
Pós-graduação em Ciência da Computação
Centro de Informática
Universidade Federal de Pernambuco

REQUISITOS DE DESEMPENHO EM SOFTWARE

Aluno: Paulo Jorge Lopes da Silva
Orientador: Ricardo Massa

Referências



- [1] Alebrahim, Azadeh. "Performance Analysis Patterns for Requirements Analysis." SOFSEM (Student Research Forum Papers/Posters). 2015.
- [2] Petriu, Dorin, and Murray Woodside. "Analysing software requirements specifications for performance." Proceedings of the 3rd international workshop on Software and performance. ACM, 2002.
- [3] Merugu, Gopichand, and Ananda Rao Akepogu. "Five Layered model for identification of software performance requirements." International Journal of Software Engineering & Applications 3.5 (2012): 47.

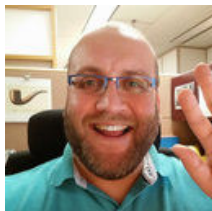


Referências



[University of Duisburg-Essen](#)

Ph.D., Computer Software Engineering
2009 – 2016



Carleton University · Department of Systems and Computer Engineering

Ph.D. M.Eng. (w/thesis), B.Eng. (high distinction)



C. Murray Woodside

Ph.D, (Cambridge)

Professor Emeritus and Distinguished Research Professor

Previously holder of the OCRI/NSERC Industrial Research Chair in Performance Engineering of Real-Time Software



Vardhaman College of Engineering

Information Technology

B.Tech, M.Tech, Ph.D



UNIVERSIDADE FEDERAL
DE PERNAMBUCO

cin.ufpe.br

Visão geral



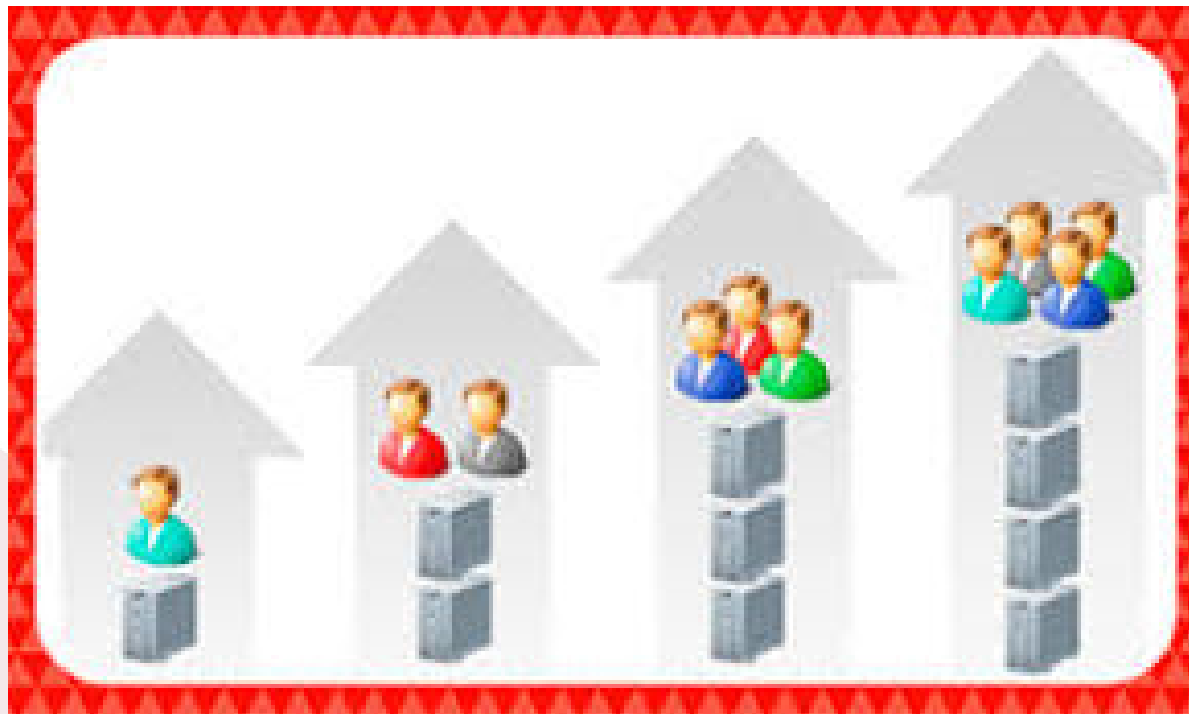
- O que é desempenho?
- Escalabilidade
- Por quê desempenho?
- Precisamos de Desempenho de Software?



O que é desempenho?

- Tempo de resposta?
- Taxa de transferência?
- Experiência do usuário ao utilizar o sistema?

Escalabilidade



Você pode ajustar o software mais tarde?

- O ajuste pode melhorar o desempenho, mas não tanto quanto o design
- Os problemas geralmente são causados por questões de arquitetura ou design, em vez de códigos ineficientes
- Muito caro (muitas vezes inviável) para alterar as escolhas fundamentais de design

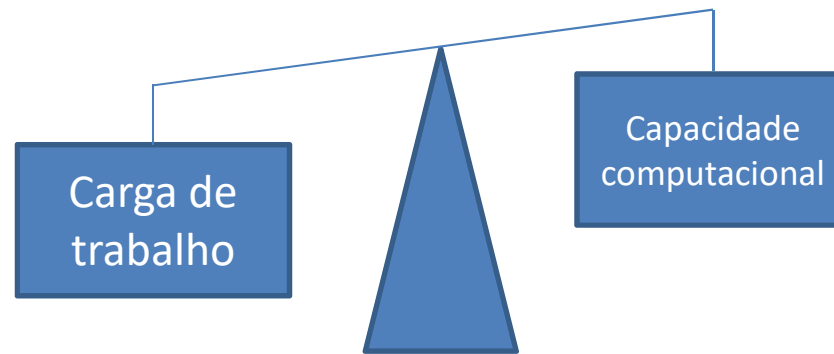
Eficiência pode implicar "código complicado"

- O ajuste pode introduzir um código complicado para resolver problemas que poderiam ter sido evitados
- Código complicado pode ser a única opção para atingir metas no final do ciclo de vida
- O desempenho aceitável é necessário e pode ser projetado em breve

Por quê desempenho?

- Melhora a experiência de uso;
- Pode diminuir custos;
- Facilita manutenção do sistema;

Especificação de Desempenho de Software é necessário?



- As Especificações de Desempenho de Software podem detectar problemas no início do desenvolvimento;

Questões de desempenho iniciais

- Seus usuários poderão concluir tarefas no tempo previsto?
- Seu hardware e rede são capazes de suportar a carga?
- Qual o tempo de resposta esperado para as tarefas principais?
- O sistema aumentará para atender às suas necessidades futuras?

Exemplos de requisitos de desempenho

- Requisitos de velocidade e latência
 - Determina a quantidade de tempo disponível para completar tarefas específicas
 - Ex: O produto deve verificar os sensores a cada 10s;
- Requisitos de precisão ou exatidão
 - Quantificação da exatidão desejada dos resultados atingidos pelo produto;
 - Ex: Precisão das leituras de temperaturas;

Exemplos de requisitos de desempenho

- Requisitos de confiabilidade e disponibilidade
 - Quantifica a confiabilidade e disponibilidade necessária do produto.
 - Ex: O produto deve estar disponível 24 horas por dia.

A Engenharia de Desempenho de Software começa na Análise de Requisitos

- Benefícios da EDS na fase inicial:
 - Produtividade aumentada - não é preciso lançar projetos ruins
 - Melhor qualidade e utilidade do produto de software resultante - seleção de opções de design adequadas
 - Custos controlados do hardware e software de suporte - identificação do equipamento necessário
 - Aumento da produtividade durante a implementação, testes e fases iniciais de operação - garantindo que a capacidade de computação esteja disponível

Obrigado!!!