

The background of the slide is a grayscale aerial photograph of the University of Porto's campus, showing numerous modern and traditional buildings, green spaces, and infrastructure.

M.EIC, 2022-23

Large Scale Software Development

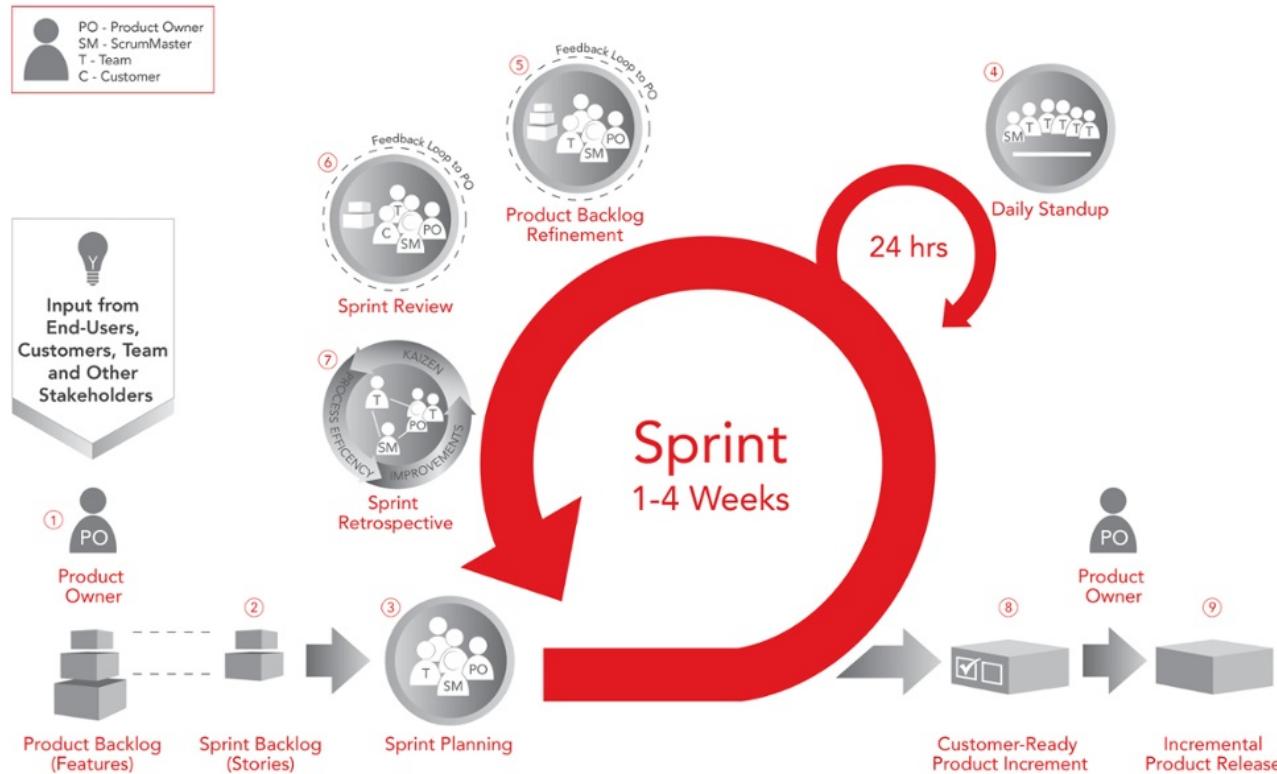
Filipe Correia, Daniel Pinho, João Pedro Dias



Scrum Patterns

A set of small, semi-transparent navigation icons located in the bottom right corner of the slide. They include symbols for back, forward, search, and other presentation controls.

Scrum





Jeff Sutherland
James O. Coplien
The Scrum Patterns Group
edited by Adaobi Obi Tulton

<https://pragprog.com/book/jcscrum/a-scrum-book>

A Scrum Book
The Spirit of the Game

Jeff Sutherland
James O. Coplien
Lachlan Heasman
Mark den Hollander
Cesário Ramos

and The Scrum Patterns Group:

Esther Vervloed, Neil Harrison, Kiro Harada, Joseph Yoder,
June Kim, Alan O'Callaghan, Mike Beedle, Gertrud Bjørnvig,
Dina Friis, Ville Reijonen, Gabrielle Benefield, Jens Østergaard,
Veli-Pekka Eloranta, Evan Leonard, and Ademar Aguiar

[The Pragmatic Bookshelf](#)
Raleigh, North Carolina

94 patterns within 2 pattern languages

Product Organization Pattern Language & Value Stream Pattern Language

- 1 The Spirit of the Game
- 2 The Mist
- 3 Fertile Soil
- 4 Conway's Law
- 5 Birds of a Feather
- 6 Involve the Managers
- 7 Scrum Team
- 8 Collocated Team
- 9 Small Teams
- 10 Cross-Functional Team
- 11 Product Owner**
- 12 Product Owner Team
- 13 Development Partnership**
- 14 Development Team
- 15 Stable Teams
- 16 Autonomous Team
- 17 Self-Organizing Team
- 18 Mitosis**
- 19 Scrum Master**
- 20 Oyatsu Jinja (おやつ神社)
- 21 Small Red Phone
- 22 Scrum (Master) Coach
- 23 Fixed Work
- 24 Sprint Planning**
- 25 Swarming: One-Piece Continuous Flow**
- 26 Kaizen Pulse
- 27 Remove the Shade
- 28 Pop the Happy Bubble
- 29 Daily Scrum**
- 30 Scrum Master Incognito
- 31 Norms of Conduct
- 32 Emergency Procedure

- 33 Illegitimus Non Interruptus
- 34 Scrum of Scrums**
- 35 Sprint Review
- 36 Sprint Retrospective**
- 37 MetaScrum
- 38 Product Pride
- 39 Vision**
- 40 Impediment List
- 41 Value Stream**
- 42 Set-Based Design
- 43 Sprint Burndown Chart
- 44 Scrum Board
- 45 Product Roadmap**
- 46 Sprint**
- 47 Organizational Sprint Pulse
- 48 Release Plan
- 49 Release Range
- 50 ROI-Ordered Backlog
- 51 High Value First
- 52 Change for Free
- 53 Money for Nothing
- 54 Product Backlog**
- 55 Product Backlog Item
- 56 Information Radiator**
- 57 Pigs Estimate
- 58 Small Items
- 59 Granularity Gradient
- 60 Estimation Points
- 61 Fixed-Date PBI
- 62 Vacation PBI
- 63 Enabling Specification
- 64 Refined Product Backlog

- 65 Definition of Ready**
- 66 Yesterday's Weather
- 67 Running Average Velocity
- 68 Aggregate Velocity
- 69 Specialized Velocities
- 70 Updated Velocity
- 71 Sprint Goal**
- 72 Sprint Backlog**
- 73 Sprint Backlog Item
- 74 Teams that Finish Early Accelerate Faster
- 75 Production Episode
- 76 Developer-Ordered Work Plan
- 77 Follow the Moon
- 78 Visible Status
- 79 Dependencies First
- 80 Good Housekeeping
- 81 Whack the Mole
- 82 Definition of Done
- 83 Team Sprint
- 84 Responsive Deployment
- 85 Regular Product Increment**
- 86 Release Staging Layers
- 87 Testable Improvements
- 88 One Step at a Time
- 89 Value Areas
- 90 Value Stream Fork
- 91 Happiness Metric
- 92 Scrumming the Scrum
- 93 Greatest Value**
- 94 Product Wake

¶64 Refined Product Backlog



¶64 Refined Product Backlog

Agile enterprises must be poised to respond quickly to capitalize on opportunities to create value, and should avoid working—or planning—too far ahead.

Therefore:

The *Scrum Team* (particularly the *Product Owner* and *Development Team*) should meet frequently to properly order the *Product Backlog* and to break down the most imminent large PBIs into smaller ones. The *Development Team* should maintain current estimates for the *Product Backlog Items* that it will eventually implement (*Pigs Estimate*).

¶64 Refined Product Backlog

- Focus particularly on items near the top of the Product Backlog.
- Pay particular attention to:
 - > Dependencies between PBIs
 - > Dependencies of PBIs on external market factors (e.g., holiday sales seasons)
 - > Dependencies of PBIs to dates when resources may become available to make it possible to build the PBI (e.g., taking delivery of raw materials or enabling technology from a supplier).
- The Scrum Team should annotate PBIs on the backlog with estimates and value attributions.
- PBIs near the top of the backlog (for the next 2~3 Sprints) should be small enough so that no single one will require more work than 10% of the development effort (See *Small Items*).

¶51 High Value First

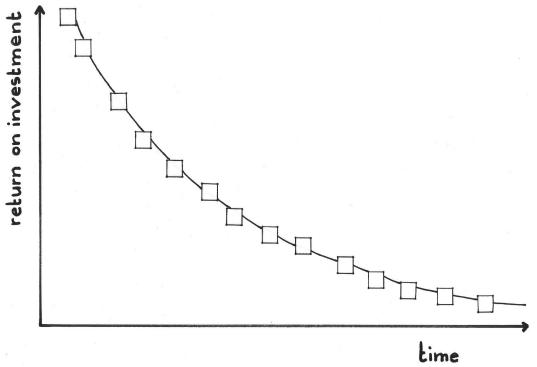


¶51 High Value First

One in the hand is worth two in the bush, and short-term developments should deliver value as early as possible.

Therefore:

Build the high value, most essential Product Backlog Items first. When the value curve and the cost curve cross, cancel the project. At any point, schedule the highest value remaining PBI in the next available iterations. If the cost of a PBI starts higher than its value, you can often split the PBI into its essential and inessential parts. Once the Scrum Team and the client re-estimate these parts, you may be in a better position to proceed.



¶75 Production Episode



¶75 Production Episode

Team members do the best work, and most efficiently complete planned work, when they can work together without distractions.

Therefore:

Compartmentalize all market-facing product realization work in a time-boxed interval, with participation limited to the *Development Team*. This uninterrupted development interval falls between *Sprint Planning* and the *Sprint Review*. During this interval, the developers take a few short scheduled discussions with the *Product Owner* to work toward a *Refined Product Backlog* but on the whole, this is otherwise uninterrupted time

¶35 Sprint Review

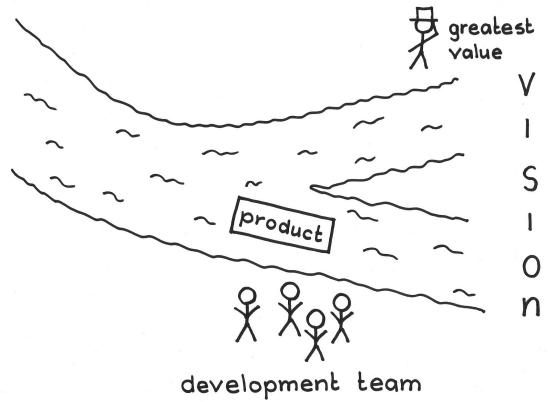


¶35 Sprint Review

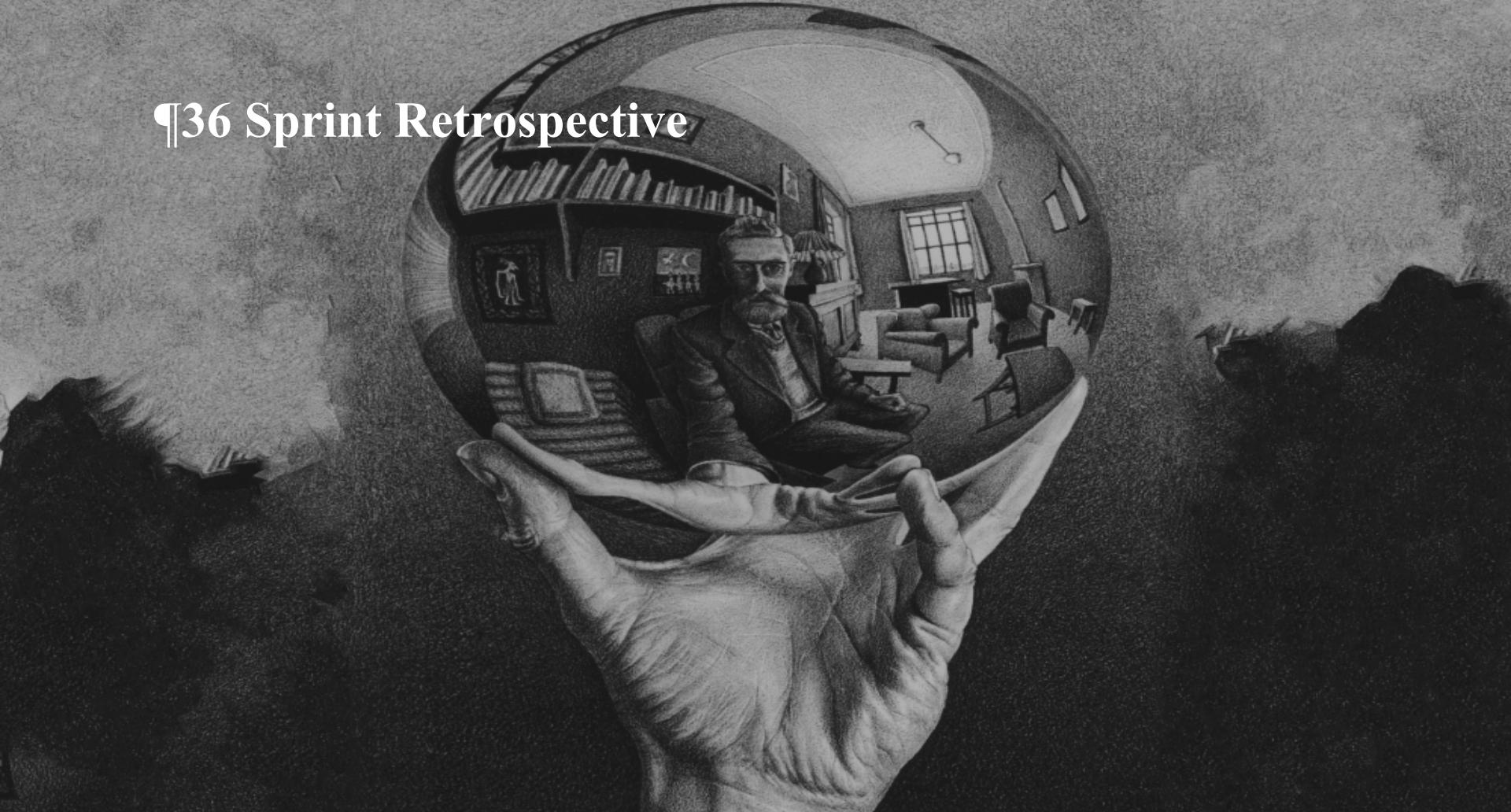
There must be closure on the state of a product after development is over; having completed a checklist of anticipated outcomes doesn't alone ensure that the product has come as far as necessary, or that the team will take the appropriate next steps in development.

Therefore:

End the *Production Episode* with an event to assess the status of the product and to learn about end-user needs, risks, opportunities, problems, and likely completion dates to ensure product is moving in the direction of *Greatest Value*.



¶36 Sprint Retrospective



¶36 Sprint Retrospective

Over time, without explicit attention, processes and discipline tend to decay. People get sloppy. Making isolated process changes without due focus feeds entropy, but without periodic change the team misses opportunities to increase value.

Therefore:

At the end of each *Sprint* have an **event where the *Scrum Team* can assess how it did** its work during the *Sprint*. The *Development Team*, the *Product Owner*, and other invited stakeholders attend the event. They work together both to discuss what parts of the *Product Increment* are and are not ready for deployment, about lessons learned about the product during the *Sprint*, and about tentative future product plans.

See *Scrumming the Scrum*.

References

- The ScrumPLoP site, The Scrum Patterns Group, <http://www.scrumplop.org>, 2010-2017.
- The Scrum Guide, Jeff Sutherland and Ken Schwaber, <http://scrumguides.org>, 2017.
- Scrum: The Art of Doing Twice the Work in Half the Time, Jeff Sutherland, 2014.
- Secrets of Scrum, Gertrud Bjornvig & James Coplien, slideshare, 2014.
- Alexander, A Pattern Language, Oxford, 1977.

FEUP.DEI



DEI - DEPARTAMENTO DE ENGENHARIA INFORMÁTICA
dei.fe.up.pt | secdei@fe.up.pt | +351 225 082 134
Director **João MP Cardoso** | Full Professor
jmpc@fe.up.pt | +351 220 413 284

FEUP - FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO
Rua Dr. Roberto Frias | 4200-465 Porto | PORTUGAL
www.fe.up.pt | feup@fe.up.pt | +351 225 081 400
 @DEI_FEUP