

Puc Minas
Engenharia de Software

**Gabriel Ramos, João Pedro Braga, João Vitor Romero, Lucas Randazzo,
Lúcio Alves**

IMPACTO DA REUTILIZAÇÃO DE SOFTWARE NA REDUÇÃO DE CUSTOS

Análise econômica sobre o uso de componentes, frameworks e bibliotecas no
ciclo de vida do desenvolvimento.

Trabalho apresentado à disciplina de Eng. Eco-
nômica para Software, Professor Michelle.

Sumário

1	Introdução	2
2	Marco Teórico	2
2.1	Conceitos de Reutilização de Software	2
2.2	Modelos Econômicos de Reutilização	2
2.2.1	Cálculo do Retorno sobre Investimento (ROI)	3
2.3	Impacto na Qualidade e Manutenção	3
3	Exemplos e Aplicações	4
3.1	Estudo de Caso: Engenharia de Linha de Produtos (SPL)	4
3.2	Aplicação de Frameworks no Mercado Web (React/Angular)	4
4	Conclusão	5

1 Introdução

A indústria de software enfrenta historicamente o desafio de entregar sistemas complexos com orçamentos limitados e prazos apertados. Dentro da Engenharia de Software, a economia da produção (*Software Economics*) busca estratégias para otimizar essa relação custo-benefício. Neste contexto, a reutilização de software — o processo de criar sistemas de software a partir de ativos pré-existentes em vez de construí-los do zero — emerge como uma das alavancas mais poderosas para a redução de custos.

O presente estudo analisa como a reutilização de componentes, frameworks e bibliotecas impacta diretamente o Custo Total de Propriedade (TCO) de um projeto. O objetivo é demonstrar que a reutilização não apenas diminui o esforço de codificação inicial, mas gera economias de escala significativas nas fases de testes e manutenção, utilizando métricas econômicas fundamentadas em literatura especializada como a *IEEE Transactions on Software Engineering*.

2 Marco Teórico

2.1 Conceitos de Reutilização de Software

A reutilização de software pode ser classificada em diversos níveis de granularidade. Segundo a literatura clássica e artigos recentes da *ACM Transactions on Software Engineering and Methodology*, destacam-se:

- **Reutilização ad-hoc:** Cópia de trechos de código (o famoso "copy-paste"), pouco gerenciável.
- **Reutilização sistemática:** Uso planejado de bibliotecas, componentes COTS (*Commercial Off-The-Shelf*) e Frameworks.
- **Engenharia de Domínio:** Criação de ativos reutilizáveis específicos para uma família de produtos.

2.2 Modelos Econômicos de Reutilização

Para melhor visualização dos modelos teóricos, temos abaixo uma simulação comparativa entre dois projetos de software com o mesmo escopo funcional (tamanho estimado em 10.000 linhas de código - KLOC), mas com estratégias de desenvolvimento distintas.

A Tabela 1 compara o Cenário A (Desenvolvimento *Greenfield* - sem reutilização) com o Cenário B (Desenvolvimento baseado em Componentes - 50% de reutilização).

Tabela 1: Comparativo de Custos de Desenvolvimento (Simulação)

Métrica	Cenário A	Cenário B
	(Sem Reutilização)	(Com Reutilização)
Tamanho Total do Sistema	10.000 LOC	10.000 LOC
Taxa de Reutilização	0%	50%
Código Novo a Desenvolver	10.000 LOC	5.000 LOC
Custo Unitário (Código Novo)	\$ 20,00 / LOC	\$ 20,00 / LOC
Custo Unitário (Reutilização)*	N/A	\$ 4,00 / LOC
Custo de Desenvolvimento	\$ 200.000,00	\$ 100.000,00
Custo de Aquisição/Adaptação	\$ 0,00	\$ 20.000,00
Custo Total do Projeto	\$ 200.000,00	\$ 120.000,00
Economia Gerada	-	\$ 80.000,00

*Nota: O custo de reutilização (US\$ 4,00) é estimado em 20% do custo de desenvolvimento de novo código, cobrindo despesas de licença, busca, entendimento e testes de integração.

2.2.1 Cálculo do Retorno sobre Investimento (ROI)

A eficácia econômica da estratégia de reutilização pode ser mensurada através do ROI. Utilizando os dados da Tabela 1, aplicamos a fórmula padrão de ROI para Engenharia de Software:

$$ROI = \frac{Economia - Custo_{investimento}}{Custo_{investimento}} \times 100$$

Considerando que o "Custo de Investimento" no cenário de reutilização foi o esforço de aquisição e adaptação (\$ 20.000,00) e a "Economia" foi a diferença bruta de desenvolvimento (\$ 100.000,00 deixaram de ser gastos em codificação nova), temos:

$$ROI = \frac{100.000 - 20.000}{20.000} \times 100 = 400\%$$

Análise: O cálculo demonstra que, para cada dólar investido na adaptação de componentes reutilizáveis neste cenário, o projeto obteve um retorno de 4 dólares em valor de software entregue. Além do ganho financeiro, a redução de 5.000 linhas de código novo implica estatisticamente em menos defeitos inseridos, reduzindo custos futuros de manutenção não contabilizados nesta tabela inicial.

2.3 Impacto na Qualidade e Manutenção

A redução de custos não ocorre apenas na codificação. Componentes reutilizados, especialmente aqueles de bibliotecas maduras ou *open source* consolidadas, já foram exaus-

tivamente testados por uma comunidade ou por usos anteriores.

- **Redução de Testes:** O esforço de teste unitário é drasticamente reduzido, focando-se apenas nos testes de integração.
- **Manutenção:** A manutenção corretiva tende a ser menor devido à maior confiabilidade dos módulos (menor densidade de defeitos).

3 Exemplos e Aplicações

3.1 Estudo de Caso: Engenharia de Linha de Produtos (SPL)

Um exemplo acadêmico e industrial robusto é a Engenharia de Linha de Produtos de Software (SPL). Empresas automotivas e de eletrônicos utilizam SPL para desenvolver variações de software para diferentes modelos de carros ou dispositivos.

Cenário: Uma montadora desenvolve um núcleo comum (plataforma) que cobre 80% das funcionalidades do software embarcado.

Resultado Econômico: Ao invés de desenvolver 10 softwares diferentes para 10 carros, desenvolve-se 1 núcleo e 10 pequenas variações.

Dado: Segundo estudos compilados na *IEEE Transactions*, empresas que adotam SPL reportam redução de até 60% no tempo de colocação no mercado (*Time-to-Market*) e redução de custos de desenvolvimento na ordem de 50% após o terceiro produto lançado na linha.

3.2 Aplicação de Frameworks no Mercado Web (React/Angular)

No mercado de desenvolvimento web, a utilização de frameworks como React ou Angular exemplifica a reutilização horizontal.

Aplicação: Uma empresa financeira precisa criar um novo portal de Internet Banking.

Sem Reutilização: Desenvolveria mecanismos de renderização de DOM, gestão de estado e segurança do zero.

Com Reutilização: Utiliza o framework (custo zero de licença, alto reuso).

Ganho: A equipe foca 100% nas regras de negócio bancárias. O custo de manutenção da base tecnológica (o framework) é externalizado para a comunidade mantenedora (ex: Meta/Google), gerando uma economia de manutenção de longo prazo incalculável.

4 Conclusão

A análise realizada demonstra que a reutilização de software é um fator crítico para a viabilidade econômica de projetos modernos. A redução de custos transcende a fase de codificação, impactando positivamente a qualidade (reduzindo custos de correção de falhas) e o tempo de entrega.

Entretanto, conclui-se também que a reutilização exige investimento inicial e governança. Não há "almoço grátis": a reutilização indiscriminada sem avaliação de compatibilidade ou licenciamento pode gerar dívida técnica. O ganho econômico real, mensurado pelo ROI (*Retorno sobre o Investimento*), é maximizado quando a reutilização é tratada como um processo sistemático e estratégico, e não apenas como um atalho operacional.

Referências

- [1] BASILI, V. R.; ROMBACH, H. D. Support for Comprehensive Reuse. *IEEE Transactions on Software Engineering*, v. 17, n. 4, p. 303-316, 1991.
- [2] LIM, W. C. Effects of Reuse on Quality, Productivity, and Economics. *IEEE Software*, v. 11, n. 5, p. 23-30, 1994.
- [3] KRUEGER, C. W. Software Reuse. *ACM Computing Surveys*, v. 24, n. 2, p. 131-183, 1992.
- [4] POHL, K.; BÖCKLE, G.; VAN DER LINDEN, F. *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer, 2005.
- [5] Information Systems and eBusiness Management.