7.3.1 Reuso

De 1960 a 1990, o software mais novo foi desenvolvido a partir do zero. Todos os codigos foram escritos em

linguagem de programacao de alto nivel. O unico reuso significativo de software foi o de funcoes e objetos em

bibliotecas de linguagem de programacao. No entanto, os custos e a pressao de cronograma significavam que

essa abordagem se tornava cada vez menos viavel, especialmente para sistemas comerciais e baseados na Internet.

Consequentemente, surgiu uma abordagem de desenvolvimento baseada no reuso de softwares existentes.

Atualmente, esta e usada com frequencia para sistemas de negocios, softwares cientificos e, cada vez mais, em

engenharia de sistemas embutidos.

O reuso de software e possivel em varios niveis diferentes:

1. O nível de abstração. Nesse nivel, voce nao reusa o software diretamente, mas usa o conhecimento das abstracoes

de sucesso no projeto de seu software. Os padroes de projeto e de arquitetura (abordados no Capitulo 6)

sao formas de representar o conhecimento abstrato para reuso.

2. O nível de objeto. Nesse nivel, voce reusa objetos diretamente de uma biblioteca em vez de escrever um codigo.

Para implementar esse tipo de reuso, voce tem de encontrar bibliotecas adequadas e descobrir se os objetos e

metodos oferecem a funcionalidade que voce precisa. Por exemplo, se voce precisa processar mensagens de

correio em um programa Java, voce pode usar objetos e metodos de uma biblioteca JavaMail.

3. O nível de componentes. Componentes sao colecoes de objetos e classes de objetos que funcionam em conjunto

para fornecer funcoes e servicos relacionados. Muitas vezes, voce tem de se adaptar e ampliar o componente

adicionando um codigo proprio. Um exemplo de reuso em nivel de componente e aquele no qual voce

constroi sua interface de usuario usando um framework. Este e um conjunto de classes de objetos em geral que

implementam manipulacao de eventos, gerencimento de displays etc. Voce adiciona as conexoes com os dados

a serem exibidos e escreve o codigo para definir detalhes especificos do display, como o iayout da tela e as cores.

4. O nível de sistema. Nesse nivel, voce reusa os sistemas de aplicacao inteiros, o que geralmente envolve algum

tipo de configuracao desses sistemas. Essas configuracoes podem ser feitas por meio da adicao e modificacao

do codigo (se voce estiver reusando uma linha de produtos de software) ou pelo uso de interface de configuracao

do proprio sistema. A maioria dos sistemas comerciais e criada dessa forma, em que sistemas genericos

de COTS (commercialoff-the-shelf) sao adaptados e reusados. As vezes, essa abordagem pode envolver o reuso

e a integracao de diversos sistemas para criar um novo.

Ao reusar softwares existentes, voce pode desenvolver novos sistemas mais rapidamente, com menos riscos de

desenvolvimento e custos mais baixos. Como o software reusado foi testado em outras aplicacoes, deve ser mais

confiavel que o novo software. No entanto, existem custos associados ao reuso:

1. Os custos de tempo gasto na procura do software para reuso e na avaliacao sobre ele atender ou nao as necessidades.

Ta Ivez voce precise testar o software para ter certeza de que vai func ionar em seu ambiente, sobretudo

se ele for diferente do ambiente de desenvolvimento.

2. Quando se aplicam os custos de aquisicao do software reusavel. Para grandes sistemas de prateleira, esses

custos podem ser muito elevados.

3. Os custos de adaptacao e configuracao dos componentes de software reusavel ou sistemas para refletir os

requisitos do sistema que voce esta desenvolvendo.

4. Os custos de integracao de componentes de software reusavel (se voce estiver usando software de diferentes

fontes) com o novo codigo que voce desenvolveu. A integracao de softwares reusaveis de diferentes fornecedores

pode ser dificil e cara, pois os fornecedores podem fazer suposicoes conflitantes sobre como seus

respectivos softwares serao reusados.

Como reusar o conhecimento e o software existente deve ser a primeira coisa em que voce deve pensar ao iniciar

um projeto de desenvolvimento de software. Voce deve considerar as possibilidades de reuso antes de projetar

o software em detalhes, assim como pode querer adaptar seu projeto para reusar ativos de software existentes.

Como discutido no Capitulo 2, em um processo de desenvolvimento orientado ao reuso, voce procura elementos

reusaveis e, em seguida, altera seu projeto e requisitos para fazer melhor uso deles.

Para um grande numero de sistemas de aplicacoes, a engenharia de software realmente significa o reuso de

software. Por isso, eu dedico varios capitulos, na secao de tecnologias de software deste livro, a esse topico (capitulos

16,1 7e 19).

7.3.2 Gerenciamento