**Texto base**

**6**

**Impacto das Mudanças**

Prof. João de Deus Freire Junior

***Resumo***

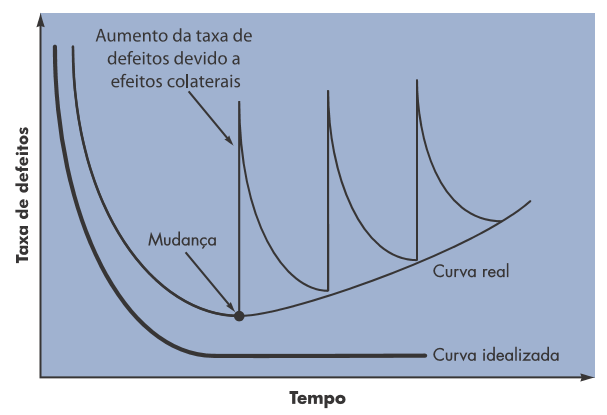
*Uma mudança, quando solicitada tardiamente num projeto, pode ser maior do que mais do que uma ordem de magnitude mais dispendiosa do que a mesma mudança solicitada nas fases iniciais*. *As mudanças impactam muito o andamento dos projetos podem inclusive inviabilizar a continuidade de um projeto devido ao grande impacto no todo. Por isso, as fases iniciais de levantamento de requisitos de um projeto são essenciais para seu sucesso.*

**1.1. Introdução**

Posso solicitar mudanças no software que está sendo desenvolvido quando eu quiser? As mudanças são facilmente recebidas e absorvidas sem impacto pela equipe de desenvolvimento? O impacto das mudanças diferem de acordo com o momento que são solicitadas? Essas perguntas serão respondidas nesta aula. É vital entendermos qual o impacto das solicitações de mudanças no ciclo de desenvolvimento de software.

**1.2. Curva de defeitos do software**

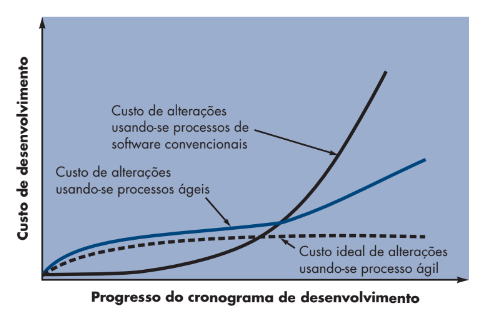
A curva de defeitos de um software é bastante peculiar e tem um comportamento bem diferente da curva de defeitos de um hardware. Ela apresenta uma alta taxa de defeitos no início do ciclo de vida do software, assim como ocorre com o hardware. No entanto, ao contrário do hardware, a curva de defeitos do software não se estabiliza ao longo do tempo. Isso se deve ao fato de que o software está sujeito a múltiplas mudanças durante o seu ciclo de vida. Essas mudanças ocorrem para corrigir defeitos identificados tardiamente, implementar melhorias no software, realizar adaptações e promover evoluções. A cada mudança realizada, como efeito colateral, há um aumento na taxa de defeitos do software. Embora o software possa ser estabilizado, o processo se reinicia com uma nova mudança. A figura 1.1 ilustra a curva de defeitos do software ao longo de seu ciclo de vida.



**Figura 1.1. Curva de defeitos de software**

**1.3. Impacto das mudanças**

O conhecimento acumulado de desenvolvimento de software (baseada em décadas de experiência) afirma que o impacto de mudanças aumentam de forma não linear conforme o projeto avança (Figura 1.2, curva em preto contínuo). É de certa forma fácil absorver uma mudança quando uma equipe de software está levantando requisitos (no início de um projeto). Pode-se ter de alterar um detalhamento do uso, de uma funcionalidade, ampliar uma lista de funções, alterar algo relacionado ao desenho das interfaces com o usuário ou editar uma especificação por escrito. Os custos de tal trabalho são mínimos e o tempo demandado não afetará negativamente o resultado do projeto. Mas, se adiantarmos alguns meses, o que aconteceria? O time de desenvolvimento estará em meio aos testes de validação (que ocorrem relativamente no final do projeto) e um importante interessado está requisitando uma mudança funcional grande, que muitas vezes provoca mudanças estruturais no software. A mudança requer uma alteração no projeto da arquitetura do software, o projeto e desenvolvimento de vários novos componentes, modificações em vários outros componentes, especificação de novos testes, e assim por diante. Os impactos crescem rapidamente e não serão triviais o tempo e custos necessários para assegurar que a mudança seja feita sem efeitos colaterais inesperados. (PRESSMAN, 2011)



**Figura 1.2. Custos de alterações como uma função do tempo em desenvolvimento**

**1.4. Você sabia?**

Há vários exemplos conhecidos de grandes projetos impactados por mudanças ou defeitos identificados na fase de manutenção do software em produção. Segue nos próximos tópicos alguns exemplos.

**1.4.1. Espaçonave da NASA**

Em sua missão a Marte em 1998, a espaçonave Climate Orbiter acabou perdida no espaço. Embora a falha tenha confundido os engenheiros por algum tempo, foi revelado que um subcontratado da equipe de engenharia não conseguiu fazer uma conversão simples de unidades inglesas para métricas. Um lapso embaraçoso que enviou a nave de US $125 milhões fatalmente para perto da superfície de Marte, após tentar estabilizar sua órbita muito baixa. Os controladores de voo acreditam que a espaçonave invadiu a atmosfera de Marte, onde as tensões associadas prejudicaram suas comunicações, deixando-a voando pelo espaço em uma órbita ao redor do sol.

**1.4.2. Ariane 5**

O mais novo foguete de lançamento de satélite não tripulado da Europa utilizou o software funcional de seu antecessor, o Ariane 4. Infelizmente, os motores mais rápidos do Ariane 5 exploraram um bug que não foi encontrado nos modelos anteriores. Trinta e seis segundos após o lançamento inicial, os engenheiros do foguete apertaram o botão de autodestruição após várias falhas do computador. Em essência, o software tentou amontoar um número de 64 bits em um espaço de 16 bits. As condições de estouro resultantes travaram os computadores principal e de backup (que estavam executando exatamente o mesmo software).

O Ariane 5 havia custado quase US $8 bilhões para ser desenvolvido e carregava uma carga útil de satélite de US $500 milhões quando explodiu.

**1.4.3. Explosão de gasoduto soviético**

O oleoduto soviético tinha um nível de complexidade que exigiria um software de controle automatizado avançado. A CIA (Central de Inteligência dos Estados Unidos) foi informada das intenções soviéticas de roubar os planos do sistema de controle. Trabalhando com a empresa canadense que projetou o software de controle de dutos, a CIA fez com que os projetistas criassem deliberadamente falhas na programação para que os soviéticos recebessem um programa comprometido. Alega-se que, em junho de 1982, falhas no software roubado levaram a uma explosão massiva ao longo de parte do oleoduto, causando a maior explosão não nuclear da história do planeta.

**1.5. Você quer ler?**

Segue duas indicações de estudo complementar. Trata-se de um artigo que explica quanto um erro de software pode custar para as companhias e apresenta vários links de outros artigos que detalham o tema.

* Leia o artigo:

ROBERTO, Jones. Quanto os erros em software podem custar para a sua empresa? 2019.

Disponível em:

<https://medium.com/xp-inc/quanto-os-erros-em-software-podem-custar-para-a-sua-empresa-409b77d08bd6>. Acesso em: 07 jan. 2021.

**1.6. Referências**

PRESSMAN, R. S.(2011) Engenharia de Software: uma abordagem profissional. 7.ed. Porto Alegre: Bookman, 2016.

ROBERTO, Jones. Quanto os erros em software podem custar para a sua empresa? 2019. Disponível em: <<https://medium.com/xp-inc/quanto-os-erros-em-software-podem-custar-para-a-sua-empresa-409b77d08bd6>. Acesso em: 07 jan. 2021.