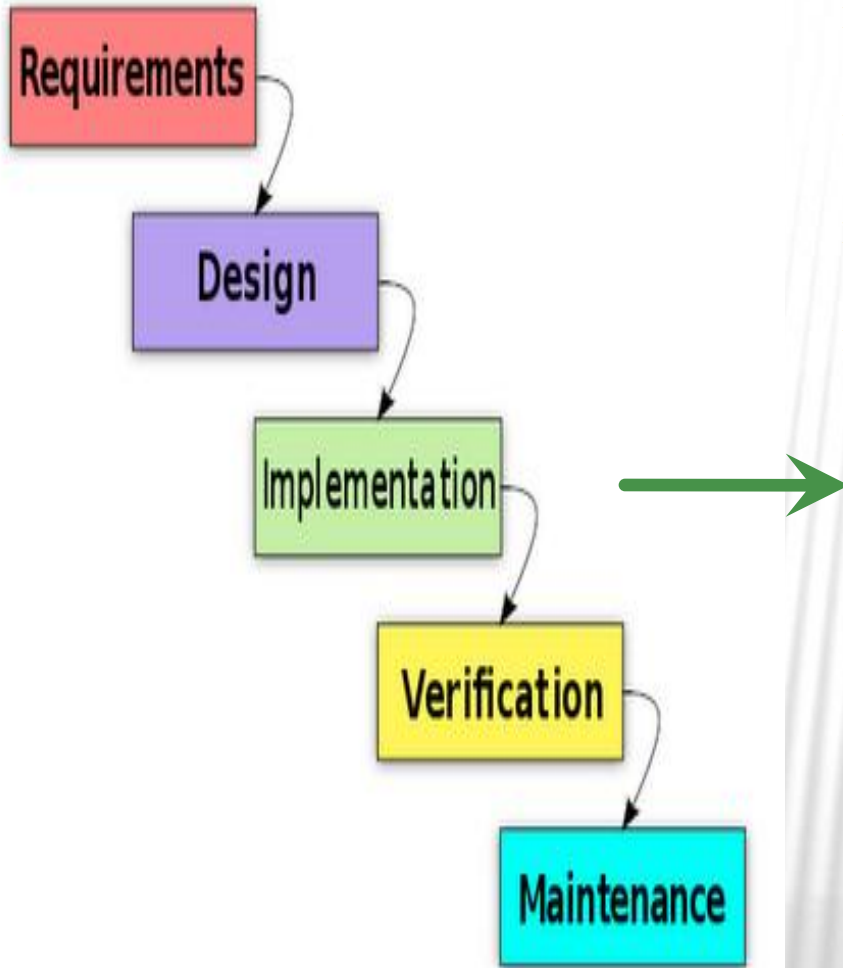
* Trial in controlled environment
* Select stable phase
* Select well-defined phase
* Map Phase to Scrum – Minimal

Requirements

Design

Implementation

Verification

Maintenance

Product Owner
Scrum Master
Scrum Team

Sprint Planning
Daily Scrums
Sprint Review

Product Backlog
Sprint Backlog
Burndown Chart(s)

24 h

30 days

Product Backlog

Sprint Backlog

Sprint

Working increment
of the software

TRY SCRUM – **MAP THE 3 ROLES**

Requirements

Design

Implementation

Verification

Maintenance

✖ Product Owner ➔ maybe
+ Real Client (best)
+ Client Rep / BA
+ Product Manager
+ Project Manager

✖ Scrum Team
+ Workers, directly working in Phase
+ Everyone else ➔ silent observers

✖ Scrum Master
+ You (for starters)
+ Let Scrum Team choose (later)

TRY SCRUM – **MAP THE 3 ARTIFACTS**

Requirements

Design

Implementation

Verification

Maintenance

- ✖ Product Backlog ➜

  List Phase Deliverables

  (Only Mandatory)

  - + Implementation Phase Plan(s)
  - + Implementation Report(s)
  - + Detailed Design(s)
  - + Interface Definition(s)
  - + Code Module(s)
  - + Unit Test(s)
- ✖ Prioritize & Guesstimate

    Deliverables

TRY SCRUM – **MAP THE 3 ARTIFACTS**

Requirements

Design

Implementation

Verification

Maintenance

- ✖ Sprint Backlog ➔

  List Common Tasks

  (Only Mandatory)

  - ✚ Draft Deliverable
  - ✚ Peer-Review Deliverable
  - ✚ Rework/Finalize Deliverable
  - ✚ Approve/Publish Deliverable

- ✖ Prioritize & Guesstimate Tasks
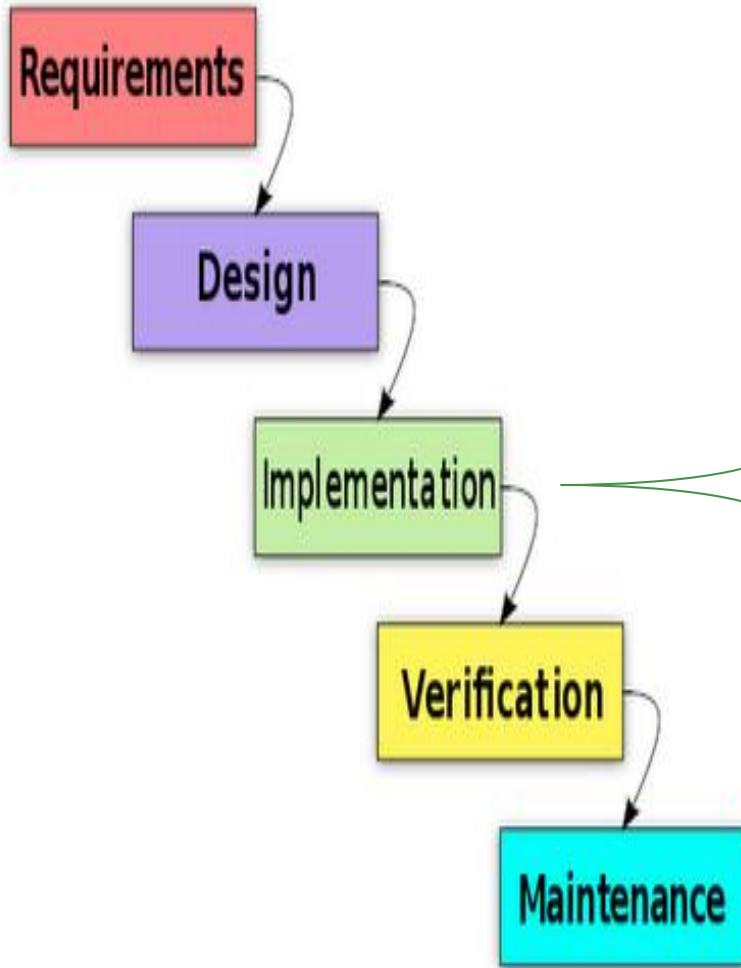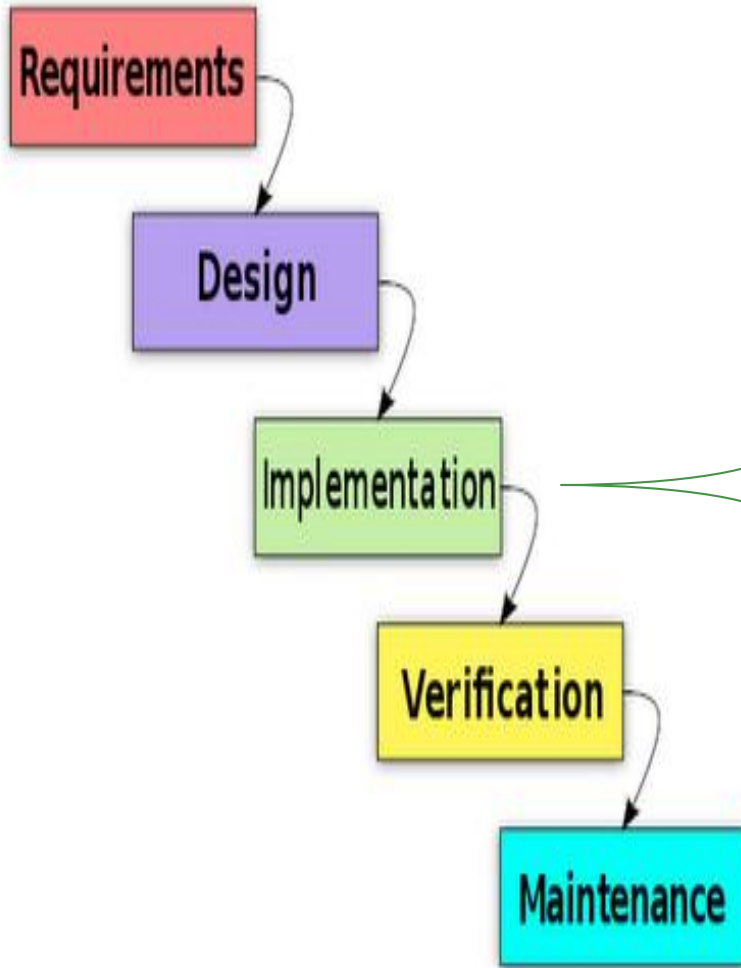
× Burndown Chart

+ Decide which Burndown to try
  × Project/Phase Burndown
  × Sprint Burndown

+ Consider if Burndown can replace any traditional Report(s)

TRY SCRUM – **MAP THE 3 ACTIVITIES**

✖ Sprint Planning (1/2)

Pre-Planning - one time:

+ Consider Sprint-0 vs. Sprint-1

+ Map Scrum Roles

+ Id Product Backlog (Deliverables)

+ Prioritize Deliverables (rank order)

+ Guesstimate Deliverables

+ Id Tasks

+ Id Sprints - e.g. 6 at 2 weeks each

+ Id Time & Place for Daily Scrums

✖ Sprint Planning (2/2)

Regular Planning – each Sprint:

+ Select Sprint Backlog (Guesstimate)
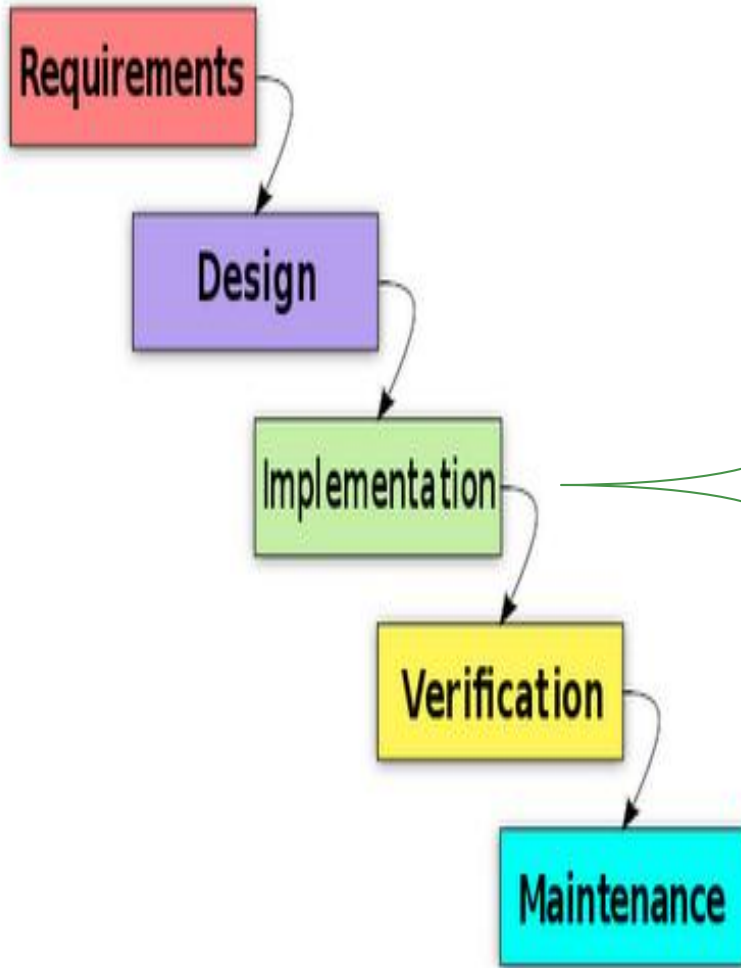+ Prioritize All Tasks (Rank order)
+ Guesstimate Specific Tasks (adjust)

Requirements → Design → Implementation → Verification → Maintenance

✖ Daily Scrums (<15min!)

+ Every Scrum Team member answers 3 Scrum Questions:
  1. What did you do yesterday?
  2. What will you do today?
  3. What issues do you face?

+ Members pick Tasks for next day

+ Scrum Master updates Burndown
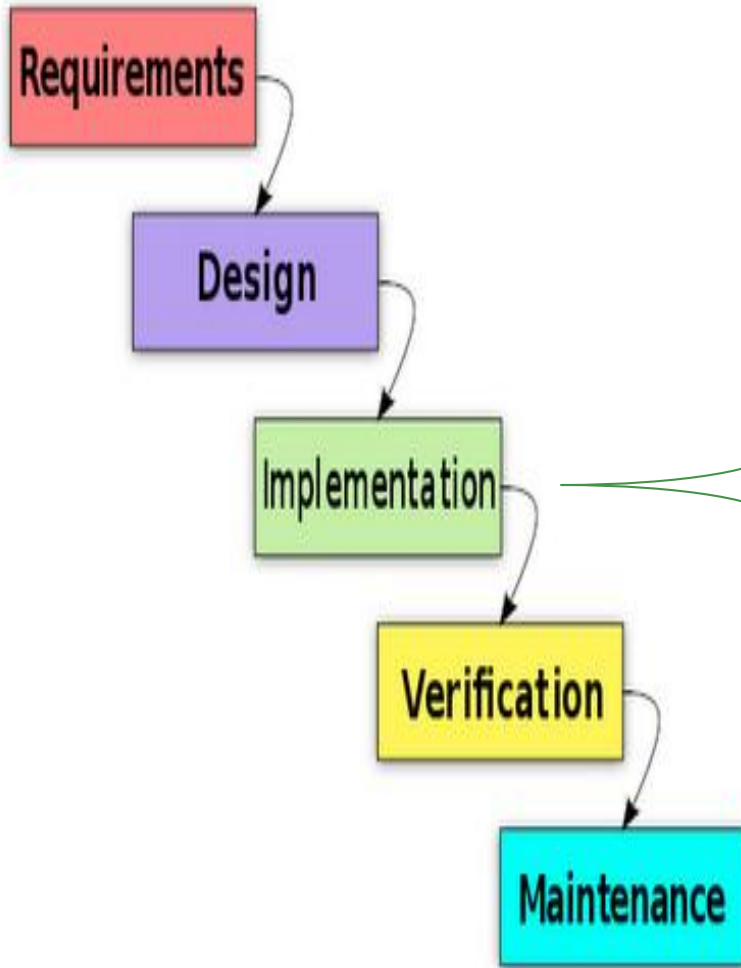
TRY SCRUM – **MAP THE 3 ACTIVITIES**

Requirements

Design

Implementation

Verification

Maintenance

✖ Sprint Review (1/2)

The Demo – Team with Client(s)

+ Present Deliverable(s)

+ Review progress

+ Preview next-steps

+ Get feedback – pros & cons

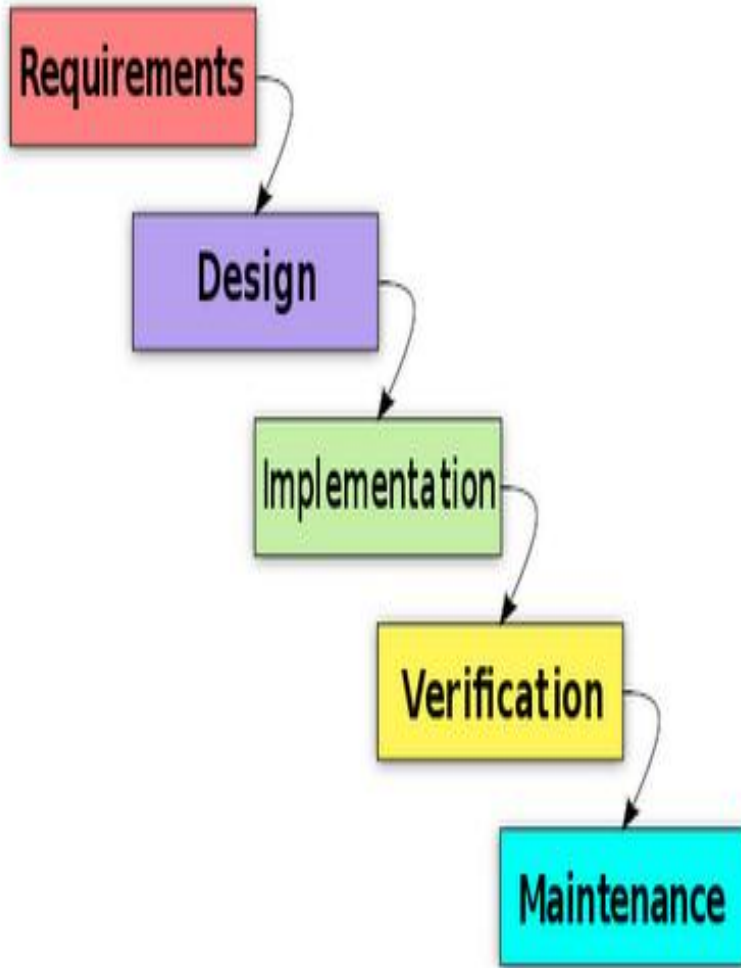**✖ Sprint Review (2/2)**

The Retrospective – just the Team

+ Review progress
+ Review Client feedback
+ Identify improvements:
  × What to keep doing
  × What to stop doing
  × What to start doing

TRY SCRUM – **EXPANDING & ALTERNATES**

× Add a phase

  + Design   or   Verification

× Try another phase

  + Requirements or Maintenance

× Try an IT department

  + QA   CM   PMO   …

× Try a Business department

  + HR   BA   Sales   Marketing

AGILE & SCRUM – **MORE INFO**

- ✖ Scrum Alliance

http://www.scrumalliance.org/

- ✖ Agile Alliance

http://www.agilealliance.org/

- ✖ APLN Chicago (Agile Project Leadership Network)

http://www.aplnchicago.org/

- ✖ Paul I. Pazderski (Software Process Consultant Inc.)

spcinc13@yahoo.com   Cell: 224-595-8846

Have fun with SCRUM!