

Modelos de Processo Evolutivos

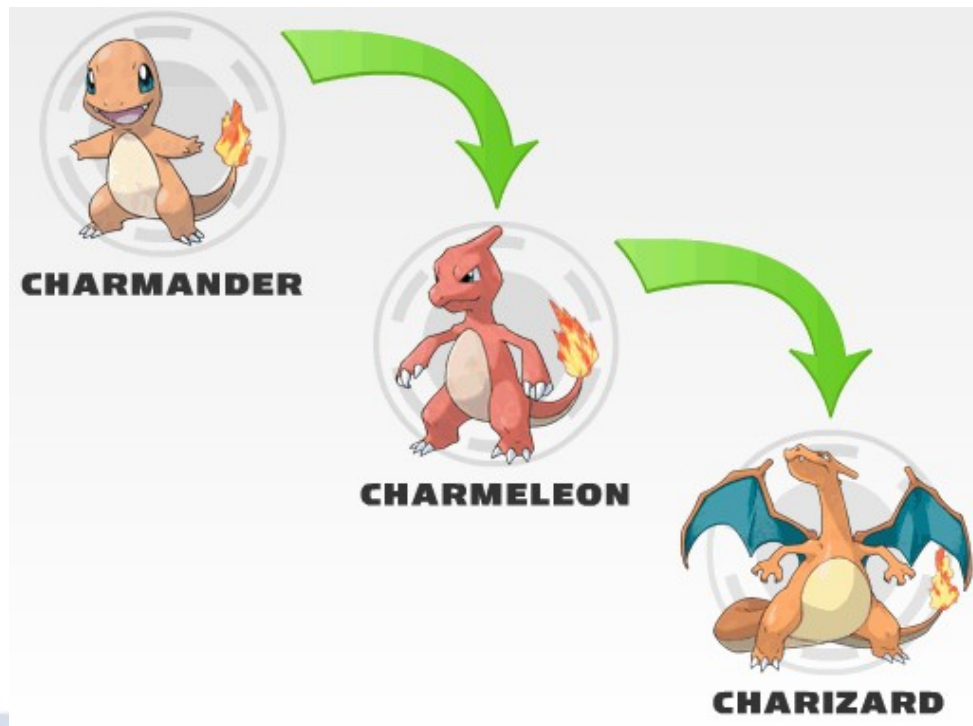
Prof. André Takeshi Endo

Até Agora...

- Modelos de processo
 - Cascata
 - Prototipação
 - RAD
 - Modelo em V
 - Desenvolvimento formal
- **Modelos de processo evolutivos**
- Unified Process (UP)
- Métodos ágeis

Modelos Evolutivos

- Natureza do software
 - “Software não se desgasta se deteriora”
- Software evolui !!!!



Modelos Evolutivos

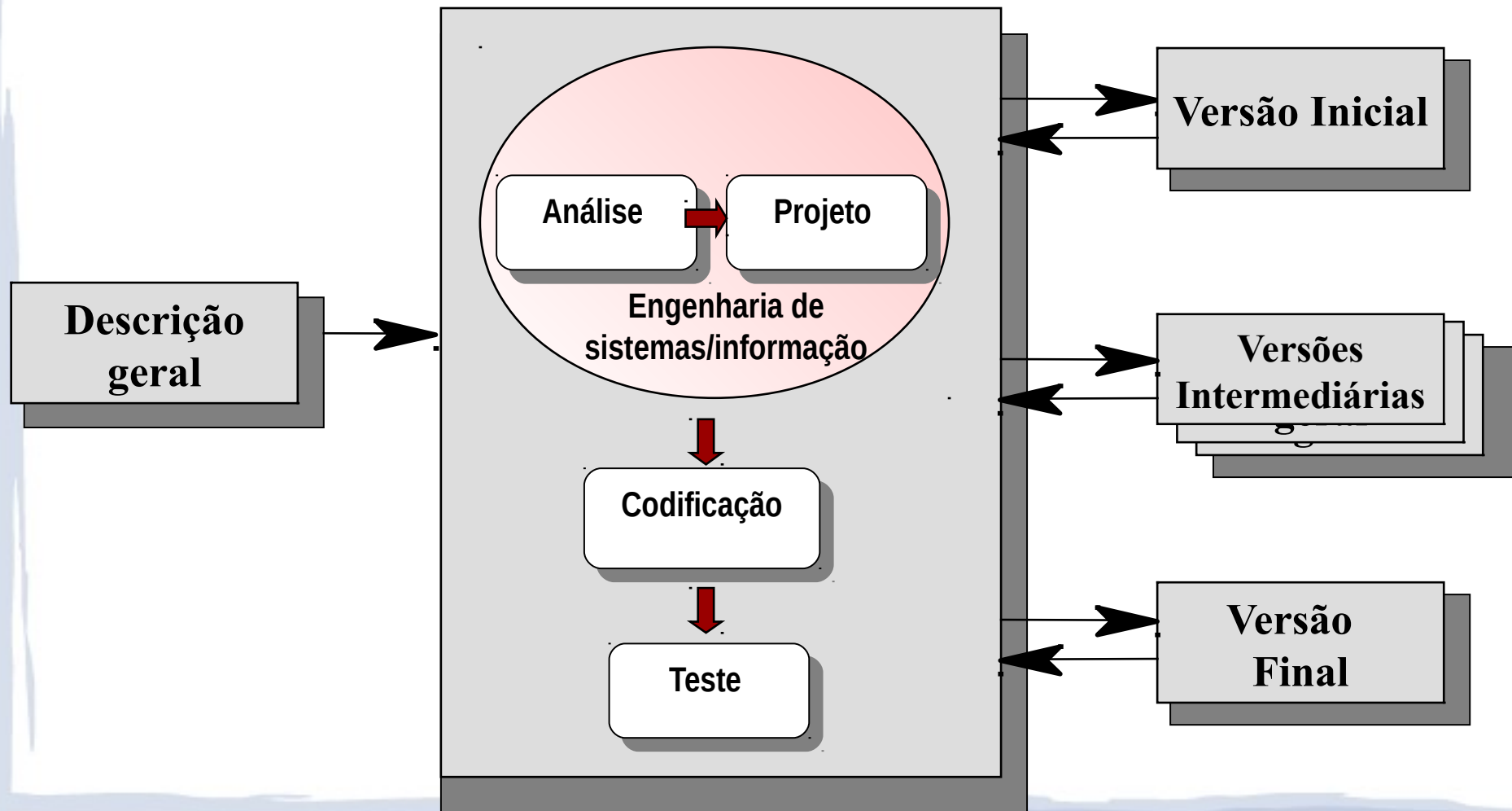
- Modelos de processo para
 - Requisitos mutantes
 - Entregas em períodos mais curtos
 - Software que evolui!!
- Modelos evolutivos são ***iterativos***
- Possibilita a entrega de ***versões*** cada vez mais complexas

Modelo Incremental

- Combina o cascata com a prototipação
 - Cascata → repetidas vezes
 - Prototipação → filosofia
- De maneira incremental
 - Descobrir requisitos
 - Entregar o produto final

Modelo Incremental

- Representação

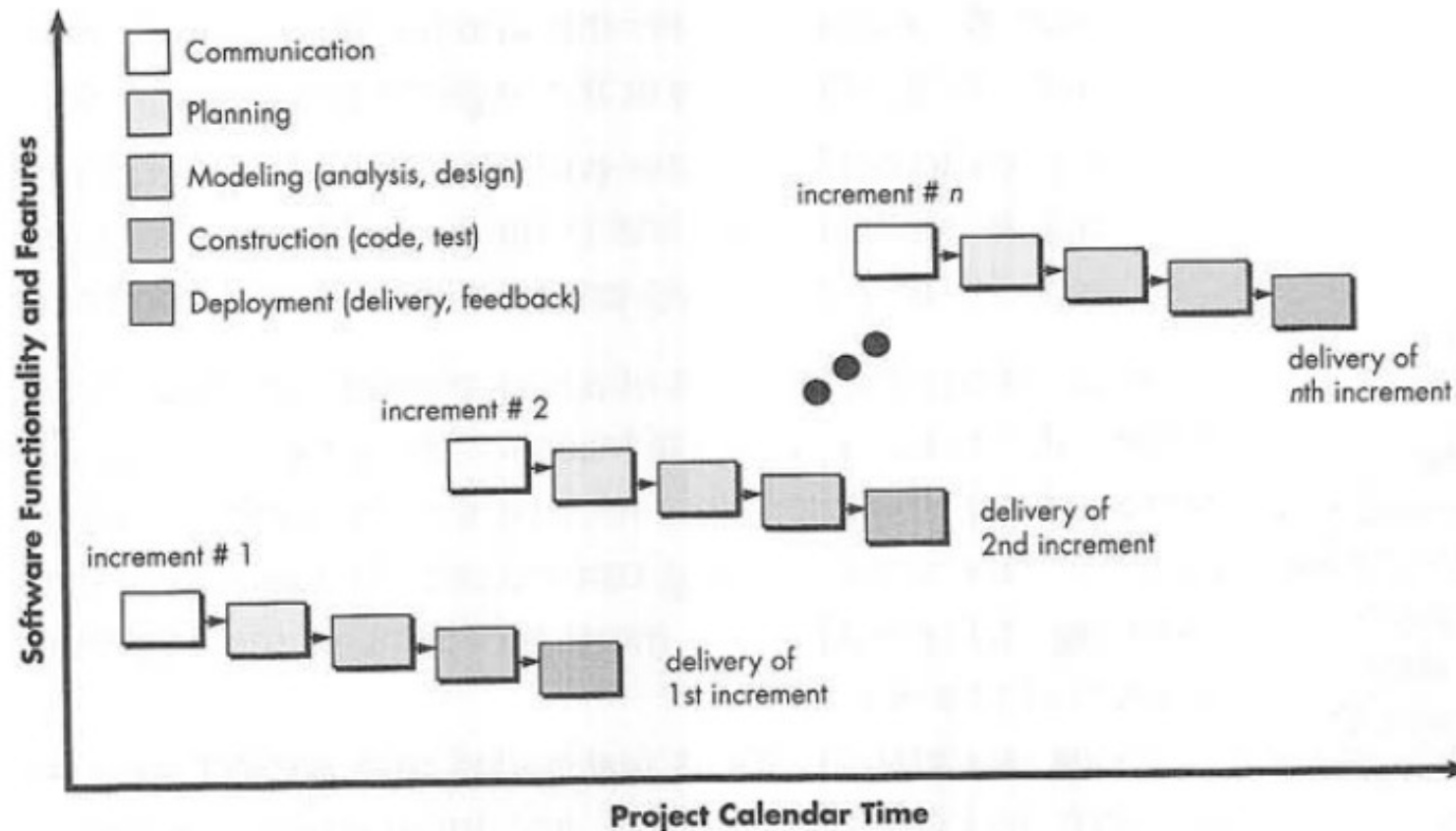


Modelo Incremental

- Versões são chamadas de **incrementos**
- O incremento ***não é*** um protótipo
 - Versão funcional e operacional
- Versão inicial é o núcleo principal
 - Produto essencial
- O planejamento do próximo incremento já considera modificações (*feedback* do usuário)

Modelo Incremental

- Representação [Pressman]



Q: o que diz?

Modelo Incremental (-)

- Priorizar as funcionalidades
- Planejar o incremento
 - Não pode ser muito grande
 - Deve entregar uma funcionalidade
- Difícil identificar facilidades básicas comum a todos os requisitos
 - ??

Modelo Incremental (+)

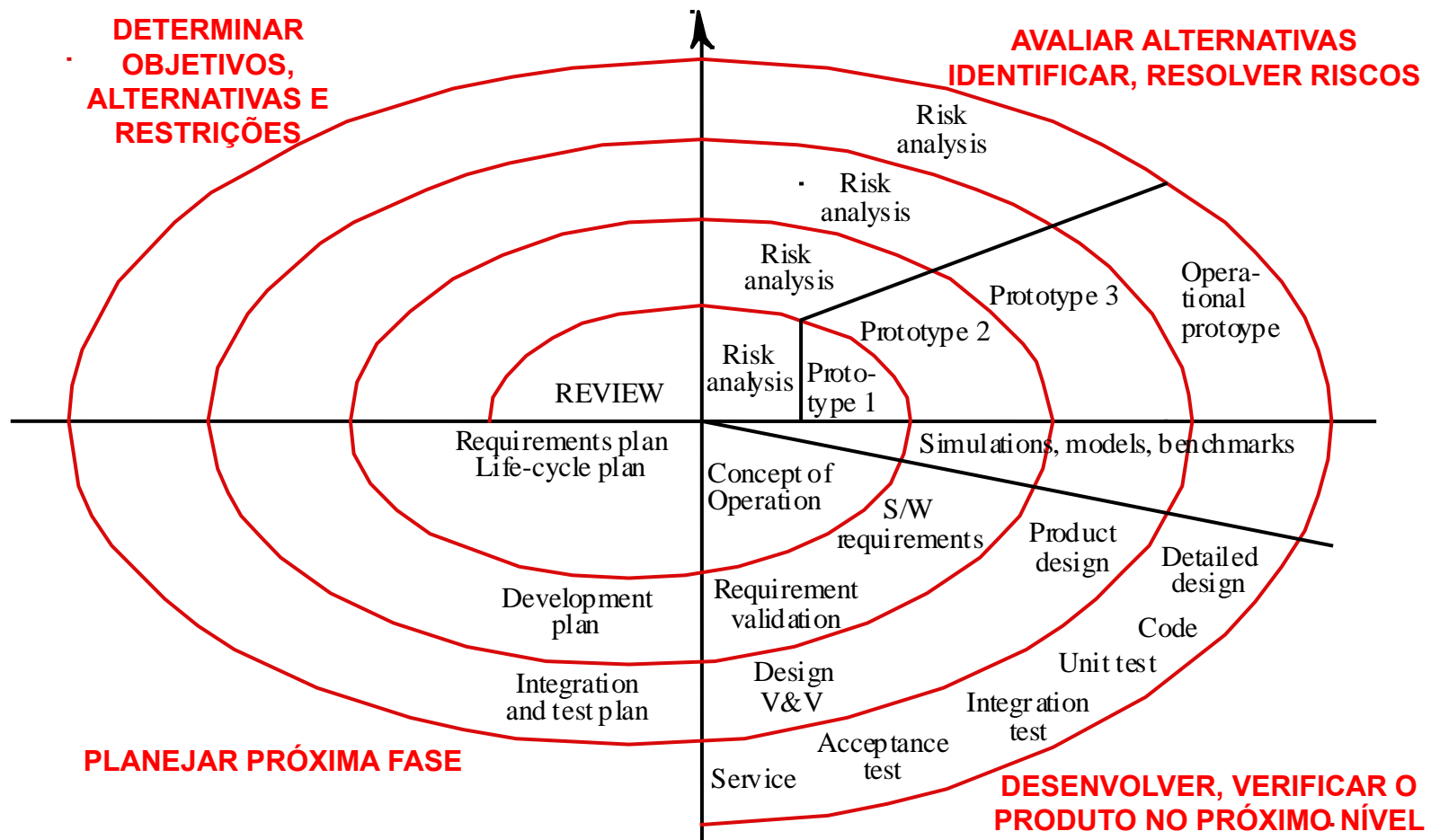
- Gerenciar riscos técnicos
 - Iniciar com as funcionalidades mais arriscadas
 - Planejar a disponibilidade de hardware
- Adequado para sistemas nos quais os requisitos são difíceis de especificar em estágios iniciais
- Começar com um equipe pequena
- Cada incremento pode usar um processo

Modelo Espiral

- Barry Boehm
- Natureza iterativa → prototipação
- Controle e sistemática → cascata
- É dividido em uma série de regiões
 - Variantes com 3 a 6 regiões

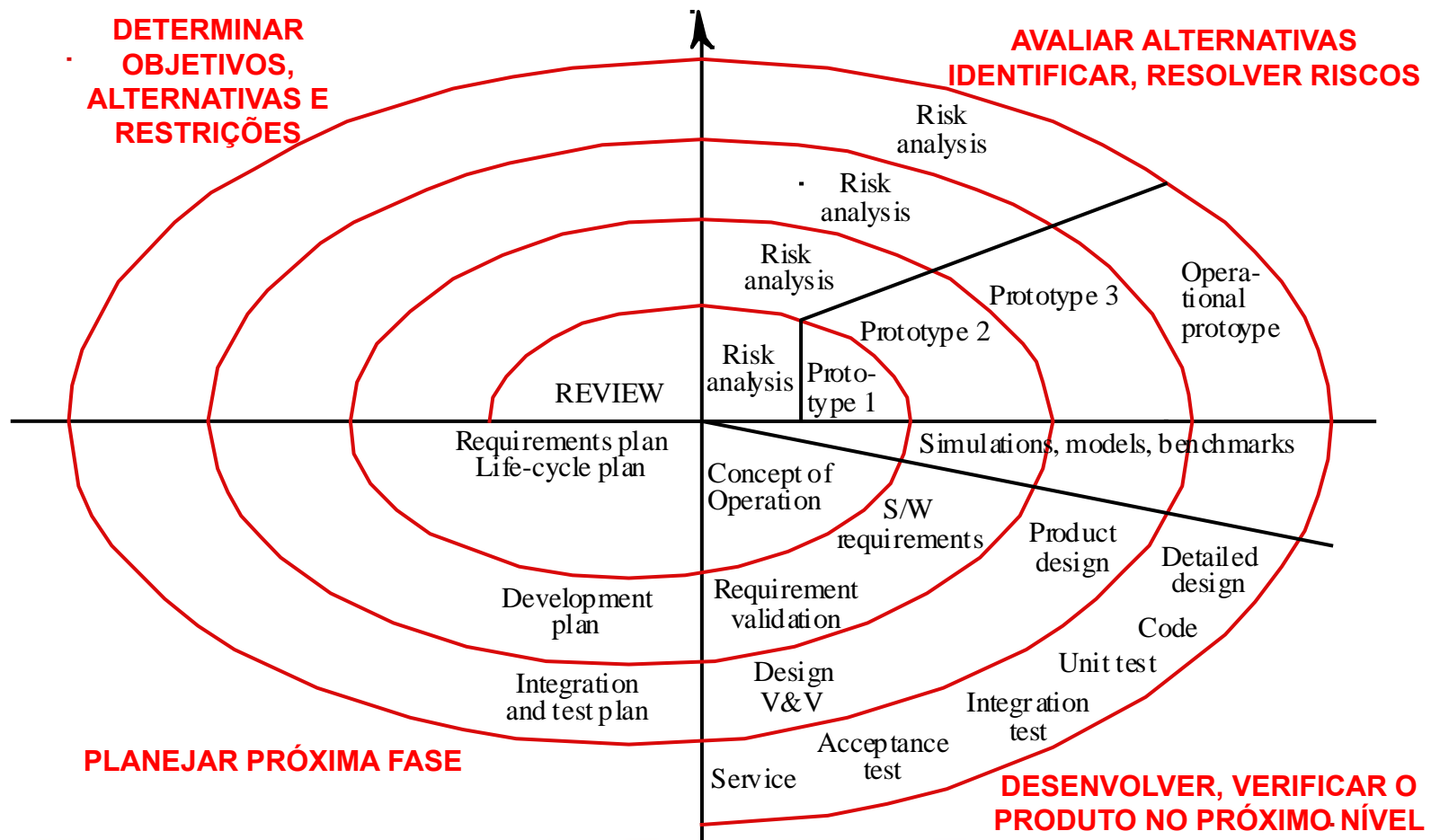
Modelo Espiral

- Representação (4 regiões) [Sommerville]



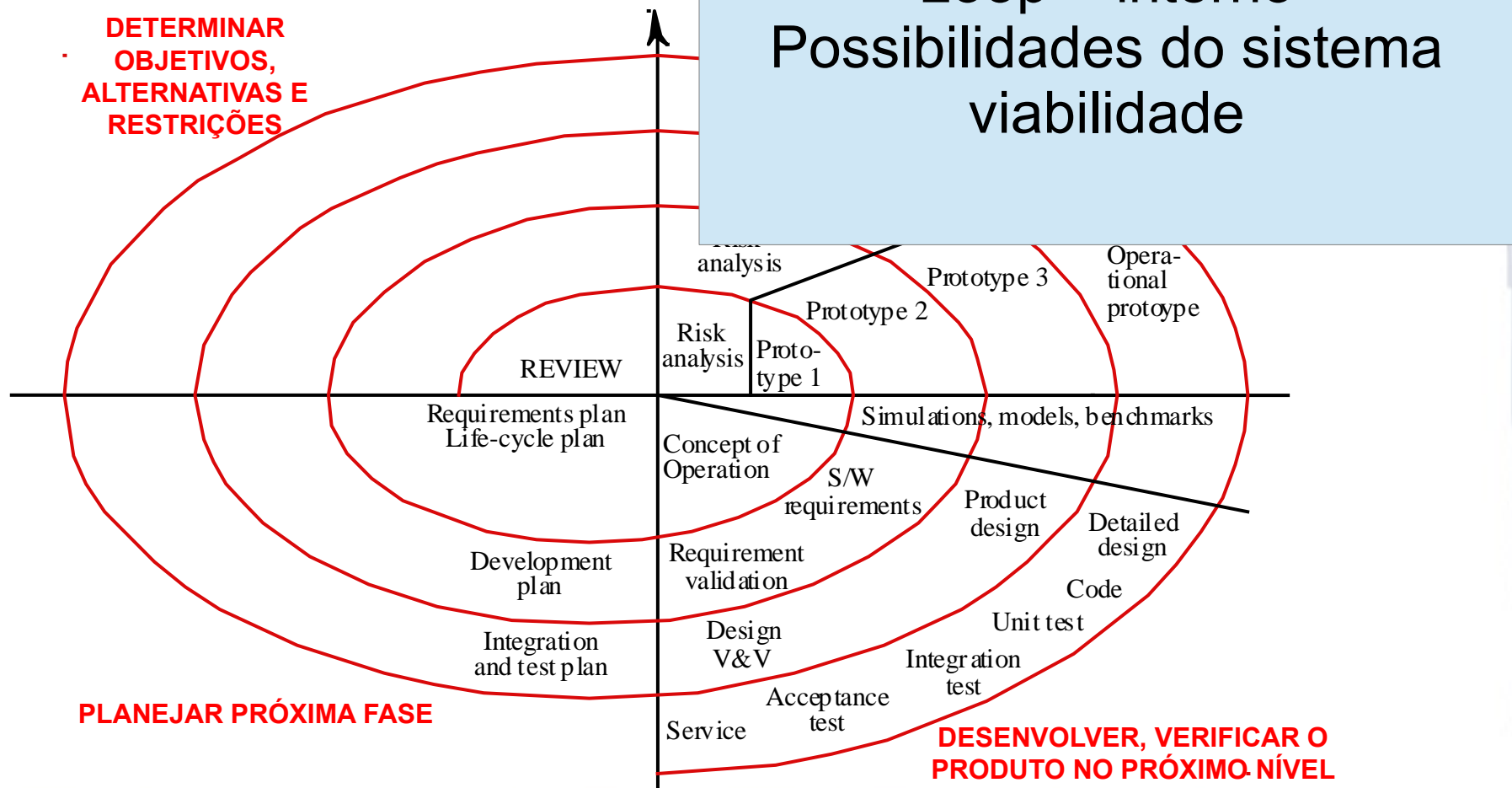
Modelo Espiral

- Cada loop representa uma fase do processo



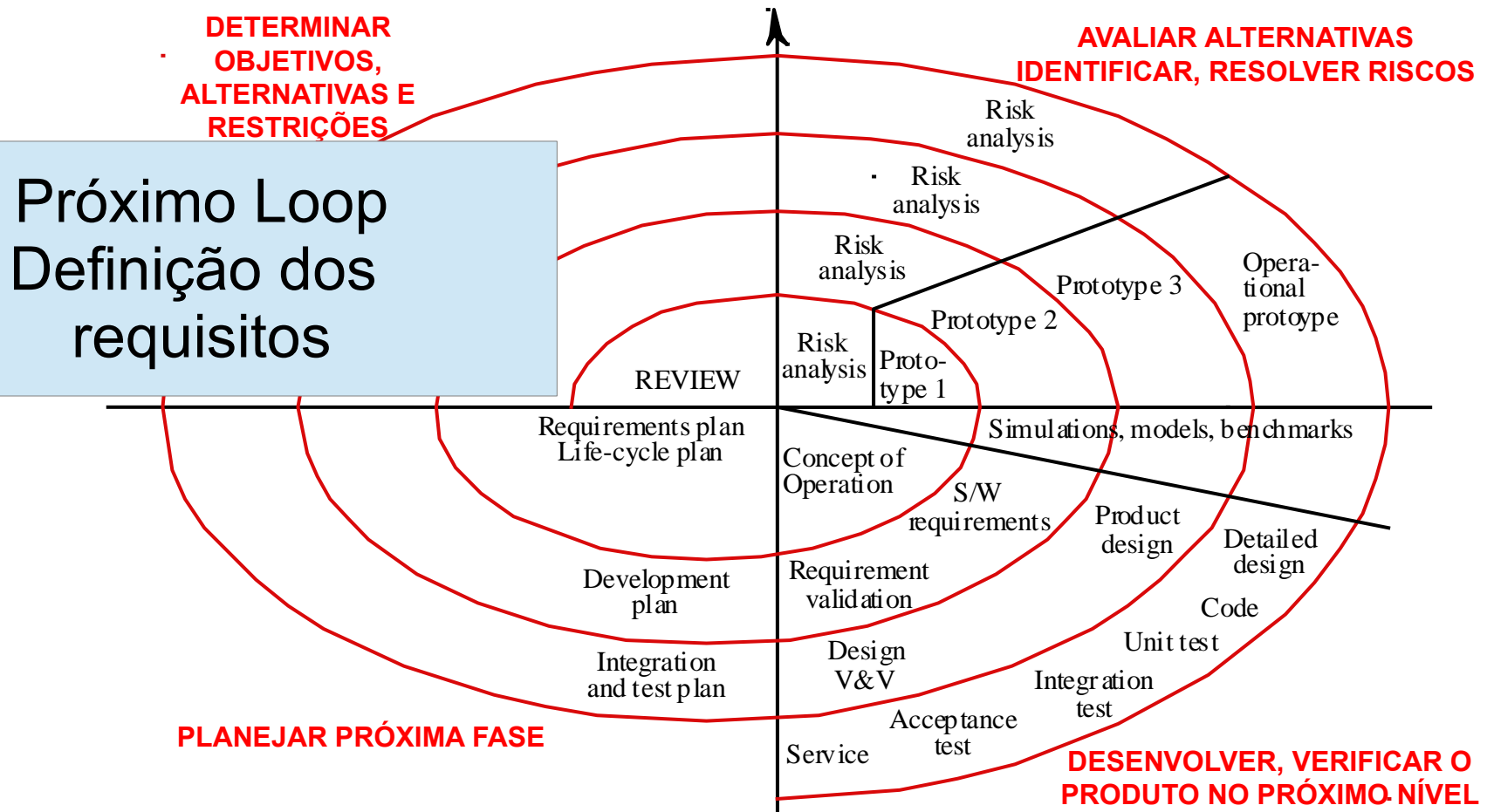
Modelo Espiral

- Representação



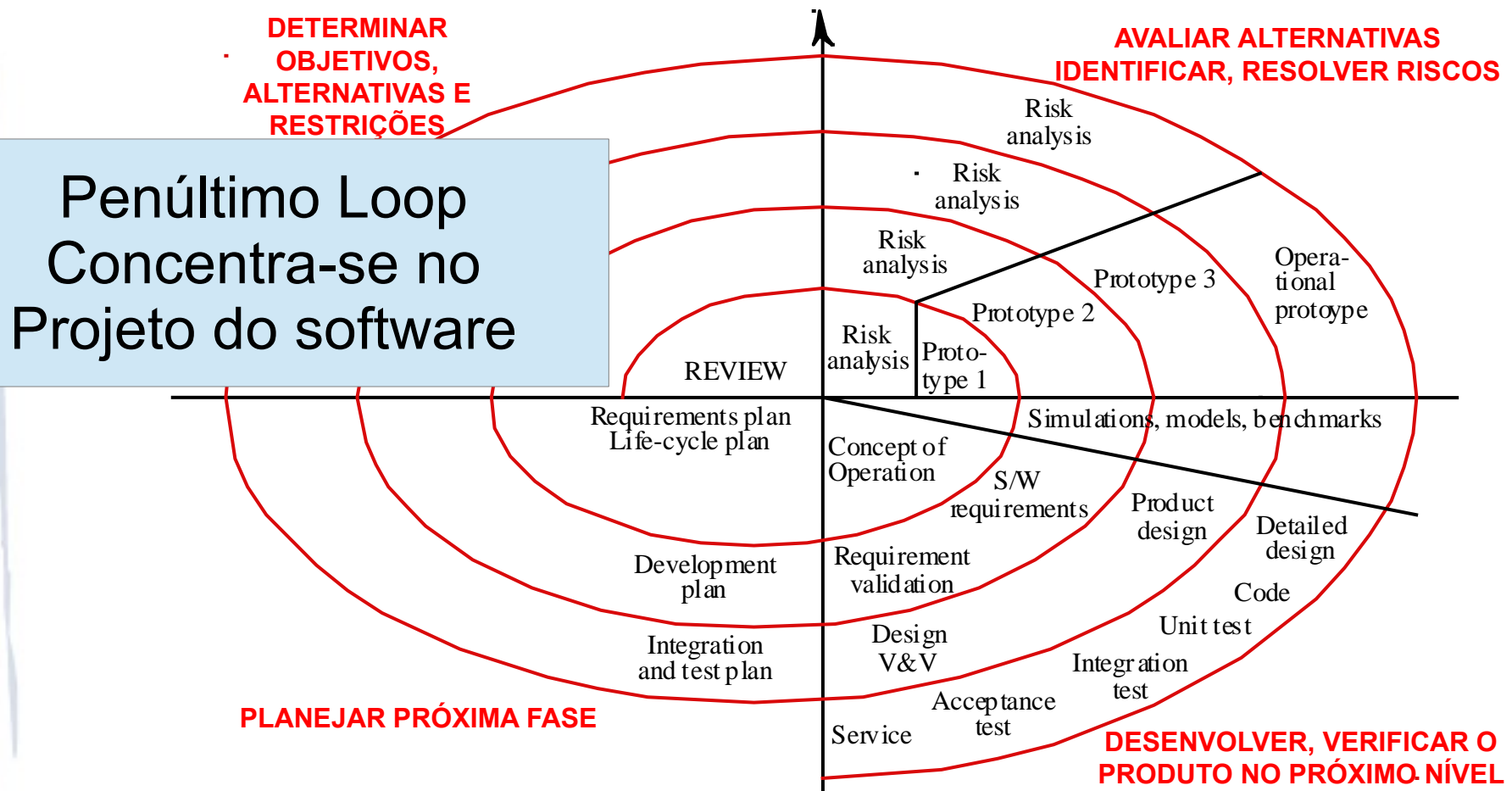
Modelo Espiral

- Representação



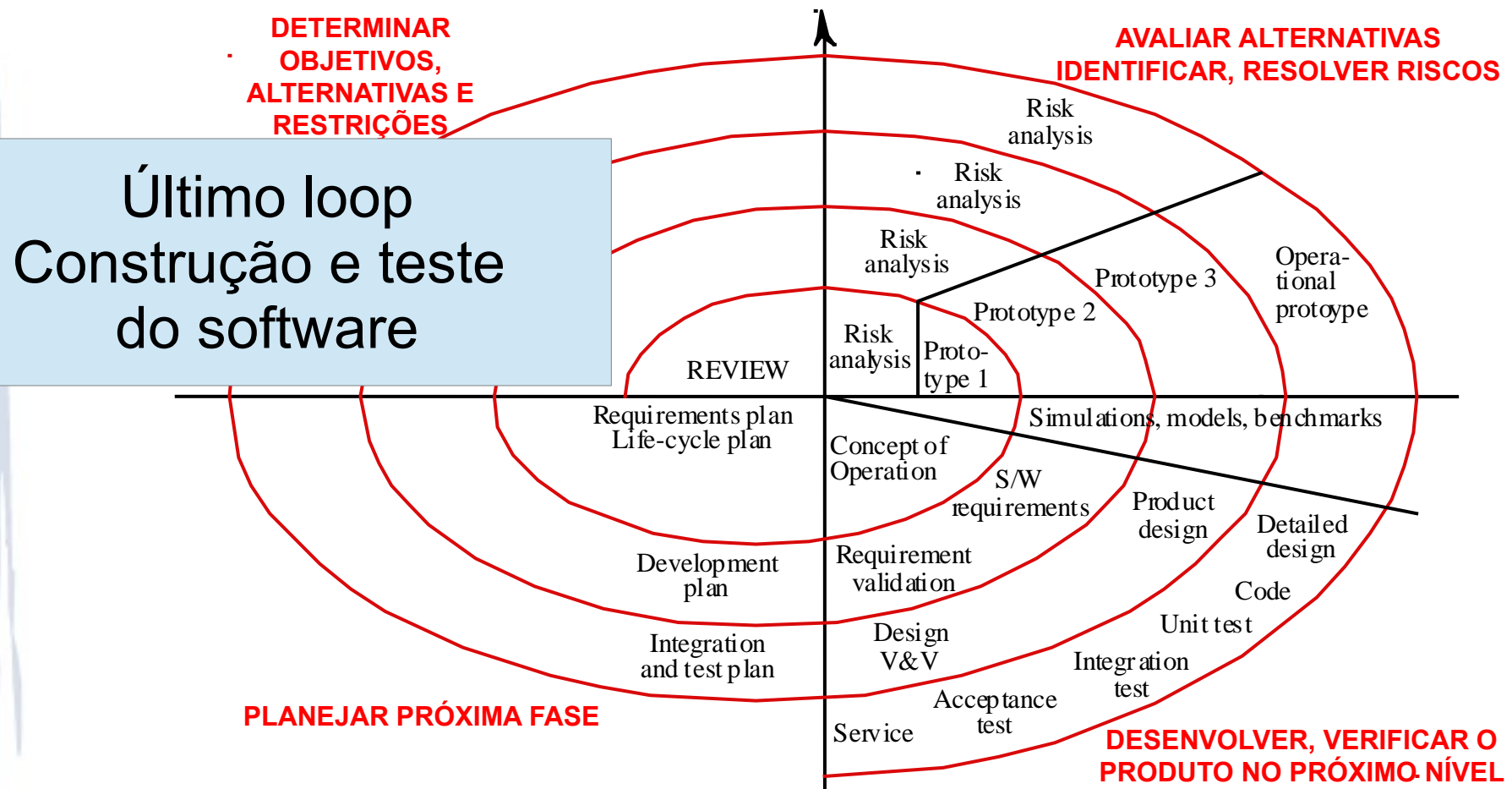
Modelo Espiral

- Representação



Modelo Espiral

- Representação

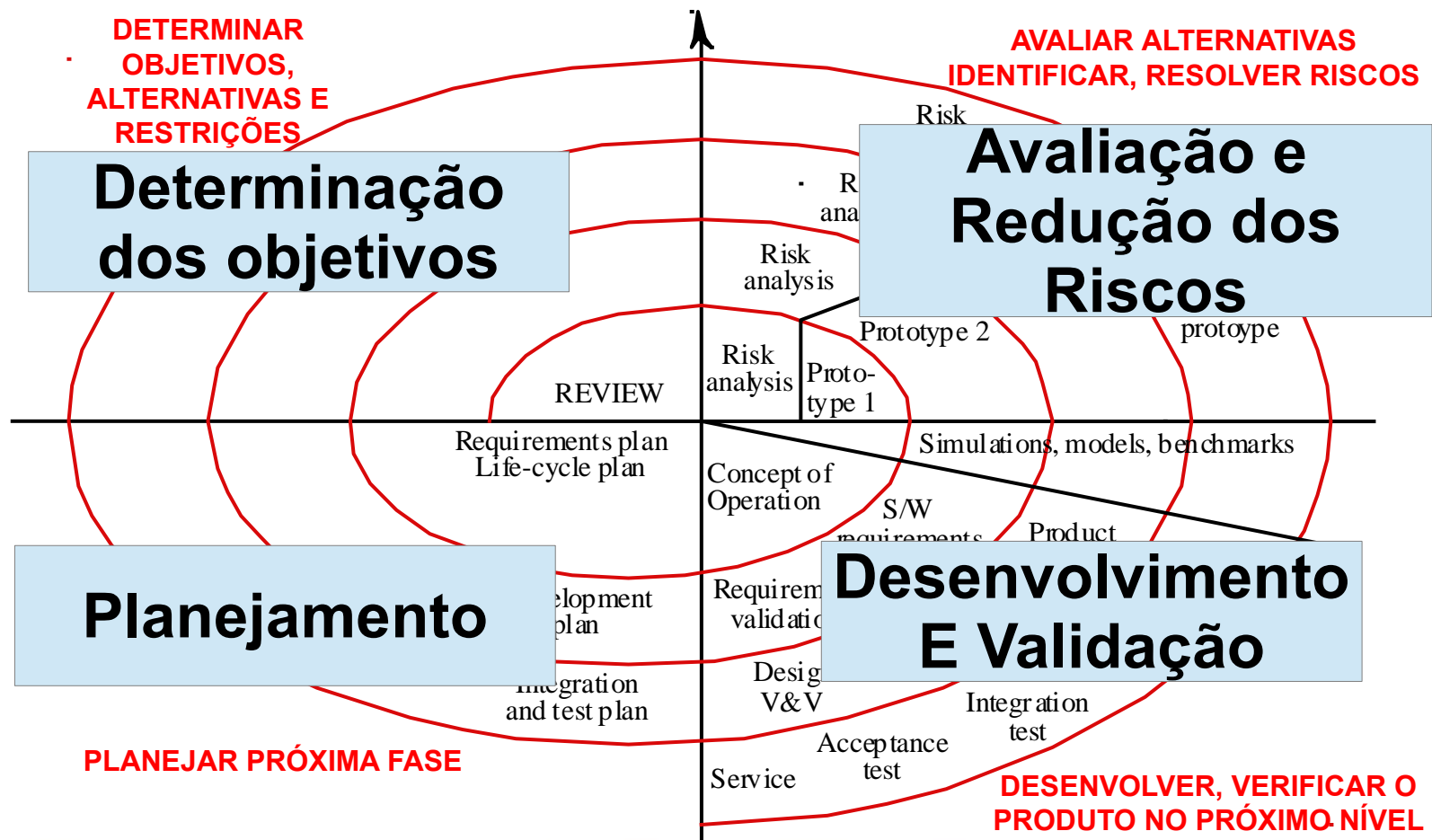


Modelo Espiral

- As fases mostradas na figura são apenas exemplos
- Não existem fases pré-determinadas
- A empresa pode definir e estruturar as fases

Modelo Espiral


- Cada loop passa por cada uma das regiões



Modelo Espiral

- Determinação dos objetivos
 - Define objetivos para cada fase
 - Identifica restrição
 - Identifica riscos
- Avaliação e redução de riscos
 - Cada risco identificado é analisado individualmente
 - Passos são tomados para reduzir esses riscos

risk

US  /rɪsk/ n [C/U]

1 danger, or the possibility of danger, defeat, or loss:

[C] There's a risk of an accident happening in this fog.

[C] I was afraid to take the risk of quitting my job before I had another one lined up.

[U] We want clean rivers and lakes, where you can swim without risk to your health.

[U] It was a high/low risk situation (= a situation with a lot of/very little danger).

2 A risk is also someone or something that could cause a problem or loss:

Modelo Espiral

- Desenvolvimento e Validação
 - O software é construído
 - Qualquer modelo pode ser escolhido
- Planejamento
 - Projeto atual é revisto
 - Planos são desenvolvidos para o próximo loop

Modelo Espiral (-)

- Convencer clientes de que é controlável
- Experiência na determinação dos riscos
 - Chave para o sucesso
 - Esquecer um risco = problemas

Modelo Espiral (+)

- Melhores características + análise de riscos
 - Modelo adaptado a riscos inerentes
- Estrutura iterativa é mais realística
- Usa a prototipação como estratégia para reduzir riscos
- Cada loop = + software, - riscos

Concluindo...

- Modelos de processo
 - Mais adequados?
- O software evolui
 - Modelos evolutivos
- Modelo Incremental
- Modelo Espiral

Bibliografia

- [Pfleeger07] S. L. Pfleeger, “Engenharia de Software: Teoria e Prática”, 2007.
- [Pressman11] R. S. Pressman, “Engenharia de Software: uma abordagem profissional”, 2011.
- [Sommerville03] I. Sommerville, “Engenharia de Software”, 2003.
- [Brooks87] “No Silver Bullet: Essence and Accidents of Software Engineering”, 1987.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1663532
- [IEEE90] “IEEE Standard Glossary of Software Engineering Terminology”, 1990.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=159342

Bibliografia

- [UUU] Materiais didáticos elaborados pelos grupos de engenharia de software do ICMC-USP, DC-UFSCAR e UTFPR-CP.