



*Engenharia de Software*

**Tema da Aula**  
**Estimativas e Métricas - III**

*Prof. Cristiano R R Portella*  
portella@widesoft.com.br

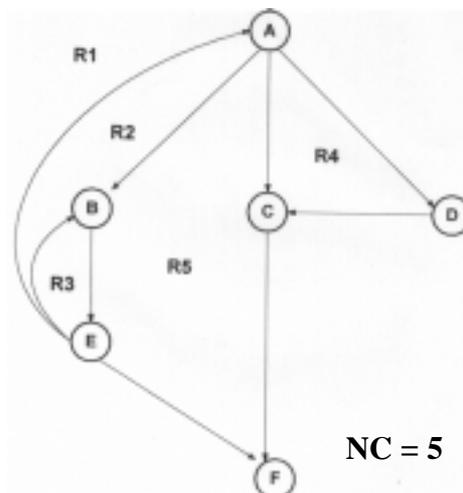


## Métricas de Complexidade

- ✓ Pode-se medir a complexidade de um software a partir de 2 enfoques:
- Medir a complexidade do problema:
    - Funções que o software realizará
    - Domínio da informação
    - Complexidade do algoritmo
  - Medir a complexidade do produto:
    - Diretamente do código fonte

Medir a complexidade do produto:

- ✓ Thomas McCabe criou a teoria da complexidade ciclomática, também chamada de Método Ciclomático.
- ✓ A técnica consiste em contar as regiões de um gráfico (fluxo) resultante da análise do código fonte, feita por ferramentas que implementam o método.
  - Região: área fechada em seus limites ou toda a área externa que cerca o gráfico.
- ✓ Da análise resulta um número chamado NC- Número e complexidade.



## Métricas de Complexidade

Se o número de desvios de um programa cresce



A complexidade cresce



O número de falhas e erros tende a crescer

Assim, o NC é um indicador de problemas (previsor de falhas).

## Métricas de Complexidade

Um módulo/programa com um NC

De 1 a 5	Considerado simples
De 6 a 10	Normal. Se for bem projetado, não deverão ocorrer falhas
De 11 a 20	Complexo. Prováveis problemas
Maior que 20	Altamente complexo. Extremamente problemático.

## Métricas de Qualidade

---

- ✓ Servem para medir a qualidade do produto de software ou do processo de software.
- ✓ Deveriam ser aplicadas durante todo o processo.
- ✓ As principais métricas de qualidade de software aferem:
  - Corretitude
  - Manutibilidade
  - Integridade
  - Usabilidade

## Métricas de Qualidade

---

- ✓ Corretitude:  
É o grau em que o software executa as funções que são exigidas (conformidade com os requisitos).  
Ex.: Número de defeitos/LOC's
- ✓ Manutibilidade:  
Facilidade com que o programa pode ser corrigido (erros), alterado (adaptação) ou ampliado (novas necessidades do cliente).  
Ex.: Tempo Médio para as Manutenções (MTTC)  
tempo para analisar o pedido de mudança, projetar a mudança, executar, testar e distribuir a nova versão.

✓ Integridade:

Capacidade de um software suportar ataques acidentais ou intencionais à sua integridade.

Medida indireta obtida através de análise das probabilidades de “ameaça” e “proteção”.

- Ameaça: Estimar a probabilidade de um determinado tipo de ataque (fraude, vírus, invasão etc.)
- Proteção: Estimar a probabilidade do ataque ser repellido ou não produzir impacto na operação.
- Integridade =

$$\sum \text{ameaças} - \sum \text{Pr oteção}$$

✓ Usabilidade:

Medida da facilidade de uso (user friendliness).

Pode ser medida a partir de quatro características:

- Habilidade intelectual exigida para se aprender a usar o sistema
- Tempo exigido para se tornar moderadamente eficiente em seu uso
- Aumento de produtividade no processo-alvo a partir do momento em que o usuário opera o sistema de maneira moderadamente eficiente
- Avaliação subjetiva obtida através de questionário sobre a percepção do usuário em relação a facilidade de uso do sistema.

- ✓ Como estamos indo ?
- ✓ Essa resposta pede uma estimativa inicial (para que um valor esperado seja calculado), ou pelo menos, a eleição de um indicador, que possa ser usado como meta a ser atingida.

Indicador é um parâmetro representativo do processo que desejamos quantificar, que servirá para compararmos a diferença entre a situação desejada e a alcançada (real), ou seja: ele indicará se temos problemas no processo.

- ✓ O Indicador deve ser quantitativo (numérico).
- ✓ Pode ser um parâmetro para medirmos o fator Qualidade ou Produtividade ou qualquer outro fator crítico para o processo.

Características de um bom Indicador:

1. Fácil de entender (não complexo)
2. Testado em campo (funciona ?)
3. Econômico (coleta automática)
4. Disponível a tempo (em que fase do processo ?)

Exemplo de indicadores de Qualidade

Número de defeitos

- ✓ Atendimento às especificações (Crosby)
- ✓ Atendimento à ISO 9000-3
- ✓ “Software de Qualidade é aquele que não apenas satisfaz as exigências, mas também é implementado a tempo e de acordo com o orçamento”. *Juran*

Qtde.não-conformidades / qtde. total

Exemplo de indicadores de Produtividade

1. LOC's/pm
2. FP/pm
3. Dias de atraso
4. ? ? ?

Produção total / Recursos consumidos

Como usar os Indicadores

1. Crie os indicadores (representativos, etc.)
2. Estabeleça o processo de coleta de dados para apurar quantitativamente o fator usado como indicador.
3. Estabeleça o período de coleta e de apuração e comparação de resultados.
4. Apure os resultados e compare o real com o desejado.
5. Em qualquer caso (acima ou abaixo do esperado), divulgue para a equipe

Como usar os Indicadores

6. Se a meta não foi atingida, faça reuniões com a equipe para descobrir as causas da diferença:
  - As causas de não se estar no estágio desejado
  - As causas de se estar no estágio atual
7. Identificar soluções para eliminar os problemas que causaram o não cumprimento das metas
8. Identificar quem irá aplicá-las, um cronograma de implantação e, se necessário, procedimentos para isso.

Como usar os Indicadores

9. Verificar se as alterações no processo pedem que se altere os indicadores eleitos, ou se é necessário eleger outros indicadores.
10. Voltar ao item "1" (Melhoria contínua).