

MODELOS DE PROCESSO DE SOFTWARE

O que é Processo Software?

Um processo de software é composto por um conjunto de etapas que são necessárias para que seja produzido. Sommerville (2007) afirma que um modelo de processo de software é uma representação abstrata, simplificada de um processo de software. Os modelos de processo incluem as atividades que fazem parte do processo de software, os artefatos de software que devem ser produzidos em cada uma das atividades (documentos) e também os papéis das pessoas envolvidas na engenharia de software. Além disso, cada modelo de processo representa um processo a partir de uma perspectiva particular, de uma maneira que proporciona apenas informações parciais sobre o processo.

O que é Processo Software?

Para que um software seja produzido, são necessárias diversas etapas, as quais são compostas por uma série de tarefas. Para esse conjunto de etapas, dá-se o nome de processo de software, o qual pode envolver o desenvolvimento de software a partir do zero, em uma determinada linguagem de programação (por exemplo, o Java ou C) ou, então, a ampliação e a modificação de sistemas já em utilização pelos usuários.

Modelos de processo de software

Na literatura, existem diversos modelos de processo de software. Em nossa disciplina, trabalharemos somente três desses modelos.

Modelos mais comuns:

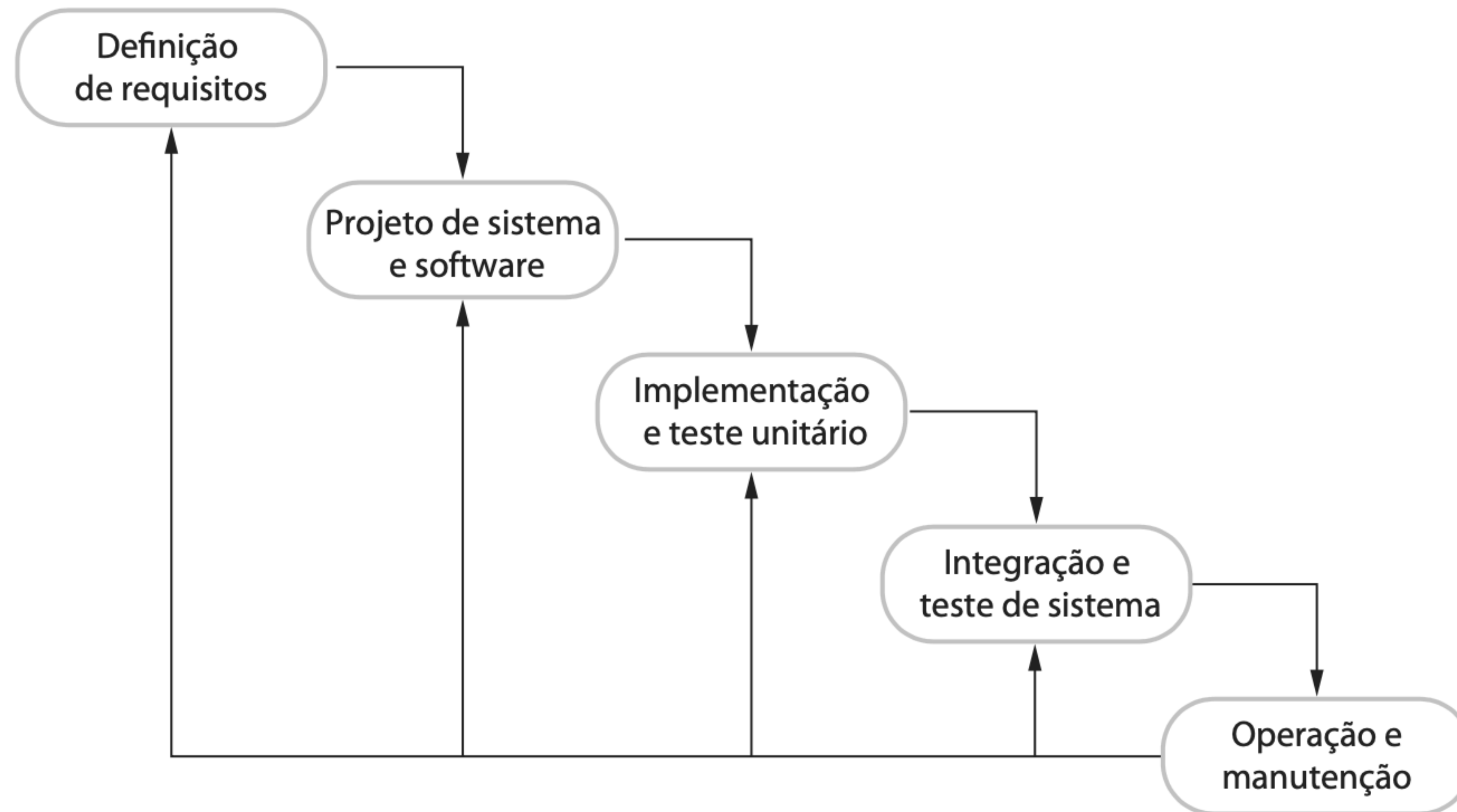
- Cascata
- Incremental
- Orientado a reuso

Modelo em Cascata (ciclo de vida clássico)

Esse modelo considera as atividades fundamentais do processo de especificação, desenvolvimento, validação e evolução, e representa cada uma delas como fases separadas, como: especificação de requisitos, projeto de software, implementação, testes e assim por diante.

Figura 2.1

O modelo em cascata



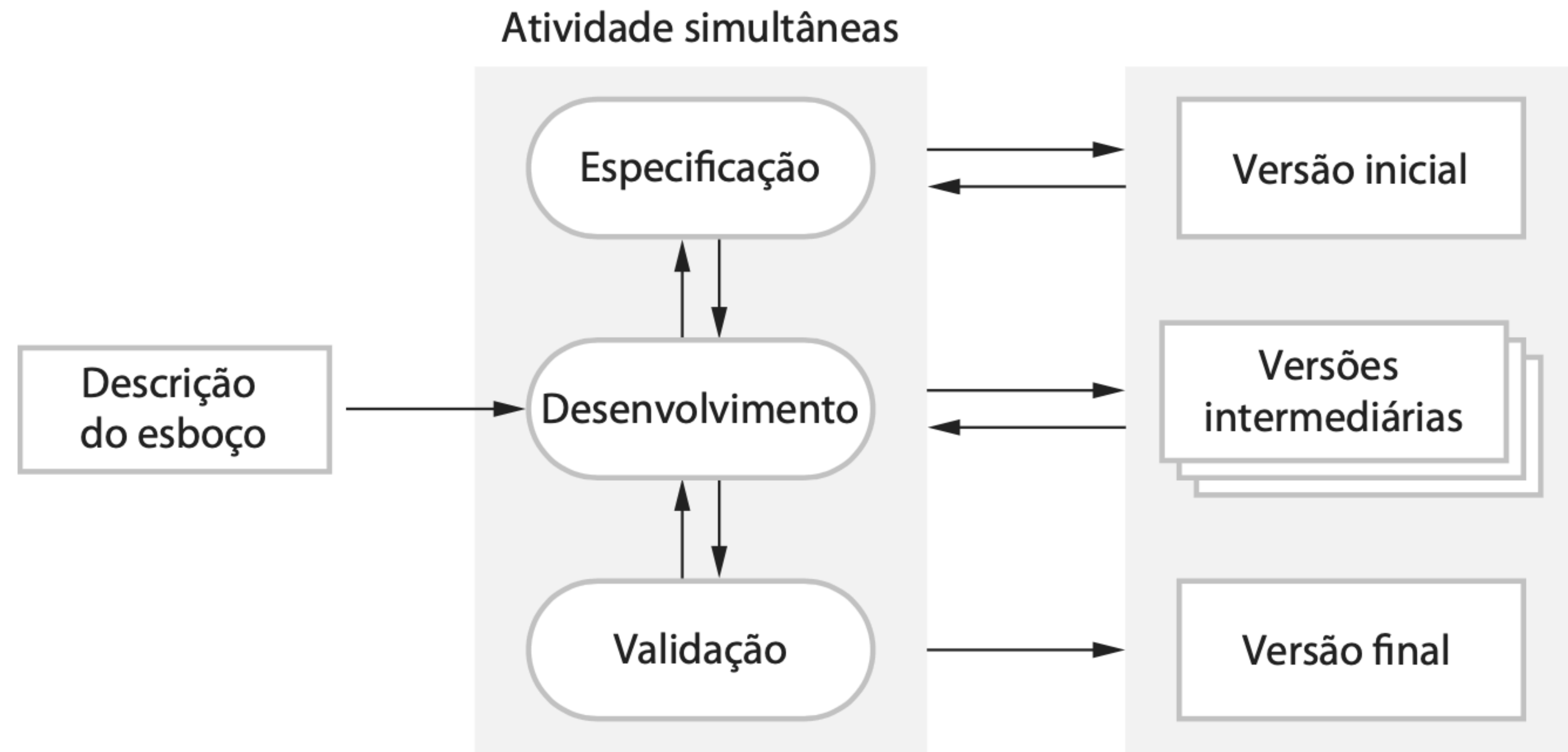
Fonte: Sommerville

Modelo de Desenvolvimento Incremental

Esse modelo intercala as atividades de especificação, desenvolvimento e validação. É um sistema inicial rapidamente desenvolvido a partir de especificações abstratas, que são refinadas com informações do cliente, para produzir um sistema que satisfaça as suas necessidades, produzindo várias versões do software.

Figura 2.2

Desenvolvimento incremental

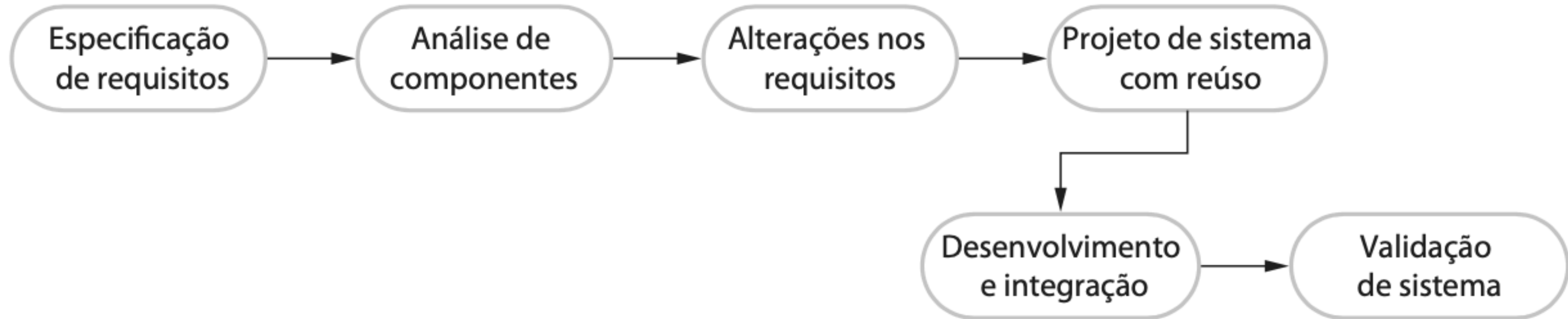


Engenharia de Software Orientada a Reuso

Esse modelo parte do princípio de que existem muitos componentes que podem ser reutilizáveis. O processo de desenvolvimento do sistema concentra-se em combinar vários desses componentes em um sistema, em vez de proceder ao desenvolvimento a partir do zero, com o objetivo de reduzir o tempo de desenvolvimento.

Figura 2.3

Engenharia de software orientada a reúso



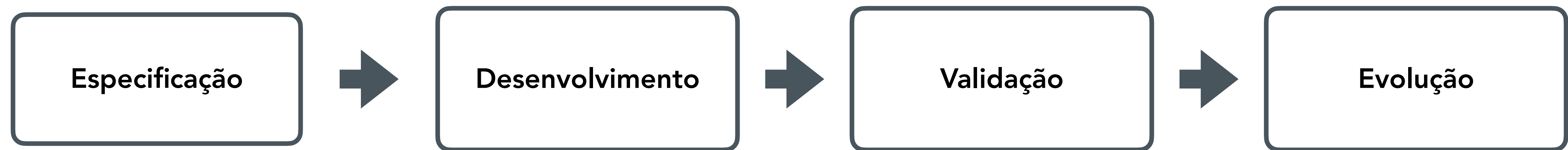
Fonte: Sommerville

Atividades do Processo Software

1. **Especificação de software:** é necessário que o cliente defina as funcionalidades do software que será desenvolvido, bem como todas as suas restrições operacionais.
2. **Desenvolvimento e implementação de software:** o software deve ser confeccionado mediante as especificações definidas anteriormente.
3. **Validação de software:** o software precisa ser validado, a fim de garantir que ele faça o que o cliente deseja, ou seja, que atenda às especificações de funcionalidade.
4. **Evolução de software:** as funcionalidades definidas pelo cliente durante o desenvolvimento do software podem mudar. Consequentemente, o software precisará evoluir para atender a essas mudanças.

O que é ruim em cada modelo?

Em resumo temos esse processo...



Fonte: Sommerville

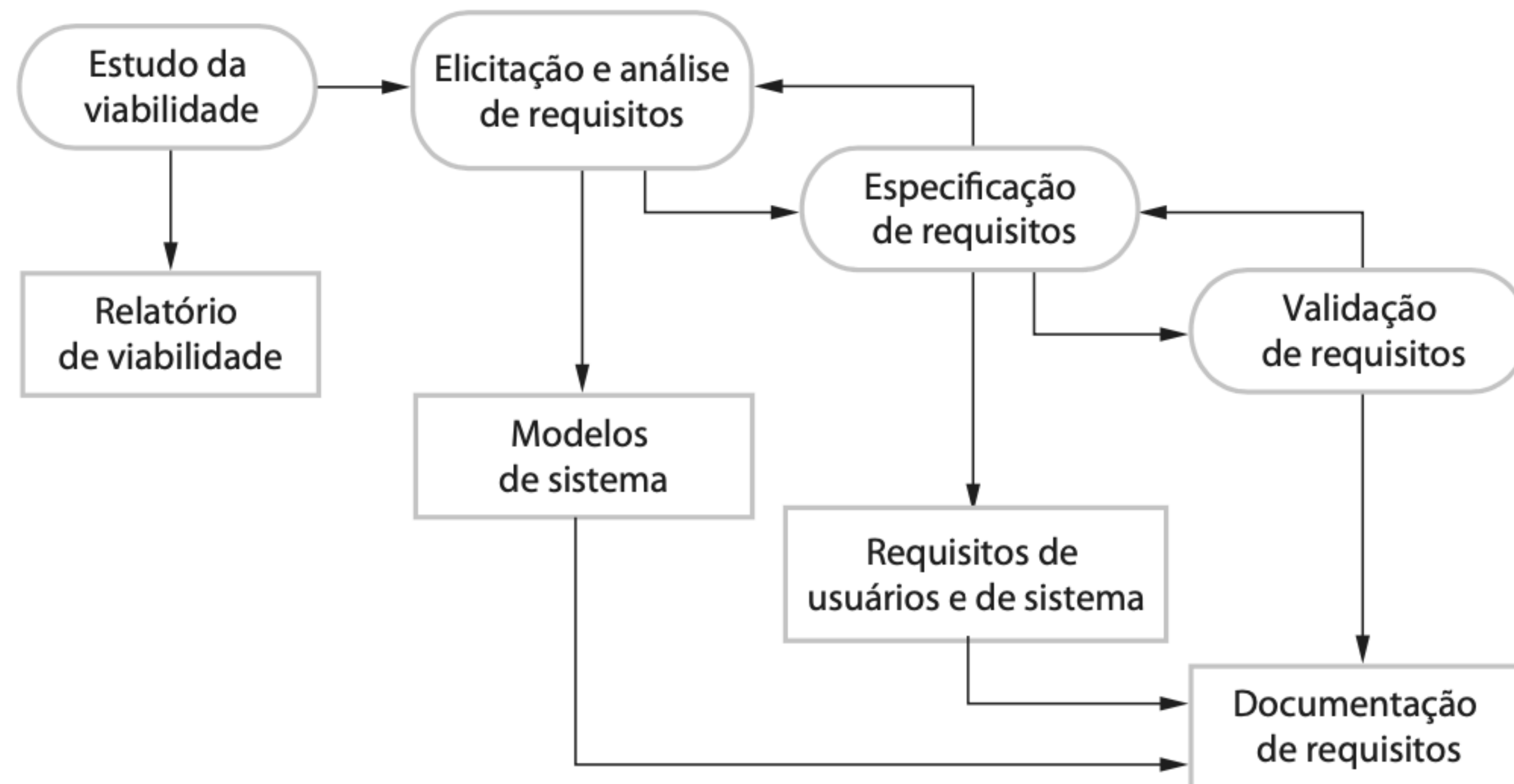
Especificação de Software

Especificação de software ou engenharia de requisitos é o processo de compreensão e definição dos serviços requisitados do sistema e identificação de restrições relativas à operação e ao desenvolvimento do sistema. A engenharia de requisitos é um estágio particularmente crítico do processo de software, pois erros nessa fase inevitavelmente geram problemas no projeto e na implementação do sistema.

Especificação de Software

Figura 2.4

Os requisitos da engenharia de processos



Fonte: Sommerville

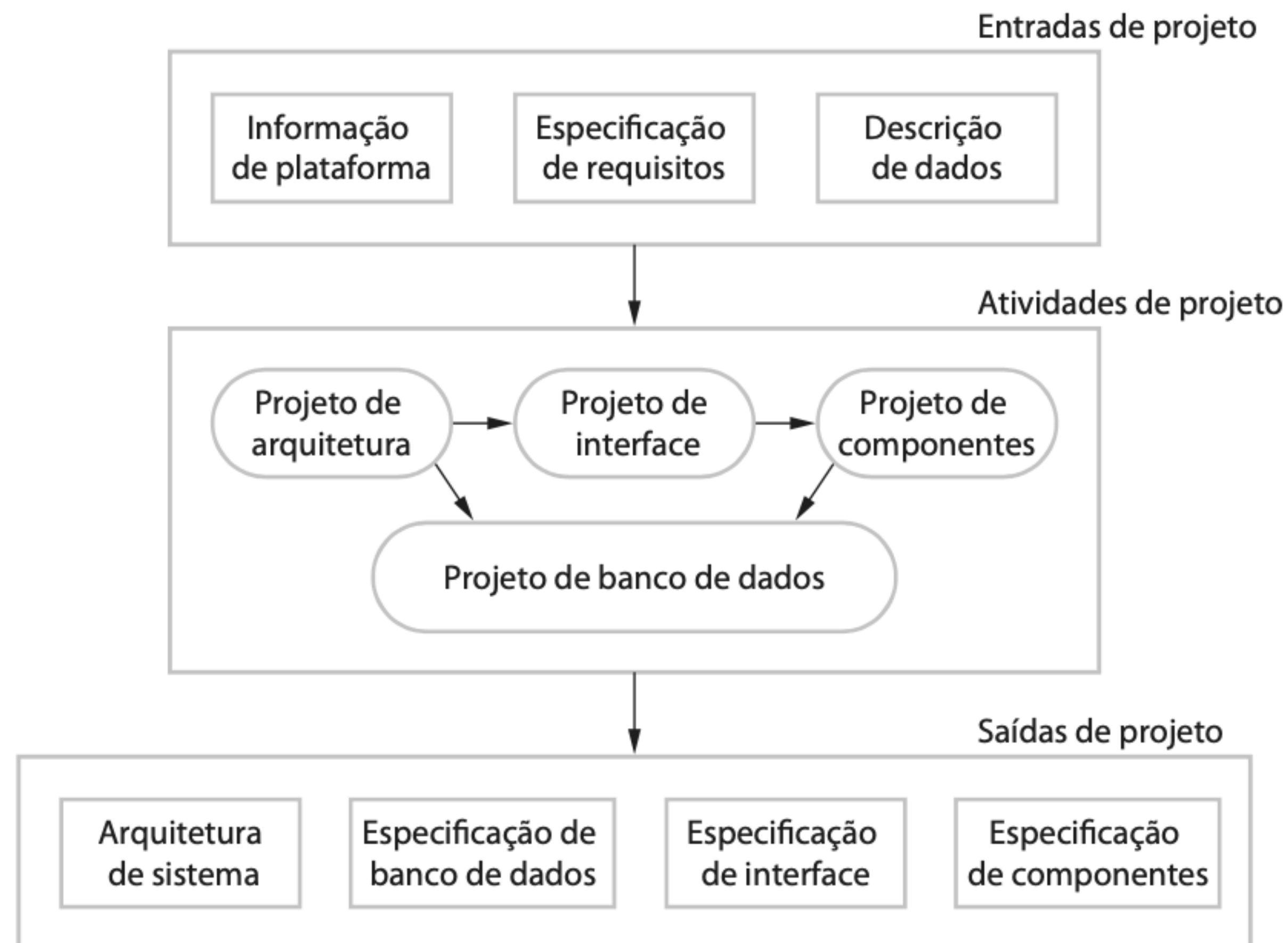
Desenvolvimento de Software

O estágio de implementação do desenvolvimento de software é o processo de conversão de uma especificação do sistema em um sistema executável. Sempre envolve processos de projeto e programação de software, mas, se for usada uma abordagem incremental para o desenvolvimento, também pode envolver o refinamento da especificação do software.

Um projeto de software é uma descrição da estrutura do software a ser implementado, dos modelos e estruturas de dados usados pelo sistema, das interfaces entre os componentes do sistema e, às vezes, dos algoritmos usados. Os projetistas não chegam a um projeto final imediatamente, mas desenvolvem-no de forma iterativa. Eles acrescentam formalidade e detalhes, enquanto desenvolvem seu projeto por meio de revisões constantes para correção de projetos anteriores.

Desenvolvimento de Software

Figura 2.5 Um modelo geral do processo de projeto



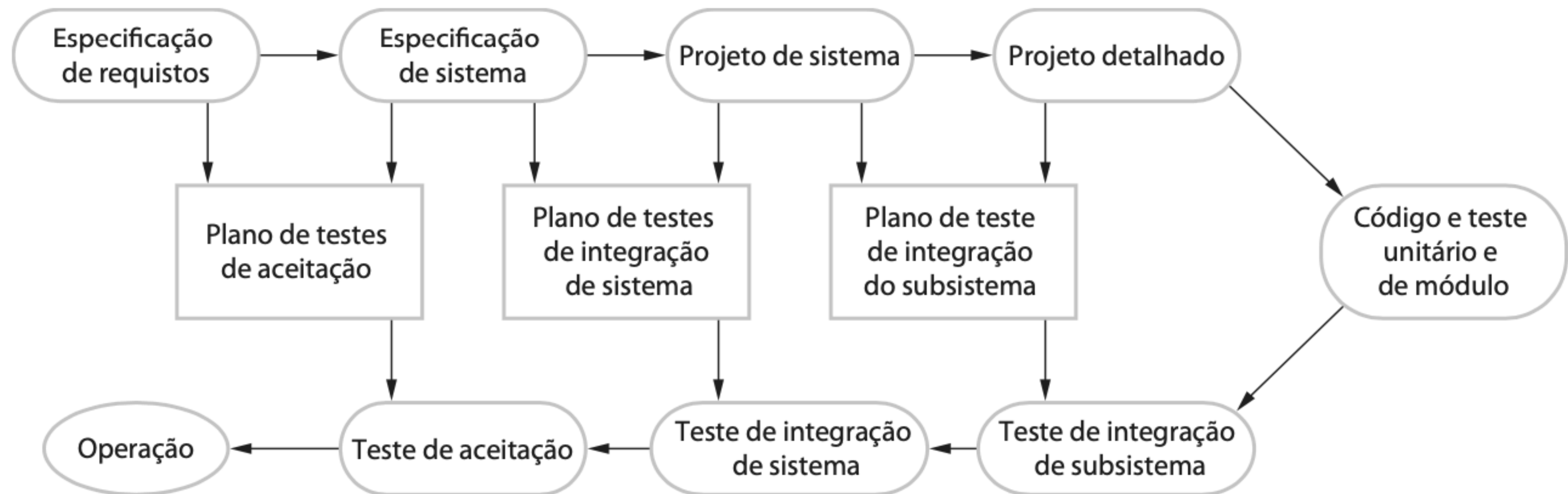
Validação de Software

Validação de software ou, mais genericamente, verificação e validação (V&V), tem a intenção de mostrar que um software se adequa a suas especificações ao mesmo tempo que satisfaz as especificações do cliente do sistema. Teste de programa, em que o sistema é executado com dados de testes simulados, é a principal técnica de validação. A validação também pode envolver processos de verificação, como inspeções e revisões, em cada estágio do processo de software, desde a definição dos requisitos de usuários até o desenvolvimento do programa. Devido à predominância dos testes, a maior parte dos custos de validação incorre durante e após a implementação.

Validação de Software

Figura 2.7

Fases de testes de um processo de software dirigido a planos



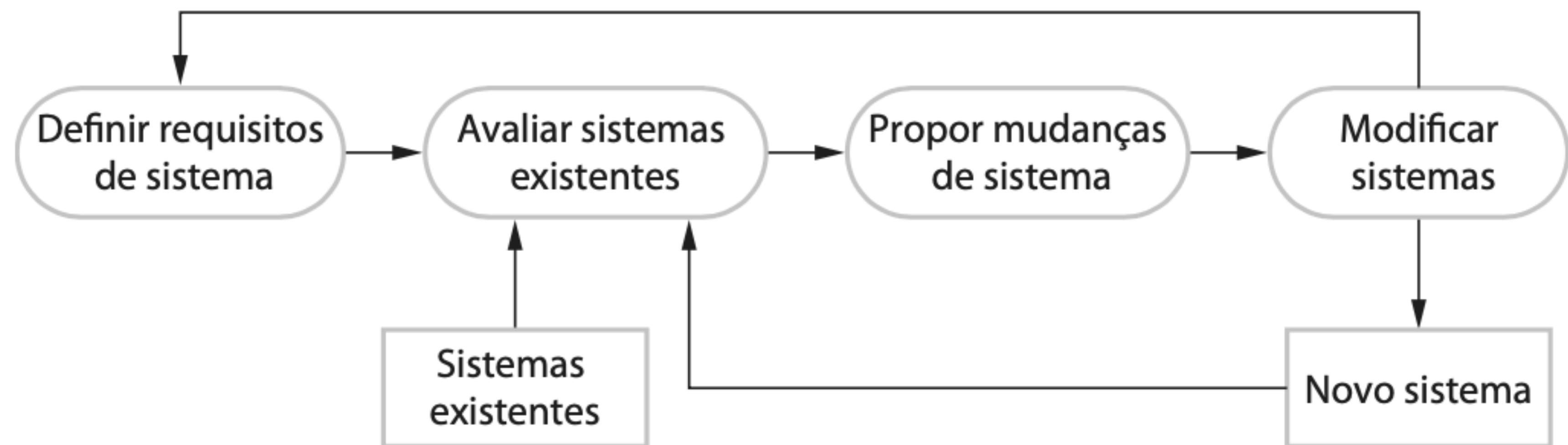
Evolução do Software

A flexibilidade dos sistemas de software é uma das principais razões pelas quais os softwares vêm sendo, cada vez mais, incorporados em sistemas grandes e complexos. Uma vez que a decisão pela fabricação do hardware foi tomada, é muito caro fazer alterações em seu projeto. Entretanto, as mudanças no software podem ser feitas a qualquer momento durante ou após o desenvolvimento do sistema. Mesmo grandes mudanças são muito mais baratas do que as correspondentes alterações no hardware do sistema.

Evolução do Software

Figura 2.8

Evolução do sistema



Bibliografia:

PAULA FILHO, Wilson de Pádua. Engenharia de Software: Fundamentos, Métodos e Padrões. LTC, 2009.

PRESSMAN, R. S. Engenharia de software. 6.ed. McGraw-Hill, 2006.

SOMMERVILLE, I. Engenharia de software. 8.ed. Addison Wesley, 2007.

SBROCCO, José Henrique Teixeira de Carvalho; MACEDO, Paulo Cesar de. Metodologias ágeis : engenharia de software sob medida . São Paulo: Érica, 2012.