agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# Doing the Right Work & Doing the Work Right

## Strategy Pattern

## Program Pattern

## Operational Pattern

DISCOVERY SECTION

DELIVERY SECTION

Mission & Vision

Objectives & Goals

Strategic Initiatives



Doing the Right Work

Doing the Work Right

# **The Program Pattern** used to launch and execute programs

## Strategy

### IDEA

Strategic Initiative
New Requirement
Enhancement

Problem
Opportunity

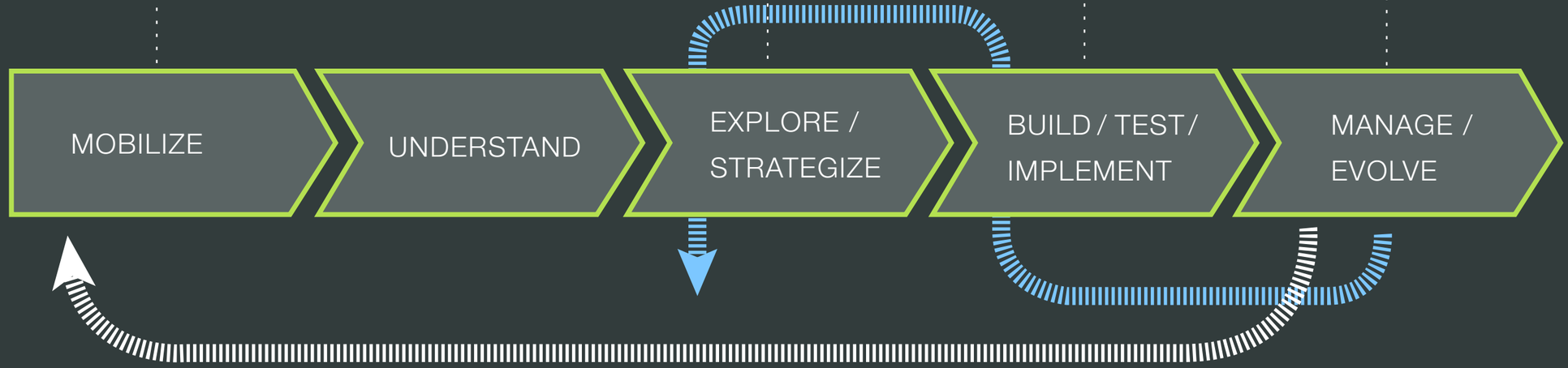
### DISCOVER

Understand & Strategize

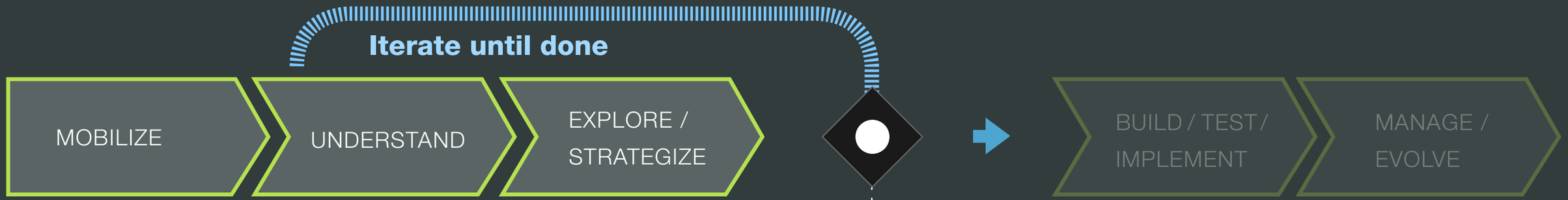
Governance /
Funding Gates

### DELIVER

Iteratively build,
test & deliver



# Discovery Practice




Discovery brief

Right stakeholders

Gate approval to start

Problem analysis

Stakeholder analysis

Desired outcome

Benefits

Blockers

Scope

Epics / Features / MVP

Risks & Dependencies

Solution options

Preferred solution

Estimation

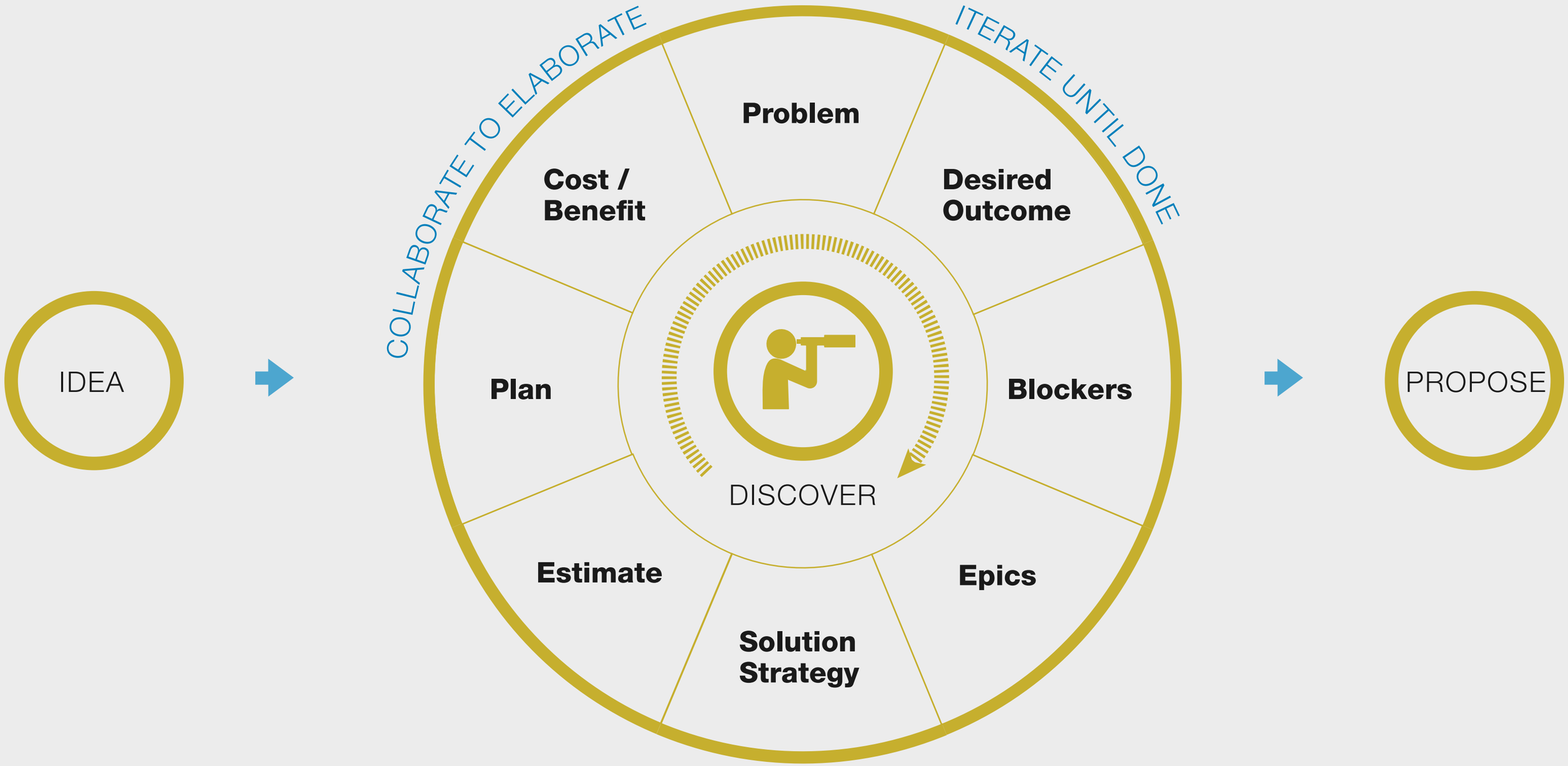
Planning

Cost / Benefit analysis

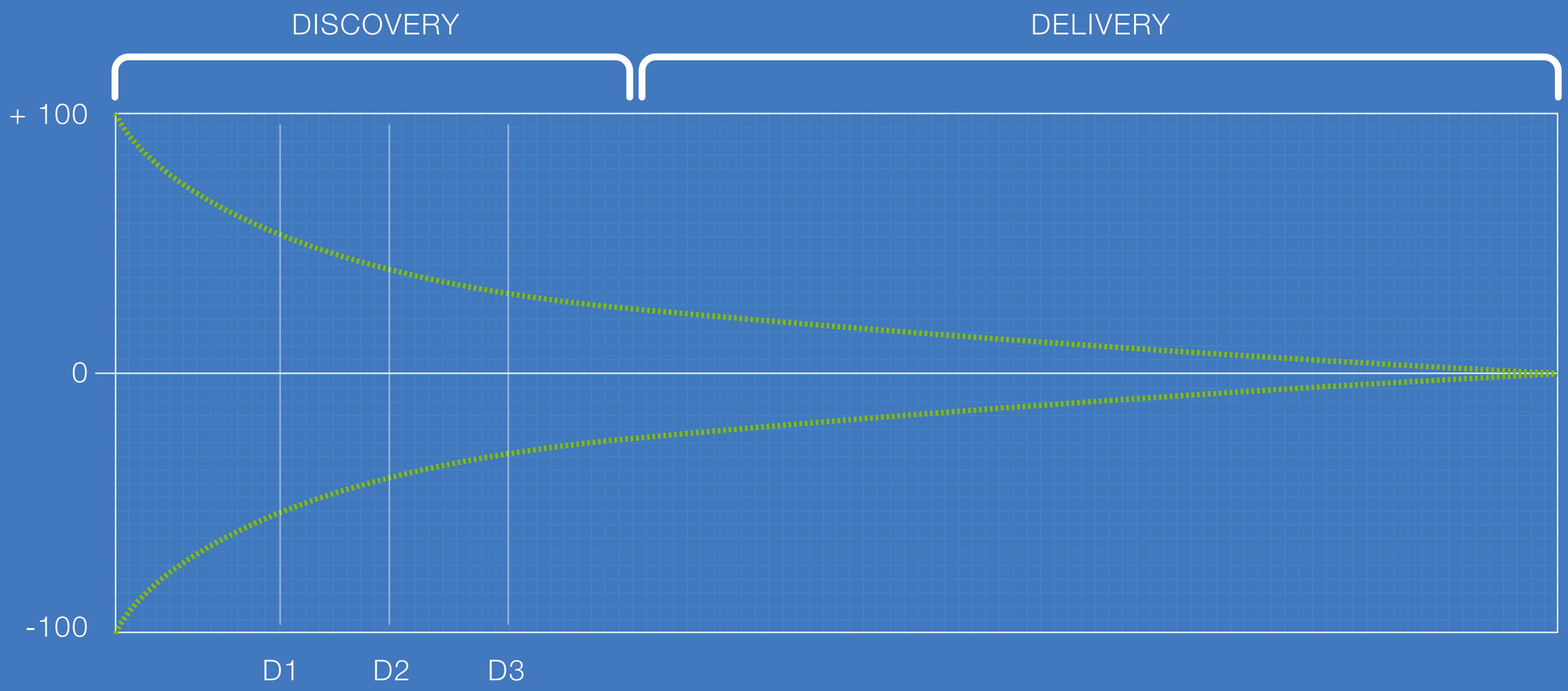
**Program Charter (Proposal) & Gate Approval**

**Collaborate to Elaborate**

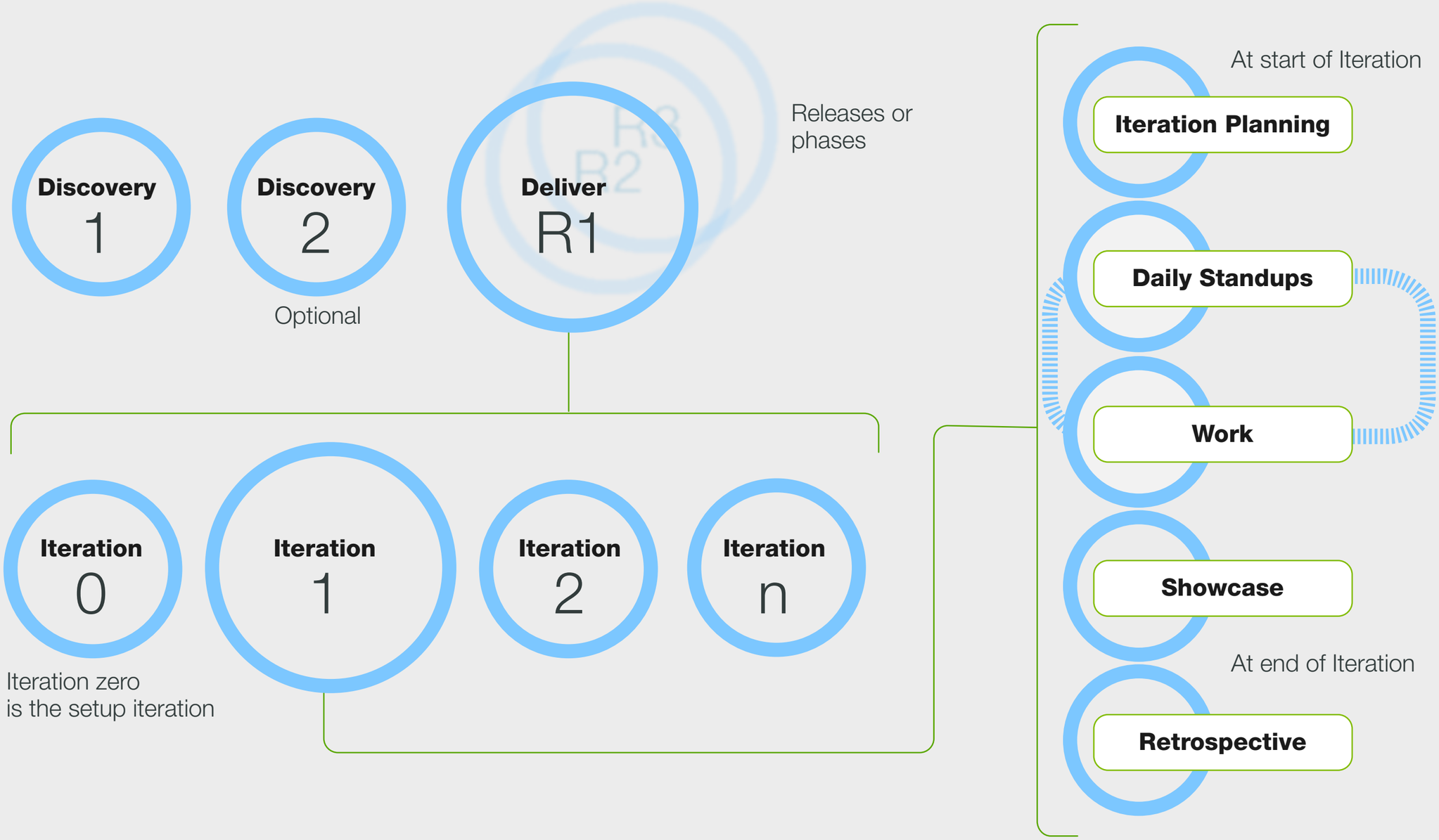
# Discovery Practice



# Cone of Uncertainty



# Lifecycle of Delivery



# Scrum

**Inputs** from customers, team, managers, execs

**Product Owner**

**Product Backlog**

A Prioritized List

**of what is required: features, bugs to fix**

**Team**

Sprint Planning Meeting

**The team commits to as much high priority backlog as can be completed by the end of the sprint**

**Sprint Backlog**

Task Breakout

**Scrum Master**

1-4 Week Sprint

**Sprint end date and deliverable do not change**

**Daily Standup Meeting**

15-30 Minutes

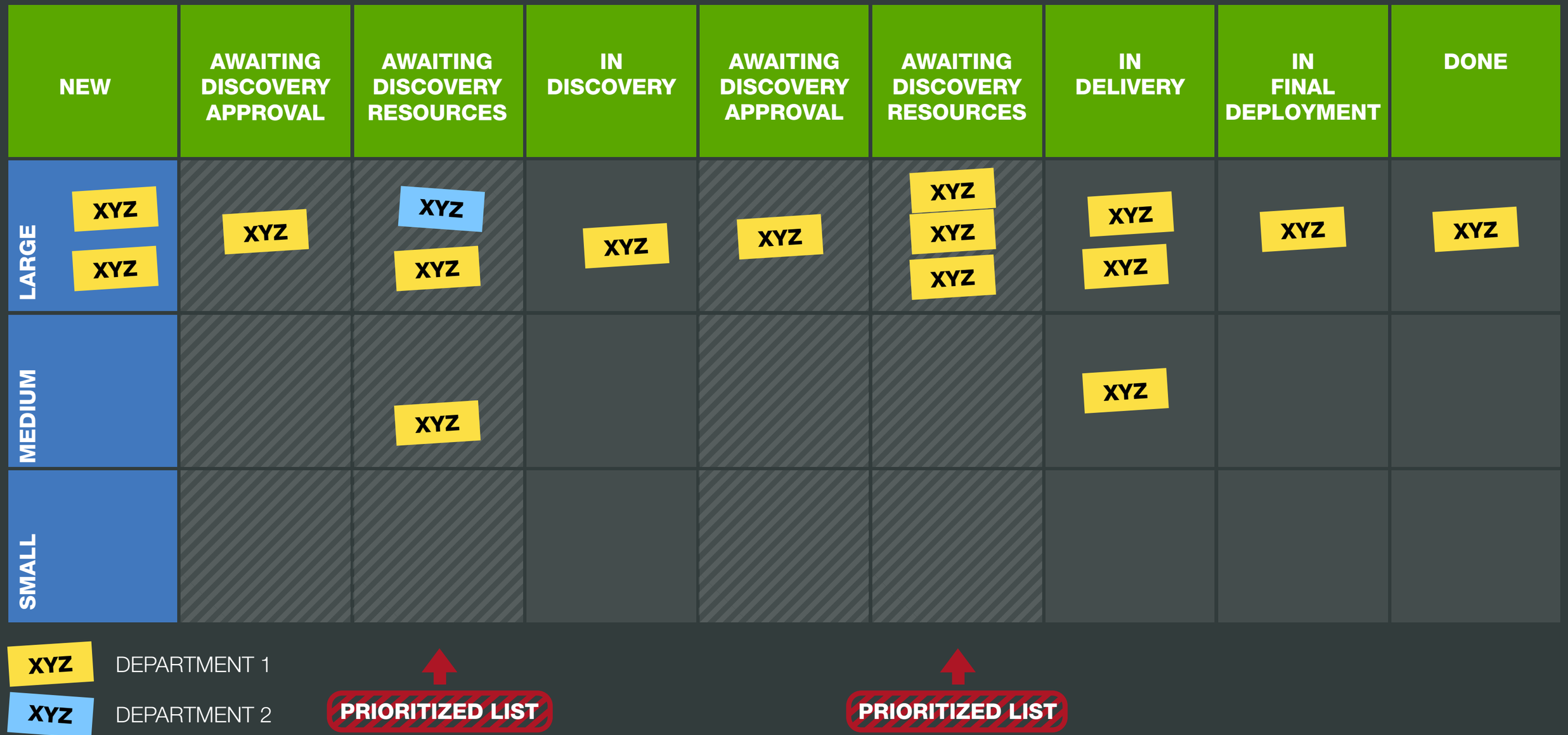
**Sprint Review**

**Finished Product**

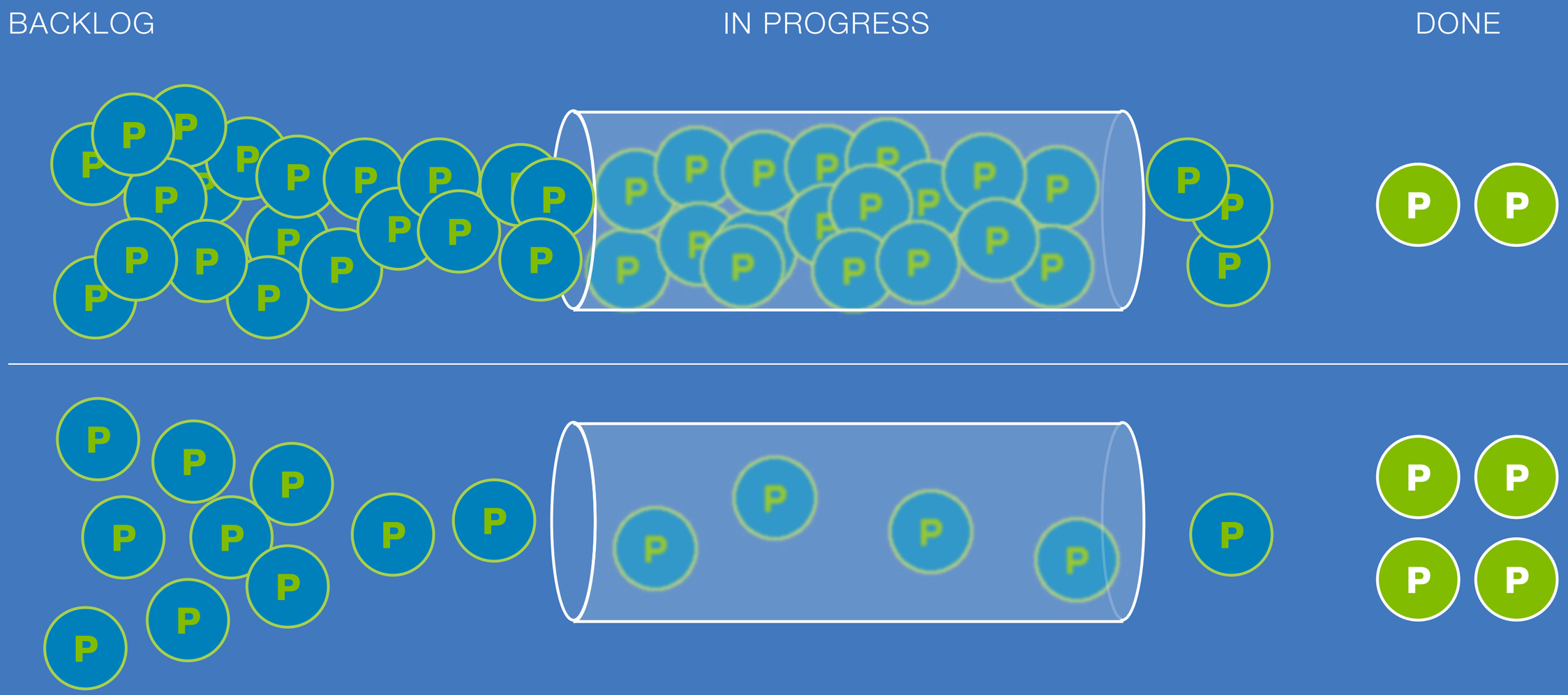
Product Increment

**Sprint Retrospective**

# The Portfolio Wall



# Slow Down in Order to do More




**Minimize WIP** by **Managing the Funnel**
and not Overburdening

# Agile Project Characteristics

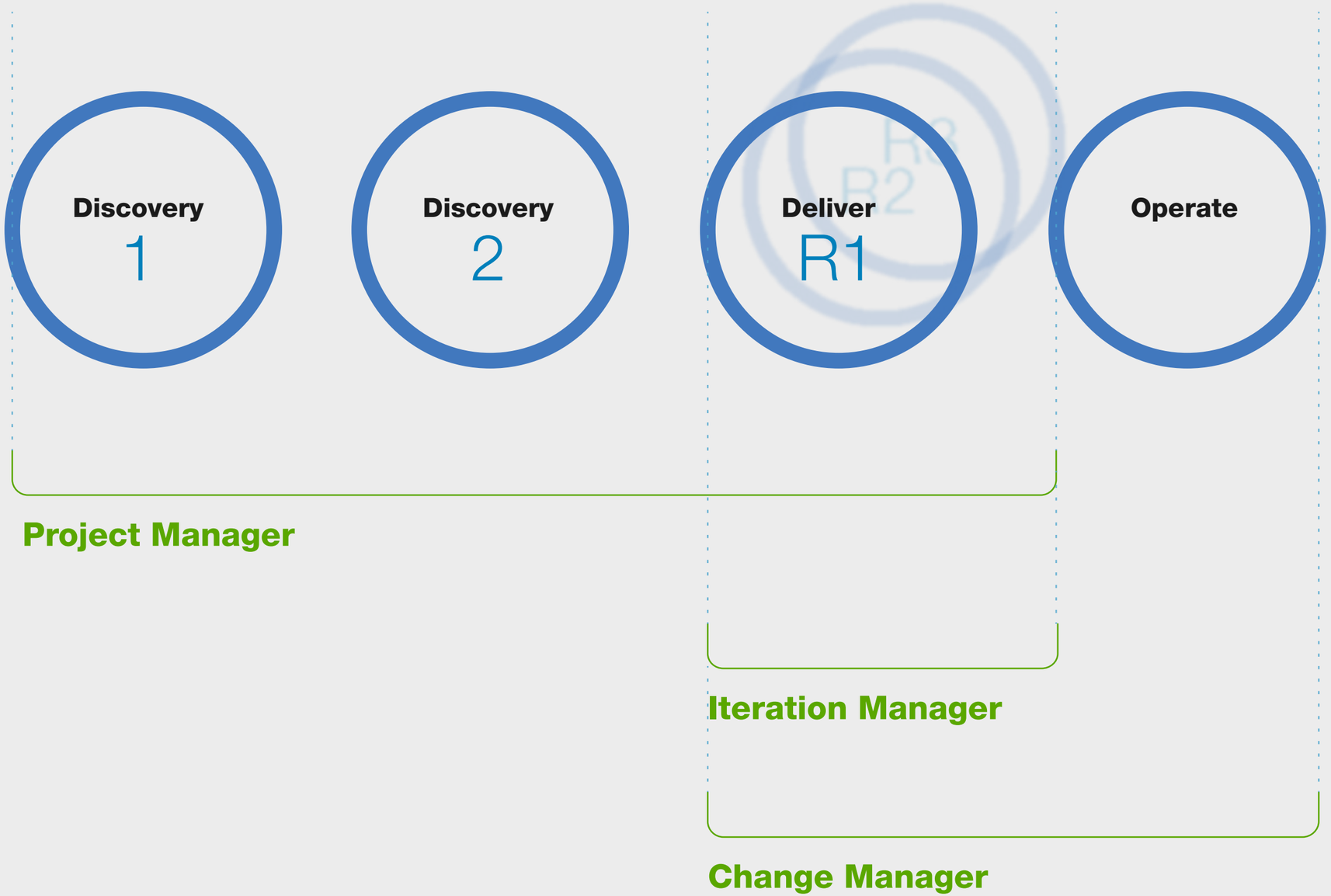
agileACADEMY

# Agile Program Fundamentals

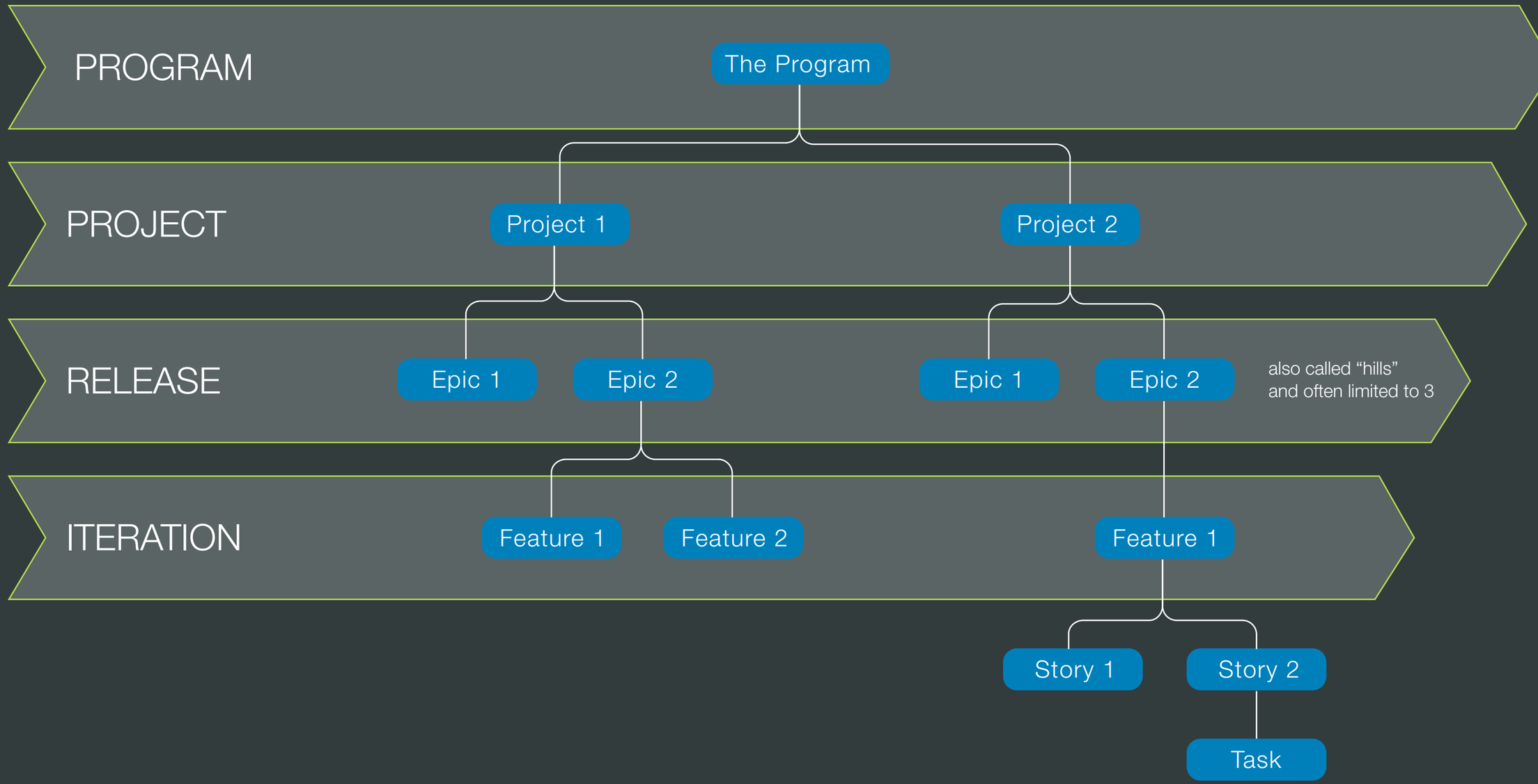
Course IAA2

## Course Topics

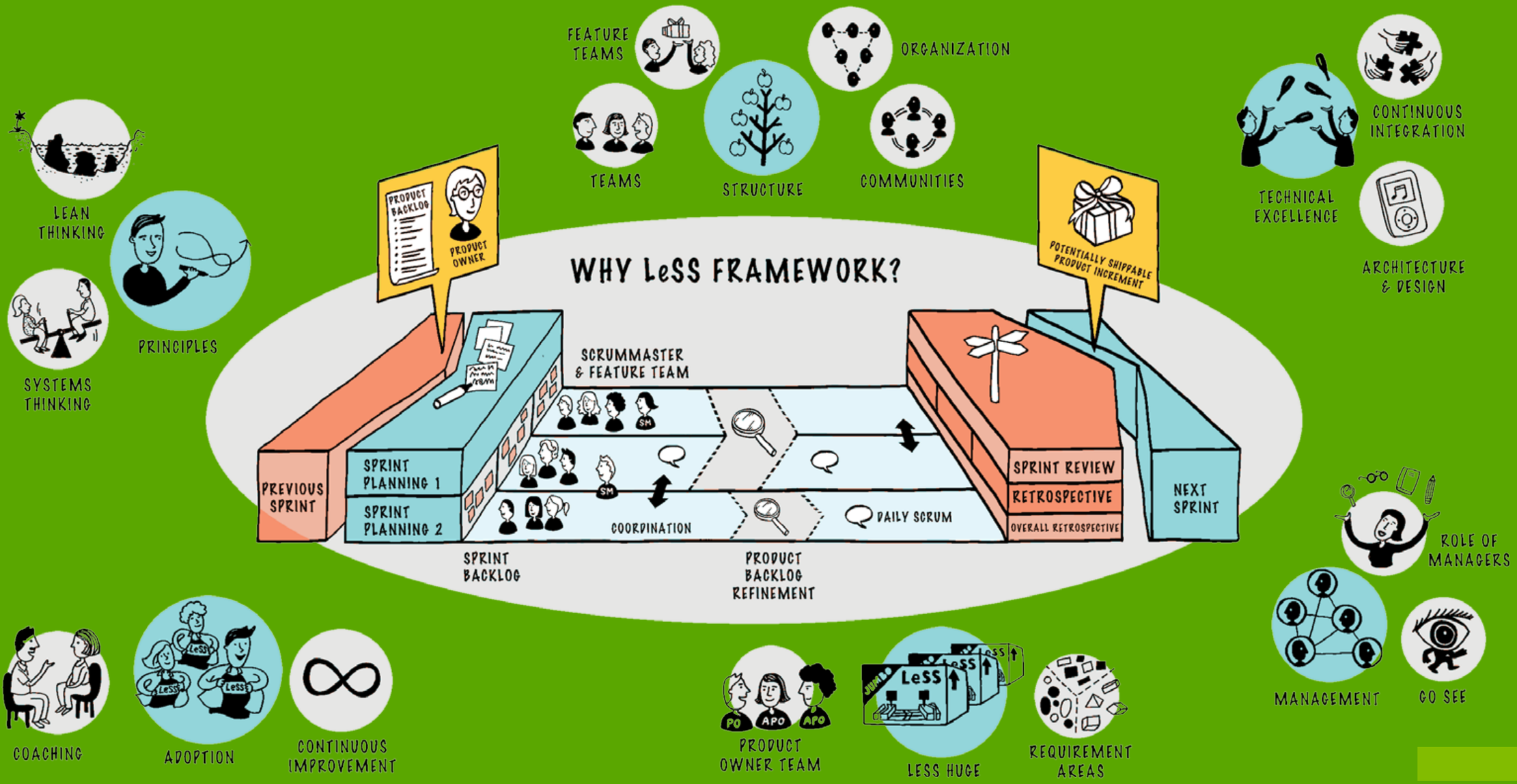
# PM, IM, CM Involvement



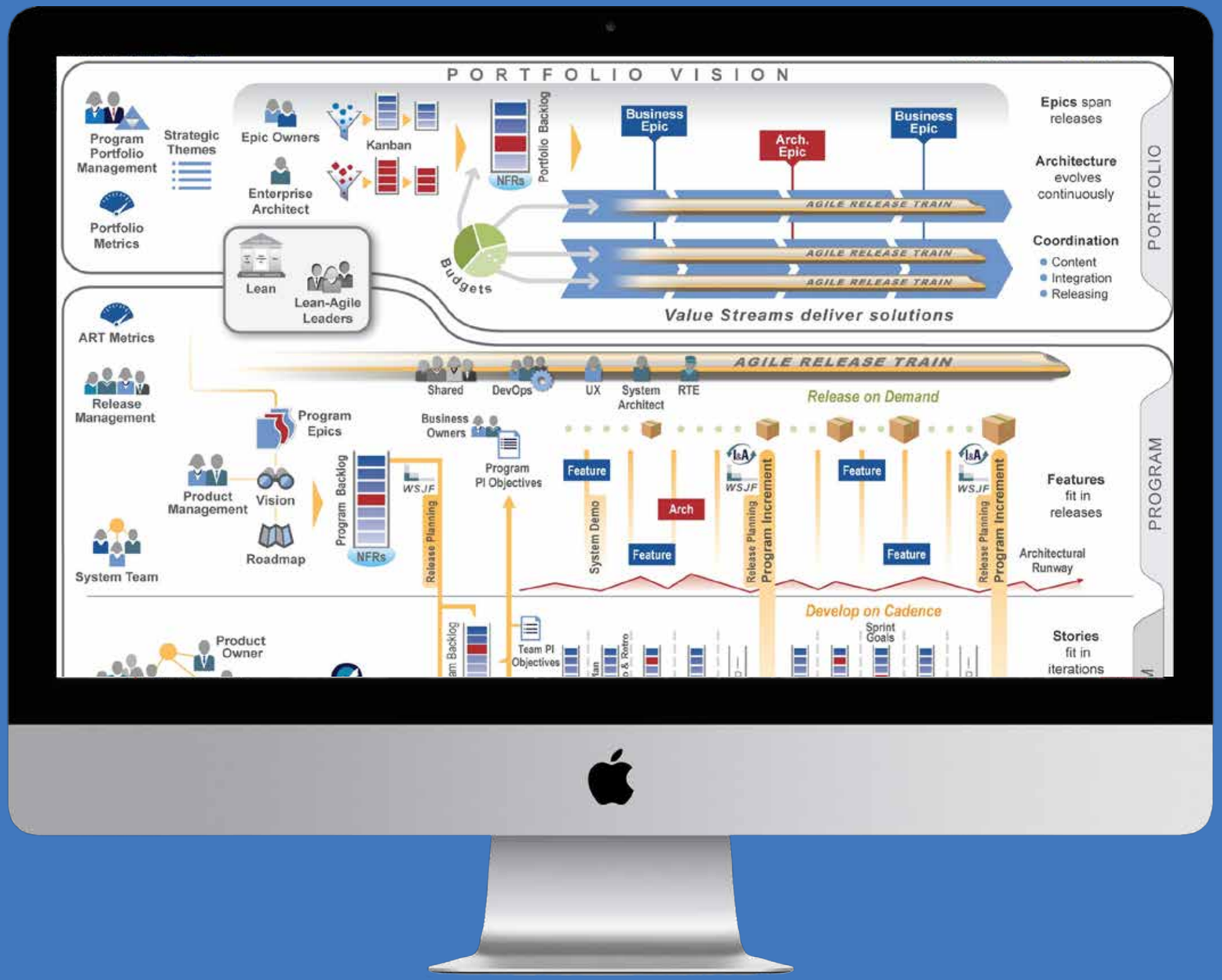
# Agile Story Hierarchy



# LeSS: Large Scale Scrum



# Scaled Agile Framework




**The Agile Program Pattern** can be applied to small and large piece of work

# Team Roles and Responsibilities

# Cross Functional Empowered Teams

IM

PM

## CORE

**5-9 People**
**Dedicated**
**Cross Functional**
**Empowered**

Business SME (Customer)
Analysts
Developers
Testers
Solution Architect

## EXTENDED

Key Stakeholders
External experts
Enterprise Architect

## GOVERNANCE

Steering committee
PMO

# Key Roles

TM The Team (the doer's)

PO The Product Owner

IM The Iteration Manager

PM The Project Manager

CM Change Manager

These are the committed parties, the people responsible for the success of the project and product. There will be other involved parties, but they won't be responsible for the success of the project or product

# Team Roles and Responsibilities

# Key Roles and Responsibilities

## Core

- IM — Iteration Manager
- SME — Subject Mater Expert
- TM — Team
- PO — Product Owner

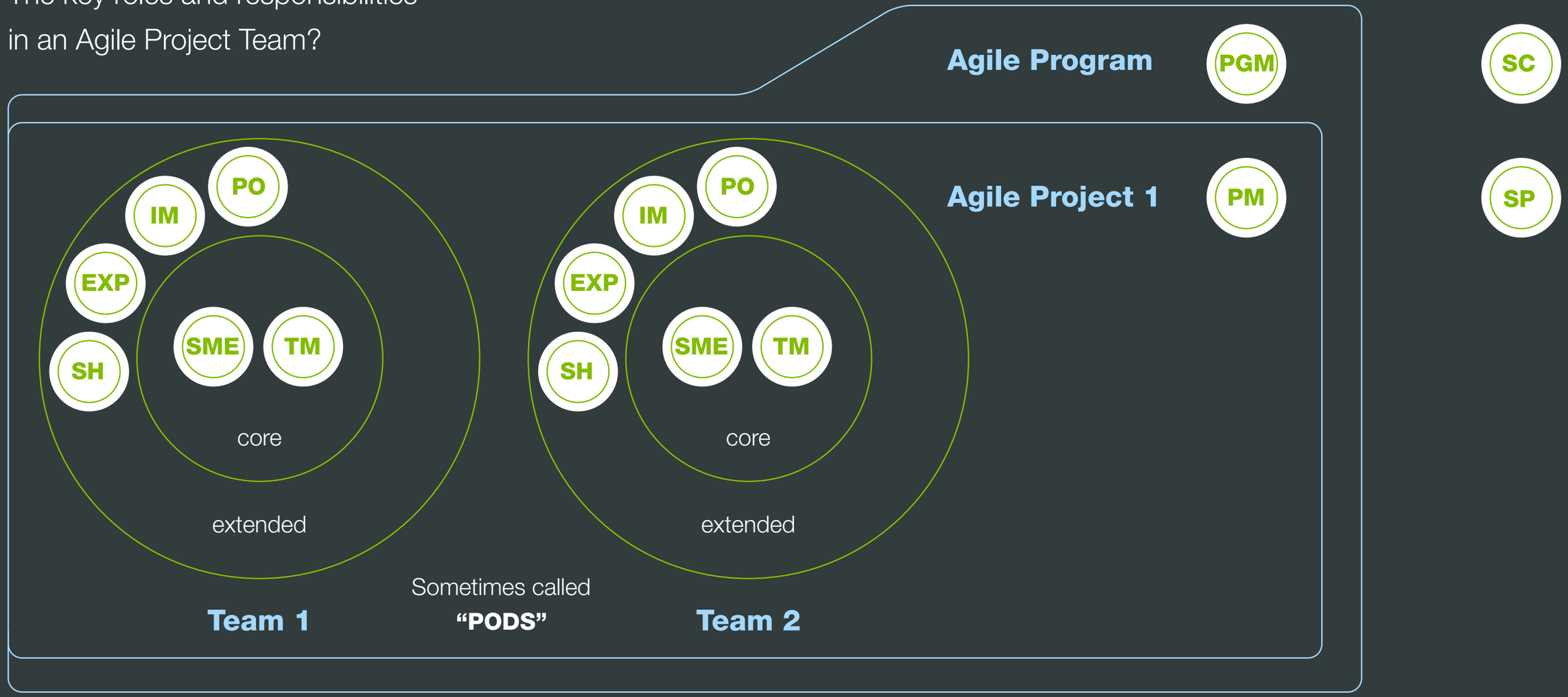
## Extended

- PGM — Program Manager
- PM — Project Manager
- SH — Stake Holder
- CM — Change Manager
- EXP — Experts

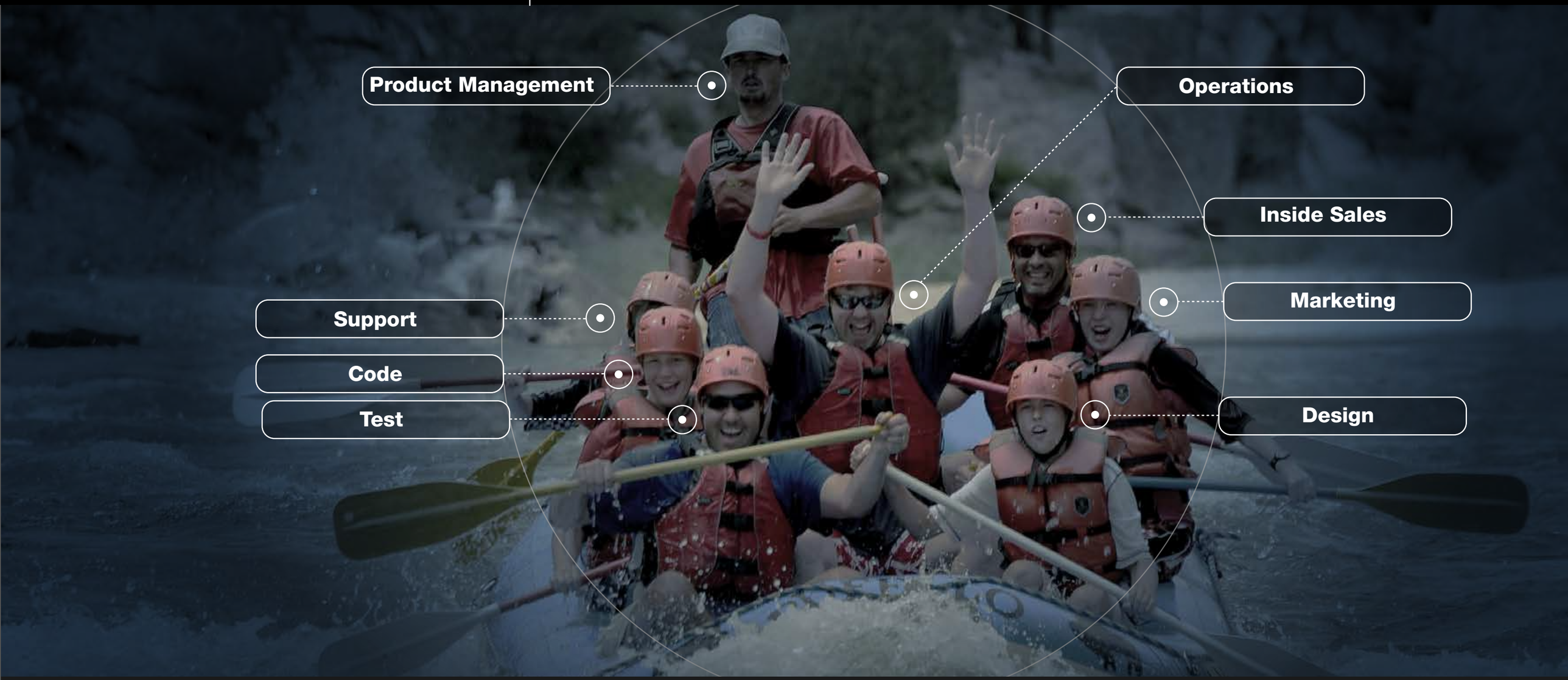
## Steering

- SC — Steering Committee
- SP — Sponsor

The key roles and responsibilities in an Agile Project Team?



# Shared Ownership




**...And shared ownership of client / user outcomes by teams**

# Team Responsibilities

Self organization

Making commitments

Meeting commitments

Management of the team – every team member is responsible for this

Respecting other members of the team

# Product Owner Responsibilities

PO

IM PM CM

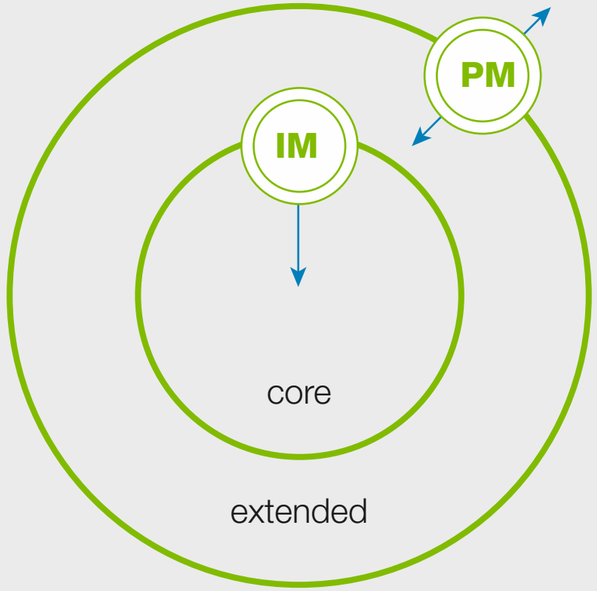
## Product Owner

- Voice of the customer/project sponsor
- Defines features of the product
- Responsible for the profitability of the product (ROI)
- Manages stakeholders and their interests
- Accepts/rejects work results
- Maintains just-enough, just-in-time feature detail
- Shares success with the team

# Iteration Manager Responsibilities



## Iteration Manager




- Servant leadership/facilitation of the Core team – POD – during delivery and deployment
- Helps remove impediments
- Acts as guardian of the iteration process/framework
- Improves lives of team members by facilitating empowerment and creativity
- Helps team improve productivity in any way possible
- Works with team and PO to ensure each iteration of the product is potentially shippable
- Works with team and PO to ensure quality is never compromised
- Coaches and helps team members in the Agile way of working
- Energize and inspire the team

# Project Manager Responsibilities



## Project Manager

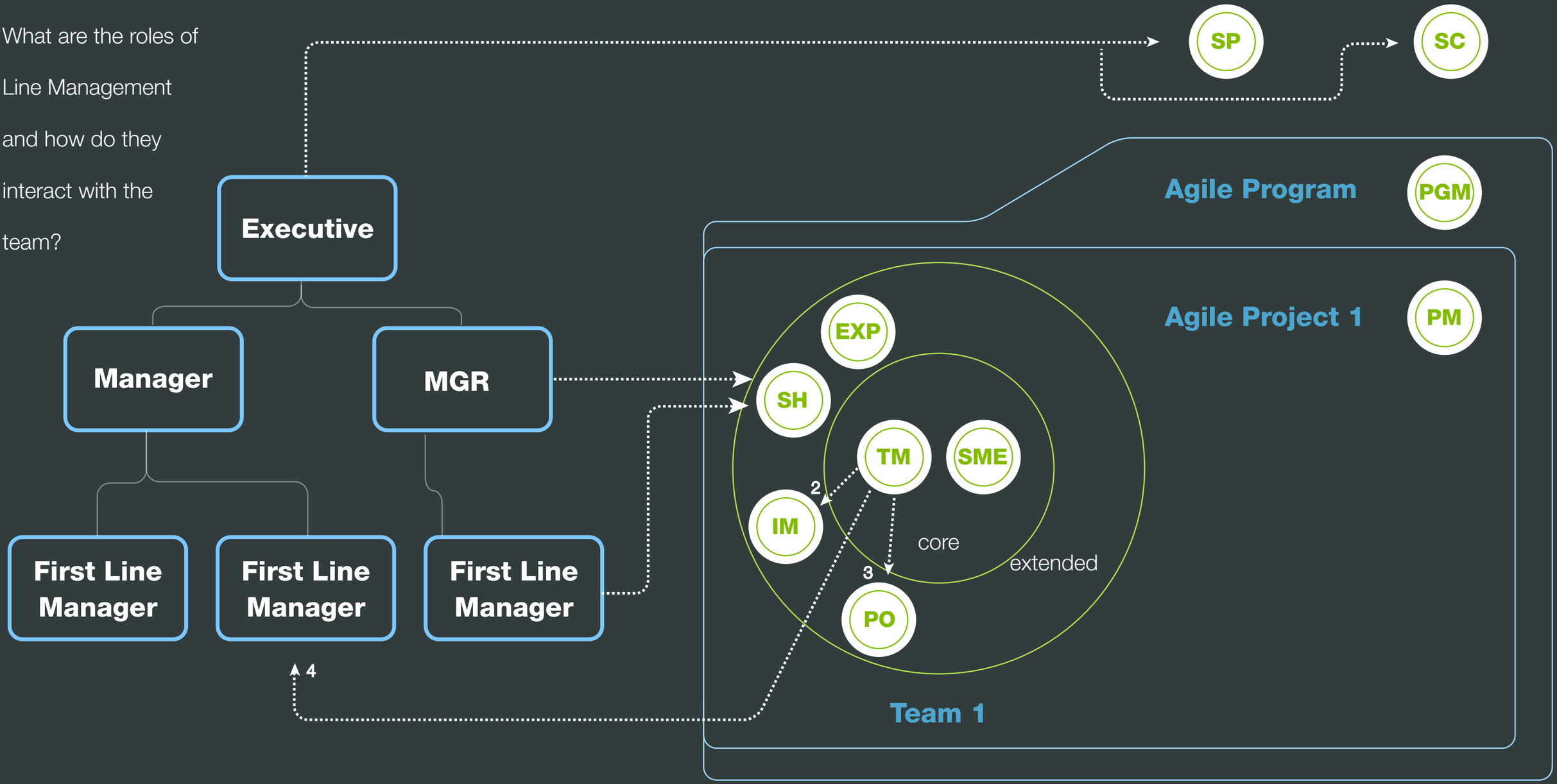
- Facilitates and manages the Discovery phases and cycles
- Serve as leadership/facilitation of the Project ( multiple Pods)
- Facilitates dependency management using the ‘Team of Teams’ model  across Projects and Pods and
- Helps remove impediments
- Acts as guardian of the overall project process/framework
- Improves lives of team members by facilitating empowerment and creativity
- Helps team improve productivity in any way possible
- Helps the PO with multi-stakeholder management
- Manages and communicates with non PO stakeholders
- Works with team and PO to ensure quality is never compromised
- Coaches and mentors the Agile way of working
- Energizes and inspires the team to greater heights

# Change Manager Responsibilities

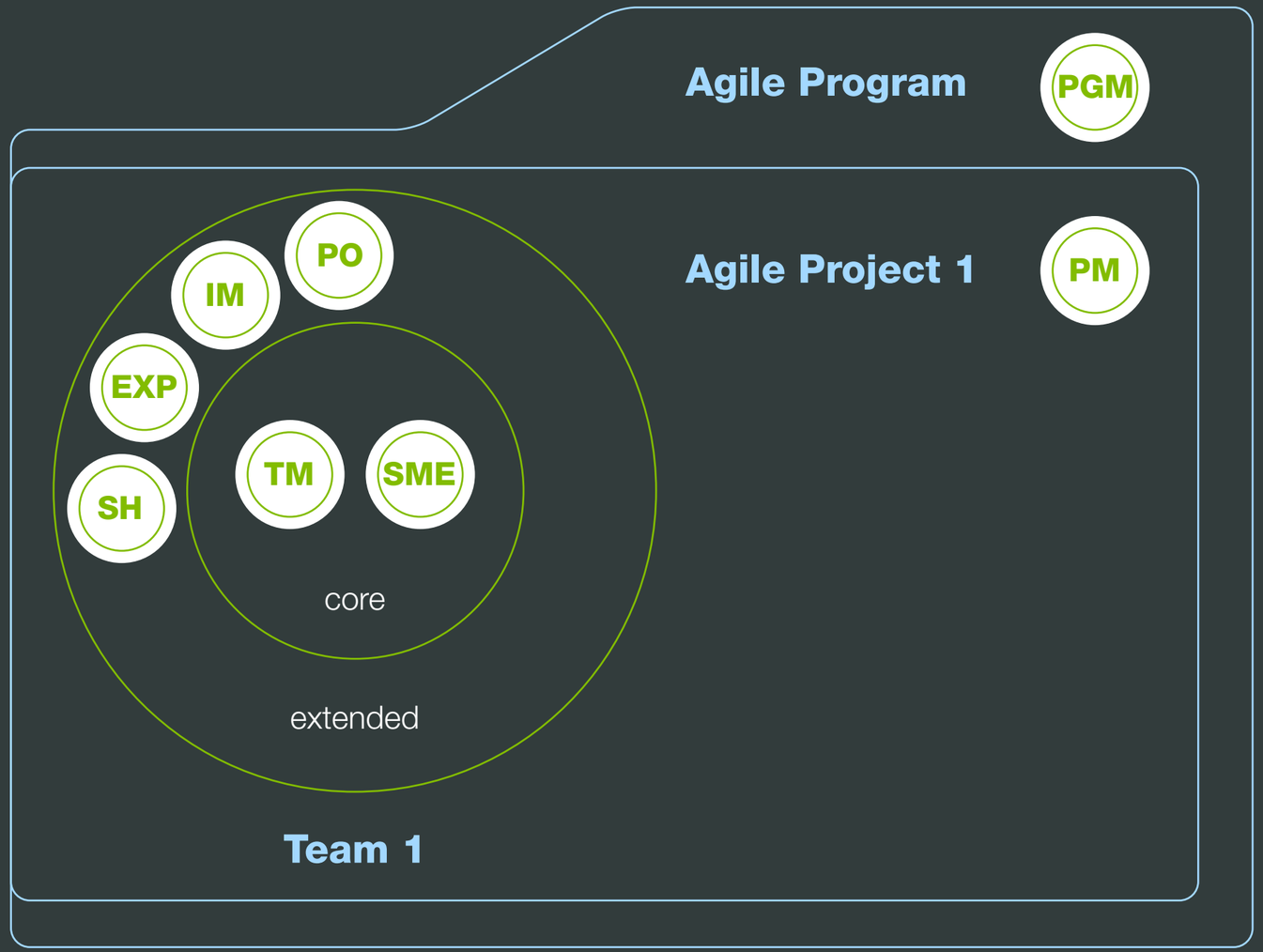

- Facilitates and manages the roll out and implementation of the project outcomes.
- Leads the Change Management Pod (sometimes called ‘Coms and Change’
- Looks after end user comms, training, process changes and overall adoption
- Closely involved in end user testing and acceptance
- Works closely with all the other IMs
- Works Agile

# Line Management

What are the roles of Line Management and how do they interact with the team?



# Steering Committee




SP SC

## Steering Committee

- Chaired by Sponsor
- Senior Execs from the BU executing the project/program
- Senior impacted BU executives – (Customers)
- Senior execs of partners or key suppliers
- Risk and Compliance (optional)

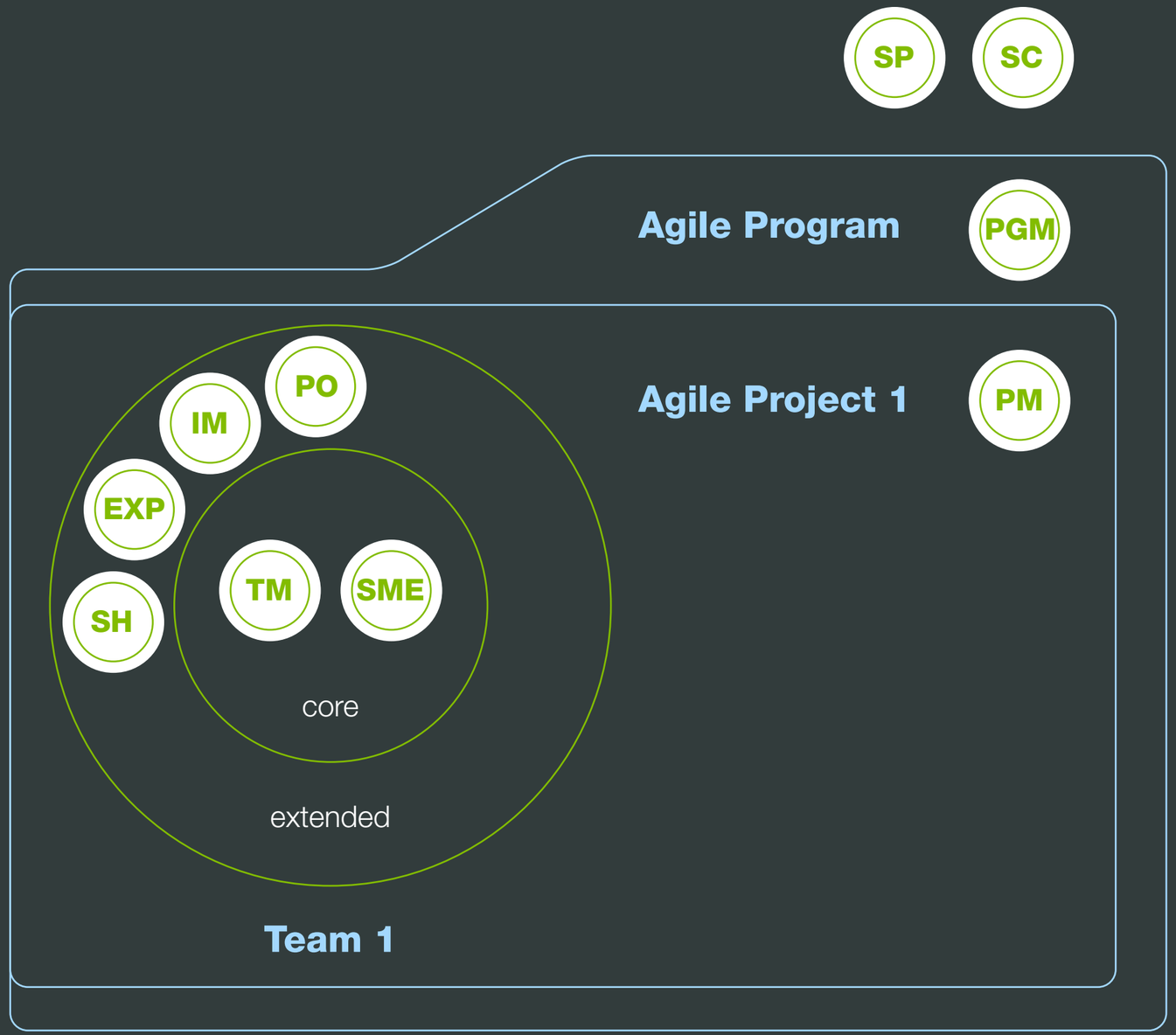
ENVISION

INSPIRE

CHALLENGE

SERVE

# Decision Work Groups




## Decision Work Groups

Senior decision makers from the business / customer groups

also called

## Product Owner Forum

To quickly make decision when there are multiple customers with conflicting of large bureaucratic processes to cut through.

# Who is Responsible for Benefits?

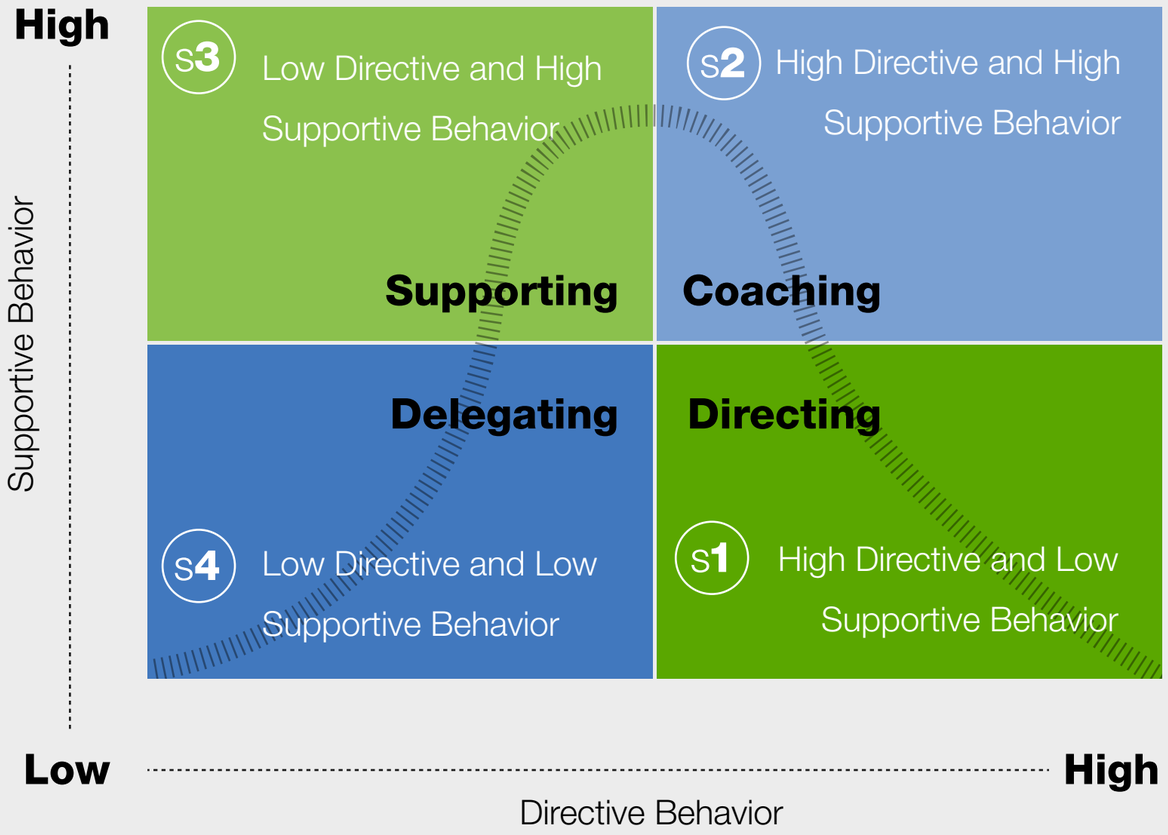
**Product Owner**

- Benefits identification?
- Benefits estimation?
- Benefits scoping?
- Ensuring the solution can deliver the benefits expected?

**Product Owner Sponsor**

- Benefits tracking?
- Benefits harvesting?

# Situational Leadership Model




The **Situational Leadership Model** is a technique used to align leadership styles with individual differences.

---

- **S1** Followers generally lack the skills required for the task but are eager to learn and willing to take direction
- **S2** Followers have some of the skills needed, but lack any real understanding of how to complete the task. These people can't succeed without some help
- **S3** Followers are capable, but lack the confidence or motivation to complete the task on their own
- **S4** Followers are skilled and even experienced at a task, and they are confident and motivated.

# Agile Project Team Characteristics

What are the Characteristics on an

## Agile Project Team?

1. Small
2. Cross functional
3. Core & cross-tended
4. Dedicated & committed team members
5. Single outcome that is shared and clearly understood by all team members
6. Very supportive and do whatever is needed to help reach the outcomes
7. Structured with a PM, IM, SME and Product Owner.
8. Clear decision making responsibilities
9. All team members collaborate and contribute continuously and effectively

agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# **The Program Pattern** used to launch and execute programs

## Strategy

### IDEA

Strategic Initiative
New Requirement
Enhancement

Problem
Opportunity

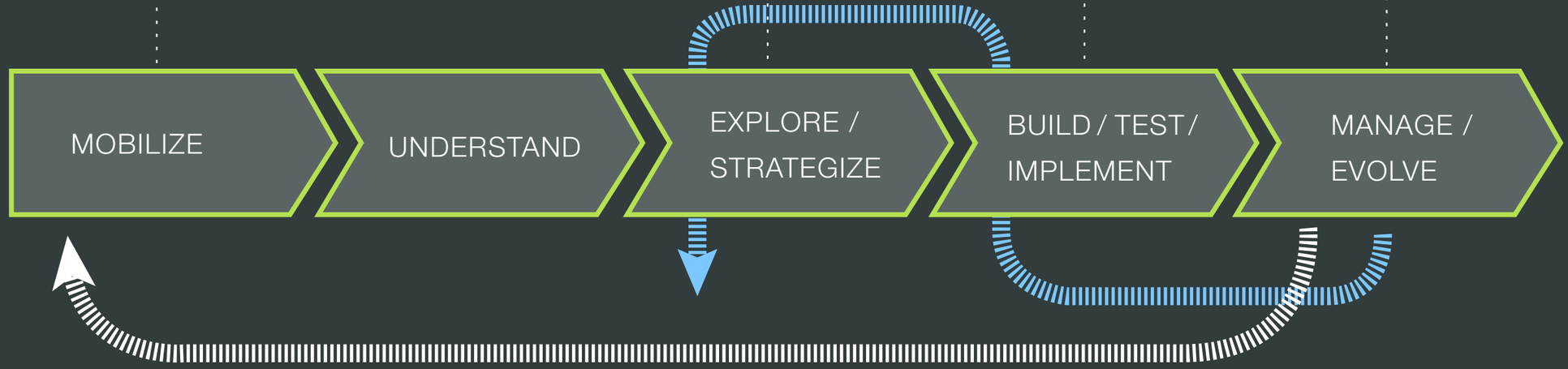
### DISCOVER

Understand & Strategize

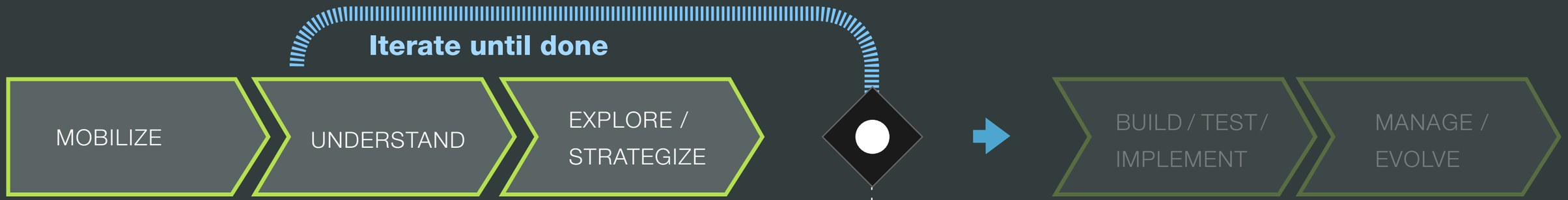
Governance /
Funding Gates

### DELIVER

Iteratively build,
test & deliver



# Discovery Practice




Discovery brief

Right stakeholders

Gate approval to start

Problem analysis

Stakeholder analysis

Desired outcome

Benefits

Blockers

Scope

Epics / Features / MVP

Risks & Dependencies

Solution options

Preferred solution

Estimation

Planning

Cost / Benefit analysis

**Program Charter (Proposal) & Gate Approval**

**Collaborate to Elaborate**

# M2.2 - Purpose and Benefits Why should we do the discovery practice?

## Objectives

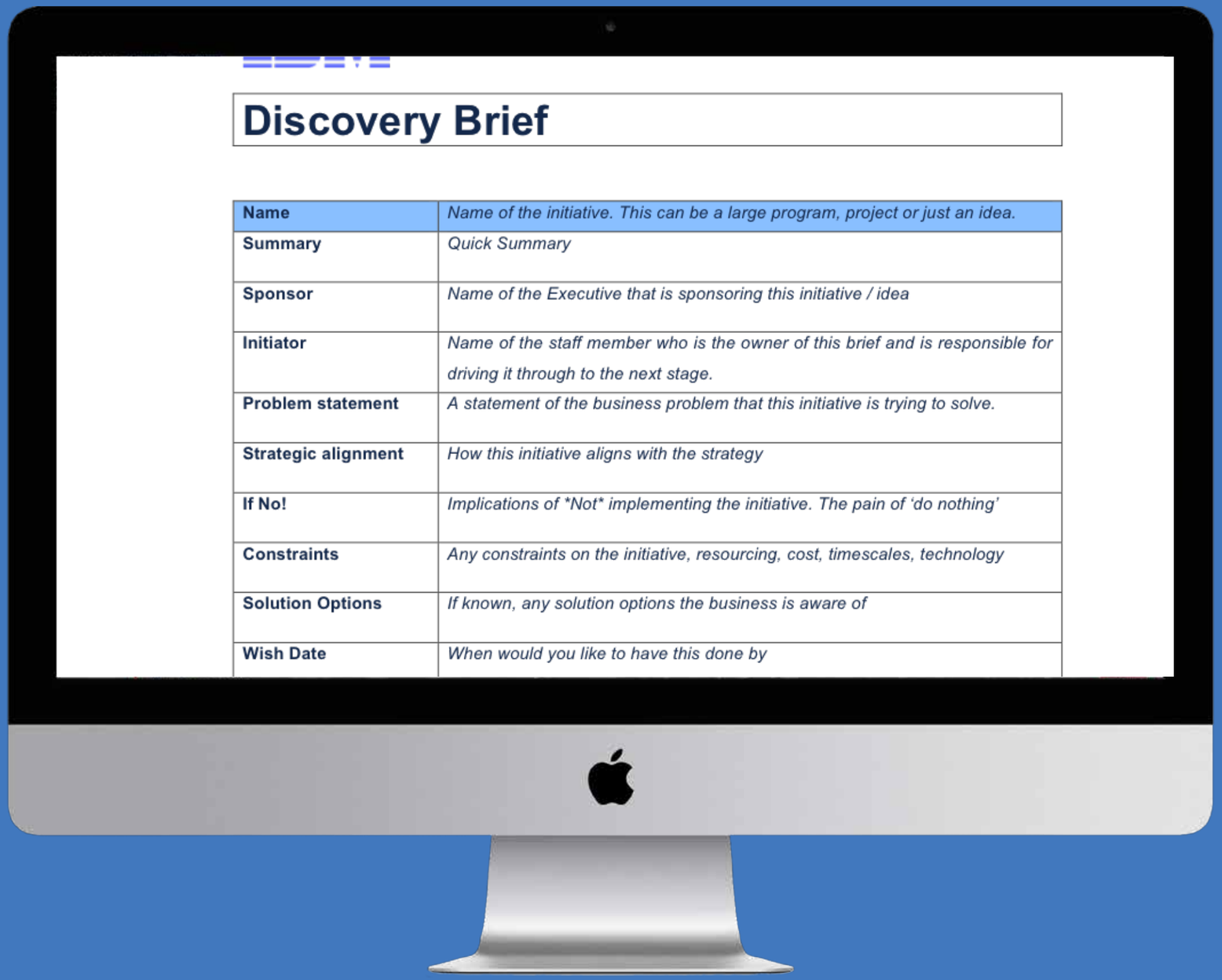
- Shared and deeper understanding
- Business Case Cost/ Benefit?
- Explore Options
- Assign scarce resource to the right initiative
- Prioritise & Manage the funnel

## Benefits

- Improved ROI
- Faster Time to Market
- Better Designs Better Products
- Happier People
- Highly Productive Teams

# M2.3 - Input to Discovery

What is the Minimum Input needed for the Discovery Practice?



## Discovery Brief

| | |
|---|---|
| **Name** | *Name of the initiative. This can be a large program, project or just an idea.* |
| **Summary** | *Quick Summary* |
| **Sponsor** | *Name of the Executive that is sponsoring this initiative / idea* |
| **Initiator** | *Name of the staff member who is the owner of this brief and is responsible for driving it through to the next stage.* |
| **Problem statement** | *A statement of the business problem that this initiative is trying to solve.* |
| **Strategic alignment** | *How this initiative aligns with the strategy* |
| **If No!** | *Implications of *Not* implementing the initiative. The pain of 'do nothing'* |
| **Constraints** | *Any constraints on the initiative, resourcing, cost, timescales, technology* |
| **Solution Options** | *If known, any solution options the business is aware of* |
| **Wish Date** | *When would you like to have this done by* |

**1-2 Hours Max!**
Filled in by the initiator

# Contents of the Discovery Brief

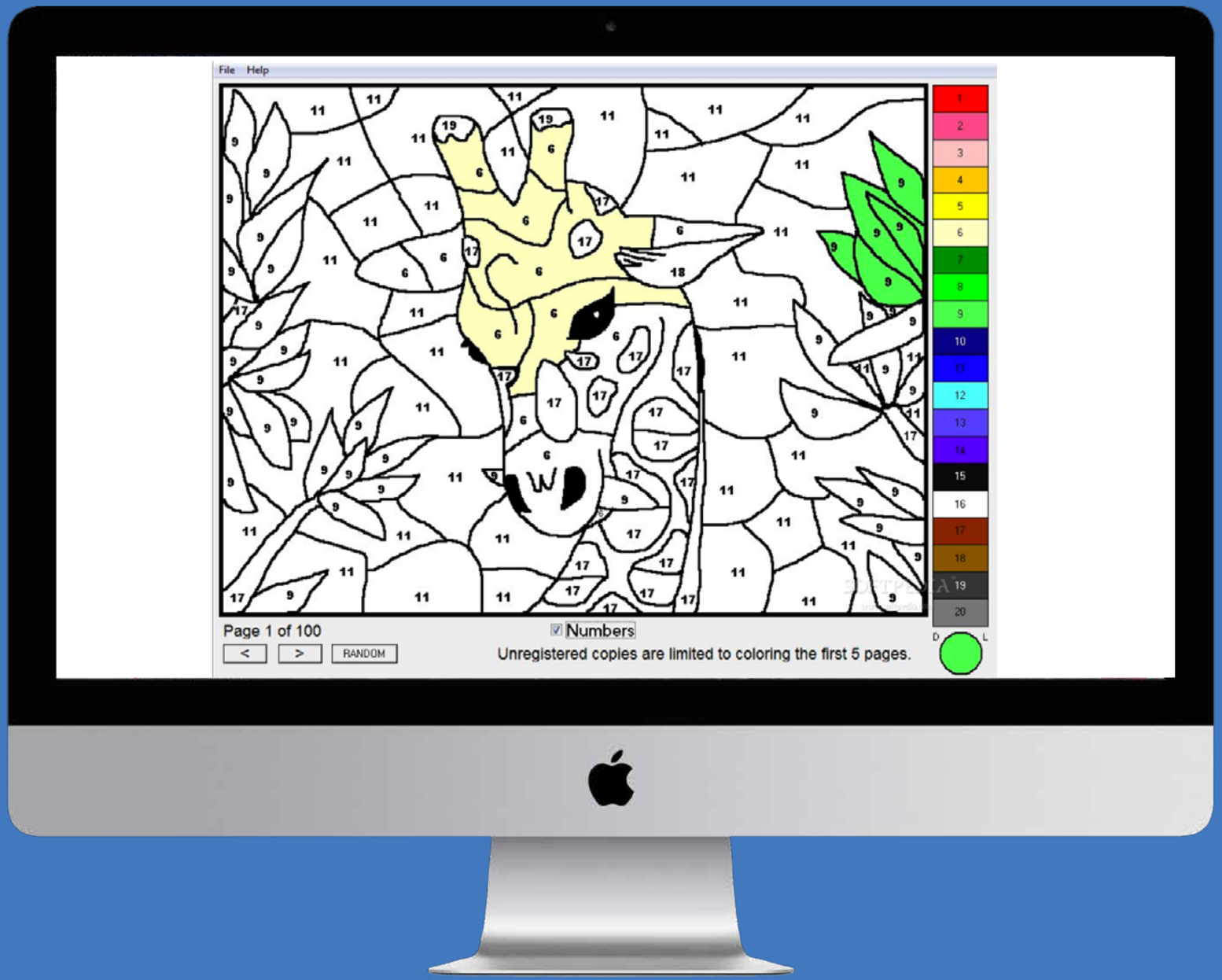
- Summary
- Sponsor
- Initiator
- Problem Statement
- Pain Impact
- Benefit Estimate
- Strategic Alignment
- Solution Options
- Constraints
- Wish Date
- Urgency
- Criticality
- Cost Appetite
- Initiative Size
- Key Stakeholders

# Who, When and for How Long

Who should be there, when shuold it be run, and how long should it take?

| Who | When | How Long |
| --- | --- | --- |
| Initiator<br>Key Stakeholders<br>Key Experts<br><br>Sponsor ( Part time) | As soon as all the key people are available<br><br>As soon as Discovery is approved if there is a gate check before discovery | Run in a facilitated workshop setting<br><br>Can be multiple workshops with breaks in between<br>Not spread over more than 3-4 weeks<br><br>Workshop time should be in the region of 1-5 days or as necessary |

# Not Paint By Numbers - Think!



# Analyzing the Problem

**Problem Grouping**

**Root Cause Analysis**

( How Come?)

**Impact Analysis**

( So What?)

# Stakeholder Analysis

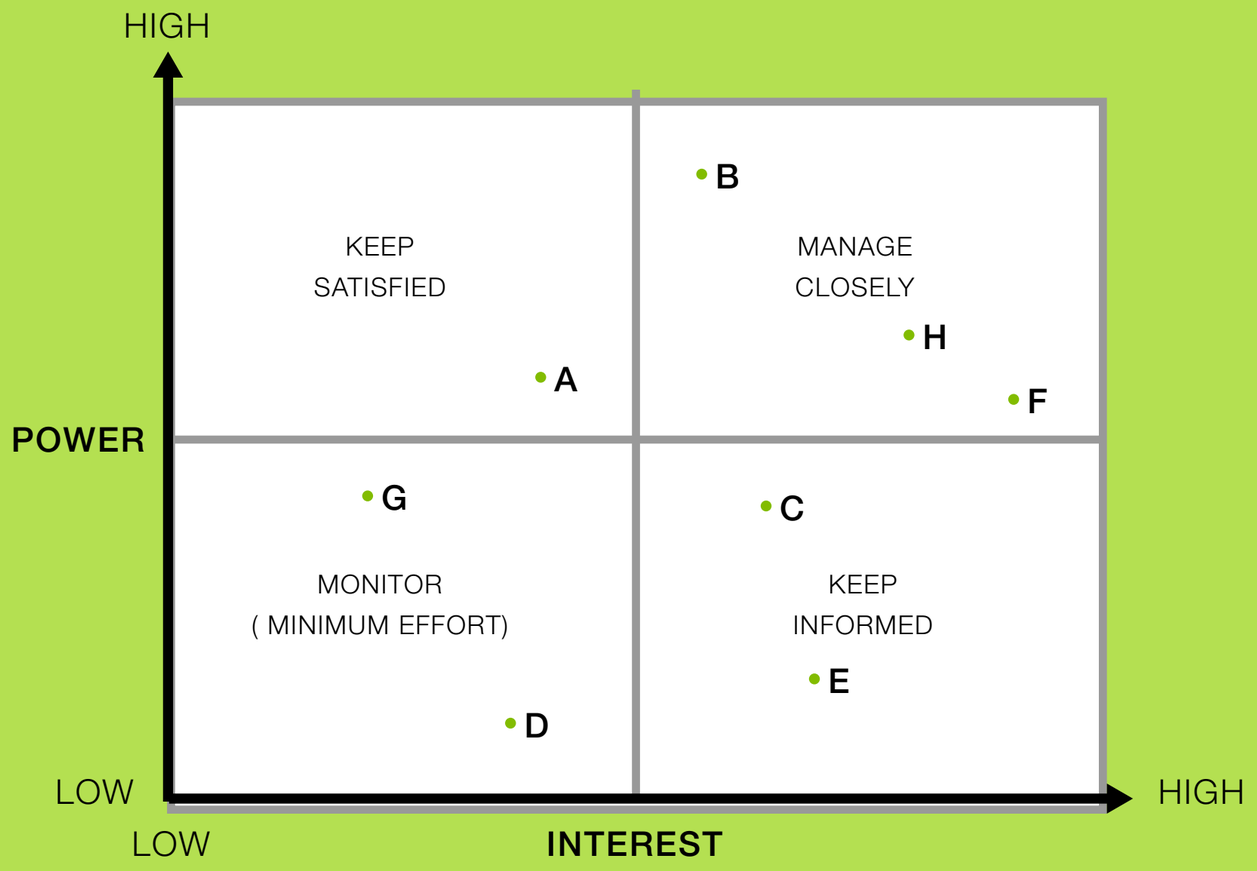
**Who is the customer?**

( Internal + External)

**High level Value Stream Map**


**Stakeholder mapping**

# Other Examples of Stakeholder Analysis




**EXAMPLE OF A POWER/INTEREST GRID**
Source: Pmbok, Fourth Edition

**It is critical that you understand the 'positioning' of your stakeholders.**

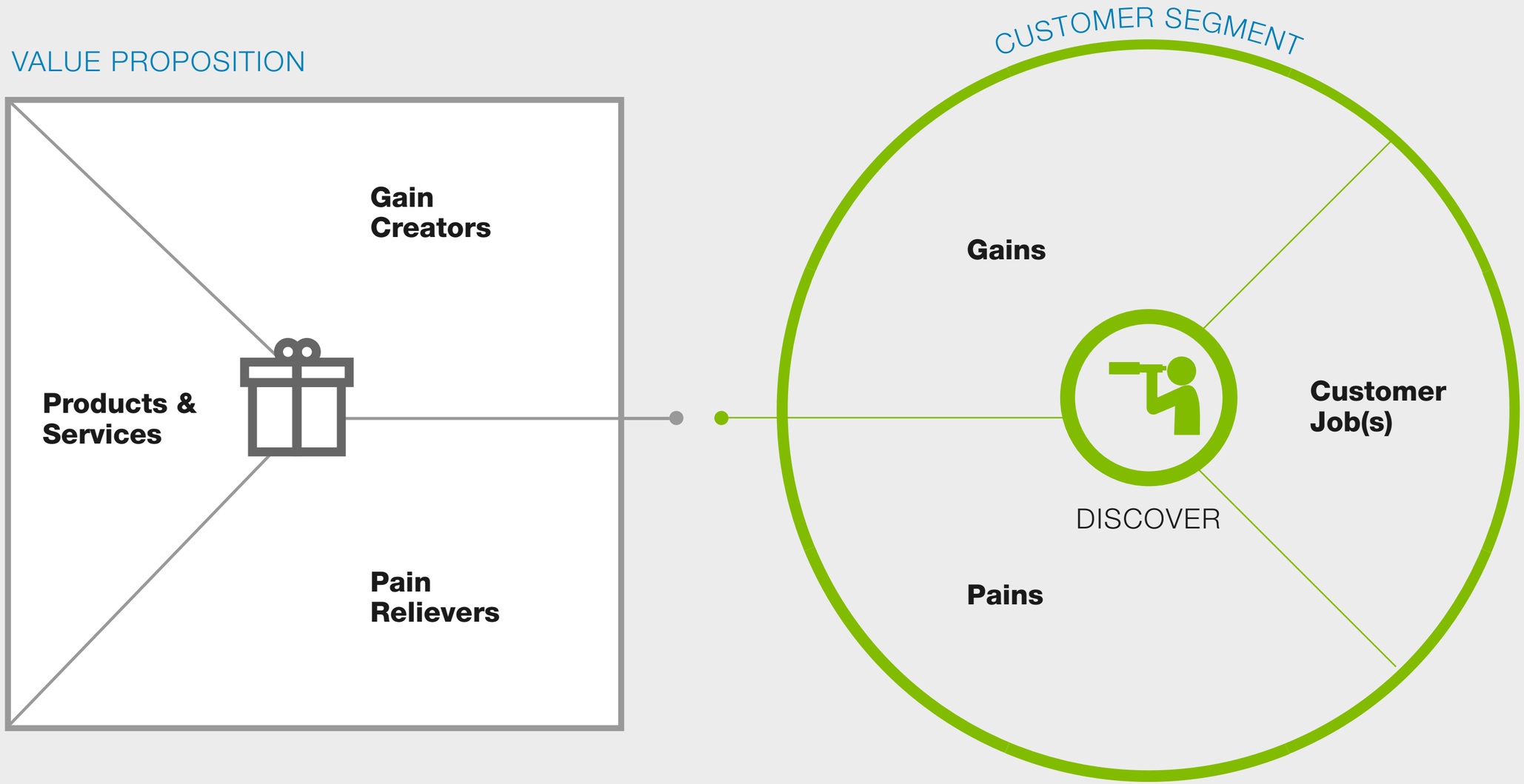
**Other 'positioning' techniques:**

Power/ Influence

Influence/ Impact

Time is a precious commodity, manage it wisely.

# Value Proposition Canvas




**DR. ALEX OSTERWALDER & DR. YES PIGNEUR**
Created by 470 practitioners from 45 countries

# Defining the Desired Outcome & Benefits

## Business Outcomes

## Estimated Benefits

Each business outcome must have one or more estimated $$ value benefits

### SMART Requirements

**S** pecific

A requirement must say **exactly what is required.** There is no ambiguity; consistent terminology; simple - avoid double requirements i.e. X and Y; appropriate level of detail

**M** easurable

Measureable when at all possible. Once the system/process has been constructed, it **can be verified** that this requirement has been met.

**A** ttainable

**Physically possible** for the system/process to exhibit that requirement under the given conditions. The consequence of unattainable requirements is that the system will never be accepted or prohibitively expensive or both

**R** ealisable

**Possible to achieve** this requirement given what is known about the constraints under which the project must be developed.

**T** raceable

Traceabiilty is the **ability to trac**e ( forwards and backwards) a requirement from its conception through its completion/deployment

## SMART Goals

Each objective should have one or more SMART goals

(Specific: Measureable, Actionable, Realistic, Time bound)

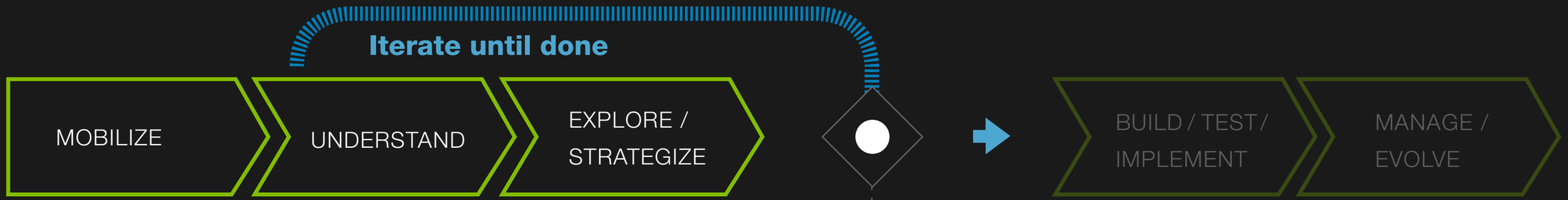
agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# Discovery Practice




Discovery brief

Right stakeholders

Gate approval to start

Problem analysis

Stakeholder analysis

Desired outcome

Benefits

Blockers

Scope

Epics / Features / MVP

Risks & Dependencies

Solution options

Preferred solution

Estimation

Planning

Cost / Benefit analysis

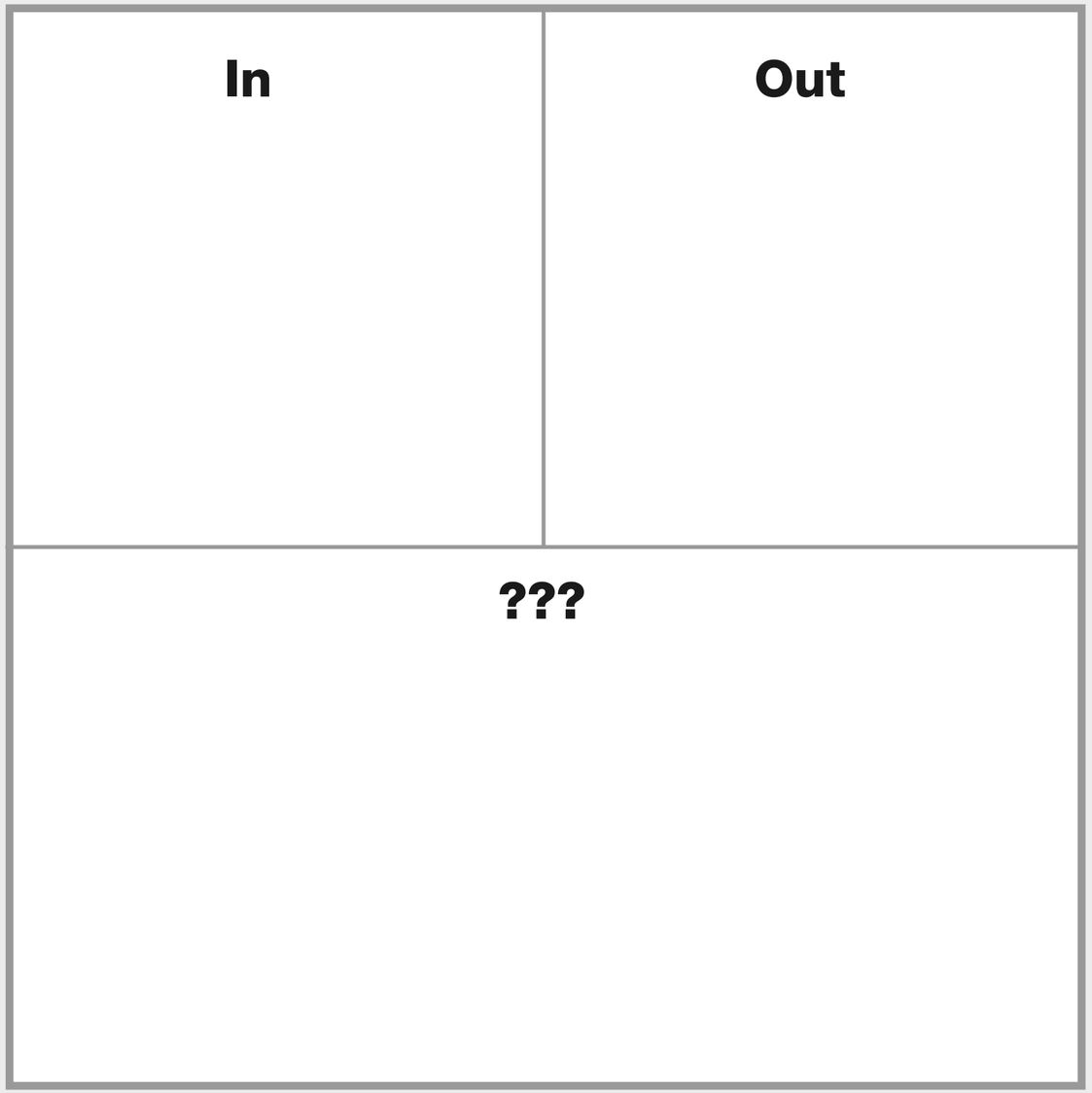
**Program Charter (Proposal) & Gate Approval**

**Collaborate to Elaborate**

# Identifying the Blockers

How should we identify the blockers to success?

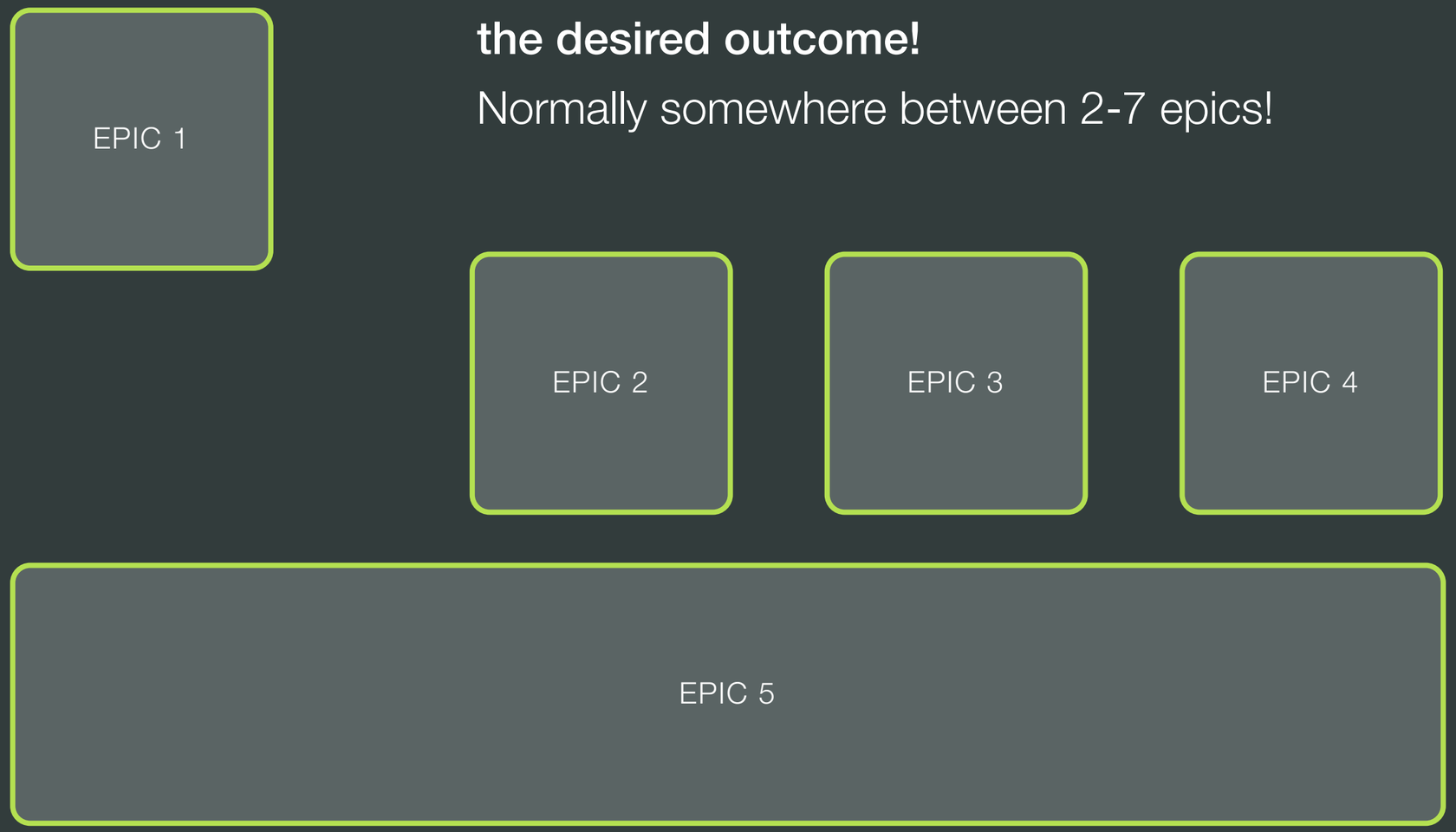
# **Agreeing Scope** How should we surface the work that needs to be done?



# **Surfacing the Epics** How should we surface the work that needs to be done?

**Those big pieces of work that need to be done to achieve the desired outcome!**

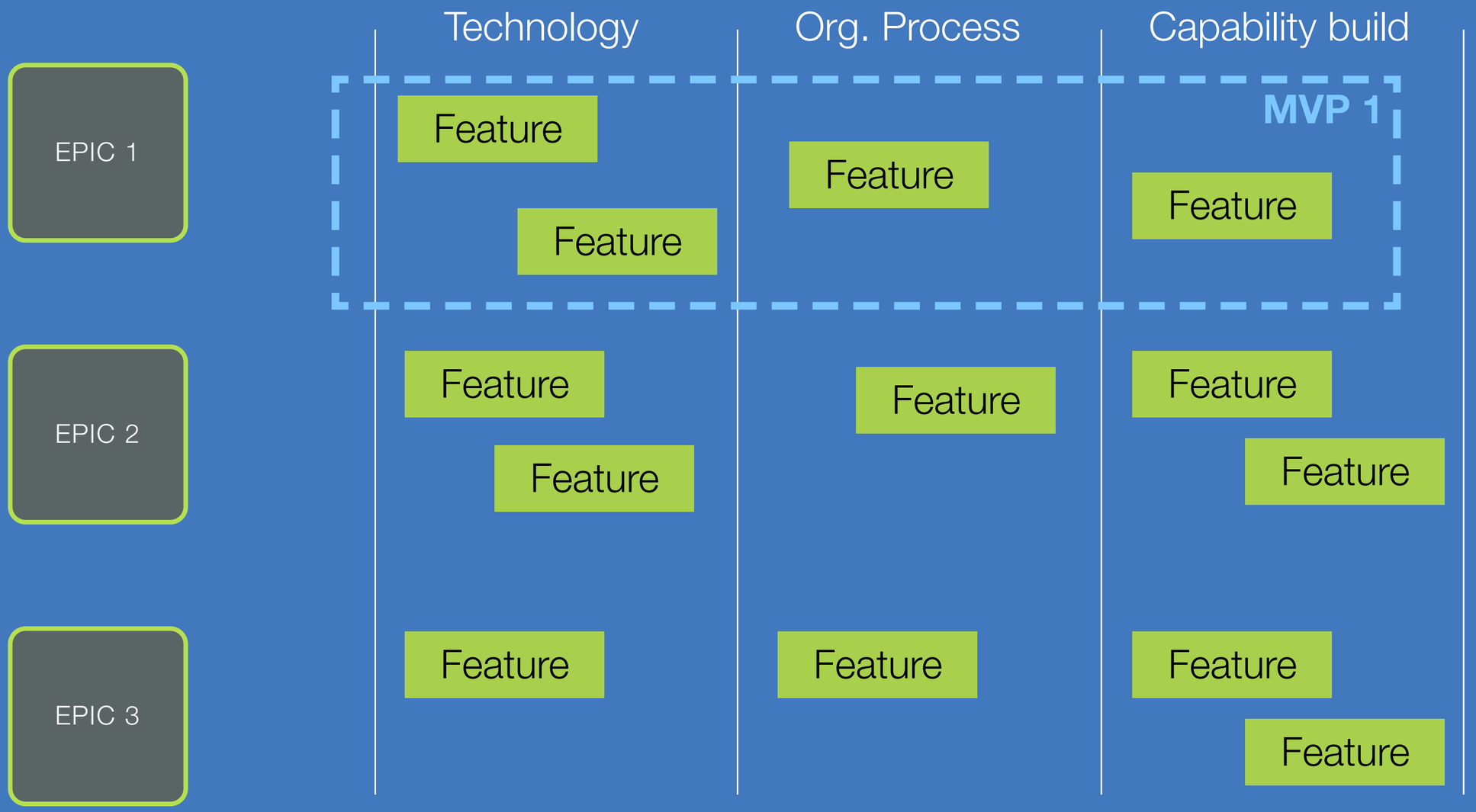
Normally somewhere between 2-7 epics!



# Features and MVP Epics and Features

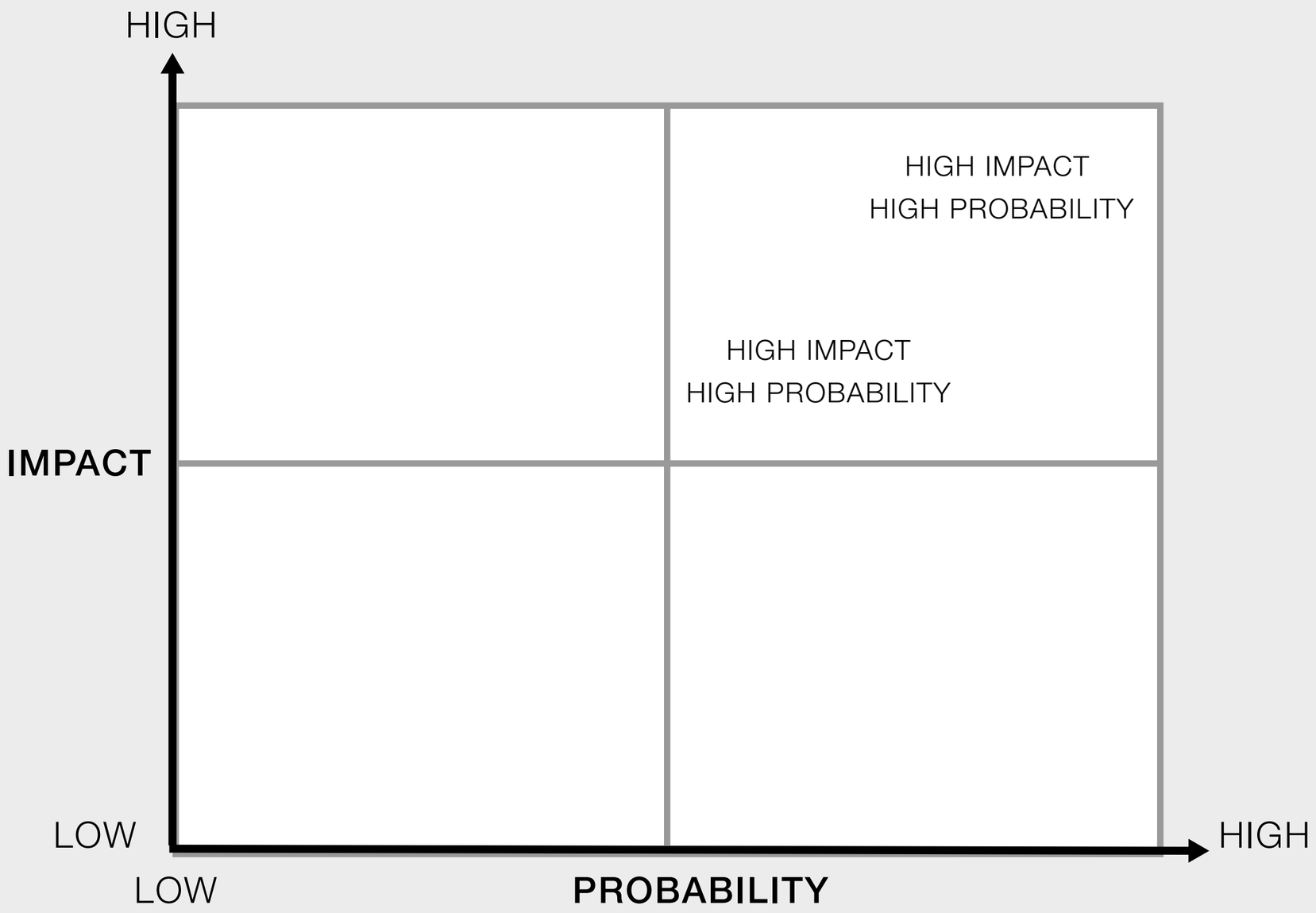
## Generating Features

## Solution Aspects




**Use:**
Design thinking
Process maps
Brainstorming

# Risks


HIGH
HIGH IMPACT
HIGH PROBABILITY
HIGH IMPACT
HIGH PROBABILITY
IMPACT
LOW
HIGH
LOW
PROBABILITY

# Dependencies

## Program

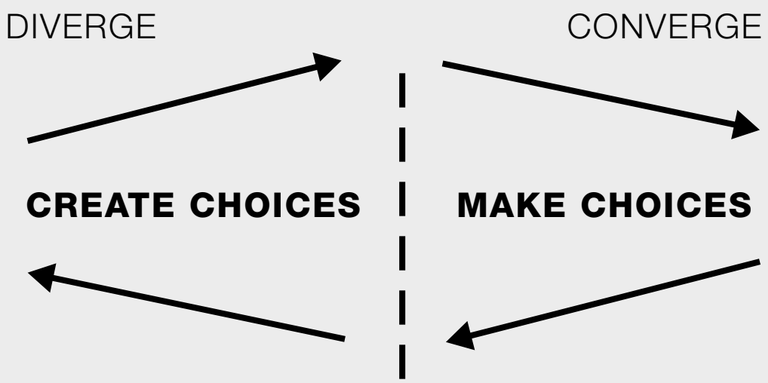
| | Internal | External |
|---|---|---|
| Technology | | |
| People | | |
| Process | | |

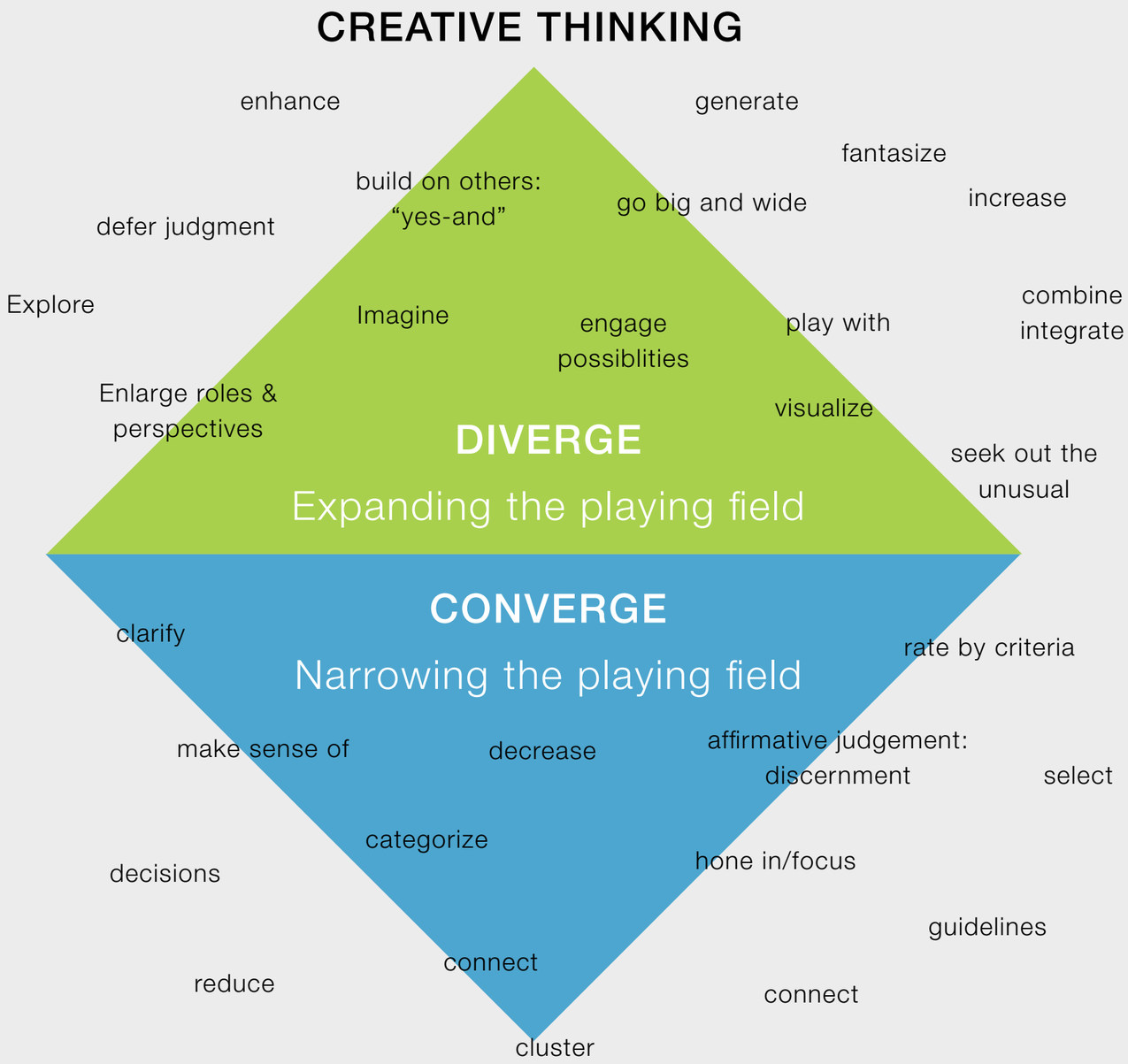
## Design Thinking:

"Determine the real problem...

Consider a wide range of potential solutions...

Converge on a proposal."



# **Preferred Solution** How to select a solution strategy?

**Analyze all aspects.**

| Solution | People | Process | Tech |
|---|---|---|---|
| Option 1 | ++ | + | - |
| Option 2 | ++ | + | ++ |
| Option 3 | + | ++ | ++ |
| | | | |
| | | | |

# **Preferred Solution** How to select a solution strategy?

## Intel consensus model

| YES | YES but... | Don’t Know | NO |
| --- | --- | --- | --- |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

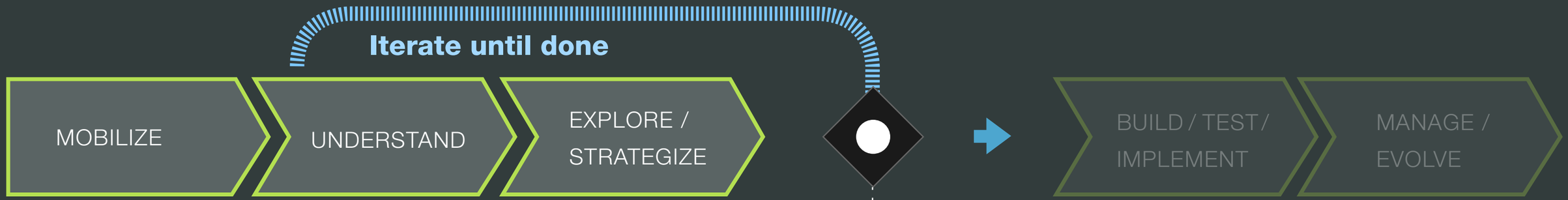
agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# Discovery Practice




Discovery brief

Right stakeholders

Gate approval to start

Problem analysis

Stakeholder analysis

Desired outcome

Benefits

Blockers

Scope

Epics / Features / MVP

Risks & Dependencies

Solution options

Preferred solution

Estimation

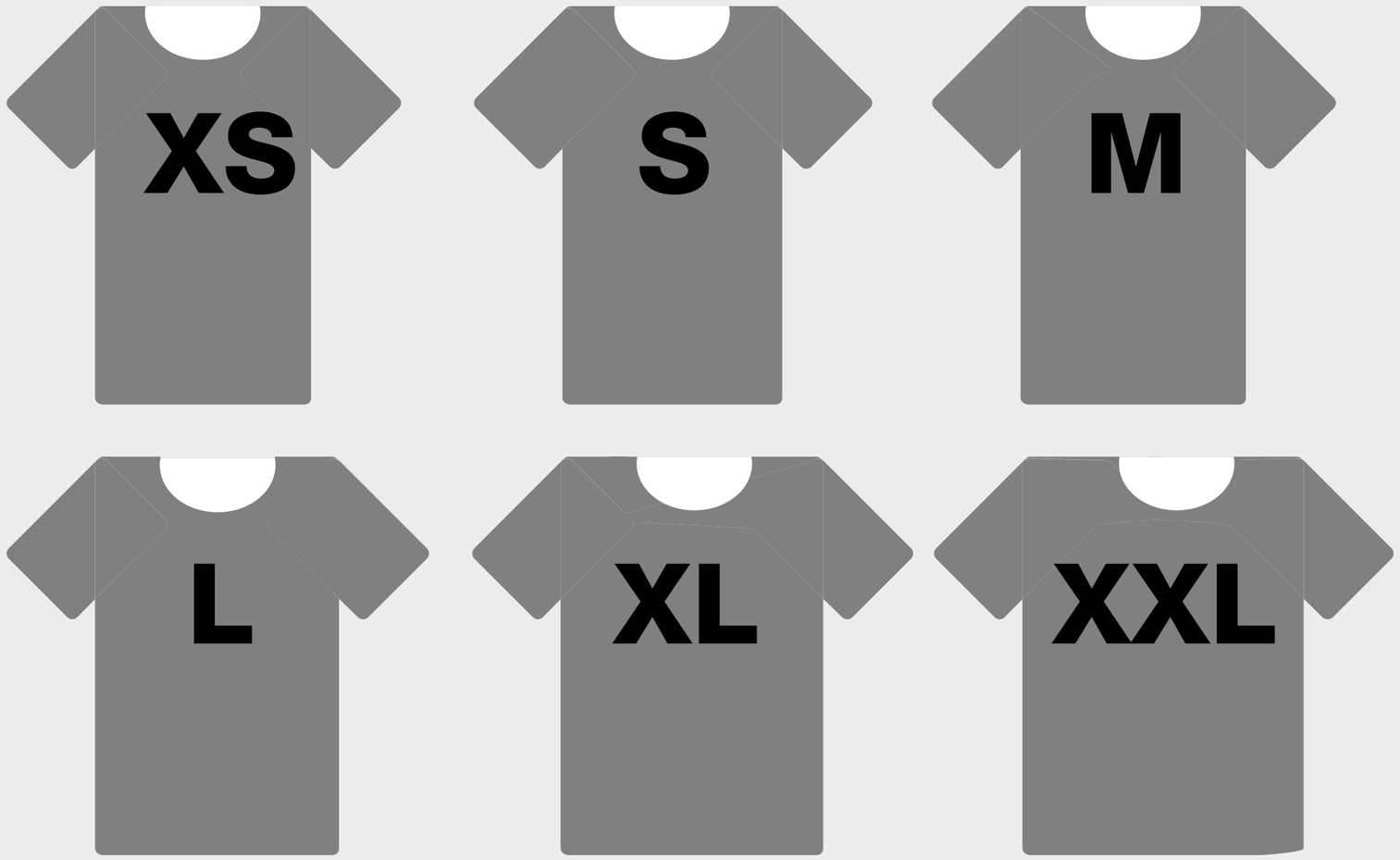
Planning

Cost / Benefit analysis

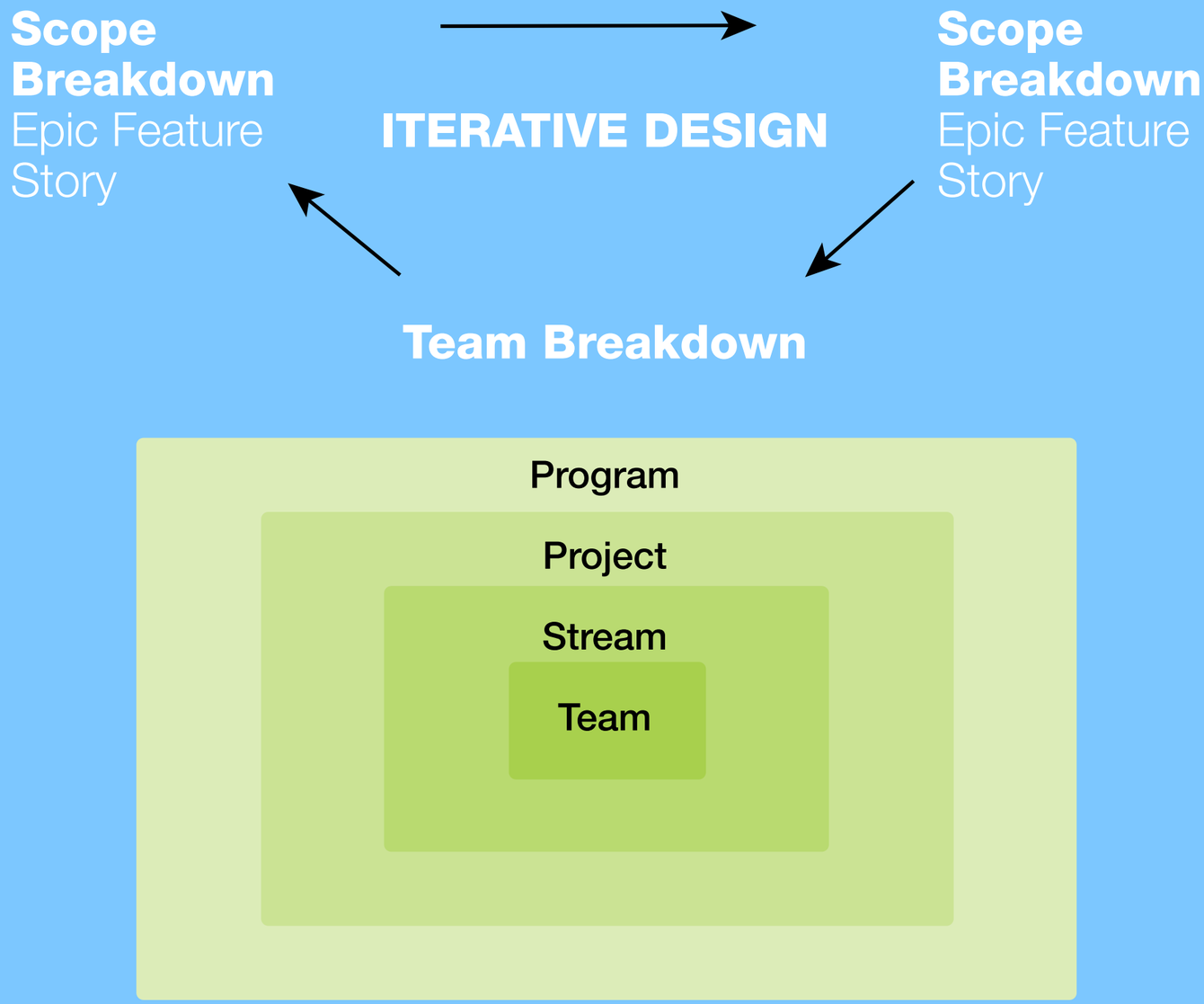
**Program Charter (Proposal) & Gate Approval**

**Collaborate to Elaborate**

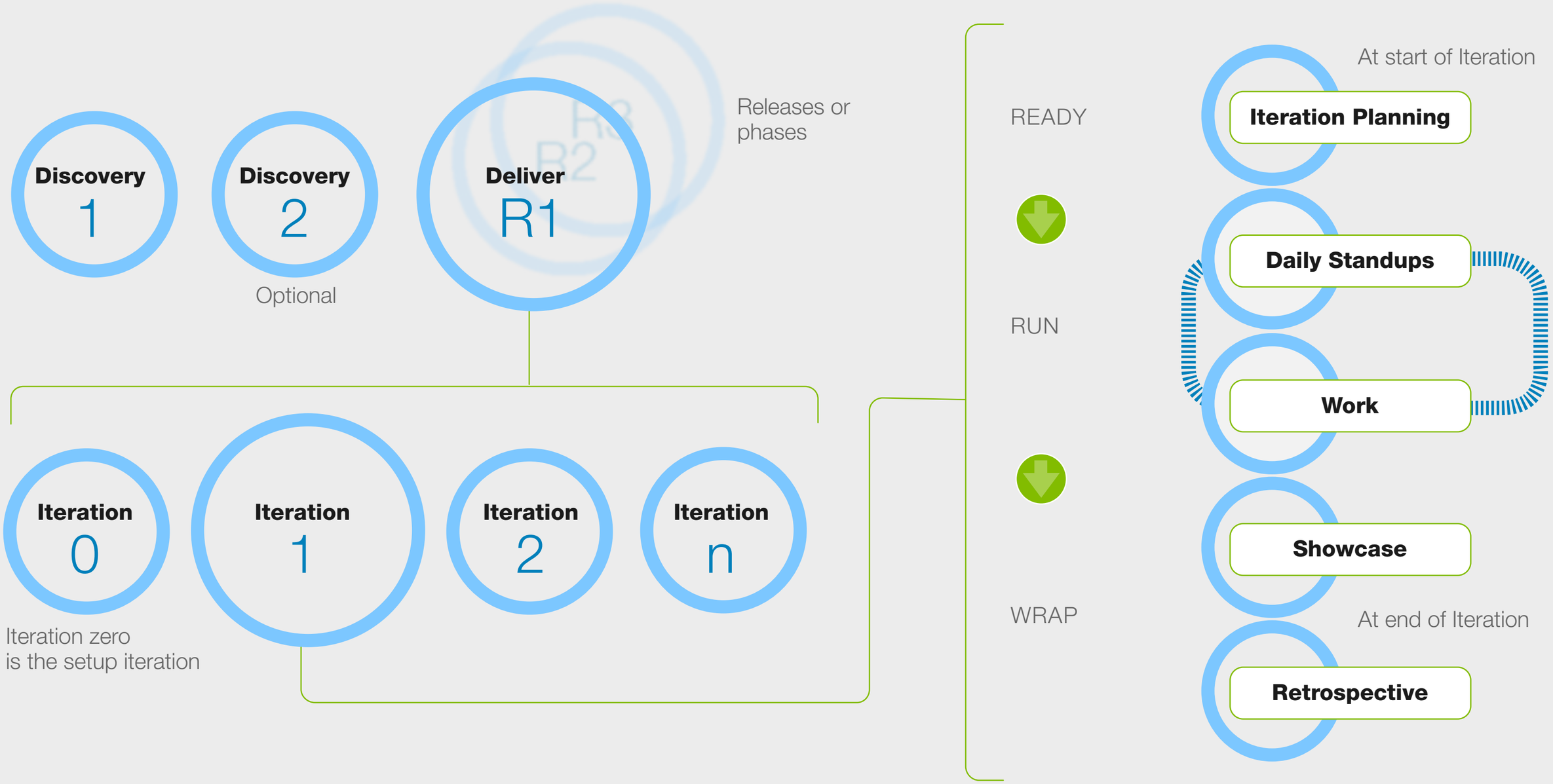
# **Estimating and costing the work** How should we estimate the work and costs?



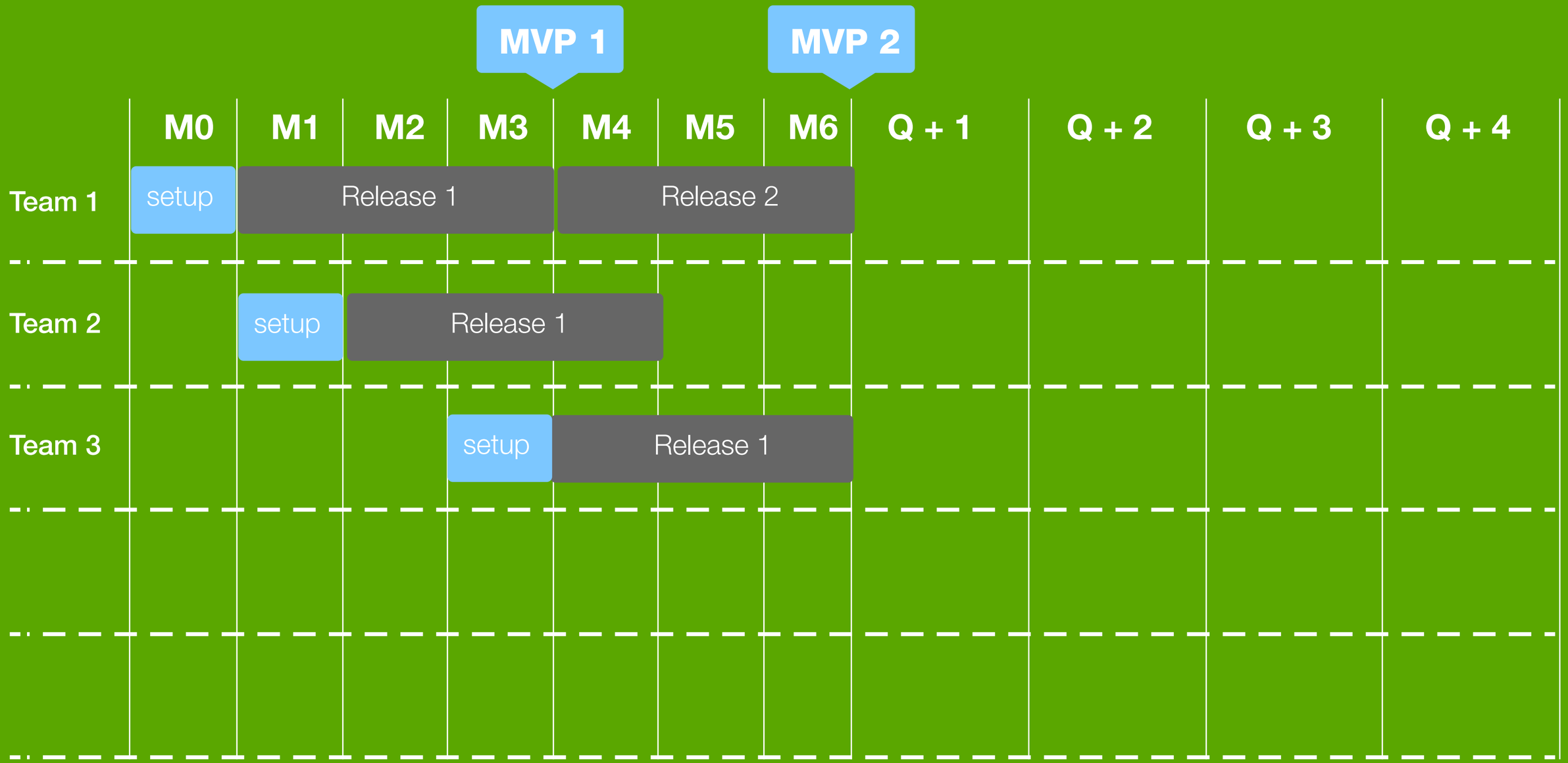
# M1.1.1 - Planning: Work and team breakdown



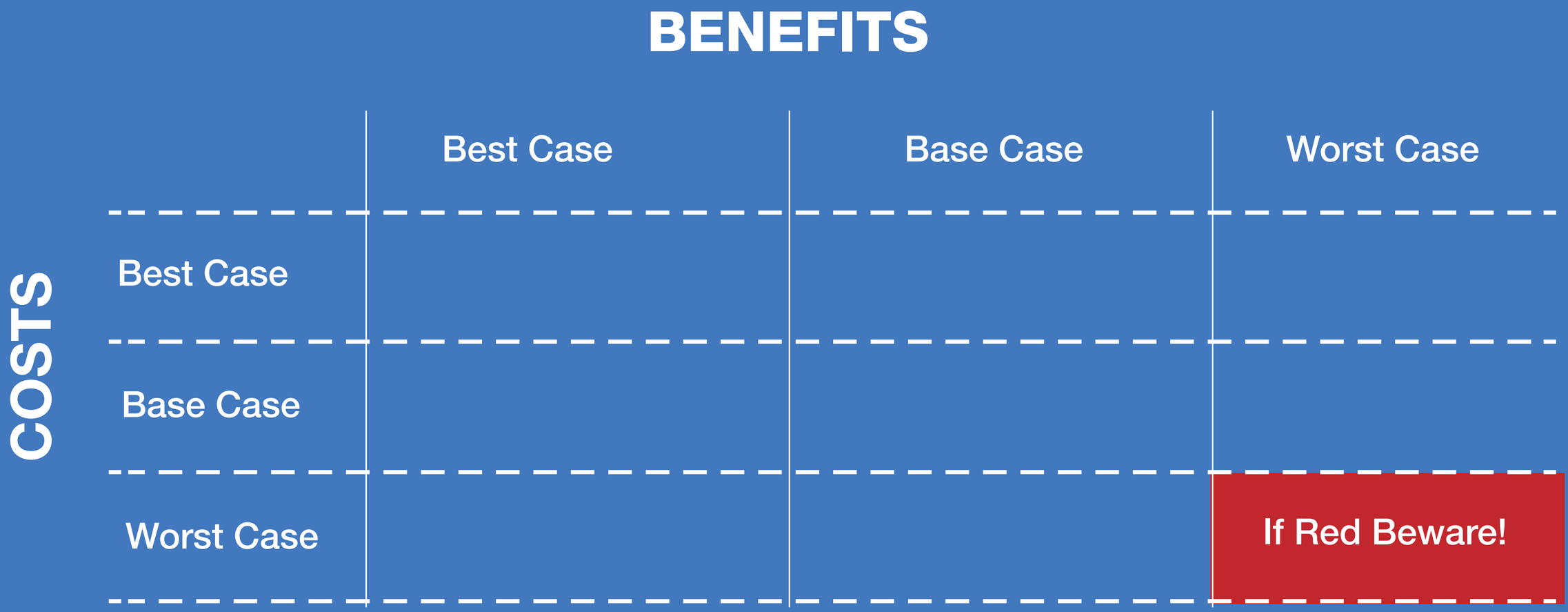
# The Lifecycle of Delivery



# M3.3 - Planning the work How should we plan the work?

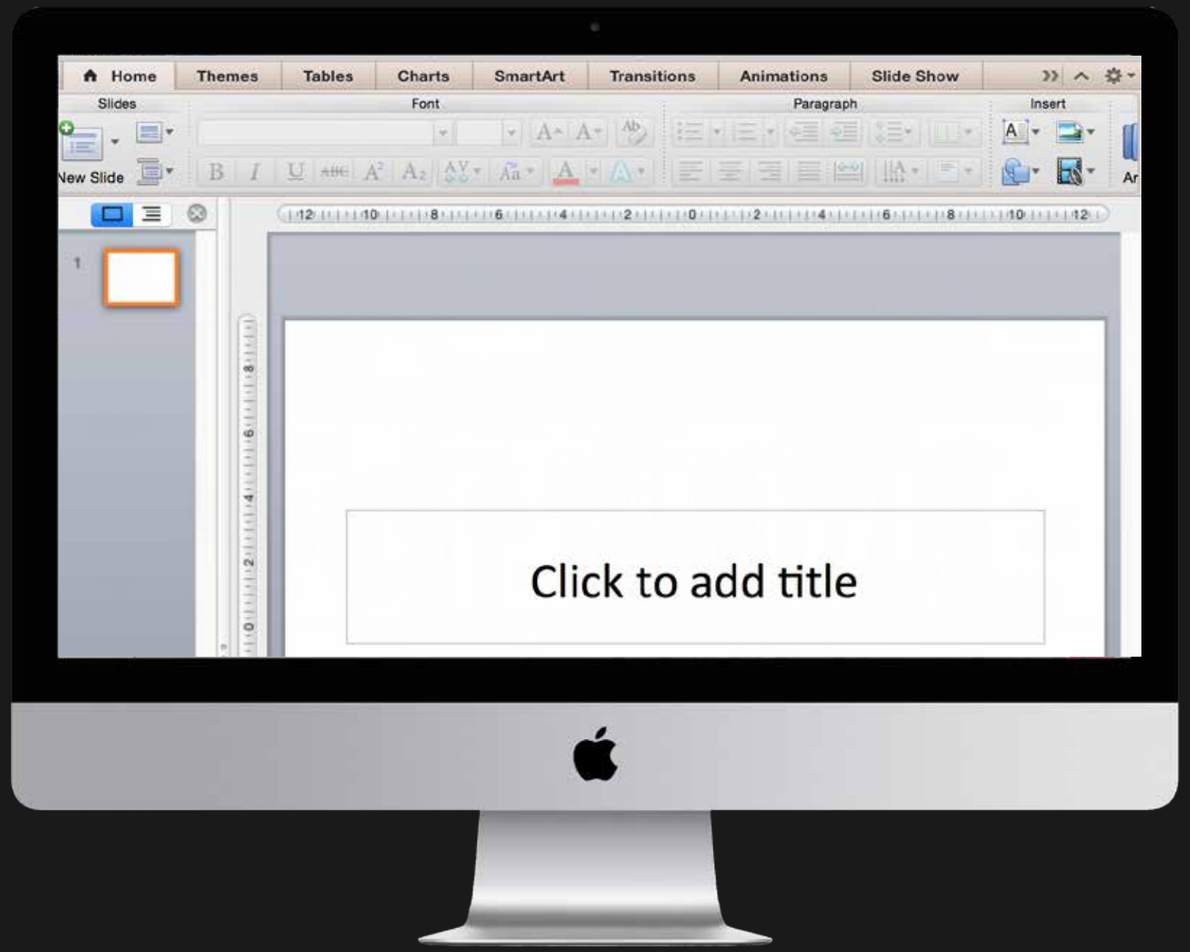


**Cost Benefit Analysis** Should we do this piece of work? Is it feasible?



# M3.5 Making the Proposal

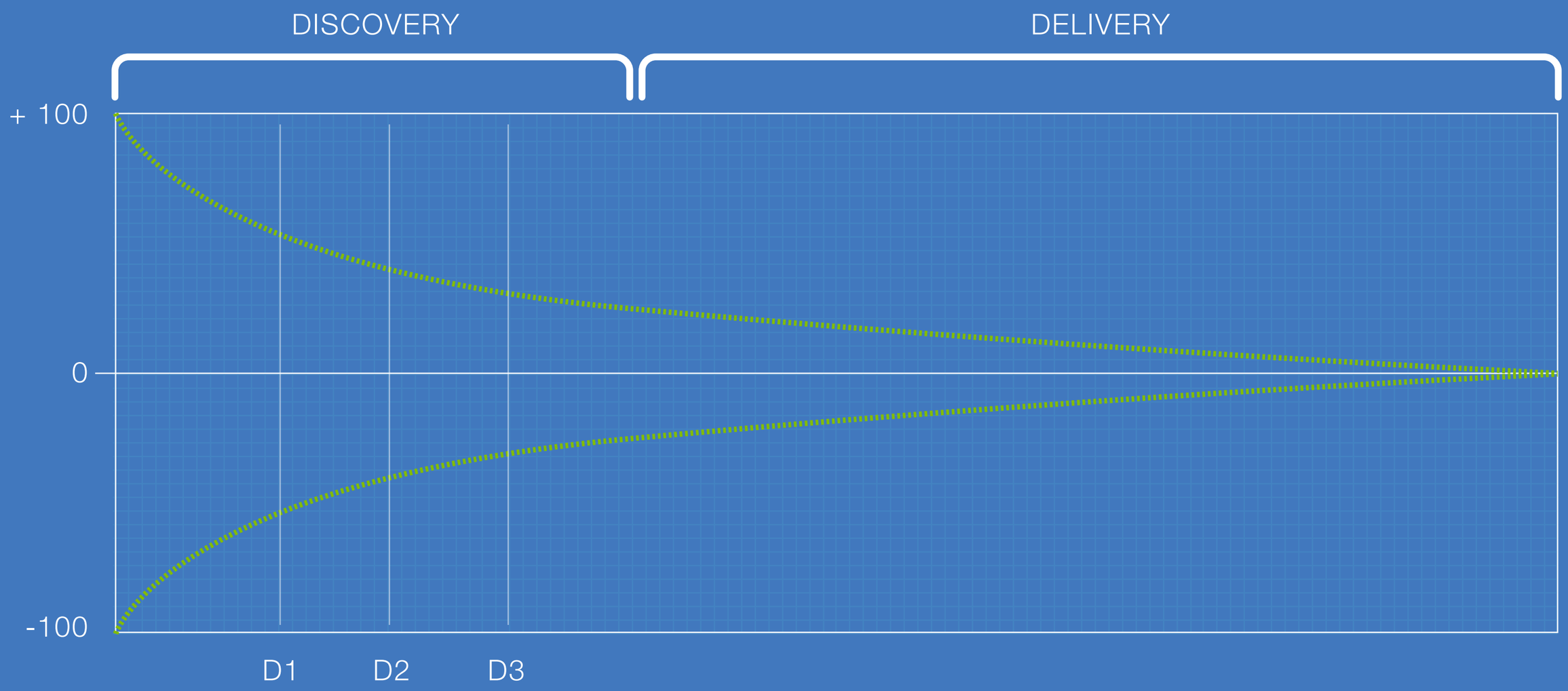
How to document the Discovery practice and make a business proposal?




## Keep it simple!

Use pictures and powerpoint to document the outputs.

# Cone of Uncertainty



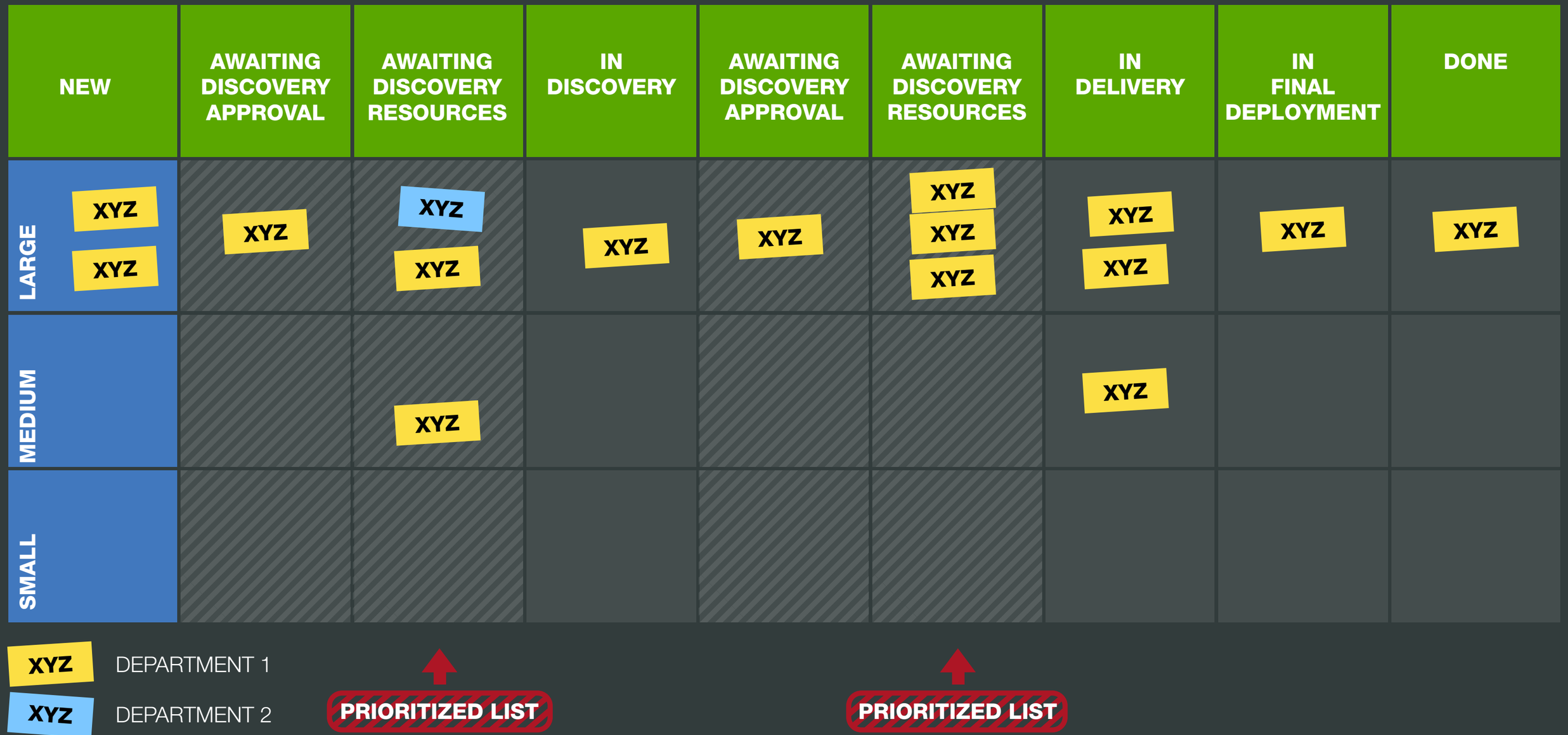
agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# The Portfolio Wall



# Setting up for Success

## 1 THE WORK

Approved project scope

## 2 RESOURCING

The team

The PM & IM

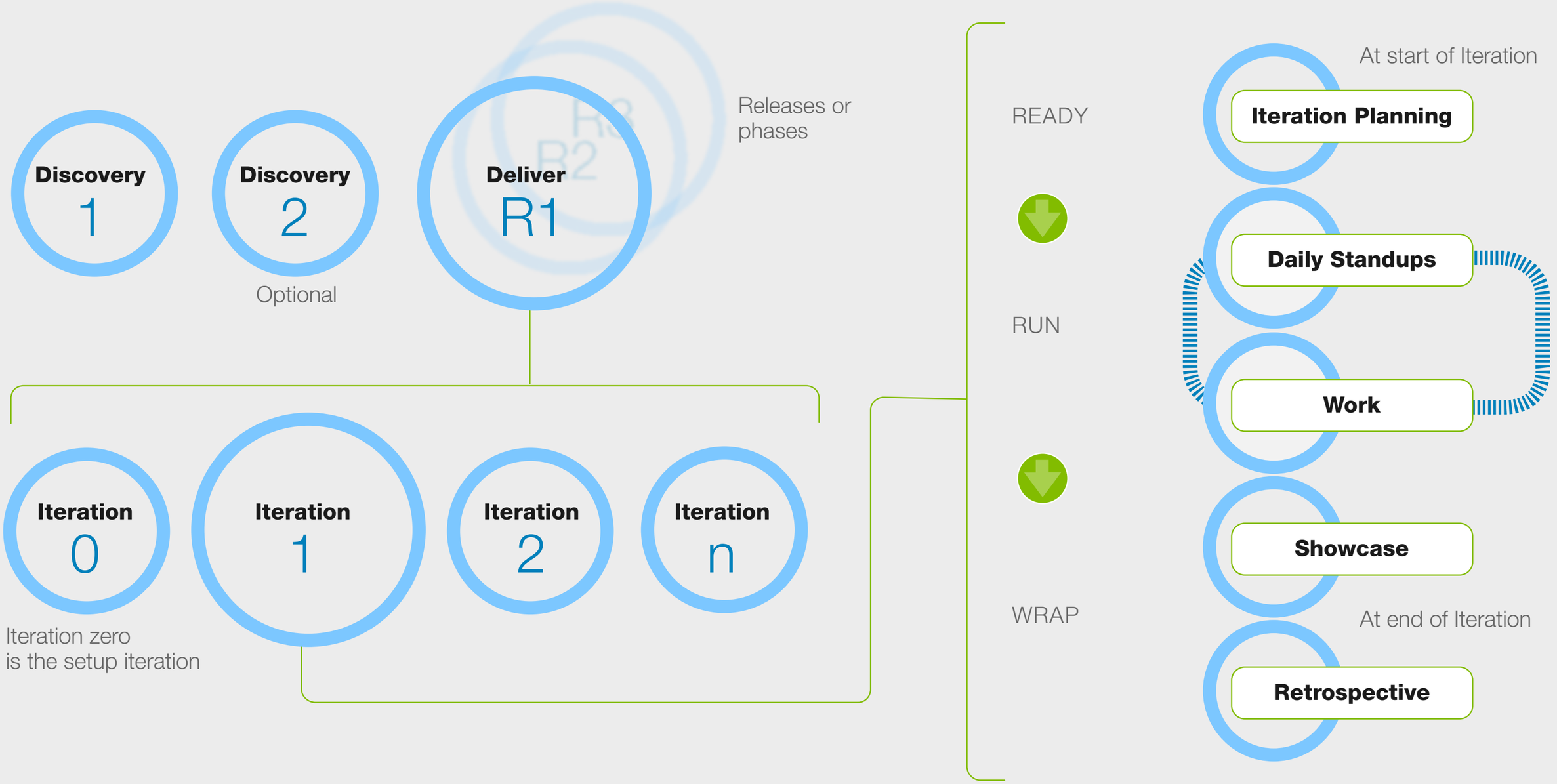
The Customer – Product

Owner

$$ - Budgets

Facility – space

# The Lifecycle of Delivery



# Scrum

**Inputs** from customers, team, managers, execs

**Product Owner**

**Product Backlog**

A Prioritized List

**of what is required: features, bugs to fix**

**Team**

Sprint Planning Meeting

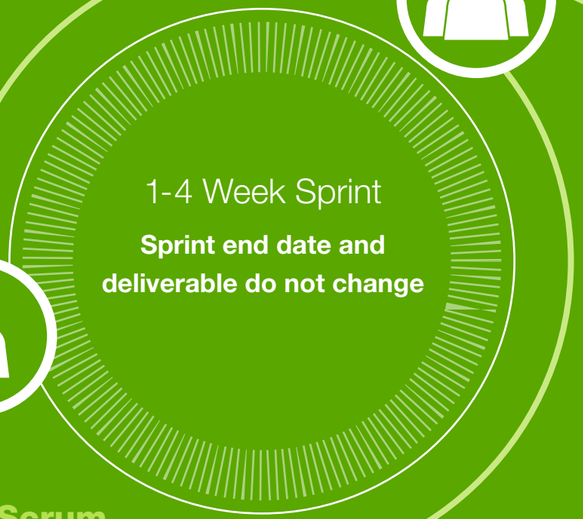
**The team commits to as much high priority backlog as can be completed by the end of the sprint**

**Sprint Backlog**

Task Breakout

**Daily Standup Meeting**

15-30 Minutes




**Scrum Master**

**Sprint Review**

**Finished Product**

Product Increment

**Sprint Retrospective**

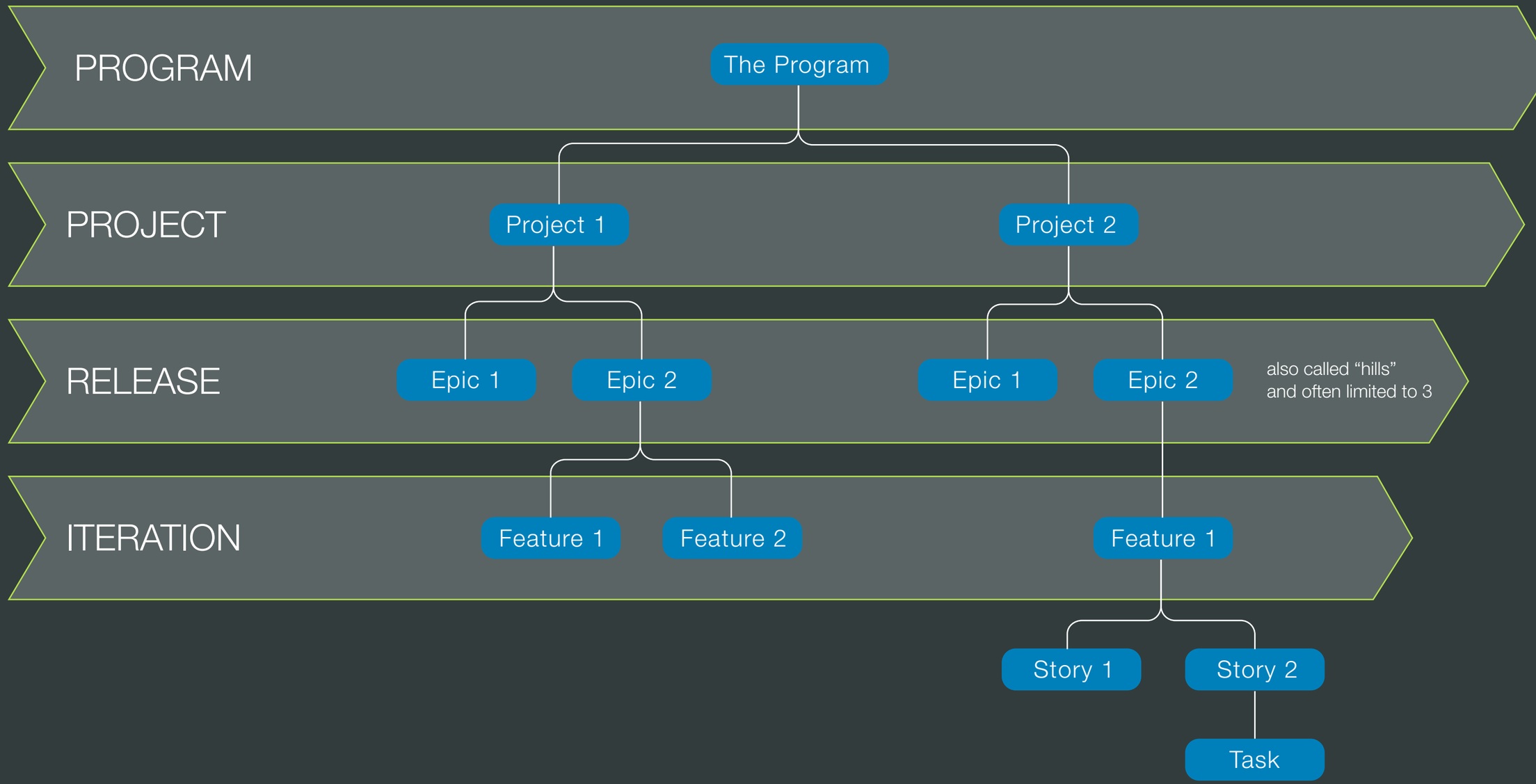
agileACADEMY

# Agile Program Fundamentals

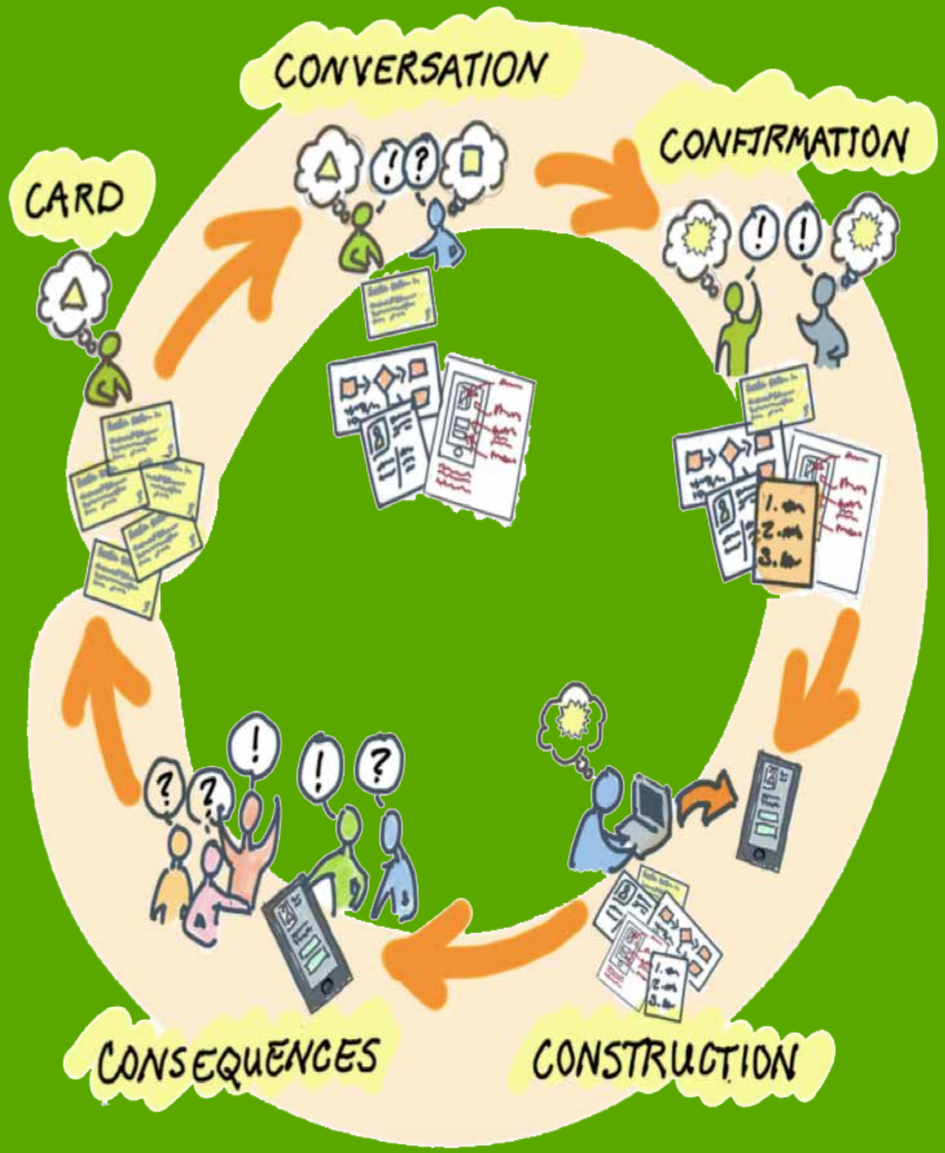
Course IAA2

## Course Topics

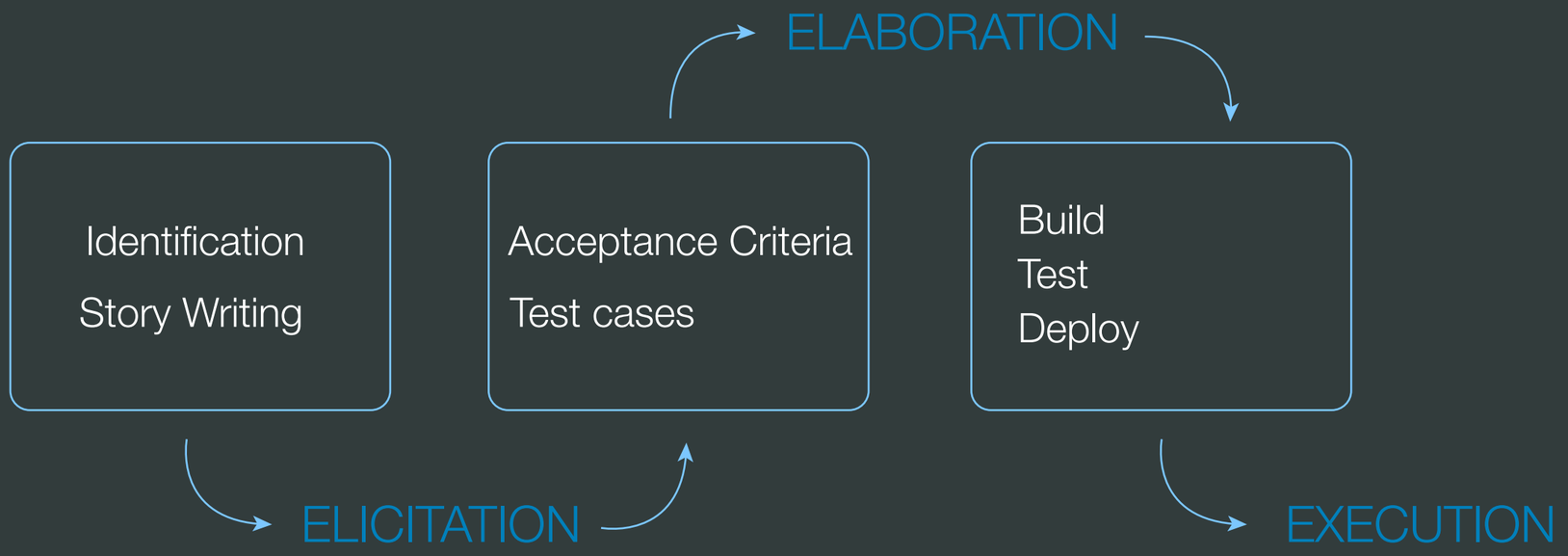
# Agile Story Hierarchy



# The Story starts with a card, then a conversation...



# The Life of a Story




**Prioritization** | **Estimation** | **Planning**

# Story Identification

**Personas**

**Process**

**Outcomes**

Feature breakdown

Design Thinking Practices

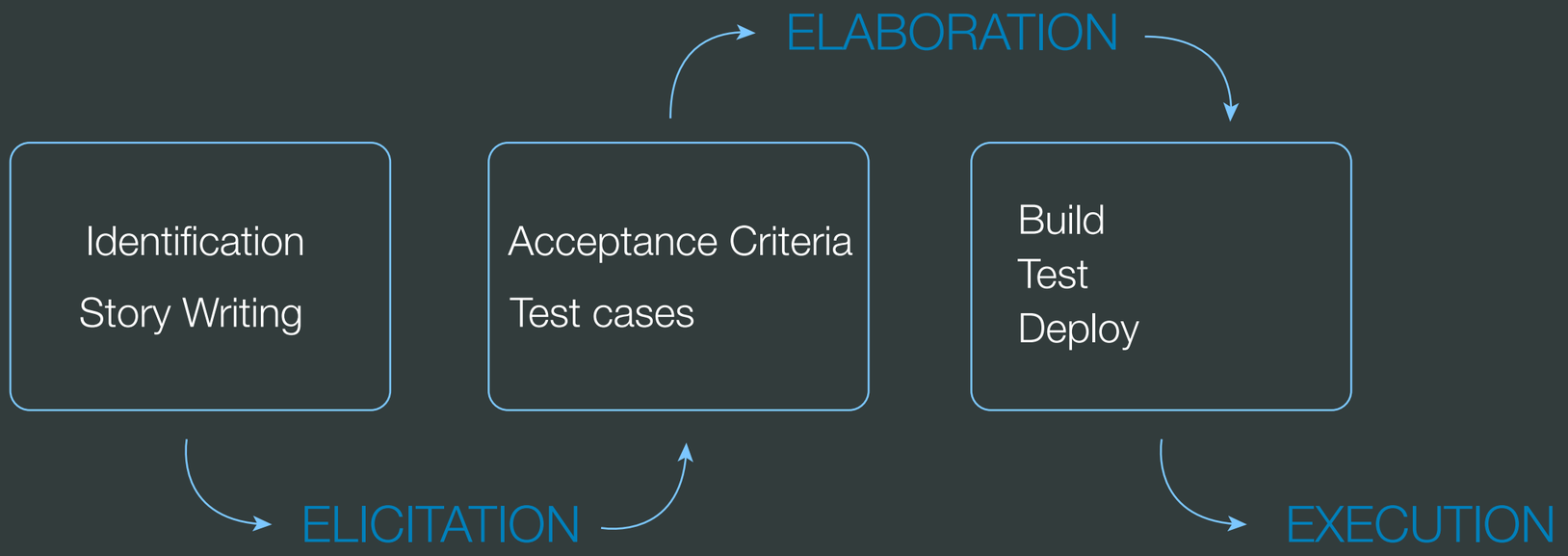
Business Canvas

Value Stream Mapping (as-is and to-be)

Brainstorming

User/customer interviews

# The Life of a Story



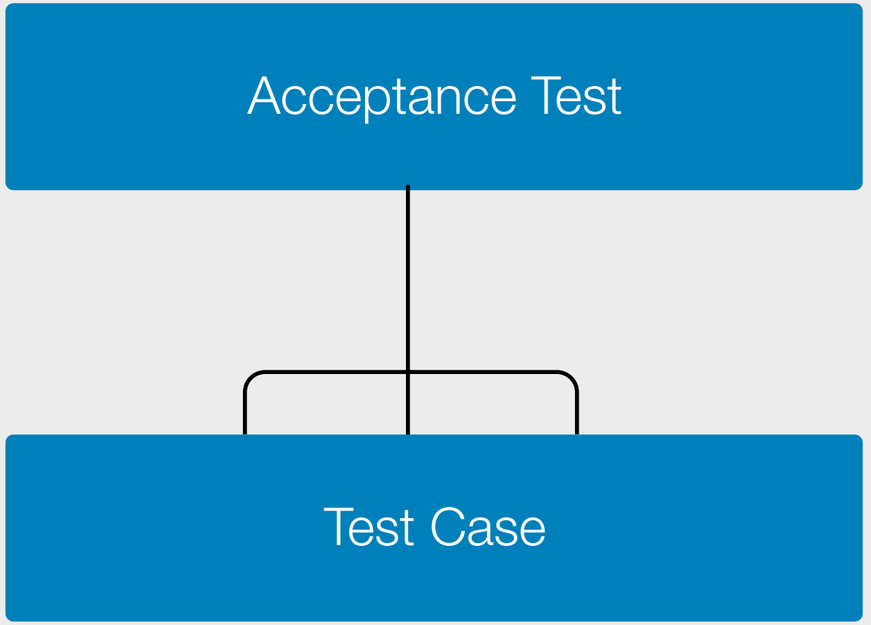

**Prioritization** | **Estimation** | **Planning**

# Story Elaboration

| ALWAYS | SOMETIMES | LESS OFTEN |
| --- | --- | --- |
| ID | Narrative | Technical Design |
| Title | Lo-fi Prototype | Data Model |
| Acceptance Criteria | Assumptions | Link to High Level Scenarios |
| Relative Size Estimate | Constraints | GUI Design |

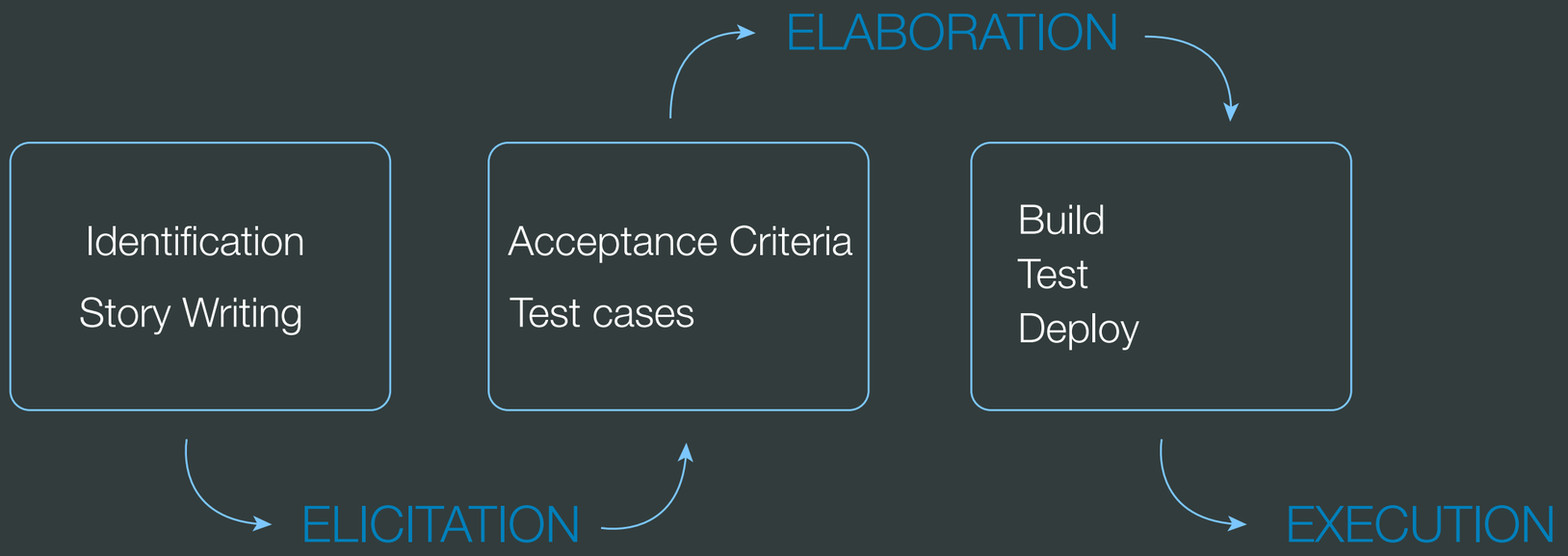
**Collaborate to Elaborate**

# ATDD - TDD




**Collaborate to Elaborate**

# The Life of a Story




**Prioritization** | **Estimation** | **Planning**

**Who**
**What**
**Why**

**Personas**
**Process**
**Outcomes**

As a _____

I want _____

So that _____

agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# Iteration Zero

- Shared understanding of Agile- Team Training
- Shared understanding of the project
- Social contract
- Story Elaboration - Max 2 iterations ahead
- Architecture and Design
- Standards & guidelines
- Tools and environment provisioning
- Detailed Release Planning

How long should it take?

**“Just enough to start.”**

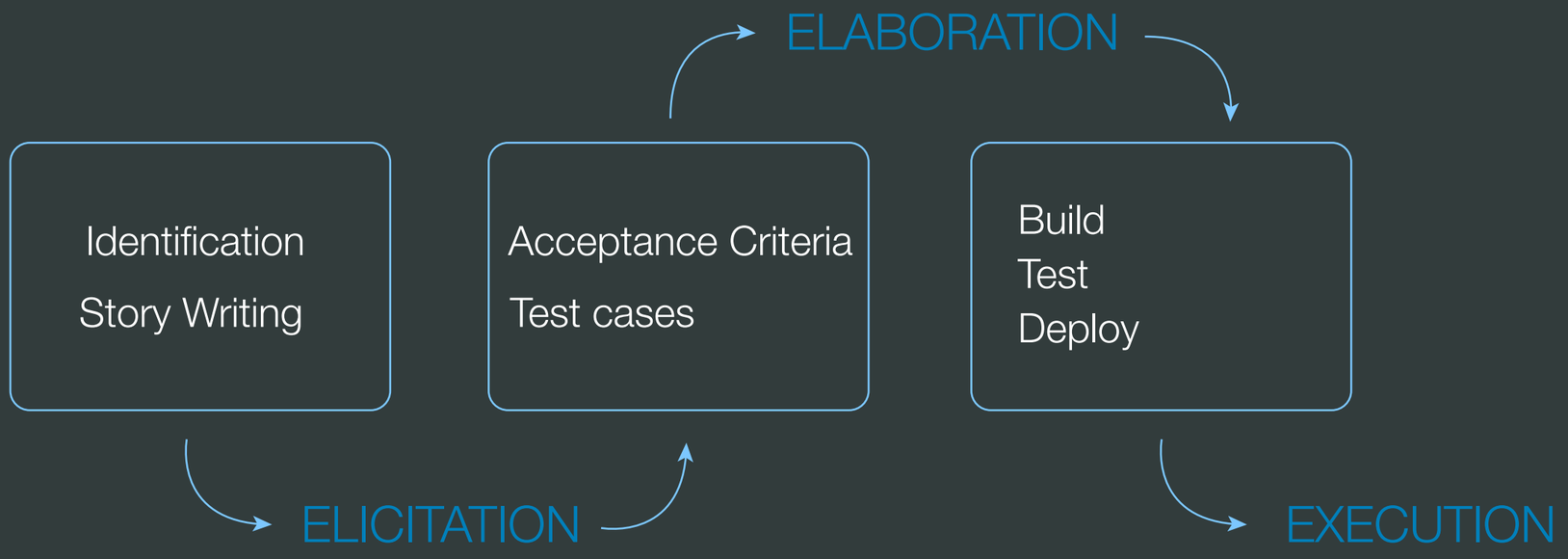
agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# The Life of a Story




**Prioritization** | **Estimation** | **Planning**

***“As the team we need a release plan so that we can set ourselves targets & milestones and socialise delivery projections with our stakeholders”***

- Prioritise features and stories
- Estimate features and stories
- Estimate velocity
- Fill the iteration buckets
- Cater for contingency
- Set up the release wall

## Iteration 1

**5th** → **16th**
November

- Prioritise features and stories
- Estimate features and stories
- Estimate velocity
- Fill the iteration buckets
- Cater for contingency
- Set up the release wall

**Total = x**

## Iteration 2

**19th** → **30th**
November

- Prioritise features and stories
- Estimate features and stories
- Estimate velocity
- Fill the iteration buckets
- Cater for contingency
- Set up the release wall

**Total = x**

## Iteration 3

**3rd** → **14th**
December

- Prioritise features and stories
- Estimate features and stories
- Estimate velocity
- Fill the iteration buckets
- Cater for contingency
- Set up the release wall

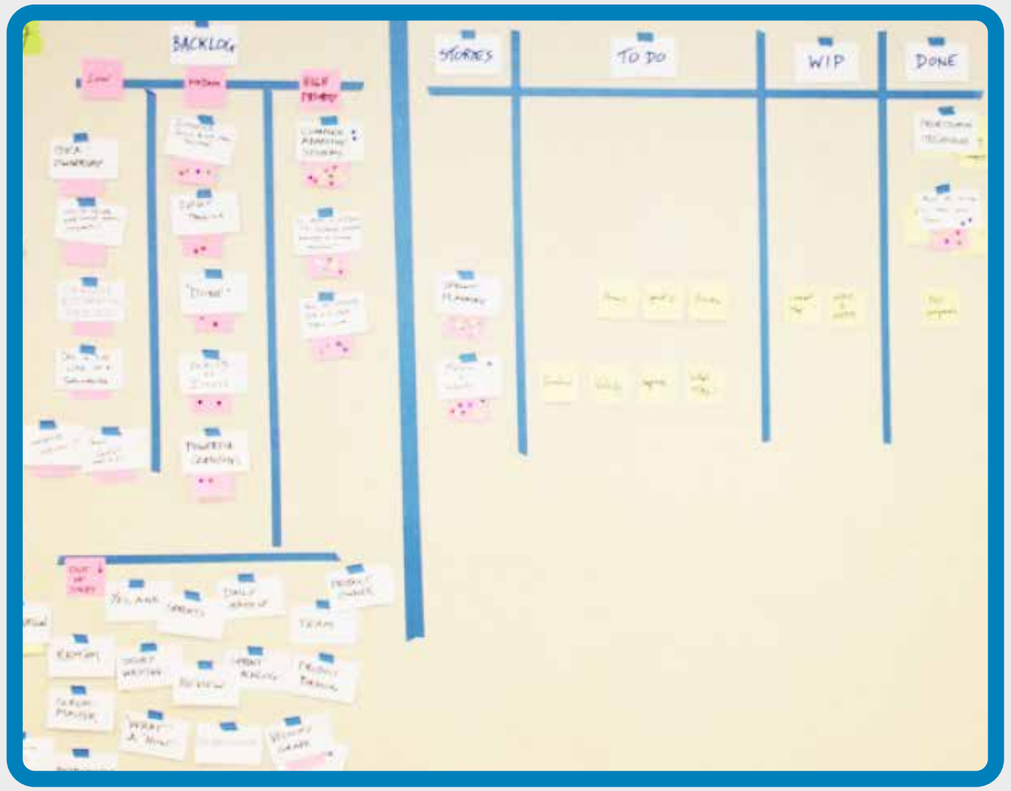
**Total = x**

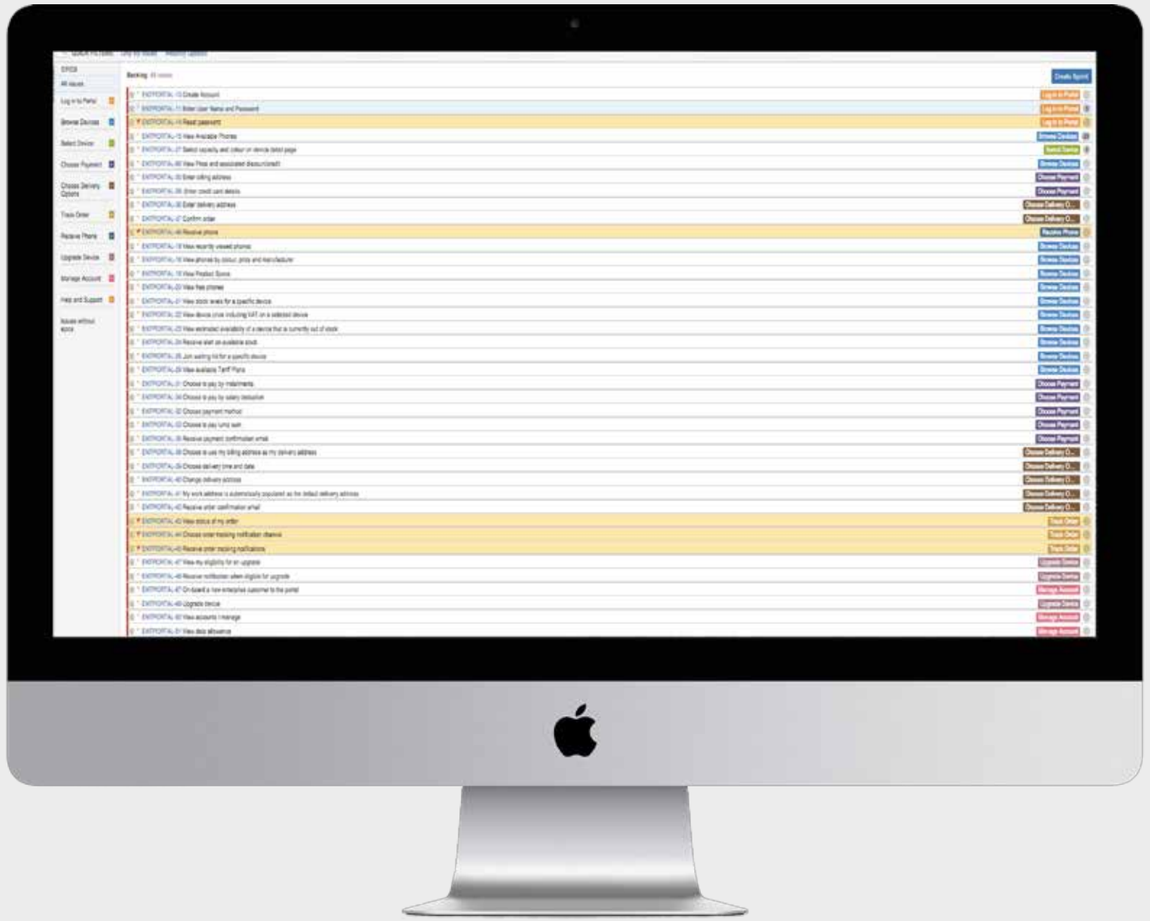
***“As the team we need an Iteration plan so that we can be clear on what to deliver in the next iteration and prepare for delivery”***

- Re-estimate hangover stories for left over
- Estimate velocity
- Fill the iteration first with hangover stories
- Pull from next iteration backlog till bucket is full
- Update the Release plan based on changes
- Update walls and burn-up charts. Assign work

## Product Backlog

On the wall and in a tool

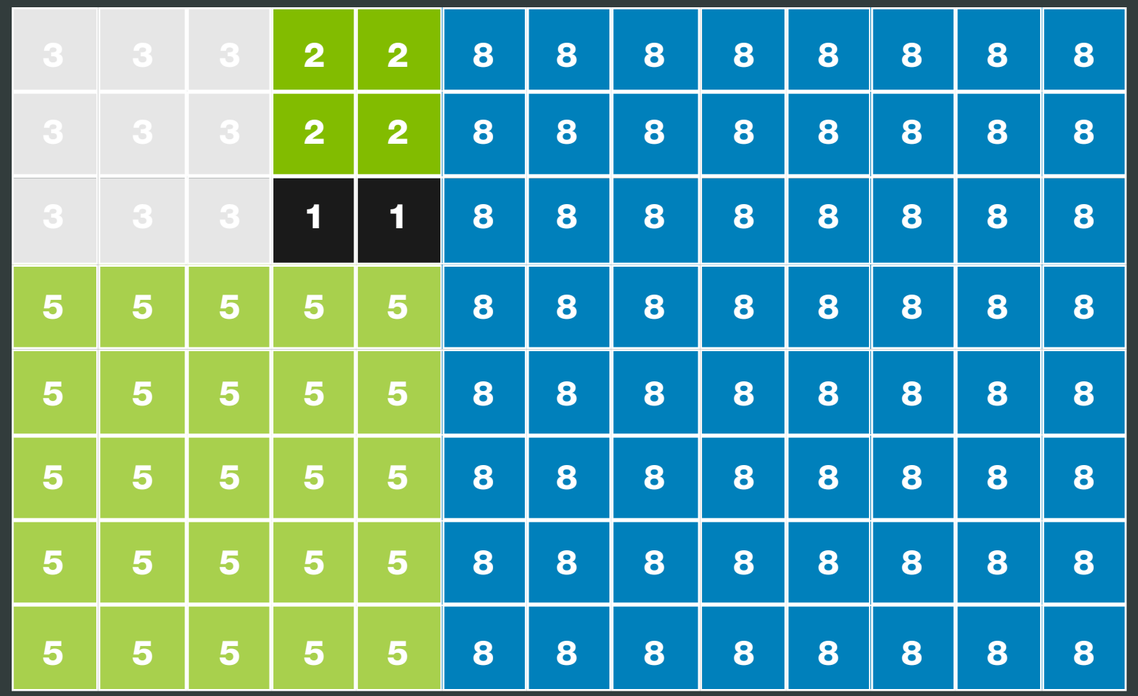


**Relative estimation using poker play and points**

Let’s go a little deeper.

# The Fibonacci Sequence



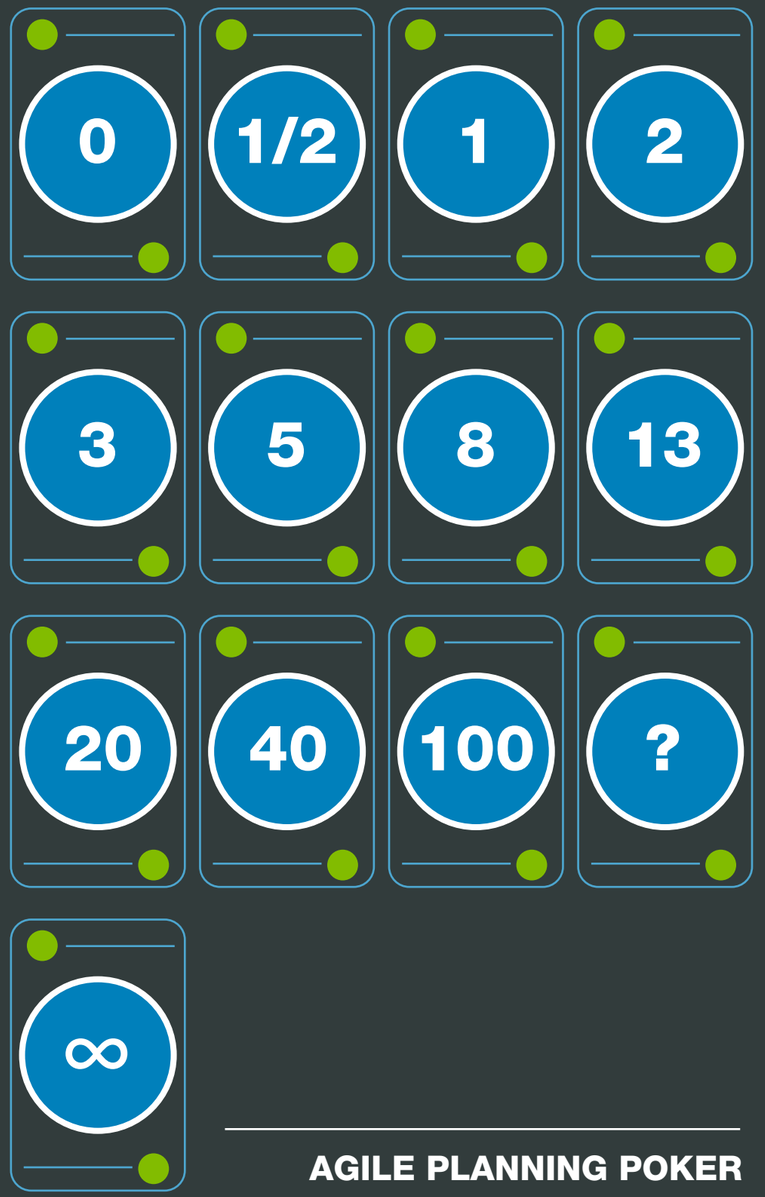

0, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144….

The first two Fibonacci numbers are 0 and 1; each subsequent number is the sum of the previous two

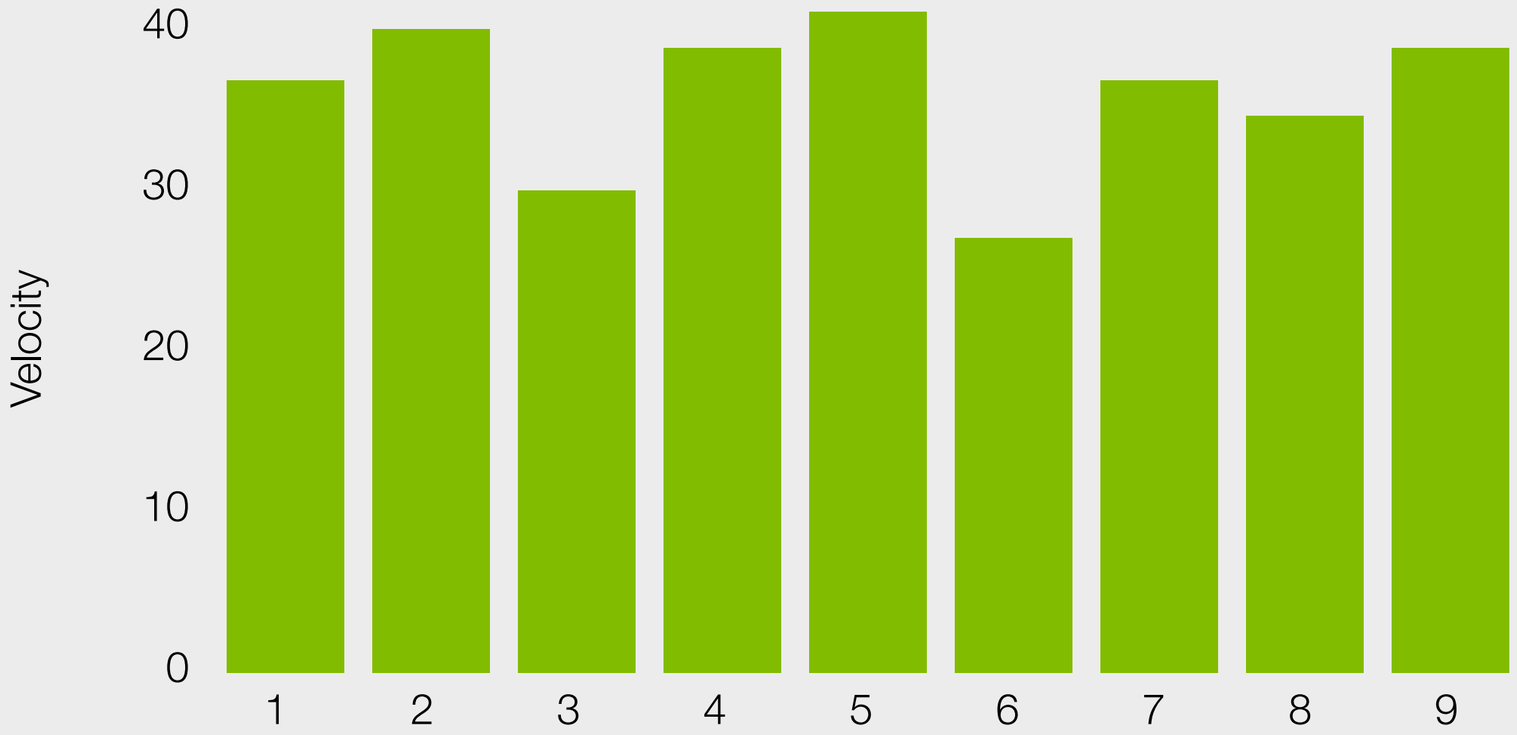
# Our Pseudo Fibonacci Sequence

**In Agile/Scrum, many use a pseudo Fibonacci sequence to assign story points to stories**

Large estimates are less accurate; larger gaps between larger estimates help us avoid splitting hairs unnecessarily



# Velocity



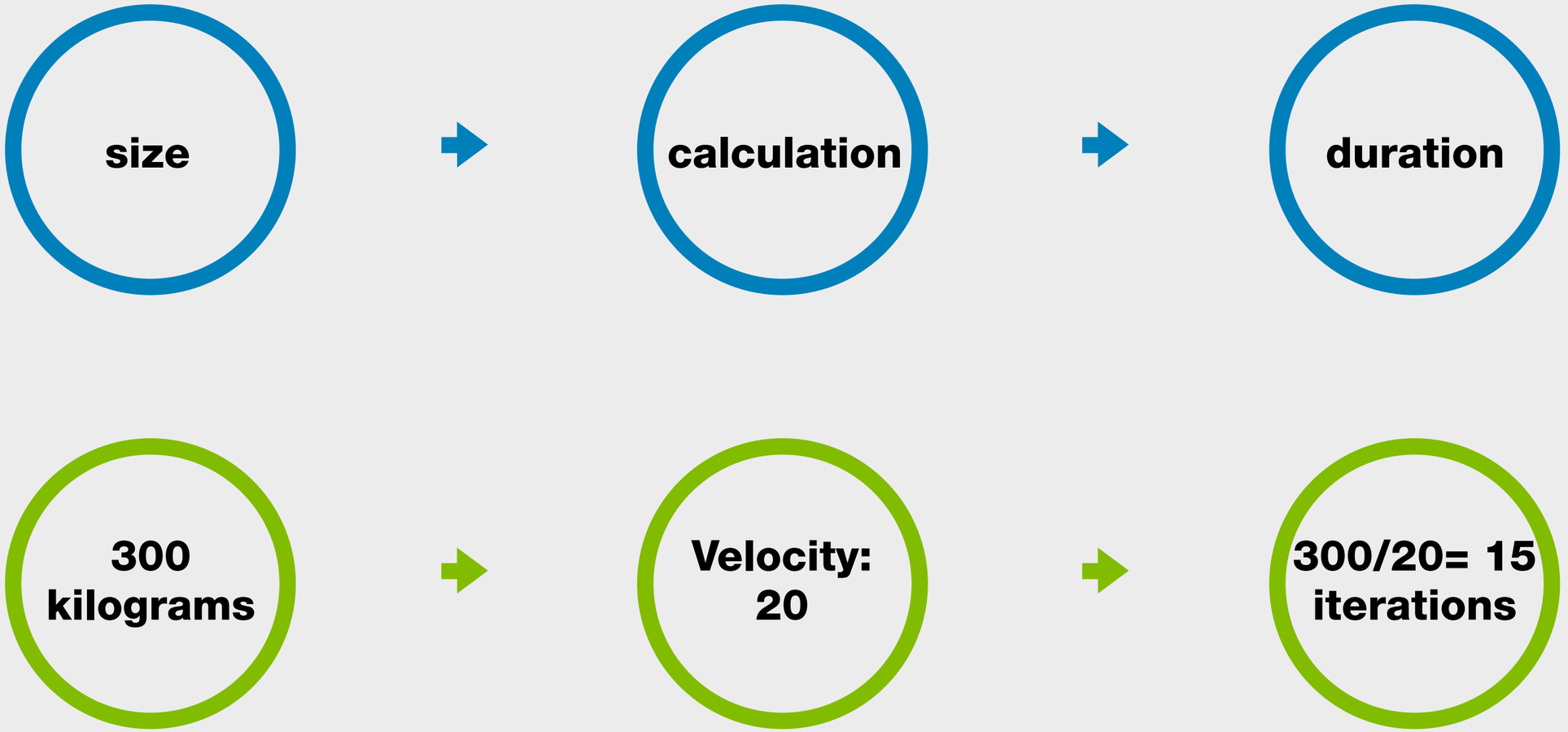

**A useful long-term measure of the amount of work completed per iteration.**

Most useful over at least a handful of iterations.

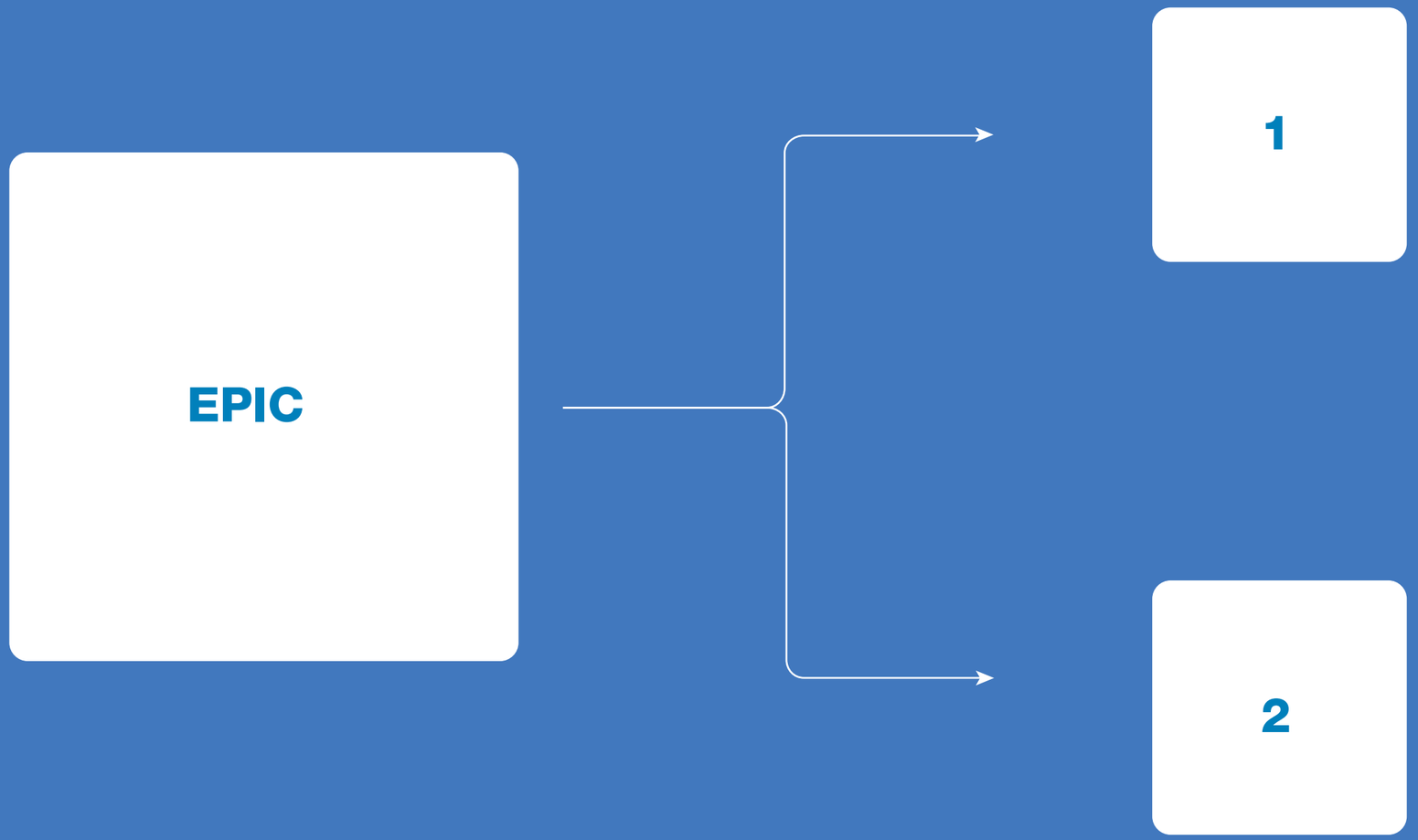
---

Velocity is measured in the units you use to estimate product backlog items.

# Estimate Size; Derive Duration

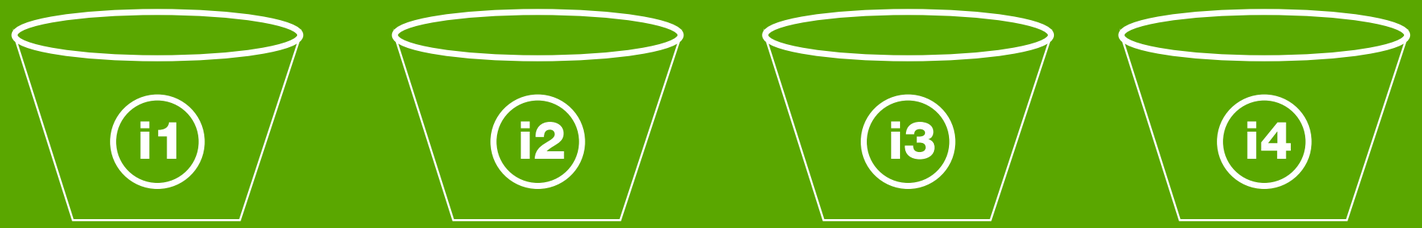


# Story Splitting



## Prioritized and estimated Story List

- A
- B
- C
- D
- E
- F

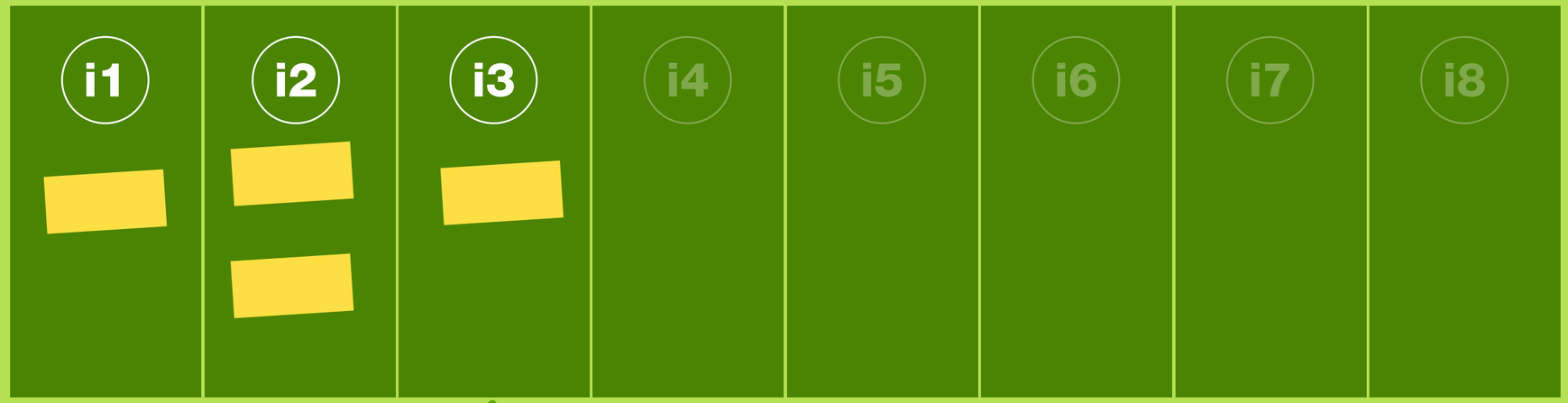


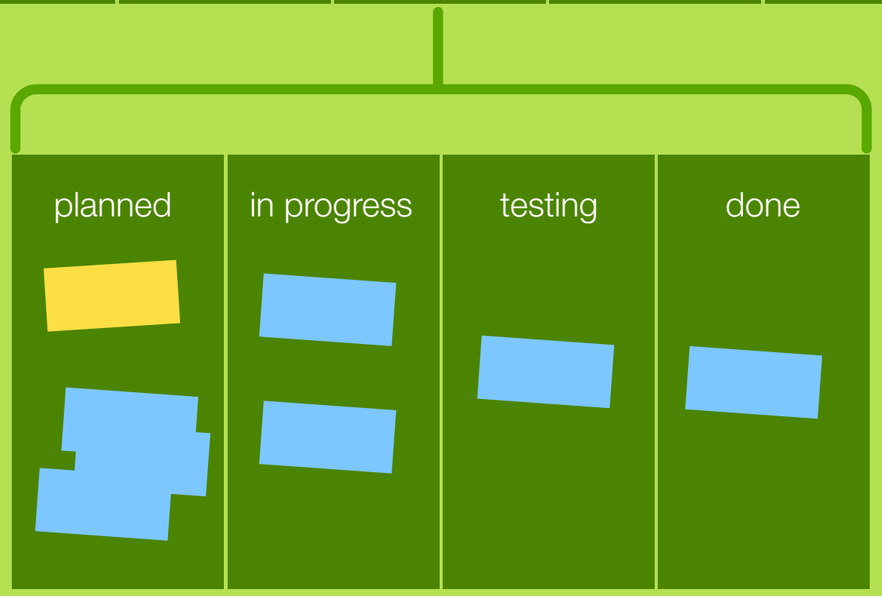

**Iteration Buckets**

Velocity = bucket size

# Release Plan

## Release Wall



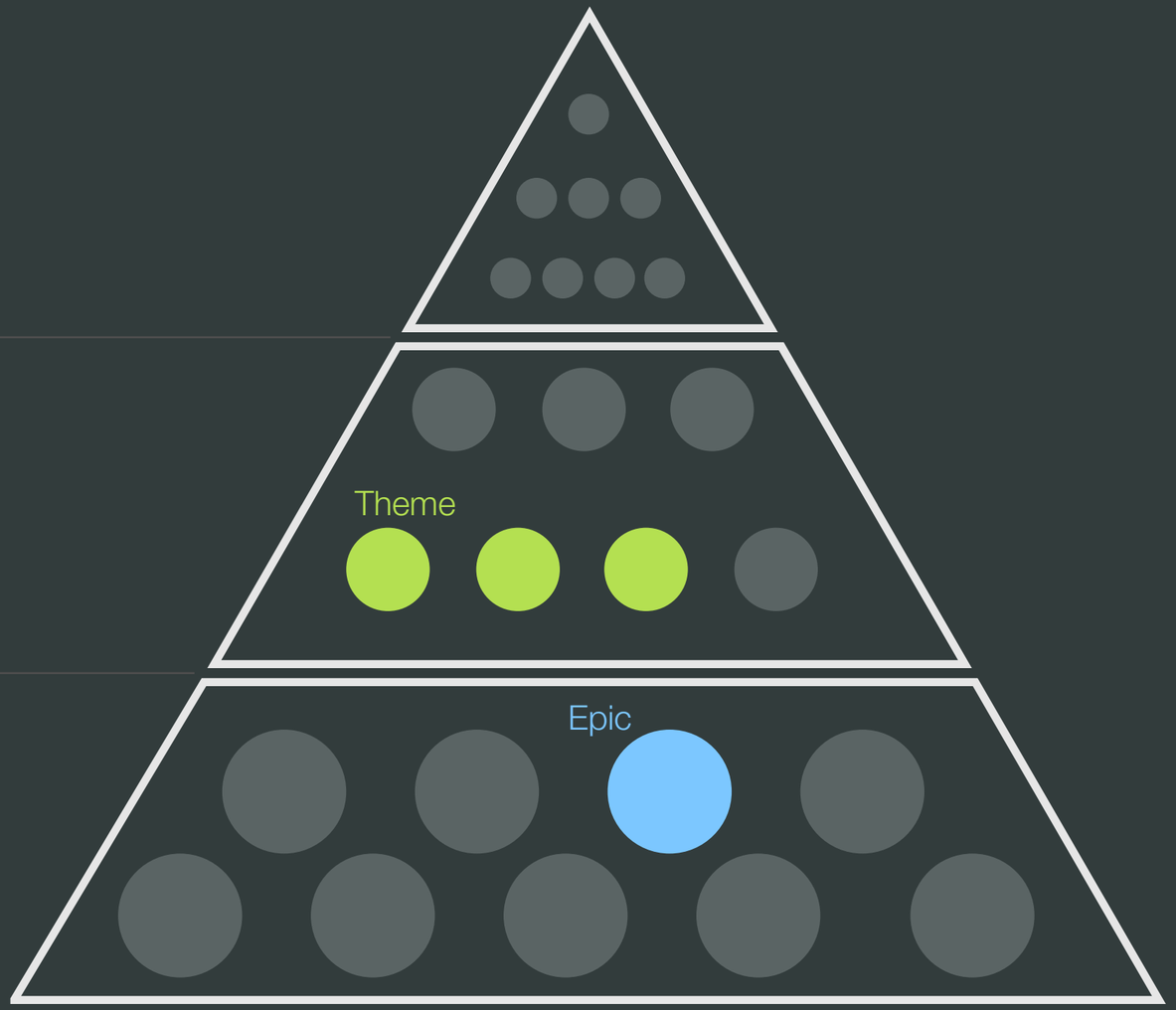

## Iteration Wall

# The Product Backlog Iceberg

Sized for a Sprint

Release

Future Release




Priority
Continuous Refinement

**Theme:** a theme is a collection of related backlog items

**Epic:** an Epic is a large backlog item

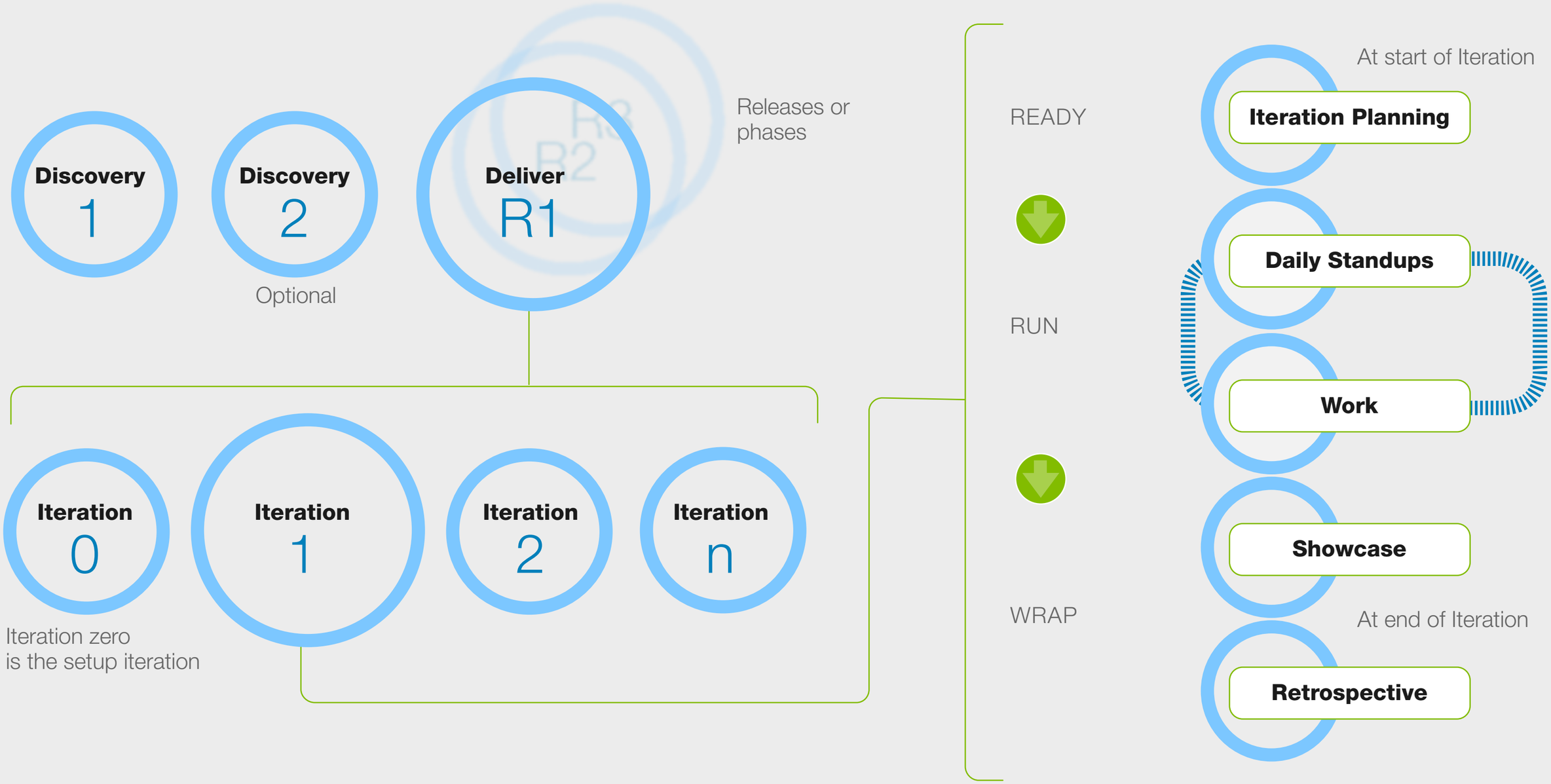
agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

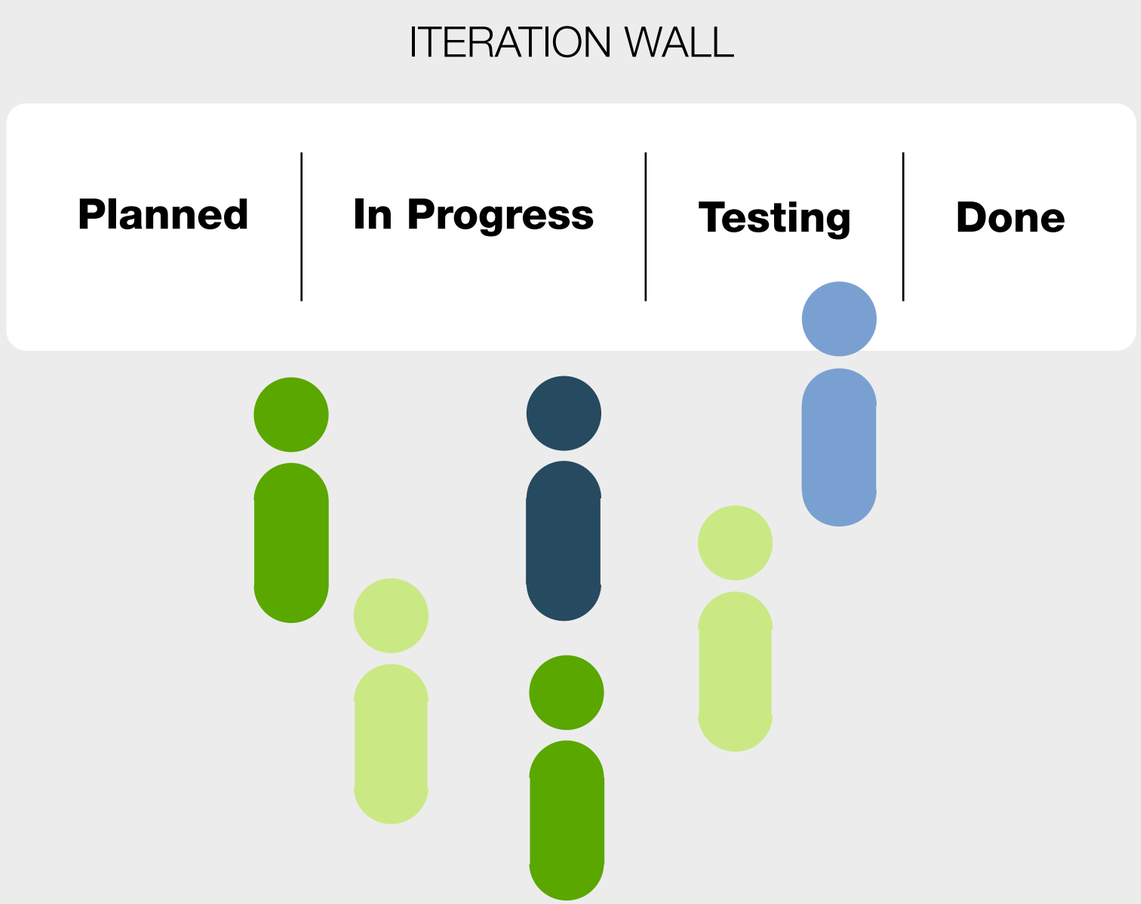
# The Lifecycle of Delivery



# JULY

| SUNDAY | MONDAY | TUESDAY | WEDNESDAY | THURSDAY | FRIDAY | SATURDAY |
|---|---|---|---|---|---|---|
| **10** | **11** | **12** | **13**<br>**Iteration Planning = 3.5 hrs (5%)**<br>**Work = 3.5 hrs (5%)** | **14**<br>Work = 7 hrs (10% | **15**<br>Work = 7 hrs (10%) | **16** |
| **17** | **18**<br>Work = 7 hrs (10%) | **19**<br>Work = 7 hrs (10%) | **20**<br>Pre- planning = 7 hrs (10%) | **21**<br>Work = 7 hrs (10%) | **22**<br>Work = 7 hrs (10%) | **23** |
| **24** | **25**<br>Work = 7 hrs (10%) | **26**<br>**Work = 3.5 hrs (5%)**<br>**Showcase + Retrospective = 3.5 hrs ( 5%)** | **27** | **28** | **29** | **30** |

# Standups




- Be punctual
- Talk to the card
- What did they do yesterday
- What's planned and left to-do
- Blockers in reaching deadline
- Make notes to discuss offline
- Keep it short and to the point
- Take longer discussions offline
- Make a note of who does not have work
- Move the card only after talking to it
- Pull a new card if needed

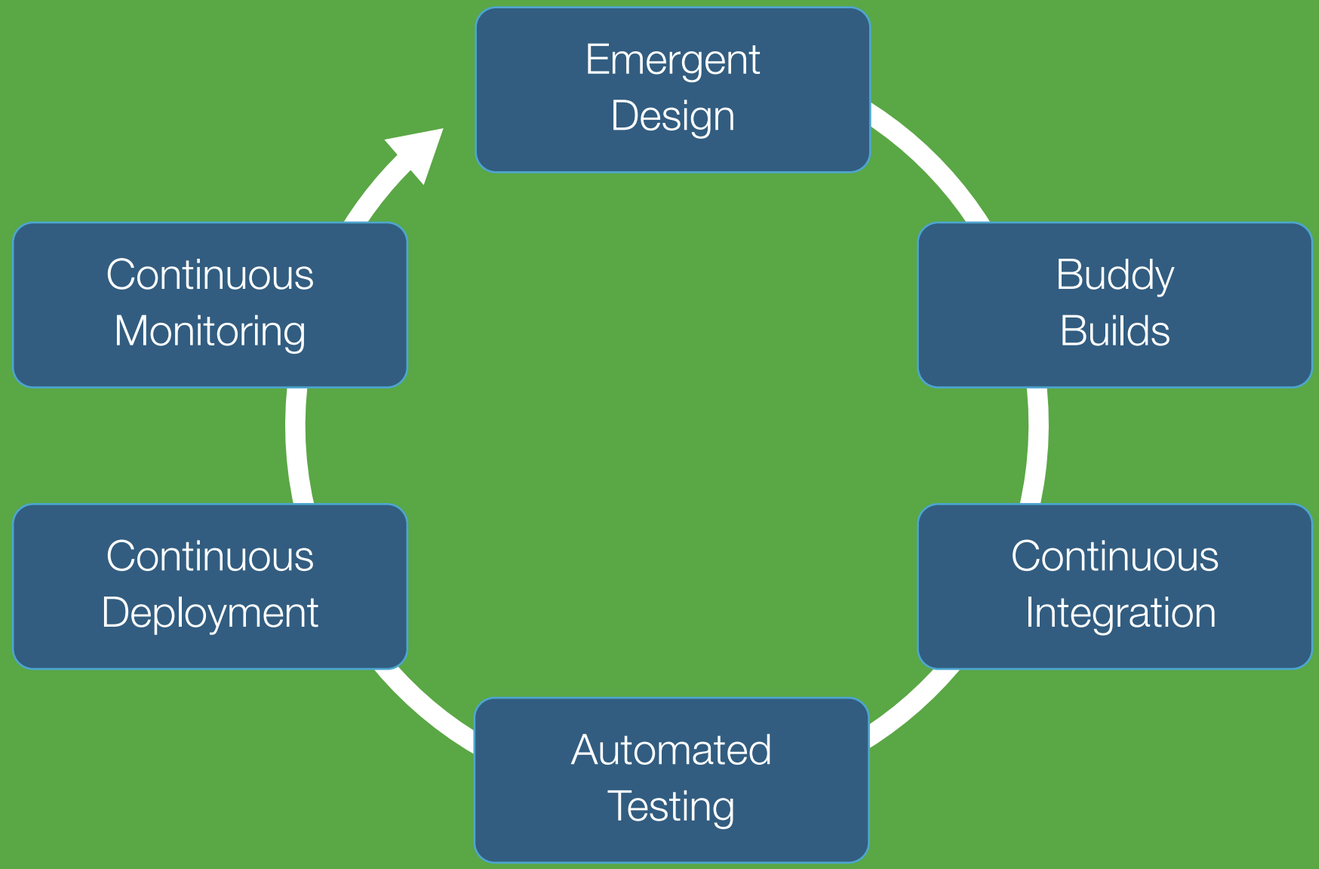
- Never embarrass anyone up in public
- Have the hard conversation in private

- Update the virtual wall later

# Hangouts

- 1:1
- Deep dives
- Take time to understand the real issues
- Listen
- Socialise
- Praise
- Talk to stakeholders
- Get more than one opinion
- Move boulders
- Carry water
- Work closely with the PM and PO

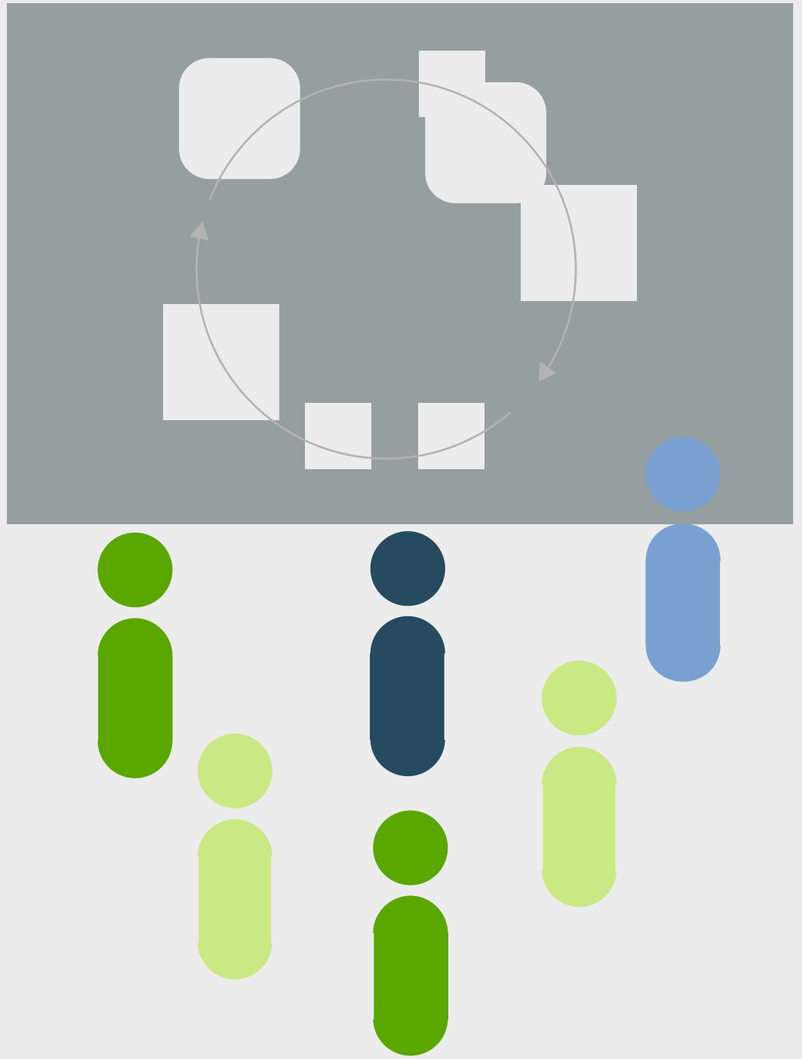
# Technical Practices- Dev-Ops



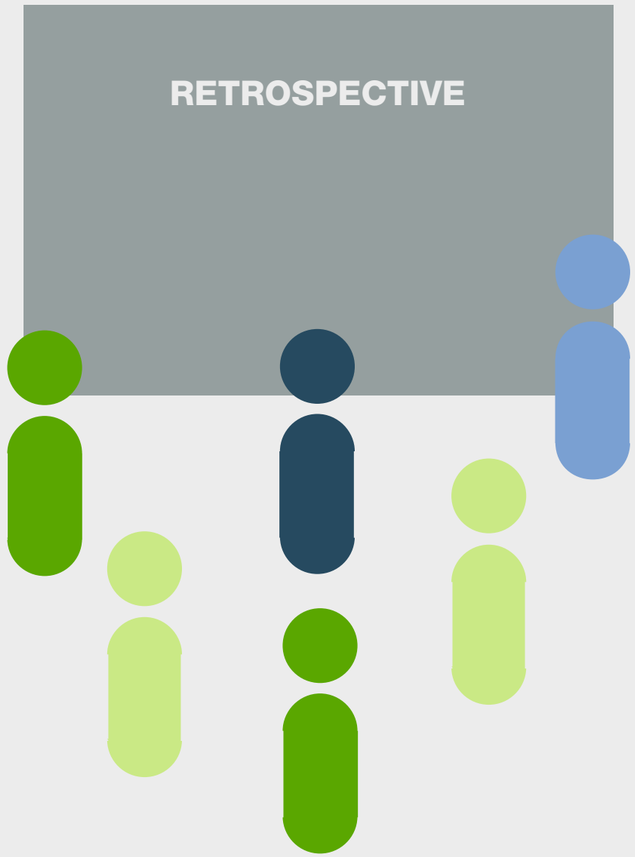

Configuration Management

Data Management

# Showcases



# Retrospective




- Working / not working / puzzles
- Group and summarise
- Vote for top 3

- Root cause analysis
  - 5 why’s

- Pick top 2 – 3 actions for next iteration
- Write up cards for it and add to Release wall

**“A retrospective is a gathering of a community at regular intervals throughout a delivery to review the events and learn from the experience.”**

Esther Derby, Agile Retrospectives

**“When the speed of failure slows, so does the speed of invention.”**

– Mike Steep,

SVP of global business operations at PARC

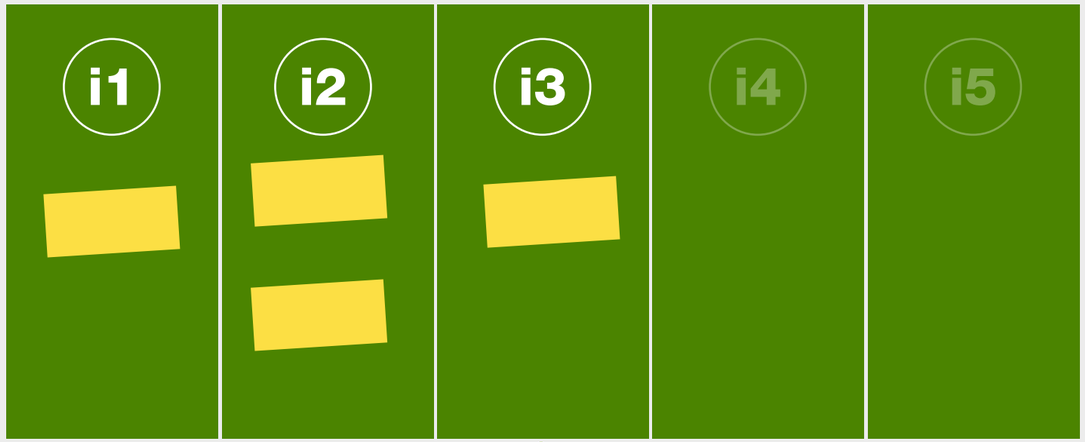
agileACADEMY

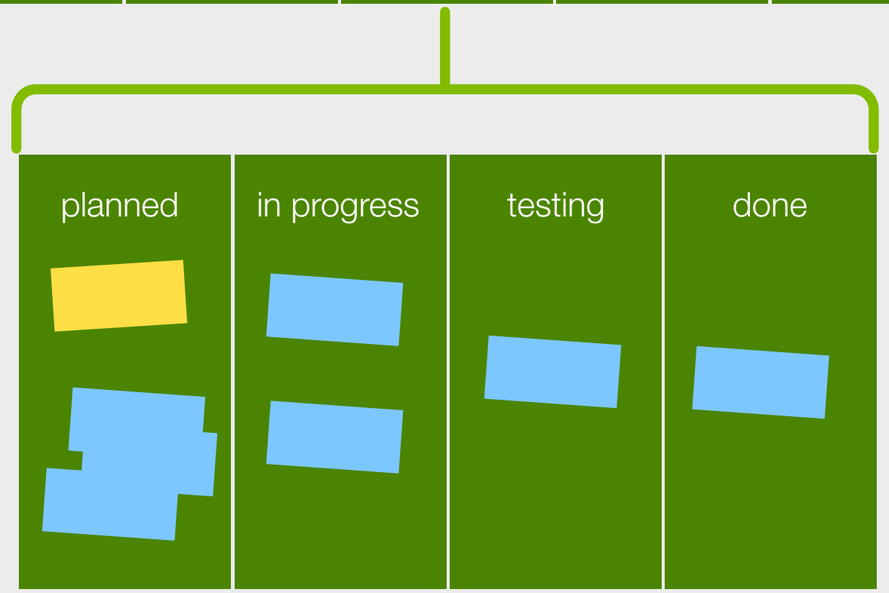
# Agile Program Fundamentals

Course IAA2

## Course Topics

Release Wall



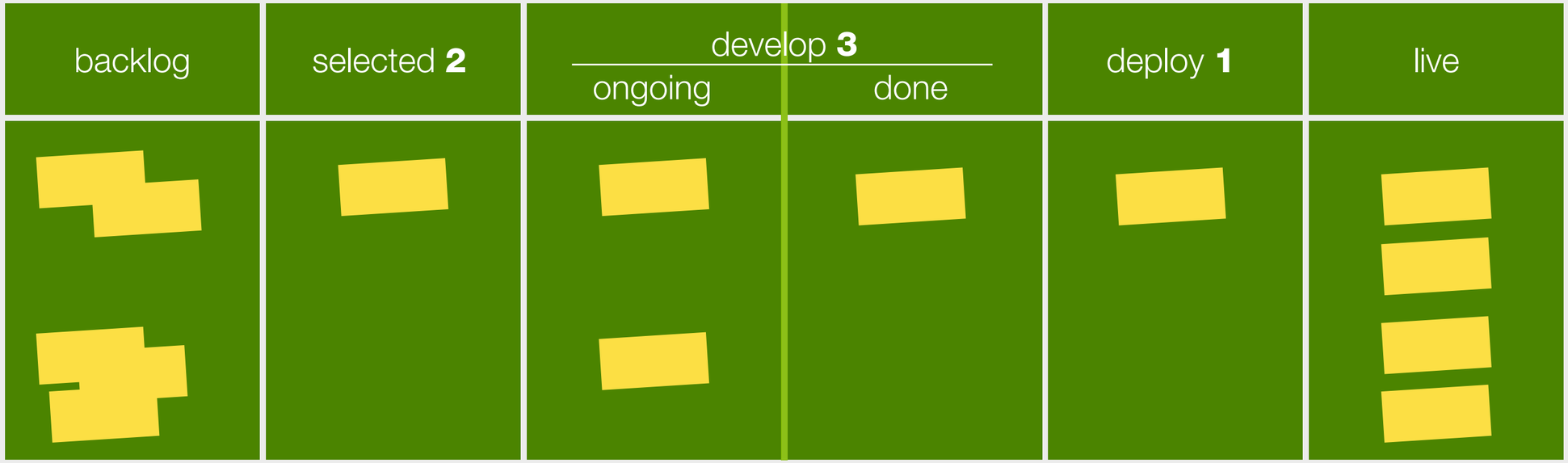

Iteration Wall

**Burn Up**

**Struggle Street**
(issues)

**Risks**

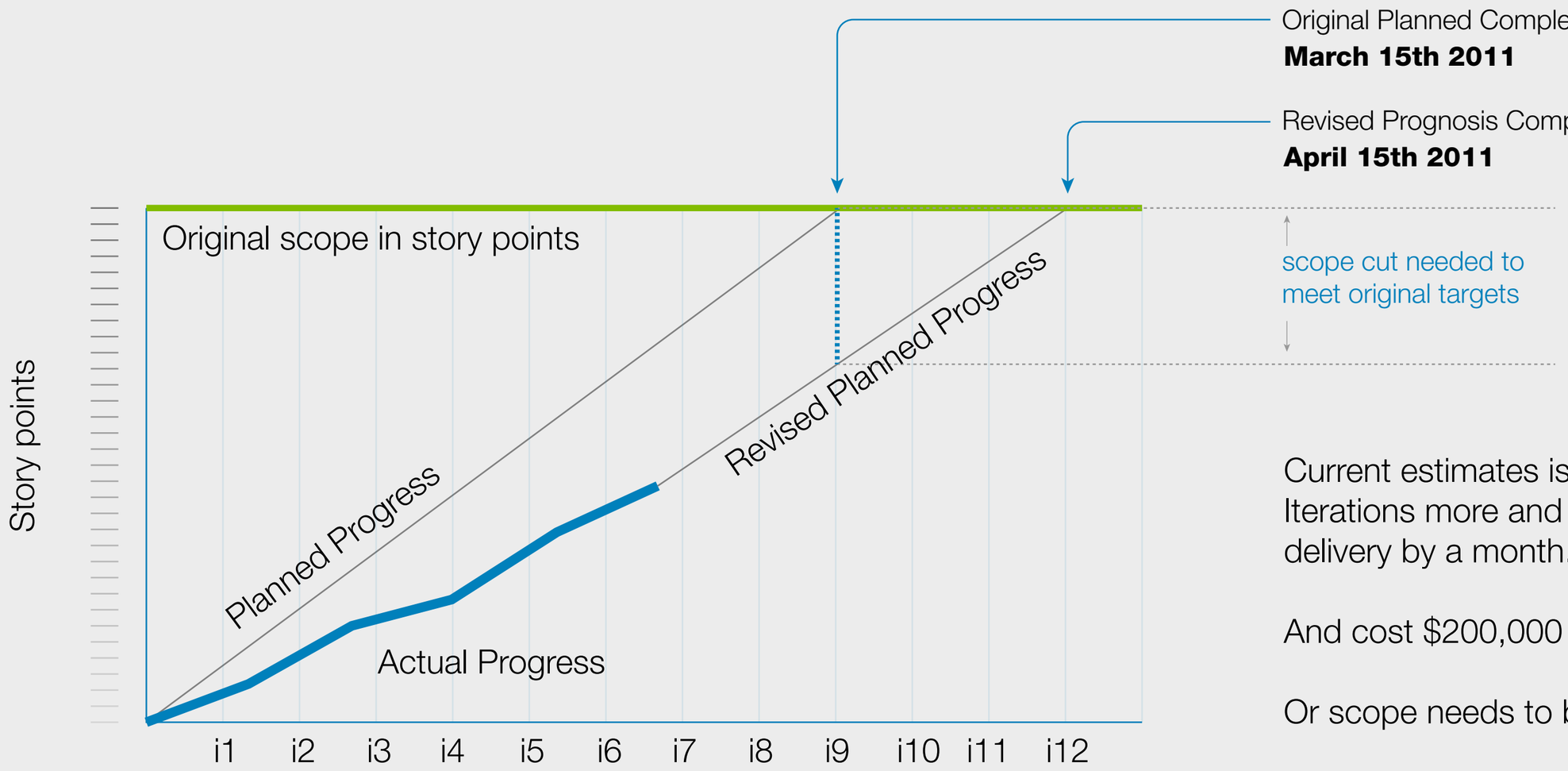
**Architecture Wall**




No iterations.... no velocity. Just continuous

**FLOW**

# Burnup Charts

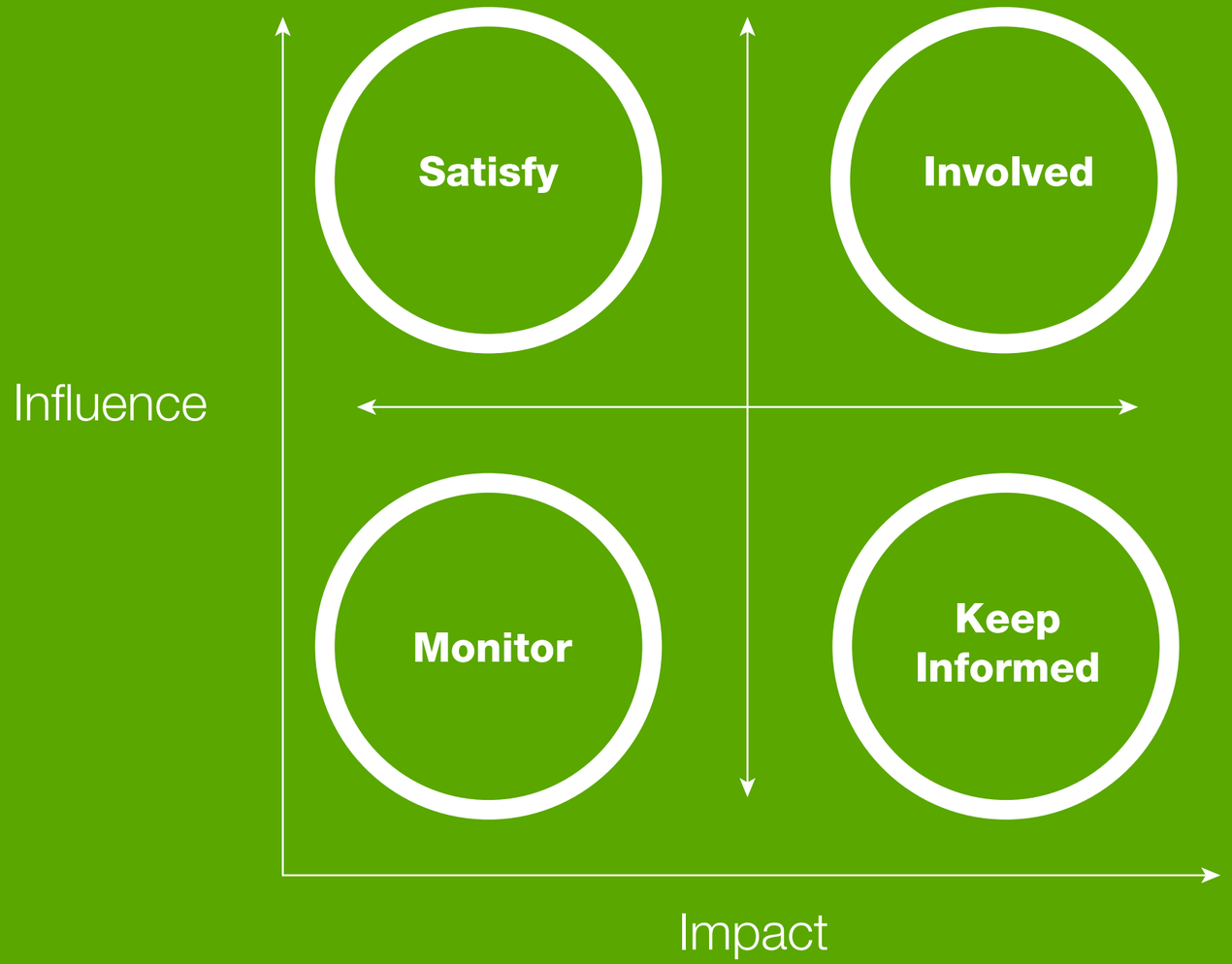



Current estimates is that we need two Iterations more and this will delay delivery by a month.

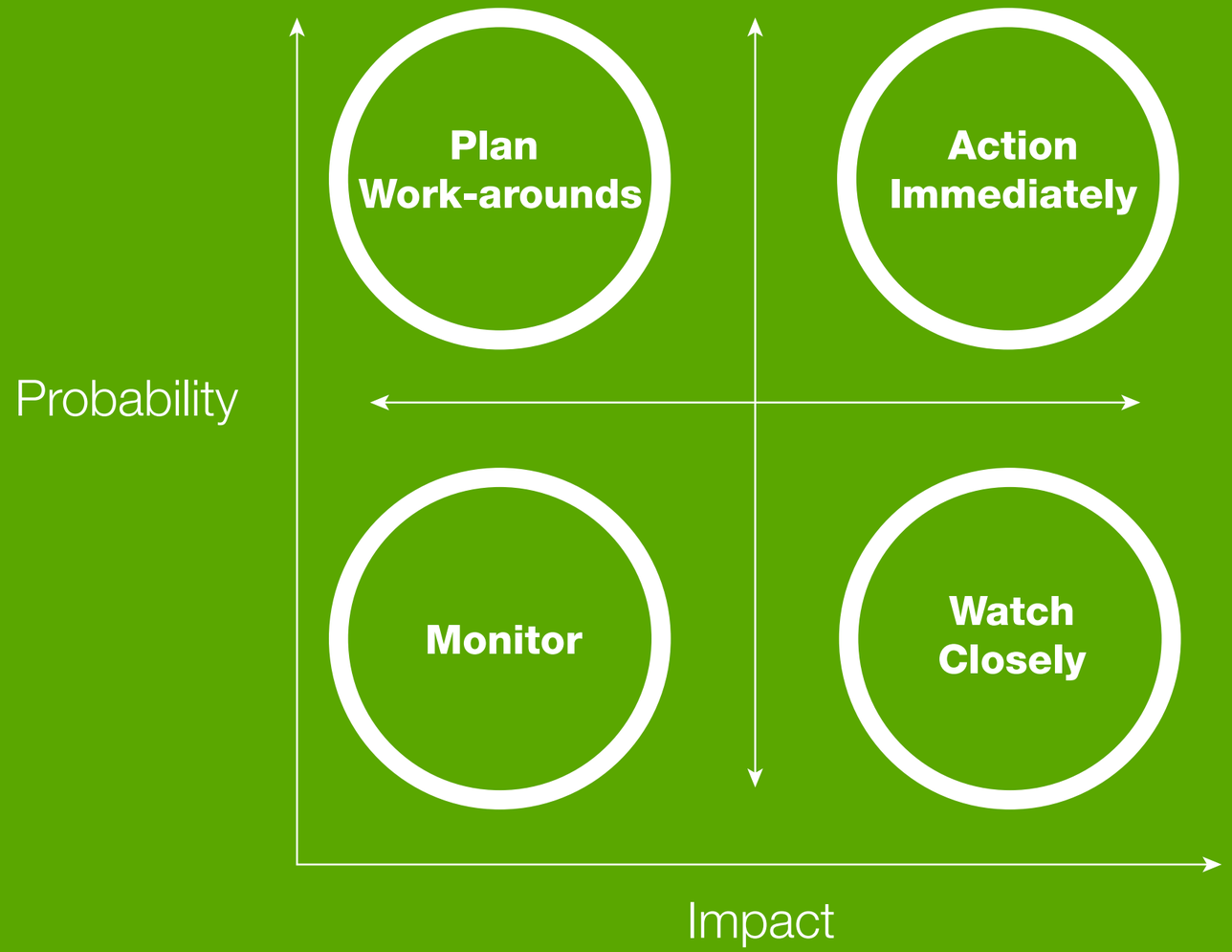
And cost $200,000 more.

Or scope needs to be cut

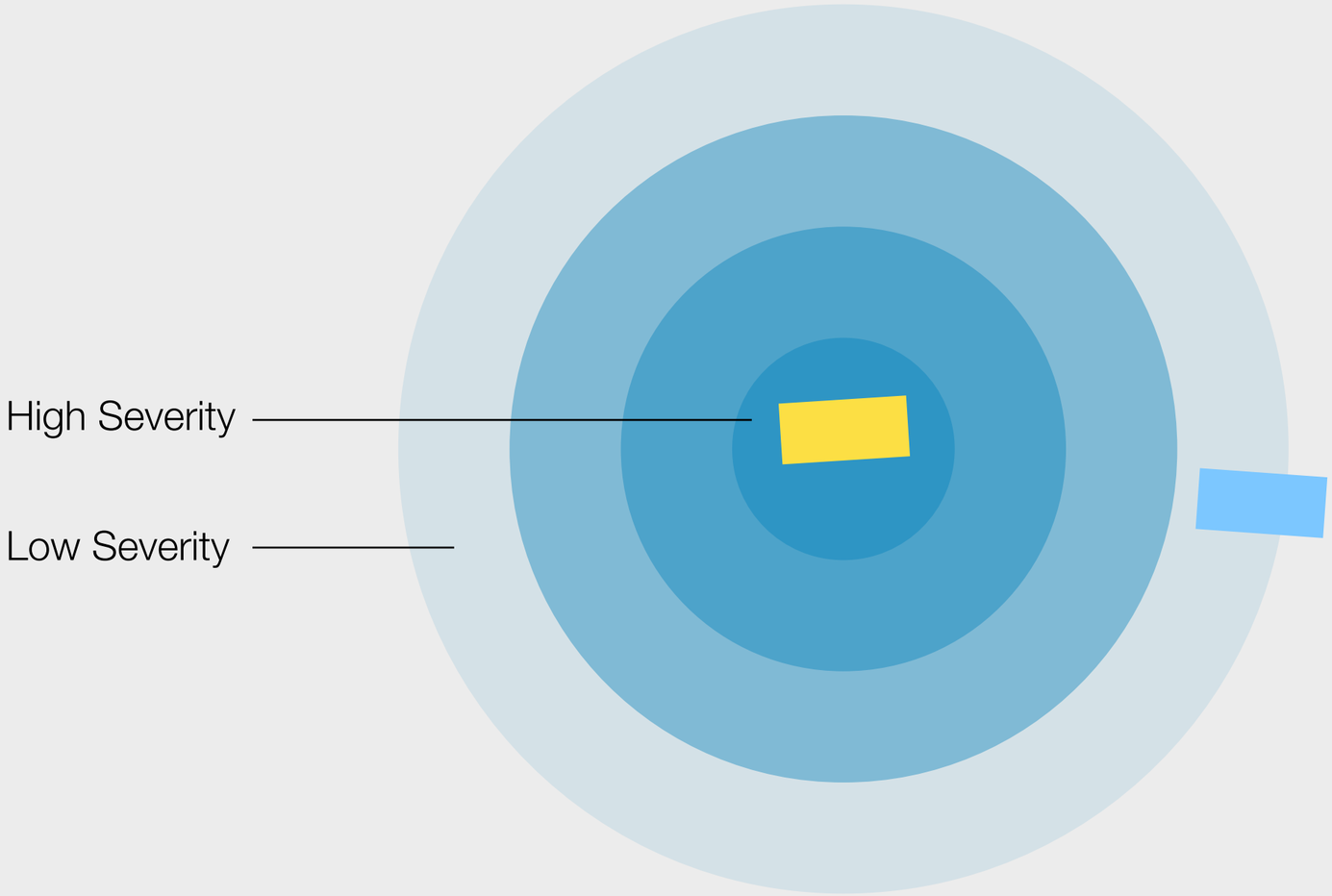
# Stakeholder Map



# Risk Wall


Plan
Work-arounds
Action
Immediately
Monitor
Watch
Closely
Probability
Impact

# Issues



agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# 7 Rules ..... of successfully distributed teams

1. Don’t
2. Don’t treat remotes as if they were locals
3. Don’t treat locals as if they were remote
4. Latitude hurts, longitude kills
5. Don’t always be remote
6. Invest in the appropriate tools and environments
7. Establish standards and agreements

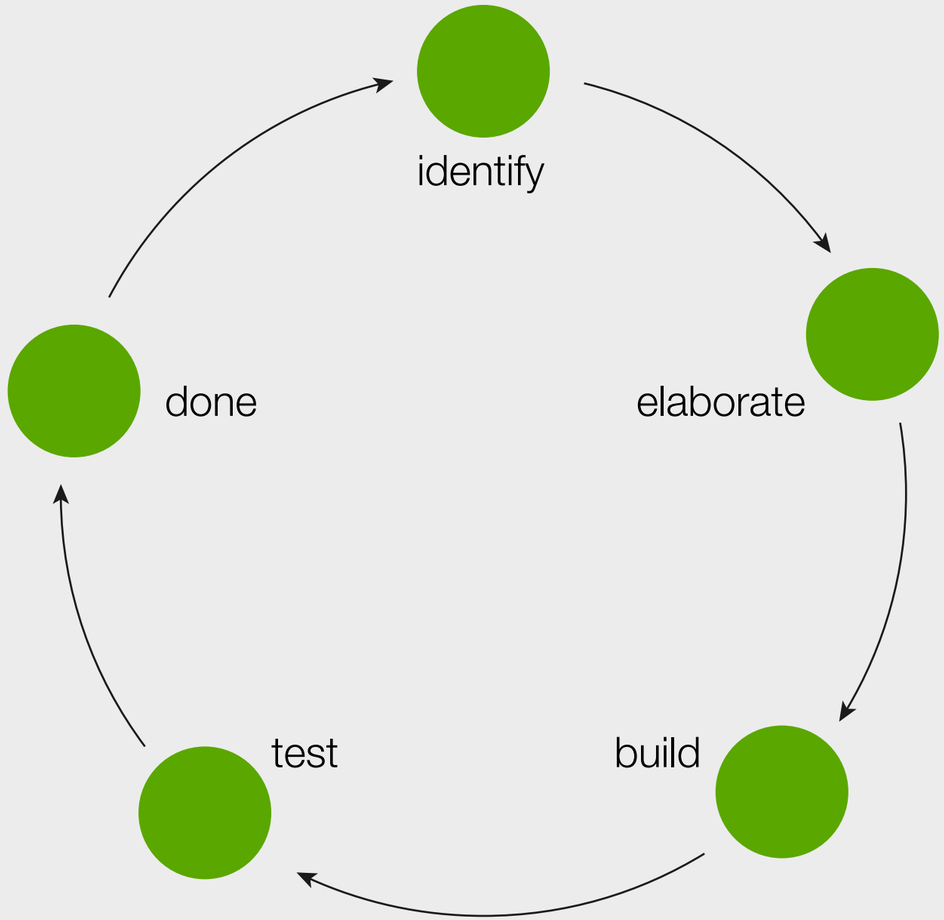
agileACADEMY

# Agile Program Fundamentals

Course IAA2

## Course Topics

# Story Issues




- Scope creep
- Hangover stories
- Changes during the iteration
- Bugs found
- Change in requirement
- Wrong estimates
- Epics not stories
- Unavailable SME
- BAU

# Iteration Smells

Missing team members

Repeating tasks

No impediments raised/same impediments raised day after day

‘Directive’ IM or PM

Specialized job roles/not acting like a team and doing whatever it takes