

# Qualidade de Software

## Gerenciamento de Qualidade de Software

- Dedicar-se a assegurar que o **nível requerido de qualidade** seja atingido em um produto de software
- Envolve a definição de padrões e procedimentos apropriados de qualidade e a garantia de que sejam seguidos
- Deve visar o desenvolvimento de uma “cultura de qualidade”
  - Qualidade vista como uma responsabilidade de todos

## O que é Qualidade?

- Qualidade, de maneira simplista, significa que **um produto deve atender à sua especificação**
- Isso é problemático para os sistemas de software
  - Tensão entre os requisitos de qualidade do cliente (eficiência, confiabilidade, etc.) e requisitos de qualidade do desenvolvedor (facilidade de manutenção, reusabilidade, etc.)
  - Alguns requisitos de qualidade são difíceis de especificar de uma maneira não-ambígua
    - Ex. **Facilidade de uso**
  - As especificações de software são, geralmente, incompletas e frequentemente inconsistentes

## Escopo do Gerenciamento de Qualidade

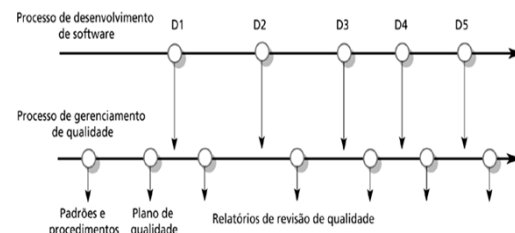
- Gerenciamento de qualidade é particularmente importante para **sistemas grandes e complexos**
- A documentação gerada é um registro do progresso
  - Apóia a continuidade do desenvolvimento quando a equipe de desenvolvimento muda
- Para sistemas menores, o gerenciamento de qualidade precisa de menos documentação
  - Foco em uma **cultura da qualidade**

## Atividade de Gerenciamento de Qualidade

- **Garantia de qualidade**
  - Estabelece procedimentos e padrões **organizacionais** para qualidade.
- **Planejamento de qualidade**
  - Seleciona procedimentos e padrões aplicáveis para um **projeto específico** e o modifica quando necessário.
- **Controle de qualidade**
  - Assegura que os procedimentos e os padrões sejam seguidos pela equipe de desenvolvimento de software.
- O gerenciamento de qualidade deve ser separado do gerenciamento de projeto para assegurar independência
  - Objetivos potencialmente conflitantes!

## Gerenciamento de Qualidade e Desenvolvimento de Software

Figura 27.1 Gerenciamento de qualidade e desenvolvimento de software.



## Qualidade de Processo e de Produto

- A qualidade de um produto é influenciada pela **qualidade do processo de produção**
  - Ex: um processo que exige dados de teste antes do projeto e codificação melhora confiabilidade do produto
  - Alguns modelos de qualidade partem dessa premissa
- Isso é importante no desenvolvimento de software, visto que os atributos de qualidade de produtos são difíceis de avaliar.
- Contudo, existe uma relação **complexa e pouco compreendida** entre processos de software e qualidade de produto.

## Qualidade Baseada em Processo

- Existe uma ligação nítida entre processo e produto nos **bens manufaturados**
- Para o software isso é mais complexo porque:
  - A aplicação de habilidades individuais e a experiência são muito importantes no desenvolvimento de software
  - Fatores externos, como a necessidade de um cronograma de desenvolvimento acelerado, podem piorar a qualidade do produto.
- Deve-se tomar cuidado para não impor padrões de processo inadequados
  - Esses padrões podem reduzir, ao invés de melhorar a qualidade do produto

## Qualidade baseada em processo

Figura 27.2

Qualidade baseada em processo.



## Padrões de Processo na Prática

- Definir padrões de processo, tais como o modo como as revisões devem ser conduzidas, o gerenciamento da configuração, etc.
- Monitorar o processo de desenvolvimento para assegurar que os padrões **estejam sendo seguidos**
- Relatar sobre o processo para a gerência de projeto e para o comprador do software
- Não usar práticas inadequadas simplesmente porque padrões foram estabelecidos

## Padrões de Qualidade

- São a chave para o gerenciamento efetivo de qualidade.
- Podem ser internacionais, nacionais, organizacionais ou de projeto.
- **Padrões de produto** definem características que todos os componentes devem exibir
  - Ex. um estilo de programação comum.
- **Padrões de processo** definem características relativas aos processos de software

## Importância dos Padrões

- Englobam as **melhores práticas**
  - Evitam a repetição de erros do passado.
- São um arcabouço para os processos de garantia de qualidade
- Padrões fornecem continuidade
  - Novos funcionários podem compreender a organização pela compreensão dos padrões que são adotados

## Exemplos de Padrões de Produto e de Processo

Tabela 27.1 Padrões de produto e de processo

Padrões de produto	Padrões de processo
Formulário de revisão de projeto	Conduta de revisão de projeto
Estrutura de documento de requisitos	Envio do documento para o CM
Formato do cabeçalho de método	Processo de liberação de versão
Estilo de programação em Java	Processo de aprovação de plano de projeto
Formato de plano de projeto	Processo de controle de mudança
Formulário de solicitação de mudança	Processo de registro de teste

## Problemas com Padrões

- Podem **não ser vistos como relevantes** ou atualizados pelos engenheiros de software
- Envolvem, frequentemente, **muita burocracia**
- Difícil manter a documentação associada
  - **Ferramentas** são fundamentais!
  - Normalmente a documentação termina ficando **defasada**

## Desenvolvimento de padrões

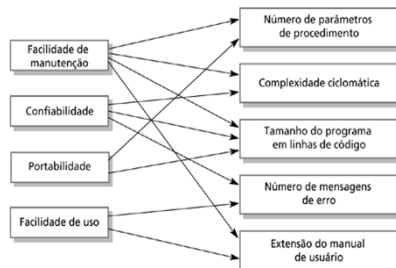
- Envolver os engenheiros na elaboração
  - Eles devem compreender as **razões** de um padrão
- Revisar padrões e seu uso regularmente
  - Padrões podem se tornar rapidamente desatualizados e isso reduz sua credibilidade
  - Incorporar **novas** “melhores práticas”
- Padrões devem ter ferramentas de apoio
  - Trabalho padronizado em excesso é a mais importante reclamação contra os padrões

## Qualidade de software: externa x interna

- Qualidade externa
  - Visível ao usuário do sistema
- Qualidade interna
  - Diz respeito aos desenvolvedores do sistema
- Fatores internos ajudam a obter fatores externos de qualidade

## Atributos Internos e Externos

Figura 27.6  
Relacionamentos entre atributos de software internos e externos.



## Métricas de Software

- A medição de software se dedica a derivar um **valor numérico** para algum atributo de um produto ou de processo de software
- Permite comparações **objetivas** entre técnicas e processos
- Embora algumas empresas tenham introduzido programas de medição, a maioria das organizações ainda não as usam de forma sistemática
- Existem poucos padrões estabelecidos nessa área

## Métricas de Software

- Representam a contagem de determinados elementos de um sistema de software, processo ou documento
  - Ex: Linhas de código em um programa, número de referências a uma variável global
- Permitem que o software e o processo de software sejam **quantificados**
- Podem ser usadas para prever atributos de produto e para controlar o processo de software.
- As métricas de produto podem ser usadas para **previsões gerais** ou para **identificar componentes anômalos**

## Suposições sobre Métricas

- Uma propriedade do software que **pode ser medida**
- Existe um **relacionamento** entre o que podemos medir e o que queremos conhecer
  - Podemos somente medir atributos **internos**, mas estamos, muitas vezes, mais interessados em atributos **externos** de software
    - Ex. Acoplamento menor => maior manutenibilidade?
- Pode ser difícil, a partir do que pode ser medido, inferir as qualidades desejáveis do sistema

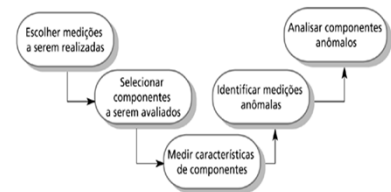
## O Processo de Medição

- Um processo de medição de software pode ser parte do controle de qualidade
  - Algumas métricas são difíceis de se coletar de forma automatizada
    - Ex. Compreensibilidade
- Os dados coletados durante este processo devem ser **mantidos** como um recurso da organização.
  - Torna possível **comparações entre projetos**

## Processo de Medição de Produto

Figura 27.7

Processo de medição de produto.



## Recomendações para o Processo de Medição

- Não colete dados desnecessários
  - As questões a serem respondidas devem ser decididas previamente e os dados necessários identificados
- Conte às pessoas porque os dados estão sendo coletados
  - Evite usá-los como parte da avaliação de pessoal
- Não dependa da memória
  - Colete dados quando são gerados, e não depois que um projeto foi finalizado

## Métricas de Produto

- Uma métrica deve ser um previsor de qualidade de produto.
- Classes de métrica de produto
  - **Dinâmicas**, coletadas em tempo de execução
    - Ajudam a avaliar a eficiência e a confiabilidade
  - **Estáticas**, coletadas em tempo de compilação
    - Ajudam a avaliar a confiabilidade, a facilidade de compreensão e a facilidade de manutenção.
- Algumas métricas são coletadas a partir de informações **sobre** o produto
  - Ex. Número de **bugs conhecidos**

## Análise de Medições

- Nem sempre é óbvio o que os dados significam
  - A **análise** de dados coletados é **muito difícil**
- É muito importante usar métricas **experimentalmente validadas**
- Estatísticos profissionais devem ser consultados, se estiverem disponíveis

## Pontos-chave

- Gerenciamento de qualidade de software dedica-se a assegurar que o software atende aos padrões necessários.
- Padrões de software são um encapsulamento de boas práticas.
- A medição de software obtém informações sobre ambos, o processo e o produto de software.
- Métricas de qualidade de produto devem ser usadas para identificar componentes potencialmente problemáticos.
- Não existem métricas de software padronizadas e universalmente aplicáveis.