

Resumo Cap. 16 Software Engineering - Ian Sommerville

O Capítulo aborda o conceito de reutilização de software, destacando as estratégias, vantagens e dificuldades associadas a essa prática. A reutilização de software refere-se ao uso de componentes já existentes no desenvolvimento de novos sistemas, algo que se tornou mais comum desde o ano 2000, impulsionado por demandas por custos mais baixos, entregas mais rápidas e uma qualidade superior no desenvolvimento de software.

Primeiramente, o uso de componentes já testados e utilizados em outros contextos aumenta a confiabilidade do software, uma vez que os erros de design e implementação já foram identificados e corrigidos. Além disso, o risco no processo de desenvolvimento é reduzido, pois os custos associados ao software reutilizado são conhecidos, ao contrário das incertezas que existem no desenvolvimento de software novo. Outra vantagem significativa é o desenvolvimento acelerado, já que menos tempo é gasto na construção e validação de novas partes do sistema. Também há a questão da conformidade com padrões: a reutilização pode facilitar o cumprimento de padrões da indústria, como interfaces de usuário, o que melhora a experiência dos usuários e reduz o número de erros.

Apesar dessas vantagens, um problema recorrente é o aumento dos custos de manutenção, especialmente se o código-fonte dos componentes reutilizados não estiver disponível, o que pode gerar incompatibilidades com o tempo. Além disso, algumas ferramentas de desenvolvimento não se integram bem com práticas de reutilização, o que complica o uso de bibliotecas de componentes. Outro obstáculo é cultural: muitos desenvolvedores preferem escrever código novo em vez de reutilizar componentes existentes, seja por falta de confiança nos componentes disponíveis ou pela crença de que podem criar soluções melhores por conta própria. O gerenciamento de uma biblioteca de componentes reutilizáveis também representa um investimento significativo, tanto em termos de criação quanto de manutenção, e encontrar, entender e adaptar esses componentes pode ser um processo complexo e demorado.

O capítulo também descreve várias abordagens para a reutilização de software. Frameworks de aplicação, por exemplo, oferecem uma estrutura base que pode ser estendida para criar aplicações específicas. Esses frameworks geralmente são compostos por classes abstratas e concretas, que devem ser adaptadas pelos desenvolvedores para atender às necessidades de um sistema específico. Outra abordagem é a criação de linhas de produtos de software, que consiste em um conjunto de aplicações relacionadas que compartilham uma arquitetura comum, mas podem ser personalizadas para diferentes usuários ou contextos. A reutilização de produtos comerciais prontos, conhecidos como COTS (Commercial Off-The-Shelf), também é discutida. Esses produtos podem ser configurados ou integrados para formar um sistema, embora esse tipo de reutilização possa apresentar desafios adicionais, como a falta de controle sobre as funcionalidades e a dependência dos fornecedores para suporte.

A reutilização de software, apesar de promissora, requer uma estratégia bem planejada e a aceitação de compromissos. Enquanto frameworks e linhas de produtos podem facilitar a criação rápida de novos sistemas, integrar e configurar produtos prontos (como sistemas ERP) pode ser desafiador, especialmente se houver um descompasso entre os modelos de negócios da organização e as suposições embutidas nos produtos. Assim, a reutilização de software pode resultar em grande economia de tempo e custos, mas as dificuldades associadas exigem que as organizações considerem cuidadosamente suas opções e as necessidades de longo prazo antes de optar por esse caminho.