



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA



Instituto Federal da Bahia
Análise e Desenvolvimento de Sistemas
INF022 – Tópicos Avançados

Medição de Software


Prof. Dr. Renato L. Novais
renato@ifba.edu.br

Agenda



- Medição de software
- Por que medir?
- Exemplos de Métricas


Análise e Desenvolvimento de Sistemas



Medição de Software

6/14/14 Prof. Renato Novais – IFBA Slide 3/TN

Análise e Desenvolvimento de Sistemas



Medição

Uma ciência é tão madura quanto suas ferramentas de medição.

Louis Pasteur

Aquilo que não pode ser medido não pode ser gerenciado.

Peter Drucker

Não se pode controlar aquilo que não se pode medir.

Tom de Marco

6/14/14 Prof. Renato Novais – IFBA Slide 4/TN

Medição



Medição é o processo pelo qual **números** são atribuídos a **atributos** de **entidades do mundo real** com o **objetivo de descrevê-las de alguma forma**.

Medição



- Exemplos de **entidades** de software
 - o código fonte de um sistema,
 - uma classe,
 - a especificação de requisitos de um sistema,
 - a arquitetura de um sistema,
 - as fases de um projeto,
 - as atividades de um processo.

Medição



- Exemplos de **atributos** de software
 - tamanho (do código ou da especificação),
 - acoplamento (de uma classe ou módulo arquitetural),
 - a duração (de uma fase do projeto),
 - o esforço (gasto em uma fase).


Medição de Software




É o processo contínuo de **definir métricas**, **coletar** e **analisar** dados dos processos e produtos de software, com o objetivo de obter informações significantes para **compreender**, **controlar** e **aperfeiçoar** os **processos** e **produtos**.

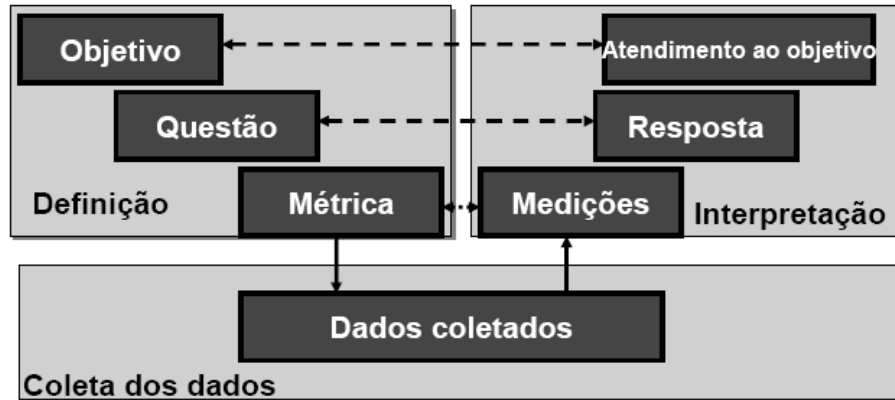
- **Medição de Processo**
 - Quantifica **atributos relacionados às atividades e ao andamento do desenvolvimento de software**
 - Ex.: produtividade (pontoFunção / dia), tempo de correção de defeitos, esforço por fase.

- **Medição de Produto**
 - Quantifica **atributos relacionados aos artefatos de software**
 - Ex.: tamanho do código, acoplamento entre módulos arquiteturais, tamanho da especificação de requisitos.
- Algumas métricas estão relacionadas a processo e produto
 - Ex: Taxa de defeitos por fase do desenvolvimento

Análise e Desenvolvimento de Sistemas	Medição		
	<ul style="list-style-type: none">• Não medir por medir<ul style="list-style-type: none">– Toda medição deve ter um objetivo claro– Meça somente se for relevante para a organização ou projeto		
	6/14/14	Prof. Renato Novais – IFBA	Slide 11/TN


Análise e Desenvolvimento de Sistemas	Goal-Question-Metric (GQM)		
	<ul style="list-style-type: none">• Define uma sistemática para planejar e executar atividades de medição com base em objetivos;• Inicia com a declaração explícita das metas/objetivos do programa de medição;• Dos objetivos, são definidas questões;• Das questões, são definidas métricas. <p style="text-align: center;">Basili, V.R.; Caldiera, G.; Rombach, H.D.; The Goal Question Metric Approach</p>		
	6/14/14	Prof. Renato Novais – IFBA	Slide 12/TN


Goal-Question-Metric (GQM)





Por que medir?



Análise e Desenvolvimento de Sistemas	Por que medir?	
	<ul style="list-style-type: none">• Avaliar e melhorar a qualidade de processos e produtos<ul style="list-style-type: none">– De acordo com diferentes atributos• Entender o negócio e criar visibilidade<ul style="list-style-type: none">– Construir baselines, identificar relacionamentos e fatores críticos• Gerenciar e controlar projetos com base em evidências quantitativas<ul style="list-style-type: none">– Planejar e estimar, acompanhar dados reais com dados estimados	
6/14/14	Prof. Renato Novais – IFBA	Slide 15/TN

Análise e Desenvolvimento de Sistemas	Por que medir?	
	<ul style="list-style-type: none">• Perguntas Possíveis do Nosso Cotidiano<ul style="list-style-type: none">– Quanto vai custar o projeto? Onde estou gastando a verba?<ul style="list-style-type: none">• Crie uma memória corporativa com referências para práticas existentes– O que deve acontecer? O que está acontecendo?<ul style="list-style-type: none">• Planeje, acompanhe e controle o desenvolvimento e evolução do projeto	
6/14/14	Prof. Renato Novais – IFBA	Slide 16/TN

Análise e Desenvolvimento de Sistemas	<h2>Por que medir?</h2>	
	<ul style="list-style-type: none">• Perguntas Possíveis do Nosso Cotidiano<ul style="list-style-type: none">- Determinados tipos de problemas são frequentes e comuns?- Quais técnicas poderão minimizar os problemas, mudar os baselines?- Qual a confiabilidade do produto após sua liberação?	
	6/14/14	Prof. Renato Novais – IFBA Slide 17/TN

Análise e Desenvolvimento de Sistemas	<h2>Por que medir?</h2>	
	<ul style="list-style-type: none">• Para evitar “achologia”<ul style="list-style-type: none">- “todo mundo usa ou faz”, não é argumento• Você sabia que dos “métodos” e técnicas ditos inovadores:<ul style="list-style-type: none">- mais de 50% nunca foram experimentadas?- quando o foram, usualmente foi um singelo estudo de caso (<i>proof of concept</i>) sem casos de controle?• Glass, R.L.; <i>Facts and Fallacies of Software Engineering</i>; New York, NY: Addison-Wesley; 2003	
	6/14/14	Prof. Renato Novais – IFBA Slide 18/TN

Por que medir?





- Você sabia que as vantagens advogadas, em grande parte **não resistem a experimentos controlados**?
 - qual é mesmo a vantagem do uso de UML?
 - Meça o desenvolvimento e evolução com UML e compare com outras linguagens de representação
 - meça com e sem editores
 - meça com e sem geradores
- Glass, R.L.; *Facts and Fallacies of Software Engineering*; New York, NY: Addison-Wesley; 2003


Por que medir?




- Precisamos medir **processos** de software
 - para **constatar experimentalmente** que as *técnicas, os métodos e os processos efetivamente possuem as vantagens apregoadas*.
 - para **poder aprimorar** as técnicas, os métodos e os processos utilizados.
 - para **poder prever riscos** e benefícios da adoção de determinada técnica, método ou processo.

Análise e Desenvolvimento de Sistemas	Por que medir?	
	<ul style="list-style-type: none">• Precisamos medir produtos de software para<ul style="list-style-type: none">– identificar as causas (riscos) do mau funcionamento<ul style="list-style-type: none">• arquitetura ou projeto inadequados• testes insuficientes• complexidade de especificações• código inutilmente complexo– identificar riscos para evolução<ul style="list-style-type: none">• muito acoplamento• má estruturação	
6/14/14	Prof. Renato Novais – IFBA	Slide 21/TN

Análise e Desenvolvimento de Sistemas	Por que medir?	
	<ul style="list-style-type: none">• Precisamos medir produtos de software para<ul style="list-style-type: none">– identificar características dos produtos que ajudem a prever<ul style="list-style-type: none">• custo• tempo de desenvolvimento• riscos de defeitos– identificar características das variações<ul style="list-style-type: none">• proficiência da equipe, do indivíduo• condições de trabalho• ferramenta disponível	
6/14/14	Prof. Renato Novais – IFBA	Slide 22/TN

Análise e Desenvolvimento de Sistemas	Cuidados	
	<ul style="list-style-type: none">• Não medir demais cedo demais<ul style="list-style-type: none">- inicie programas de medição gradativamente- meça somente depois de saber por que medir<ul style="list-style-type: none">• a medida que for identificando objetivos relevantes• somente realize medições complexas depois que souber tratar os resultados• somente realize medições se o custo / benefício estimado valer a pena	
6/14/14	Prof. Renato Novais – IFBA	Slide 23/TN

Análise e Desenvolvimento de Sistemas	Cuidados	
	<ul style="list-style-type: none">• Não medir de menos tarde demais<ul style="list-style-type: none">- se existe um objetivo relevante procure atingi-lo- procure responder todas as questões- procure estar informado quanto a problemas com artefatos e processos• Não medir as coisas erradas<ul style="list-style-type: none">- evite atingir objetivos irrelevantes- evite responder questões irrelevantes- evite métricas caras ou complexas demais	
6/14/14	Prof. Renato Novais – IFBA	Slide 24/TN

Cuidados



- Defina bem as métricas
 - métricas e critérios devem estar claramente definidos
 - existindo a possibilidade de medir utilizando procedimentos, desenvolva-os
 - padronize as medições de forma a assegurar independência de observador

Exemplos de Métricas



Qualidade dos Requisitos





- Estabilidade de requisitos
 - #requisitos novos / #requisitos
 - #requisitos alterados ou excluídos / #requisitos
- Taxa de defeitos
 - #requisitos com defeitos / #requisitos
- Taxa de validação
 - #requisitos validados sem problemas / #requisitos


Qualidade dos Requisitos




- Taxa de rastreabilidade
 - #requisitos mapeados a casos de uso / #requisitos
 - #requisitos mapeados a componentes / #requisitos
 - #requisitos mapeados a casos de teste / #requisitos

Análise e Desenvolvimento de Sistemas	Defeitos	
	<ul style="list-style-type: none">• Taxa de defeitos<ul style="list-style-type: none">- #defeitos / Unidade de tamanho- Unidade de tamanho<ul style="list-style-type: none">• KLOC, pontos de função• Taxa de defeitos por release<ul style="list-style-type: none">- #defeitos / KLOC total- #defeitos / KLOC do incremento	
6/14/14	Prof. Renato Novais – IFBA	Slide 29/TN

Análise e Desenvolvimento de Sistemas	Defeitos	
	<ul style="list-style-type: none">• Taxa de defeitos por fase de desenvolvimento<ul style="list-style-type: none">- #defeitos encontrado na fase / KLOC• Taxa de remoção de defeitos por fase<ul style="list-style-type: none">- #defeitos removidos na fase / #defeitos latentes <p style="text-align: center;"><i>defeitos latentes = defeitos removidos na fase + defeitos encontrados depois</i></p>	
6/14/14	Prof. Renato Novais – IFBA	Slide 30/TN

Análise e Desenvolvimento de Sistemas	Defeitos	
	<ul style="list-style-type: none">• Taxa de remoção de defeitos por fase de teste<ul style="list-style-type: none">- #defeitos removidos na fase de teste / #defeitos latentes <p style="text-align: center;"><i>defeitos latentes = defeitos removidos na fase + defeitos encontrados depois</i></p>	
6/14/14	Prof. Renato Novais – IFBA	Slide 31/TN

Análise e Desenvolvimento de Sistemas	Manutenção	
	<ul style="list-style-type: none">• Taxa de problemas resolvidos<ul style="list-style-type: none">- #problemas resolvidos no período / #problemas reportados no período• Tempo de resposta de manutenção<ul style="list-style-type: none">- Média do tempo de resolução de todos os problemas reportados• Taxa de problemas resolvidos no prazo<ul style="list-style-type: none">- #problemas resolvidos no prazo / #problemas resolvidos	
6/14/14	Prof. Renato Novais – IFBA	Slide 32/TN

Manutenção



- Taxa de problemas resolvidos que geraram outro problema
 - $\frac{\text{\#problemas resolvidos que geraram problemas}}{\text{\#problemas resolvidos}}$
- Taxa de artefatos afetados por alteração
 - $\frac{\text{\#artefatos modificados para resolver um problema}}{\text{\#problemas resolvidos}}$

Manutenção



- Taxa de problemas por usuário em um período
 - $\frac{\text{\#problemas reportados pelos usuários}}{\text{\#usuários}}$

Casos de Teste




Análise e Desenvolvimento de Sistemas

- Estabilidade de casos de testes
 - #casos de teste novos / #casos de teste
 - #casos de teste alterados ou excluídos / #casos de teste

6/14/14 Prof. Renato Novais – IFBA Slide 35/TN

Estimativa



Análise e Desenvolvimento de Sistemas

- Qualidade de estimativa de tempo
 - duração real do projeto / duração estimada do projeto
 - duração real do caso de uso / duração estimada do caso de uso
- Qualidade da estimativa de esforço
 - esforço real do caso de uso / esforço estimado do caso de uso

6/14/14 Prof. Renato Novais – IFBA Slide 36/TN

Métricas Estruturais




- Acoplamento entre Classes
 - Conta o número de outras classes com as quais uma classe está acoplada.
- Profundidade da Árvore de Herança
 - Conta quão profundamente na hierarquia de herança uma classe está declarada.
- Falta de Coesão dos Métodos
 - Mede a falta de coesão de uma classe em termos da quantidade de pares de métodos que não acessam pelo menos um atributo em comum.

Métricas Estruturais



- Linhas de Código
 - Conta o número de linhas de código.
- Número de Atributos
 - Conta o número de atributos de cada classe ou aspecto.
- Peso das Operações por Componente
 - Conta o número de métodos de cada classe, mais o peso de cada método (ex: número de parâmetros).

Análise e Desenvolvimento de Sistemas	Perguntas	
	<h1>?</h1>	
	6/14/14	Prof. Renato Novais – IFBA Slide 39/TN

Análise e Desenvolvimento de Sistemas	Referências	
	<ul style="list-style-type: none">• Material fornecido pelo Professor Claudio Sant'anna (UFBA)	
	6/14/14	Prof. Renato Novais – IFBA Slide 40/TN