Product Owner Workshop

Pollyanna PIXTON

Co-Founder, **Accelinnova**
President, Evolutionary Systems
Director, Institute of Collaborative Leadership

Paul GIBSON

Senior Consultant, **Accelinnova**
Development Operations Manager, IBM

→ what do **you** want from this class?

Workshop Overview

- Roles & Responsibilities
- Tools and Tips
  - Planning
  - Sprints
- Scenarios
- Q&A

Product Owner Responsibilities

- Assess product opportunities
- Define product to be built from the customer and user point of view
- Does NOT define the solution The team defines the solution

How do we do this in Agile?

Key Ideas in Agile

From Ron Jeffries

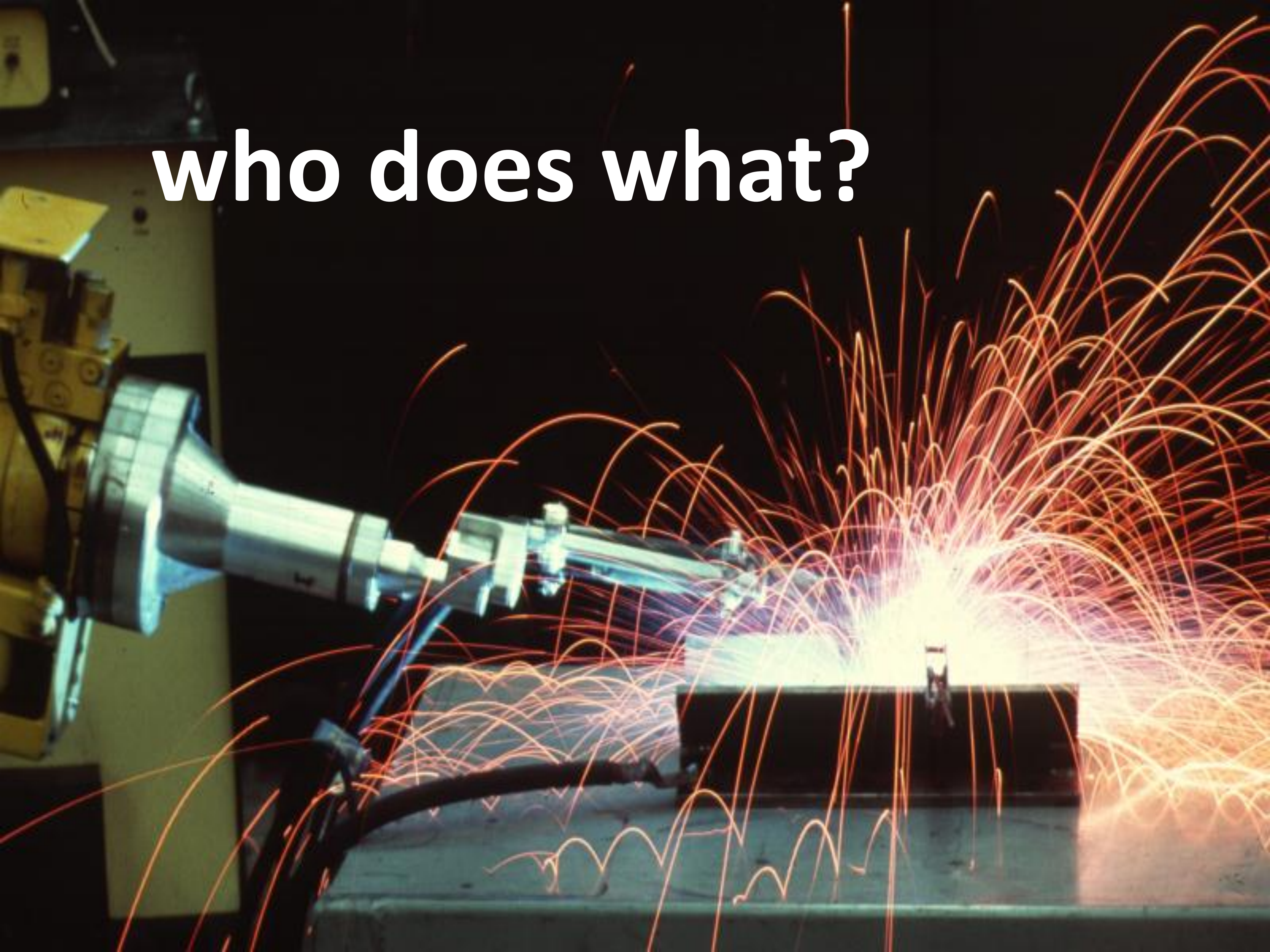
Key Ideas in Agile

- Putting people with needs in direct contact with people who can fulfill those needs
- Populating projects with all the needed people and capabilities to get the job done
- Building work incrementally and checking results as you go

Key Ideas in Agile

- Preparing for and influencing the future but not predicting it
- Making tasks concrete and quickly finishing them
- Giving people work to do and the knowledge to do it, not pushing them around like pawns on a chessboard
- Focusing on providing value frequently and rapidly, not directly on cost

who does what?



Roles and Responsibilities Review

Roles are Responsibilities

- Stakeholders:
  Input to Product Business Objectives
- Product Owner Team:
  Deliver the right product
- Delivery Team:
  Deliver the product right
- Scrum Master:
  Help Product Owner Team and Delivery Team work together

Product Owner

Builds and prioritizes, by business value, product and release backlog in collaboration with:

- Business Leaders
- Stakeholders
- Delivery Team
- Scrum Master

Scrum Master

- Removes barriers between development and customer so customer directly drives development
- **Facilitates** creativity and empowerment
- Improves productivity of delivery team in any way possible
- Improves engineering practices and tools so each sprint is ready to deploy

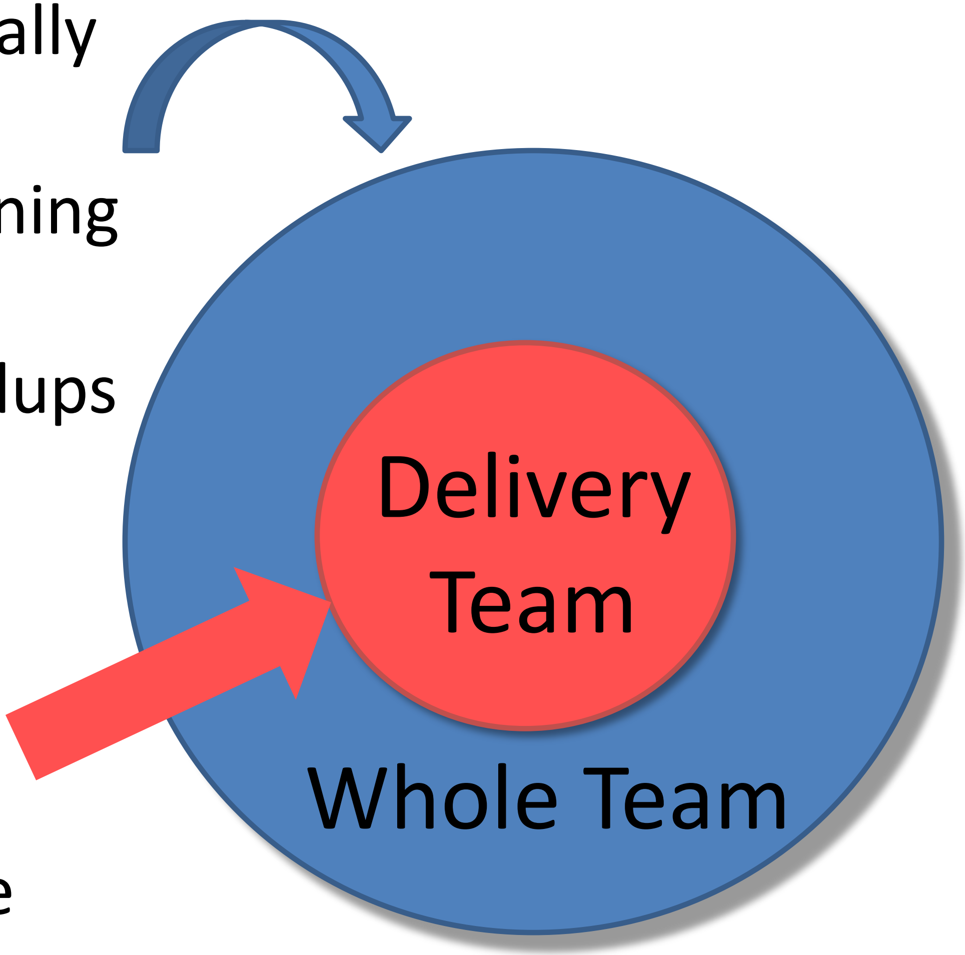
Scrum Master

- Is **not** a project manager
  - Team manages itself
- Does **not** have authority over the team
  - Team makes decisions
- Always asks the question:

  "How are the Product Owner and Team doing?"
- Challenges the organization, key-role in the change

Whole Team & Delivery Team

- Involved but not personally committed to delivery
- May be involved in planning & retrospectives
- May **observe** daily standups



- Committed to delivery
- Active participants in planning & retrospective
- Active participant in daily standup

Product Owner Does What?

- Interfaces with the Product Manager
- Owns the product and release backlog but does **not decide alone** what is in it and how it's prioritized
- Organizes and facilitates product planning sessions with Product Manager, stakeholders and team to generate themes and epics

Product Owner Does What?

- Organizes and facilitates the release planning session with stakeholders and team to generate the user stores
- Attends the scrum planning session to answer any team questions
- Does **not decide** what is in the sprint backlog
- Connects team with customers

Product Owner Does What?

- Clarifies questions from team from a business point of view
- Removes blockers from business view
- Does **not** attend the team retrospective unless invited
- The product owner does not manage the team. They manage themselves.

➡ What else????

Scrum Master Does What?

- Facilitates daily standup
- Keeps blocked list and resolves blocking issues with and for the team
- Makes sure the team has what they need to succeed
- Keeps the information radiator up to date
- Facilitates the retrospective

Team does what?

- Estimates the bigness of the user stories
- Decides what stories should be completed in the sprint
- Delivers done, done, done stories
- Keeps technical debt low
- Helps each other succeed
- Performs code reviews
- Holds each other accountable

Inspire Innovation

- Reward and acknowledge new ideas
- Create the freedom to fail
- Form an Inception Team at the beginning of the project

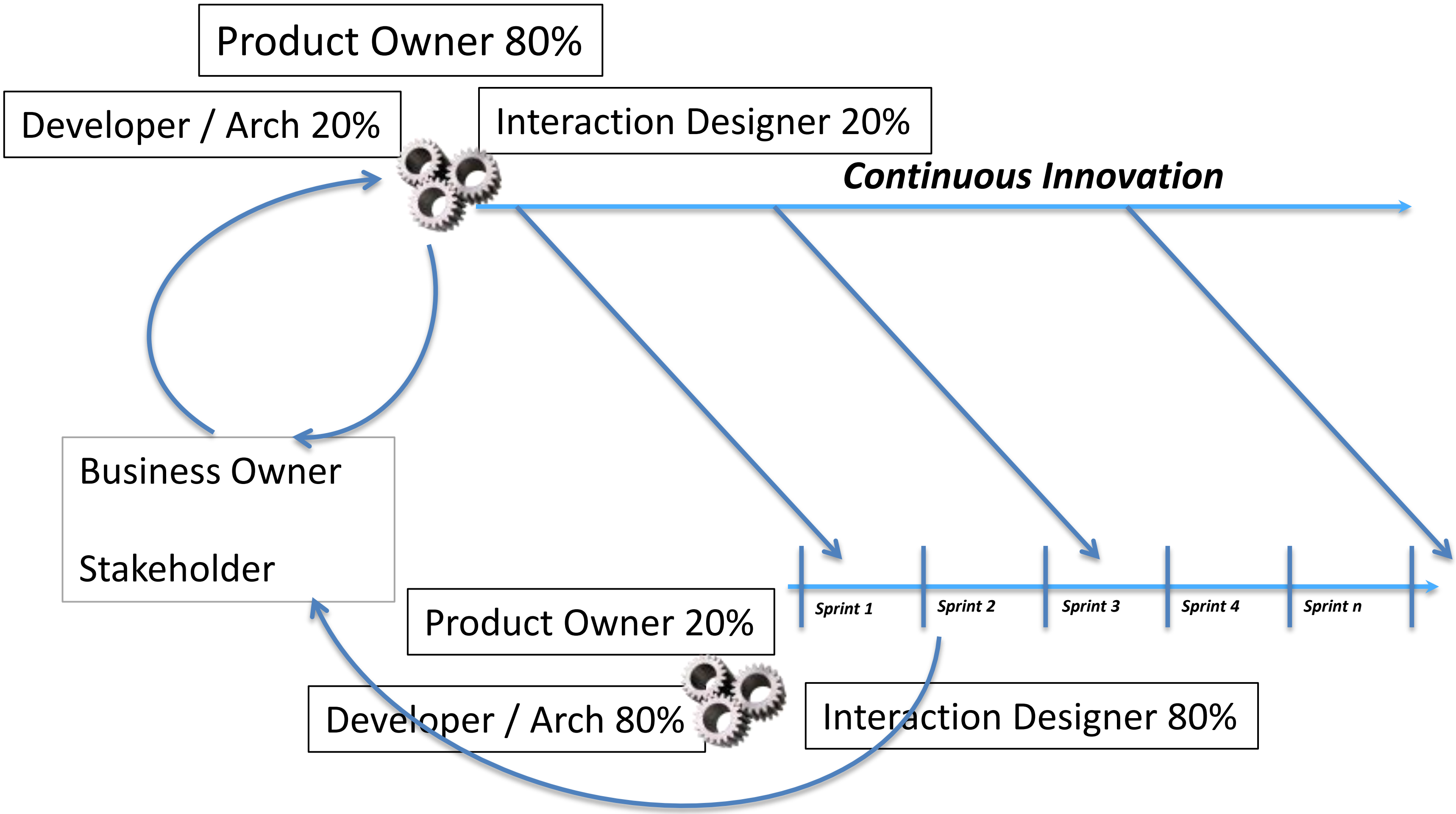
"I have not failed. I've just found ten thousand ways that won't work."

- Thomas Edison

Inception Team

- Responsible for continuous innovation during the release
- Consists of:
  - Product Owner
  - Architect
  - UX Developer

Inception Team



Product Owner 80%

Developer / Arch 20%

Interaction Designer 20%

Continuous Innovation

Business Owner

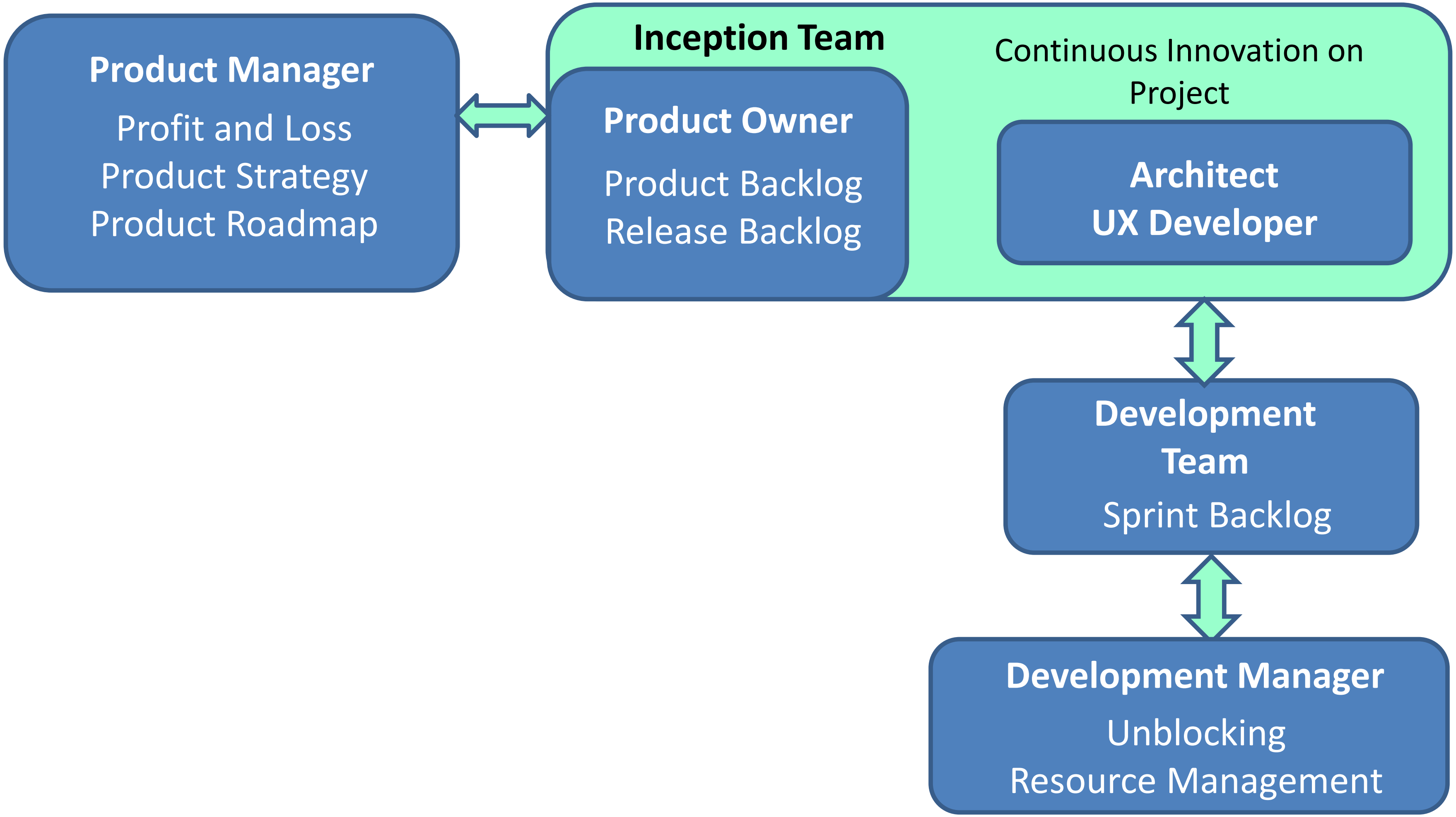
Stakeholder

Product Owner 20%

Developer / Arch 80%

Interaction Designer 80%

Agile Organizational Structure



Product Manager

Profit and Loss
Product Strategy
Product Roadmap

Inception Team

Product Owner

Product Backlog
Release Backlog

Continuous Innovation on Project

Architect
UX Developer

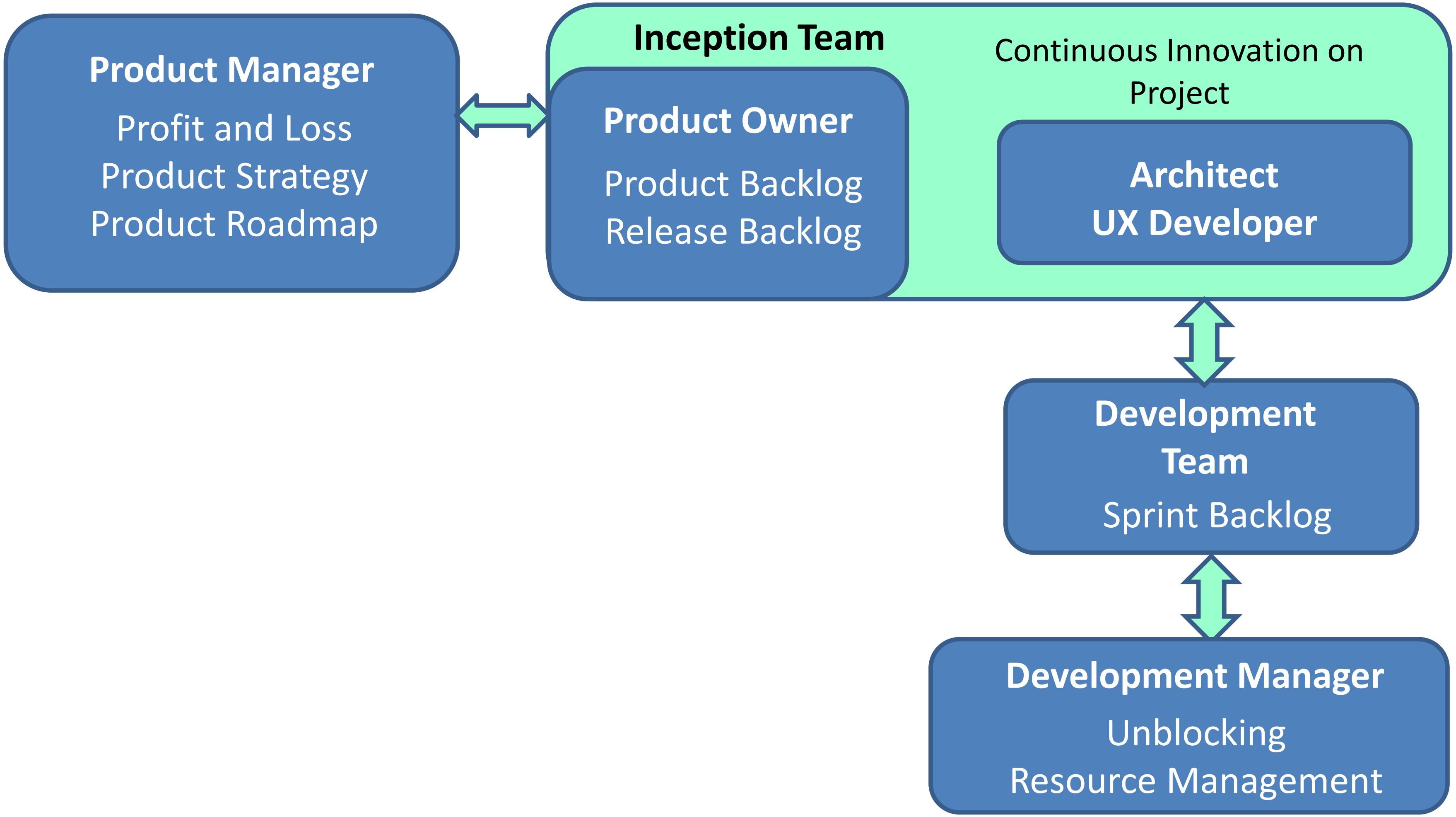
Development Team

Sprint Backlog

Development Manager

Unblocking
Resource Management

Who is who on your team?



Product Manager

Profit and Loss
Product Strategy
Product Roadmap

Inception Team

Product Owner

Product Backlog
Release Backlog

Continuous Innovation on Project

Architect
UX Developer

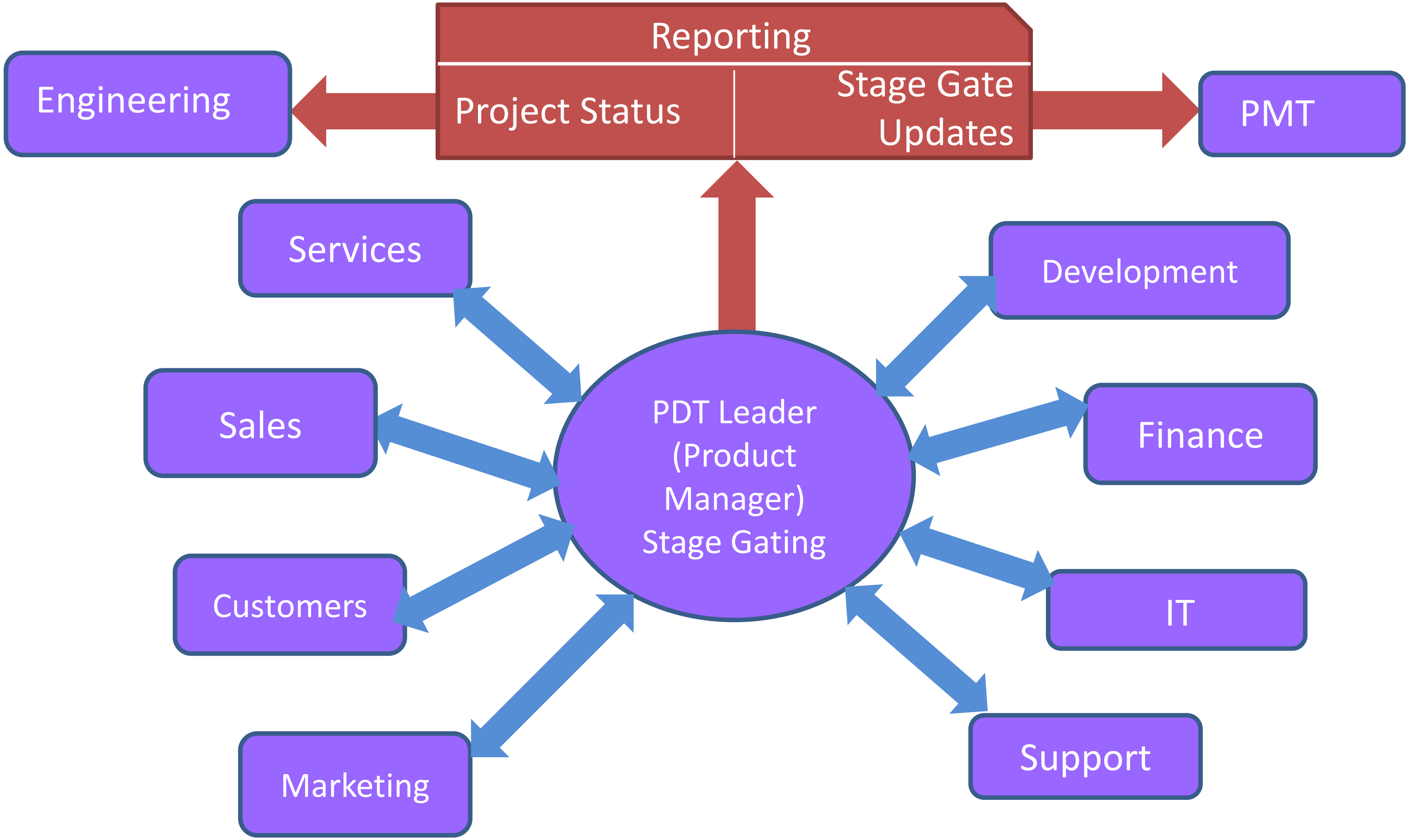
Development Team

Sprint Backlog

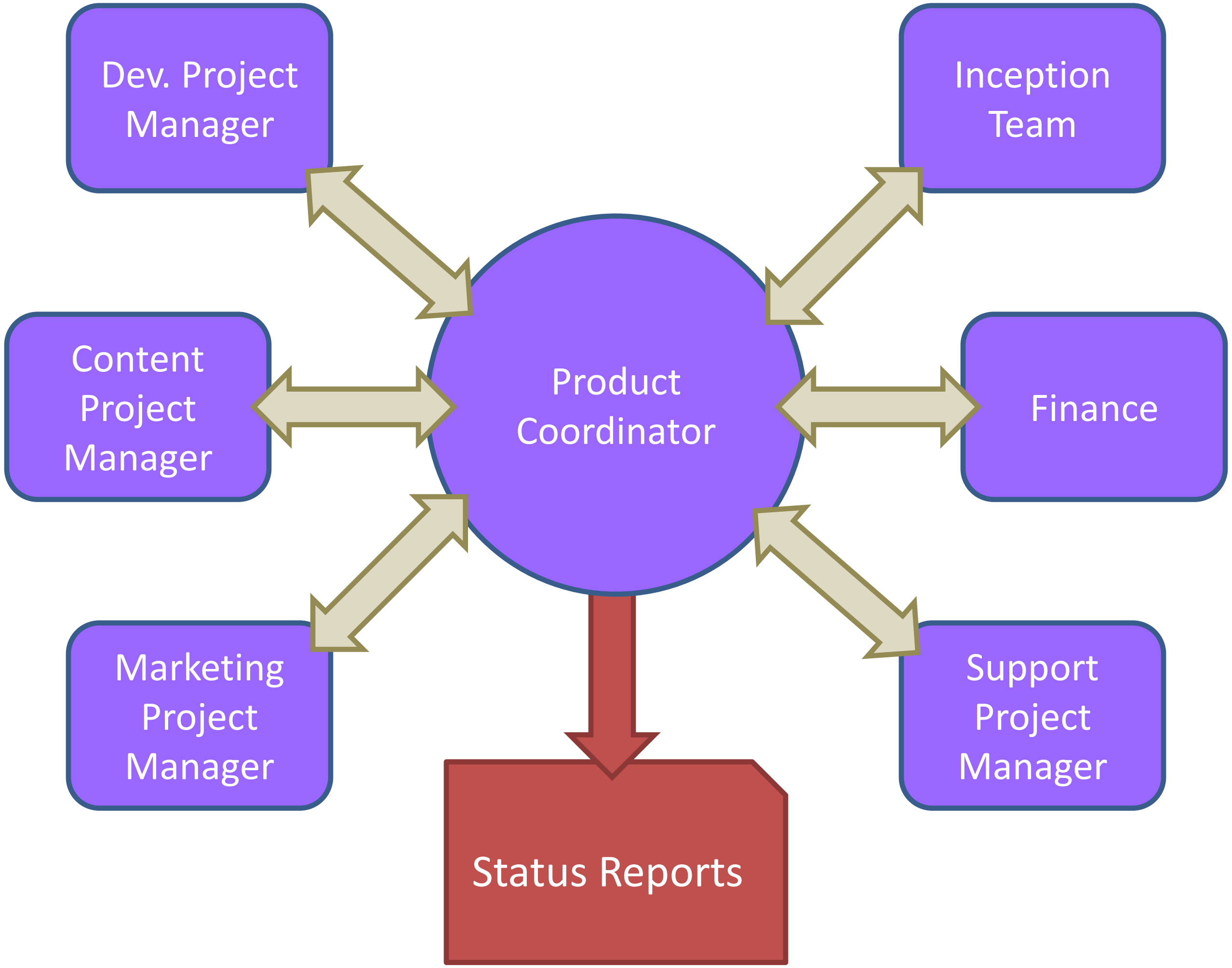
Development Manager

Unblocking
Resource Management

Product Development Team



Who are these people?



Dev. Project Manager

Inception Team

Content Project Manager

Product Coordinator

Finance

Marketing Project Manager

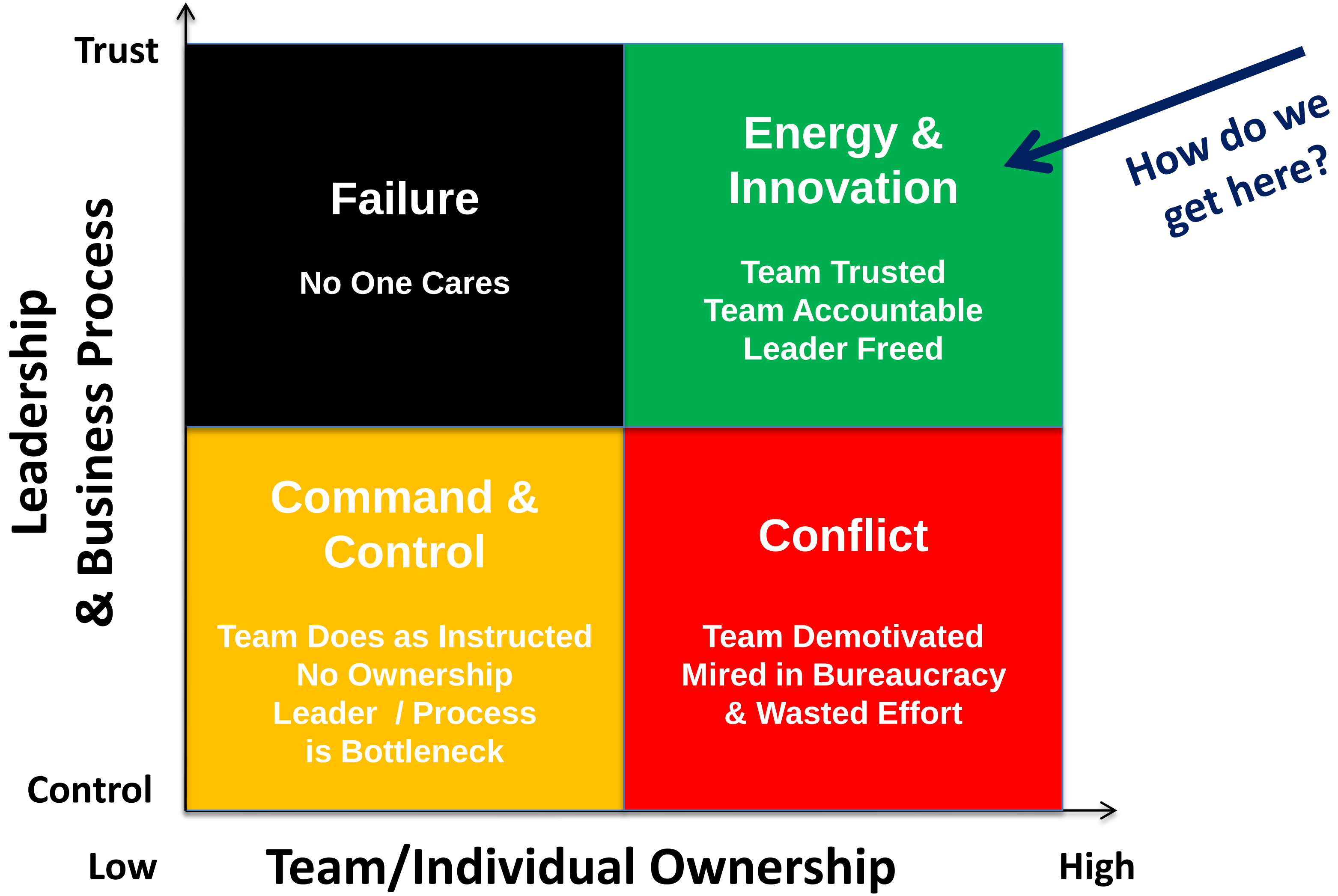
Support Project Manager

Status Reports

Ownership

Ownership Review

Trust/Ownership Model



Trust

Control

Leadership & Business Process

Low

Team/Individual Ownership

High

Failure

No One Cares

Energy & Innovation

Team Trusted
Team Accountable
Leader Freed

Command & Control

Team Does as Instructed
No Ownership
Leader / Process is Bottleneck

Conflict

Team Demotivated
Mired in Bureaucracy & Wasted Effort

How do we get here?

Requires Trusting Environment

Leader’s View

- The team won't let me down
- The team understands what we need
- They will do the right thing
- They will tell me if they need help

Requires Trusting Environment

Individuals within the Team

- We understand the vision and the need
- We are jointly committed to meeting our goals
- We stand or fall together
- We have Ownership

use

authentic motivation

- Alfie Kohn

Punished By Rewards

foster

collaboration

teams

collaborate

to

make their

decisions

let **people**

choose

let people

choose

how,

what,

and

when

content

provide
meaningful
work

Helping the team take ownership

- Clearly explain the business context and the value of the stories being delivered

- Allow team members to select what they can do within the sprint. Peer pressure will build ownership.

create an **open environment?**

brainstorm

group

in

silence

prioritize

Helping the team take ownership

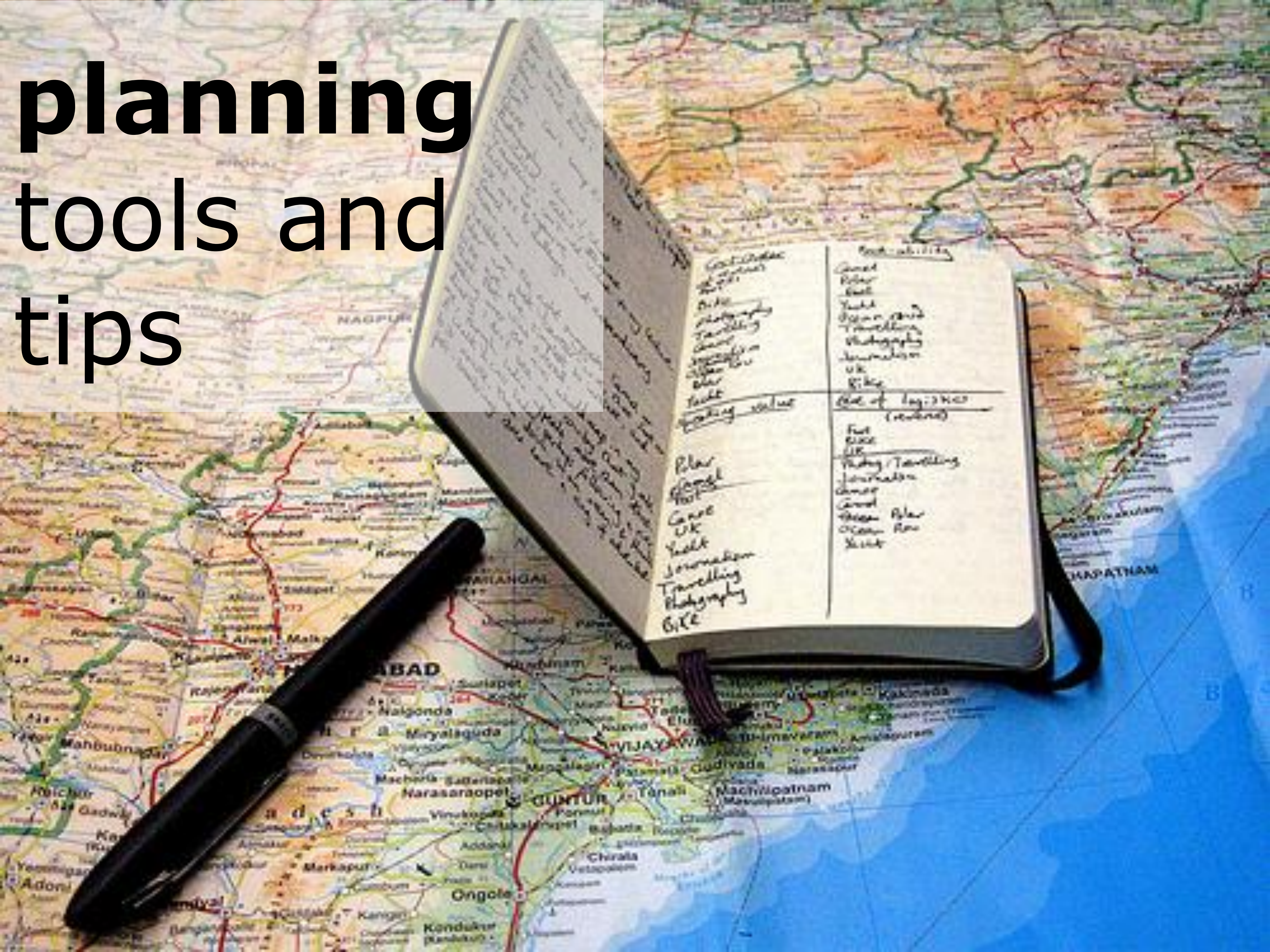
- Be visibly seen to support the team and actively address issues outside the team’s responsibility
- Create a culture of trust
- Prove that you trust the team
- Be totally transparent and honest
- Continually “ask questions to guide”
- Don’t take back ownership by giving the team the solution

Ownership Exercise

- What can you do to help your team take ownership?

- How can you show you trust the team?

planning tools and tips



Planning Review

Product Planning Preparation

1. Exactly what problem will this solve?
2. For whom do we solve this problem?
3. How big is the opportunity?
4. How will we measure success?
5. What alternatives are out there now?

Product Planning Preparation

6. Why are we best suited to pursue this?
7. Why now?
8. How will we get this product to market?
9. What factors are critical to success?
10. Given the above, what’s the recommendation?

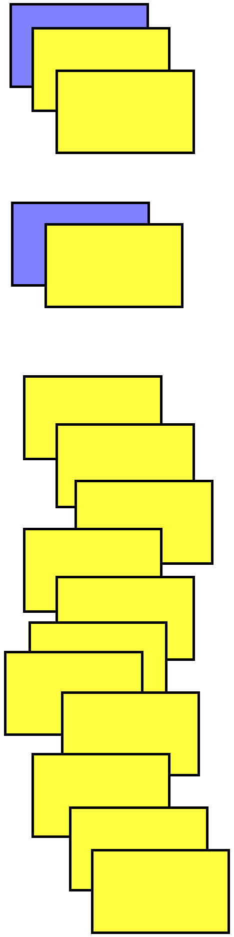
Product Planning Exercise

Answer the 10 questions for your project

Share them with your entire team.

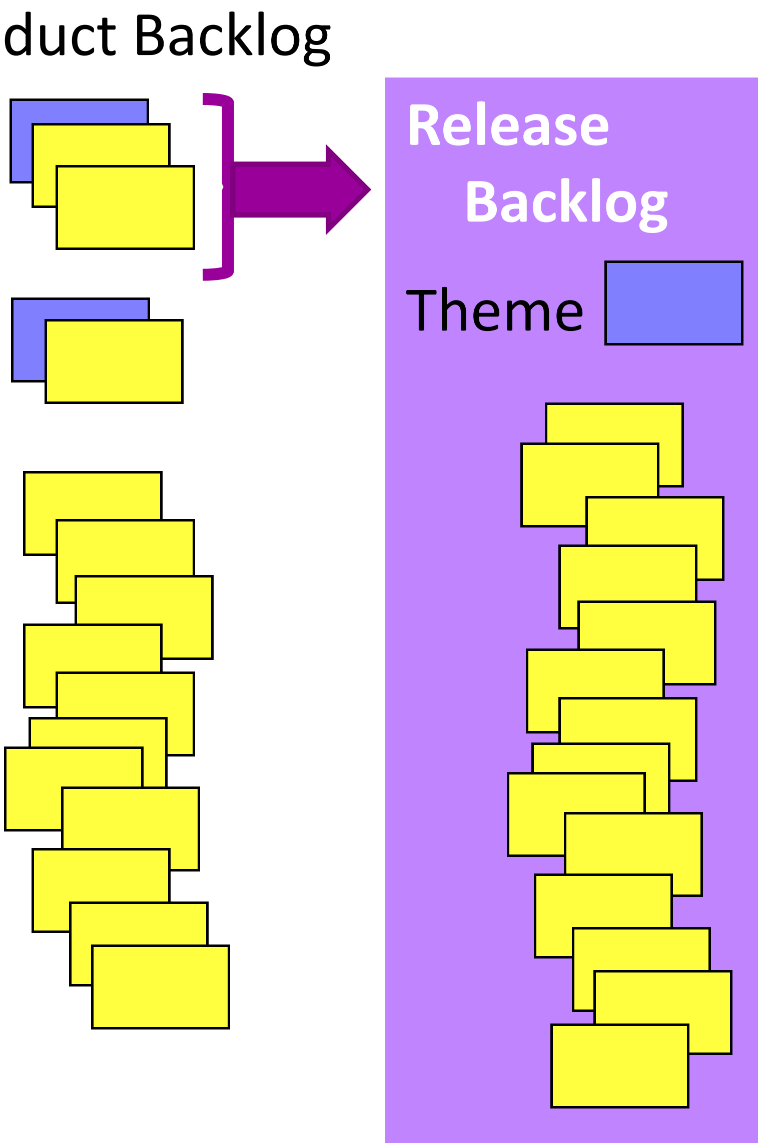
Product Planning

Product Backlog



Input	SOW, Purpose
Output	**Prioritized Product Backlog** (Epics and Release Themes)
Who	Business, Product Owner, Dev Team representatives

Release Planning, part 1



Input	Product Backlog
Output	**Prioritized Release Backlog** (User Stories)
Who	Business, Product Owner, Dev Team

Release Planning, part 1 Tips

- Create decisions filters
  - One for the product
  - One for the release
  - One for the sprint
- Use them!
- “Does this pass all the decisions filters?”

the **“billboard”** test…

Decision Filters Example

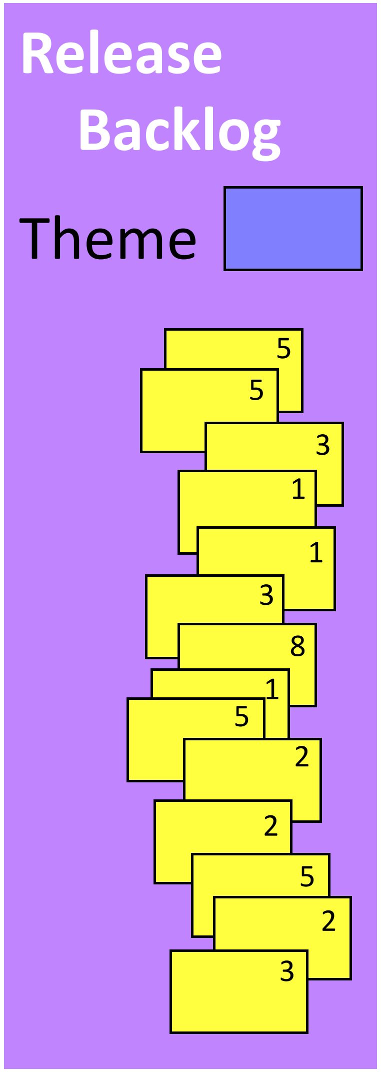
For WebConnect (Nova)

- Product:
  - We are the go-to company for your customer connections management needs
- Release (Phase II)
  - One stop shopping for mailing and shipping
  - What can we get done in 6 months

Decision Filters Exercise

- What is the billboard for your product?
- What are the decision filters for the product you are working on?
- What are the decision filters for the release you are working on?

Release Planning, part 2



Release Backlog

Theme

Input	Release Backlog
Output	**Estimated and Prioritized Release Backlog** (story points)
Who	Development Team using planning poker

CAUTION

Solution without a problem!

get inside consumer's mind

User Stories

Two Types of User Stories

- Epic User Stories (aka Epics)
  - High level features of a product
  - Fit into a release
  - Each release has a Release Theme
  - All epics form the Product Backlog
- User Stories
  - Breakdown of Epics into smaller features
  - Fit into a sprint (iteration)
  - User Stories form Release Backlog

What is a User Story?

A concise, written description
of a piece of functionality
that will be valuable to a stakeholder.

As a <role>,
I can <goal>
so that <business value>

Example: NASA User Story

As an < astronaut >

I want to < write easily with a ball point pen while in Zero gravity >

So that < I can record key information that I might otherwise forget >

User Story Example

NASA specified and developed, at great expense, a ball point pen that Apollo astronauts could use in space where gravity would not make the ink flow.

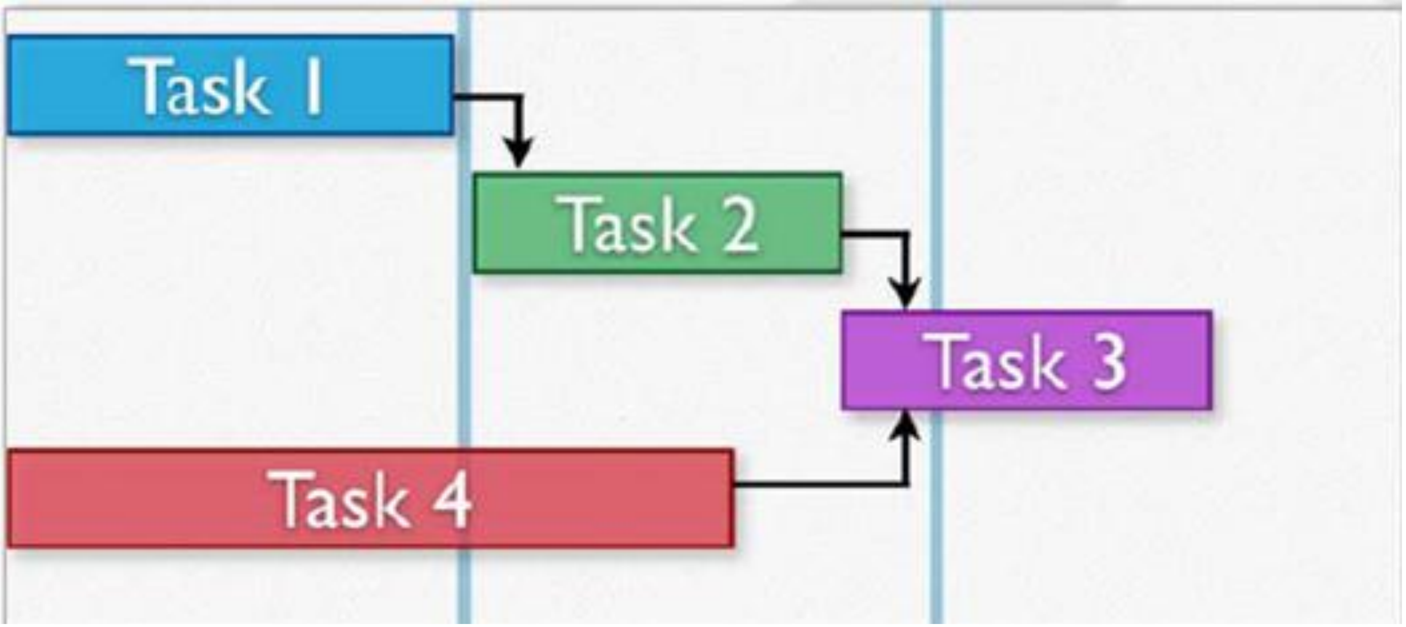
Russian cosmonauts used crayons.

Moral: Specify what you **want** to achieve, **not how** to achieve it.

working with dependencies

Working with dependencies

Dependencies are a fact of life

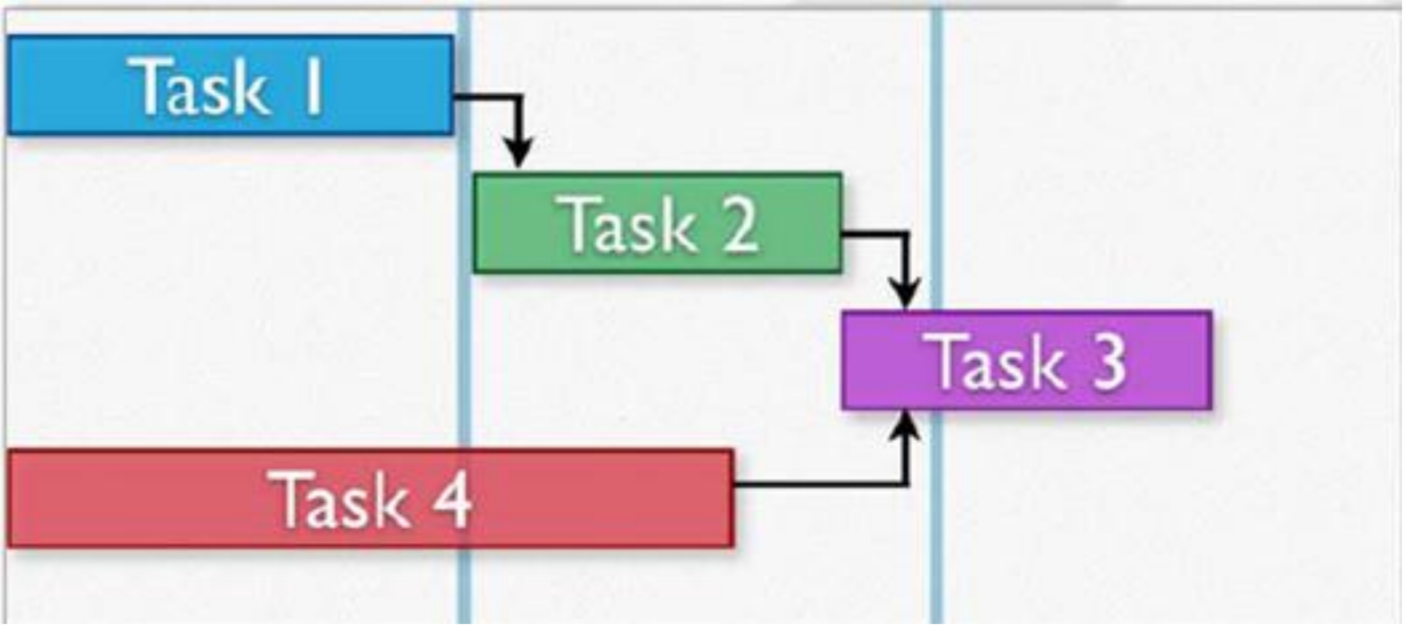


Dependencies Tips and Tools

- Understand the "Last Responsible Moment" for each dependency
- PO and Scrum Master work with other teams to address dependencies
- Use Release Plan to set/revise expectations
- Remember – Agile teams can easily respond to changes in dependency dates before the LRM

Working with dependencies

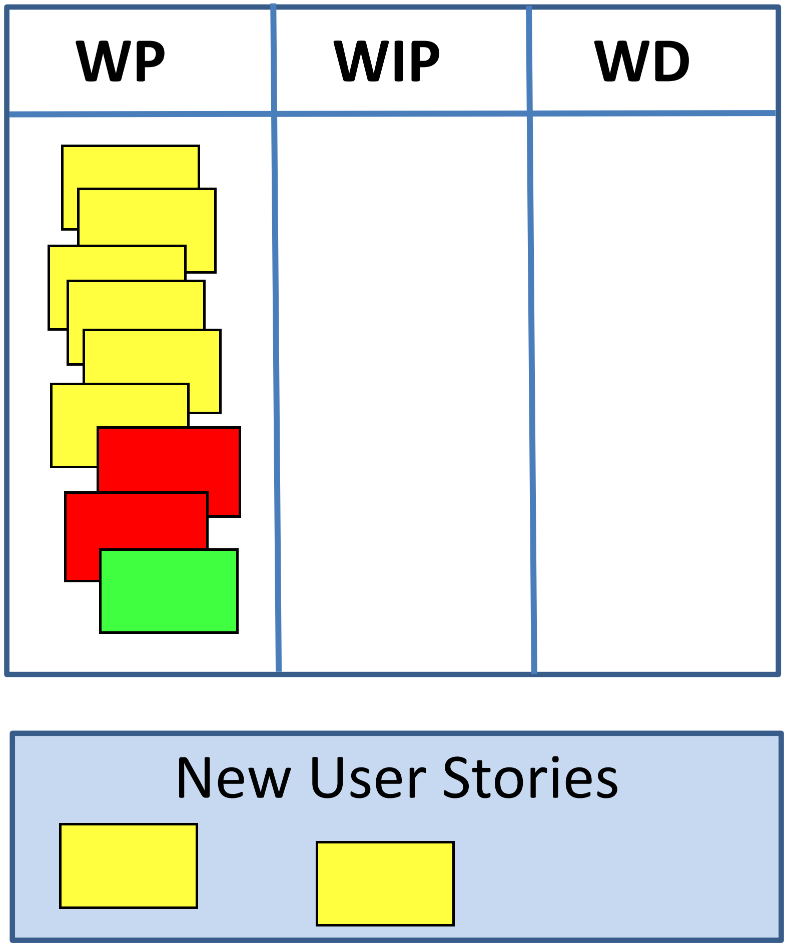
How do you deal with dependencies?



Sprint Planning

Sprint Planning Review

Sprint Planning



WP

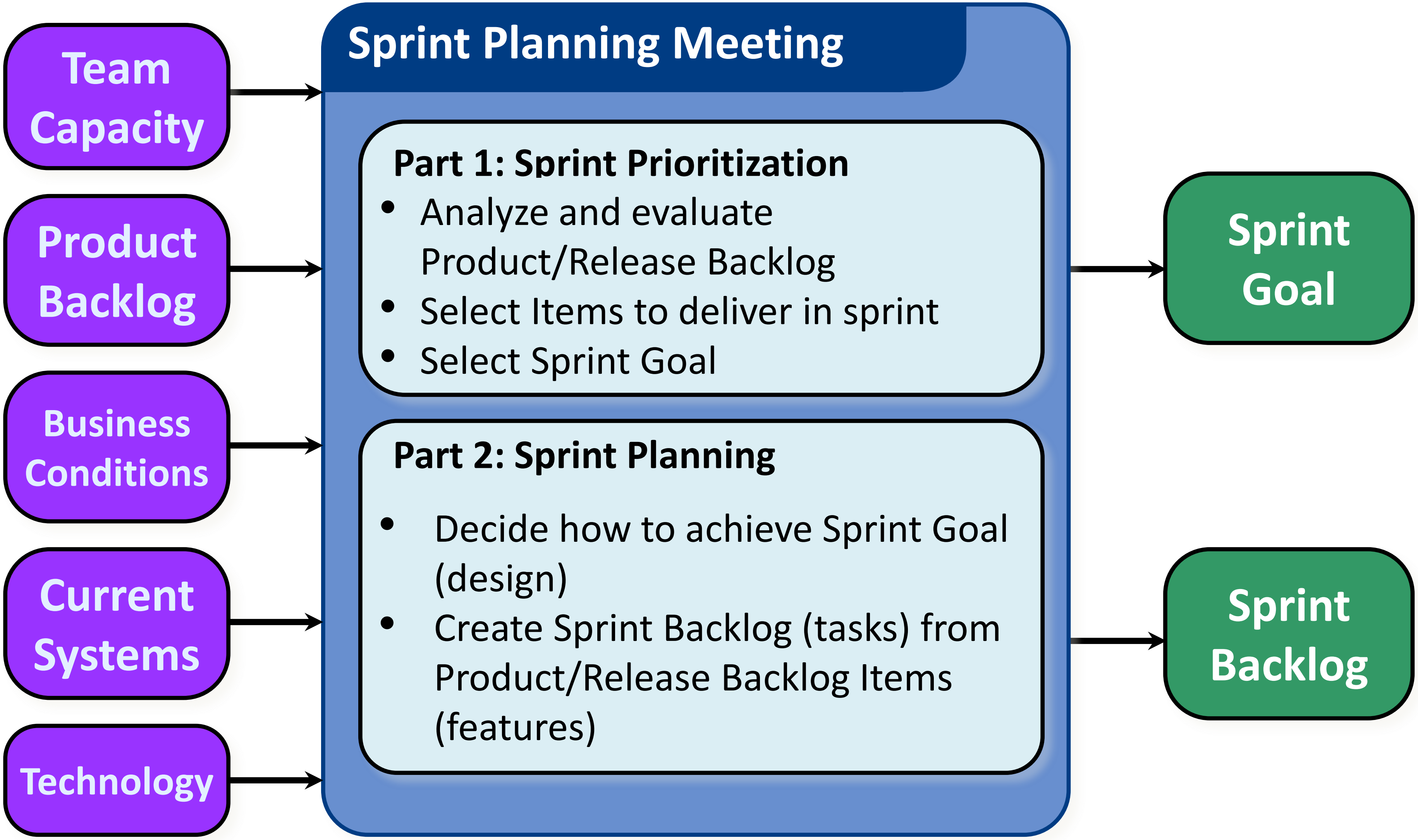
WIP

WD

New User Stories

Input	Release Backlog
Output	**Sprint Backlog, Definition of Done, Architecture Spike, Information Radiator**
Who	Delivery Team

Sprint Planning



Team Capacity

Product Backlog

Business Conditions

Current Systems

Technology

Sprint Planning Meeting

Part 1: Sprint Prioritization

- Analyze and evaluate Product/Release Backlog
- Select Items to deliver in sprint
- Select Sprint Goal

Part 2: Sprint Planning

- Decide how to achieve Sprint Goal (design)
- Create Sprint Backlog (tasks) from Product/Release Backlog Items (features)

Sprint Goal

Sprint Backlog

Optimize Sprint Planning

Prepare

Before the meeting:

- **ONLY** add Stories that pass **ALL** the decision filters
  - For the product
  - For the release
- “Does this pass all the decisions filters?”

Optimize Sprint Planning

Prepare (con't)

Before the meeting:

- Sequence the backlog based with **highest value** stories first
- Ensure stories are sufficiently **small** for inclusion in the sprint

Optimize Sprint Planning

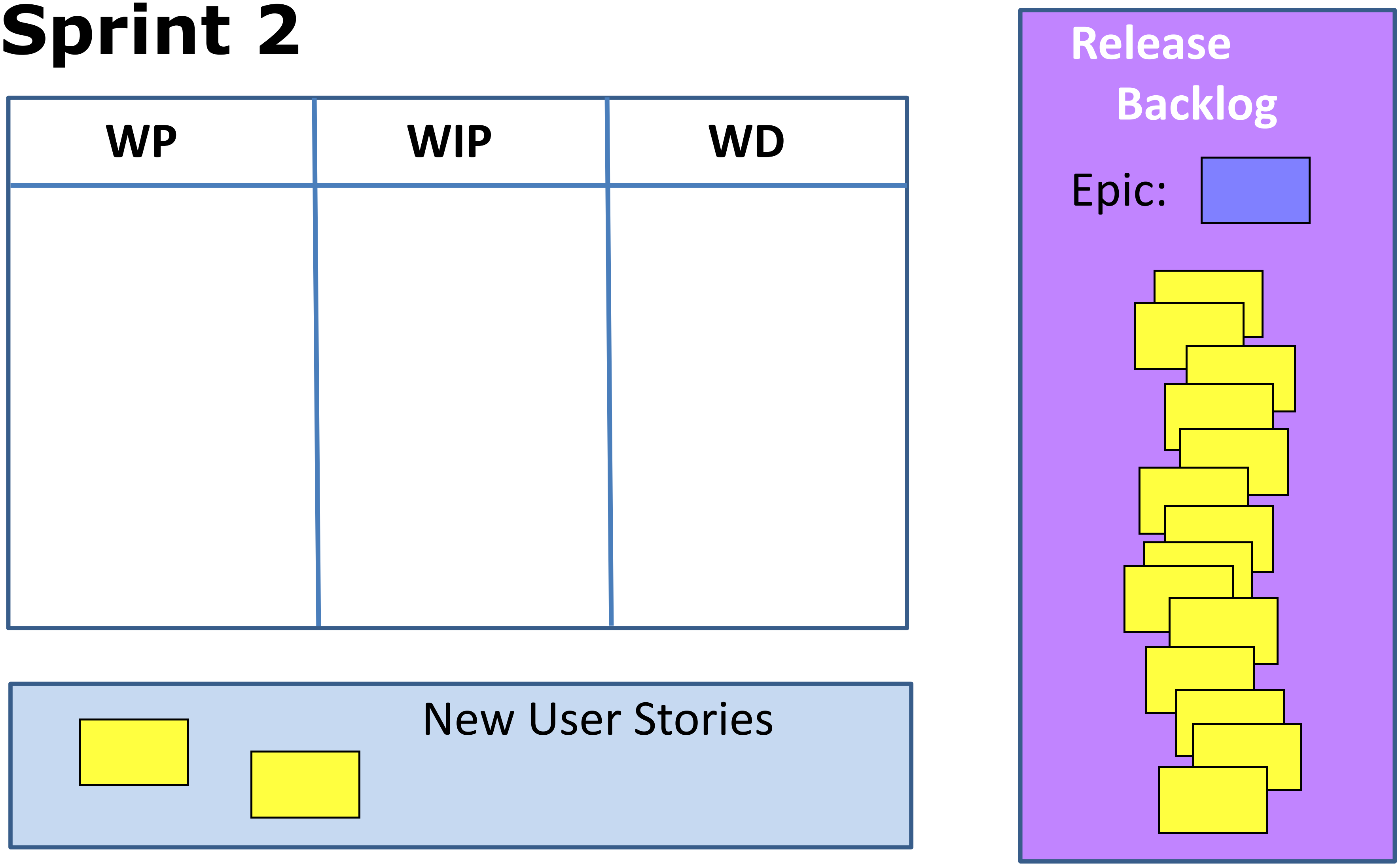
Commitment

In the meeting:

- Starting with the **highest value** stories, Delivery Team suggests reordering based on **feasibility**
- Team commits to delivering stories using **commitment based planning**

Commitment Based Planning

Sprint 2



WP	WIP	WD

New User Stories

Release Backlog

Epic: 🟦

Commitment

- Team is not told what to deliver, team commits to delivering a set scope
- Once this commitment is made, no one can change it but the team
- Everyone on the team is responsible for meeting the commitment

PO’s Role at Sprint Planning

- Make sure the team is attacking the latest view of the highest value stories
- Guide the team on their understanding of the requirements
- Reinforce the business goals
- Share the latest market view

PO’s Role in Sprint Planning

- Can send a proxy and **is clear** that proxy can make decisions and re-prioritize
- PO is the **bottleneck.** Team can’t proceed with planning without the PO
- Do not tell the team what they need to do in the sprint. Team decides.

I have to have it all by <date>!

Make sure you are clear on **why**

Managers Want Certainty

- It is impossible to get it – we don’t know what we don’t know.

- Establish first pass of release plan
- Reset expectations after each sprint
- Pro-actively manage risk to minimize cone of uncertainty to set delivery date

proactively
managing risk

uncertainty

complexity

uncertainty

market uncertainty

technical uncertainty

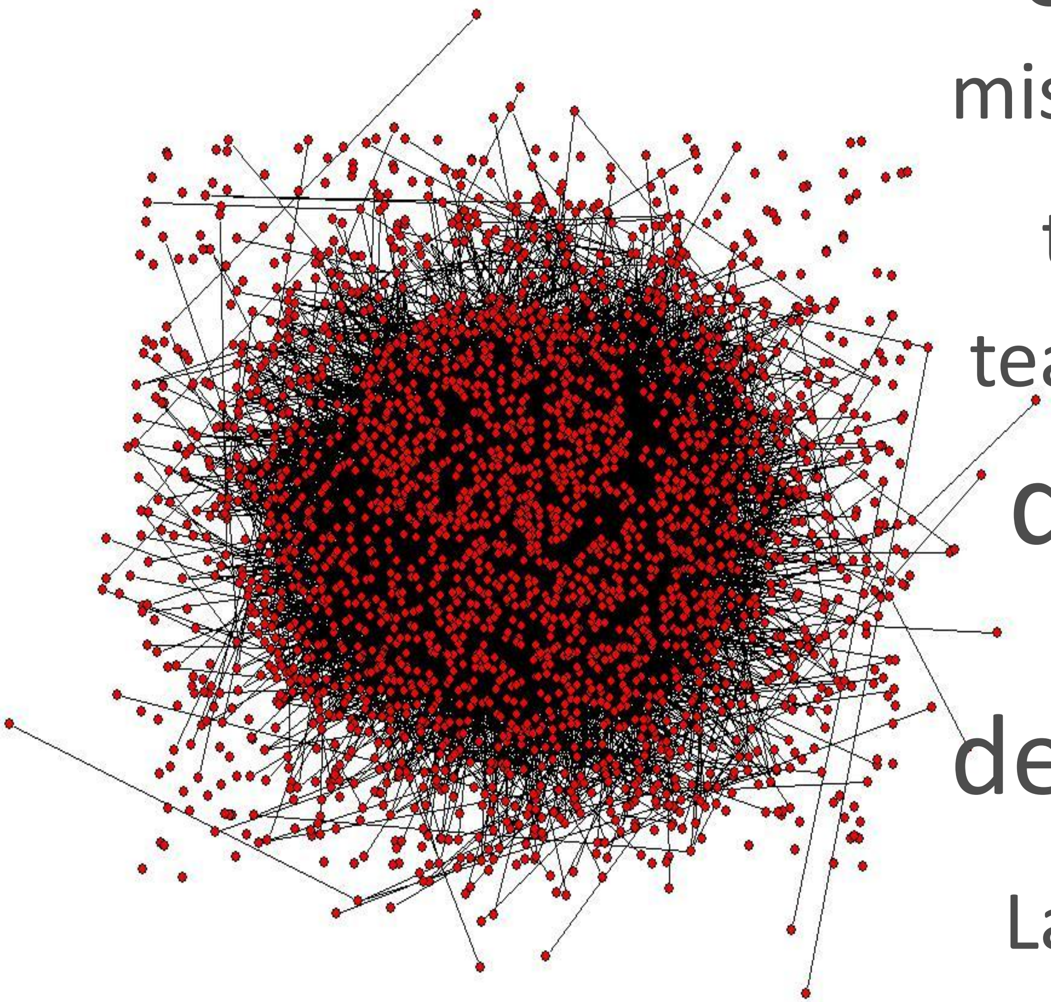
project **duration**

number of customers

scope dependencies

flexibility

complexity



team size

Culture

mission criticality

team location Time Zones

team maturity

domain

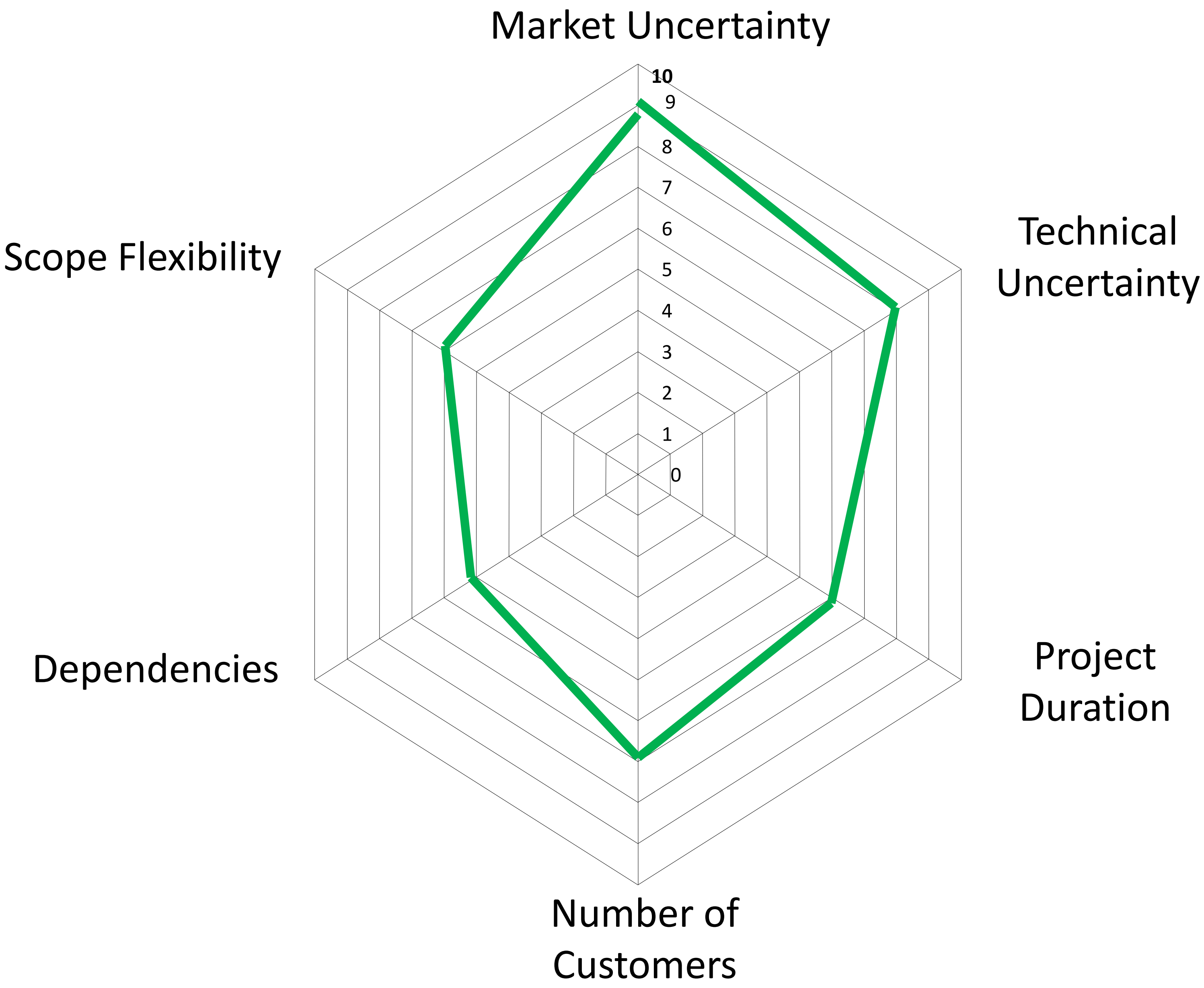
knowledge gaps

dependencies

Lack of Trust

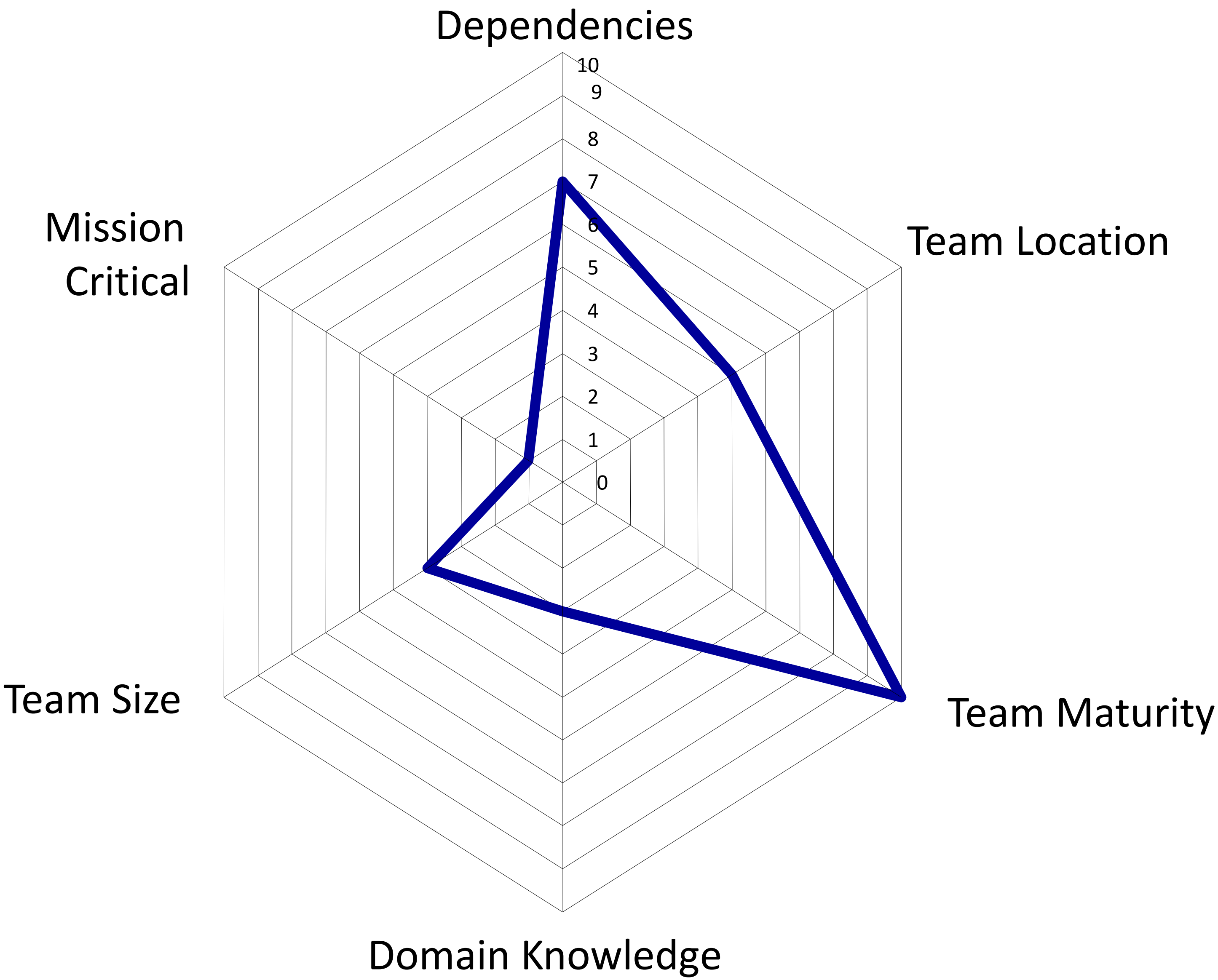
Uncertainty Assessment

Attribute	*Score*
Market	**9**
Technical	**8**
Project duration	**5**
Number of customers	**6**
Dependencies	**4**
Scope flexibility	**6**


Market Uncertainty
Technical Uncertainty
Project Duration
Number of Customers
Dependencies
Scope Flexibility
10
9
8
7
6
5
4
3
2
1
0

Complexity Assessment

Attribute	Score
Dependencies	**7**
Team Location	**5**
Team Maturity	**9**
Domain Knowledge	**3**
Team Size	**4**
Mission Critical	**1**


Dependencies
Team Location
Team Maturity
Domain Knowledge
Team Size
Mission Critical
10
9
8
7
6
5
4
3
2
1
0

One way to determine score

	Low Impact	Medium Impact	High Impact
Low Likelihood	**1**	**4**	**7**
Medium Likelihood	**2**	**5**	**8**
High Likelihood	**3**	**6**	**9**

Some attributes apply to both uncertainty and complexity

Impact of Dependencies on Risk

- Those you need results from (uncertainty)
  - Weigh their ability to execute
  - Weigh their reliability in the past
- Those who need deliverables from you (complexity)

➡ For your project…

Identify Dependencies

- Identify who you need results from (uncertainty)
  - Weigh their ability to execute
  - Weigh their reliability in the past
- Identify who needs deliverables from you (complexity)

Assess sources of risk

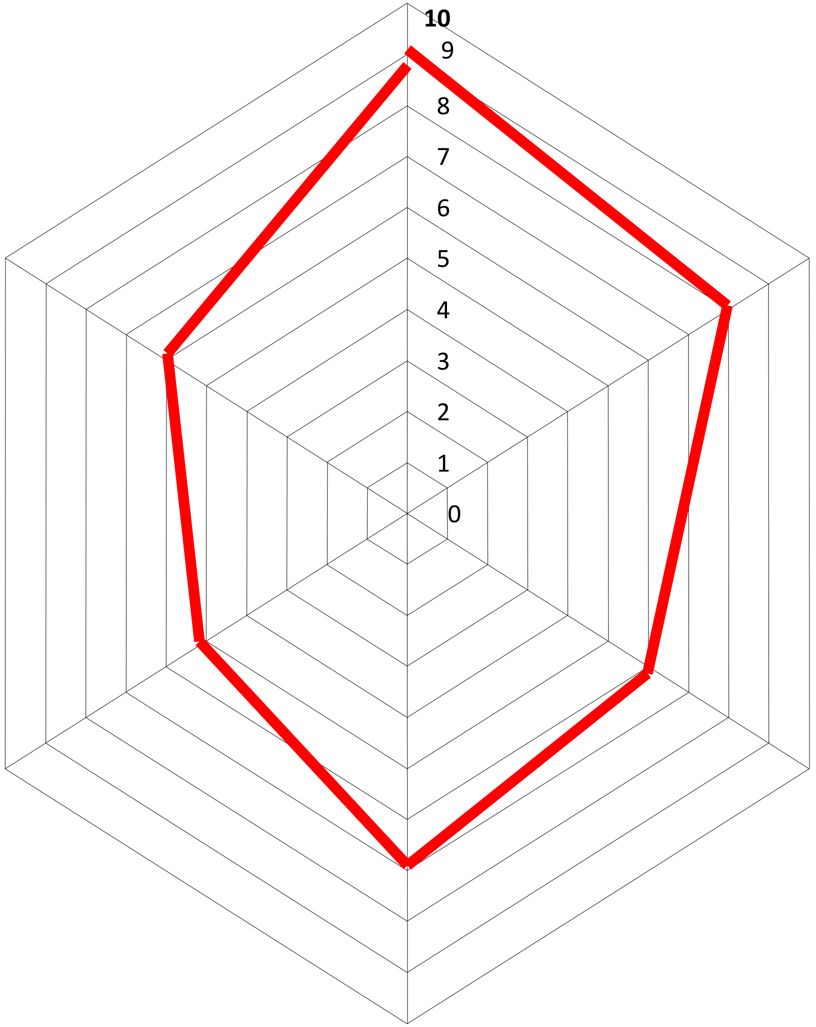
- Score your uncertainty risks.
- Score your complexity risks
- Chart the **top six** risks on a spider diagram

Uncertainty Assessment

Attribute	*Score*
Market	
Technical	
Project duration	
Number of customers	
Dependencies	
Scope flexibility	

Complexity Assessment

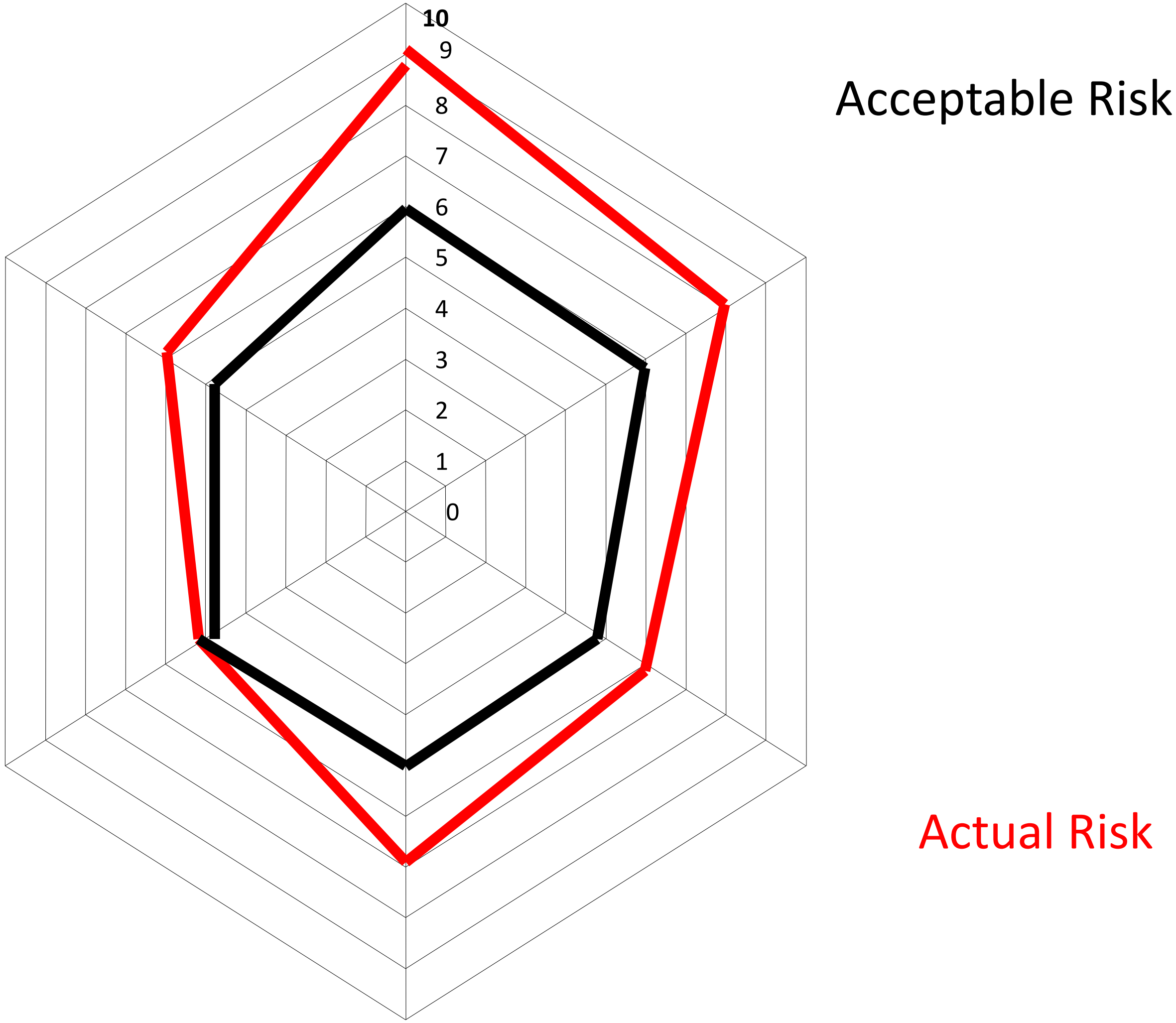
Attribute	*Score*
Dependencies	
Team Location	
Team Maturity	
Domain Knowledge	
Team Size	
Mission Critical	


10
9
8
7
6
5
4
3
2
1
0

Actual vs. Acceptable

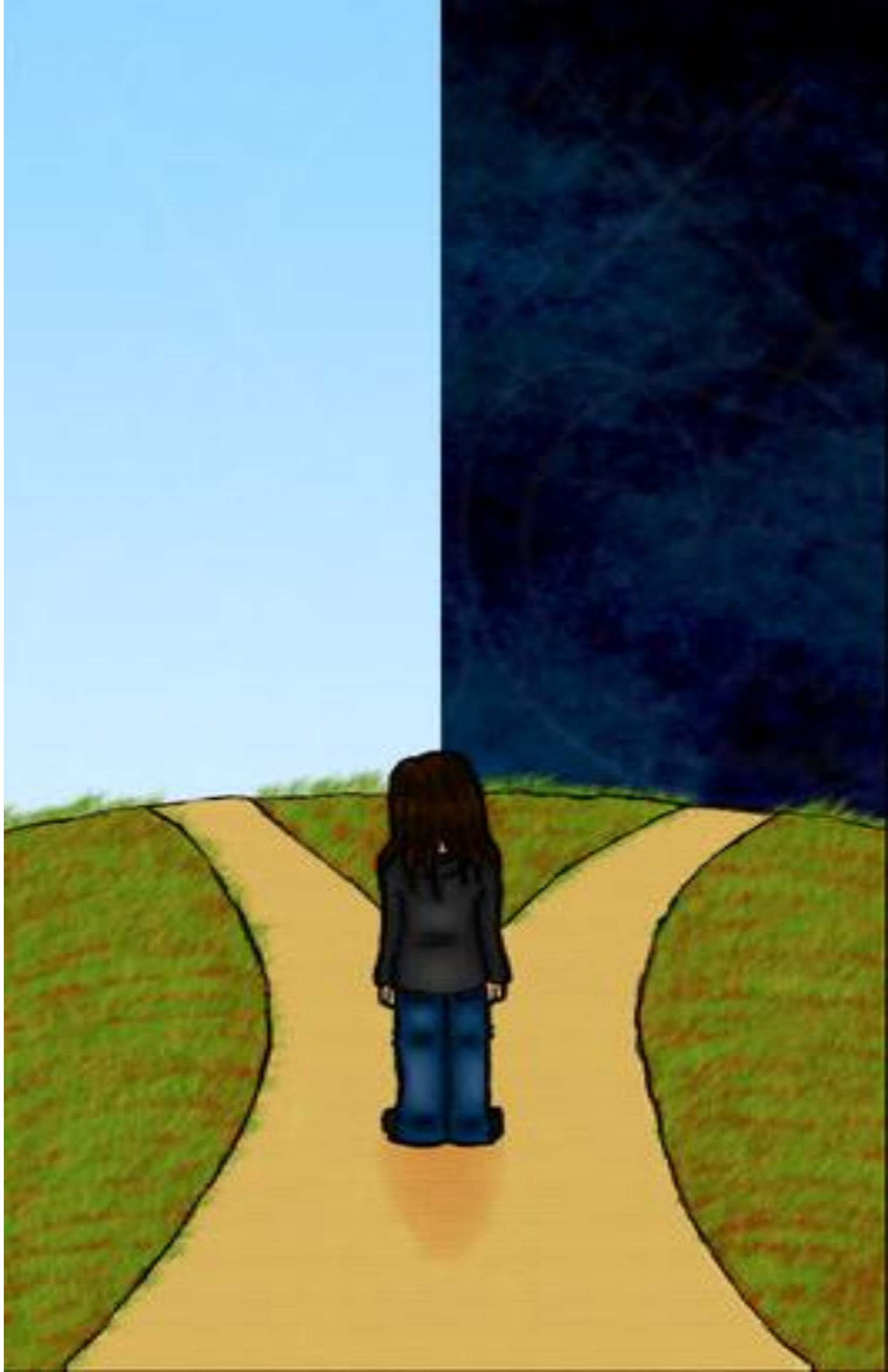
- What is your acceptable uncertainty score?
- What is your acceptable complexity score?

Actual vs. Acceptable



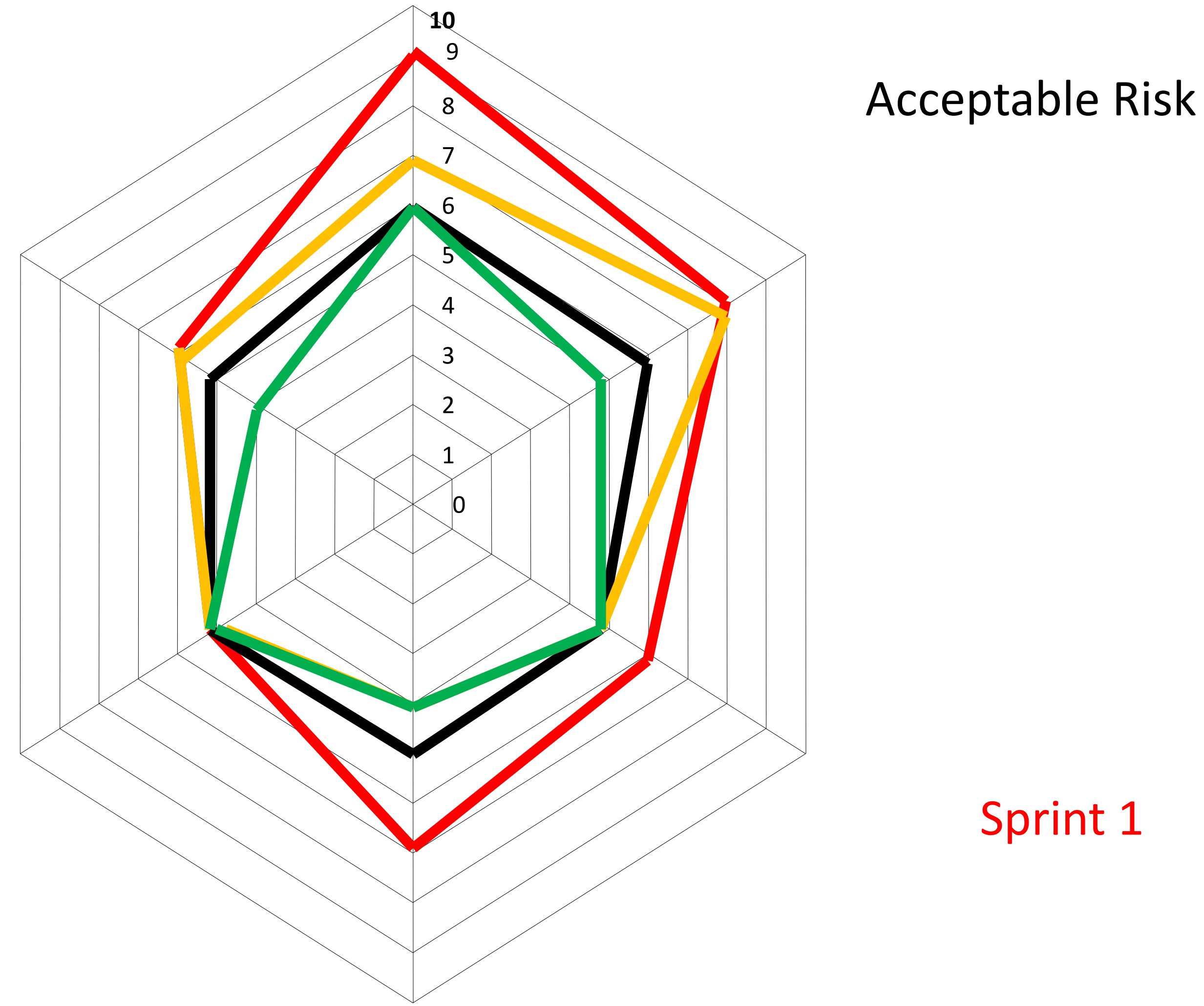
making commitments

➡ What must be true **before** you can make a commitment?



When does it make sense to make a commitment?

Cone of Uncertainty for Your Project



Now we can ask…

Pro-active risk management helps us to answer:

- Do we think we are doing the right things?
- Do we think we can commit to delivery?

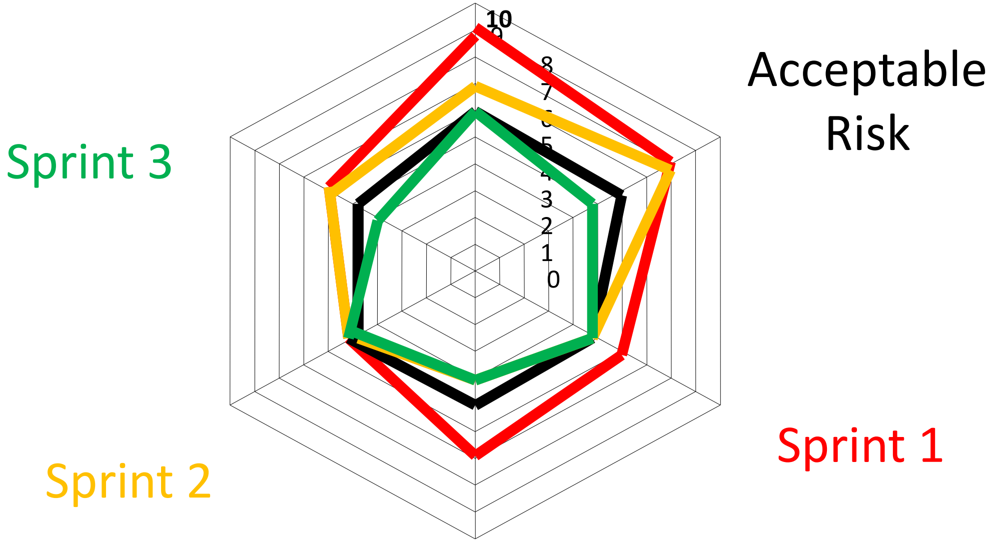
If we cannot answer **yes** to both questions, what must we do in the next sprint to help us with these questions.

Make Risks Visible

- How can we use our risk profiles (for example, spider charts) to visibly communicate risks?
- How can we increase the visibility of our risk mitigation plans, levers and lever expiration?
- How can we make our shrinking cone of uncertainty visible?

Exercise

Develop visible displays you will add to your project's information radiator



customer feedback

Getting Customer Feedback

- Create a mechanism which allows customers to download demo level code from a secure area.
- Recruit key representative customers who are keen to help shape the new solution
- Put in place a non-disclosure agreement

Getting Customer Feedback

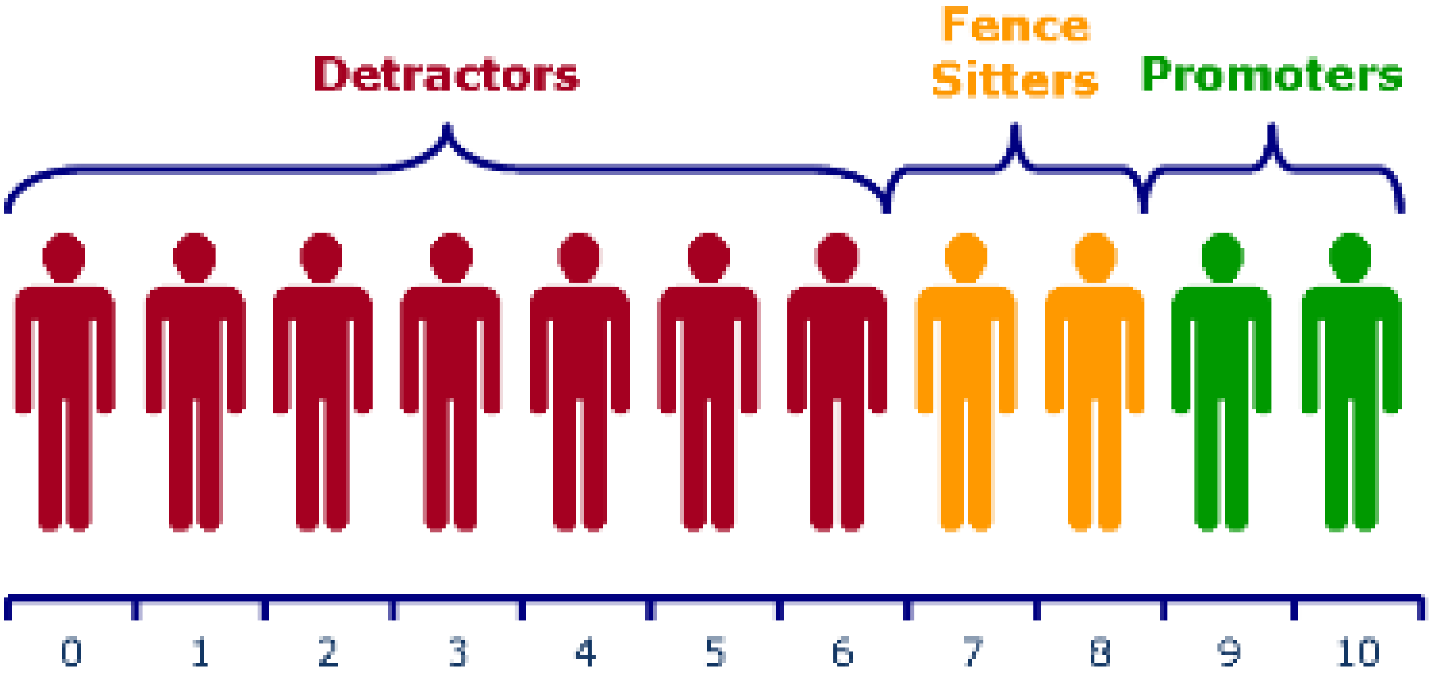
- Position them carefully. Not all they see will appear in the product.
- Give them access to the forum and ask for their input.
- Give the development team access and make sure they engage in dialog with the customers.

Getting Customer Feedback

High Fidelity Prototyping

- Build the minimum function necessary
- Focus on UX: Can users figure out how to use it and want to use it?
- Test it on real customers

Net Promoter Score



Net promoter score = % promoters - % detractors*

Sprint Support

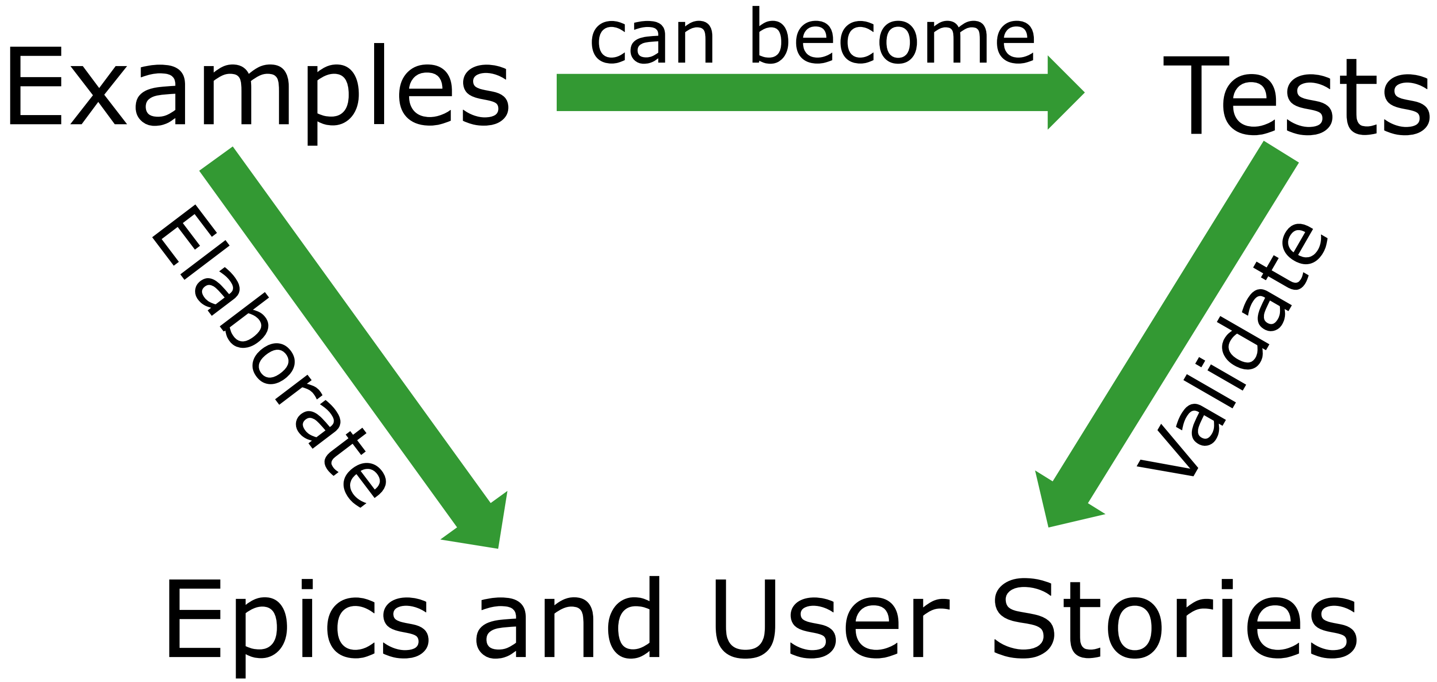
Sprint Support

- Attend the daily stand ups
- Just listen!
- Answer team questions as fast as possible (don’t be the bottleneck)
- Provide examples for developing the acceptance criteria
- Work with the Scrum Master to unblock the team when needed
- Give the team air cover

Acceptance Criteria

Acceptance Criteria Review

Writing Acceptance Criteria



Examples

can become

Tests

Elaborate

Validate

Epics and User Stories

What does ‘Testable’ mean?

Format:

- **Given** <Precondition>
- **When** <Action>
- **Then** <Result>

Example:

- **Given:** User account exists in system
- **When:** User enters existing user name and incorrect password
- **Then:** System displays: “You provided an incorrect password”

Examples as Process

As a presenter I want to provide a copy of my slides so that attendees can refer to the slides before and/or after the presentation

When I associate a file
< 5MB
Then File is uploaded
And system notifies me of success

When I associate a file
> 5MB
Then System indicates file is too large

Examples as Tables

As the chair, I want to be able to accept a 10 minute talk session from a presenter with two accepted sessions so that I can have popular presenters giving lightning talks.

Accepted Session 1 type	Accepted session 1 duration	Accepted session 2 type	Accepted Session 2 duration	Session to accept type	Session to accept duration	Can Accept Session?
Talk	60	Tutorial	90	Talk	60	No
Talk	60	Tutorial	90	Tutorial	10	No
Talk	60	Tutorial	90	Talk	10	Yes
Talk	10	Tutorial	90	Talk	60	Yes
Talk	10	Tutorial	90	Tutorial	10	Yes
Talk	10	Tutorial	90	Talk	10	Yes

Acceptance Criteria Tools and Tips

- Use the same examples to communicate requirements & describe tests
- Examples should be real world
- Use combination of examples to describe rules

Acceptance Criteria Tools and Tips

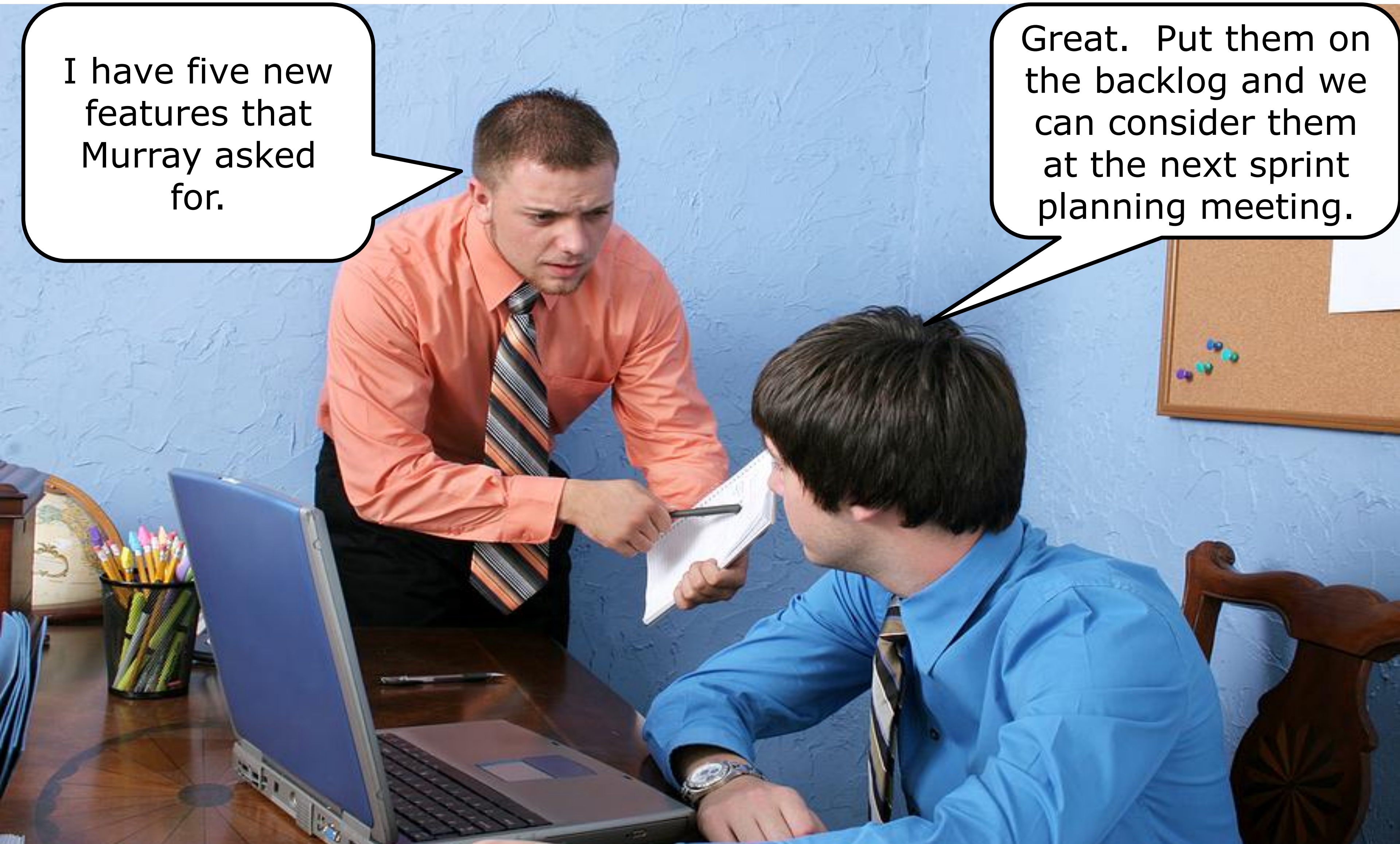
Examples to consider:

- Normal, edge, and negative cases
- Abnormal but reasonable use
- Abnormal and unreasonable use

➡ Other examples?

Sprint Blowup Prevention

Has this ever happened to you?


I have five new features that Murray asked for.
Great. Put them on the backlog and we can consider them at the next sprint planning meeting.

Or this?

I must have this in the backlog!
How does it meet our decision filter?

Blowup Prevention Tools and Tips

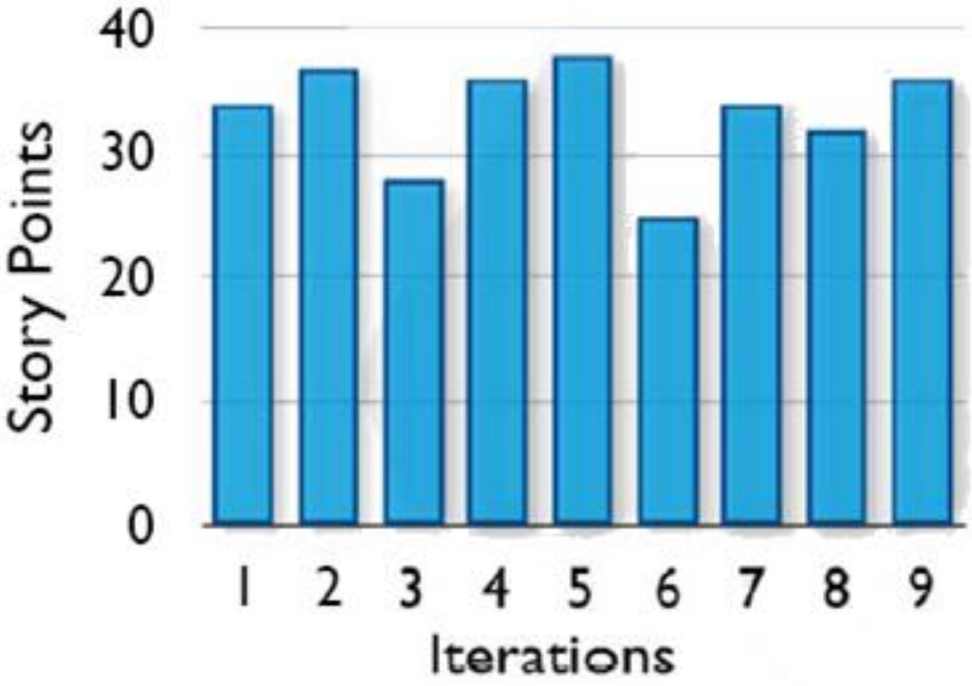
- If you can’t wait for changes, is the sprint duration too long?
- Don’t say no to your stakeholders, just remind them they will have to wait until the next sprint
- Direct changes to the release backlog
- Remember the decision filters

Velocity

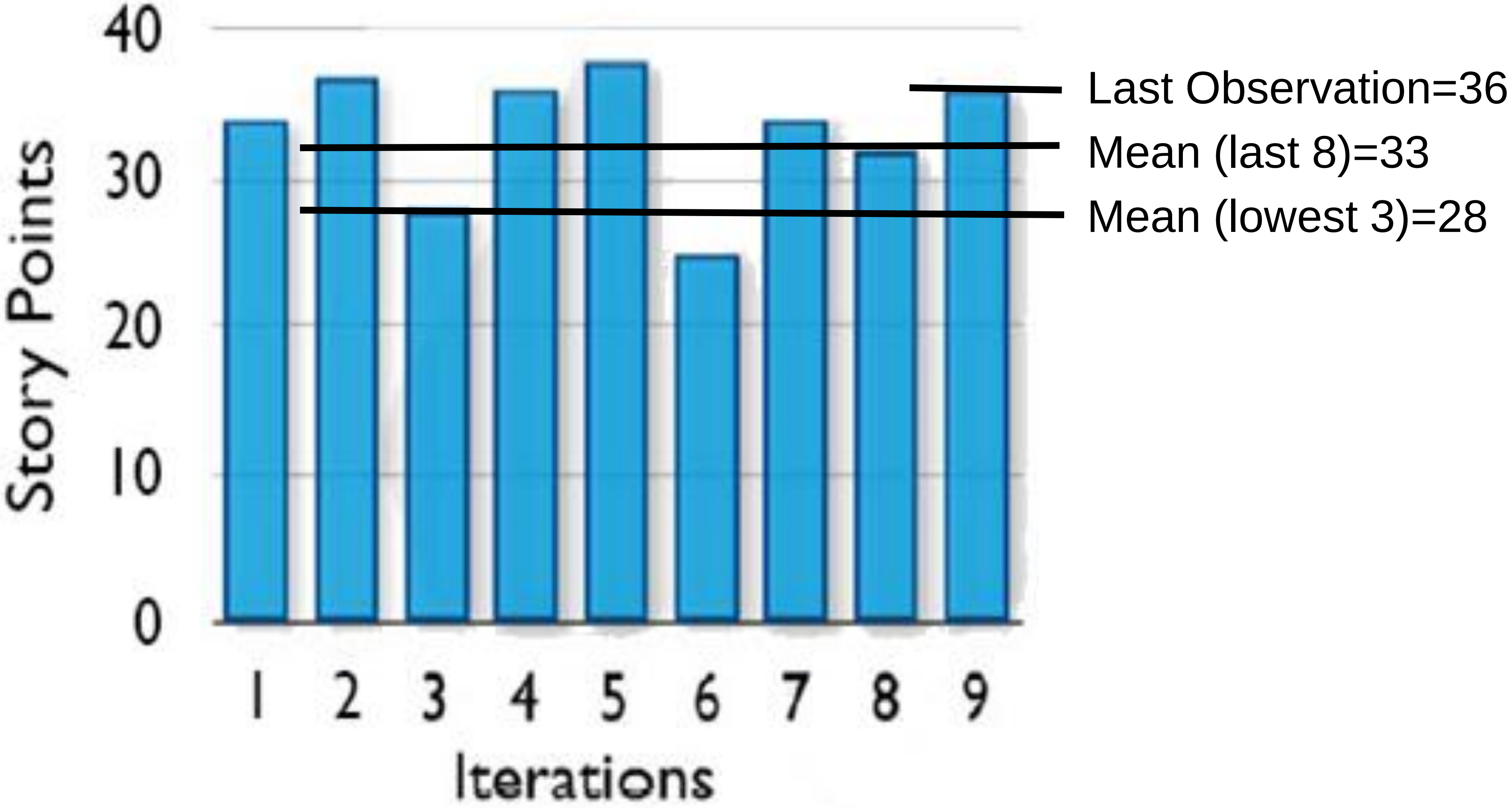
Velocity Review

Velocity

- **Long-term** measure of work completed in sprints
- A **GUIDE** not a goal**.**



Track Velocity Multiple Ways



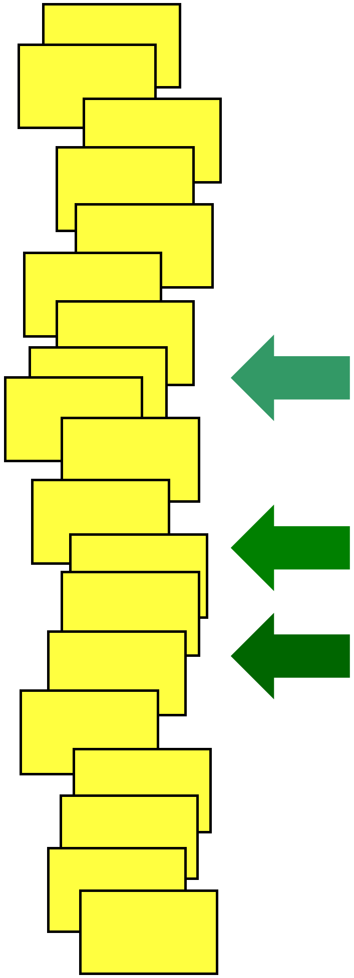
Last Observation=36

Mean (last 8)=33

Mean (lowest 3)=28

Extrapolate from Velocity

Assume five sprints left



Finish here at **lowest** velocity: 5x28

Finish here at **average** velocity: 5x33

Finish here at **current** velocity: 5x36

Velocity Tips and Tools

- Use the 60/40 rule
- Example:
  - 9 months to deliver
  - 6 months of activities planned
- This becomes a decision filter:
  - “Can we get this done in 6 months”
- Integrity matters
- Don’t **‘sort of’** pass the filters!

You get what you measure

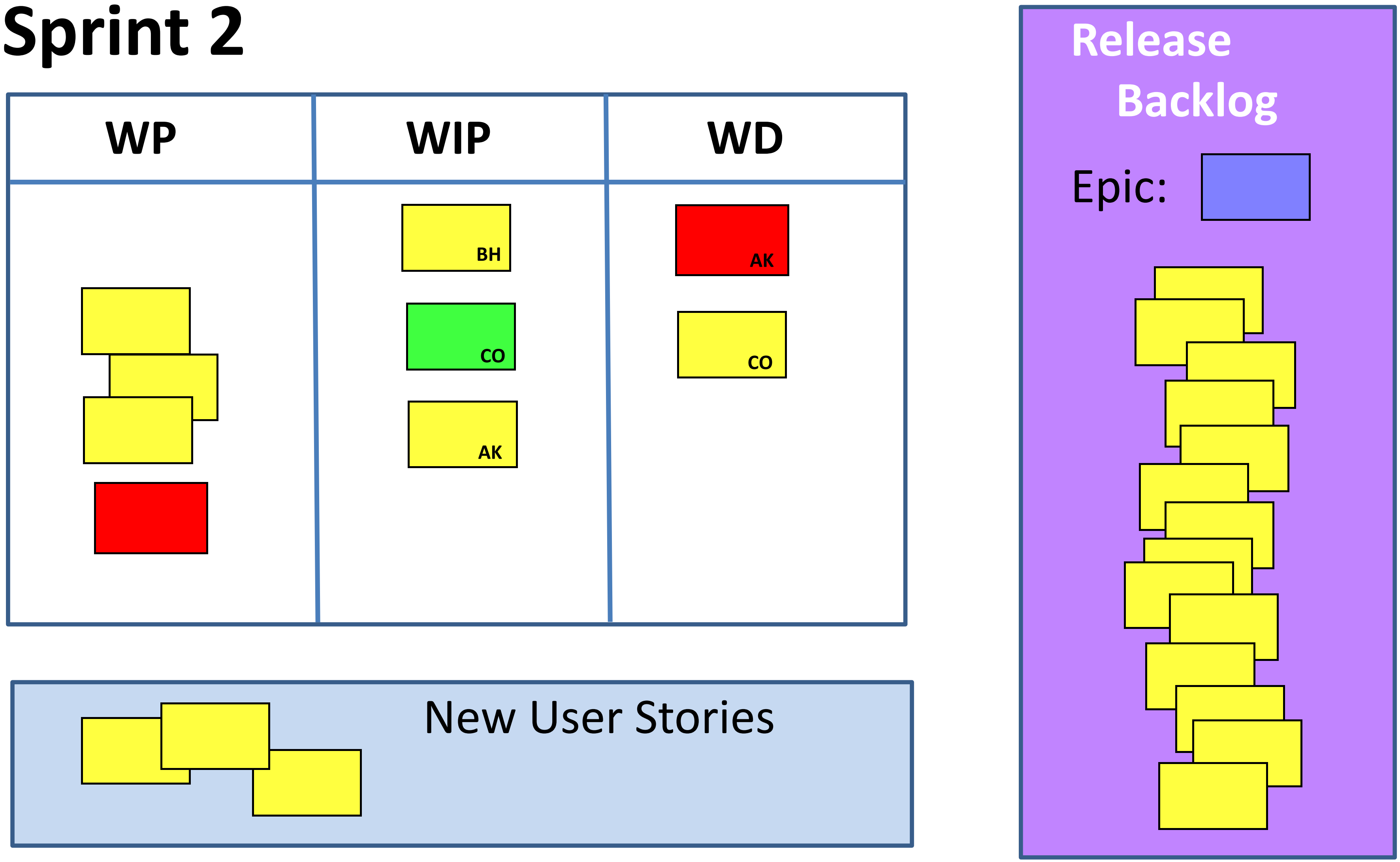
Watch out for Vanity Metrics

Make it Visible

of
ommunity
are!
Learning Science Research
Encourage research on value of technology tools to teaching & learning
opportunity for debate
Event where teachers from different areas of UBC can discuss their use of teaching and learning technology
er
e of
ty on
npus

Visibility Review

Information Radiator



Sprint 2

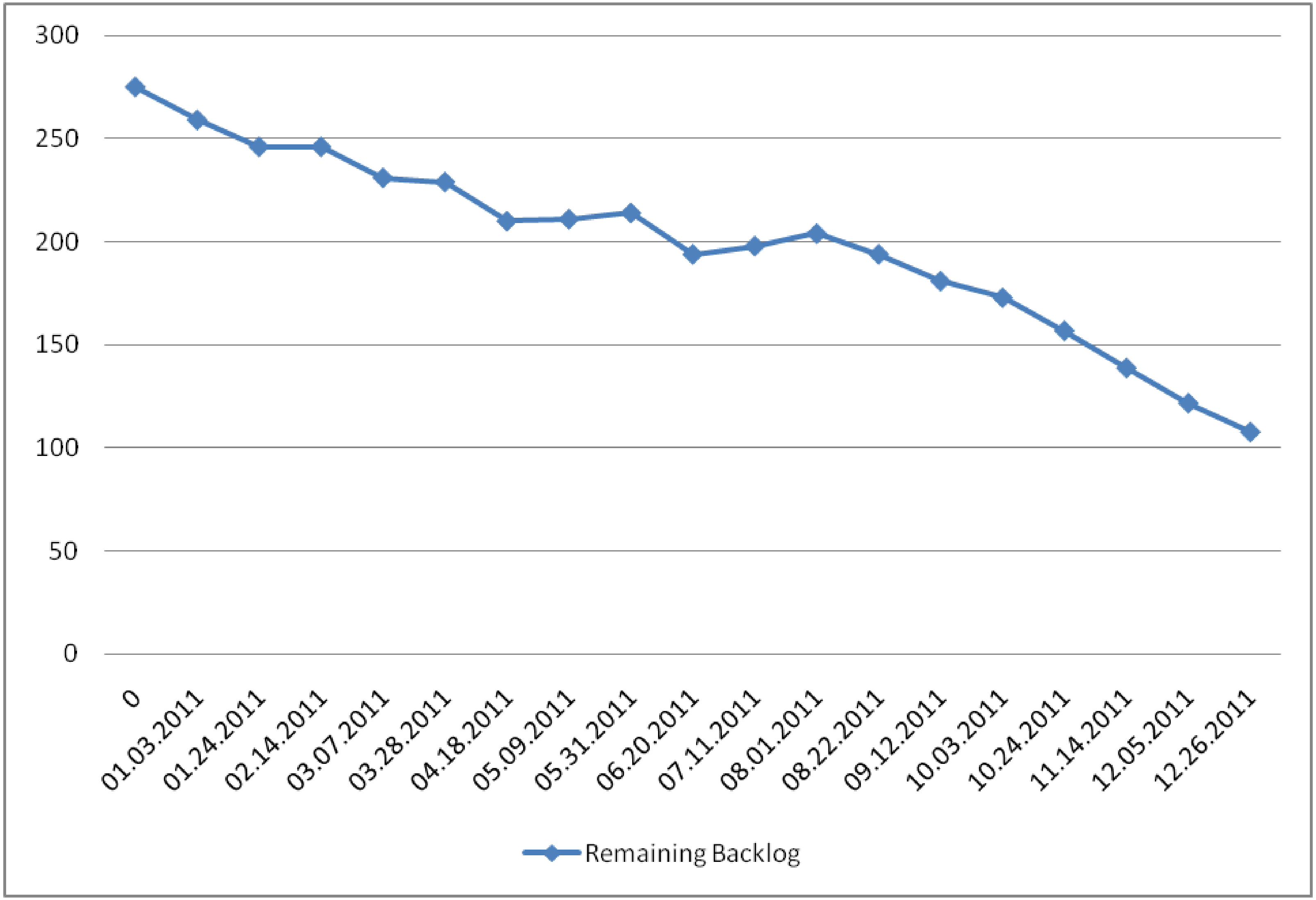
WP	WIP	WD
	BH	AK
	CO	CO
	AK	

New User Stories

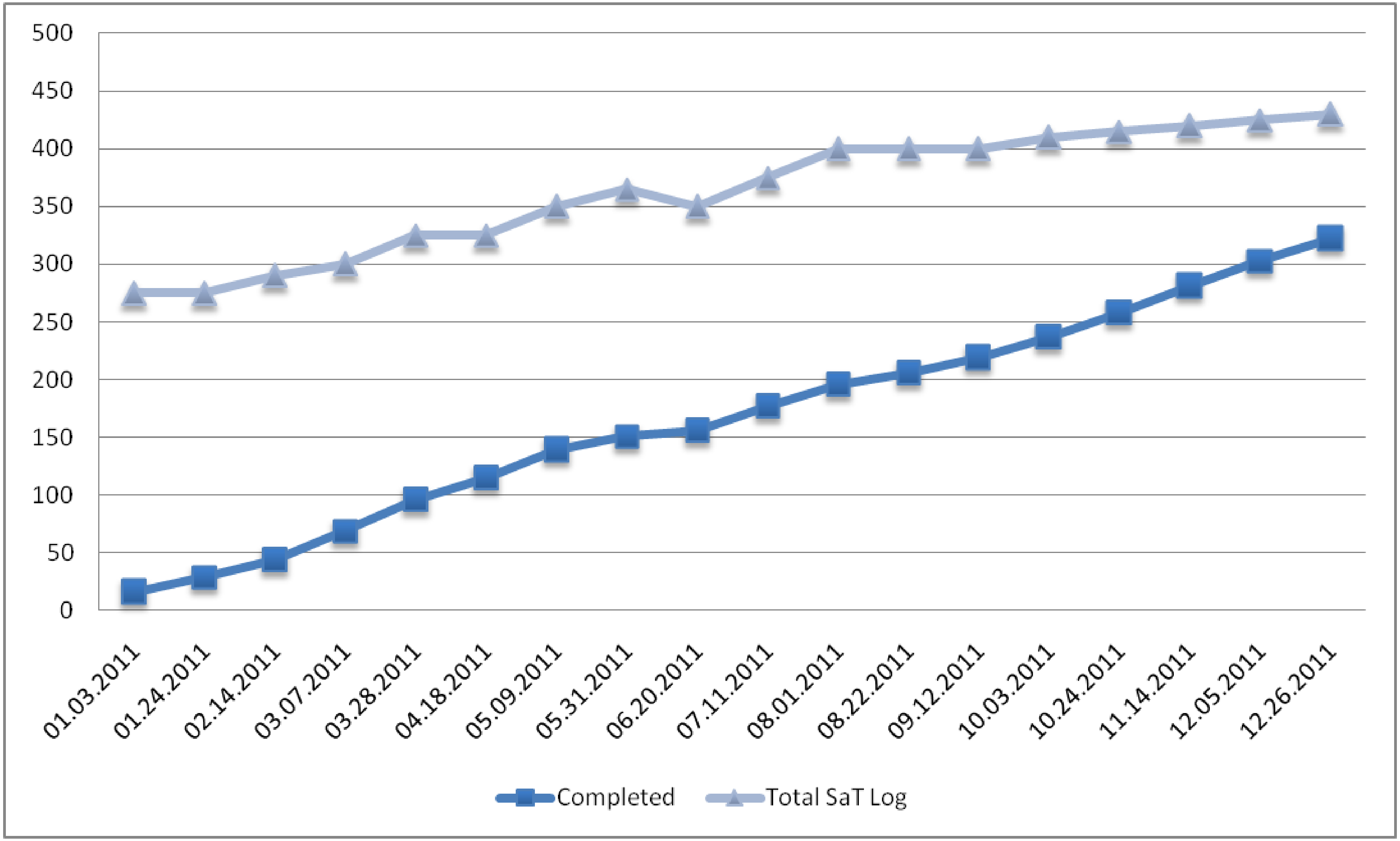
Release Backlog

Epic: 🟦

Tracking Progress – Burn Down



Tracking Progress – Burn up



Make it Visible to Everyone!

- Information Radiator
  - Scrum Backlog and progress
  - Dependencies
  - Actions from retrospective
  - Spider diagrams (risk)
  - Burn up/burn down chart
- Decision Filters
- Billboard

Make it Visible

- Rules of Engagement
  - Definition of Done, Done, Done
  - Standup meeting info
  - Coding standards
- Product Owner contact info and backup contact
- Team and Scrum Master contact info
- What else???

Visibility Tips and Tools

- Information Radiator:
  - Trello.com a free web-based tool where you can drag and drop cards
- Set up a customer forum for comments and inputs
- Keep wireframes and screen shots on the walls
- Webcam
- Wiki

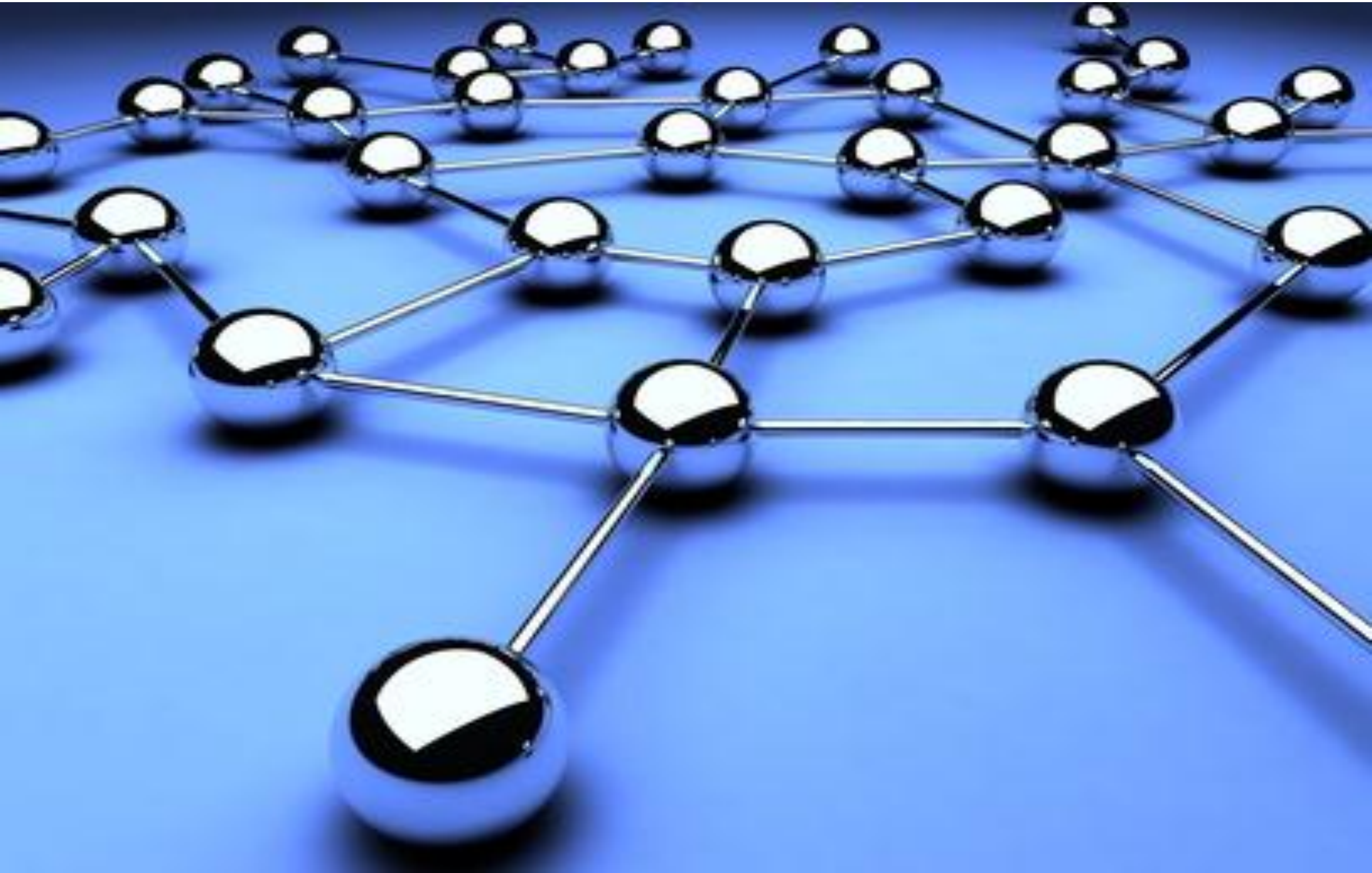
Sprint Demo

PO’s Role at the Demo

- To get an understanding of the progress and out look
- To give the team feedback and steering on the stories delivered so far
- To build quality and delivery ownership in the team

Logistics Tips and Tools

distributed teams



Distributed Teams

- Sort out the communications
  - Video
  - Audio
  - Community space
  - email
- Build aligned vision and goals
  - Focus especially on cultural differences where teams are in different countries
  - Personal relationships across teams

Distributed Teams

- Clear organization and structure - understood by all
  - Clear responsibilities for each team
- Inter-team
  - Project Management / Co-ordination = "Scrum of Scrums"
  - Team to team technical interfaces
  - Overall status reporting

Distributed Teams

- Common Tools
  - Development
  - Communications
- Increase frequency of integration
- Set up alert system: Push not pull

Distributed Teams Tips and Tools

- A Scrum Master at onshore and offshore sites
- Scrum of Scrums
- PO is available for some time to each team
- Post PO availability time and contact
- Co-locate entire team for one sprint
- Project wide wiki

Distributed Teams Tips and Tools

- Encourage informal communications across members of distributed teams
- Scrum of Scrum meeting rotates the time zone meeting time
- Create a “community of practice” across geography teams to share resources

multiple teams – same project

Multiple Teams Tips and Tools

- Set a process for making decisions
- Sync the sprints on the same timeframe
- Scrum of Scrums with Development Managers as well as Scrum Masters
- Create a communications plan and process

becoming a great Product Owner

Leadership Review

autocracy dampens people's creativity and motivation

- Ricardo Semler,
  The Seven-Day Weekend

give up command

and control

free team to question, analyze and investigate

the opposite of **control** is

discovery

a place where

people want to be

people

have

what they need

to

succeed

People don’t resist change;
they resist being changed.

- Peter Scholtes

Great Product Owner Tips

- Get developers in front of users and customers
- Involve developers in exploring what is becoming possible technically
- Involve development (architect and UX) from the beginning of product discovery

Great Product Owner Tips

- Keep the focus on minimal viable product
- Minimize churn (changing product definition and changing requirements)
- Answer the development team’s questions as fast as possible

brainstorm scenarios

group and

prioritize

Work through scenarios

full time job

your Product Owner
take aways

References

- *Inspired: How to Create Products Customers Love*, Marty Cagen
- http://svpg.com/examples
- Linkedin Agile and Lean Software Development Group
- http://www.goodagile.com/distributedscrumprimer/
- *Agile Adoption Patterns: Roadmap to Organizational Success*, Amr Elssamadisy