UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO - BACHARELADO

SISTEMA ESCALONADOR DE REQUISIÇÕES PARA SERVIDORES DE APLICAÇÕES DISTRIBUÍDAS

RODRIGO SIEWERDT

Prof. Antônio Carlos Tavares, Especialista - Orientador

Roteiro da Apresentação

- Introdução
  - Objetivos do trabalho
- Fundamentação teórica
  - Sistemas distribuídos
  - Técnicas de otimização de desempenho
  - Tecnologia COM/DCOM
- Desenvolvimento do sistema
  - Fluxo das informações no sistema
  - Cálculo do número de processos
- Resultados
- Conclusões
  - Extensões e sugestões de trabalhos futuros

Introdução

- Incentivos para a utilização de sistemas distribuídos:
  - Crescimento do setor tecnológico
  - Tecnologia de circuitos integrados
  - Aplicações cada vez mais complexas e com tempo de resposta reduzido.
- Otimização de desempenho, onde se encontra o escalonamento de processos.

Objetivos

Desenvolver um escalonador de processos que seja capaz de dividir as tarefas e promover o balanceamento das cargas de processamento entre os servidores de um grupo que atende as requisições de determinada aplicação.

Sistemas distribuídos

- Execução em várias máquinas diferentes (servidores)
- Arquitetura cliente/servidor
- Aplicação básica (trivial) da tecnologia são os sistemas de ERP
- Módulos executando em diferentes servidores
- Distribuição depende das configurações do próprio sistema.

Técnicas de otimização de desempenho

- Recursos poderosos que visam aumentar ainda mais o desempenho
- Não é comum encontrar sistemas utilizando estas técnicas
- Objetivos principais:
  - Conhecer o grau de ociosidade dos membros de um grupo de máquinas
  - Evitar o gargalo do sistema pela exploração demasiada de apenas um membro do grupo

Escalonamento de processos

- Evitar desperdício de recursos devido a ociosidade de alguns membros e a sobrecarga de outros
- Metodologias de desenvolvimento:
  - Algoritmos estáticos (determinísticos)
  - Informações centralizadas
  - Políticas de escalonamento
    - Política de localização
    - Política de informação

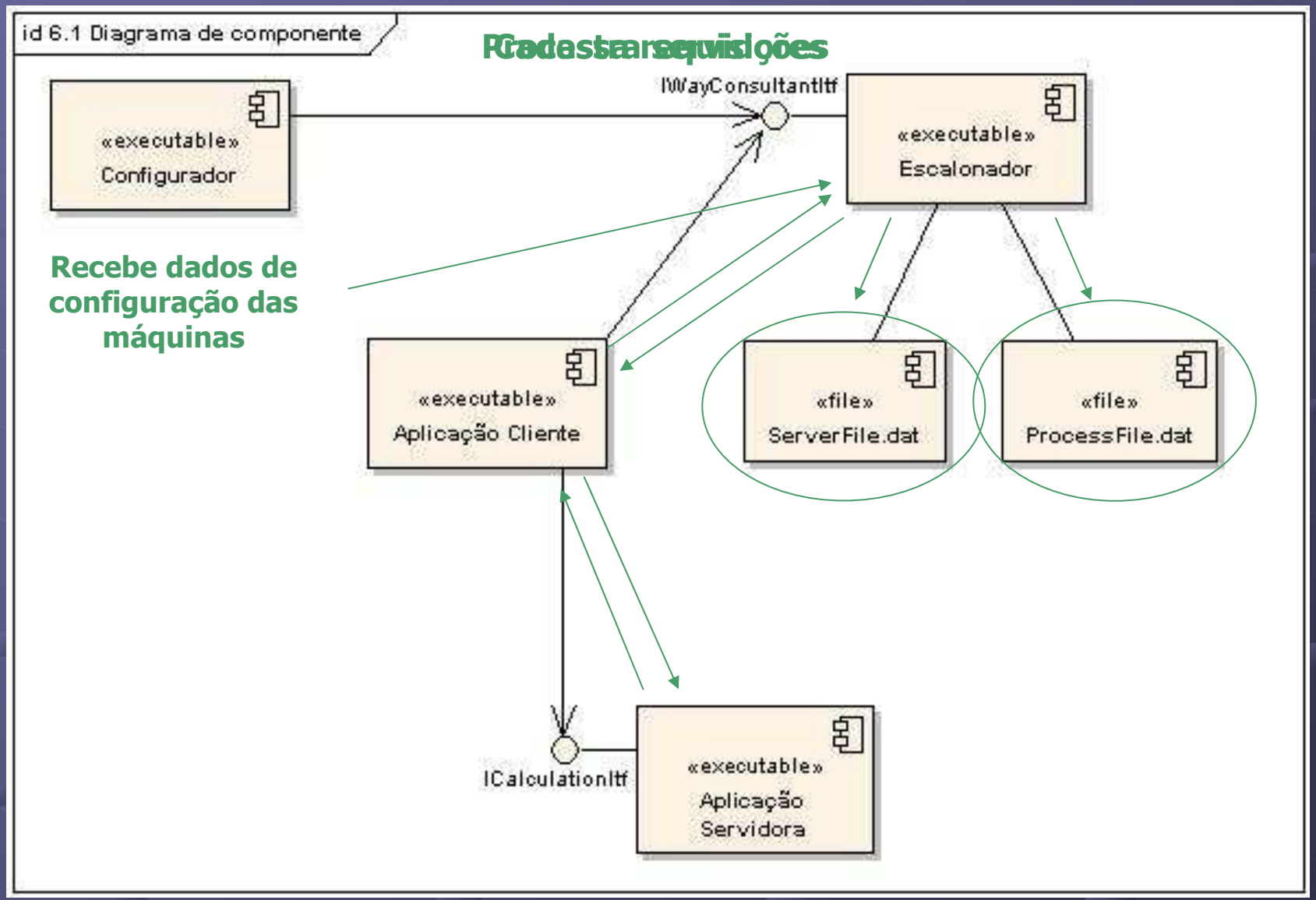
Tecnologia COM/DCOM

- Baseado em Remote Procedure Calls (RPC) criado pela Open Software Foundation (OSF) em 1992
- Proposta primária de interligar diversos computadores pessoais e criar um padrão de comunicação.
  - Filosofia de desenvolvimento baseada em componentes acessados por interfaces
  - Padrão de codificação e orientação à objetos
  - Conjunto de ferramentas para desenvolvimento de sistemas distribuídos

Desenvolvimento do sistema

- Componentes individuais para cada tarefa
- Algoritmo de escalonamento baseado nas políticas de localização e informação
- Informações de cadastro e nível de carga dos membros centralizadas
  - Todas as requisições e avisos de término de execução são recebidas pelo escalonador
- Componentes desenvolvidos utilizando Visual C++ e DCOM
- Perfeita integração entre ferramenta e tecnologia de comunicação entre objetos.

Fluxo das informações no sistema



Cálculo do número de processos

- Variáveis determinantes: CPU e memória RAM
- Fator de ajuste individual por máquina, controlado pelo usuário para melhor dimensionamento da carga de processos

- *iMaxLoad* = fator informado + (((clock da cpu/1000) +1) * ((qtde memória RAM/1024) +1))

- *Ex.: CPU = 3GHz* RAM = 1Gb Fator = 20

  iMaxLoad = 20 + (((3000/1000)+1 * ((1024/1024)+1)) = 20 + (4*2) = 28

Resultados e discussão

- *Equipamentos utilizados nos testes em laboratório:*

Endereço IP	***CPU***	***RAM***
172.16.1.189	450	196
172.16.1.202	450	256
172.16.1.208	450	196
172.16.1.209	475	196
172.16.1.216	1466	512
172.16.1.222	475	196

- *Configurações bem distintas, com uma máquina bastante superior em relação às outras que tem diferenças pequenas na velocidade da CPU e quantidade de memória RAM.*

Resultados e discussão

E	Fator	Servidor	Cliente	Escalonador
1	50	Todas	172.16.1.202	172.16.1.202
2	50	172.16.1.202	172.16.1.202	172.16.1.222
		172.16.1.208	172.16.1.222	
		172.16.1.216		
3	20	172.16.1.202	172.16.1.202	172.16.1.222
		172.16.1.208	172.16.1.222	
		172.16.1.216		
4	20	172.16.1.216	172.16.1.202	172.16.1.222
			172.16.1.222	

Resultados e discussão

- As máquinas que executaram as várias instâncias da aplicação cliente tiveram seu desempenho bastante comprometido
- O serviço de escalonamento de requisições também utilizou recursos de processamento da máquina, porém mais amenos que nas aplicações cliente
- O fator utilizado para cálculo da quantidade de processos a executar em cada máquina influencia muito no resultado final e por isso necessita de muita atenção
- Apesar da diferença de configuração do hardware, o escalonamento de requisições contribui para a melhoria do desempenho global do sistema

Conclusão

- A tarefa de melhorar o desempenho global do sistema obteve um resultado positivo
- O ajuste fino para cálculo do número de processos de cada servidor não alcançou um resultado ótimo, porém satisfatório
- O escalonamento de requisições entre diversos membros foi efetuada com sucesso
- A sobrecarga por excesso de comunicação foi mínima, graças a utilização de um membro específico apenas para tarefas de escalonamento

Extensões e sugestões de trabalhos futuros

- Implementar um algoritmo dinâmico, utilizando as informações em tempo de execução, já mantidas pelo sistema, para escolha do servidor ideal

- Construir um ambiente de escalonamento que encaminhe diretamente as requisições à aplicação servidora e retorne apenas o resultado para o cliente