

# Engenharia de software baseada em componentes: definição, conceituação e projeto com reuso

DEFINIR E CONCEITUAR O DESENVOLVIMENTO BASEADO EM COMPONENTES BEM COMO O REUSO DE COMPONENTES.

## Engenharia de software baseada em componentes

O desenvolvimento baseado em componentes ou CBD – component-based development também é conhecido como component-based software engineering (CBSE) ou, simplesmente, como componente de software.

Atualmente é prática comum os desenvolvedores utilizarem componentes de software que são encontrados em bibliotecas de uso gratuito ou mesmo disponíveis para compra. Esses componentes são conhecidos como COTS – Commercial-Off-The-Shelf, ou software comercial de prateleira.

Geralmente, eles oferecem funcionalidades com interfaces bem definidas que podem ser facilmente integrados no software desenvolvido.

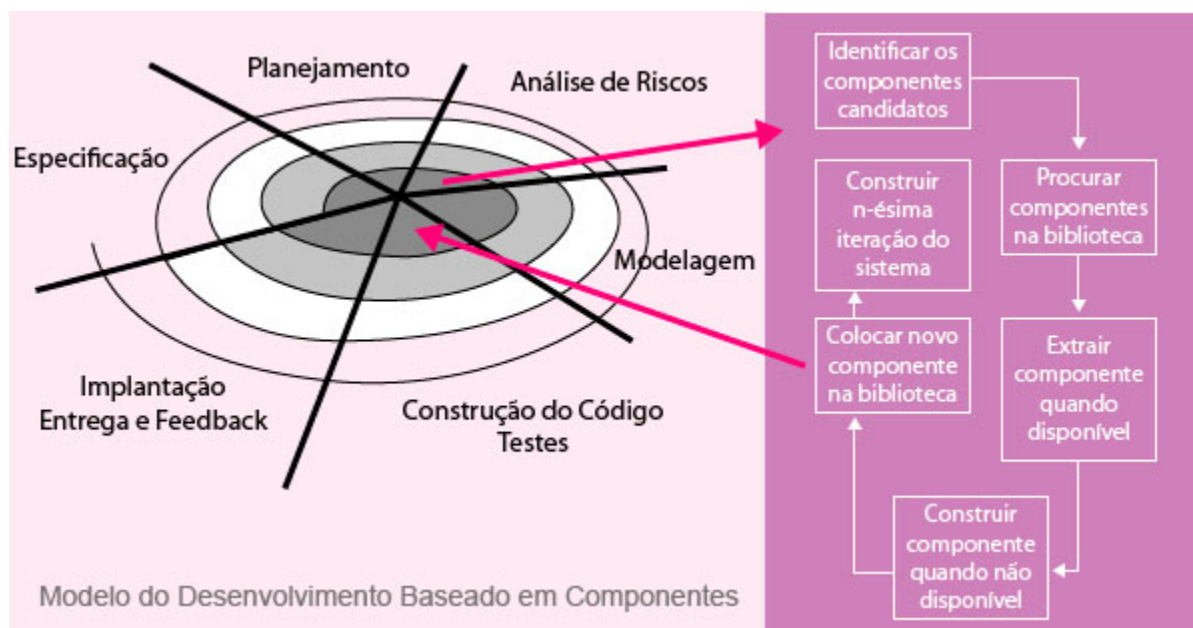
O método de desenvolvimento baseado em componentes pode, por exemplo, incorporar características de construção de componentes de biblioteca ao modelo espiral (como em vários outros modelos de processo). De natureza evolucionária, ele adota uma abordagem iterativa para a criação de software. Assim, o modelo constrói aplicações a partir de componentes de software previamente preparados.

Nas atividades de modelagem e construção é necessário identificar os componentes candidatos, pois eles podem ser projetados como módulos de software convencional, como classes ou pacotes de classes orientadas a objeto.

Sabemos que no paradigma orientado a objetos uma classe encapsula dados e algoritmos, que também podem ser utilizados para manipular os dados.

Por meio desta abordagem, uma biblioteca de classes pode ser construída com as classes identificadas no desenvolvimento do software e a partir de toda iteração da espiral deverá verificar o conteúdo da biblioteca que pode ser reutilizado.

A figura a seguir ilustra este processo, utilizando o modelo espiral como base.



O modelo de desenvolvimento baseado em componentes leva ao reuso de software, o qual fornece aos desenvolvedores vários benefícios.

Com base nesta reusabilidade, estudos relatam uma redução de 70% no prazo do ciclo de desenvolvimento e de 84% no custo dos projetos desenvolvidos neste modelo de processo, com índices de produtividade em torno de 26,2 superior ao índice padrão de 16,9 da indústria. [PRESSMAN, 2006].

## Reuso de software

O reuso de software se refere à utilização de software existente para o desenvolvimento de um novo. A decisão quanto à utilização de componentes reutilizáveis envolve comparações entre o investimento necessário para a sua aquisição e os gastos para o desenvolvimento de uma solução customizada.

Devem ser estimados os custos e benefícios líquidos no investimento em reuso de software e serem avaliados os ganhos com a adoção do reuso.

Muitos métodos para o reuso estão em prática no mercado atual, e sua escolha é determinada pela sua adequação ao modelo de negócio ou às tecnologias utilizadas.

O RiSE – Reuse in Software Engineering, por exemplo, é um esforço em direção a um efetivo framework de métodos, processos, ferramentas e boas práticas relacionados ao reuso de software que compreende aspectos técnicos e não técnicos.

O CRUISE - Component Reuse in Software Engineering, no formato de livro online e com foco em reuso de software, é um dos primeiros esforços em mapear o reuso de software que cita as áreas chave, desenvolvimento baseado em componentes, reengenharia, ferramentas, qualidade de componentes de software, métricas para reuso, repositórios, engenharia de domínio, linhas de produtos, entre outros aspectos.

O RAD – Rapid Application Development é um modelo de processo de desenvolvimento de software iterativo e incremental, o qual enfatiza um ciclo de desenvolvimento extremamente curto (entre 60 e 90 dias). Seu foco é considerável no reuso a fim de atingir um prazo justo e o tornar técnica de quarta geração, a qual reutiliza componentes de programas existentes ou cria componentes reutilizáveis.

Agora que você já estudou esta aula, resolva os exercícios e verifique seu conhecimento. Caso fique alguma dúvida, leve a questão ao Fórum e divida com seus colegas e professor.

EXERCÍCIOS ([https://ead.uninove.br/ead/disciplinas/impressos/\\_g/pdsoft80\\_100/a07ex01\\_pdsoft80\\_100.pdf](https://ead.uninove.br/ead/disciplinas/impressos/_g/pdsoft80_100/a07ex01_pdsoft80_100.pdf))

EXERCÍCIOS ([https://ead.uninove.br/ead/disciplinas/web/\\_g/pdsoft80\\_100/a07ex01\\_pdsoft80\\_100.htm](https://ead.uninove.br/ead/disciplinas/web/_g/pdsoft80_100/a07ex01_pdsoft80_100.htm))

## REFERÊNCIA

PRESSMAN, R. S. Engenharia de software. 7ª ed. São Paulo: McGraw-Hill, 2010.

SOMMERVILLE, Ian. Engenharia de software. São Paulo: Addison-Wesley, 2007.



