IVAR JACOBSON INTERNATIONAL

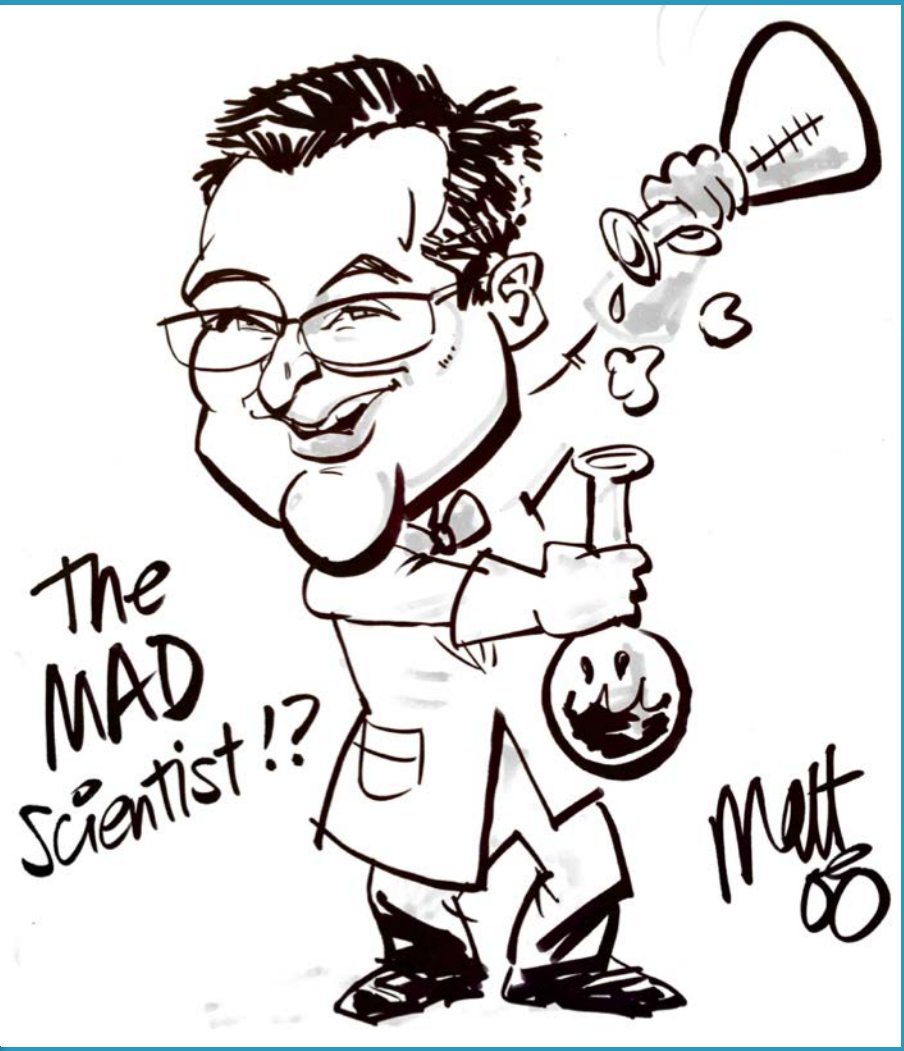
Creating winning teams.

PRAGMATIC PRODUCT MANAGEMENT: HINTS AND TIPS FROM THE FEATURE COAL FACE

www.ivarjacobson.com

WARNING!

This session will contain some experiments!

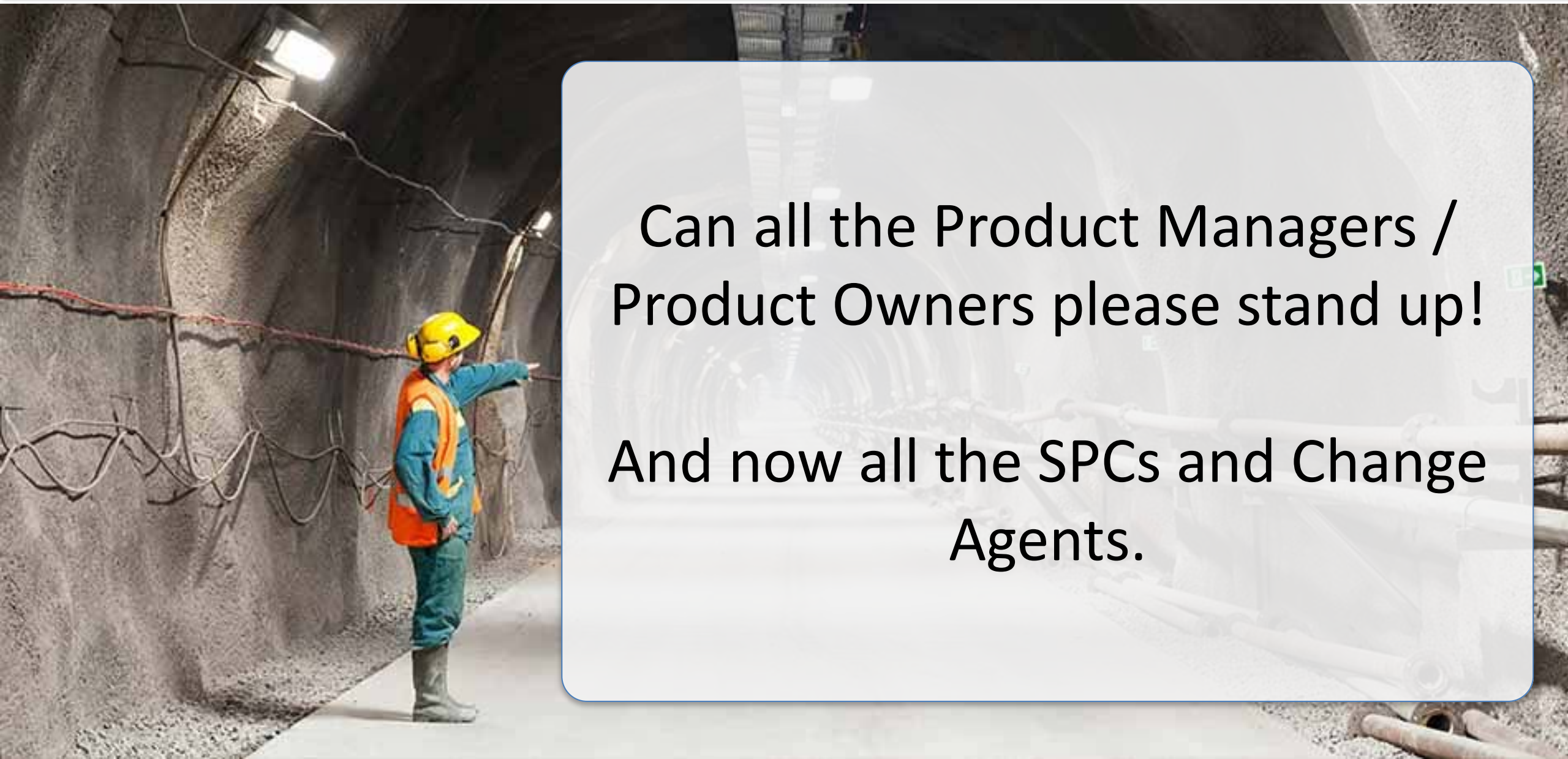


Outline

- Check-In: The lifecycle of a Feature
- On your marks: Identifying the best Features
- Get set: The power of PI Planning
- Go: Shipping the right Features
- Winning the race: Delighting your customers
- Check Out: Sticking to our principles

Check In

Check In – Who's in the audience?



Can all the Product Managers / Product Owners please stand up!

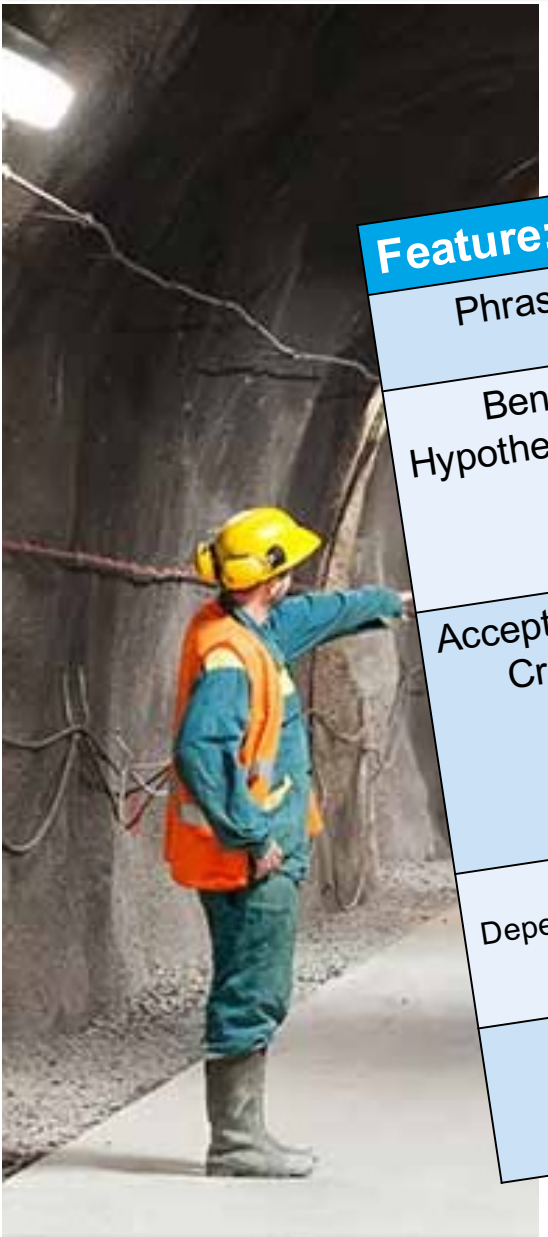
And now all the SPCs and Change Agents.

Check In – What is a Feature…

Feature:	
Phrase:	
Benefit Hypothesis:	
Acceptance Criteria:	
Risks & Dependencies	
Notes :	

…and why are they so important?

Check In – What is a Feature…

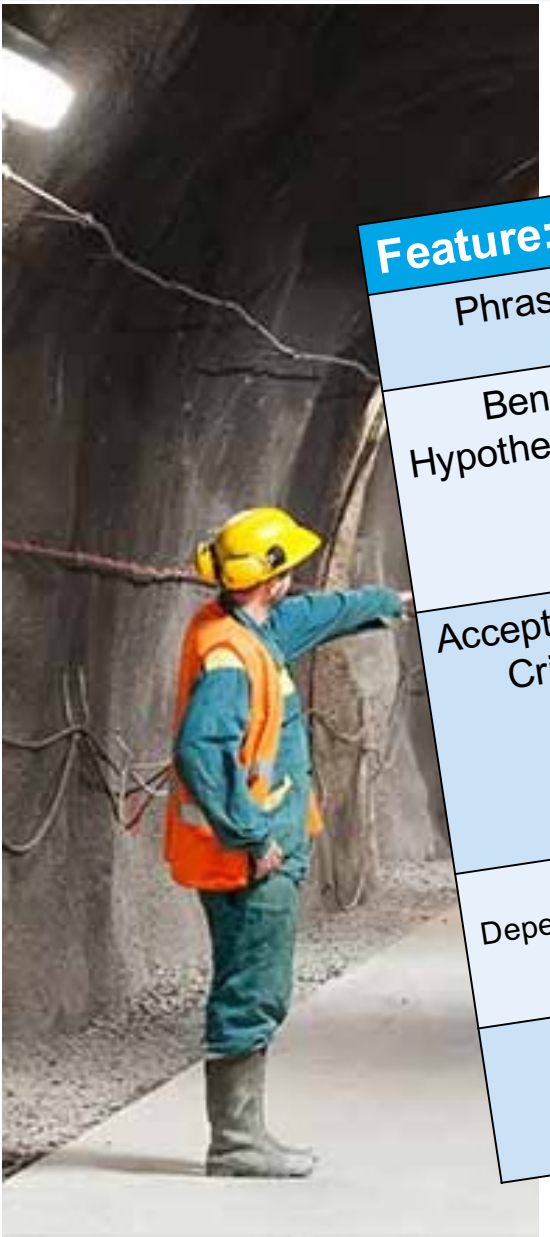


Feature: 2.1	
Phrase:	**Loyalty Discounts:** The system shall allow customers to open and maintain loyalty accounts, collect points and receive loyalty discounts.
Benefit Hypothesis:	More repeat business. Increased customer retention.
Acceptance Criteria:	Existing loyalty cards are reflected in the on-line store. New loyalty cards can be set up and issued from within the new system. Additional points are available for on-line shopping. Additional rewards are available for on-line shopping. On-line customers can benefit from the loyalty scheme without being issued a physical card.
Risks & Dependencies	Integration with the exiting physical card based loyalty points system.
Notes :	It is assumed that store accounts can be easily added to the Customer Preferences. It is assumed that, as for the stores, special offers are already 'points' aware.

Feature:

…and why are they so important?

Check In – What is a Feature…



Feature: 2.1

Phrase:	**Loyalty Discounts:** The sys… maintain loyalty accounts, c…
Benefit Hypothesis:	More repeat business. Increased customer reten…
Acceptance Criteria:	Existing loyalty cards… New loyalty cards c… Additional points a… Additional reward… On-line custome… a physical card…
Risks & Dependencies	Integration with…
Notes :	It is assumed that store accounts ca… It is assumed that, as for the stores, specia…

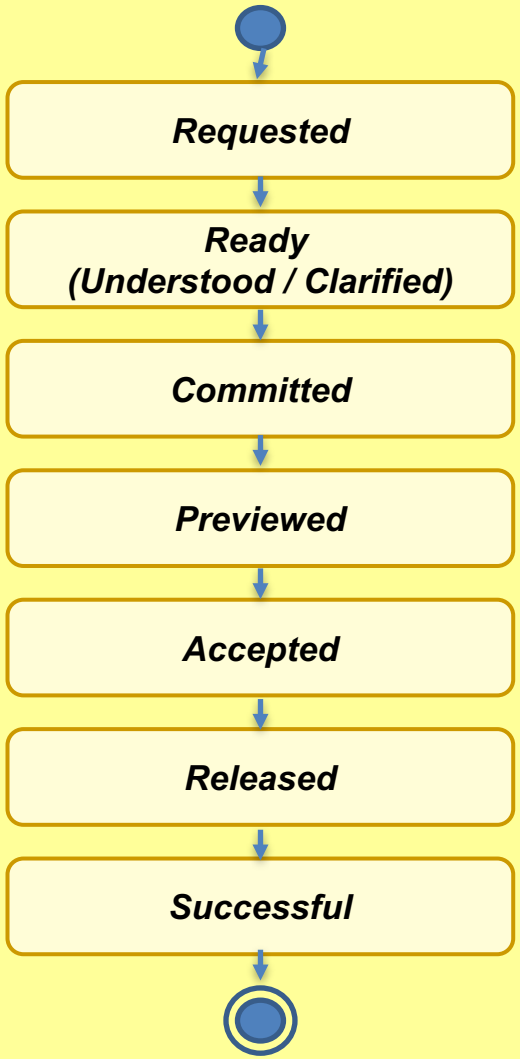
Feature:

Feature: E001

Phrase:	Spike – Investigate impediments to Release on Demand
Benefit Hypothesis:	We understand what needs to be done to move to a continuous delivery model: • Overall costs • Tooling options
Acceptance Criteria:	• At least the Top 5 impediments are clear • Costs to establish first round of improvements are clear • Potential for improvements is clear • A credible roadmap is available
Risks & Dependencies	Technical subject matter experts are not available
Notes :	Time-Box to no more than 4 weeks.

…and why are they so important?

Check In – The lifecycle of a Feature



A request has been received for the system to provide a new or improved service.

The Feature is clear, well understood and small enough for a team to be able to plan its completion.

The Feature has been pulled by a Team that has committed to its delivery.

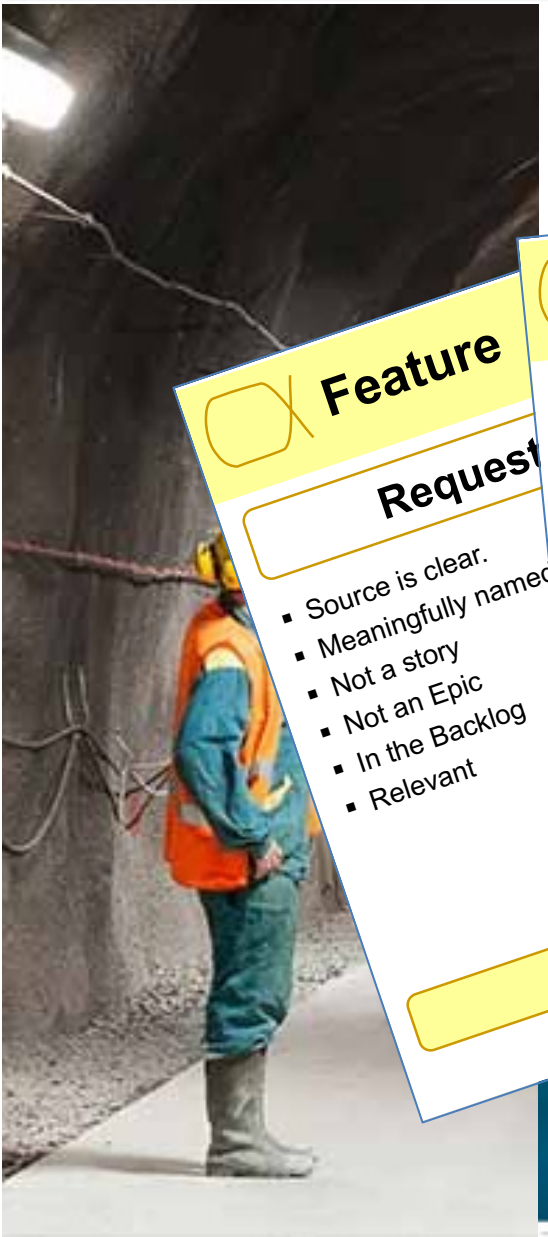
Enough of the Feature has been implemented, and demonstrated, to confirm the benefits hypothesis and verify the key implementation decisions.

The Feature has been shown to meet its acceptance criteria and has been accepted by Product Management as ready to be released.

The Feature is fully operational and has been made generally available to its users.

The Feature is being used for its intended purpose, the hypothesized benefits are being accrued and stakeholder feedback is positive.

Check In – How do your Features flow?



Feature

Request

- Source is clear.
- Meaningfully named
- Not a story
- Not an Epic
- In the Backlog
- Relevant

Feature

Ready

- The Feature has a clear own
- The Feature description is c
- Any 'fixed' requirements are
- Level of key stakeholder inv is understood.
- Small enough to fit within a
- The cost of delay is clear
- The Feature is testable.
- The Feature is feasible
- Potential benefits are und

2 / 7

IVAR JACOBSON INTERNATIONAL

Feature

Commit

- Stories Outlined
- Dependencies Und
- Lead team known
- Working Relations
- New Estimate Av
- Completion fored
- Acceptance crit
- Benefits hypothesis
- Technical Impact understood

3 / 7

IVAR JACOBSON INTERNATIONAL

Feature

Previewed

- Key decisions
- Unwanted stor
- Story priorities
- Product Mana
- The gap to co
- Release Fore
- Acceptance c
- Risk under co
- Benefits hypo
- Viability conf

Feature

Accepte

- No loose ends
- Accepted by the Fe
- Actuals known
- Acceptance Crite
- All necessary st
- Benefits hypot
- Releasable
- Fully tested
- Code comp

6 / 7

IVAR JACOBSON INTERNATIONAL

- Available to use
- Help available
- Actuals known
- Success forecast
- Benefits hypothesis ass
- All stories deployed

Feature

Successful

- In use
- Feedback positive
- Actuals known
- Benefits hypothesis accepted.

7 / 7

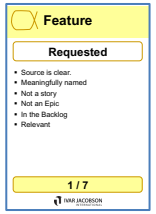
IVAR JACOBSON INTERNATIONAL

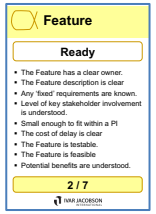
And how far do they go?

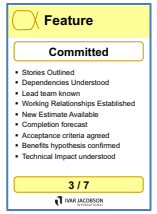
Let’s play with the cards

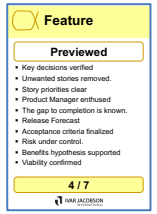
- Step 1 – Create your game board
  - Step 1.1 – Lay the cut deck of cards out left to right / states 1 to 7
  - Step 1.2 - Pick a real case on your table
  - Step 1.3 – Add a row for each group involved including at least Product Management, Dev Team and Ops
    - Add any additional rows as needed
- Step 2 – Lay out the cards to identify who is in the lead for each state
  - Place the cards in each row maintaining the order
  - If the state is not covered or used by your train leave it where it is
- Step 3 – Identify any additional states or handovers using the post-its
- Step 4 – Record your results
  - Identify (and count) any hand-overs and or additional states
- Step 5 – Repeat for other scenarios on your table until your 15 minutes is up

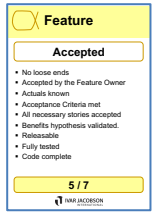
Play Your Cards Right: Layout your game board

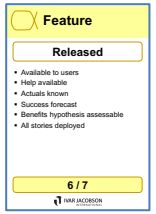


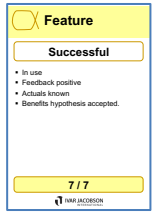












PM

Dev

System
Test

Ops

Play Your Cards Right: Align the states

PM

Review with CCB

Dev

Coded

Tested

System Test

System Tested

Regress-ion Tested

Release Tested

Ops

Let’s play with the cards

- Step 1 – Create your game board
  - Step 1.1 – Lay the cut deck of cards out left to right / states 1 to 7
  - Step 1.2 - Pick a real case on your table
  - Step 1.3 – Add a row for each group involved including at least Product Management, Dev Team and Ops
    - Add any additional rows as needed
- Step 2 – Lay out the cards to identify who is in the lead for each state
  - Place the cards in each row maintaining the order
  - If the state is not covered or used by your train leave it where it is
- Step 3 – Identify any additional states or handovers using the post-its
- Step 4 – Record your results
  - Identify (and count) any hand-overs and or additional states
- Step 5 – Repeat for other scenarios on your table until your 15 minutes is up

Check In – Software Factory or Software Laboratory?

Features as ideas.
Full collaboration.
Focus on value.

It goes in - it may come out

Features as orders.
Clear boundaries and handovers.
Focus on productivity.
It goes in - it comes out

On your marks – Identifying the best Features

On your marks – An open door and an early exit

g teams.

On your marks – Identifying Features

Most customer requests will be Features or, if too large, sliceable Feature Sets.

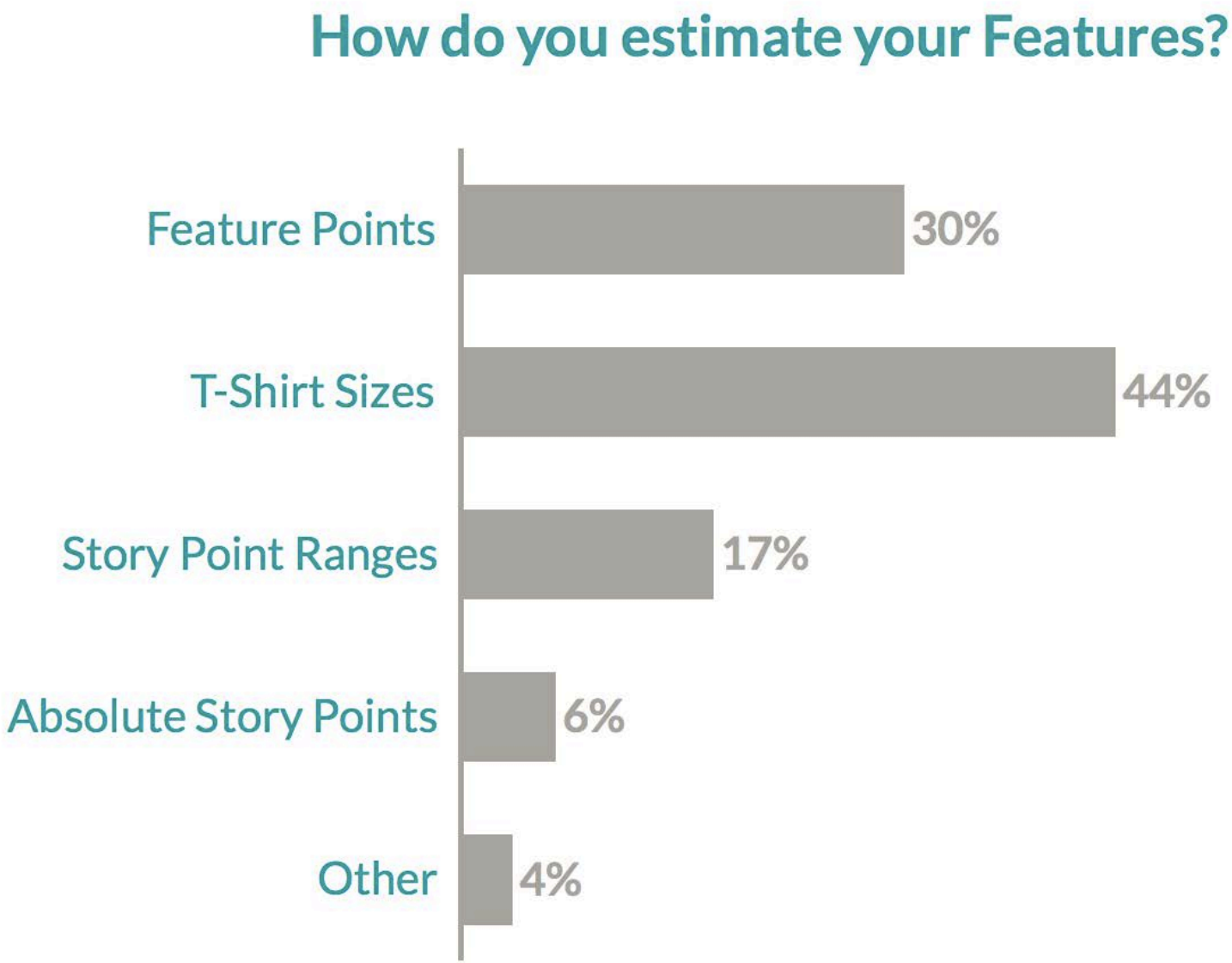
But some will be irrelevant
Or become irrelevant.

On your marks – How do you estimate your Features?

How do you estimate your Features?

1. Feature Points
2. T-Shirt Sizes
3. Story Point Ranges
4. Absolute Story Points
5. Other

On your marks – How do you estimate your Features?



On your marks – How do you estimate your Features?

Forecasting Estimate

- When: as soon as the Feature is queued
- use story point ranges

Planning Estimate

- When team pulls a Feature
- use sum of Story Story Points

On your marks – How do you slice Features?

① PREPARE THE INPUT FEATURE

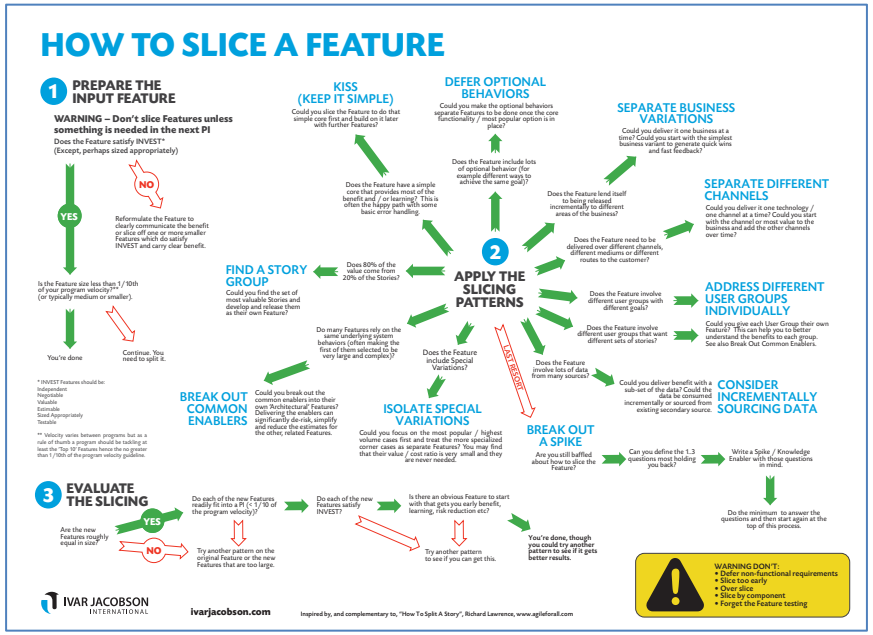
- WARNING – Don't slice Features unless something is needed in the next PI ☺

② APPLY THE SLICING PATTERNS

- KISS (Keep It Simple)
- Defer Optional Behavior
- Separate Business Variations
- Separate Different Channels
- Address Different User Groups Individually
- Consider Incrementally Sourcing Data
- Isolate Special Variations
- Break Out Common Enablers
- Find a Story Group
- Break Out a Spike

③ EVALUATE THE SLICING

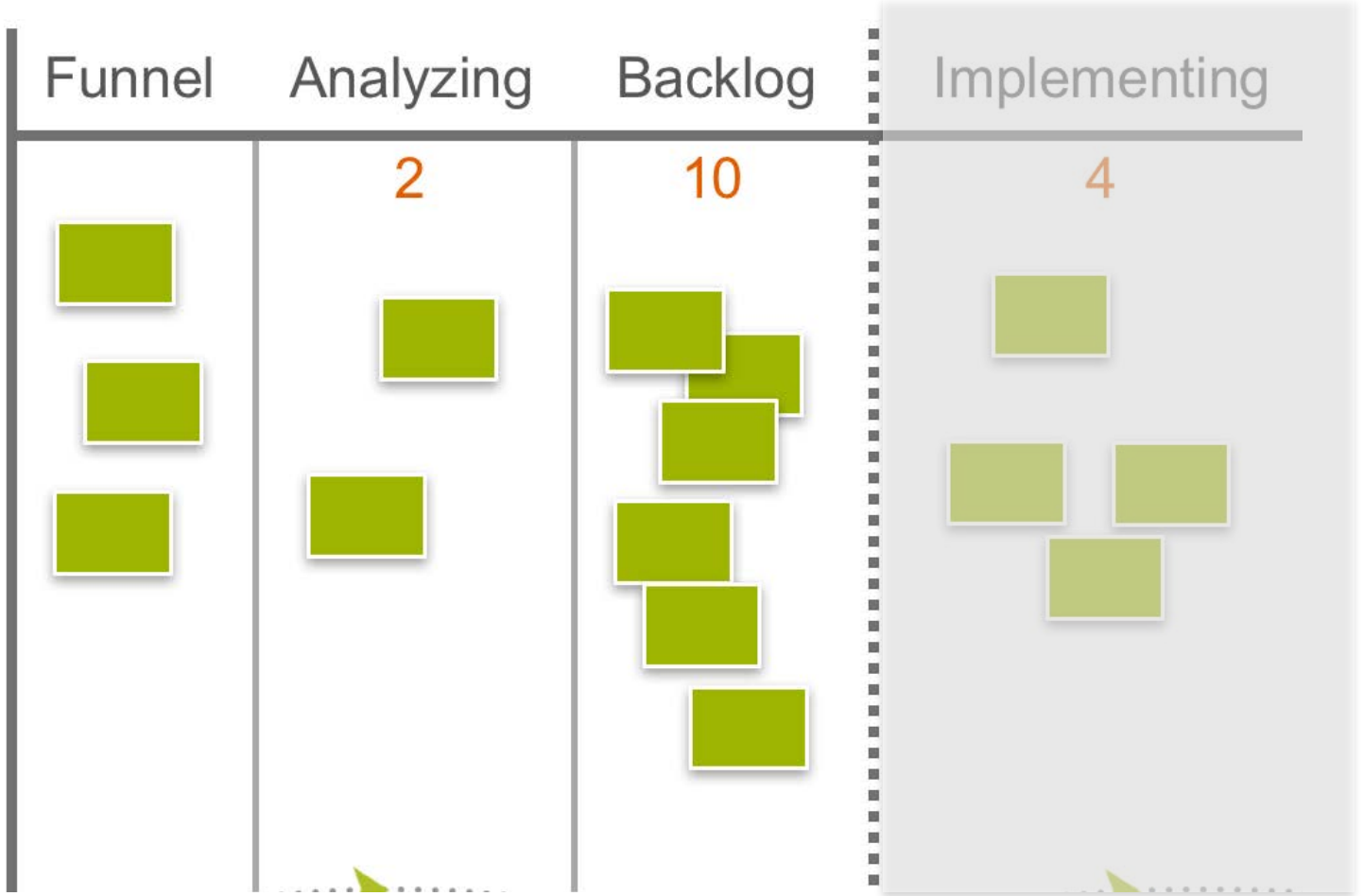
WARNING DON'T:
- **Defer non-functional requirements**
- **Slice too early**
- **Over slice**
- **Slice by component**
- **Forget the Feature testing**



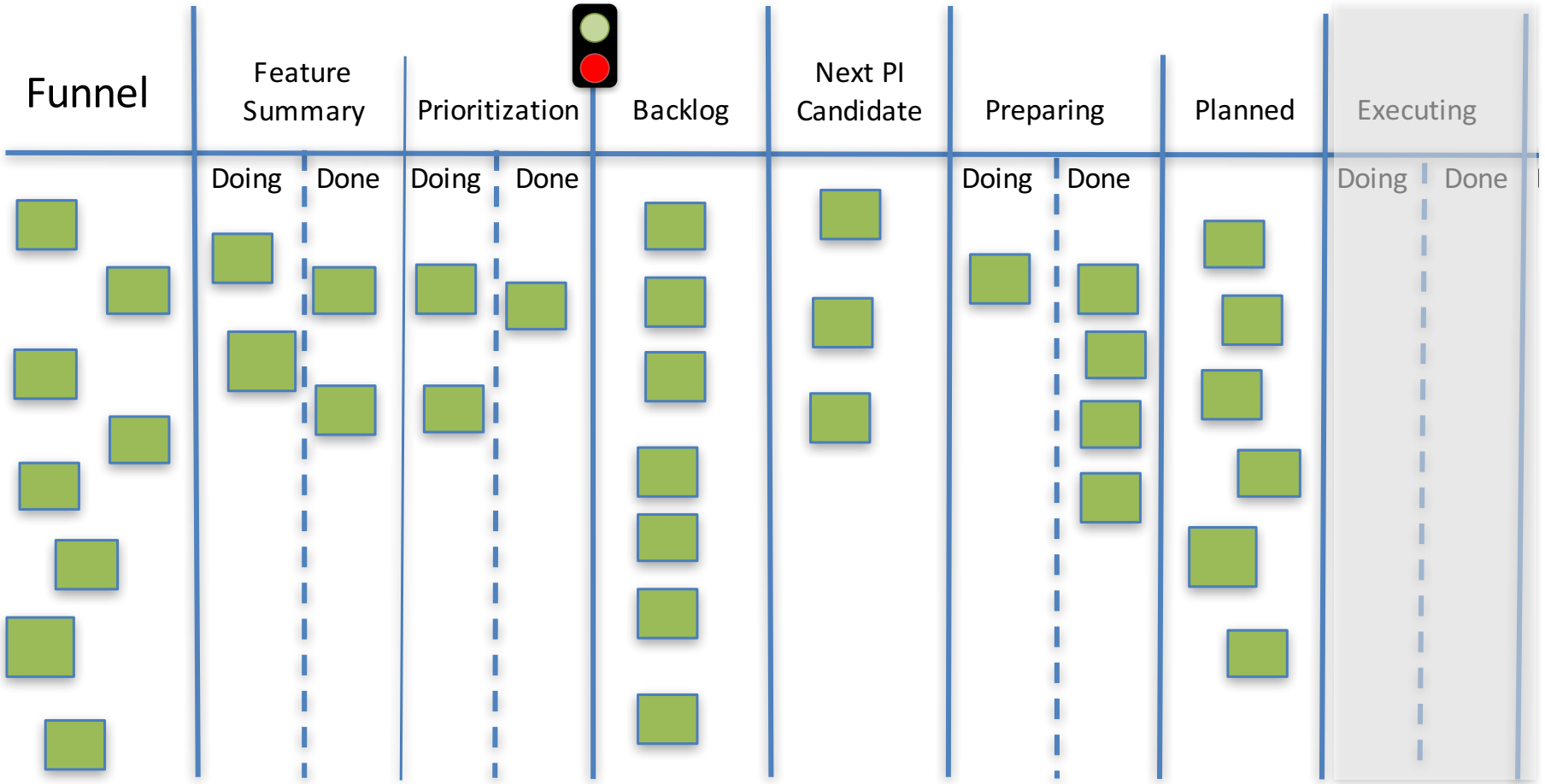
All summarized on this handy poster.
(Available from www.ivarjaobson.com)

Creating winning teams.

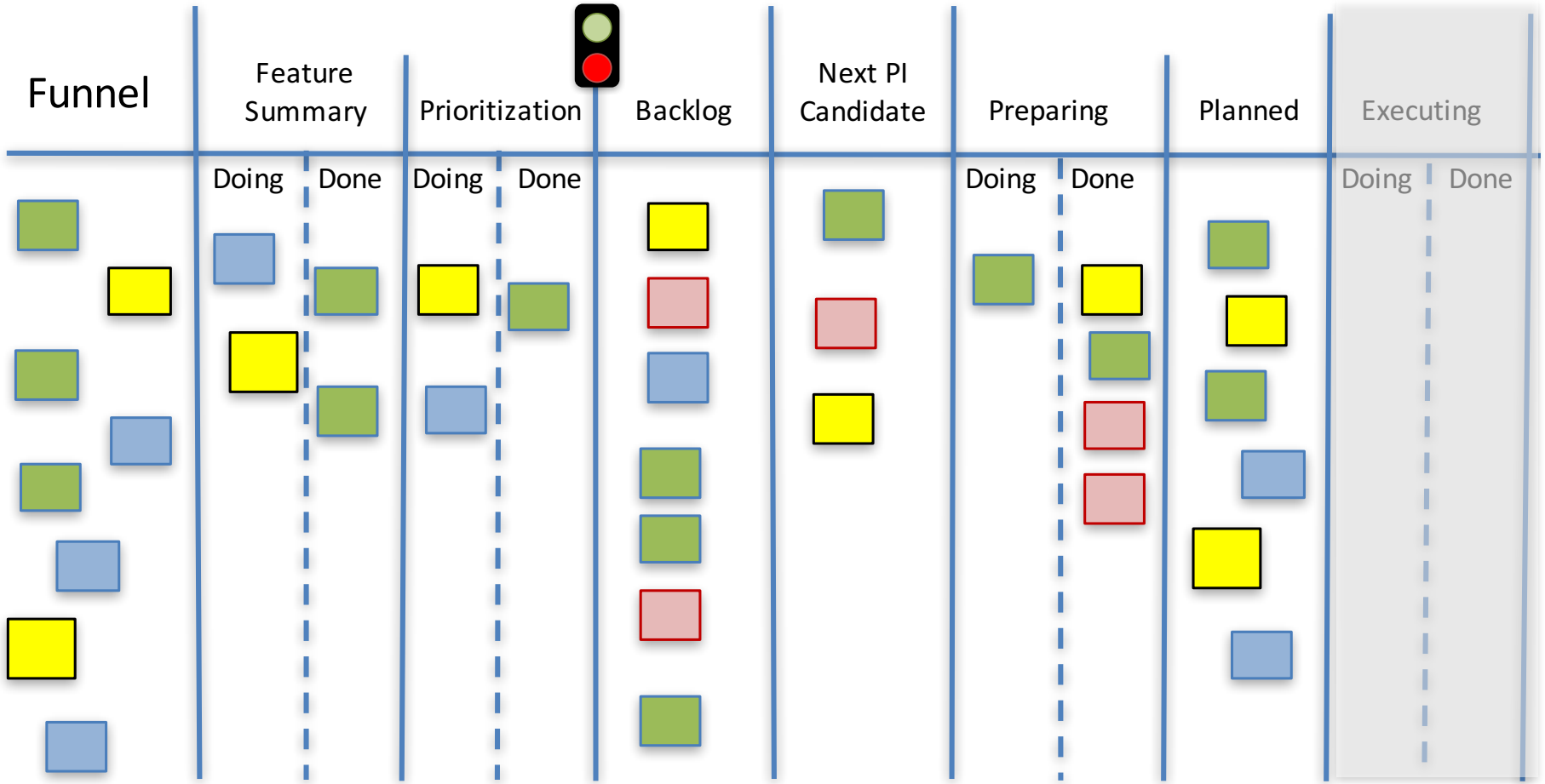
On your marks – There's more than one way to skin a cat



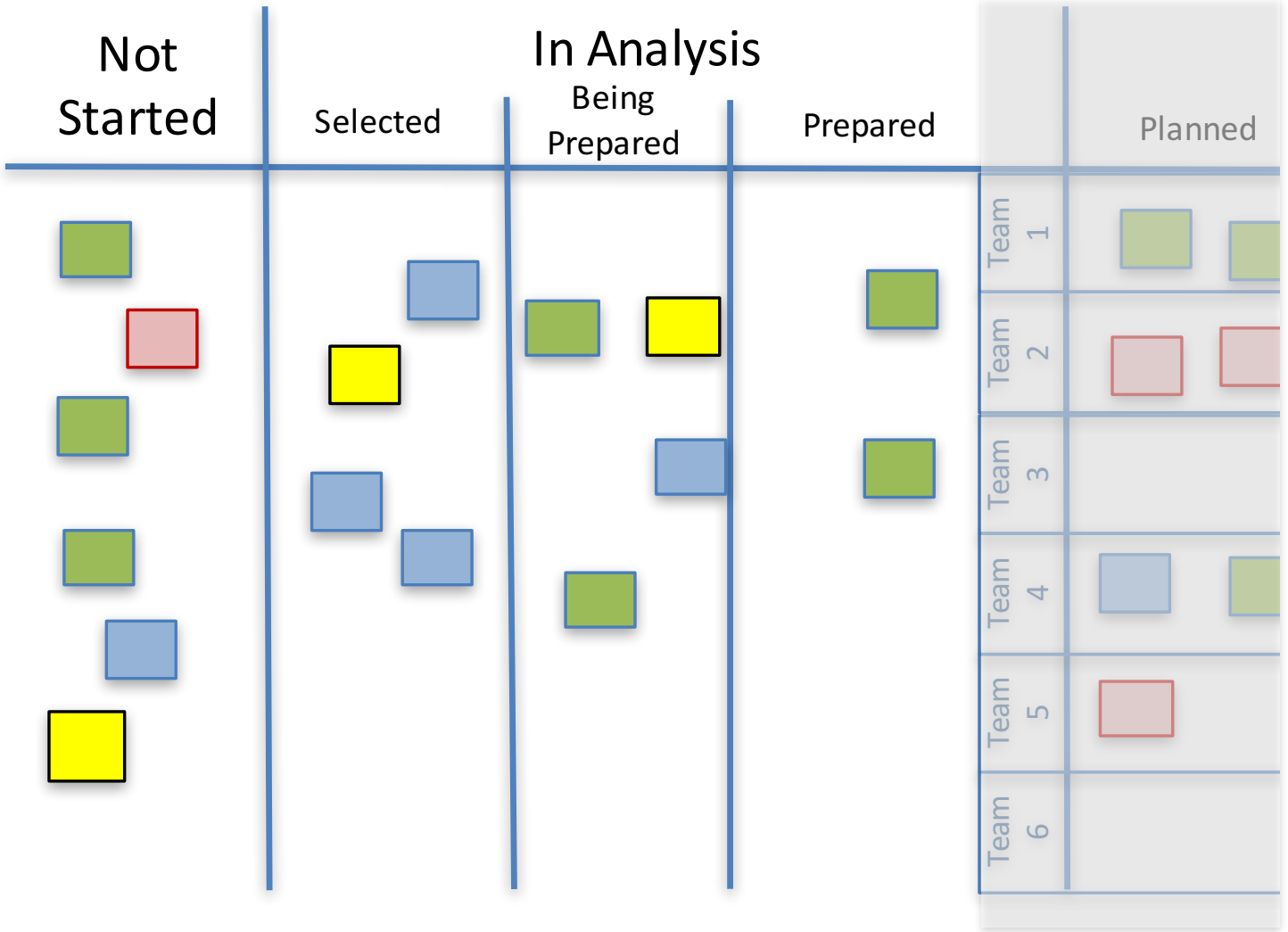
On your marks – There's more than one way to skin a cat



On your marks – There's more than one way to skin a cat



On your marks – There's more than one way to skin a cat



On your marks – So what can go wrong?

In pairs take a few minutes to discuss things that can wrong during Feature preparation.

Which do you think is the worst:

1. Too much preparation?
2. Too little preparation?

On your marks – The Seven Sins of Feature Preparation

Seven Deadly Sins of Feature Preparation

Pride - Pre-defining all the Stories

Sloth - Freezing the Scope

Lust - Gold Plating

Greed - Maximum Possible Features

Wrath - Disenfranchised POs & Teams

Gluttony - Believing the Initial Estimate

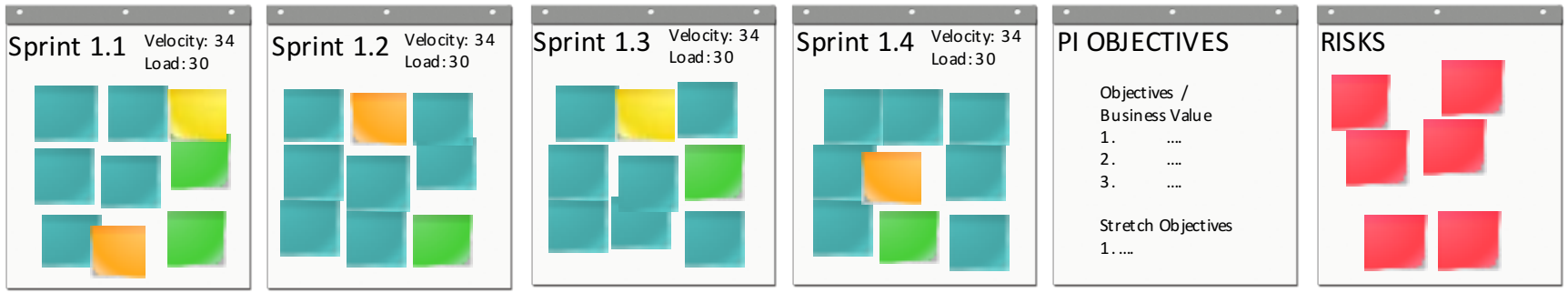
Envy - My Feature, My Team

\+ The Original Sin:
The Pre-allocation of Features

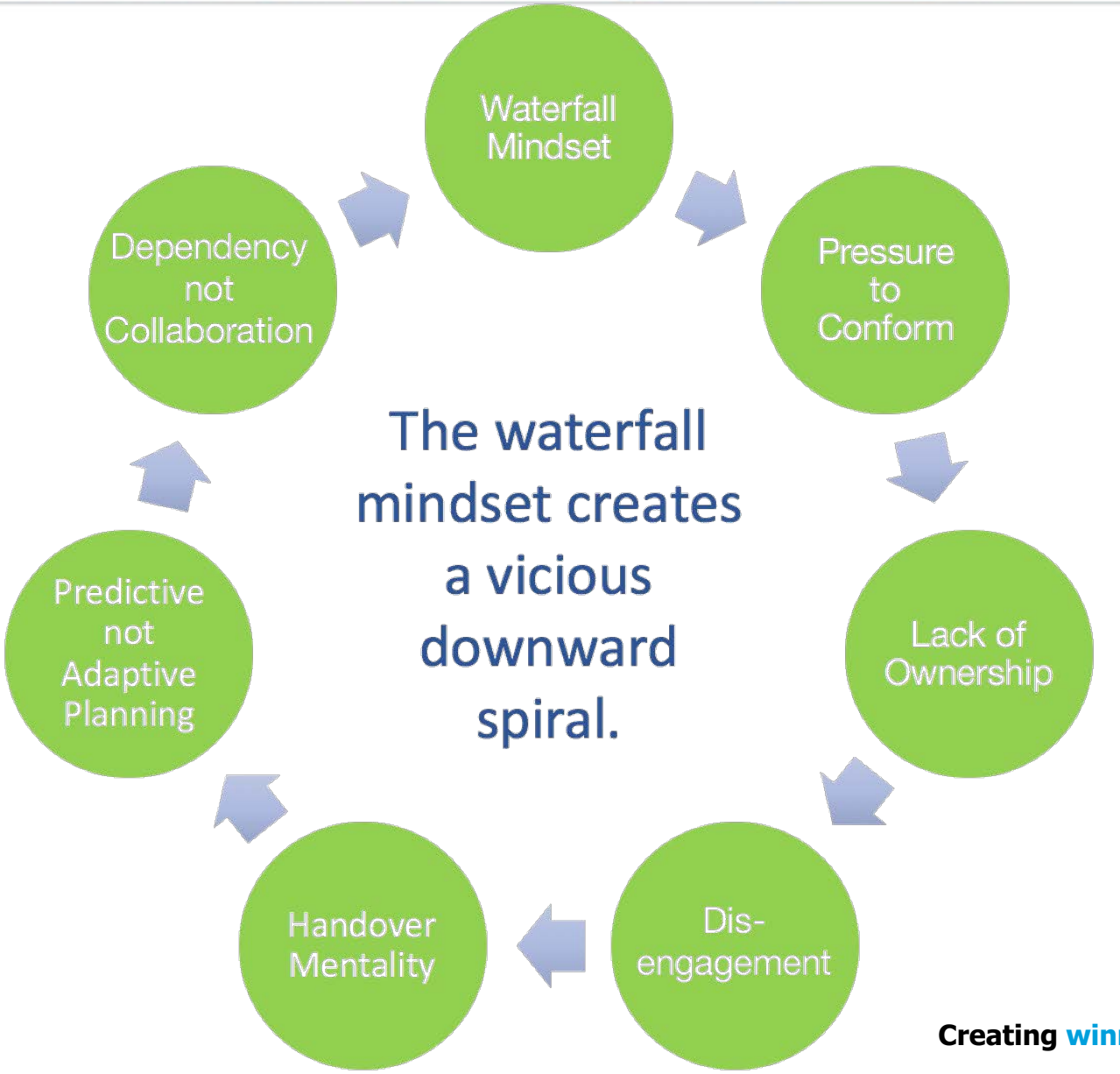
Get Set – The Power of PI Planning

Get Set – The Power of PI Planning

- Pull don't push
- Negotiate
- Don't waterfall…..
- …Don't try to shrink the event by preparing your features…

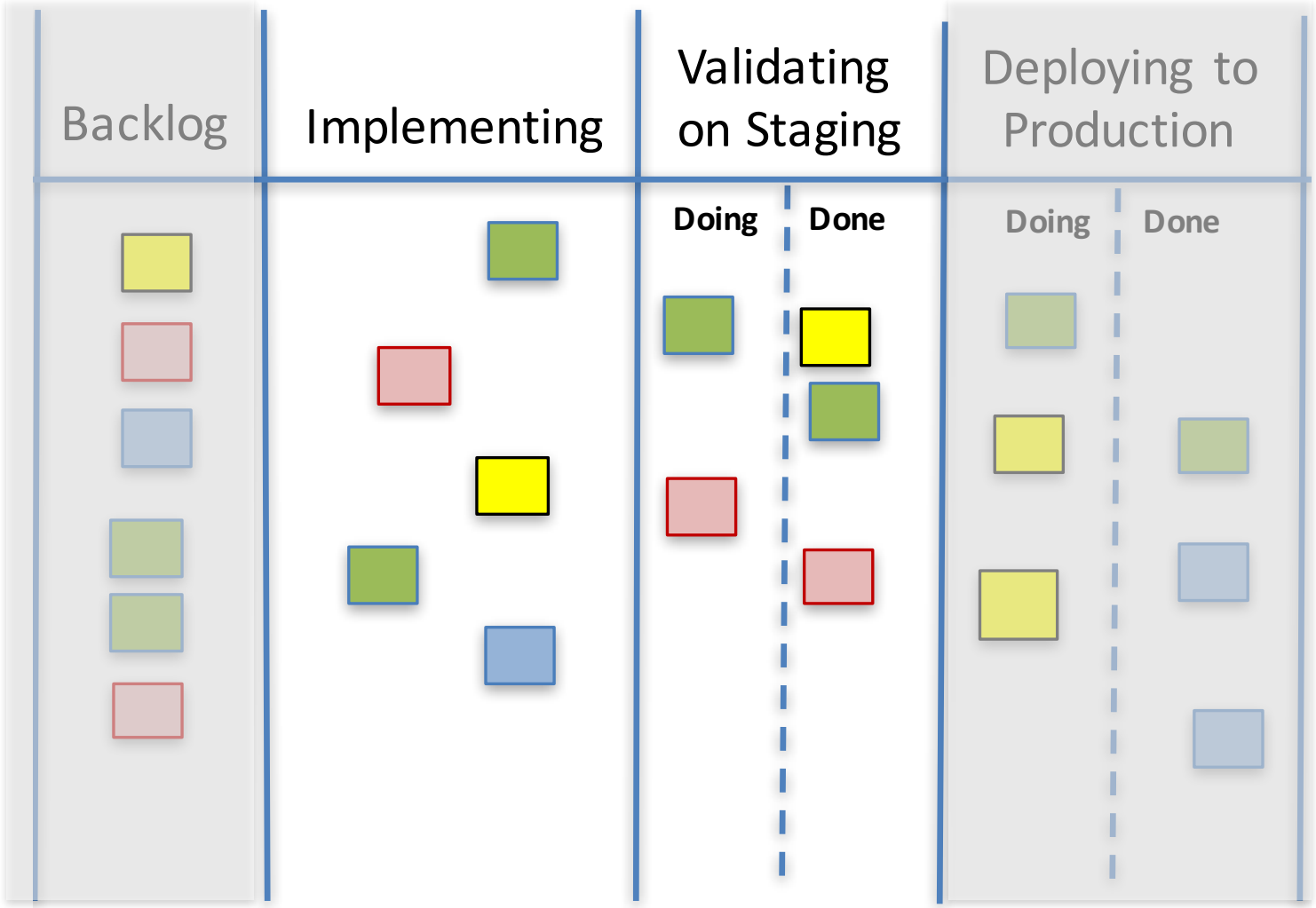


Get Set – Just say no to waterfalling your Features



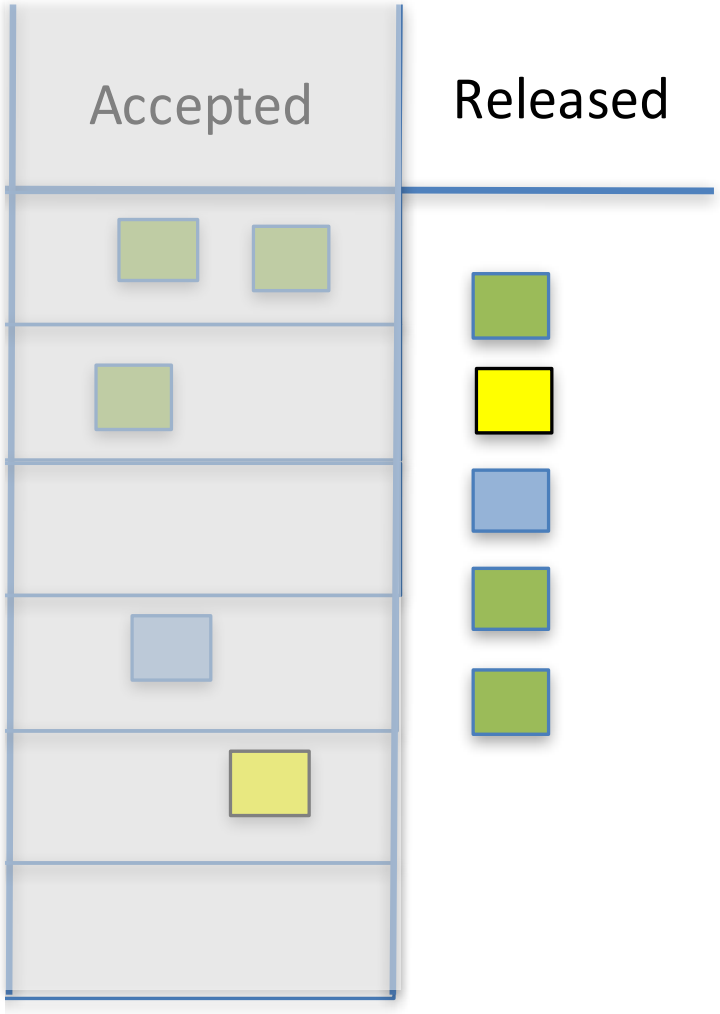
Go: Shipping the right Features

Go: Don’t develop in secrecy

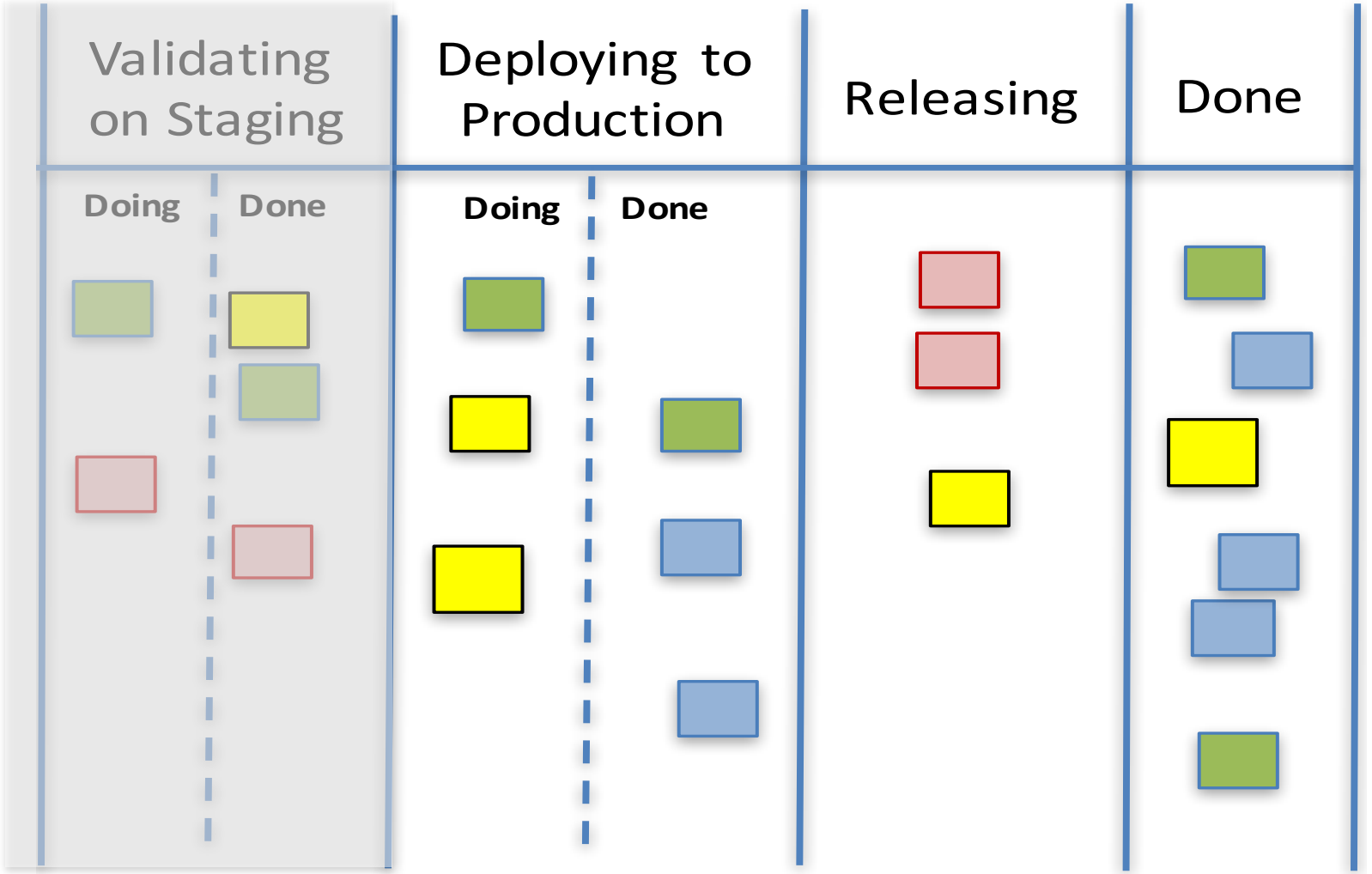


Go: Don’t develop in secrecy

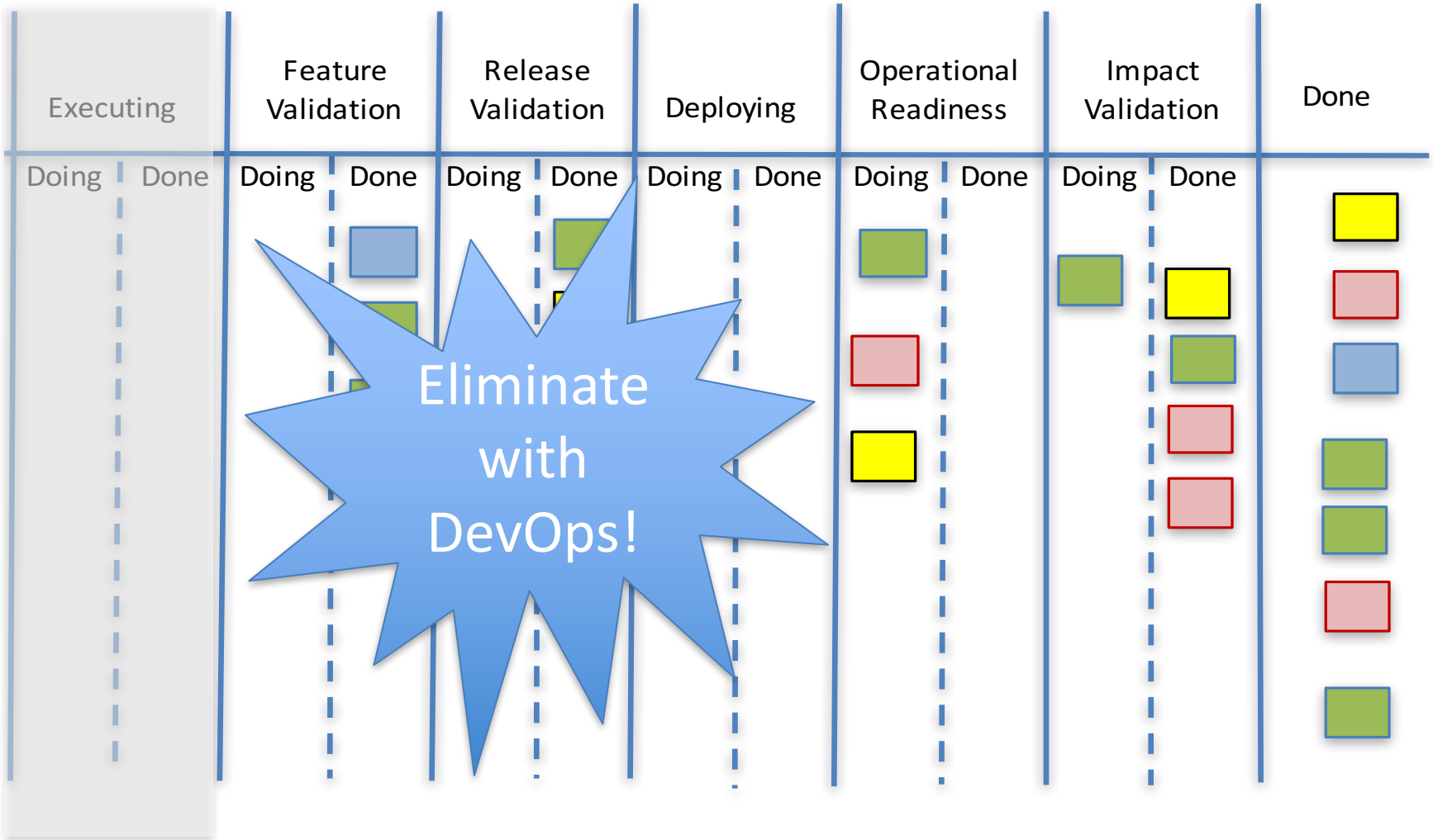
Go: Don’t ship in silence



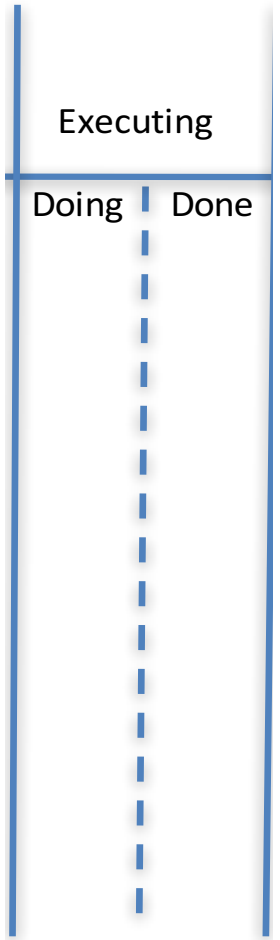
Go: Don’t ship in silence

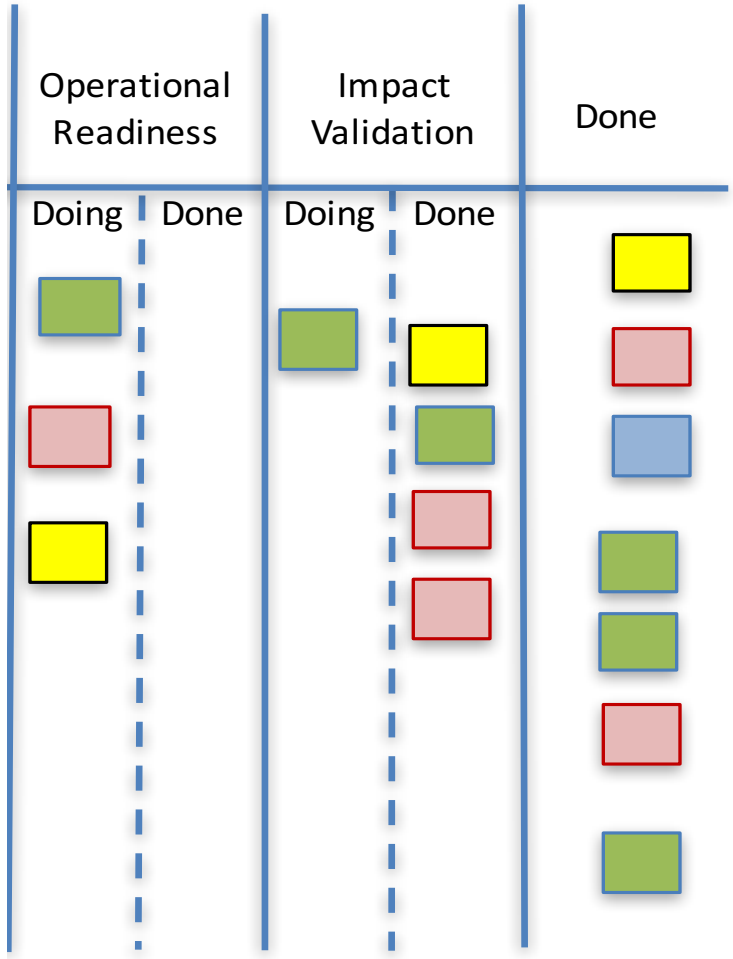


Go: Don’t ship in silence



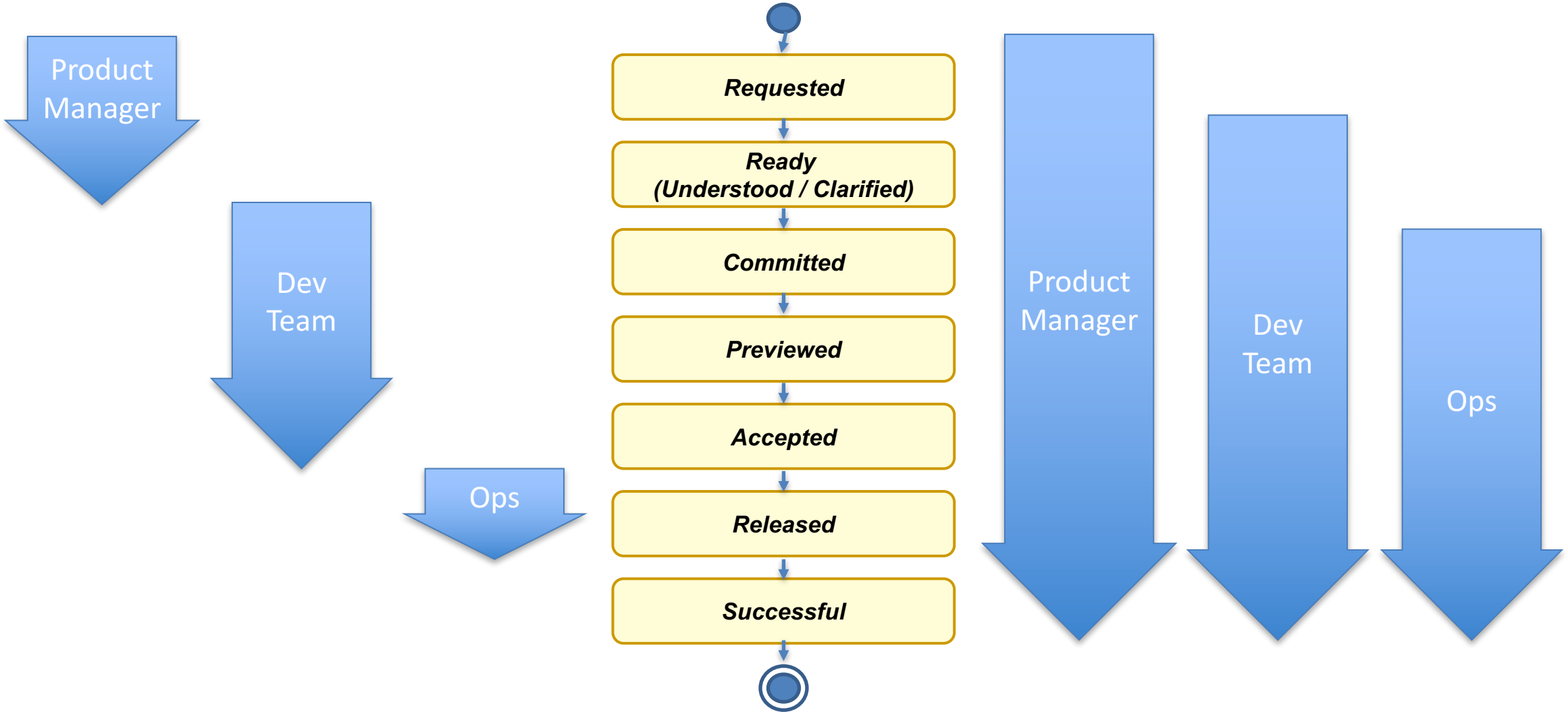
Go: Don’t ship in silence





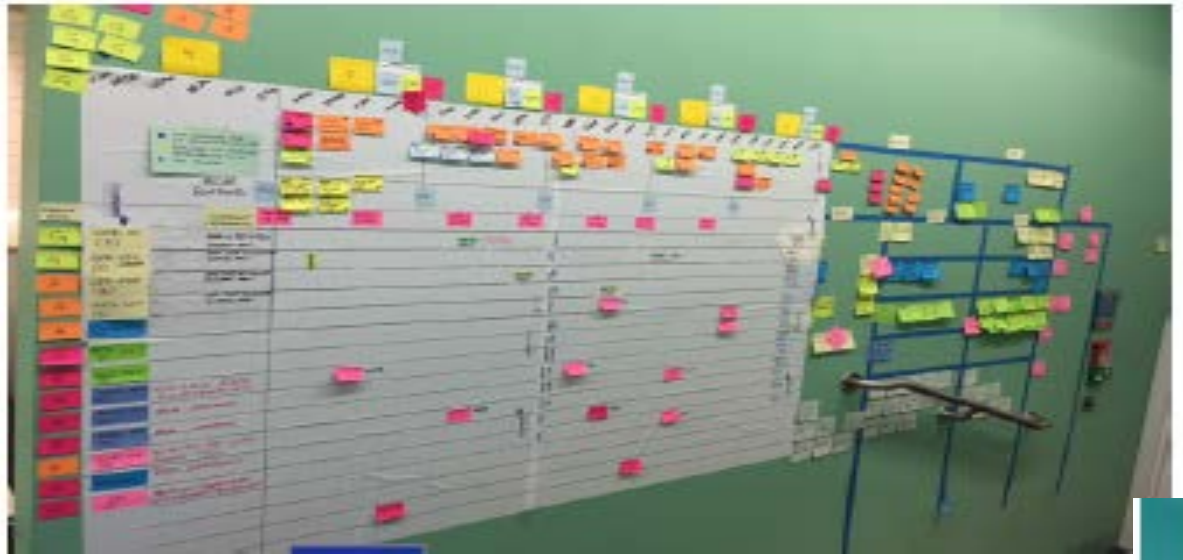
Winning the Race

Winning the race: What does it really mean to be done?



Winning the race: Value realisation

Winning the race: Balancing Flow with Forecasts



EPIC Board (Stakeholder View)

- Long term view
- PI Velocity vs Forecast Features
- Forecast vs Time Critical Events
- Competing Objectives (Causal Model)
- Influencing decisions

Program Board

- Epic & Feature end to end Flow
- “Real-time” view
- Transparency of Epic & Feature state
- Risk & Issues

Check Out: Stand by your Principles

Check Out: Stand by your Principles

#1 Take an Economic View

…to sustain and grow your business.

All outstanding work (whether it has been started or not) should be sequenced to minimize cost of delay. Sunk costs should be ignored to ensure the maximum business benefit is generated from the money about to be spent.

Stop Starting, Start Finishing

☺ *The backlog is prioritized with the stakeholders using cost of delay and relative estimates of size. Incomplete Features are re-estimated and re-prioritized every PI.*

☹ *The HiPPOs rule – it's my backlog and I set the priorities only adjusting them when someone more important overrules me. Once work is started then it is never stopped.*

Check Out: Stand by your Principles

Step 1 – Discuss and rank the principles

- They can have the same rank if you like

Step 2 – Complete the happiness radiator

- Tick each row to indicate how well you believe your Product Management Team embodies the principle

Principle	Rank	☺	😐	☹
# 1	# 1	✓		✓
# 2	# 3			✓ ✓
# 3	# 1	✓	✓	✓
# 4	# 2		✓ ✓	✓

Check Out: Stand by your Principles

Principle		Happy	Ambivalent	Sad
# 1	Take an Economic View	13%	41%	46%
# 2	Apply Systems Thinking	34%	32%	34%
# 3	Assume variability; preserve options	18%	40%	42%
# 4	Build incrementally, with fast integrated learning cycles	17%	34%	48%
# 5	Base milestones on objective evaluation of working systems	18%	43%	39%
# 6	Visualize and limit WIP, reduce batch sizes, and manage queue lengths	20%	34%	46%
# 7	Apply cadence, synchronize with cross-domain planning	43%	26%	31%
# 8	Unlock the intrinsic motivation of knowledge workers	15%	58%	27%
# 9	Decentralize decision-making	24%	21%	56%

Check Out: Get hold of the props and so much more

www.ivarjacobson.com

- for electronic / printable versions of the all cards
- for blogs
- for posters
- for pocket guides