SOFTWARE ENGINEERING

Chapter 4: Software Development Methodologies

Part II – Iterative Methodologies

Iterations!

We have been talking about this a lot!, but... what is it?

Iterations

1. A procedural repetition of the same activity (Verstegen Barnard & Pilot, 2006)

2. Iterating is designing and more specifically, understanding what one is designing through actually creating it

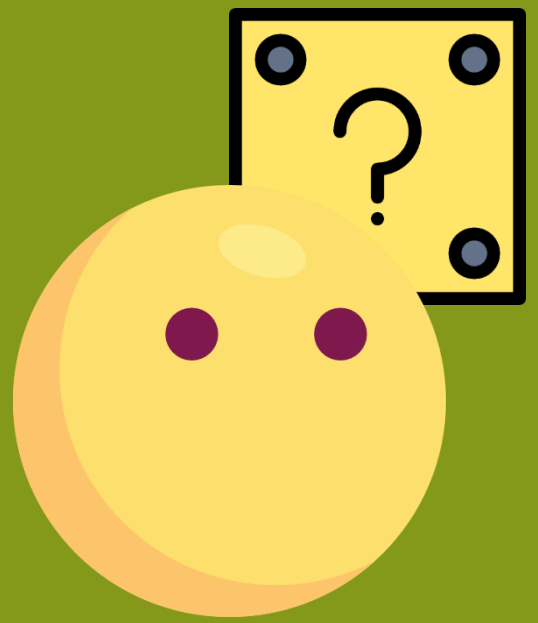
   (Ramsay, 2009)

Iterations

3 Iterations… allow stakeholders to continue to have a means of evaluating decisions and making corrections within project constraints (Allen, 2012)

4 Iterative design is a design methodology based on a cyclical process of idea generation, evaluation and design improvement (Park & Wong, 2010)

So... Can we start?

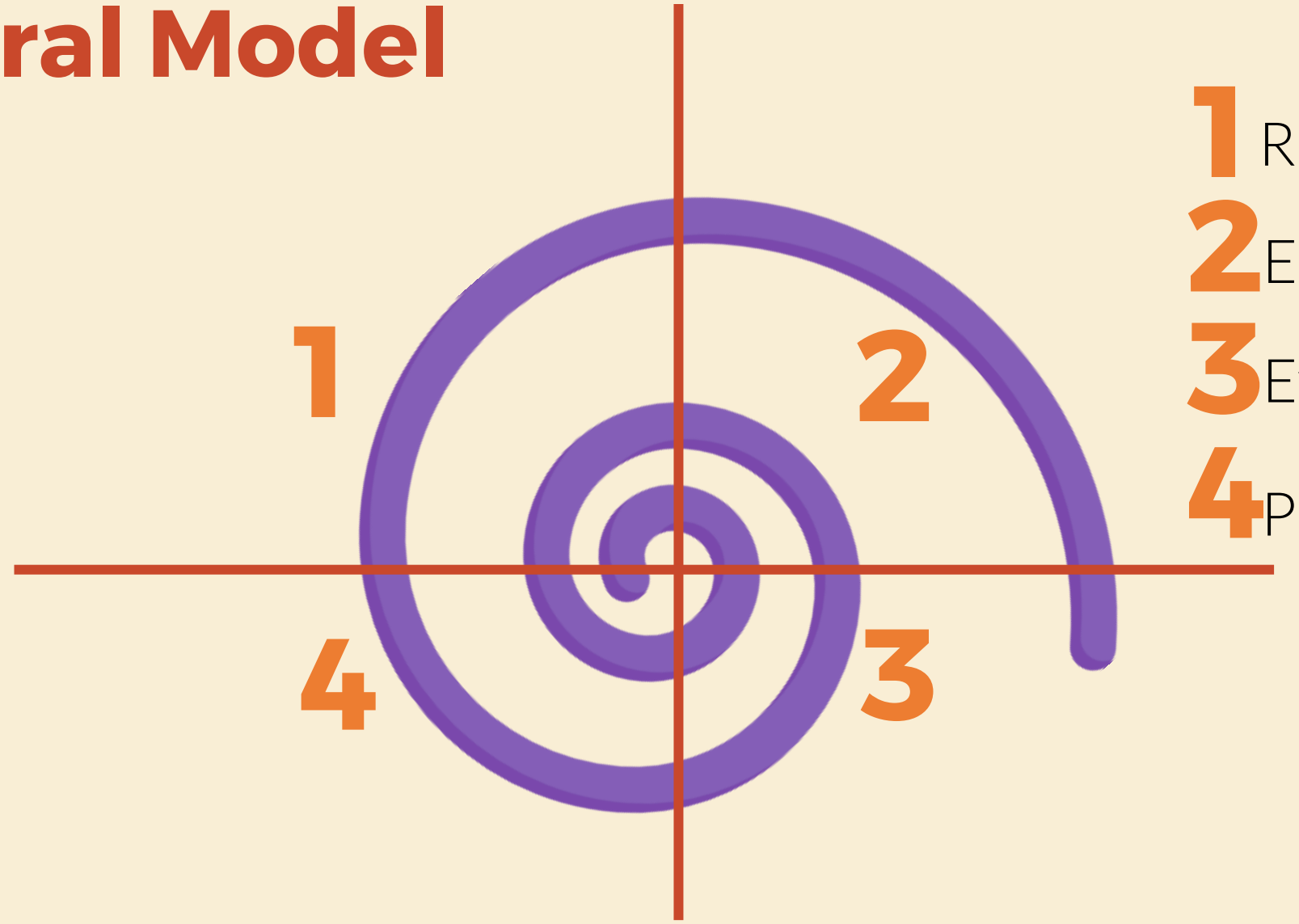


Spiral Model

Literally it’s how it sounds like!,

Take our 5 software processes and start to… iterate!

Spiral Model



1 Risk analysis

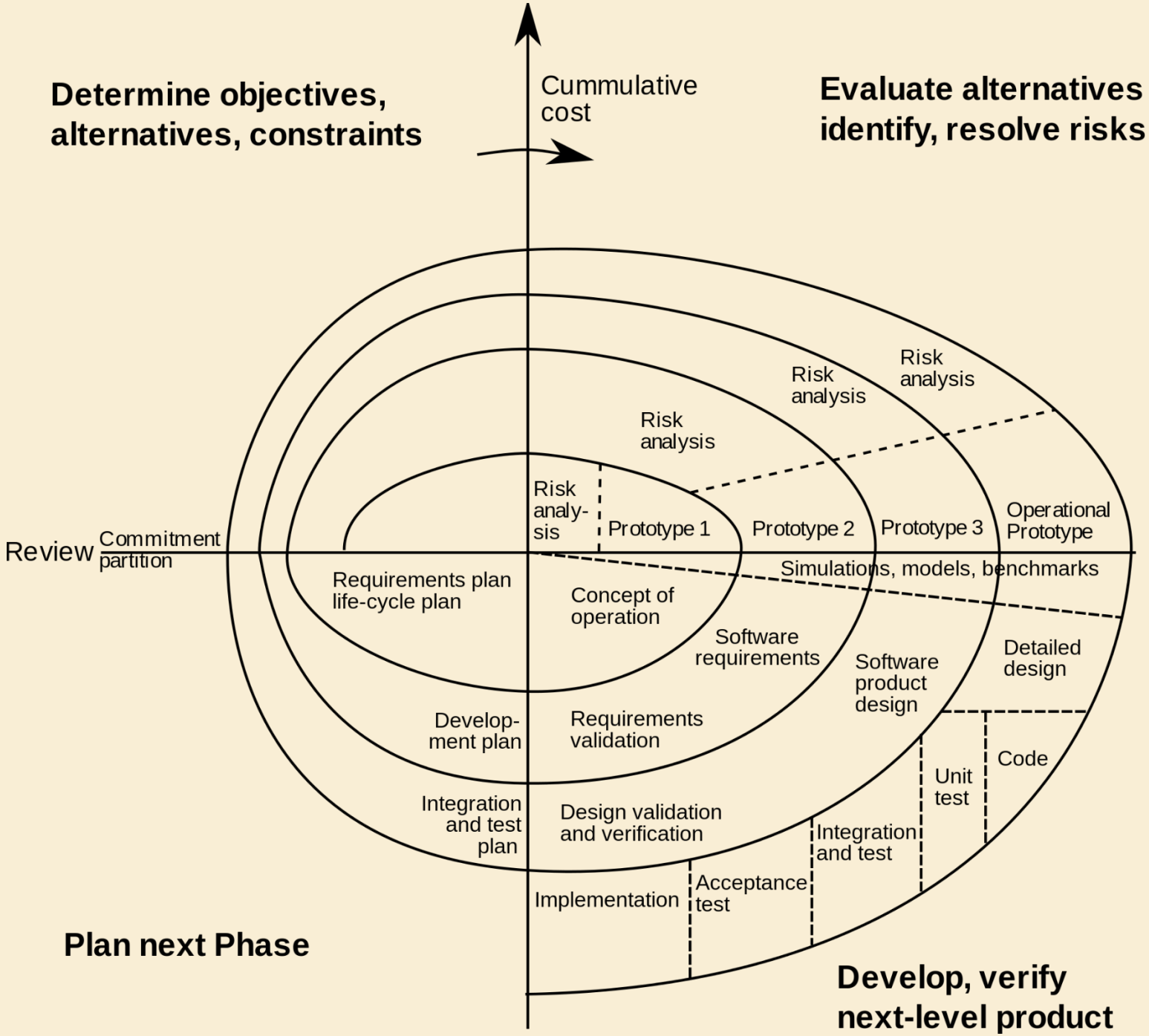
2 Engineering

3 Evaluation

4 Plan

Spiral Model… Classic Version

- Taken from Boehm (1988)
- Each phase is tested in different iterations.
- The advances can be incremental!



Risk analysis example…

Risk analysis: In-house development versus SaaS

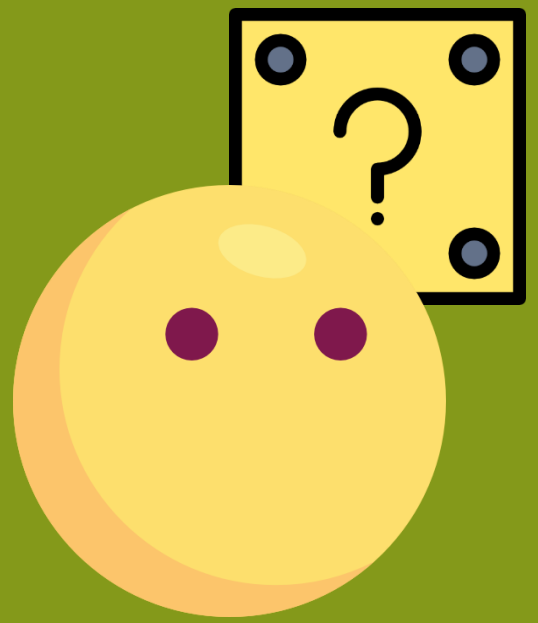
Qualities	In-house development	Software as a Service
Meets the requirements	With enough money; it will almost always meet the requirement	Existing solutions are more difficult to change
Costs	High	Relatively low (thereby: costs are fixed in advanced)
Usability	Teething troubles in the beginning of using the system	Not a solution specific for the problem. This means certain functionalities will miss
Fault resistance	This can be taken into account (in detail) during development	Services are applicable in several situations. Therefore the risk of user errors is higher.
Security	The risk might be higher that security issues go wrong with implementing it by yourself in comparison with conducting a service	The experience of security is may be larger. However, you give someone else the responsibility of the security of your data
Manageability	Updates and such must be carried out by yourself	Support in case of problems
Time	High risk of running out of time during development	Can be used almost immediately

Legend

	There is a high risk this solution will not meet the requirements regarding this quality
	There is a risk this solution will not meet the requirements regarding this quality
	There is a low risk this solution will not meet the requirements regarding this quality

- Taken from Ginkel
- Every single decision can become a risk
- This kind of step tries to mitigate the vast majority of them.

What about RUP?



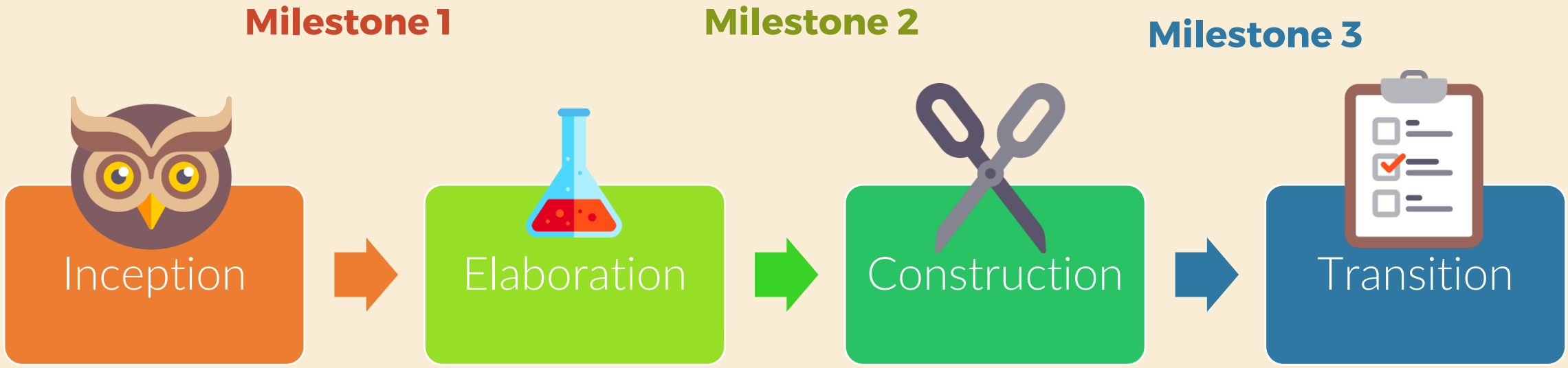
RUP

Rational Unified Process

Formal Definition 

Provides disciplined approach to assigning tasks and responsibilities within a development organization. Its goal is to ensure the production of high-quality software that meets the needs of its end-users, within a predictable schedule and budget

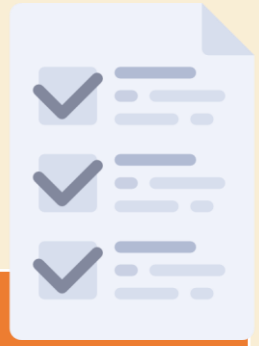
RUP Phases


Milestone 1
Milestone 2
Milestone 3
Inception
Elaboration
Construction
Transition

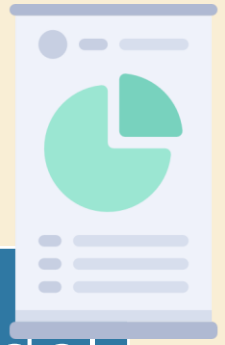
Inception

- Gets the scope of the whole project
- Stablishes the business case for the product!

Inception Deliverables



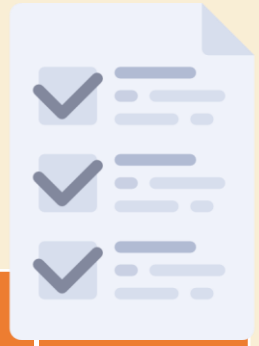
Vision document



Use case model (10-20%)

Domain Model

Inception Deliverables



Initial business case

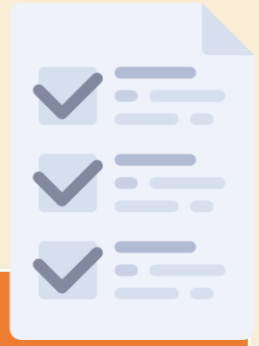
Risk management

First prototype

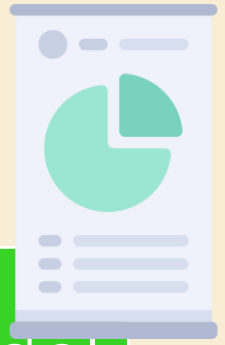
Elaboration

- Stablishes the first architectural foundation
- Non functional requirements are key for this phase, this one is the most critical phase of the project
- This one excels at component based architectures!

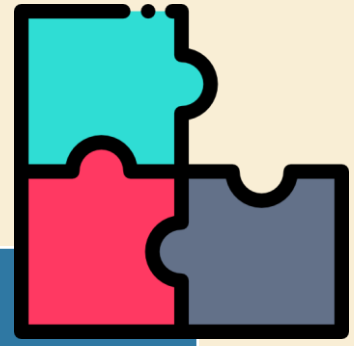
Elabotation Deliverables



Requirements full Scope!

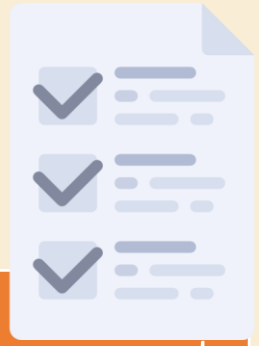


Use case model (80-90%)



Architecture!

Elaboration Deliverables



Development plan

Full Risk Scope

Architectural Prototype

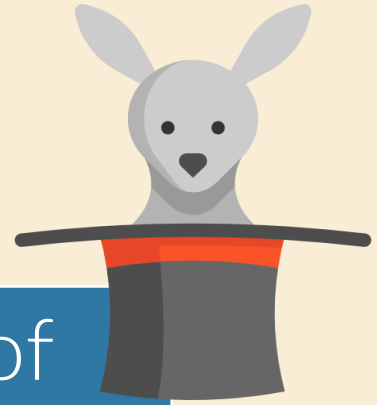
Construction

- Pieces are created and put together!
- Features are developed and tested
- Fine detail or parallel development is also established

Construction Deliverables

1st Beta

Beta testing

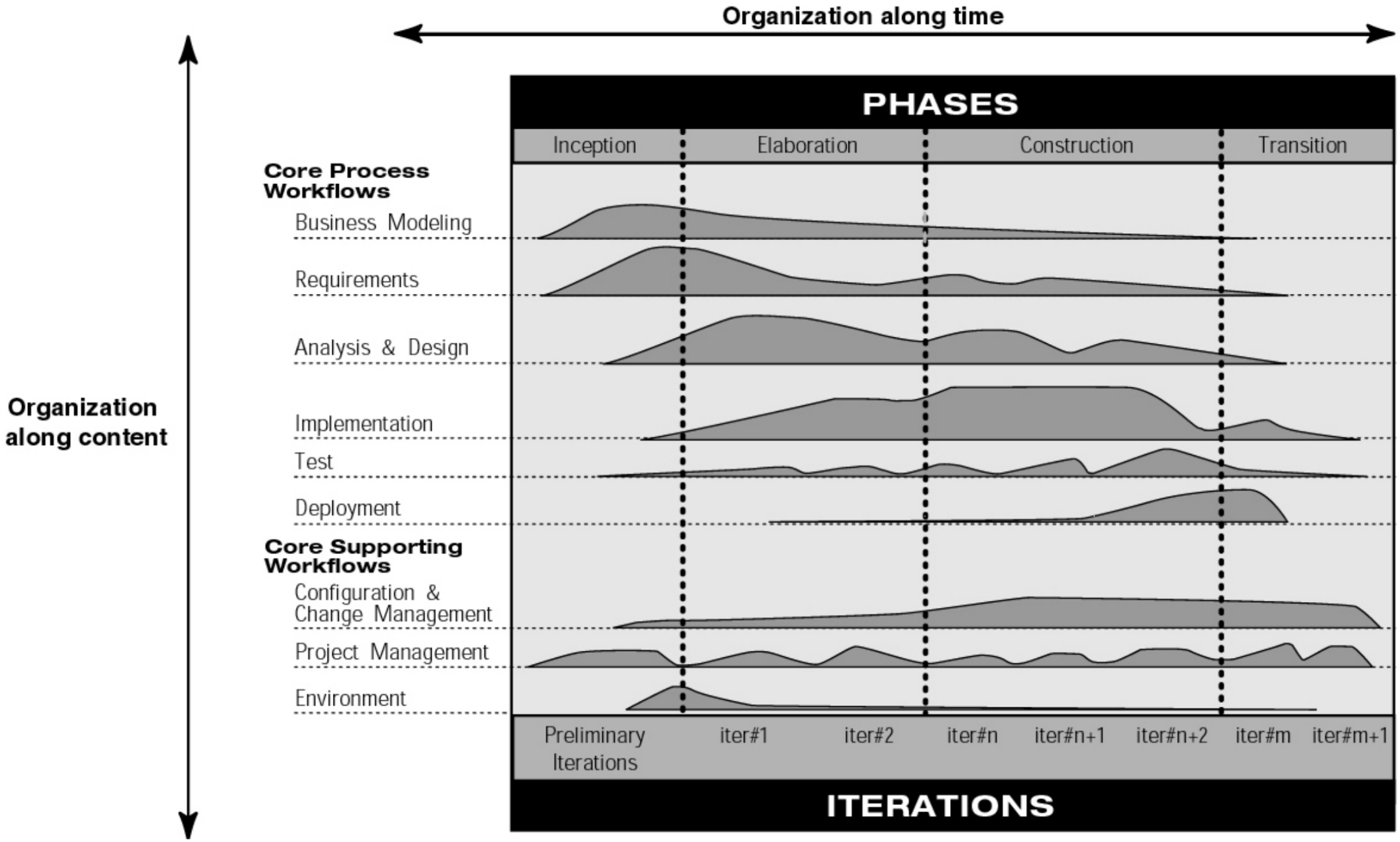


Training of Maintenance team!

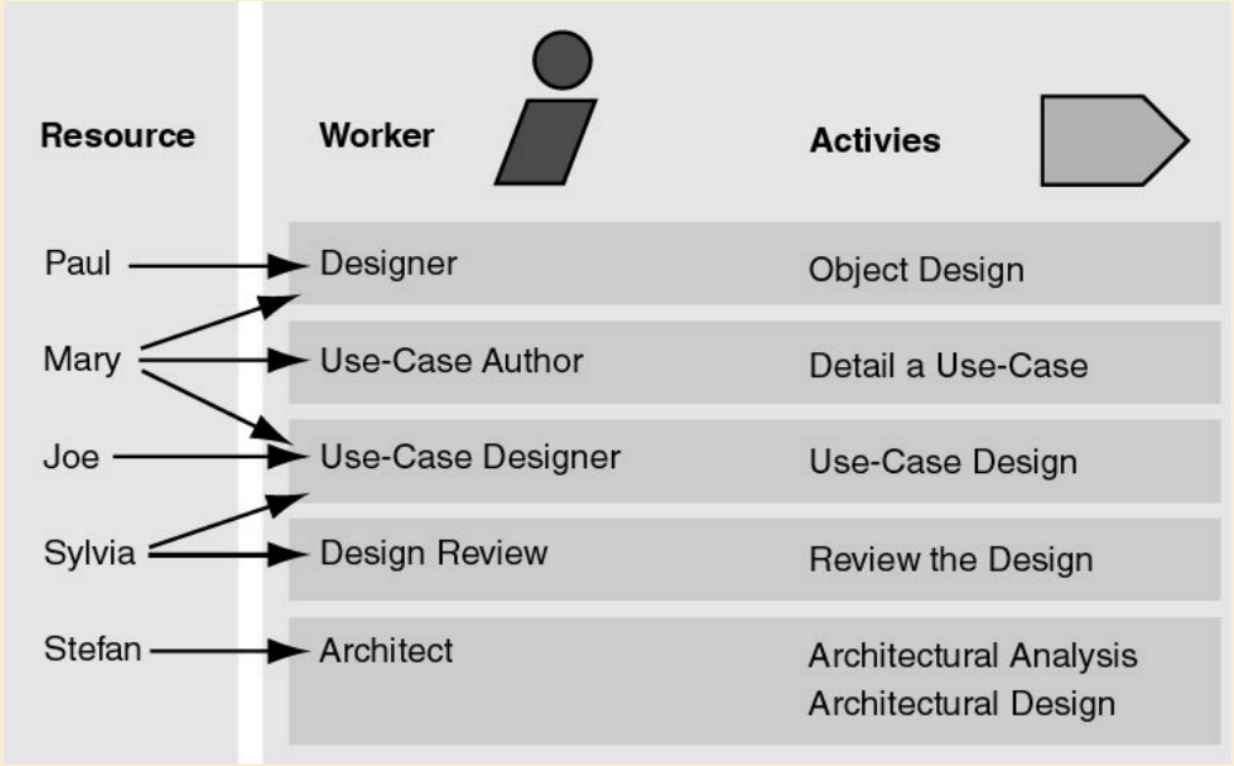
Transition

- The product is ready to be deployed
- Risks are absolutely minimum about the product

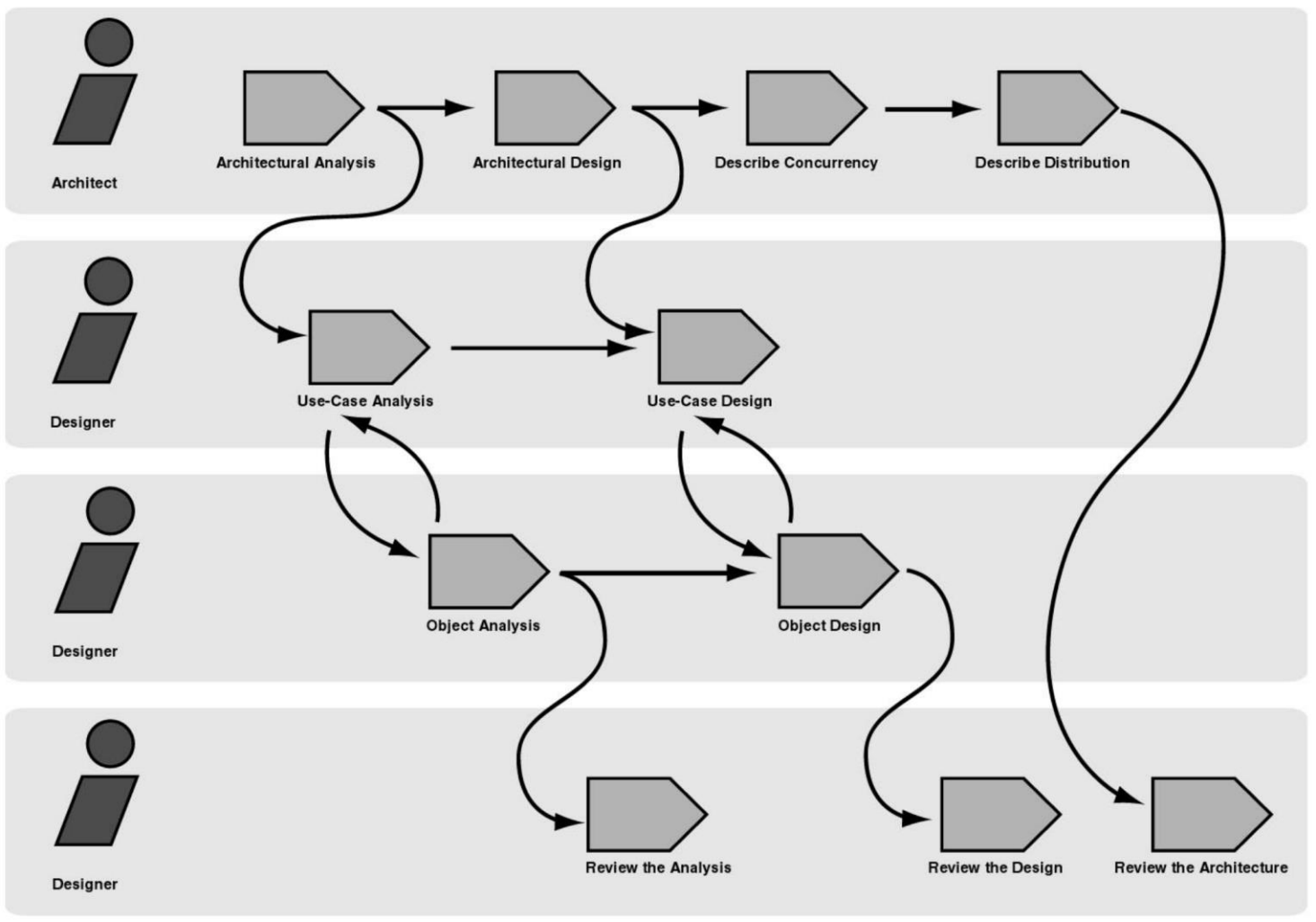
RUP Phases


Organization along time
Organization along content
PHASES
Inception
Elaboration
Construction
Transition
Core Process Workflows
Business Modeling
Requirements
Analysis & Design
Implementation
Test
Deployment
Core Supporting Workflows
Configuration & Change Management
Project Management
Environment
Preliminary Iterations
iter#1
iter#2
iter#n
iter#n+1
iter#n+2
iter#m
iter#m+1
ITERATIONS

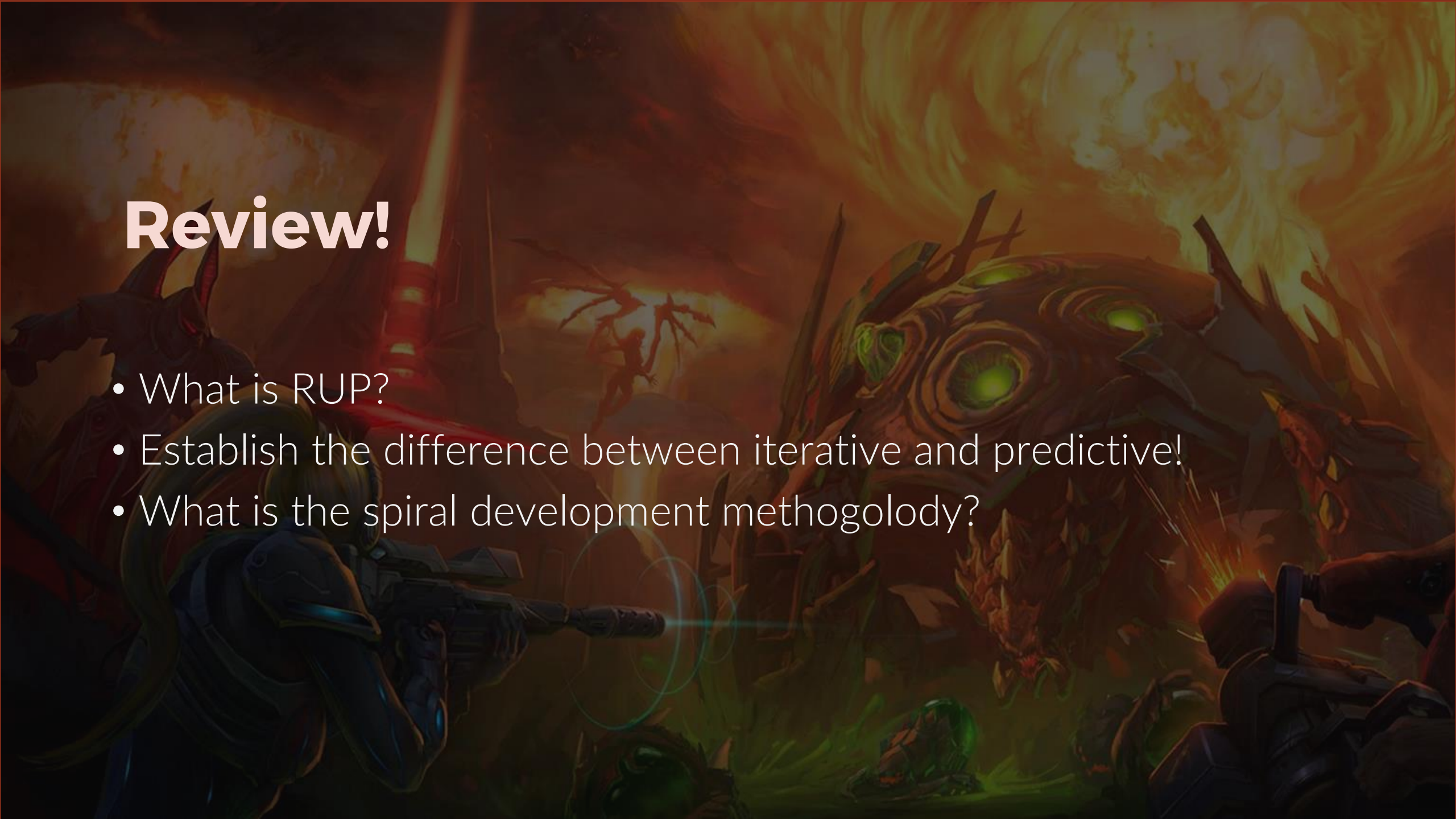
RUP Actions and Roles!



RUP Management Example!




One Shot Review



Review!

- What is RUP?
- Establish the difference between iterative and predictive!
- What is the spiral development methogolody?

References

- [SOMMERVILLE] Ian Sommervile. *Software Engineering 9th Edition*
- [STEPHENS] Beginning Software Engineering. 2015
- [CROOKSHANKS] Software Development Techniques. 2015
- DESPA (2014), Comparative study on software development methodologies Mihai Liviu DESPA. Bucharest University of Economic Studies
- Kruchten, P. (2004). *The rational unified process: an introduction*. Addison-Wesley Professional.
- Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, *21*(5), 61-72.


Class has died... for today!