

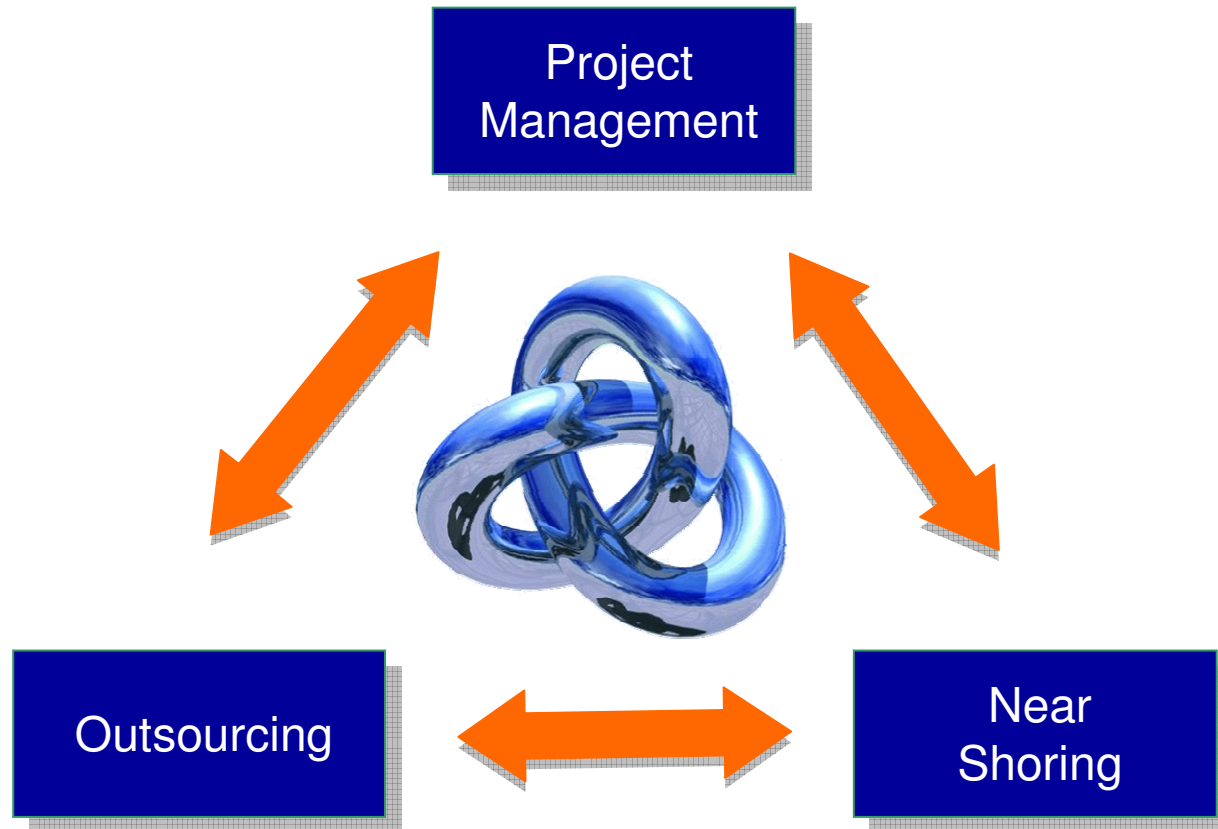
Matching Agile / Scrum with classic Project Management methods in nearshoring projects

Paul Krekel (ASML)

Wessel Haasnoot (Strypes)



Matching worlds



ASML's project management requirements

Deliver solutions

- Focus on Value for the Customer
- Think ahead and outside the box
- Show flexibility to changes

On time

- Enable an early warning system
- Cope with incomplete Work Breakdown
- Handle planning instabilities

Fair price

- Expose entrepreneurship
- Demonstrate effectiveness
- Take end-to-end responsibility (ownership)

Right Quality

- Specifications do change all the time
- Manage complex context: domain knowledge required



Prerequisites for outsourcing projects

Knowledge build-up

- The basis for thinking ahead and thinking outside-the-box
- Takes time to firmly embed: requires a planned investment
- Solution domain must be clear

Control

- Mutual trust
- Prevent micro-management
(fight the fear of losing control)
- Relies on close contact and timely feedback

Culture

- Time must be invested to get the teams “think alike”
- Teams must be stimulated to bring in their knowledge

Unit of Measure: communication

- Communication problems do signal underlying problems
- Fighting communication problems does not address the root cause!



Preconditions for near shoring

Stability in resources

- Teams shall be stable
- People “running away” are capital loss
- Availability and employability of new people

Culture and language

- “Thinking alike” requires similar culture
- Mastering Complexity involves many discussions
- Remote people do lack (context) informalities (coffee corner talk)

Mobility

- Bridge time zone differences
- Easy transport

Infrastructure

- Converge to common infrastructure / tooling
- provide remote access to local IT infrastructure

Useful tools from the Agile world

Scrum

- Incremental development in sprints (time-boxed iterations)
- Start sprint with an agreed list of tasks (sprint backlog)
- Check at the end what has been delivered
- Retrospectives (what went well / what could be improved?)

Kanban Method

- Visualize the work flow
- Manage the flow
- Limit the work in process

Useful tools from Project Management

End-to-end planning

- Means to define and track the overall timeline
- Links iterations to the overall progress

Earned Value tracking

- Measures the value of features completed
- Can be used per iteration and end-to-end

Dependency management

- New insights mean new dependencies, tasks, change requests
- Risk Management determines order and priorities
- Independent (decoupled) tasks help to keep sustainable pace

Our customization of Scrum - I

Sprint

- 4 weeks sprint: balance of delivery overhead and reasonable control

Iterative development

- Cadence in demo's and deliveries at the end of the sprint
- Prevents unpredictable big bang integration

Retrospectives

- Necessary means of process control

Scrum teams and roles

Stand-up

Board



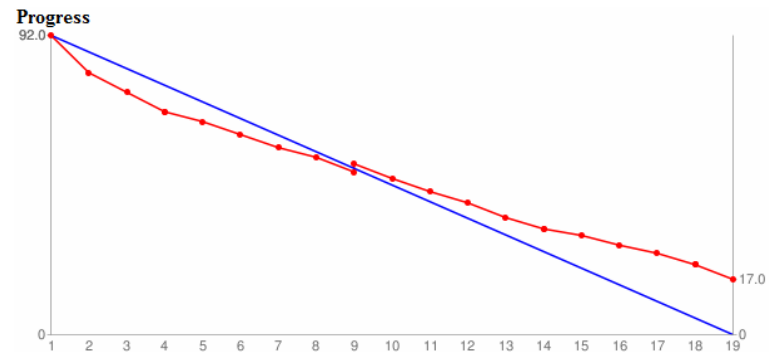
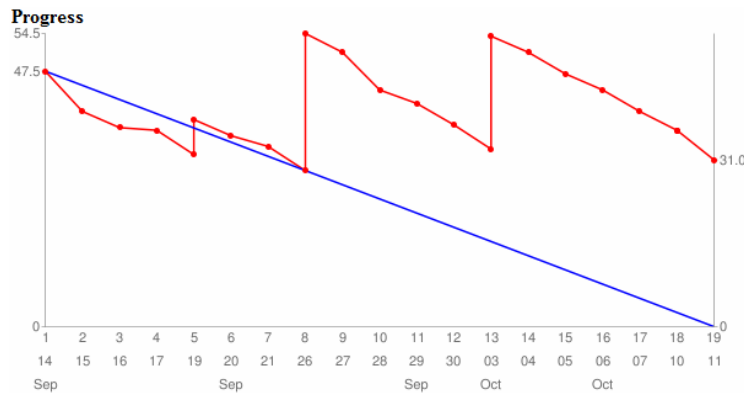
Our customization of Scrum - II

Sprint planning

- Sprint plan to rigid
- Replaced by Product Backlog from Kanban method

Burn down

- Unusable due to sprint instability
- Does not show the value delivered
- Replaced by Earned Value graph

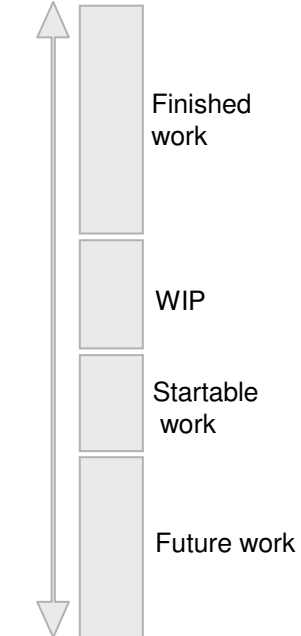


Our application of the Kanban method - I

Actively manage the product backlog

- Future Work
- Startable Work
- Work In Process
- Done

project budget



Status	StoryID	Story	Size	Actual	Spec ready	Go PL	Stream ?	Reviewer	Done
Done this sprint	3	Story 3	3	5	D002334	Y	p2388	Nico	Doc accepted, stream sign-off by PL
	4	Story 4	12	11	D002335	Y	p2389	Paul	Tool available and tested OK on bench
	1	Story 1	4	3	D002330	Y	p2390	Nico	Stream signed-off by PL
			19						
Work in Progress	6	Story 6	4		D002336	Y	p2345	Nico	Stream signed-off by PL
	8	Story 8	7		D002341	Y	p2347	Mark	Doc accepted, stream sign-off by PL
	9	Story 9	6		D002342	Y	p2349	Jerry	Tool available and tested OK on bench
	12	Story 12	3		D002345	Y	p2343	Nico	Stream signed-off by PL
			20						
Startable	11	Story 11	13		D002336	Y	p2350	Jeroen	Stream signed-off by PL
	5	Story 5	5		D002331	Y	p2333	Nico	Doc accepted, stream sign-off by PL
	10	Story 10	4		D002337	Y	p2332	Mark	Tool available and tested OK on bench
	13	Story 13	7		D002338	Y	p2391	Harald	Stream signed-off by PL
			29						
Future Work	2	Story 2	20		D002333	N	p2395	Nico	Design reviewed OK by Mark
	7	Story 7			Nov-23	N	p2392	Nico	
	14	Story 14	3		D002344	Y	p2340		Stream signed-off by PL
	15	Story 15			Dec-1?	N	Rqst'ed		
	16	Story 16			D002347	N	Rqst'ed	Jeroen	
	17	Story 17			?	N			
	18	Story 18			?	N	p2331	Mark	
	19	Story 19			?	N			
20	Story 20			?	N				

Our application of the Kanban method - II

Manage flow

- Spent time on getting work startable (Agile: 'ready')
- Have sufficient work available (both startable and future work)
- If work not started yet, order and priorities are flexible
- When done, pull next task

Limit Work In Process

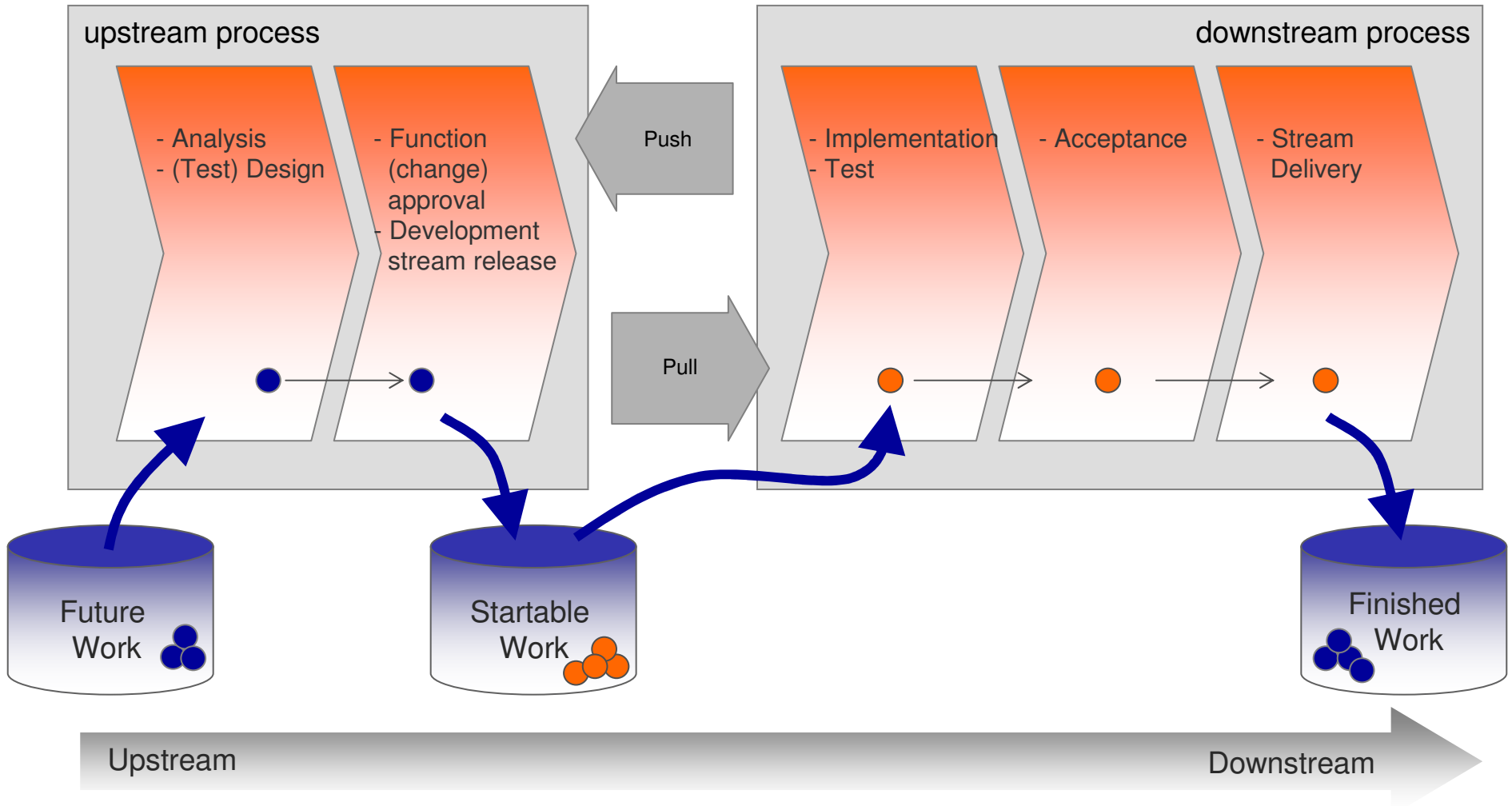
- Complete work first before next task is picked up
- Solve blocked work (impediment) instead of picking up a new story
- Drive to completion

Workflow

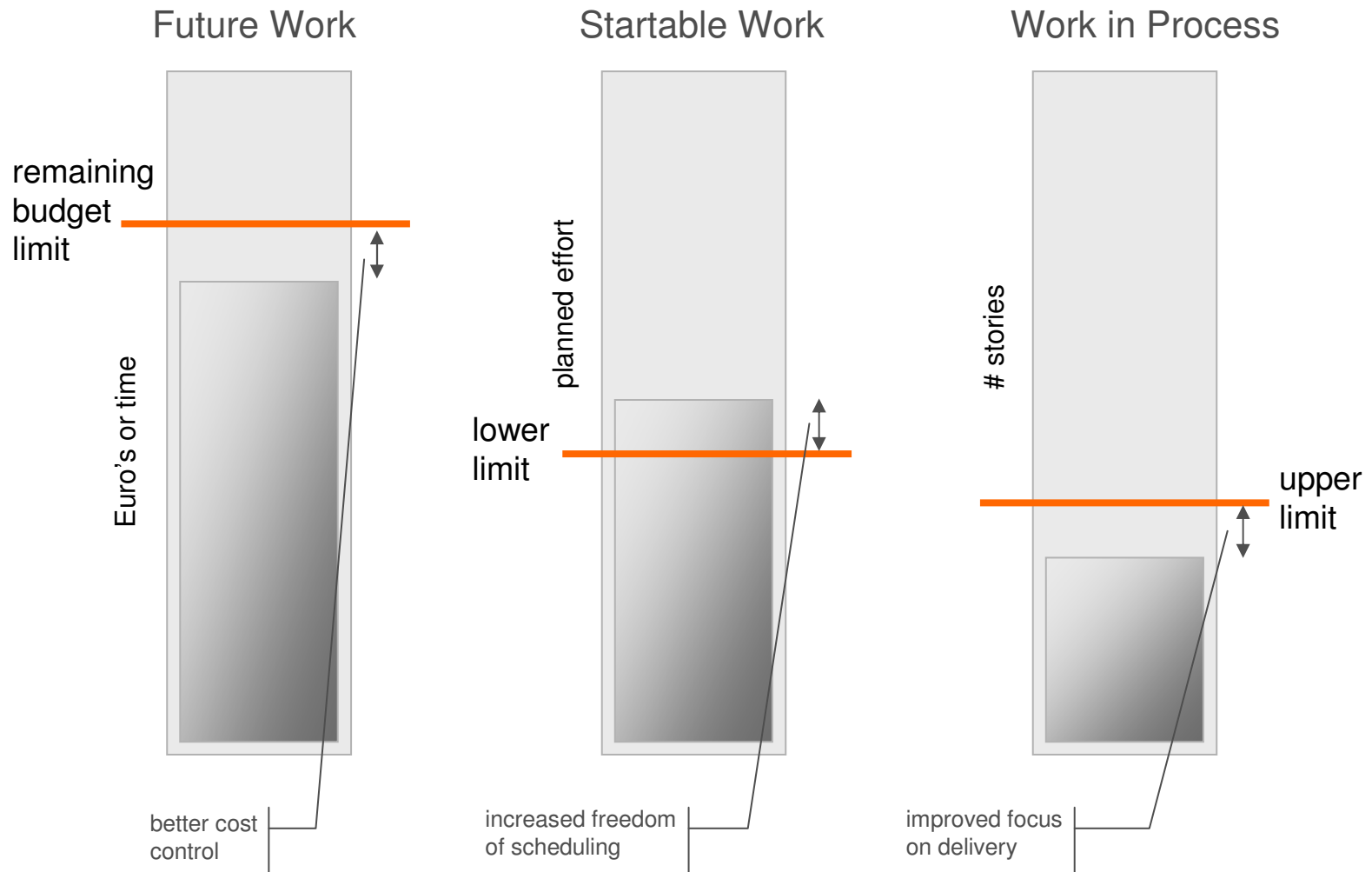
Push information: instruction to transform “Future Work” parts into “Startable Work” parts



Pull things: withdraw of a “Startable Work” part into the downstream process



Manage WIP and Startable Work level



Our use of Earned Value tracking

Supports Kanban method

- Shows how many features are delivered
- Indicates effectiveness of the team
- Insensitive to sprint changes

Per sprint

- Measurement on day to day basis
- Task done → add its value to the graph

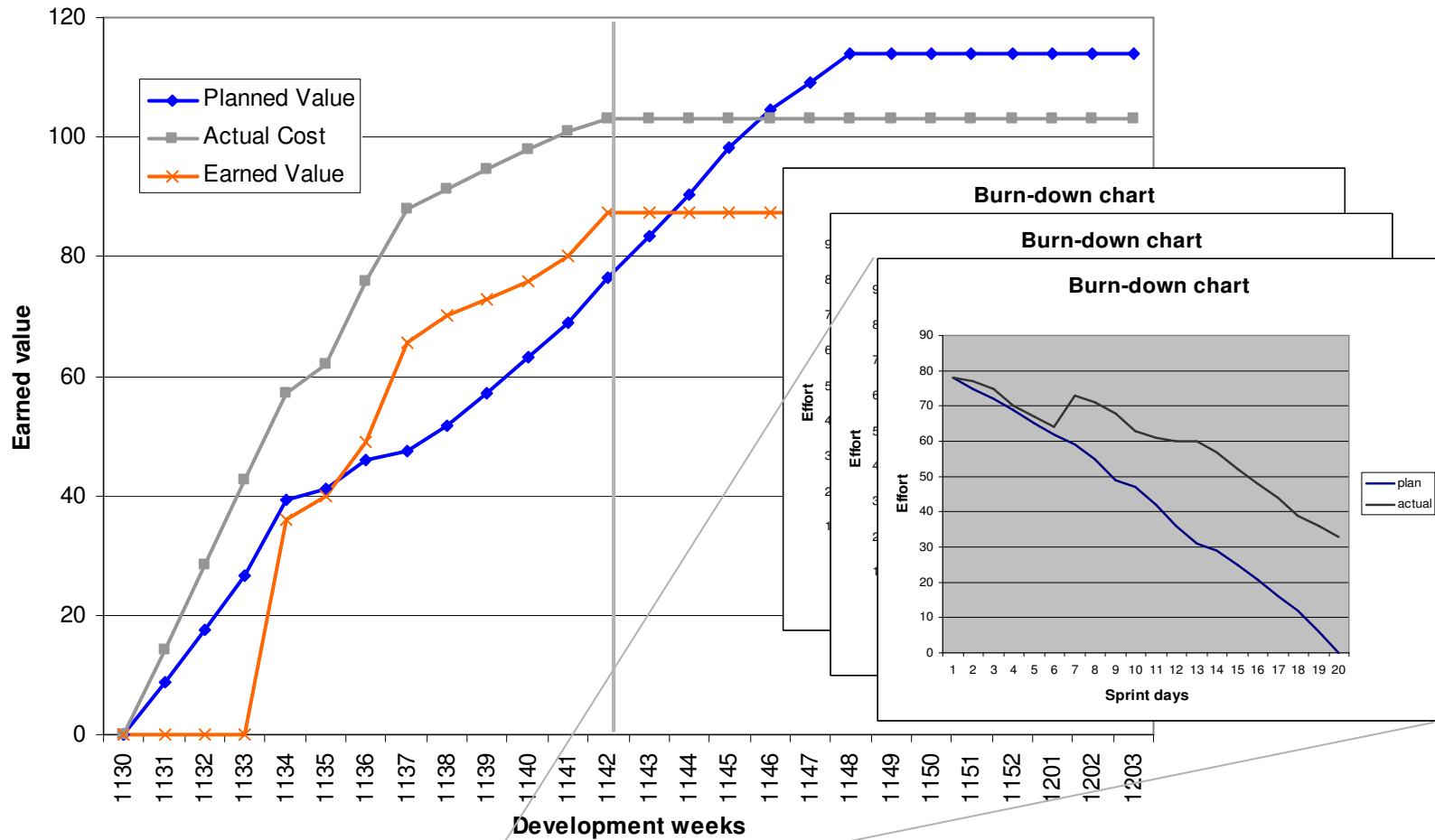
End-to-end tracking

- Measurement per feature (story)
- Story done → add its value to the graph

The use of Earned Value chart

The sum of Burn-down charts per sprint does not give a clue what the overall progress is, the EV chart however does !

Earned Value Chart



□ sprint window

Observed effects - I

Teams focuses onto added value

- Earned Value method is a 1:1 measure for entrepreneurship of teams
- Smooth negotiation on stories content and effort estimates
- Ownership

Planning performance get within 20%

- Start criteria are much better defined
- Unclear work is not started (specification or process wise)
- Estimated versus actual effort within 20% accuracy range

Observed effects - II

Progress is clearly visible

- Delivery content is lean and comprehensible
- Specifications are far less troublesome to get agreed
- Completed work is more easily demonstrated
- True focus on finishing the job committed

Team keeps a sustainable pace

- mutual feeling of trust and control
- there is capacity for (process) improvements

Improvements for the future

- Manage via self steering teams
- Better limit Work In Process
- Limit change requests on WIP (dependency control)

Thank you !



References

From these resources we've got most of our inspiration

Kanban *by David J. Anderson*

- Getting more out of our way of working

The software project manager's bridge to agility *by Sliger and Broderick*

- Yes, agile and project management can be matched!

Projectmanagement op basis van ncb3 *by Bert Hedeman et al.*

- Control mechanisms and soft skills

Training Certified Scrum Master and Certified Product Owner

by The Scrum Alliance

- Understanding why it works and how to apply it in our situation