Project Management

Thomas Luvö
@tomluvoe
tom@samoht.se

Agenda

- Objectives for this lecture

  - What a Project is and what the tasks and responsibilities of the Project Manager are

  - In what environment Projects exist, i.e. different organizational designs in software companies

  - Traditional Project Management approaches and planning

  - Project management triangle in Agile

  - Project types and Agile

Projects

- A planned undertaking to reach of one or several objectives

- Is temporary, all projects have an ending (as opposed to operations, e.g. accounting)

- Creates a unique deliverable

Project Management

- Plan, organize, motivate and control a project

- To reach the project's objectives within the project's constraints

- To optimize the effort vs. output

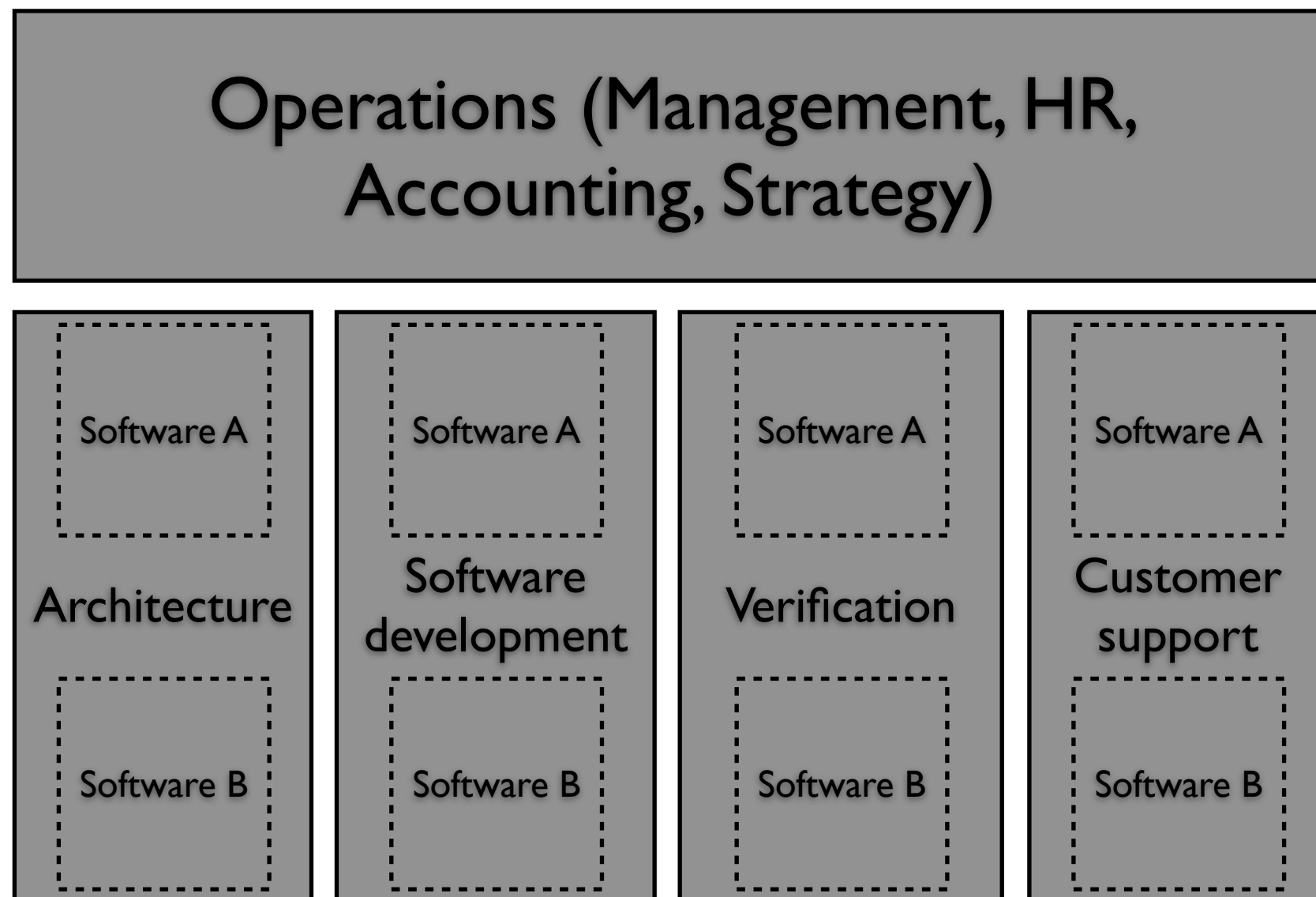- I.e. Deliver the product to the customer on time and within budget

Organizational Basics

Basic Software Co.

Operations (Management, HR, Accounting, Strategy)
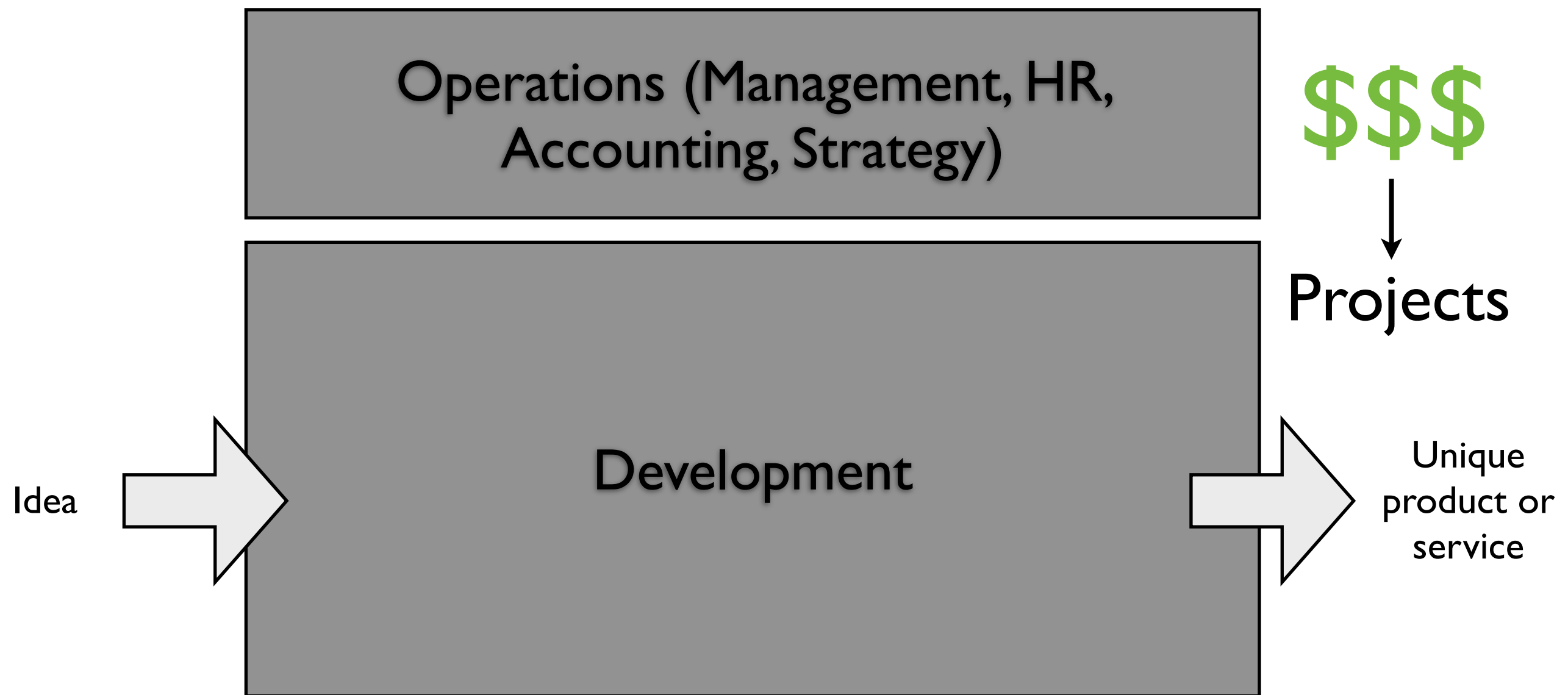
$$$

↓

Projects

Idea →

Development

→ Unique product or service

Functional Organization

Basic Software Co.
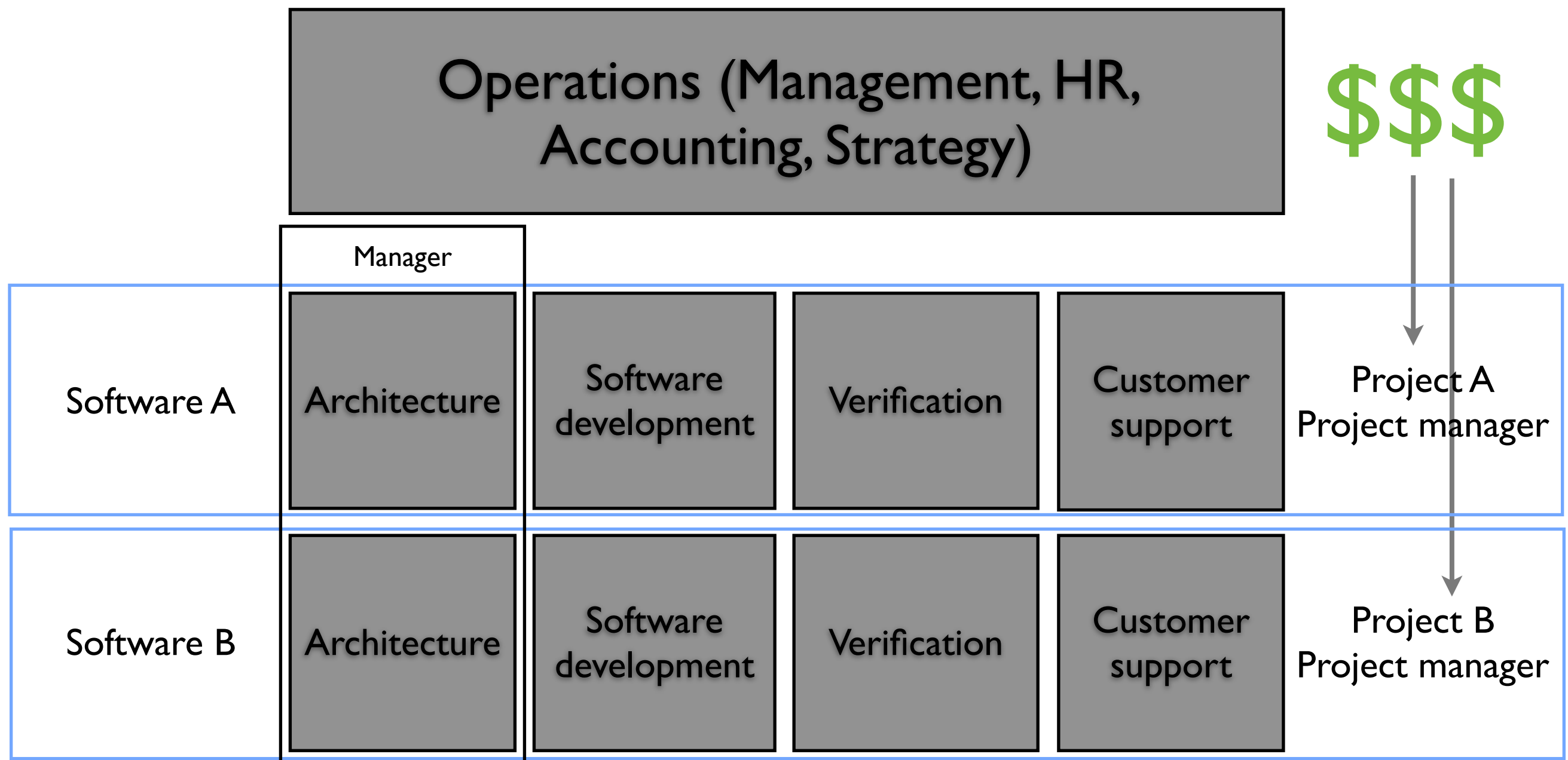
Divisional Organization

Basic Software Co.

| Operations (Management, HR, Accounting, Strategy)	
Software A (architecture, design, verification, support)	Software B (architecture, design, verification, support)

Matrix Organization

Basic Software Co.

	Manager				
	Operations (Management, HR, Accounting, Strategy)				**$$$**
Software A	Architecture	Software development	Verification	Customer support	Project A Project manager
Software B	Architecture	Software development	Verification	Customer support	Project B Project manager

Team oriented (Agile) Organization

Basic Software Co.

Operations (Management, HR, Accounting, Strategy)

$$\$\$\$$$

Manager   Manager   Manager   Manager

Product A
Product owner

Product B
Product owner
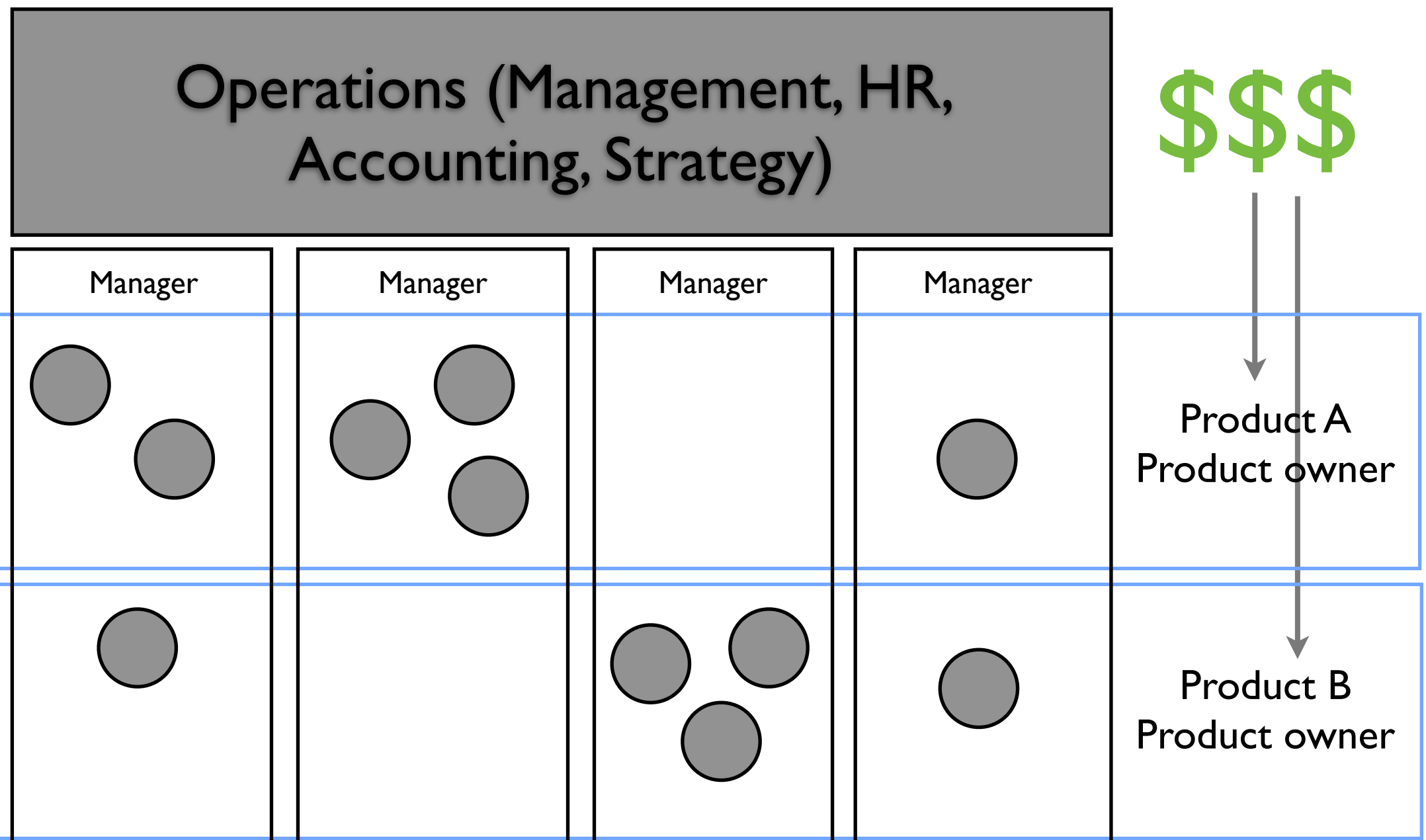
Traditional Project Management Approaches

- Waterfall

- Critical Chain

Traditional Waterfall

Requirements

Design

Implementation

Integration

Test and debug

Release

Maintenance

- Originates in manufacturing and construction where changes are expensive

- Each step should be completed before moving to the next

- Bugs or design flaws found early in the projects are cheap to fix

- Structured and focus on documentation makes it easy to spread information

- Functional or matrix organization

- Reality is seldom as predictable, and it was actually presented as a flawed model for software projects..

"Waterfall" in practice
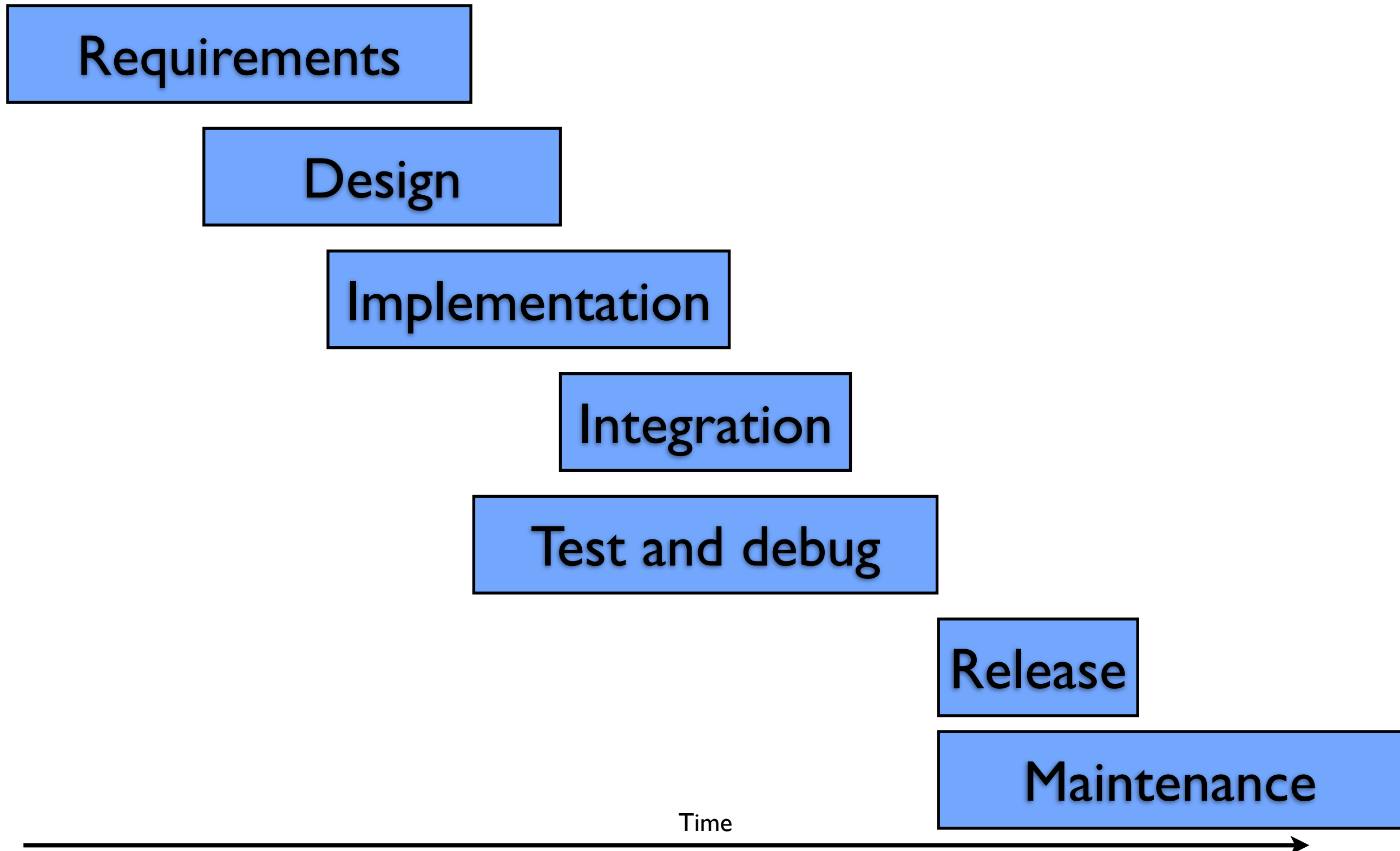
Requirements

Design

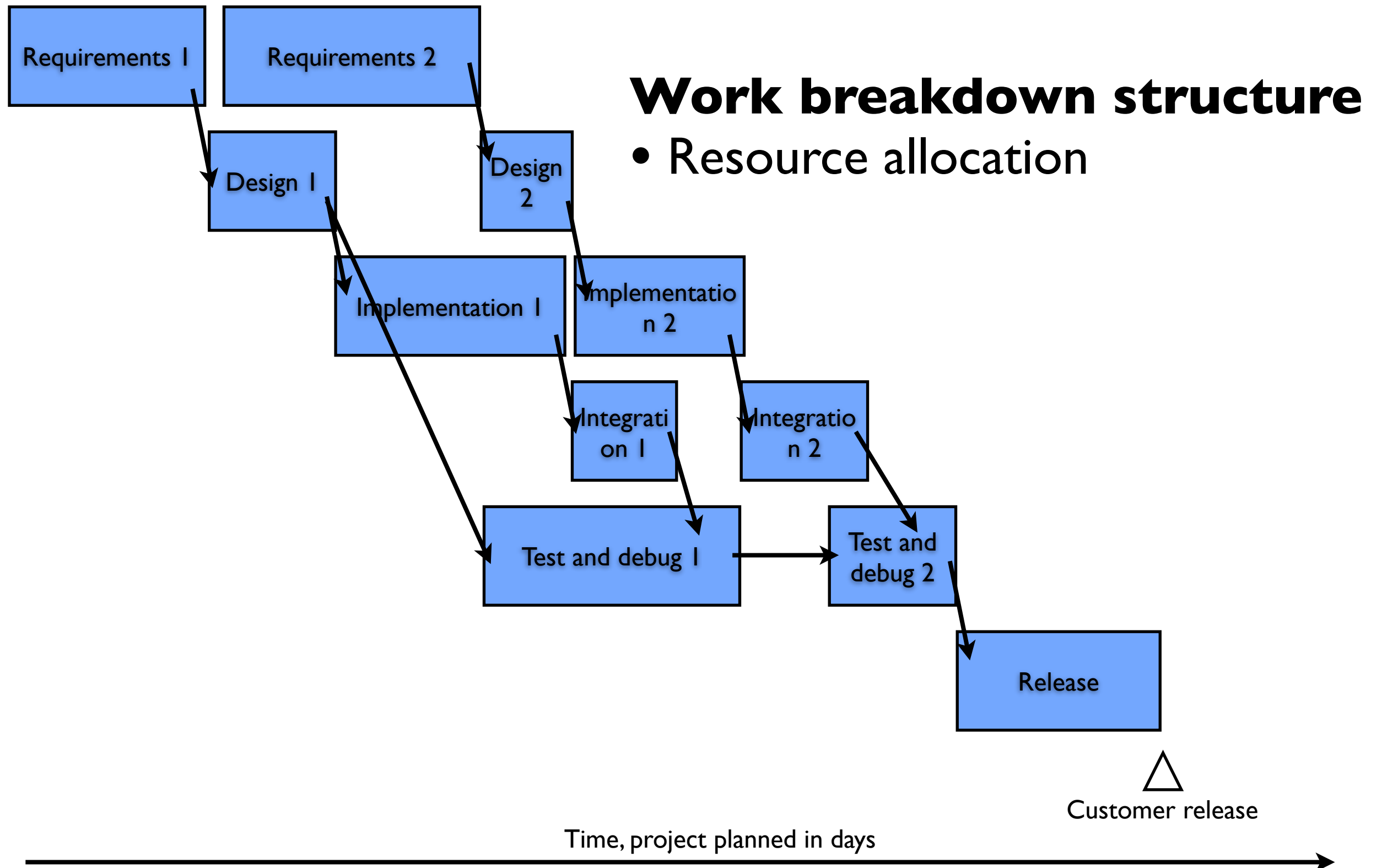Implementation

Integration

Test and debug

Release

Maintenance

- Design, implementation, integration and verification is done during the execution phase of the project

- Project work is done in functional teams, coordinated by a project manager team

- Typically three or four releases are done to customer during a software project
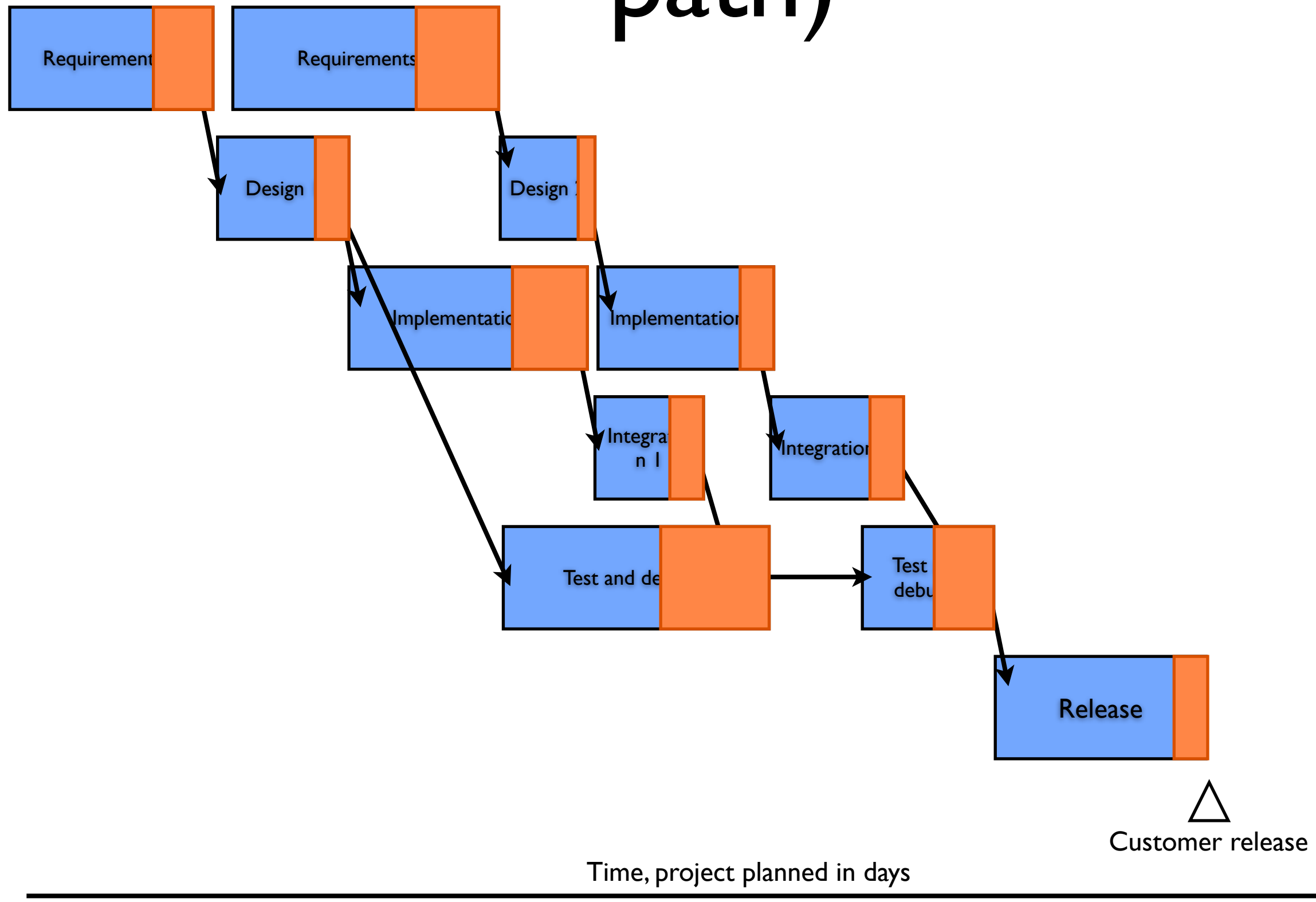
Gantt chart

Requirements

Design

Implementation

Integration

Test and debug

Release

Maintenance

Time

Gantt chart

Requirements 1

Requirements 2

Work breakdown structure
- Resource allocation

Design 1

Design 2

Implementation 1

Implementation 2

Integration 1

Integration 2

Test and debug 1

Test and debug 2

Release

△
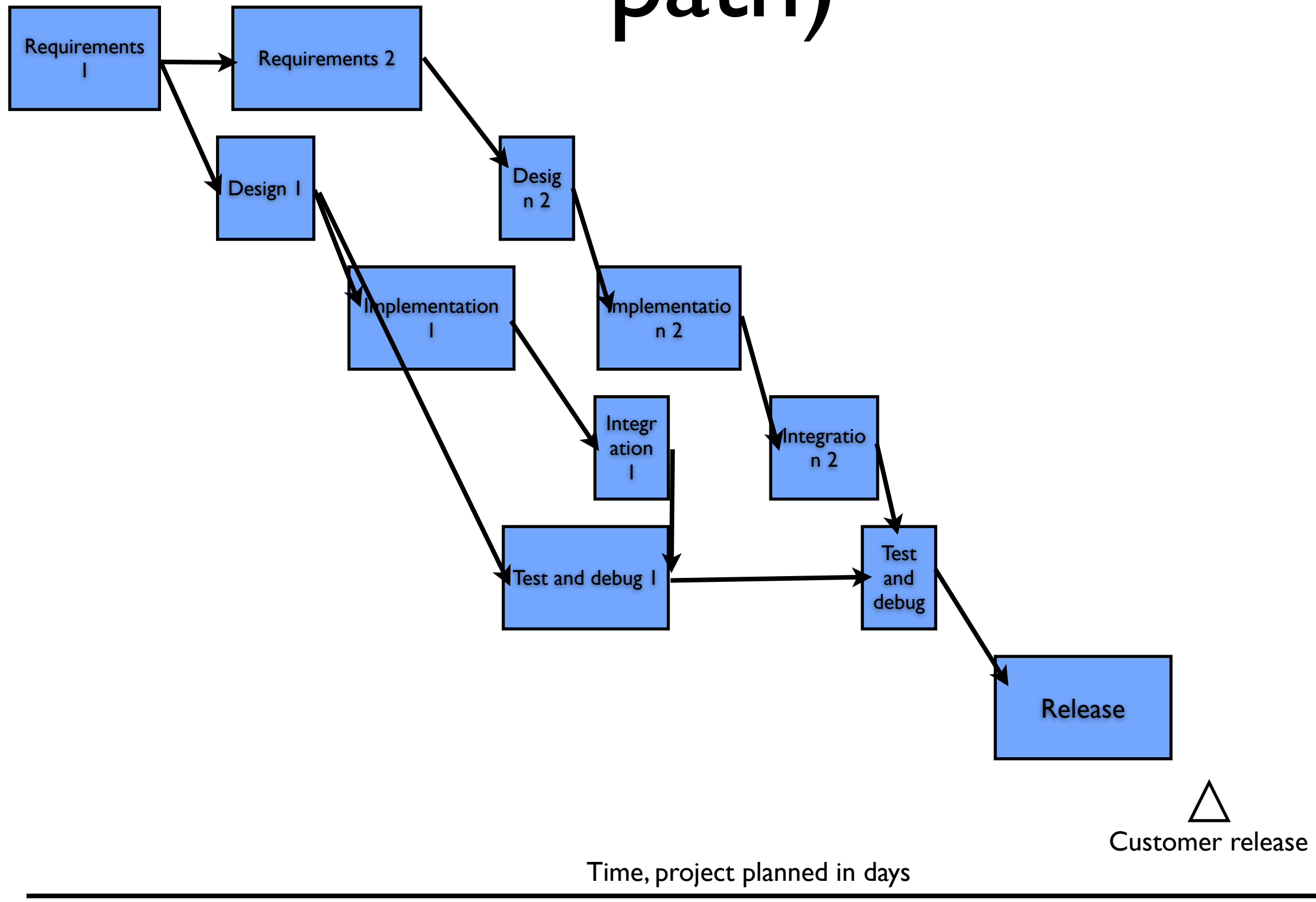Customer release

Time, project planned in days

Gantt chart limitations

- Does not show how much work a box represents as the box only shows lead-time

- Complex plan for complex projects

- Delays may have huge impact on chart and real allocations

- If work is finished early, there is a tendency to wait until next activity is planned
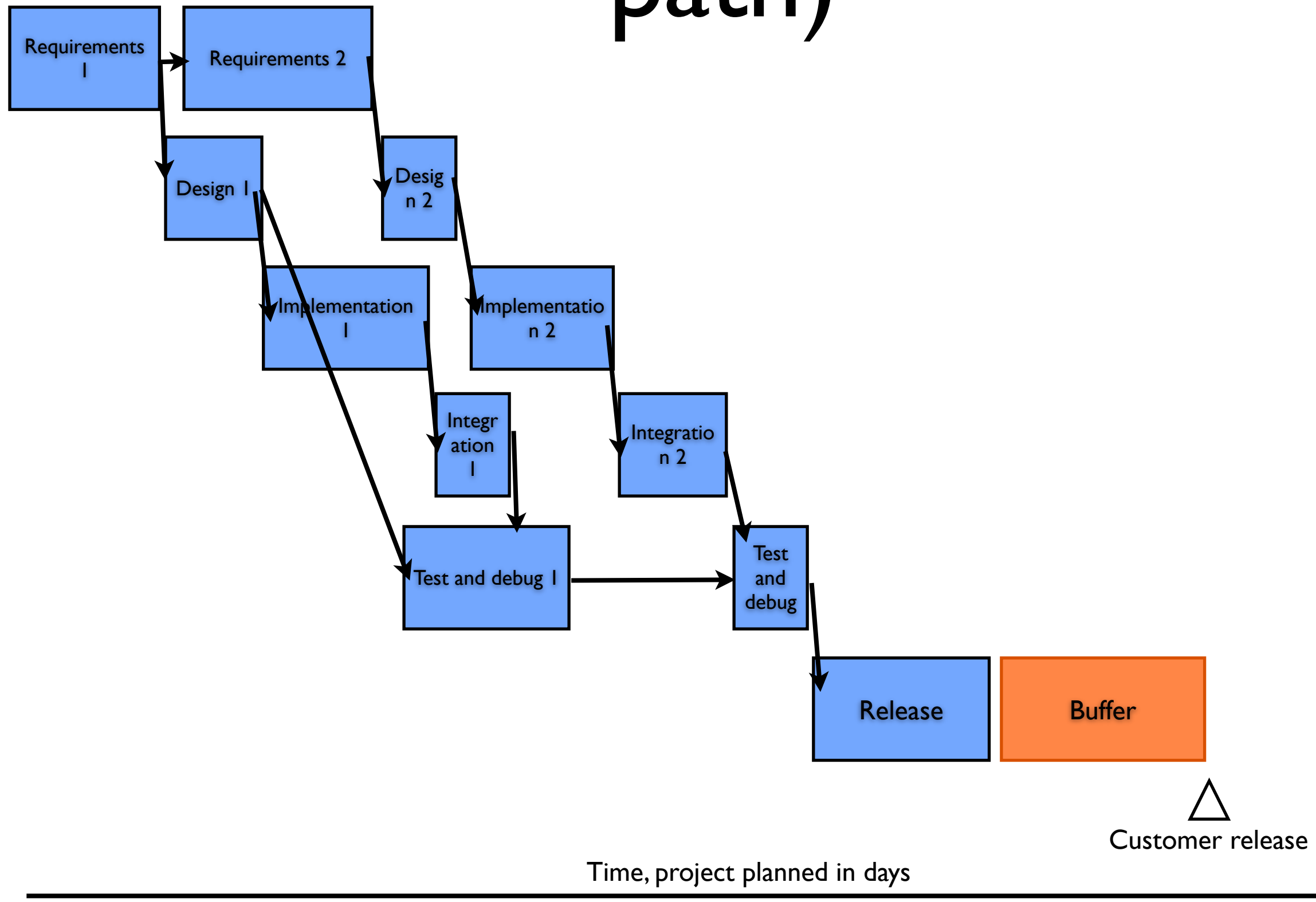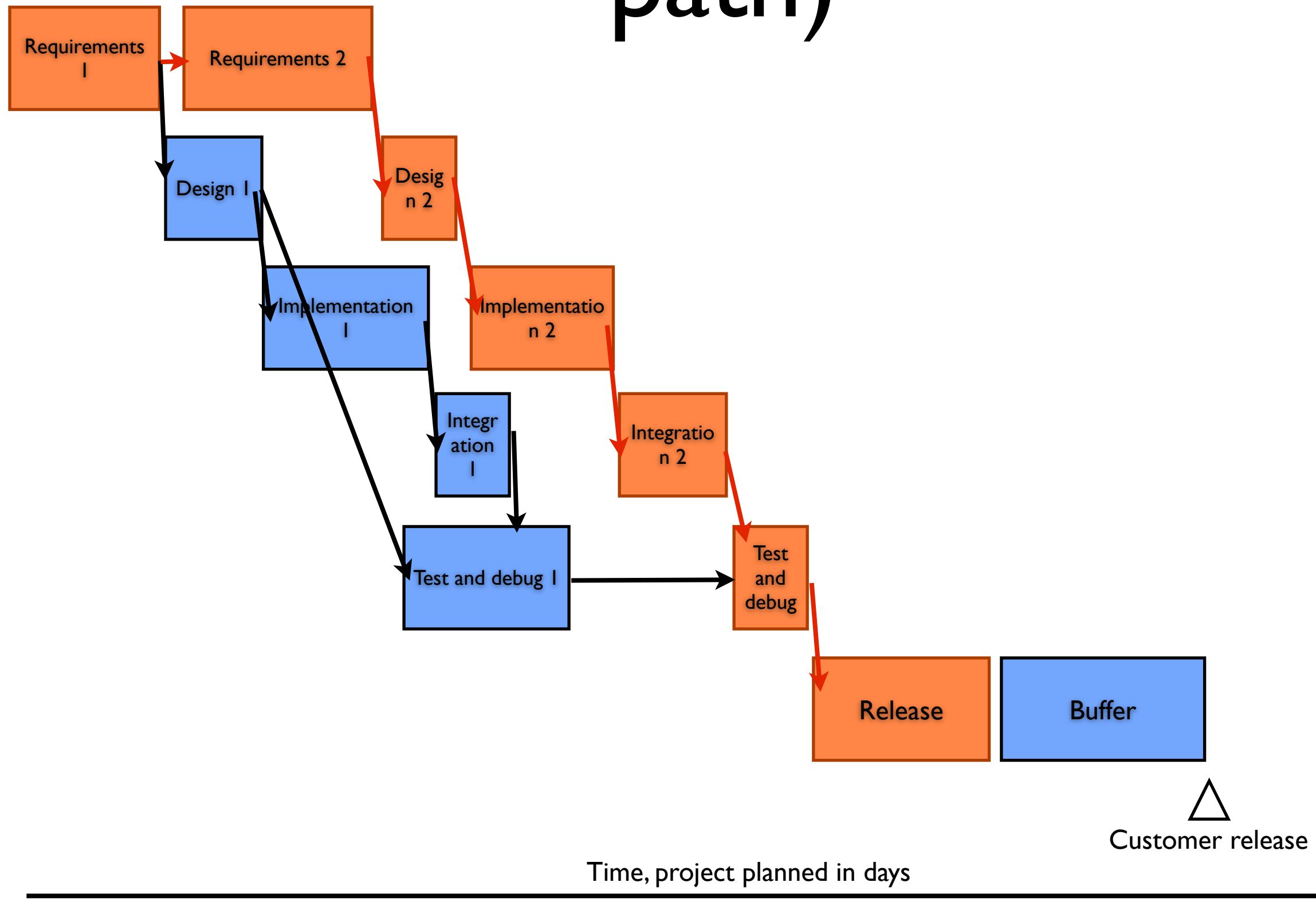
Improvements (e.g. Critical path)

Release

△
Customer release

Time, project planned in days

Improvements (e.g. Critical path)

Time, project planned in days

Customer release

Improvements (e.g. Critical path)

Requirements 1 → Requirements 2 → Design 1 → Design 2 → Implementation 1 → Implementation 2 → Integration 1 → Integration 2 → Test and debug 1 → Test and debug → Release → Buffer

△
Customer release

Time, project planned in days

Improvements (e.g. Critical path)

Requirements 1

Requirements 2

Design 1

Design 2

Implementation 1

Implementation 2

Integration 1

Integration 2

Test and debug 1

Test and debug

Release

Buffer

Customer release

Time, project planned in days

Project Reality

- Product requirements will change between specification of a product and delivery

- Humphrey's Requirements Uncertainty Principle: A system can not fully be understood before it has been used

- Ziv's Uncertainty Principle for Software Development: That uncertainty and unpredictability is always part of software development

- Wegner's Lemma: It is not possible to fully specify an interactive system

Iterative way of working

Incremental product development

Team oriented

Scrum (Agile)

Product backlog
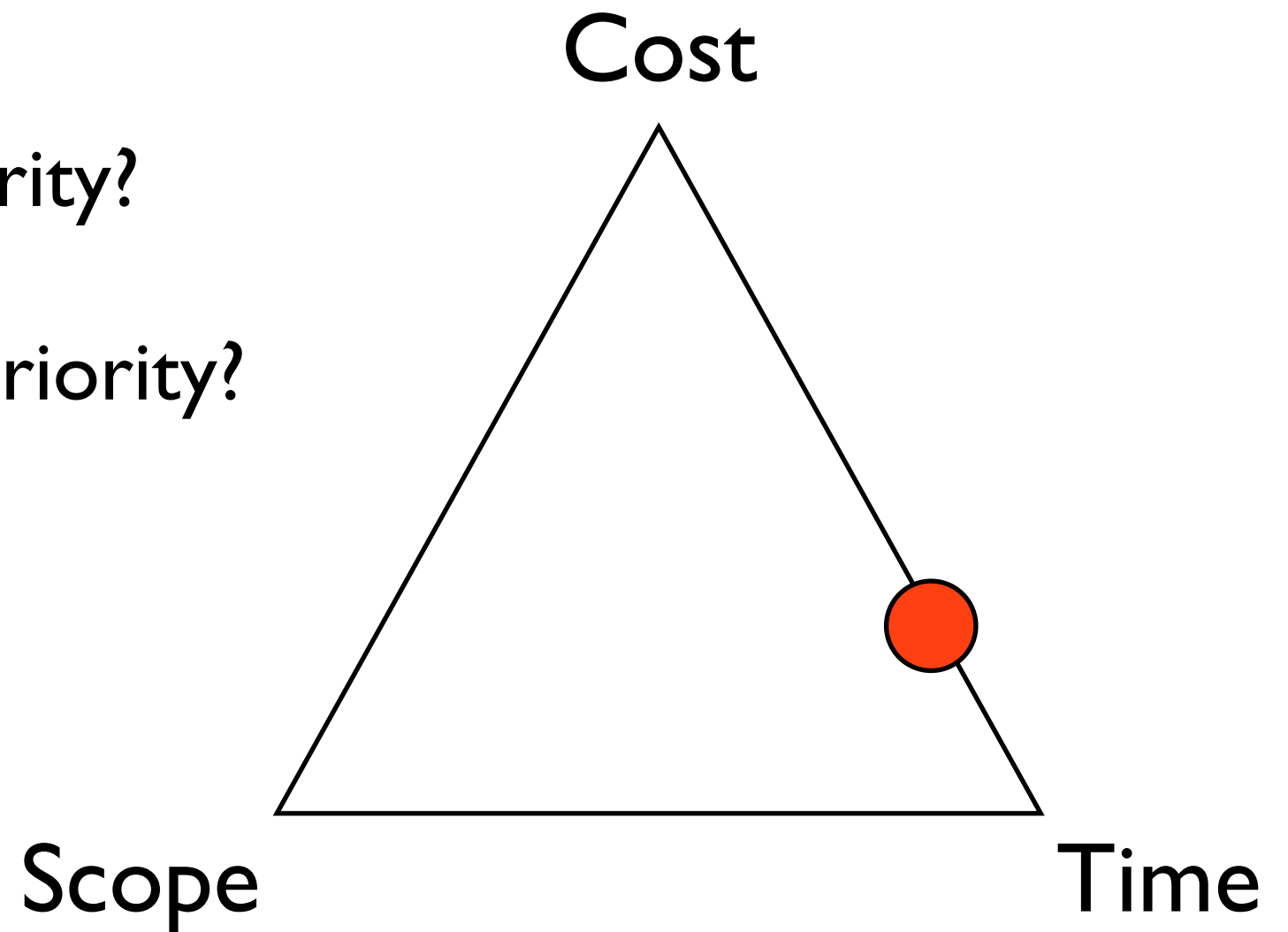
PO

Customer

90%

75%

ScM
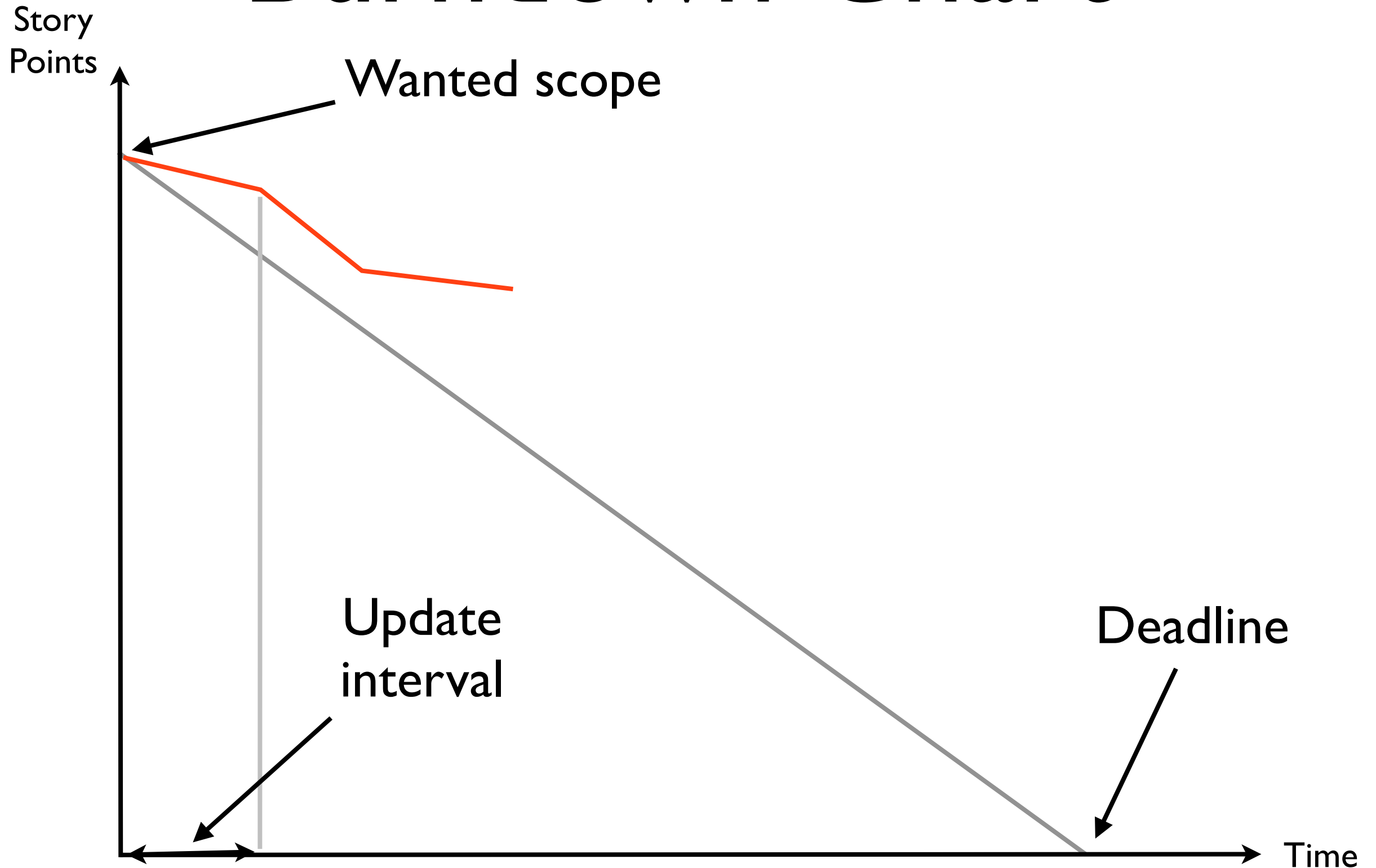
Development team
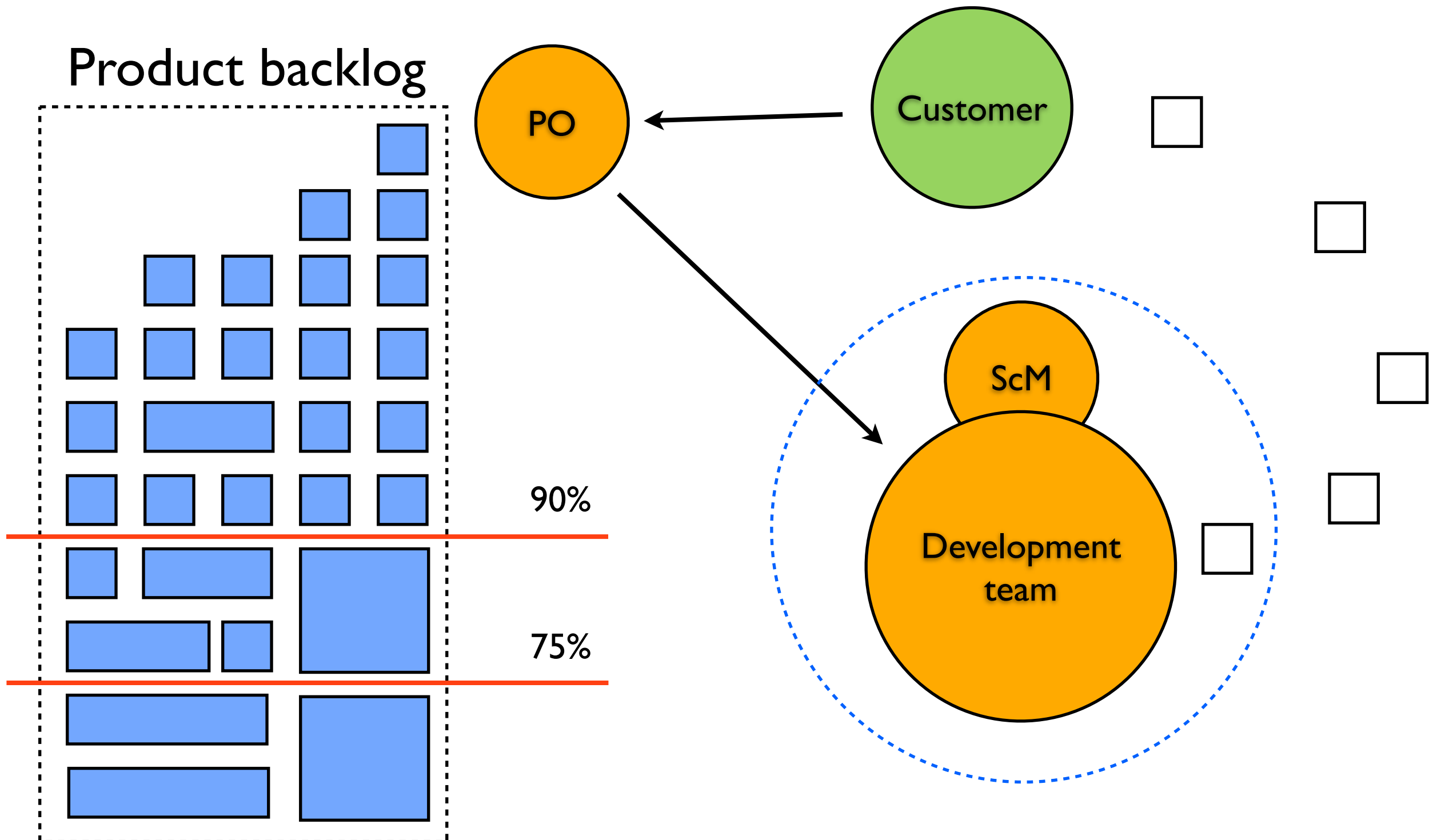
Project Management Triangle

- Cost

  - The amount of money available for the project. Can be translated to both material (e.g. computers, lab equipment) as well as number of project members (employees, consultants)

- Scope

  - The deliverable's scope, i.e. number of features agreed with customer

- Time

  - The agreed delivery date of the finished product
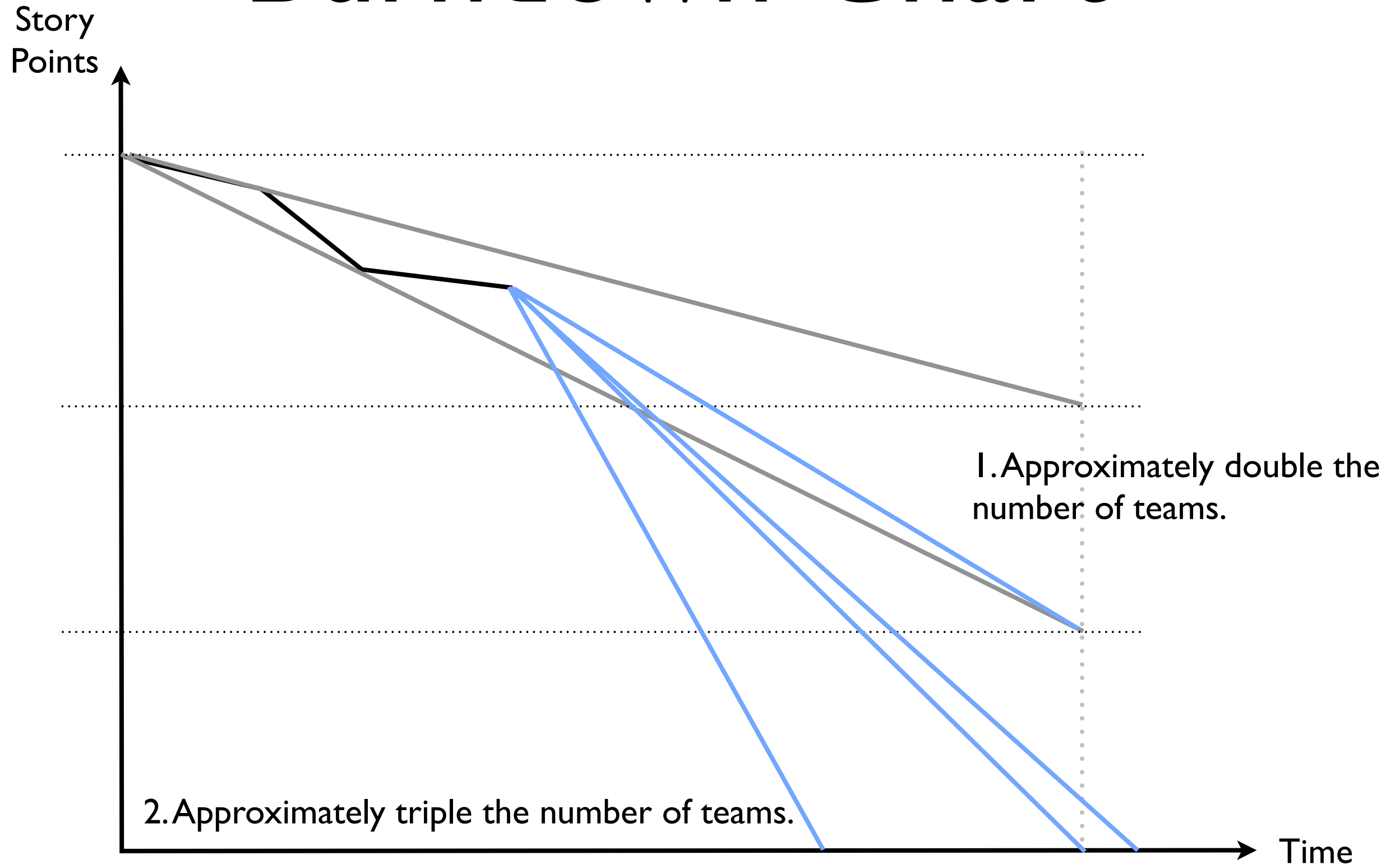
Project Management Triangle

1. What is your first priority?
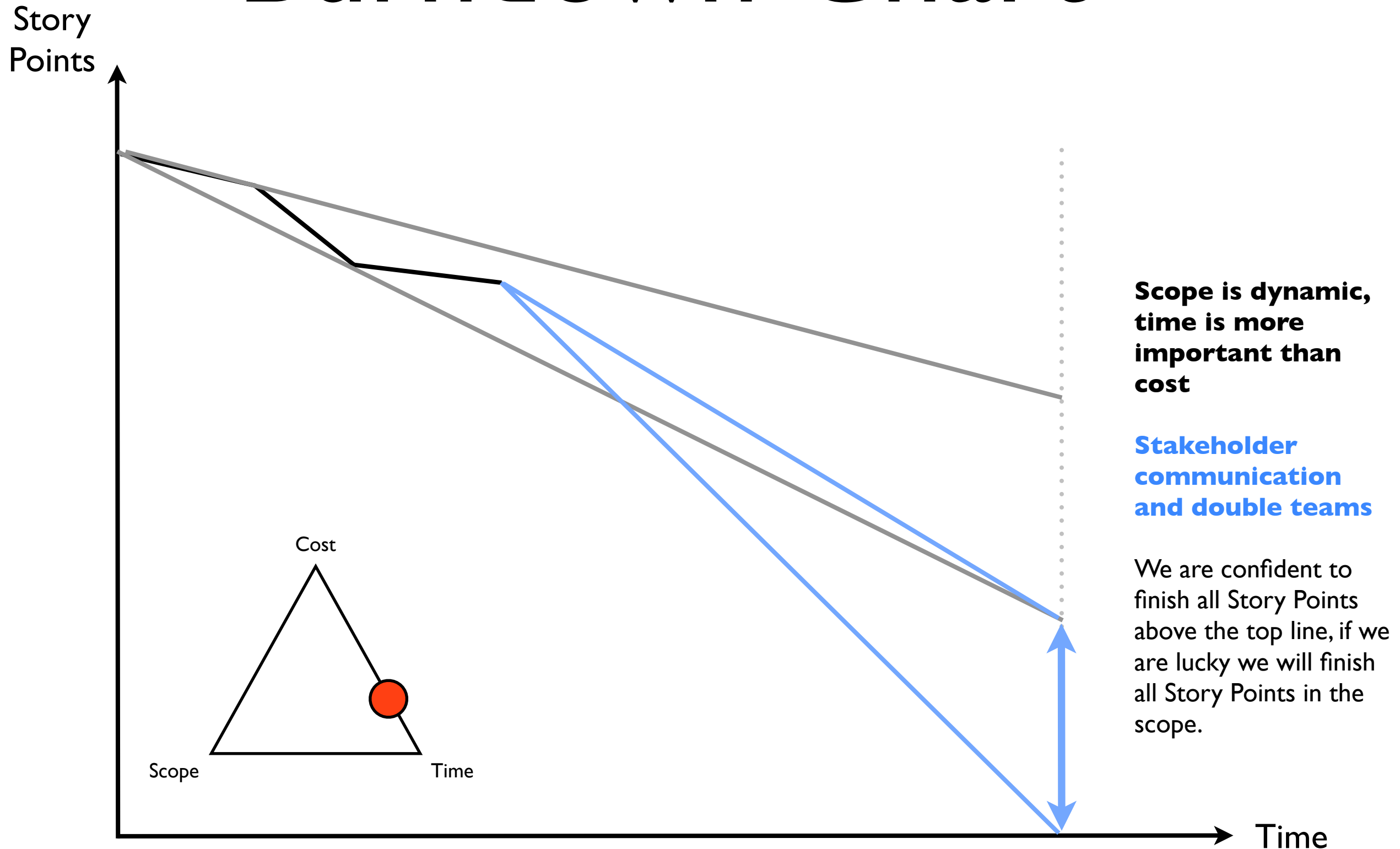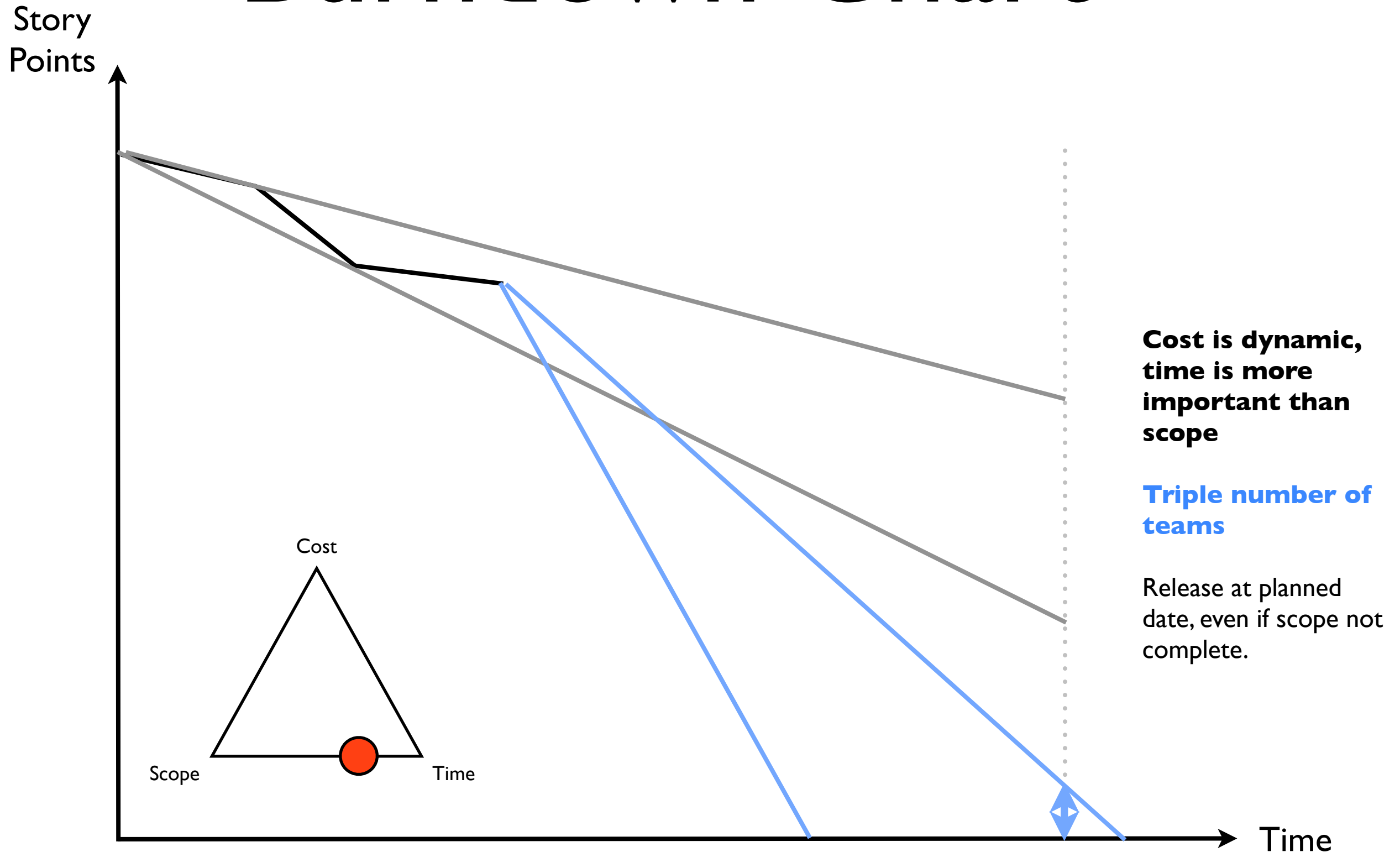
2. What is your second priority?

Cost

Scope

Time

Burndown Chart

Story
Points

Wanted scope

Update
interval

Deadline

Time

Burndown Chart

Story
Points

1. Approximately double the number of teams.
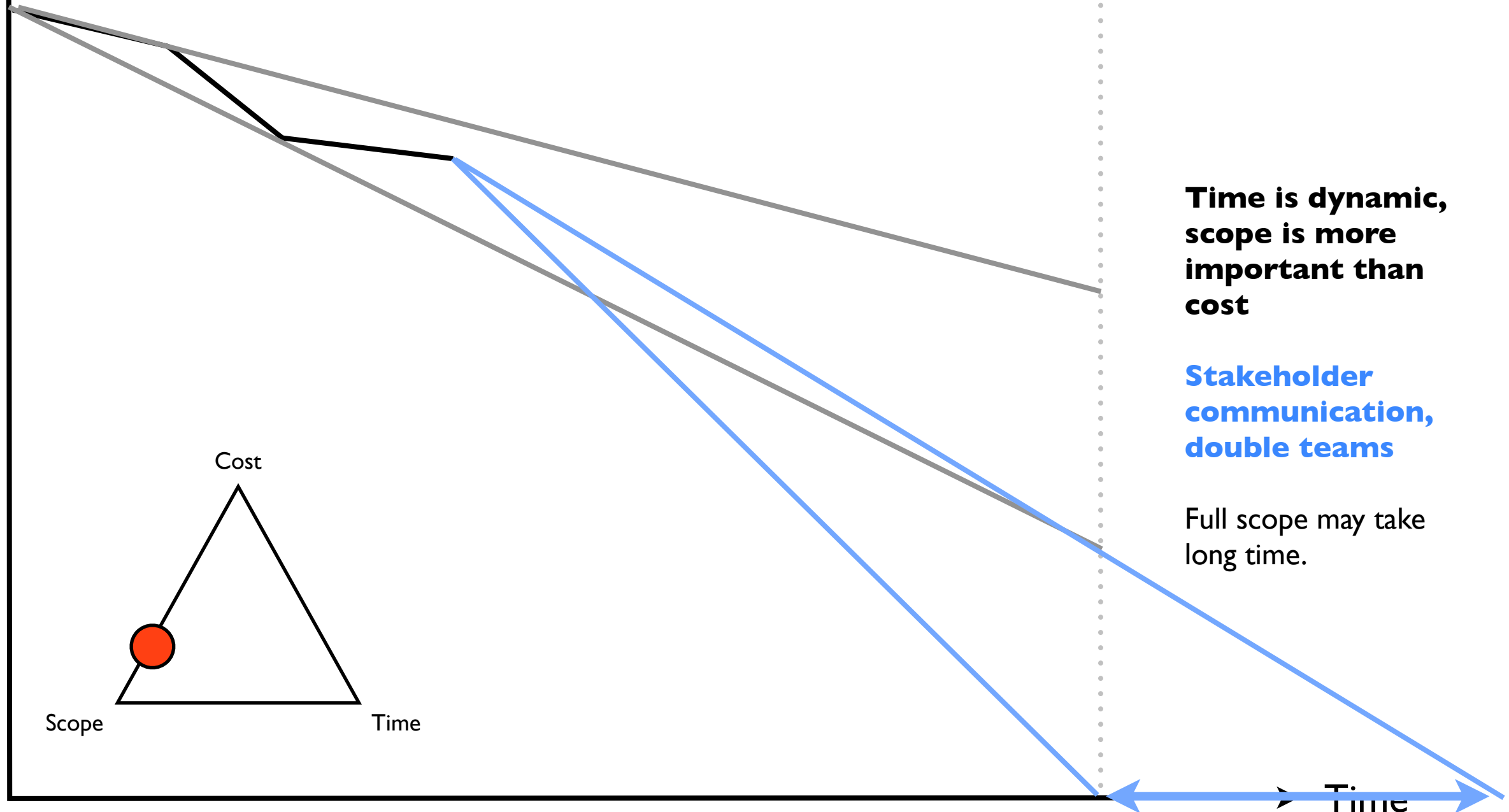
2. Approximately triple the number of teams.

Time

Burndown Chart

Story Points

Scope is dynamic, time is more important than cost

Stakeholder communication and double teams

We are confident to finish all Story Points above the top line, if we are lucky we will finish all Story Points in the scope.

Cost

Scope   Time

Time

Burndown Chart

Story Points

Cost is dynamic, time is more important than scope

Triple number of teams

Release at planned date, even if scope not complete.

Cost

Scope   Time

Time

Burndown Chart

Cost is dynamic, scope is more important than time

Triple number of teams

Finish complete scope before release, even if it takes longer than planned.

Burndown Chart

Story Points

Time is dynamic, scope is more important than cost

**Stakeholder communication, double teams**

Full scope may take long time.

Cost

Scope   Time

Time

Burndown Chart

Story Points

Time is dynamic, cost is more important than scope

Stakeholder communication, complete scope

Full scope may take a long time.

Cost

Scope   Time

Time

Burndown Chart

Story Points

Time is dynamic, cost is more important than scope

Stakeholder communication, complete scope

Full scope may take a long time.

Cost

Scope   Time

Time
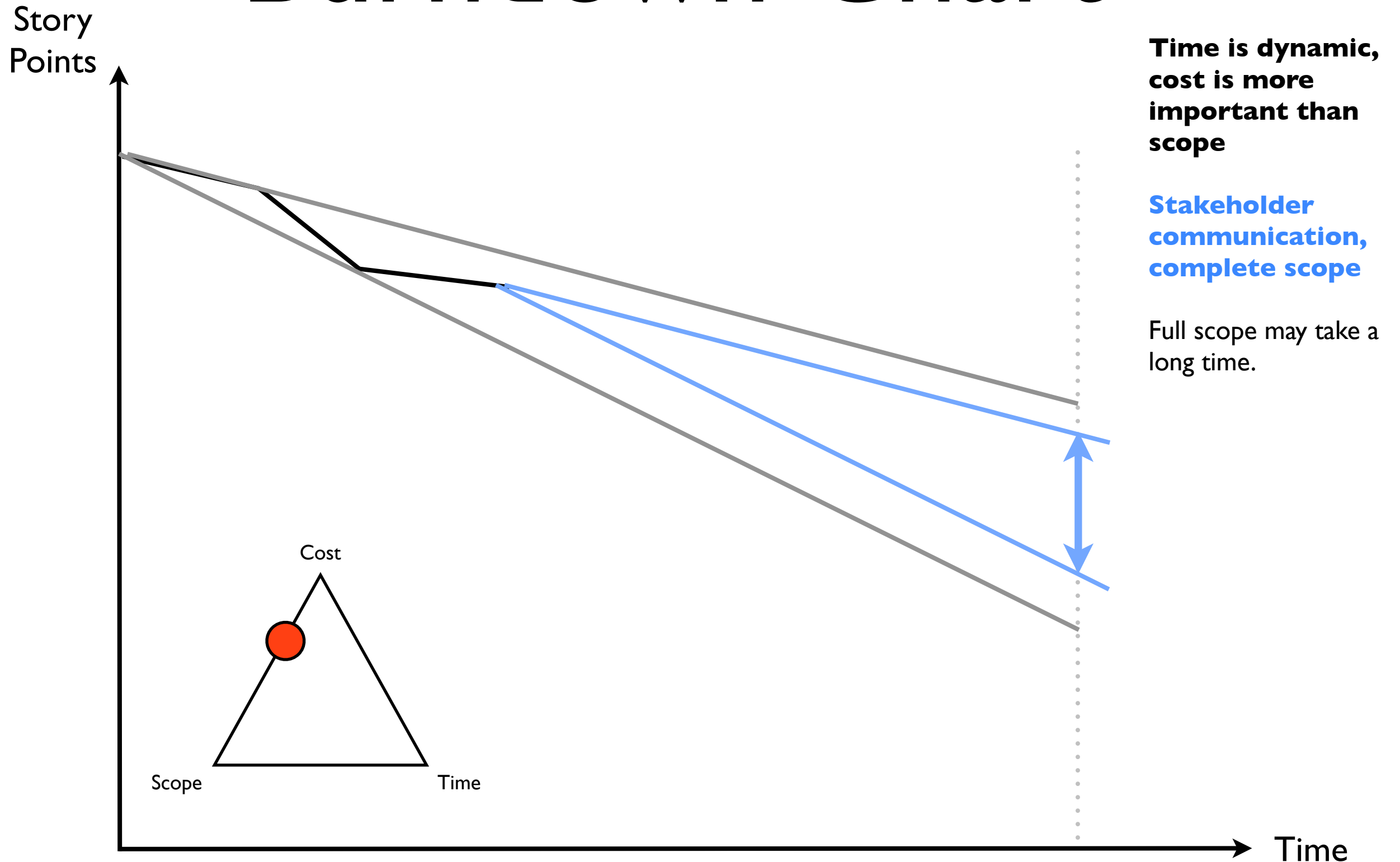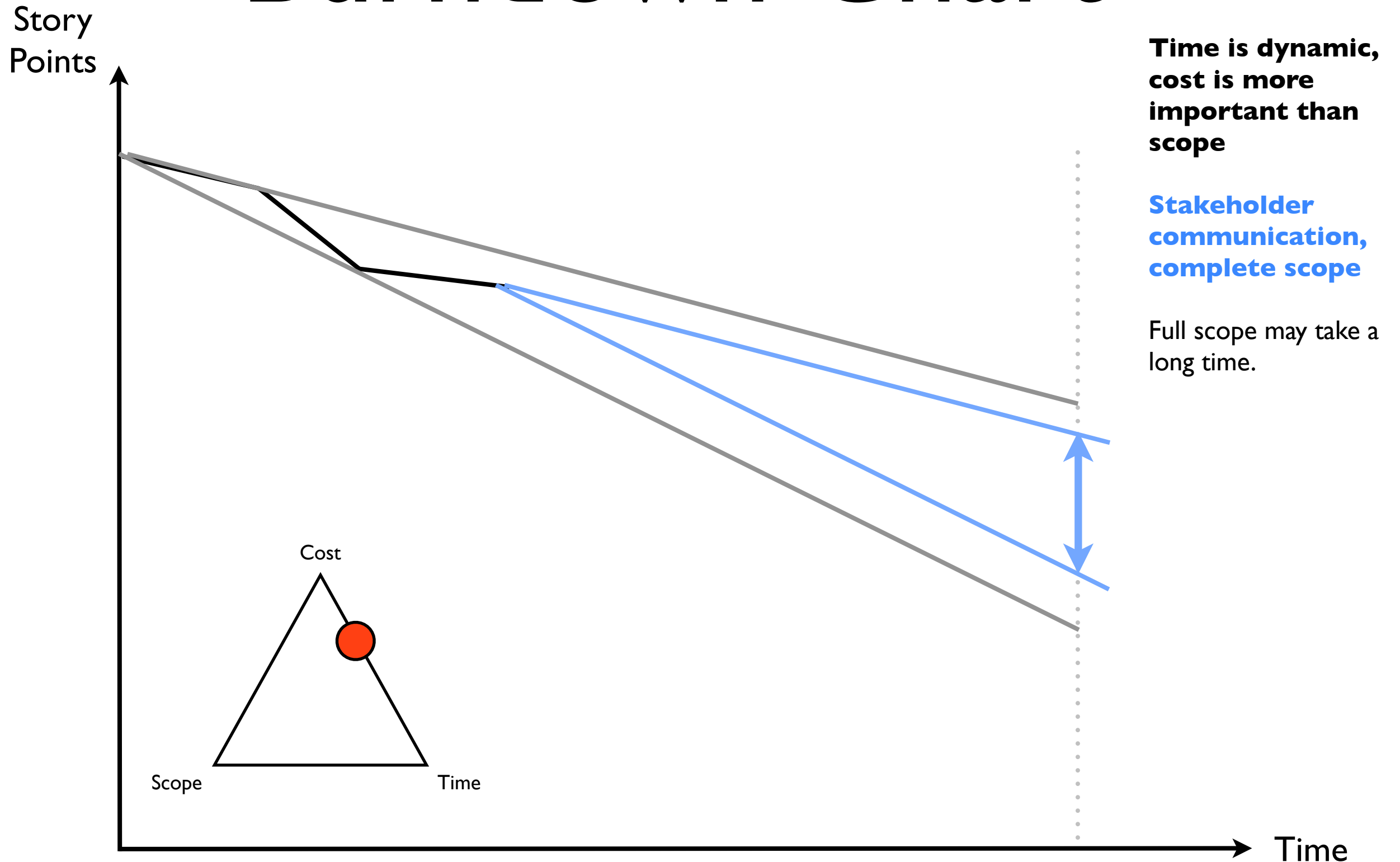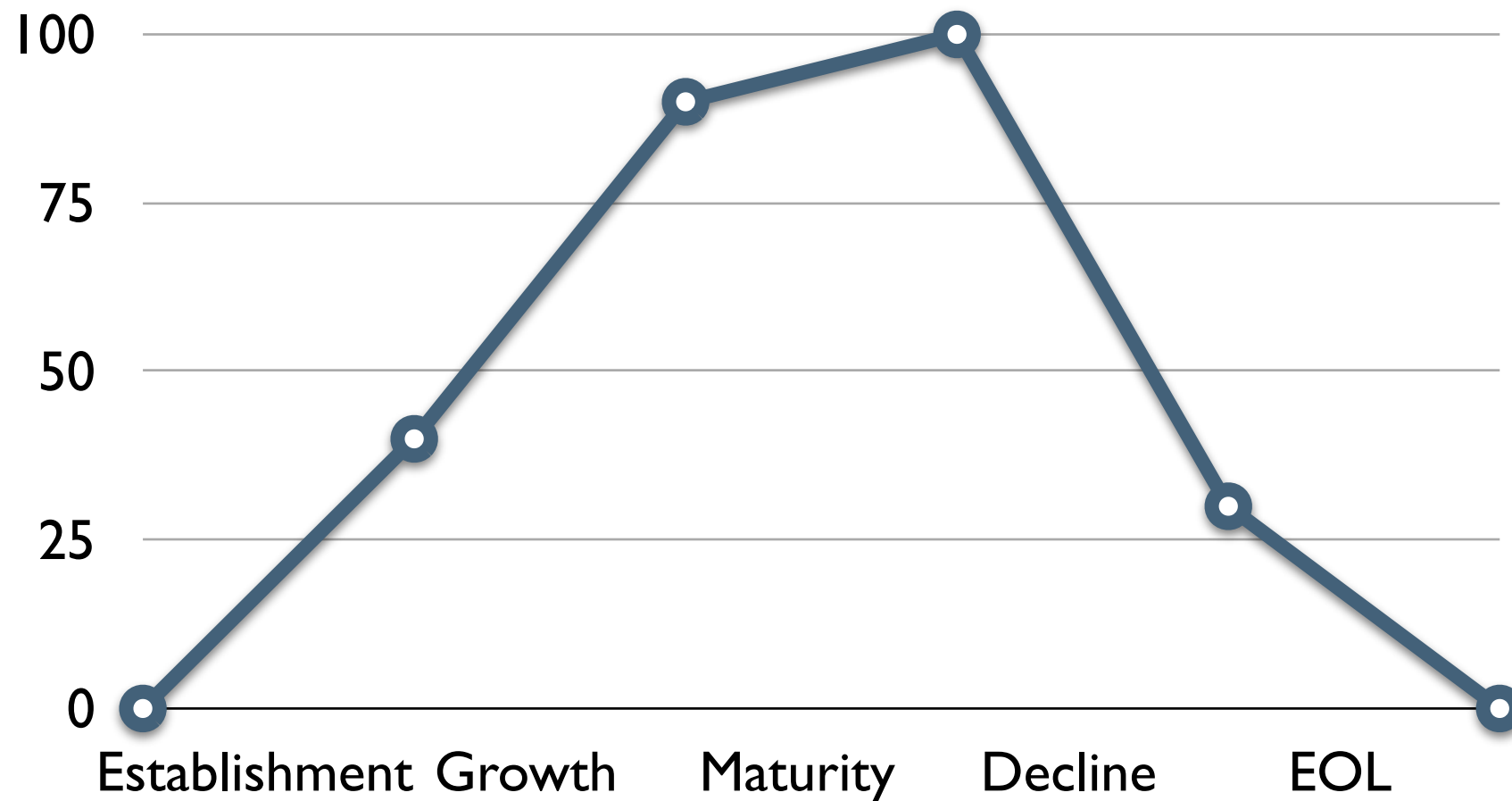
Project Types

Examples of project types

- Research project
- Development project
- Maintenance project

Agile tries to avoid projects by focusing on products instead.

Product focus

- Avoids different types a projects by prioritizing all tasks (new features, bug fixes, customer issues, etc.) together

- Benefits (less overhead, simple organization, one prioritization)

- Risks (customer issues have higher priority than ongoing development)

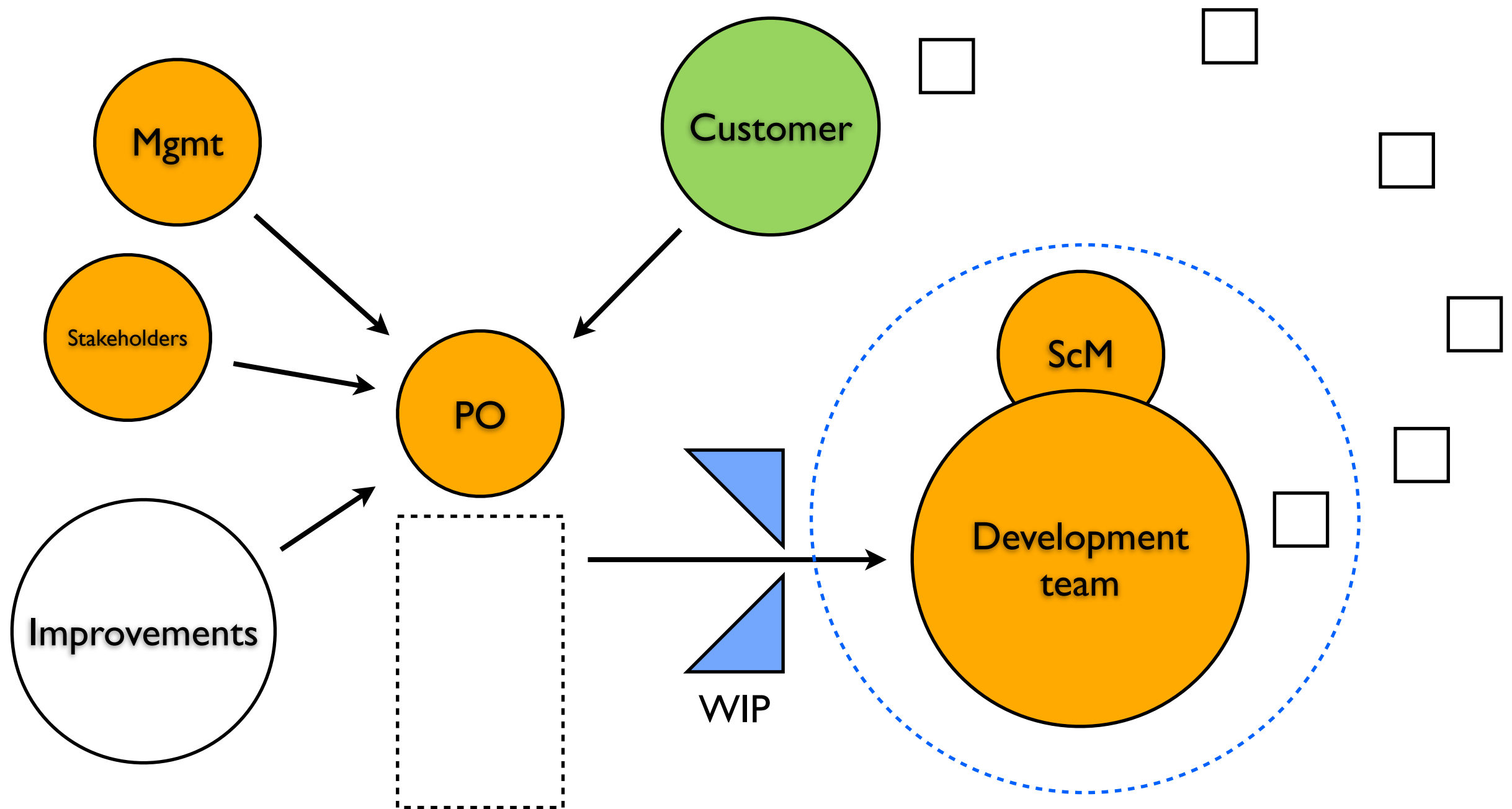# Team responsibility (or is there a need to go beyond "just teams"?)
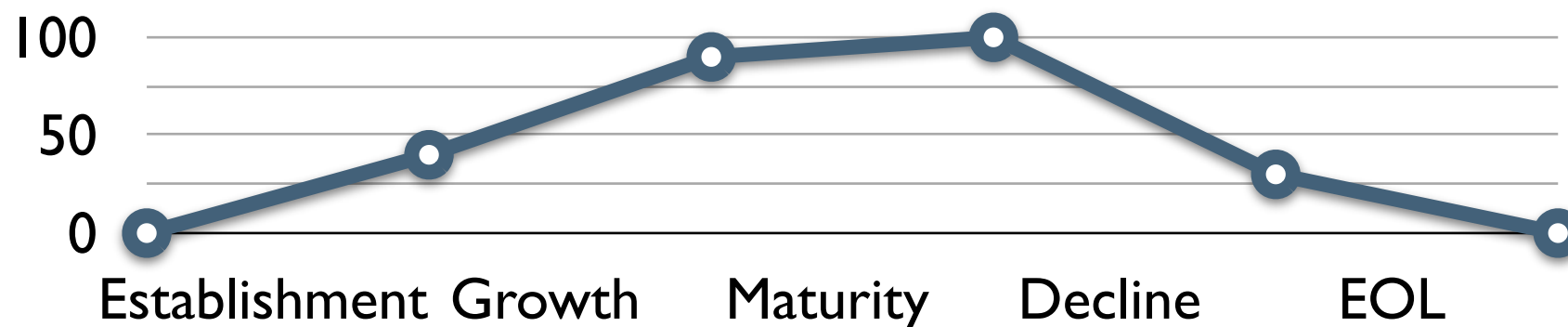
- Product research

- Product development

- Product verification

- Release environment

- Development environment

- Customer installation (if needed)
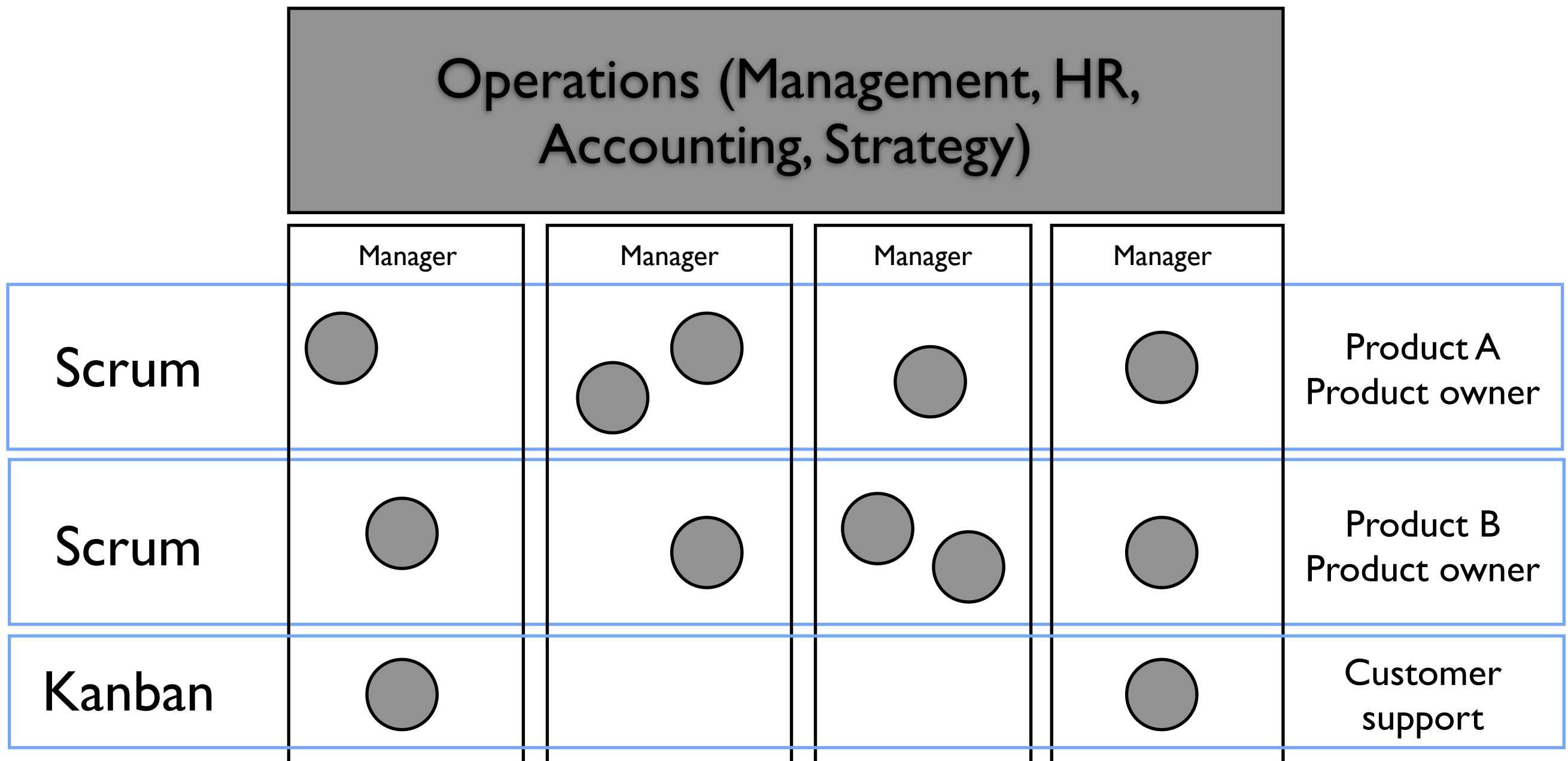
- Customer support

Scrum Team Environment
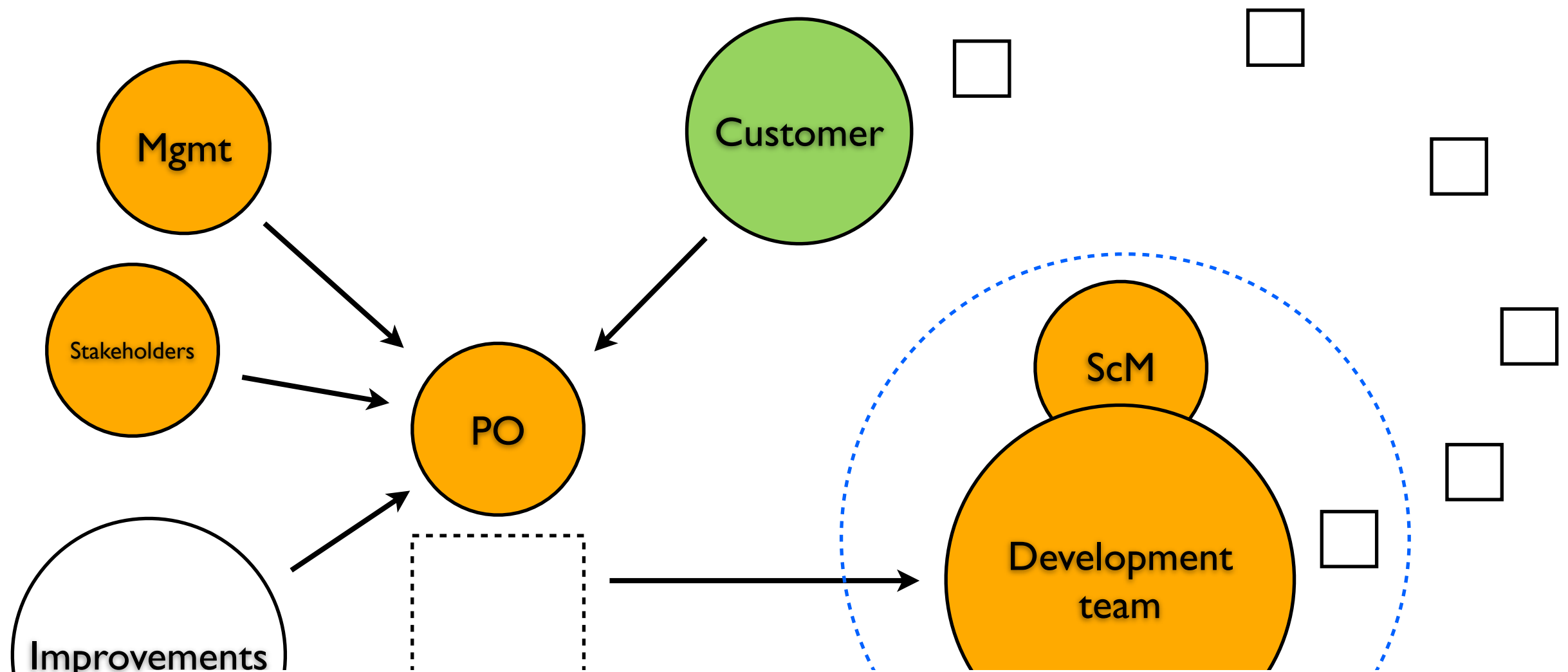
Sprints adds a lead time of average 0.5 Sprints.
I.e. not suitable for customer support.

~~Scrum~~ Kanban Team Environment

Agile Organization

Basic Software Co.

	Operations (Management, HR, Accounting, Strategy)				
	Manager	Manager	Manager	Manager	
Scrum	●	● ●	●	●	Product A Product owner
Scrum	●	●	● ●	●	Product B Product owner
Kanban	●			●	Customer support

One last thing...

- All methods, strategies, risk management, processes aside

- Project Management is first and foremost about leading people

Q&A