

47006- ANÁLISE E MODELAÇÃO DE SISTEMAS

OpenUP and the AMS project assignment

Ilídio Oliveira

v2020-10-23 | TP-06

Learning objectives for this lecture

- Describe the structure of the Unified Process (phases, milestone objectives, iterations)
- Identify the key activities required in the project assignment
- Map technical disciplines to the OpenUP phases

THE UNIFIED SOFTWARE DEVELOPMENT PROCESS

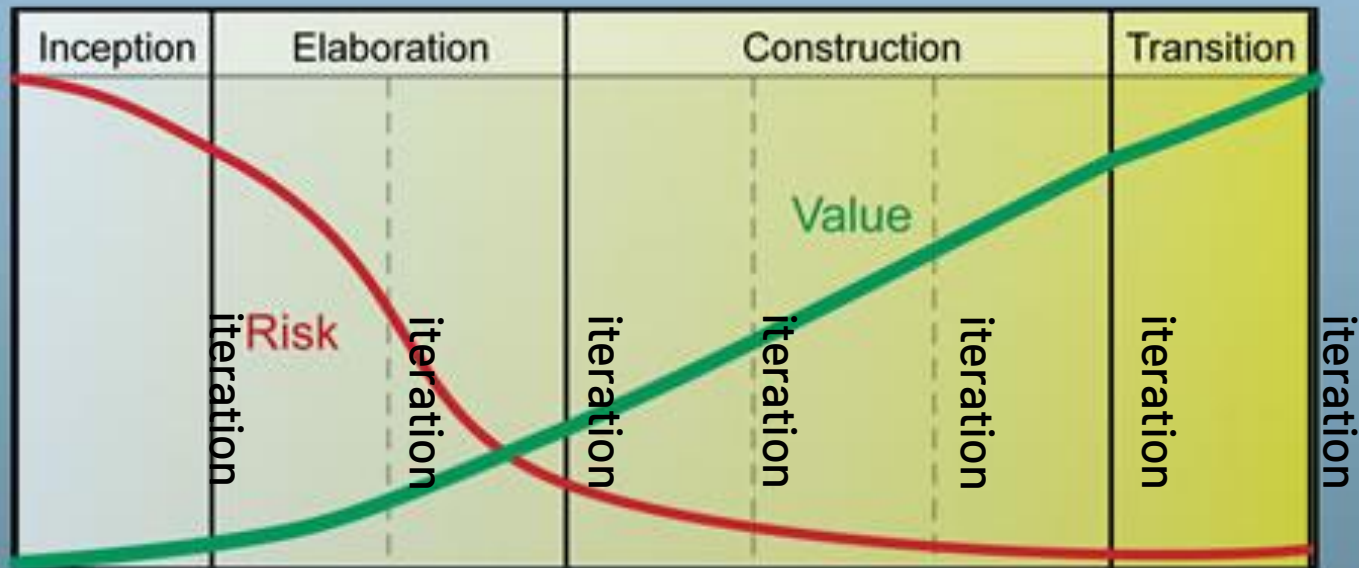
IVAR JACOBSON
GRADY BOOCH
JAMES RUMBAUGH



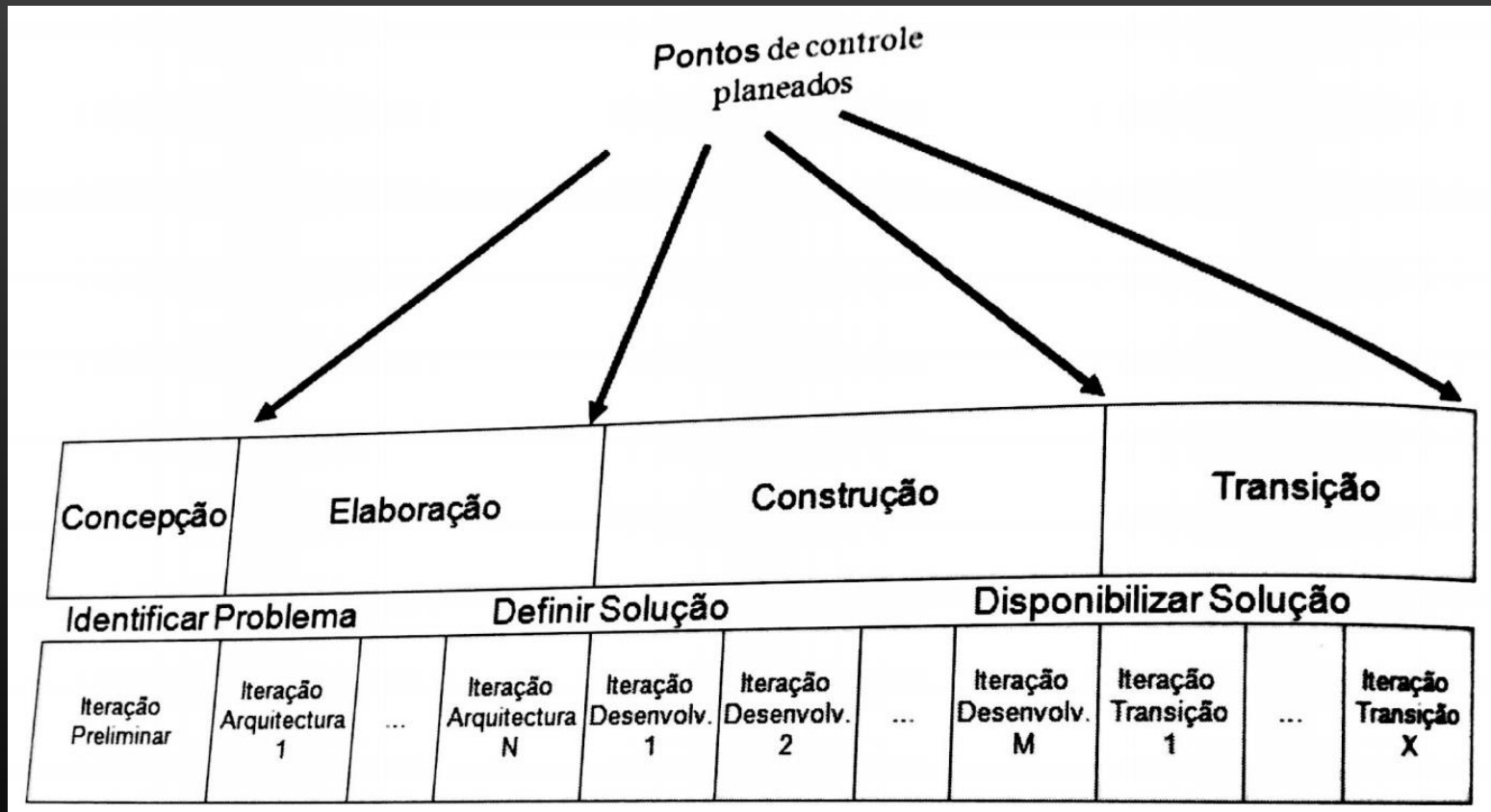
*The complete guide
to the Unified
Process from the
original designers*



Project Lifecycle



PT: Fases, iterações e pontos de controlo



OpenUP/Unified Process activities

The UP offers an approach to the SDLC visualized as a **matrix**, crossing different **technical disciplines** with evolving **iterations** in the project. (Note: UP phases ≠ SDLC phases)

Requirements analysis is mainly performed at the beginning of the project (requirements baseline) but also during the iterations (evolutionary requirements).

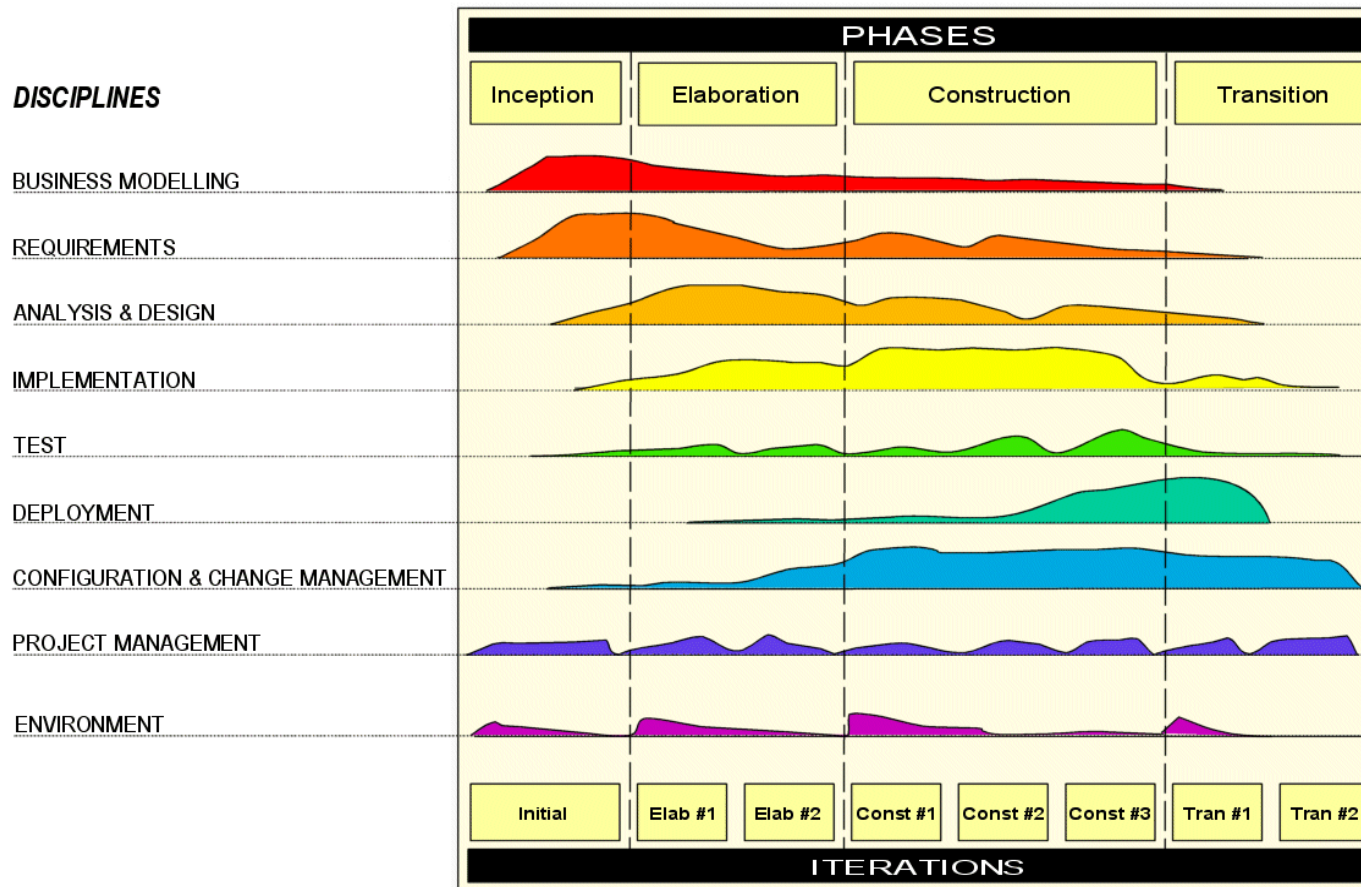
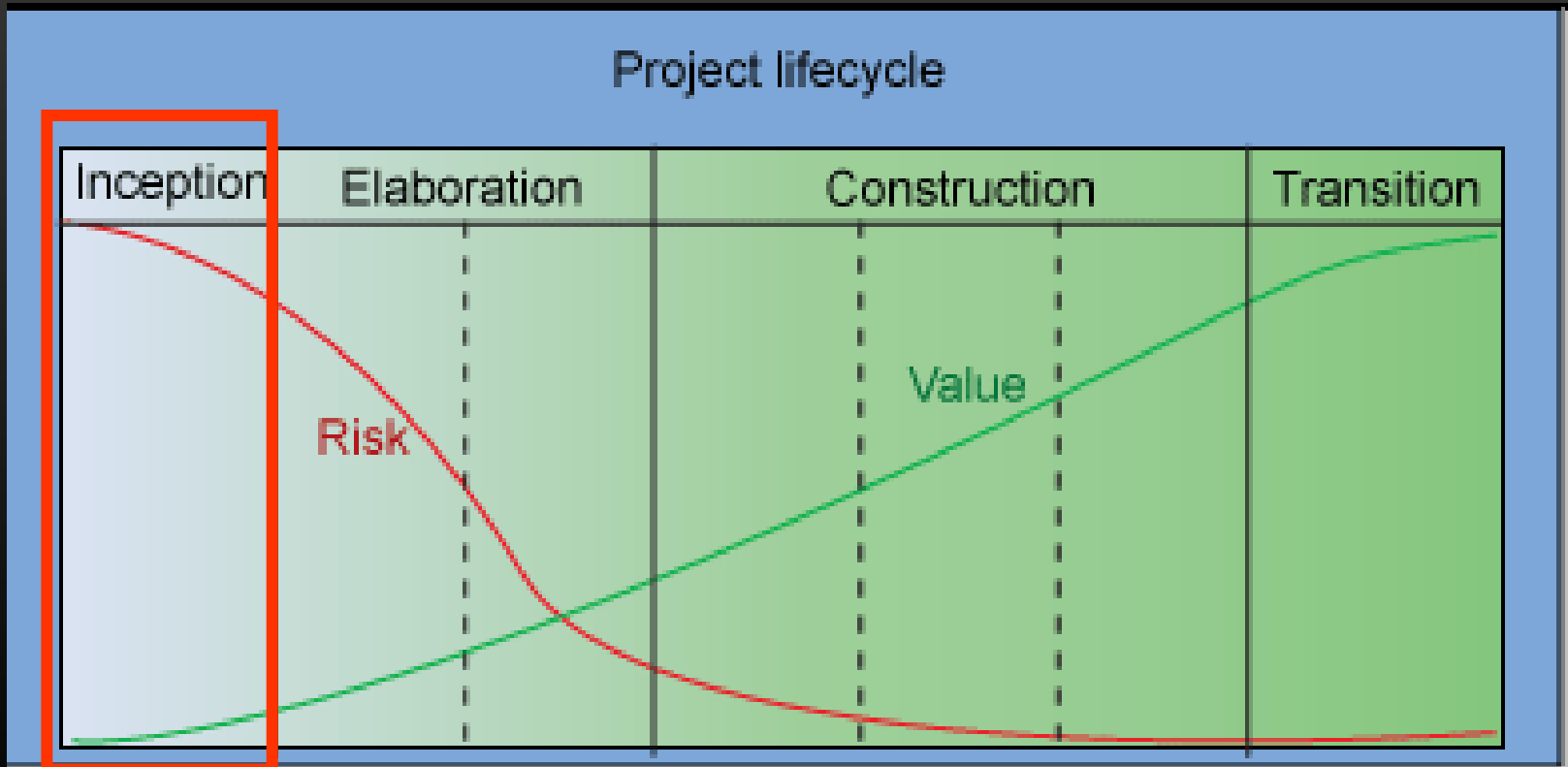


Figura Project lifecycle

The phases: Inception

Do we agree on project scope and objectives, and whether or not the project should proceed?



Inception: Know What to Build

Typically one short iteration

Produce **vision** document and initial business

Develop high-level project requirements

Initial use-case and (optional) domain models (10-20% complete)

Focus on **what is required** to get agreement on 'big picture'

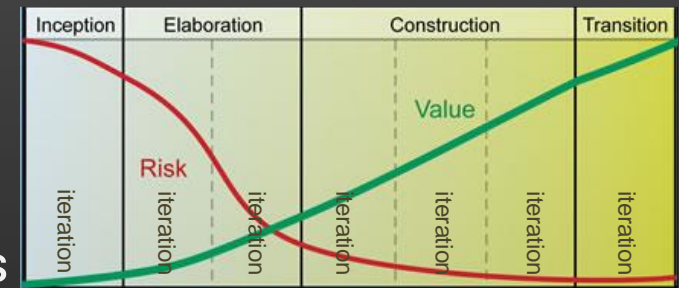
Manage project scope

Reduce risk by identifying key requirements

Acknowledge that requirements will change

Manage change, use iterative process

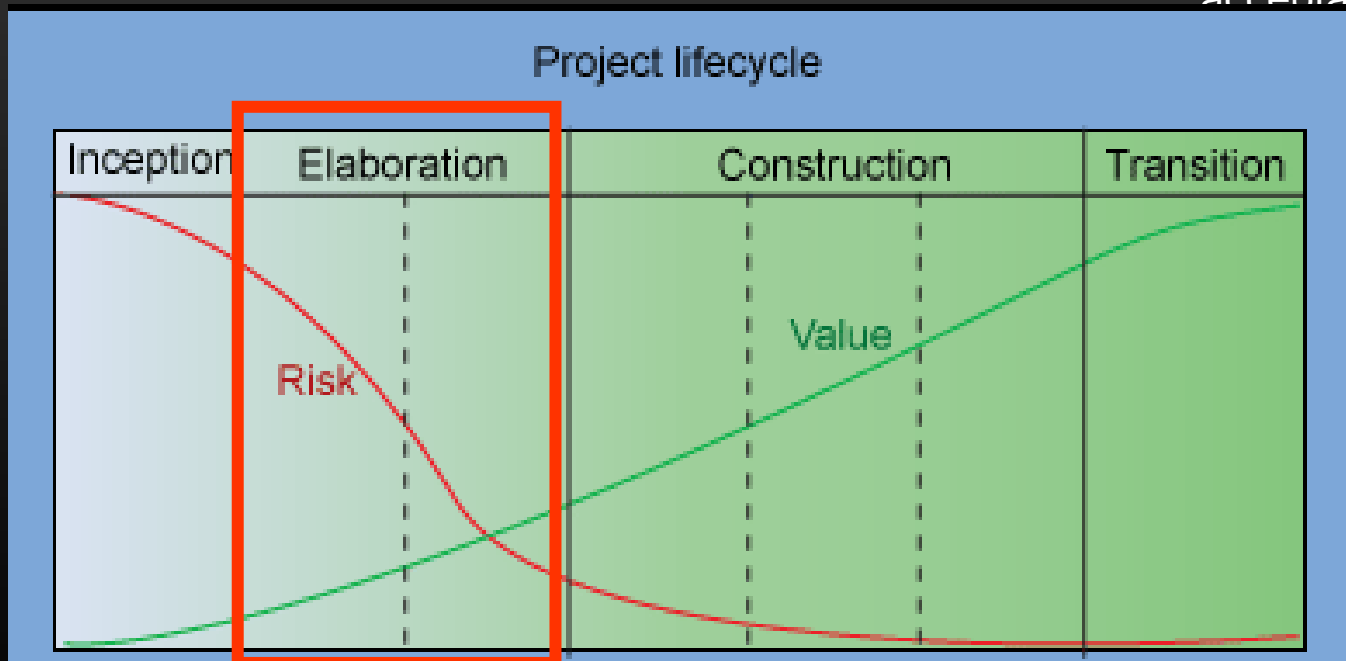
Produce conceptual prototypes as needed



Credit: Per Kroll (IBM)

The phases: elaboration

Do we agree on the executable architecture to be used for developing the application and do we find that the value delivered so far and the remaining risk is acceptable?



Elaboration: Know How to Build It by Building Some

Elaboration can be a day long or several iterations

Balance

mitigating key technical and business risks with producing value (tested code)

Produce (and validate) an executable architecture

Define, implement and test interfaces of major components. Partially implement some key components.

Identify dependencies on external components and systems. Integrate shells/proxies of them.

Roughly 10% of code is implemented.

Drive architecture with key use cases

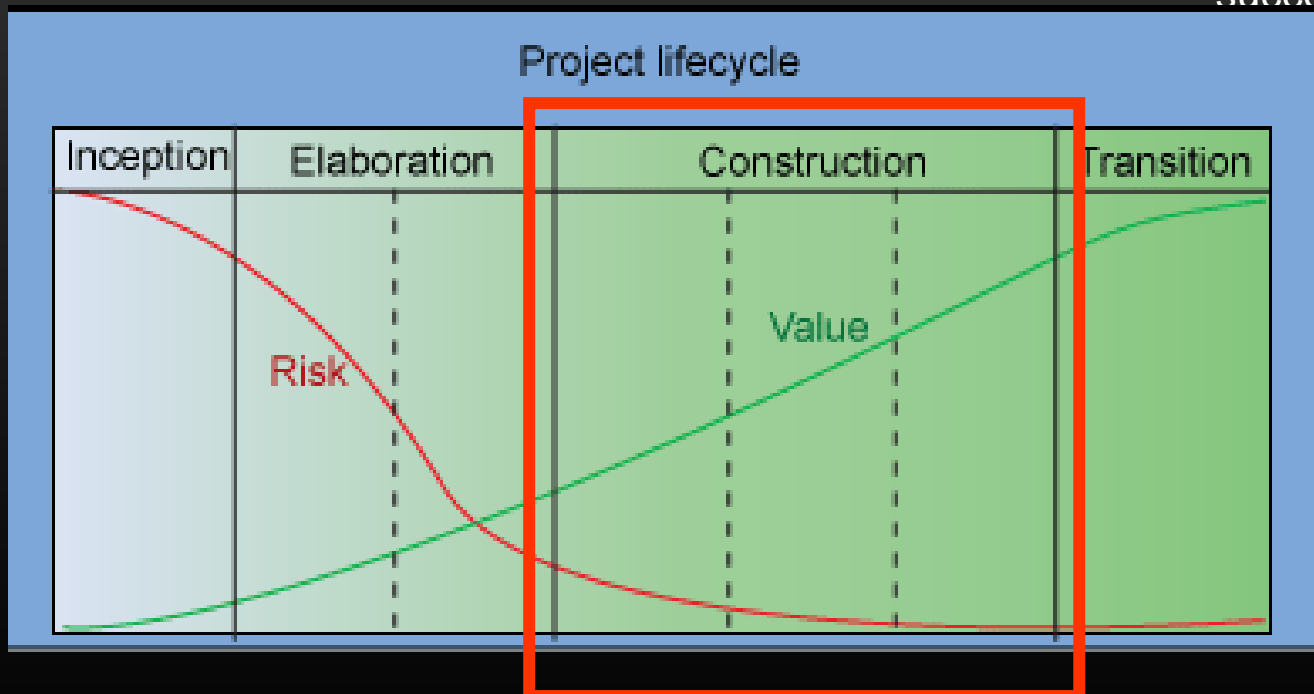
20% of use cases drive 80% of the architecture

Credit: Per Kroll (IBM)

The phases: Construction

Figura

Do we find that we have an application that is sufficiently close to being released that we should switch the primary focus of the team to tuning, polishing and ensuring successful deployment?

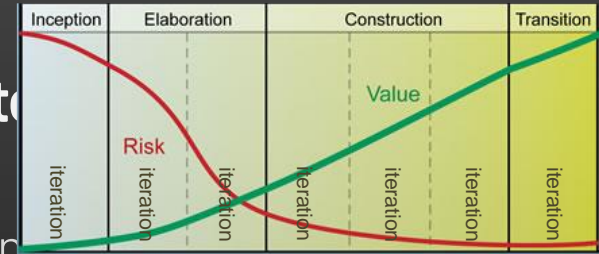


Construction: Build The Product

Incrementally define, design, implement and test scenarios

Incrementally evolve executable architecture to complete system

Evolve architecture as you go along



Frequent demonstrations and partial deployment

Partial deployment strategy depends greatly on what system you build

Daily build with automated build process

You may have to have a separate test team if you have

Complex test environments

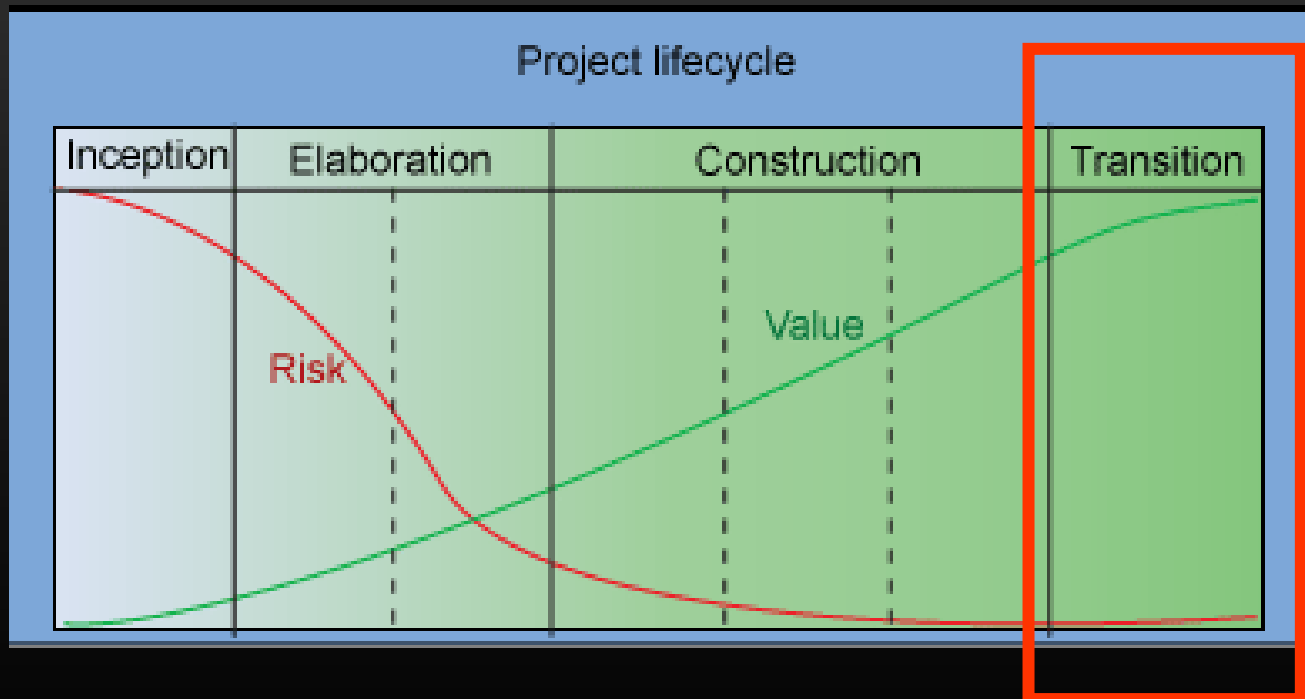
Safety or mission critical systems

Credit: Per Kroll (IBM)

The phases: Transition

Figura

Is the application ready to release?



Transition: Stabilize and Deploy

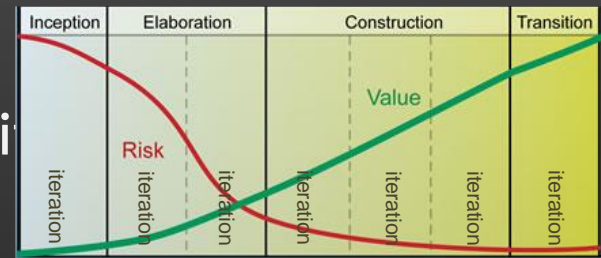
Project moves from focusing on new capabilities and tuning

Produce incremental 'bug-fix' releases

Update user manuals and deployment documentation

Execute cut-over

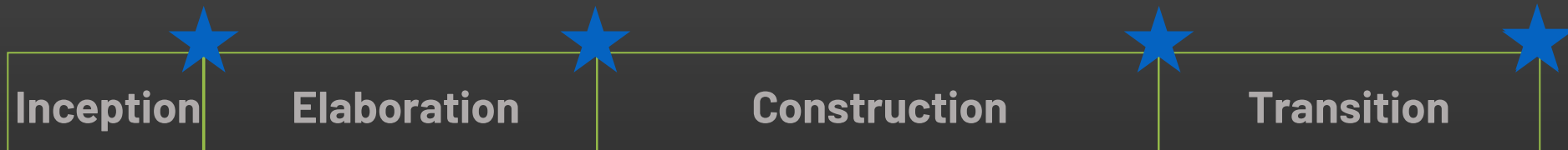
Conduct "post-mortem" project analysis



Credit: Per Kroll (IBM)

Recap main control points (lifecycle objective milestone)

Major Milestones



Inception: Agreement on overall scope

Vision, high-level requirements, business case

Not detailed requirements

Elaboration: Agreement on design approach and mitigation of major risks

Baseline architecture, key capabilities partially implemented

Not detailed design

Construction: Agreement on complete operational system

Develop a beta release with full functionality

Transition: Validate and deploy solution

Stakeholder acceptance, cutover to production

Project assignment

“Partial” OpenUP

Iteration #1

- Covers Inceptions activities
- Develop the concept

Iteration #2

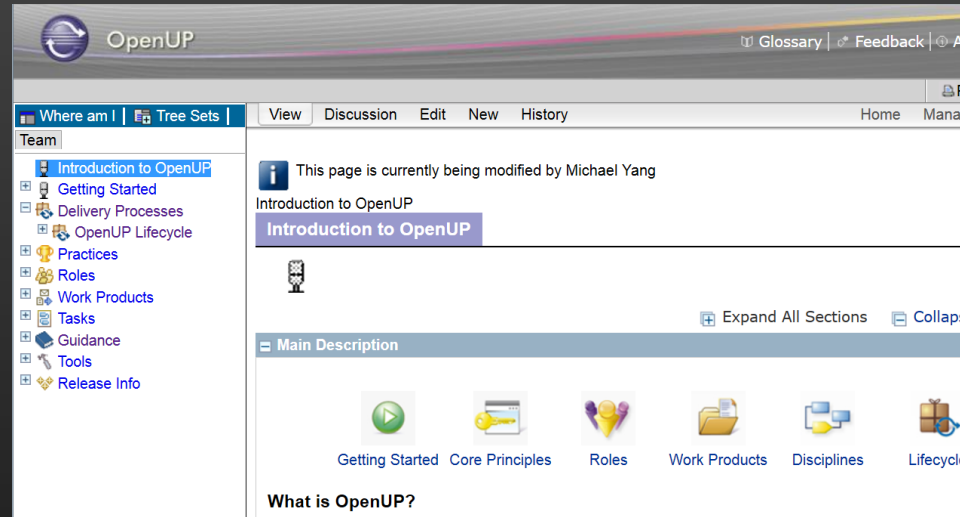
- Elaboration for a small set of core use cases
- Focus on detailing the use cases and prototype

Iteration #3

- Elaboration for more use cases
- Architecture validation by building some

Iteration #4

- Implement core use cases



Key reference:

→ [Project execution plan](#) doc

UP: Conceção

Objetivo	Atividades	Produtos
<ul style="list-style-type: none">• Atingir um consenso entre os diversos <i>stakeholders</i> acerca dos objetivos e âmbito do projeto.• Garantir que as condições necessárias à viabilidade do projeto estão reunidos.	<ul style="list-style-type: none">• Elaborar modelo de requisitos de alto nível.• Identificar interações com entidades externas.• Casos de utilização levantados (os de maior risco podem ser detalhados).• Planeamento das fases subsequentes e pontos de decisão.	<ul style="list-style-type: none">• Visão geral do problema• Modelo de Casos de Utilização (alto nível, parcial)• Avaliação de risco inicial• Justificação da viabilidade do projeto• Plano de projeto• Protótipos iniciais (para mitigação de risco, se necessário).

UP: Elaboração (*Elaboration*)

Objetivo	Atividades	Produtos
<ul style="list-style-type: none">• Validar os cenários• Definir a arquitetura	<ul style="list-style-type: none">• Detalhar o modelo de casos de utilização• Analisar domínio• Definir arquitetura candidata• Validar arquitetura com implementação exploratória	<ul style="list-style-type: none">• Modelo de Casos de Utilização (especificação abrangente)• Requisitos (incluindo não-funcionais)• Descrição da arquitetura do software• Protótipos (mitigação de risco).• Implementação exploratória (validar arquitetura).• Plano de projeto revisto• Medidas para mitigação do risco

UP: Construção

Objetivo	Atividades	Produtos
<ul style="list-style-type: none">Entregar valor (cenários nucleares)	<ul style="list-style-type: none">Definir realização dos casos de utilização (interações entre objetos)Desenvolver UI/UXImplementar códigoValidar incremento (garantia de qualidade)	<ul style="list-style-type: none">Implementação parcial (cenários core).Garantia de qualidade: demonstração de testes funcionais