



Beating the Post-Agile Blues

Presented by Susan Block, CBAP
Project Manager, The Vanguard Group



May 2015

The Post-Agile Blues

+ The Syndromes

- The Symptoms
- The Cause
- The Remedy



The Syndromes

- The Price is Right
- Hockey Stick Effect
- All Work and No Play
- Being Mistaken for a Cheetah
- This Doesn't Seem Agile
- The Necessary "Evil"
- Promises, Promises
- We, the People
- Surprise!



Syndrome: The Price is Right



Symptoms / Cause

- Since the business is paying for the project, everything the team does must be correlated to business need
- The product backlog contains much more than the budget supports

Remedy

- Recognize that the “business of IT” has value and ensure that IT internal needs are accounted for in the product backlog
- Value the cost of features in product backlog
- Clean out any technical debt

Syndrome: Hockey Stick Effect



Symptoms / Cause

- Stories spill over from sprint to sprint and new stories are piled on
- Too many stories are opened at the same time
- Stories don't close until a mad rush at the end of the sprint

Remedy

- The team should avoid overcommitting
- Avoid swirl by making sure that the requirements are clear and that the product owner is accessible to clarify
- Staff the team with multiple skillsets and leverage them for swarming

Syndrome: All Work and No Play



Symptoms / Cause

- Unrealistic deadlines
- Pressure to constantly increase productivity
- Lack of communication, teamwork, and possibly trust

Remedy

- Plan to include slack for other activities
- Provide for downtime & fun-time – together as a team and as individuals
- Take action from retrospectives

Syndrome: Being Mistaken for a Cheetah



Symptoms / Cause

- Sprinting for too long
- Quality is sacrificed to gain speed
- No rest for the weary

Remedy

- Measure velocity and adjust periodically
- Alter the intensity of the sprints
- Follow the cadence of the development lifecycle

Syndrome: This Doesn't Seem Agile



Symptoms / Cause

- Sprint is often interrupted for immediate “fire drills”
- Lack of vision or feature prioritization
- The team is going through the motion of following Agile protocols

Remedy

- Create operational model to respond to production support needs
- Roadmap the product features iteratively
- Calibrate Agile adoption to make it real



Must

Syndrome: The Necessary “Evil”

Symptoms / Cause

- Regression testing is arduous
- The code is not maintainable
- No allowance for refactoring
- Either no product backlog “pruning” or excessive care given

Remedy

- Invest in test automation
- Allocate time for refactoring
- Consider architectural direction if change is warranted
- Prune the product backlog in the right dose

Syndrome: Promises, Promises



Symptoms / Cause

- Delivery deals are made without the team's knowledge
- Team estimates are improbable
- False constraints are imposed

Remedy

- Empower the team to be engaged and accountable
- Refine the estimation method
- Apply classic impact analysis on scope, schedule, and cost

Syndrome: We, The People



Symptoms / Cause

- The team is not empowered
- The team is not expanding their skills
- Continuity or duration of resource allocation is challenging
- Wrong resources are assigned

Remedy

- Adopt a self-directed approach and protect the team
- Demonstrate commitment to learning
- Plan rotational opportunities carefully
- Staff project by skill, not just by role

Syndrome: Surprise!



Symptoms / Cause

- Business priorities shift or conflict
- Resources are strained due to unforeseen issues
- Technical hurdle occurs

Remedy

- The product owner must prioritize
- Deepen the bench by pairing
- Engage ancillary resources
- Make impediments extremely visible



http://www-tc.pbs.org/mediashift/wp-content/uploads/sites/8/2013/12/agile_broussard_sized.jpg

Agile relies on the people who invest in the extraordinary discipline it takes to sustain success

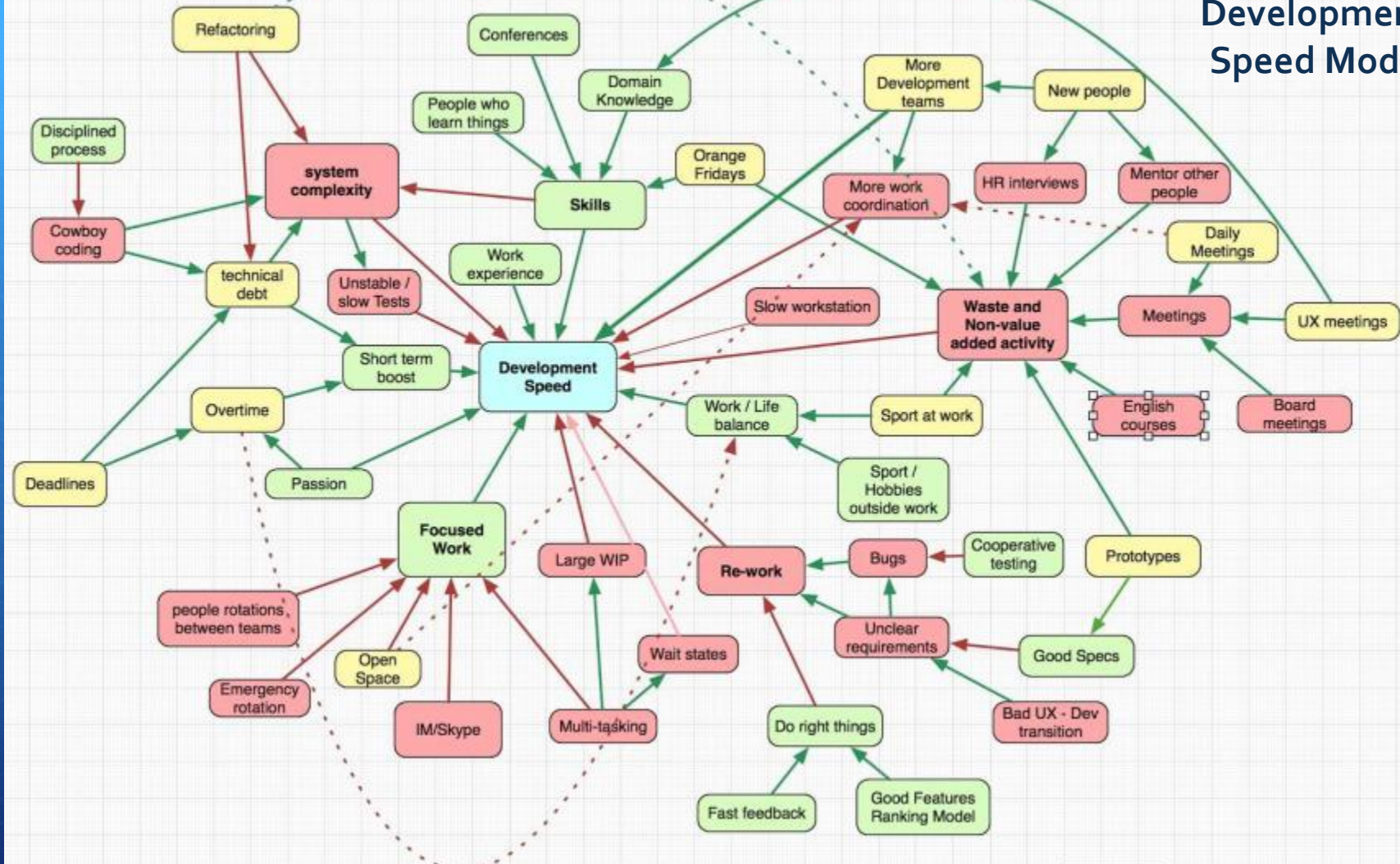
Susan Block

Project Manager
Information Technology
The Vanguard Group



susan_block@vanguard.com

Software Development Speed Model



<http://www.targetprocess.com/articles/speed-in-software-development.html>

Activity increases speed
 Activity maximum exists
 Activity slows down dev