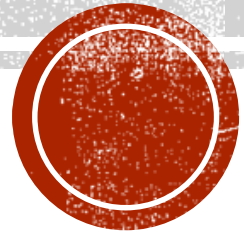
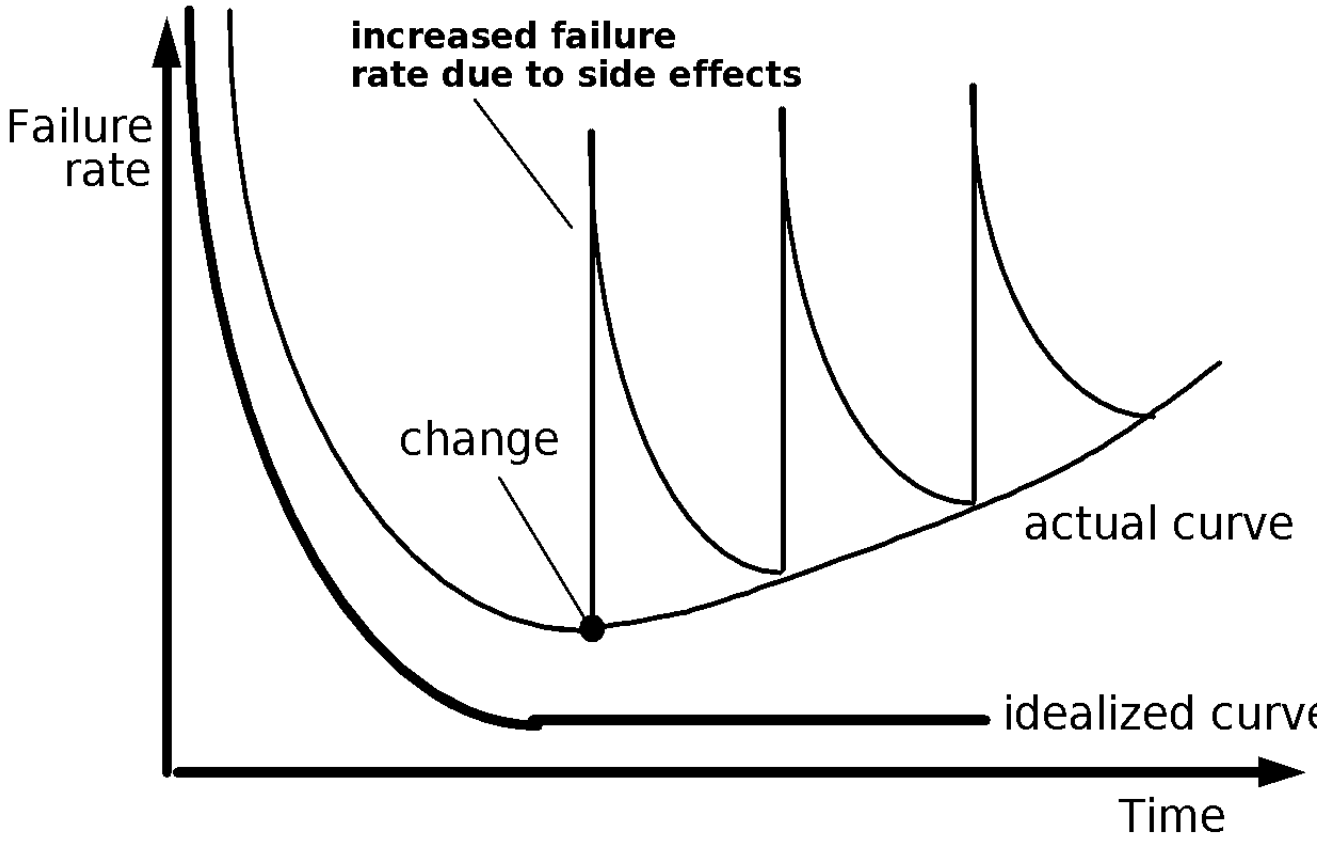
Software Engineering 2
SE1 Introduction (1)



Ehsan Sharifi

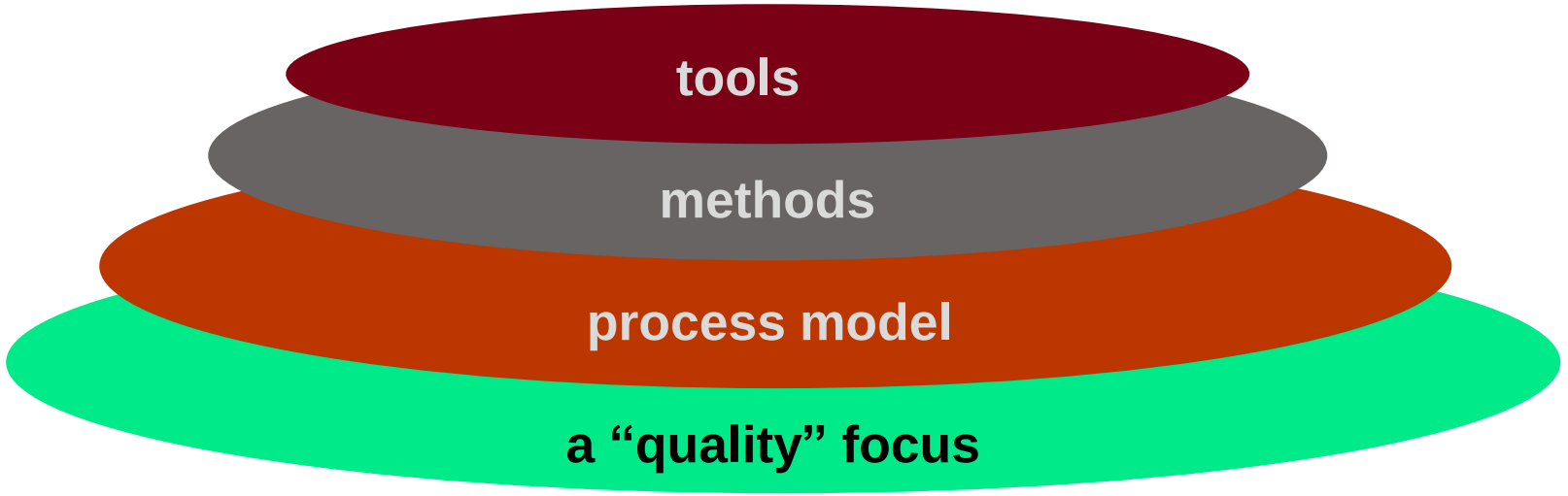
Department of Computer Engineering Amirkabir University of Technology

Hardware vs Software



The hardware begins to wear out
Software doesn't wear out. However, it does deteriorate with maintenance

A Layered Technology



method/process vs methodology

Level of abstraction	Example of application	Typical product
Task	Developing a first-cut class diagram	Specific version of a class diagram
Technique	Description of how to carry out a technique, e.g. UML class modelling	Any UML class diagram
Method/Process	Specific techniques used on a particular project that lead to a specific software product	A product costing system
Methodology	General selection and sequence of techniques capable of producing a range of software products	A range of business software applications

Process Models

Predictive approach

Waterfall

Process Models

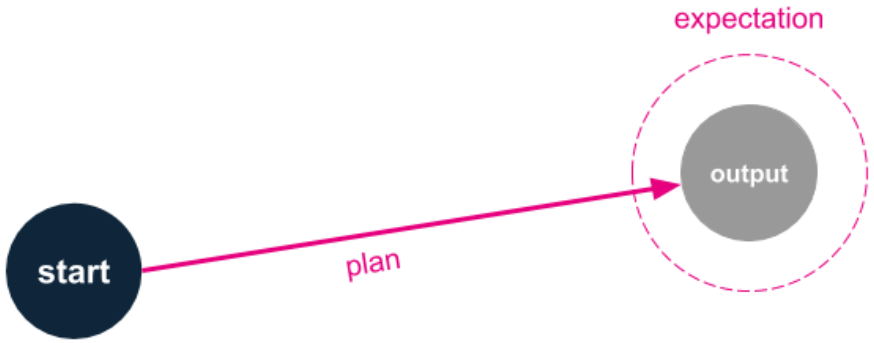
Predictive approach

Waterfall

Process Models

Predictive approach

Waterfall



Process Models

Predictive approach

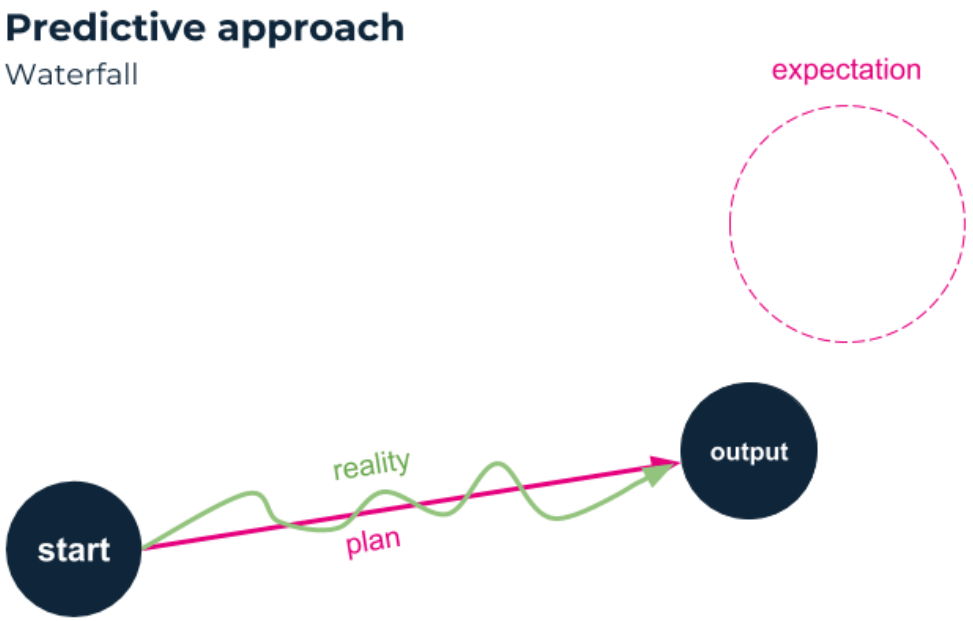
Waterfall

Process Models

Predictive approach

Waterfall

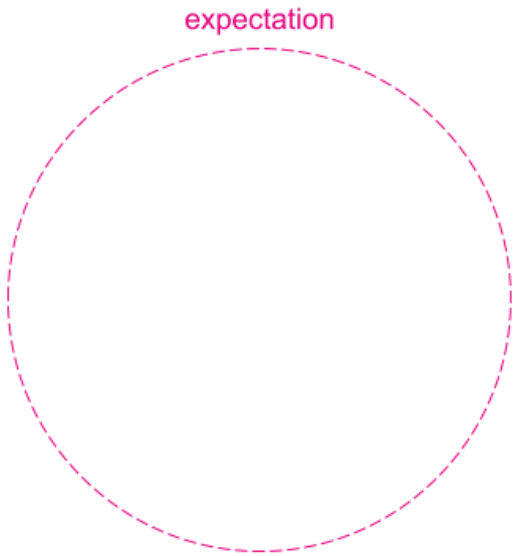
Process Models



Process Models

Adaptive approach

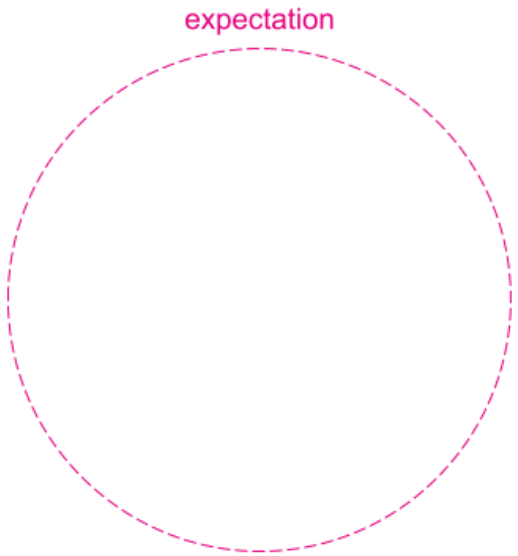
Agile



Process Models

Adaptive approach

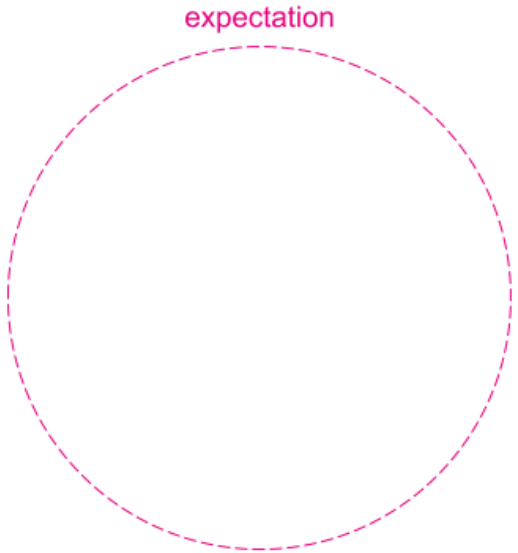
Agile



Process Models

Adaptive approach

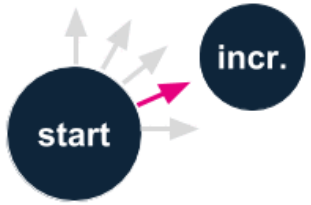
Agile

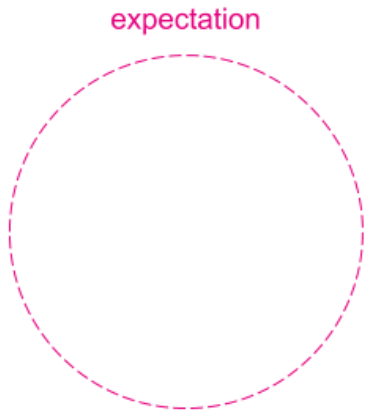


Process Models

Adaptive approach

Agile

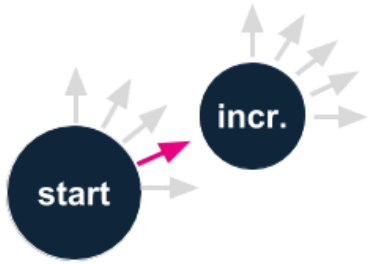


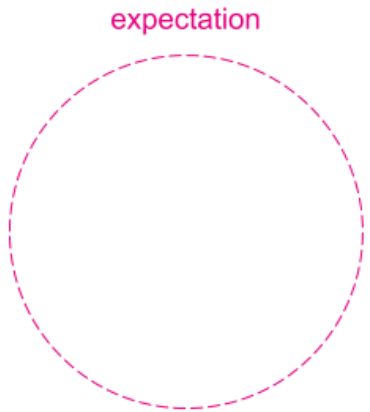


Process Models

Adaptive approach

Agile

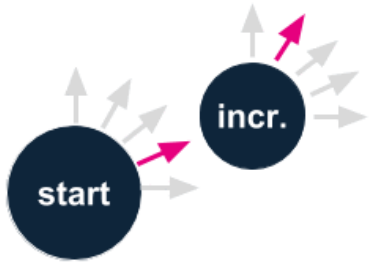


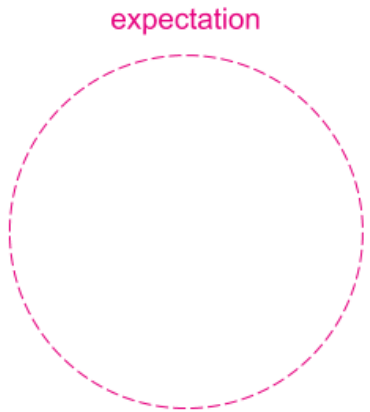


Process Models

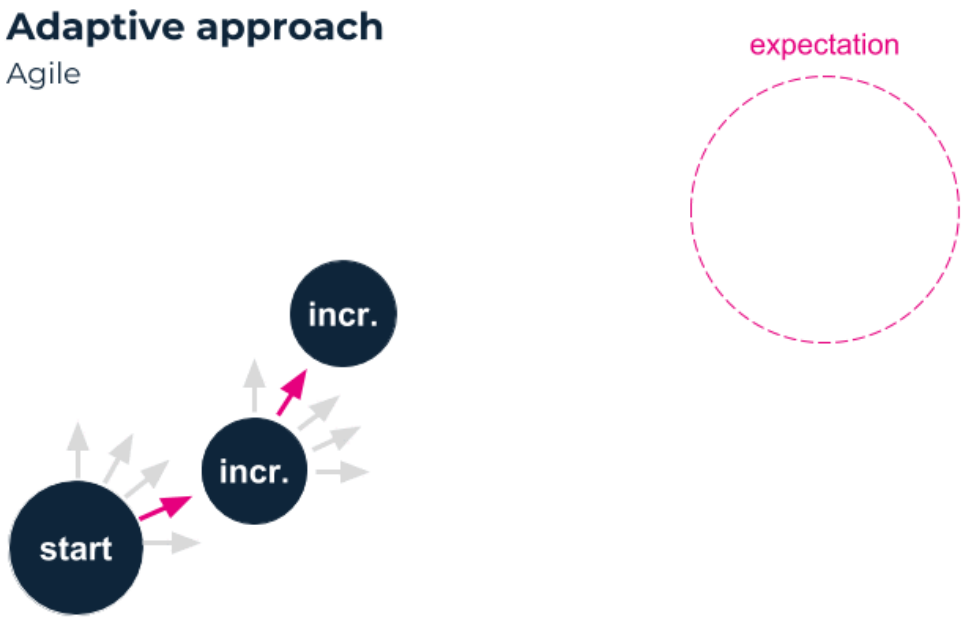
Adaptive approach

Agile

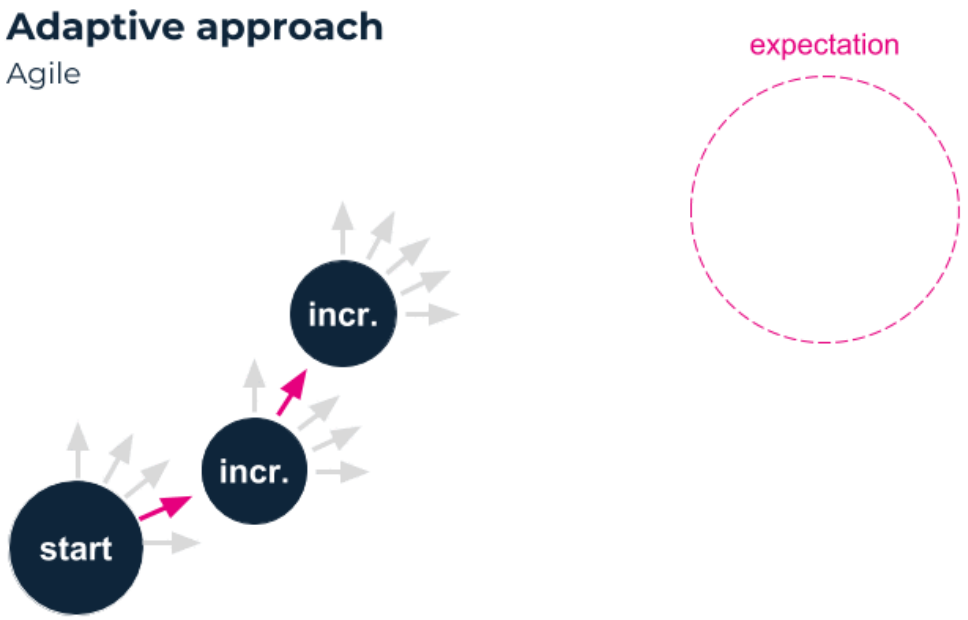




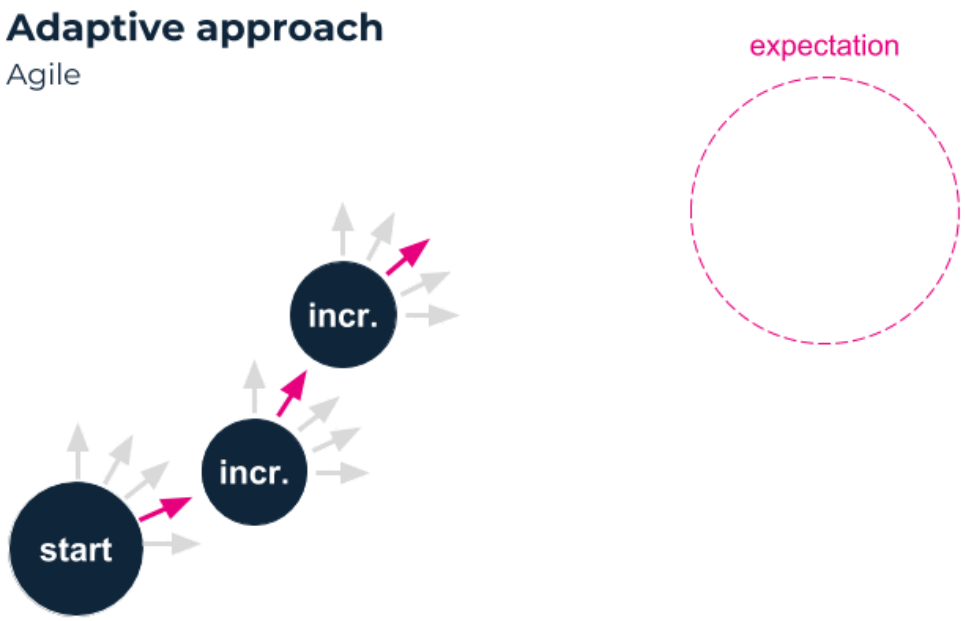
Process Models



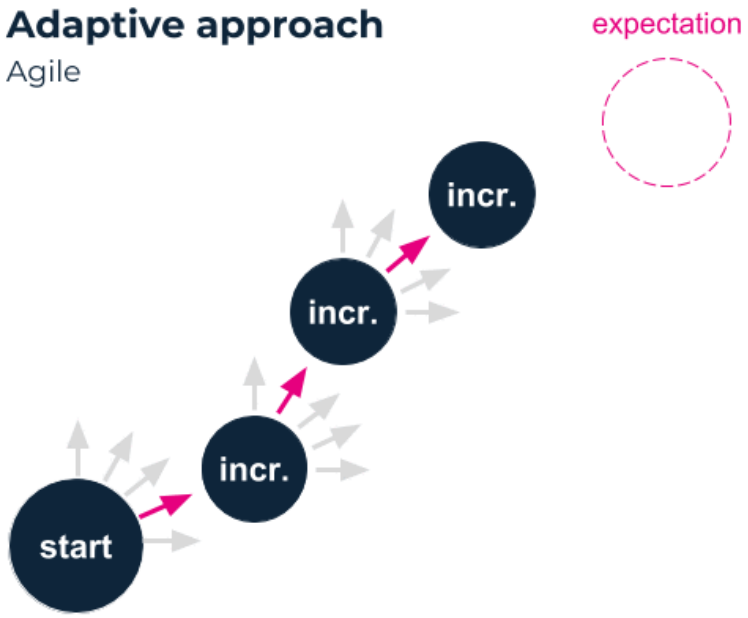
Process Models



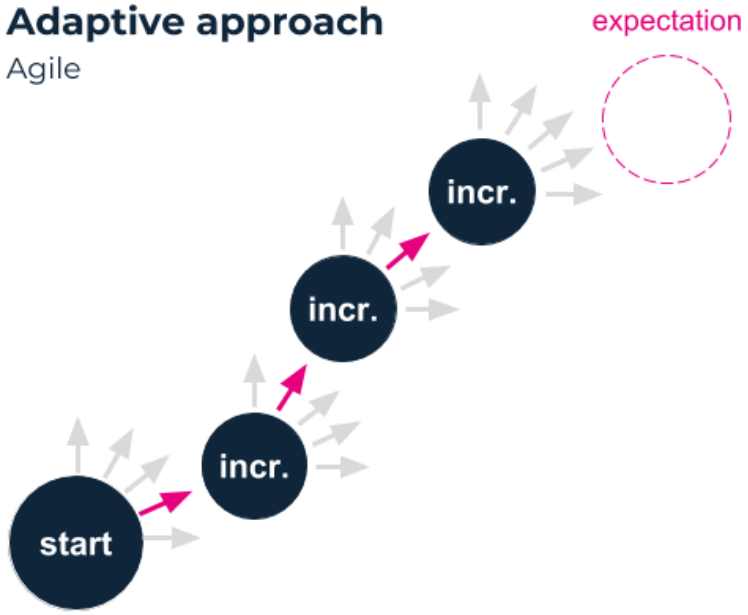
Process Models



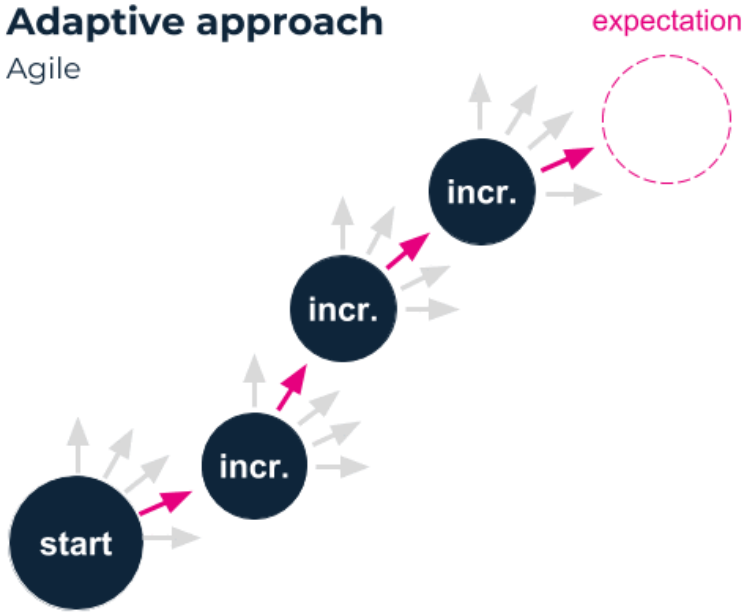
Process Models



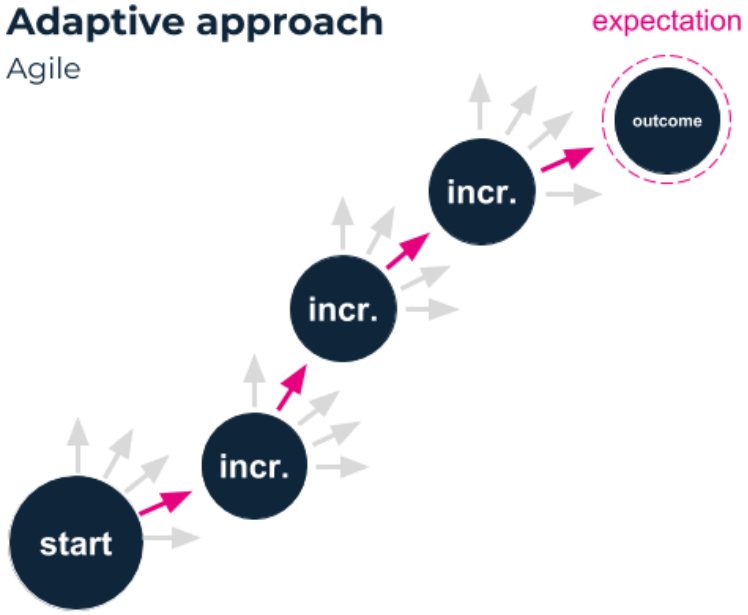
Process Models



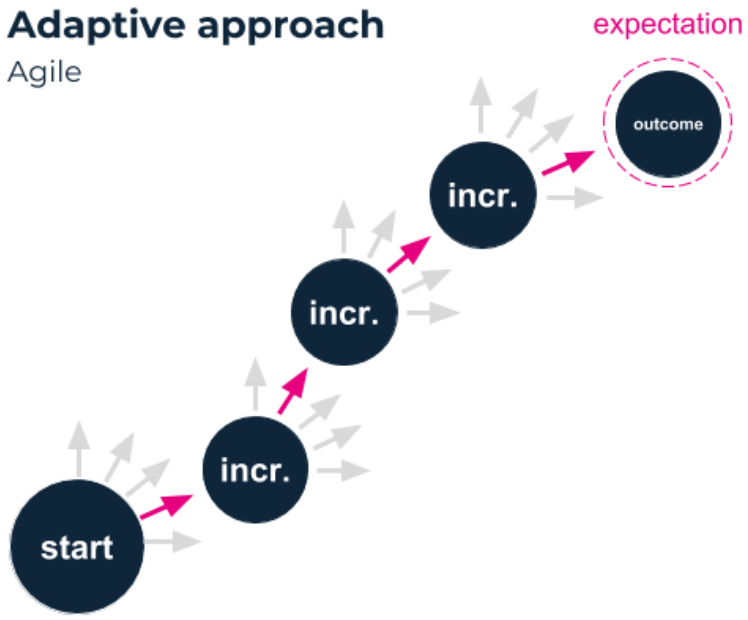
Process Models



Process Models



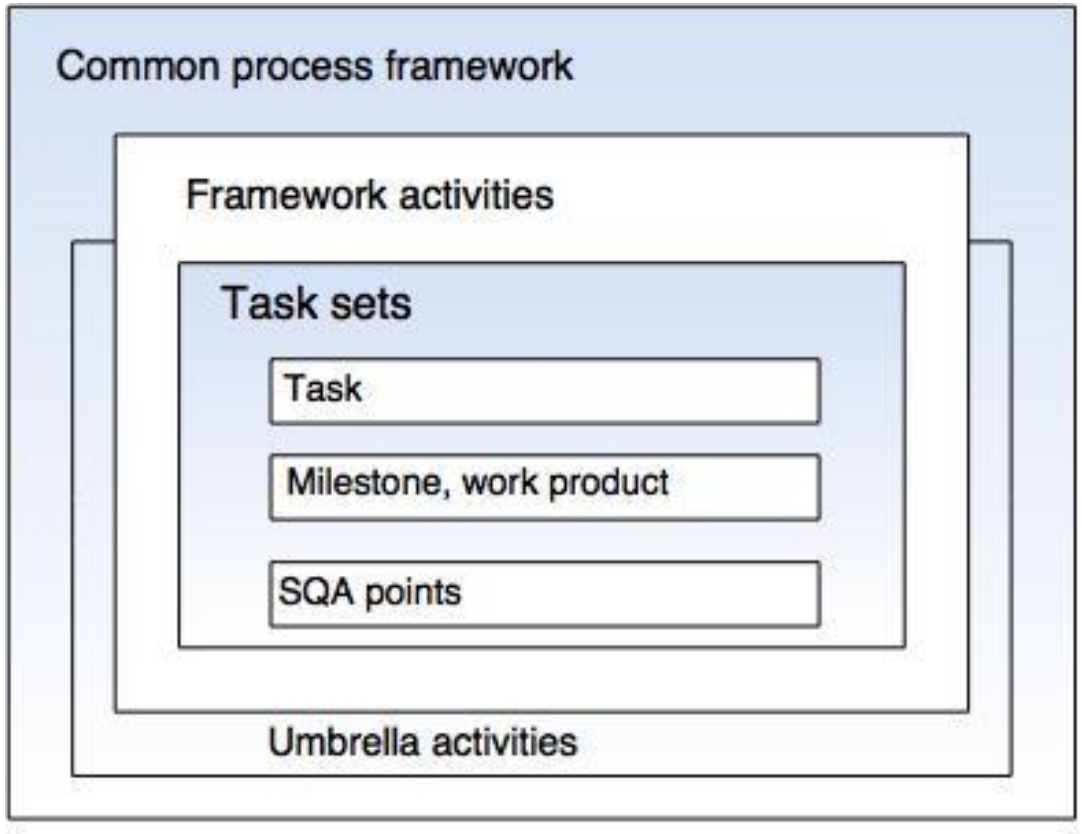
Process Models



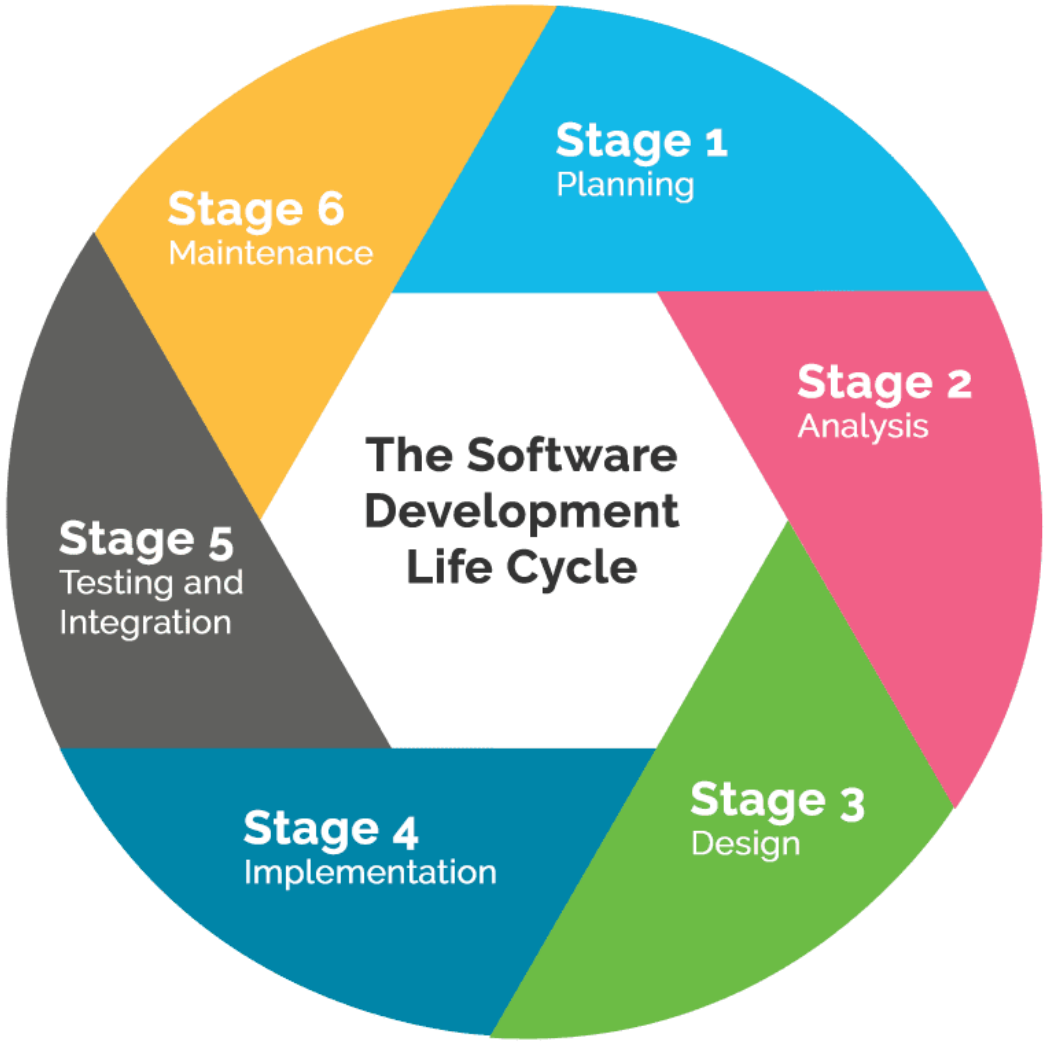
Iterative development

Incremental delivery

A Software Process Framework

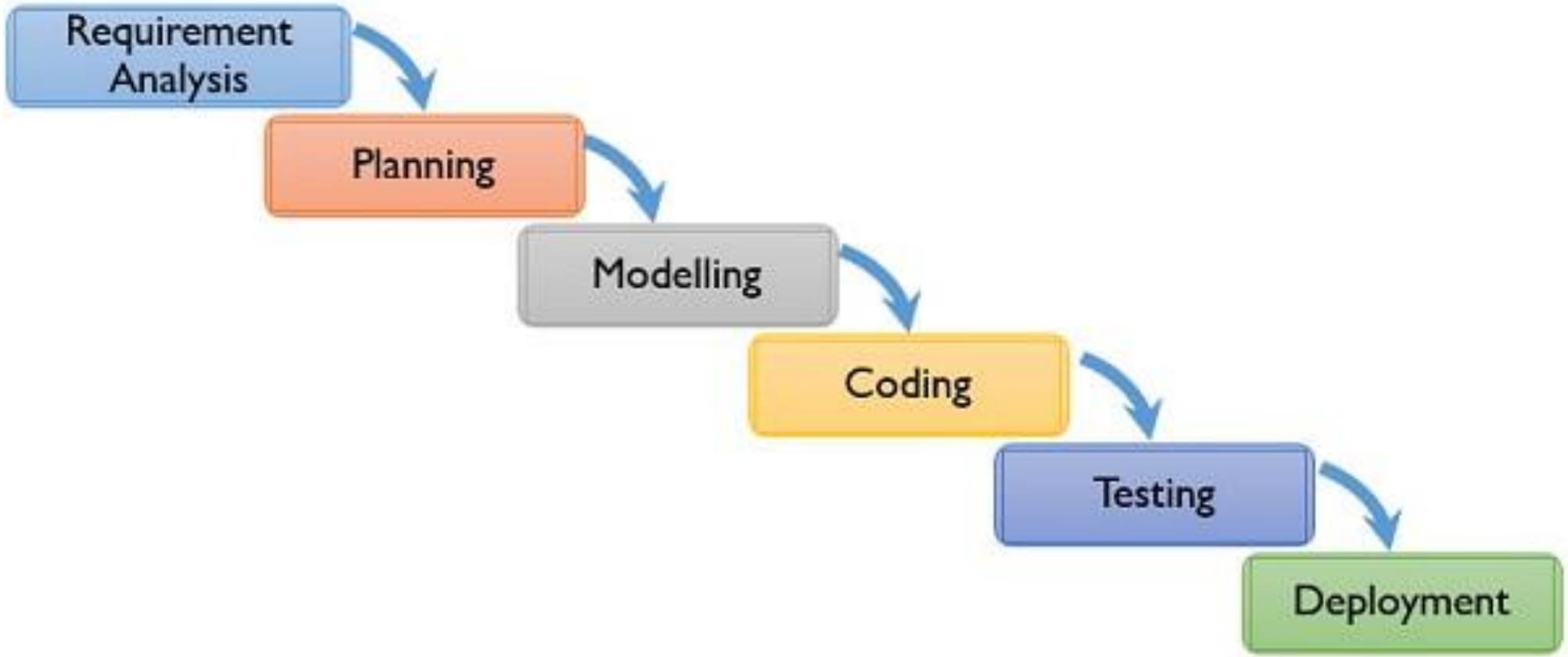


Framework Activity

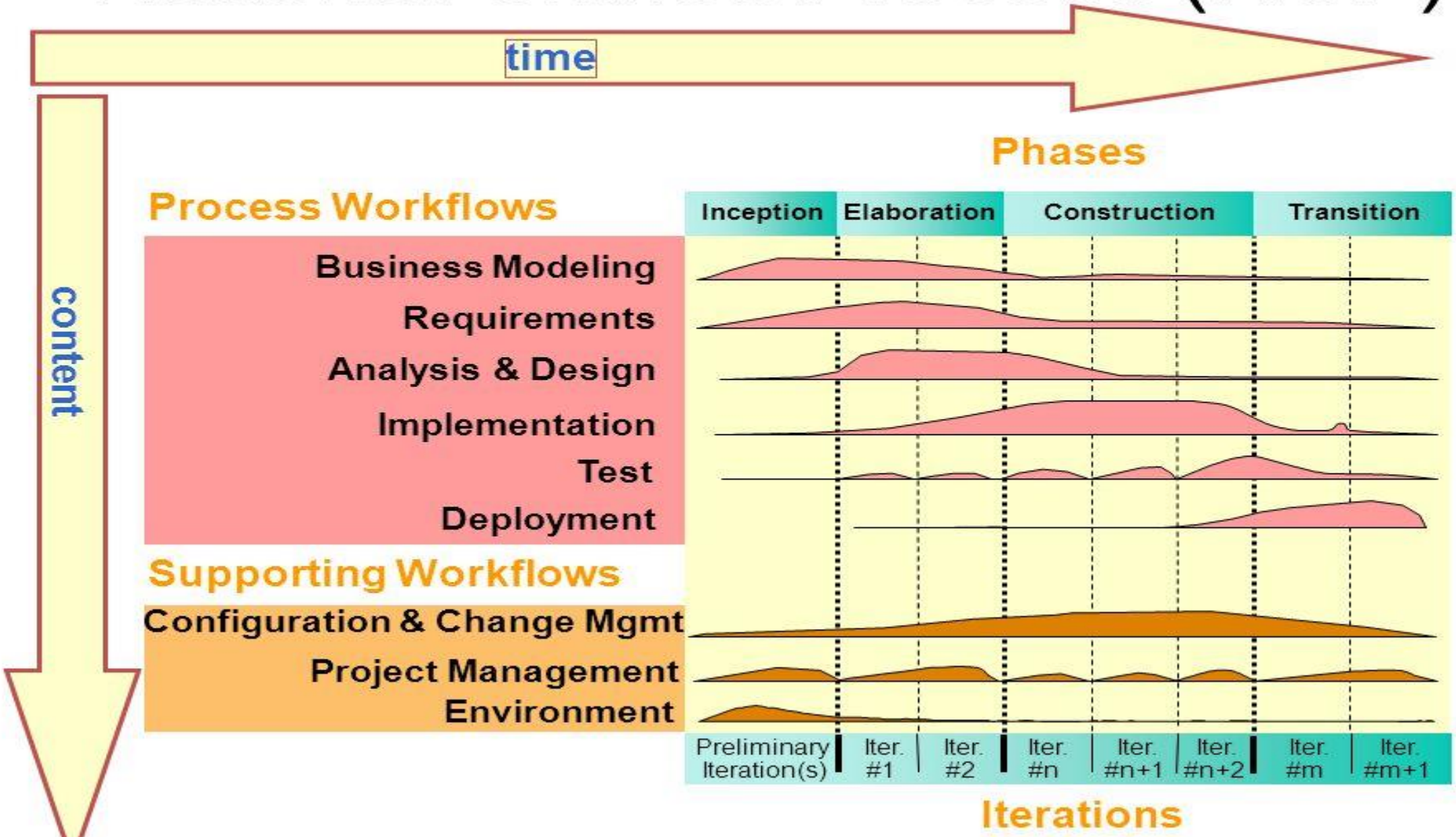


Umbrella Activities

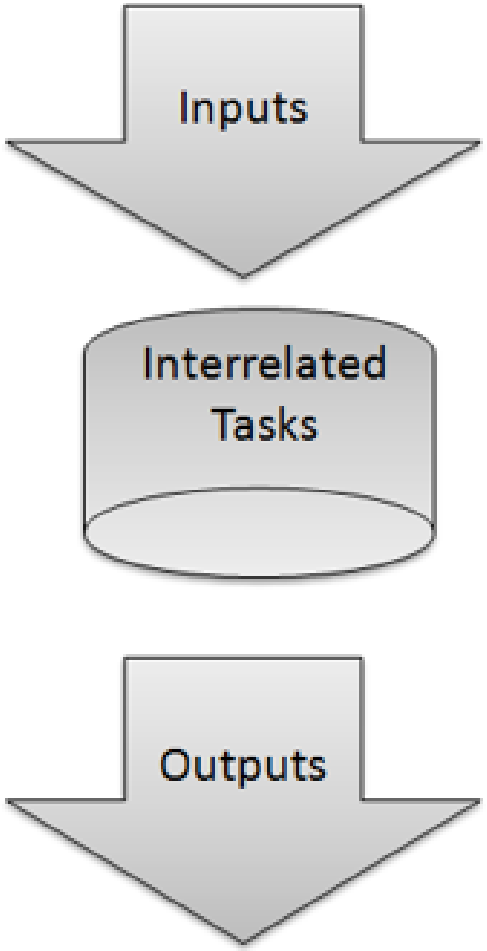
Waterfall


Requirement Analysis
Planning
Modelling
Coding
Testing
Deployment

Rational Unified Process (RUP)



Process



Methodology

- Processes
- Tools
- Techniques
- Principles
- Practices

E.g. XP

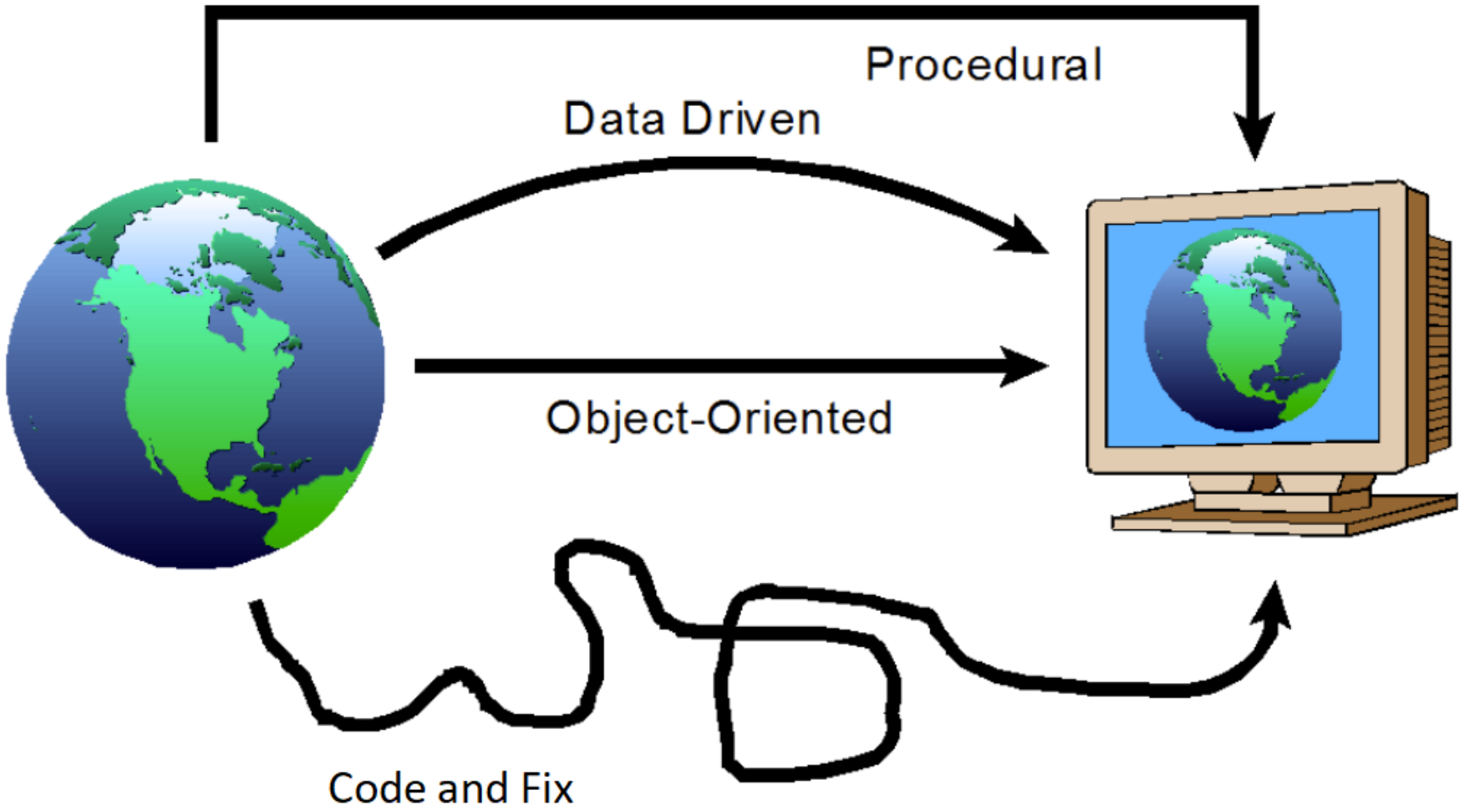
Framework

An Essential Supporting Structure

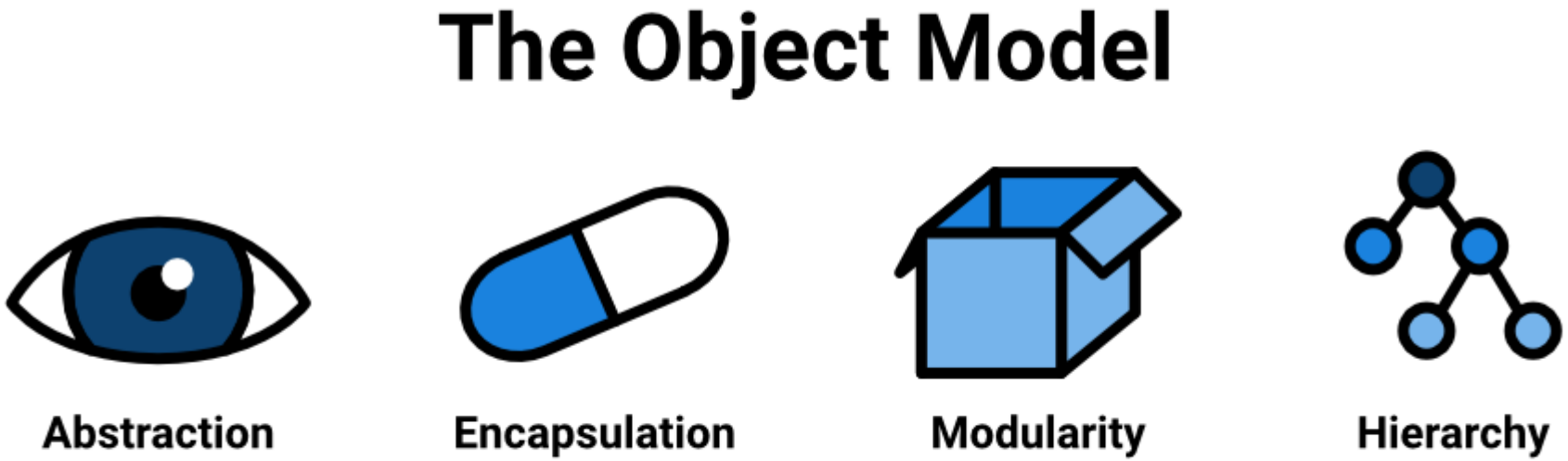
E.g. Scrum

- Scrum – Artifacts, Roles, Events, Activity and few rules
- You can derive Tools and Techniques from Methodologies such as XP

Software Development Paradigms

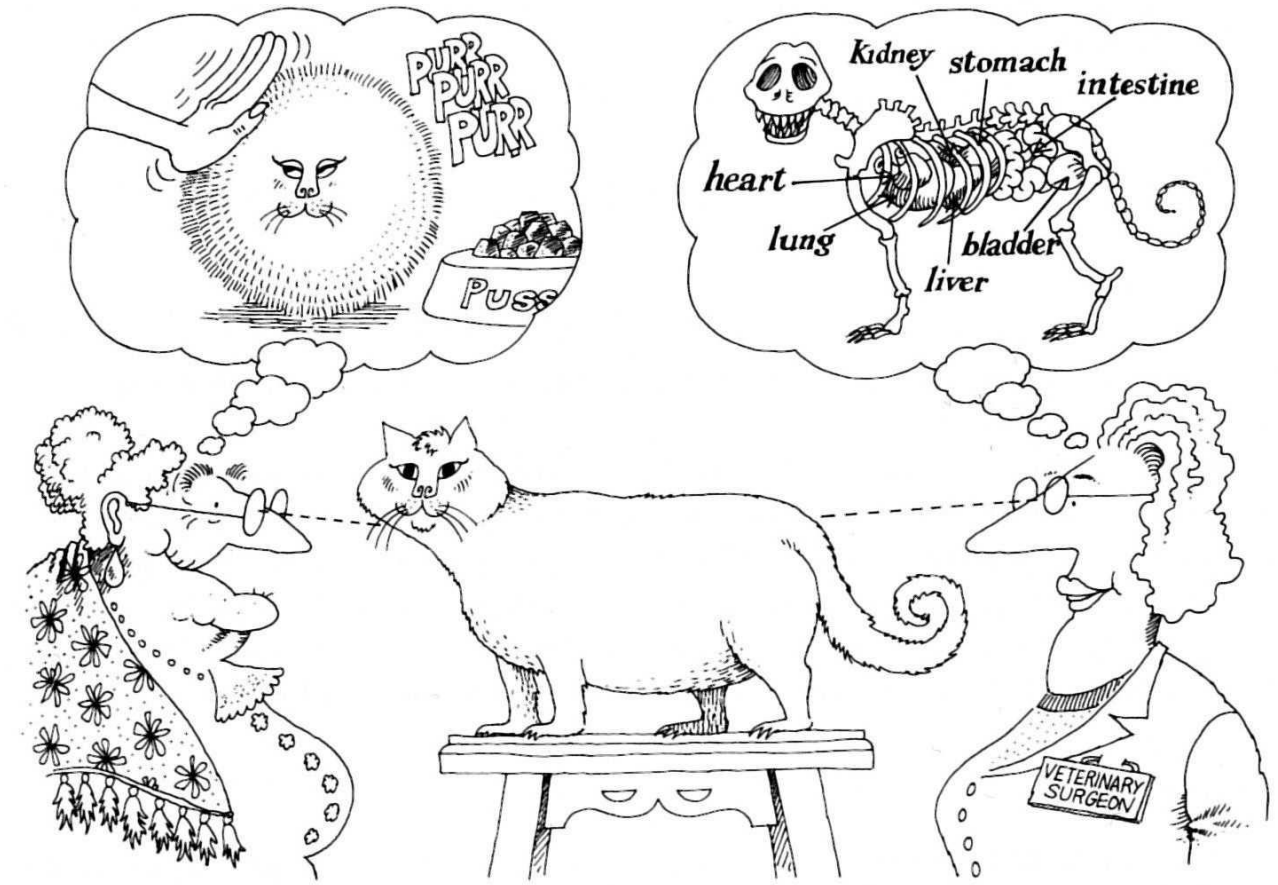


BOOCH' OBJECT MODEL



Abstraction

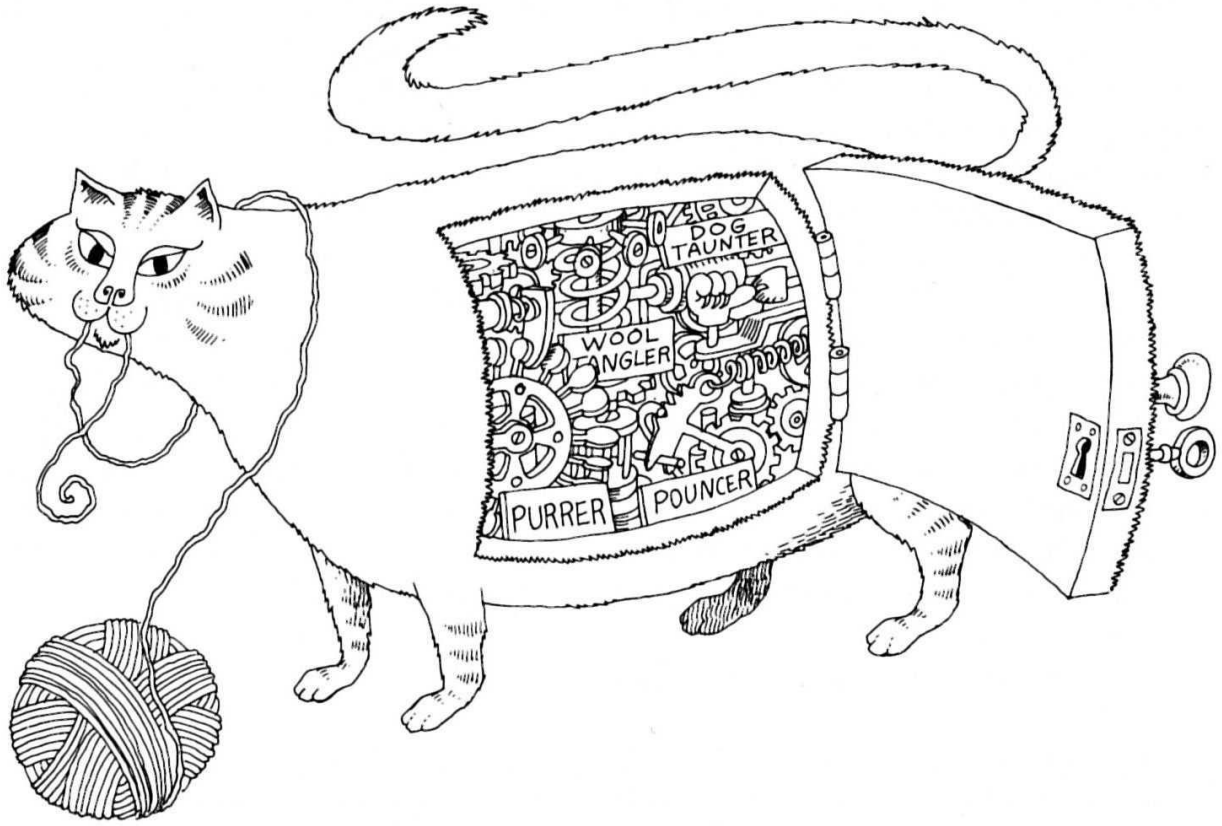
- Helps to deal with complexity by focusing on certain features and suppressing others.
- Focus on interface (outside view)
- Separate behaviour from implementation



Abstraction focuses upon the essential characteristics of some object, relative to the perspective of the viewer.

Encapsulation

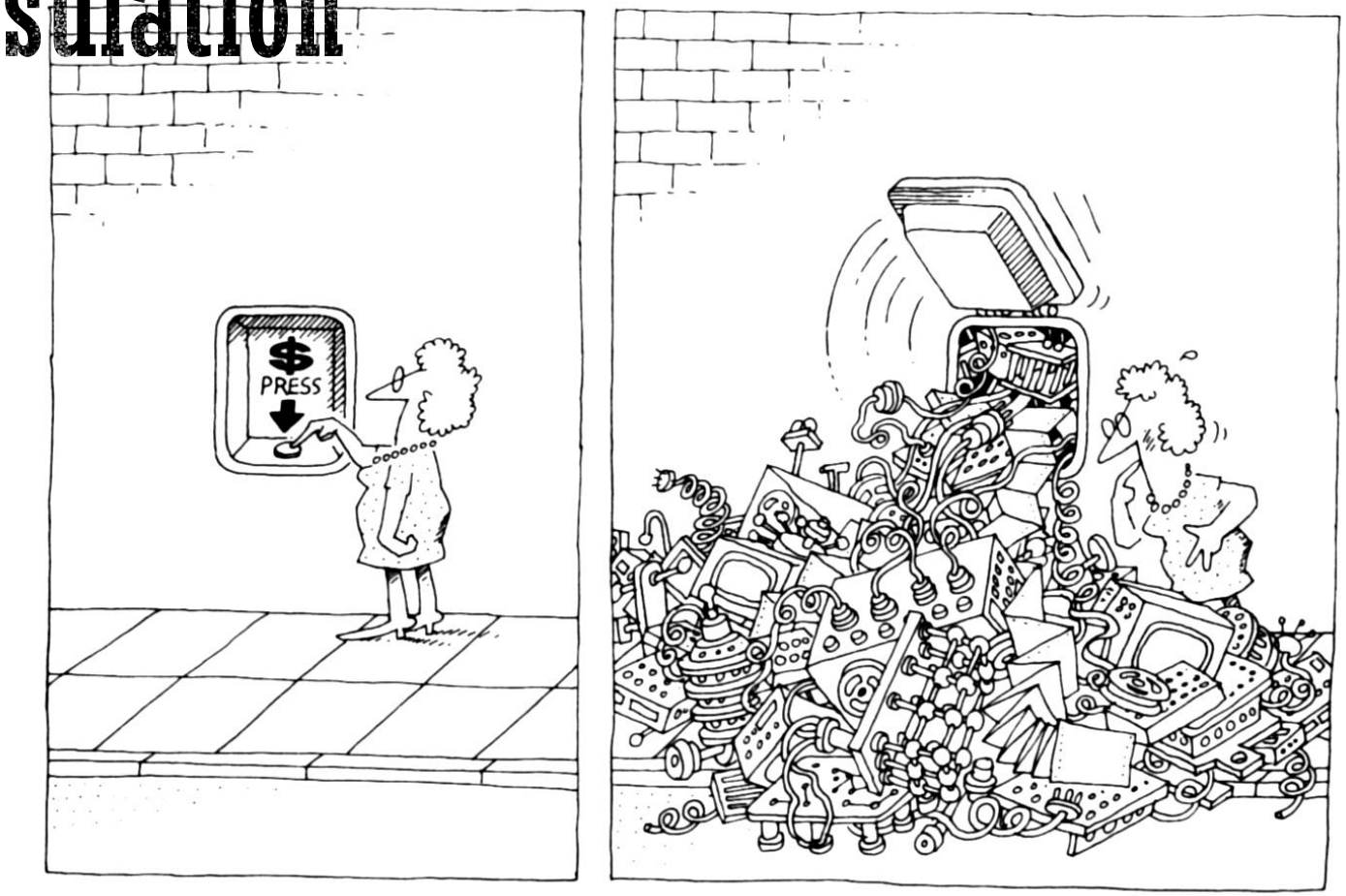
- Also known as information hiding
- Hides the details of the implementation
- Complementary to abstraction



Encapsulation hides the details of the implementation of an object.

Abstraction, Encapsulation

- Interface should be simple providing the required behaviour.
- User is presented with high level abstract view. The detail of the implementation hidden from user.
- The designer may change the implementation keeping interface the same.



The task of the software development team is to engineer the illusion of simplicity.

Modularity

- A common "Divide and conquer" approach
- Partitions a problem into sub-problems reduced complexity
- Modularity packages abstractions into discrete units
- In Java classes are the basic modules providing encapsulation and abstraction

Modularity packages abstractions into discrete units.

Hierarchy

- A way of ordering abstractions
- Object hierarchical abstractions ("HAS A" or "PART OF" relationship)
- Interfaces and behaviours at each level
- Higher levels are more abstract

Abstractions form a hierarchy.