

Universidade de São Paulo

Instituto de Ciências Matemáticas e de Computação

Departamento de Sistemas de Computação

SSC643 – Avaliação de

Desempenho de Sistemas

Computacionais

Aula 4

Sarita Mazzini Bruschi

Material baseado nos slides de:

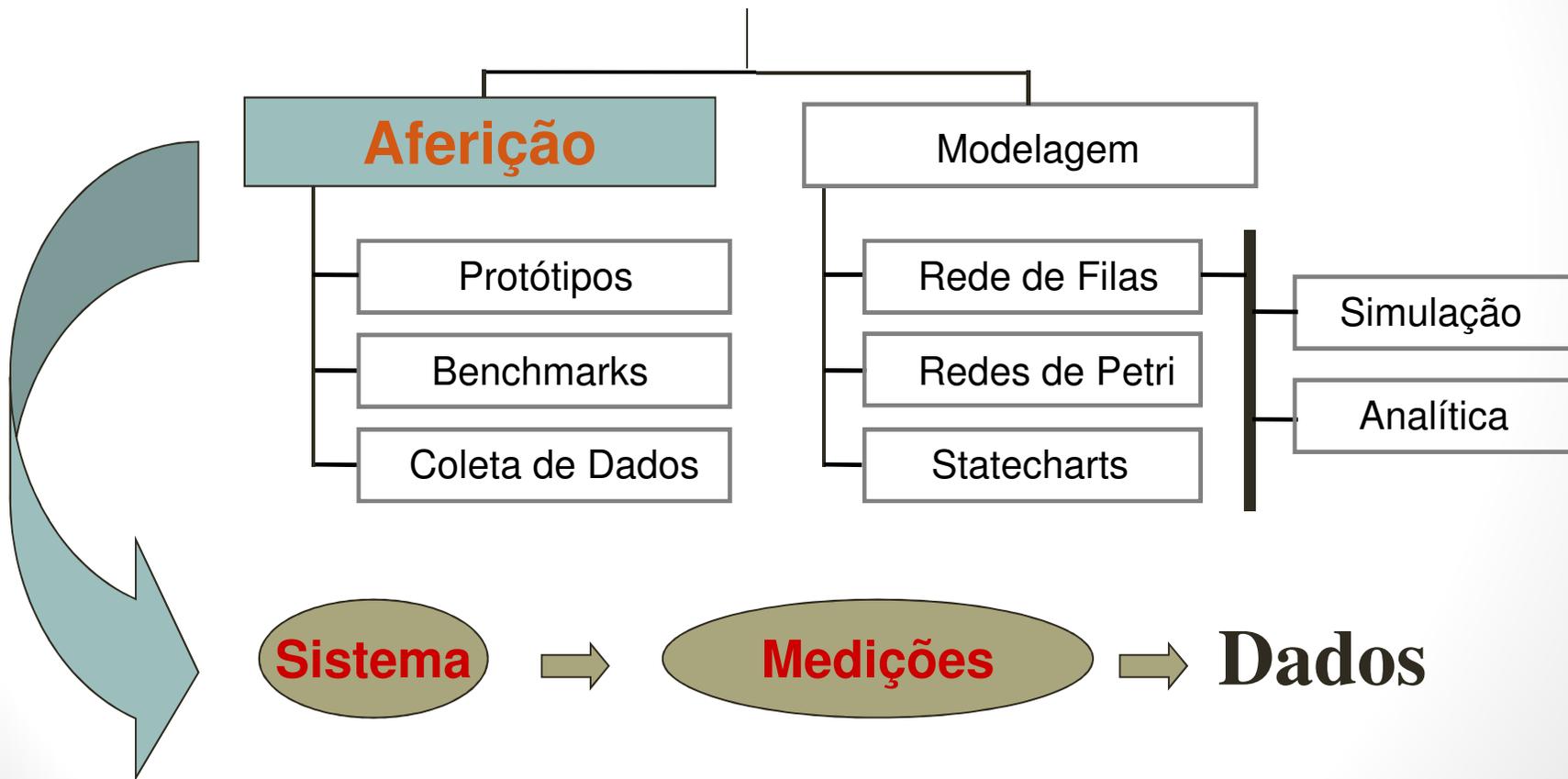
Marcos José Santana

Regina Helena Carlucci Santana

Conteúdo

1. Planejamento de Experimentos
2. **Técnicas para Avaliação de Desempenho**
 - Apresentação das técnicas
 - Técnicas de Aferição:
 - Protótipos, Benchmarks e Monitores
 - Técnicas de Modelagem:
 - Solução Analítica e por Simulação
 - Exemplos
3. Análise de resultado

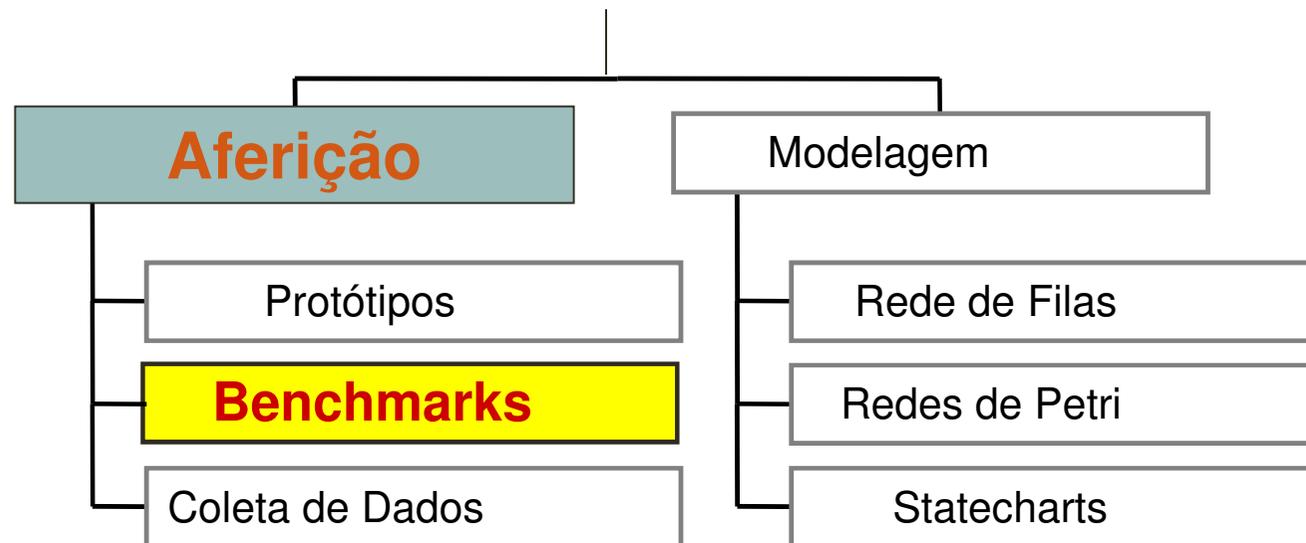
Técnicas de Avaliação de Desempenho



Técnicas de Avaliação de Desempenho

- Aferição
 - Medidas no próprio sistema
 - Sistema deve existir e estar disponível
 - Experimentação restrita
 - Muito cuidado com aquisição dos dados

Técnicas de Avaliação de Desempenho



Benchmarks

- Instrumento fixo, que permite comparar uma medida (mark - marca) a um padrão preestabelecido
- Deve-se ter um ponto de observação (bench - banco)
- Ponto fixo ou referência para comparações

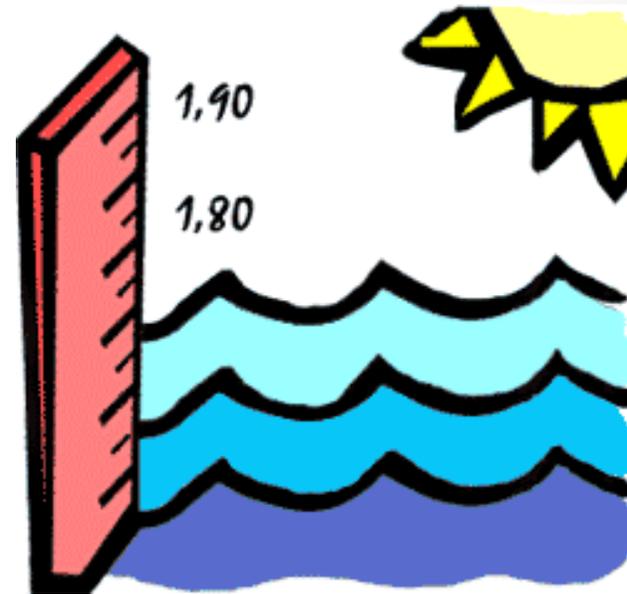


Benchmarks

- Empresas
 - Utilizam como modelo
 - Onde elas pretendem chegar
- Ponto fixo ou referência para comparações
- Definir um benchmark para a vida....
- Exemplos:
 - Tábua das mares
 - Termômetro

Benchmarks

- Tábua das mares



ALTURA	NÍVEL DA MARÉ
1,00m - 1,50m	Maré alta
1,60m - 1,70m	Observação
1,80m	Atenção

Benchmarks

- Termômetro



Vinho



Técnicas de Aferição

- Benchmarks
 - Programa escrito em linguagem de alto nível, representativo de uma classe de aplicações, utilizado para medir o desempenho de um dado sistema ou para comparar diferentes sistemas

Benchmarks

- Abordagem muito utilizada para a avaliação de desempenho por aferição

- Exemplo

- Qual a diferença entre um i5 e um i7?
- Qual a influência no desempenho?

i5	i7
2 ou 4 núcleos	4 ou 6 núcleos
Não possui Hyper-threading	Possui Hyper-threading (2 núcleos lógicos para cada físico)
DMI – Direct Media Interface (taxa de transferência ~2Gb/s)	QPI – Quick Path Interconnect (taxa de transferência > 4,8 Gb/s)
Quantidade de canais para acesso a memória: 2 (acessa 2 pentes ao mesmo tempo)	Quantidade de canais para acesso a memória: 3 (acessa 3 pentes ao mesmo tempo)

Benchmarks

- <http://www.cpubenchmark.net/>
- PassMark Performance Test

Processador	Benchmark	Preço (\$)
<u>Intel Core i7 980X @ 3.33GHz</u>	10225	1189,00
<u>Intel Core i7 975 @ 3.33GHz</u>	6822	1019,47
<u>Intel Core i5 760 @ 2.80GHz</u>	4585	232,44
<u>Intel Core i5 680 @ 3.60GHz</u>	3508	326,35
<u>Intel Core i7 740QM @ 1.73GHz</u>	3627	407,53

Benchmarks

- Uso:
 - Comparar desempenho de máquinas diferentes
 - Re projetar hardware e software
 - Decidir sobre aquisição de sistemas
 - Ajudar na otimização de programas
 - Previsão de desempenho de aplicações em computadores específicos

Benchmarks

- Permitem avaliar o sistema como um todo, incluindo aspectos relacionados com:
 - Arquitetura do sistema
 - Eficiência do compilador
 - Sobrecarga do sistema operacional
 - Sobrecarga de comunicação
 - Identificam pontos fortes e fracos de um sistema em particular ou de aplicações

Benchmarks

- Como escolher um benchmark?
 - Ideal: aplicação do usuário
 - O ideal pode ser inviável quando os sistemas são de propósito geral
 - Necessita-se de algo mais amplo e representativo

Benchmarks

- Programa escrito em linguagem de alto nível
- Representativo de alguma categoria de programação
- Que possa ser avaliado facilmente
- Que possua larga distribuição

Benchmarks

- Divididos em:
 - Benchmark de componente;
 - Benchmark de sistema;
 - Benchmark de aplicação – utiliza programas representativos
 - Benchmark sintético - imitam ambientes de execução de programas reais. Geram dados estatísticos

Benchmarks

- Classificação quanto a arquitetura:
 - seqüenciais;
 - paralelas;
 - Comitê Parkbenck (Parallel Kernels and Benchmarks)

Benchmarks

- Onde usar benchmark?
 - avaliar sistemas computacionais diferentes;
 - avaliar o desempenho mínimo;
 - tarefas genéricas ou específicas.

Áreas de Aplicação e Objetivos

- Três grandes áreas:
 - projeto
 - não aplicável
 - avaliação
 - aplicável, porém insuficiente
 - seleção
 - técnica ideal

Medidas de Desempenho Frequentemente Utilizadas

- Comum aos outros casos:
 - Tempo de resposta,
 - Utilização,
 - Throughput,
 - Tempo/Tamanho de filas.
- Freqüência de clock - MHZ
 - Pode ser UMA medida
 - Problemas: é necessário considerar:
 - Arquitetura do processador
 - Velocidade e quantidade de memória
 - Disco

Medidas de Desempenho Frequentemente Utilizadas

- MIPS
 - Milhões de Instruções por segundo
- MFLOPS
 - Milhões de Instruções de Ponto Flutuante por Segundo
- Fabricantes apresentam números de pico de MIPS e MFLOPS superiores aos reais, valores obtidos em situações ideais
- Comparação entre máquinas com conjunto de instruções diferentes – CISC X RISC

Medidas de Desempenho Frequentemente Utilizadas

- MIPS e MFLOPS são relevantes quando aplicados em computadores de mesma arquitetura para análise comparativa
 - Apresentam características similares de hardware e software
 - Números de MIPS e MFLOPS mais coerentes

Sistemas Sequenciais e Paralelos

- Sequencial
 - todas as considerações dos slides anteriores
- Paralelo
 - todas as considerações dos slides anteriores
 - Porém, o poder de processamento não é o único fator determinante da eficiência. Muito depende do quão inato é o paralelismo da aplicação, e qual a magnitude deste paralelismo que pode ser explorado pela máquina

Aspectos Relacionados aos Benchmarks

- Quando os resultados dos Benchmarks forem interpretados, deve-se conhecer:
 - arquitetura do processador;
 - processadores;
 - clock;
 - barramento;
 - memória e caches;
 - disco rígido;
 - sistema operacional;
 - Compilador
 - etc.

Aspectos Relacionados aos Benchmarks

- Problemas...
 - Sistemas com configurações diferentes geram medidas de desempenho diferentes
 - Otimização do compilador: influencia diretamente no desempenho medido

Tipos de Benchmarks

- Benchmarks mais comuns
 - Whetstone, Linpack, Dhrystone
- Outros programas de Benchmarks
 - Stanford Small Programs Benchmark Set
 - EDN Benchmarks
 - Sieve of Eratosthenes
 - Livermore Fortran Kernels
 - Perfect Club Benchmarks
 - SPEC Benchmarks
 - EuroBen Benchmarks

Tipos de Benchmarks

- Whetstone
 - Primeiro grande programa da literatura escrito para Benchmarking
 - Elaborado para análise de programação numérica de ponto flutuante intensivo
 - Apenas a versão Pascal é oficialmente controlada
 - Resultado: número de loops por segundo

Tipos de Benchmarks

- Características do Whetstone
 - Possui alto percentual de dados e operações de ponto flutuante
 - Alto percentual de tempo de execução é gasto em funções matemáticas
 - Ao invés de variáveis locais, Whetstone utiliza muitos dados globais

Whetstone

- Ranking das melhores máquinas
- Mwips: Million Whetstones Instructions Per Second
- Whetstone Benchmark Detailed Results On PCs
 - <http://www.roylongbottom.org.uk/whetstone.htm>

Rank	Machine	Mflop ratings (VI=1024)			Total CPU (seconds)	MWIPS
		N2	N3	N8		
1	Intel Woodcrest 3.0GHz 4MBL2 DC	1966	4588	2907	3.3	10560
2	Intel Woodcrest 3.0GHz-533 4MBL2 DC	1966	4588	3069	3.3	10451
3	IBM eServer p5 570/1.9	1966	1966	1625	6.2	6219
4	SunFire V20 2.2GHz (EKO)	1311	1298	1481	7.7	4496
5	IBM eServer p5 575/1.5	1966	1529	1315	7.8	4874
6	AMD Opteron852/2600 (EKO 2.2)	1513	1547	1771	8.1	4488
7	HP DL380 Pentium4/3600 (EM64T)	1966	1720	607	8.4	4408
8	Dell PowerEdge 1850/3600 1MBL2	1966	1720	607	8.4	4351
9	Dell PowerEdge 1850/3600 2MBL2	1966	1720	607	8.5	4370
10	AMD Opteron875/2200 DC (EKO 2.0)	1311	1251	1497	8.6	4543

VL = Vector loops

MWIPS = million whetstones instructions per second

N2,N3 e N8 – diferentes instruções de ponto flutuante no loop

Tipos de Benchmarks

- Livermore Fortran Kernels
 - Este benchmark foi criado em 1970 por F.H. McMahon
 - Consiste de 24 “kernels” ou loops de processamento numérico
 - Analisa desempenho de computadores e compiladores em operações com ponto flutuante

Tipos de Benchmarks

- SPEC Benchmarks
 - SPEC (System Performance Evaluation Cooperative ou Standard Performance Evaluation Corporation) fundada em Outubro de 1988 por Apollo, Hewlett-Packard, MIPS e Sun Microsystems.
 - Possui como meta, realismo de medidas de sistemas computacionais avançados para a indústria e educação dos consumidores quanto aos aspectos de desempenho.

Tipos de Benchmarks

- SPEC Benchmarks
 - CPU
 - Aplicações Gráficas
 - Aplicações de Alto Desempenho
 - MPI e OpenMP
 - Cliente/Servidor Java
 - Projetado para medir o desempenho de servidores rodando aplicações Java
 - Mail Servers
 - Mede desempenho de servidores de e-mail
 - Network File Systems
 - Web Servers
 - Simula usuários enviando requisições

Benchmarks

- Benchmarks para Arquiteturas Paralelas e Sistemas Distribuídos:
 - problemas com MIPS e MFLOPs: não basta somá-los:
 - comunicação;
 - balanceamento;
 - etc.
 - cada tipo de arquitetura: benchmark específico;
 - Exemplos:
 - PING e PING-PONG:
 - utilizados na avaliação de comunicação;
 - TTCP:
 - avaliação de comunicação usando TCP ou UDP.

Benchmarks

- Para utilização específica:
 - Servidores Web
 - Redes
 - HD
 - Servidores de e-mail
 - Virtualização
 - SOA
 - Servidores de arquivos
 - Etc

WebStone

- Desenvolvido pela Silicon Graphics
- Objetivo: Medir desempenho de software e hardware de servidores Web
- Versão 2.0.1
- Testa o servidor Web, incluindo sistema operacional, CPU e velocidade de acesso a rede.
- Não testa influência da velocidade de acesso ao disco
- Principais medidas
 - Throughput máximo
 - Tempo médio de resposta

Surge

- Simula uma coleção de usuários acessando um servidorWeb
- Arquitetura
 - Conjunto de clientes conectados a um servidor Web
 - Cliente
 - Conjunto de threads, especificado pelo usuário
 - Requisições a documentos no servidor
 - Estados ON e OFF
 - ON -> transferência de arquivos
 - OFF -> Ativo e Inativo

httperf

- Ferramenta para medir o desempenho de servidores Web
- Gera várias cargas HTTP
- Arquitetura
 - Maior independência do SO
 - Única thread
 - Mecanismo próprio de Timeout
 - Três módulos
 - HTTP
 - Gerenciamento de conexões
 - Geração de requisições http
 - Geração de carga
 - Regras para a geração de requisições
 - Coleta de estatísticas

httperf

- Carga
 - Conexões persistentes e requisições em cascata
 - Geradores de carga
 - Geração de requisições
 - Requisições a uma taxa fixa. Padrão: 1 req/con
 - Sessões a uma taxa fixa. 1 sessão: várias rajadas de requisições. Think time: 0.
 - Geração de URLs
 - Sequência de URLs que devem ser acessadas no servidor.

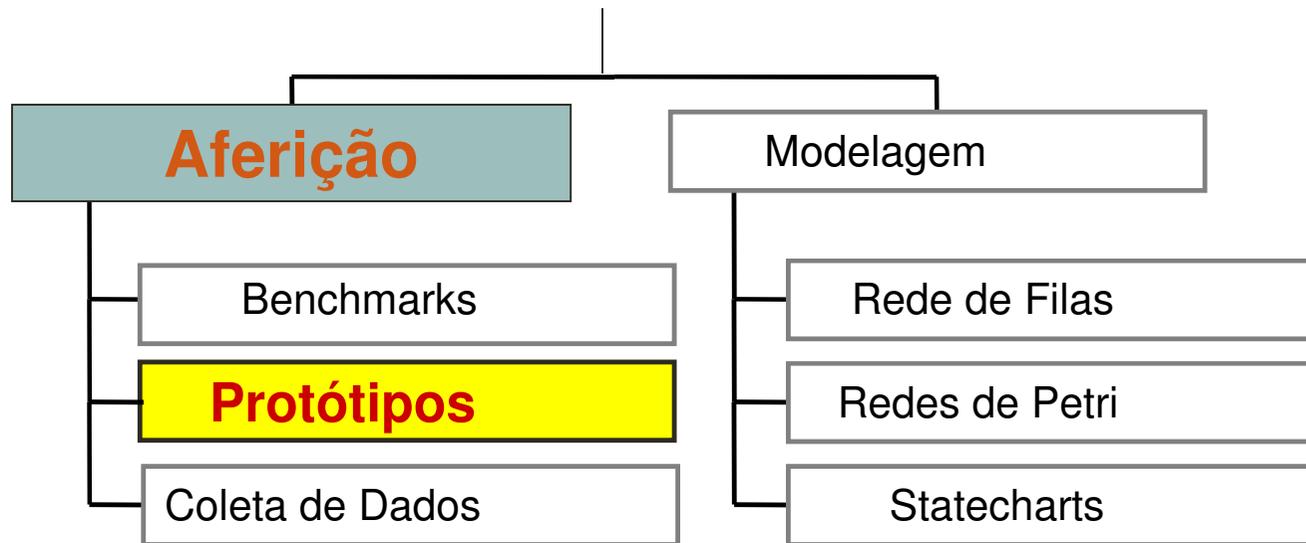
Concluindo....

- Benchmarks podem ser utilizados para verificar diversos tipos de sistemas ...
 - Servidores Web,
 - Banco de dados,
 - Processadores,
 - Redes de comunicação
- ... sendo utilizados com diferentes objetivos...
 - Codificação de vídeo e edição de imagens,
 - Jogos,
 - Processamento intensivo,
 - Processamento de textos, etc.
- ... querendo avaliar diferentes características...
 - Produtividade
 - Desempenho
 - Confiabilidade, etc.

O importante é...

- Escolher o Benchmark adequado
- Aplicar o Benchmark de forma adequada
- Analisar os resultados obtidos com critério

Técnicas de Avaliação de Desempenho



Técnicas de Aferição

- Construção de Protótipos
 - Versão simplificada de um sistema computacional que contém apenas características relevantes para a análise do sistema

Técnicas de Aferição

- Construção de Protótipos
 - uma implementação simplificada do sistema real;
 - abstração das características essenciais;
 - Sistemas em fase de projeto;
 - produz resultados com boa precisão;
 - recomendado para verificação do projeto final;
 - problema: custo e alterações.

Construção de Protótipos

- Considerações envolvidas:
 - identificar os objetivos do projeto;
 - abstrair as características essenciais;
 - definir a estratégia de coleta de dados no protótipo;
 - desenvolver o protótipo;
 - avaliar o seu desempenho;
- Além disso, devem ser considerados:
 - viabilidade da prototipação do sistema;
 - melhorias no protótipo, em função da avaliação e análise.

Construção de Protótipos

1. Analisar se o sistema é um bom candidato a prototipação
2. Delimitar e conhecer perfeitamente os domínios funcionais e comportamentais do sistema
3. Desenvolver o protótipo
4. Testar o protótipo
5. Coletar dados do protótipo

Após o passo 4, o protótipo já provê uma ideia do sistema real.

- Os passos 3, 4 e 5 são repetidos iterativamente até que se tenha toda a funcionalidade do sistema representada no protótipo

Construção de Protótipos

- Concluindo.....
 - Ótima opção para verificação de projetos
 - Bom para alguns tipos de sistemas
 - Custo pode ser um problema
 - Flexibilidade não é ponto forte!