



MAC5780 - Introdução ao
Escalonamento e Aplicações

Escalonamento em Grid Computing

Charles Boulhosa Rodamilans

Agenda

- Objetivo
- Grid Computing
- Escalonamento em Grid Computing
- Simulação
- Heurísticas
 - MaxMin
 - XSufferage
- Workload e Ambiente
- Trabalho Realizado e Continuação

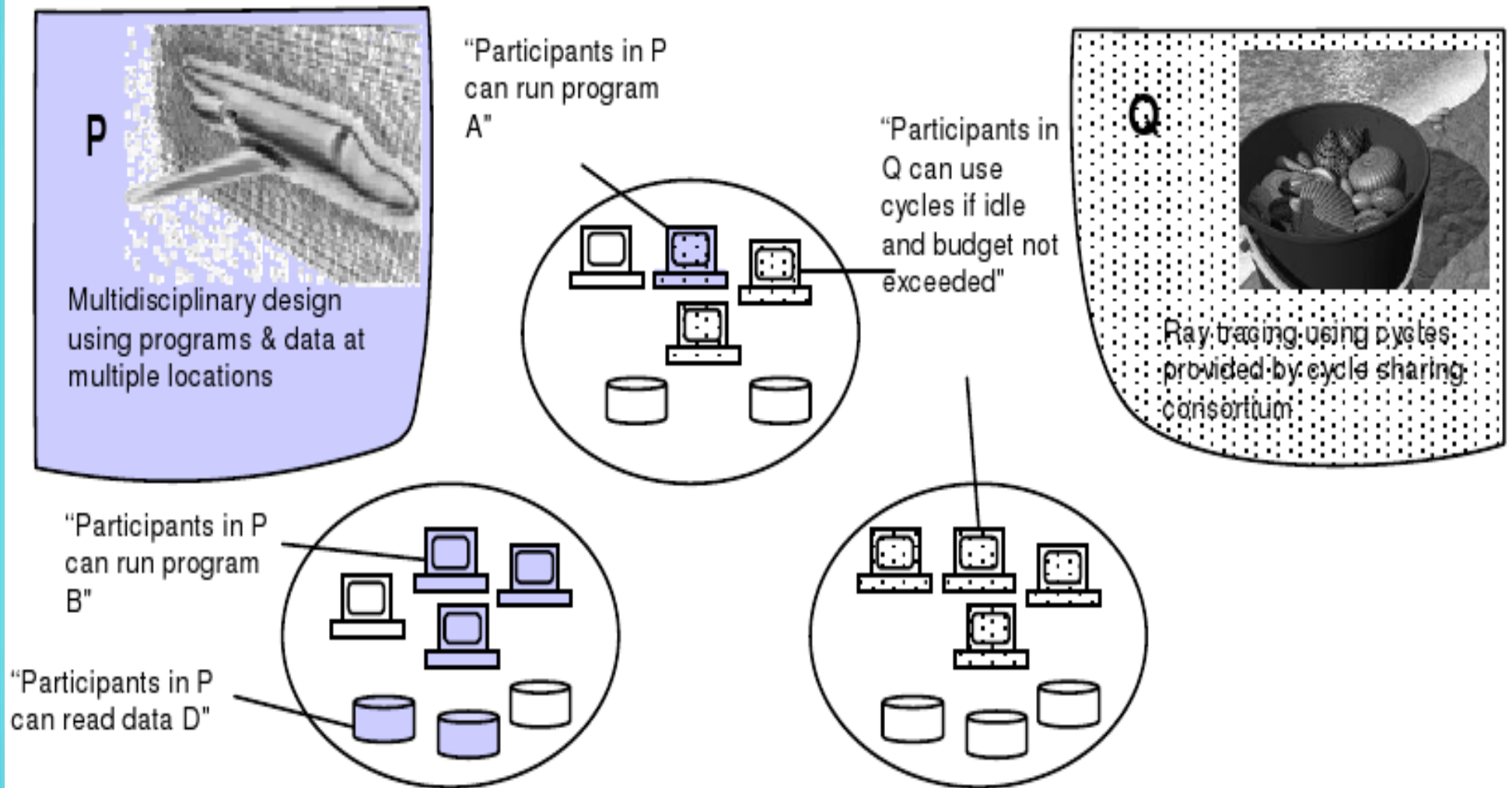
Objetivo

O objetivo deste trabalho é implementar e comparar os algoritmos MaxMin e XSufferage para o escalonamento em ambiente de Grid Computing utilizando o simulador GridSim, em máquinas Q_m , com tarefas independentes de tamanhos diferentes, utilizando o makespan como critério de otimização.

Grid Computing

- Coordenação de recursos sem controle centralizado
- Utilização de padrões abertos para protocolos e interface
- Coordenação de recursos sem controle centralizado
- Qualidade de Serviços (QoS)
- Heterogeneidade
- Escalabilidade
- Dinamicidade ou adaptabilidade

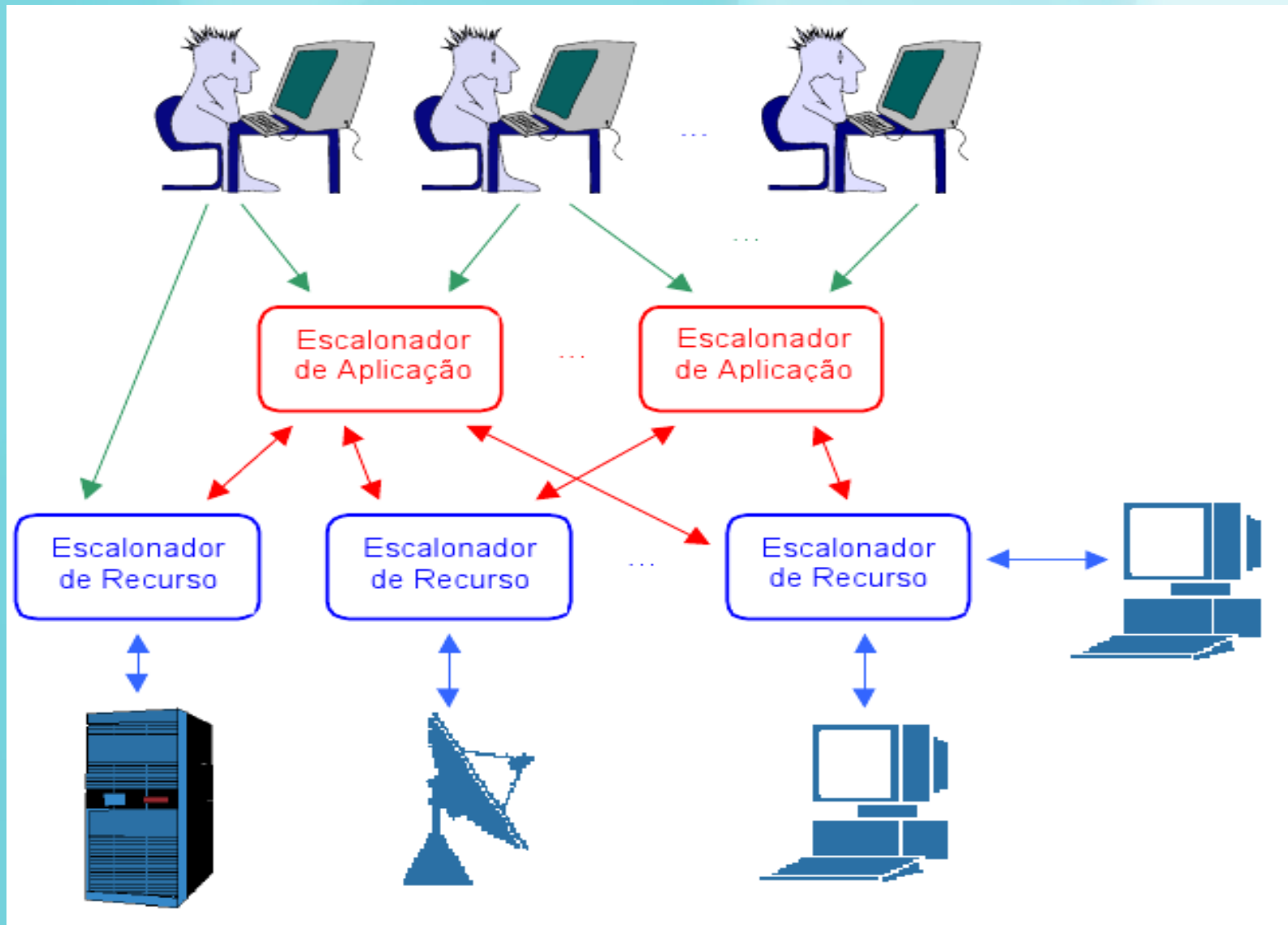
Grid Computing



Escalonamento em Grid Computing

- Compartilhamento de recursos em diversos domínios administrativos
- Escalonador de Aplicação (ou Tarefas)
 - Não controla os recursos que usa
- Escalonador de Recurso
 - Gerenciador do Recurso Local
 - Política Administrativa Local

Escalonamento em Grid Computing



Cirne, 2002

Simulação

- Por que usar Simulação [Murshed,2002]:
 - Menor tempo para criação da Infraestrutura de testes
 - Menor tempo de execução das tarefas
 - Análise dos algoritmos em diferentes ambientes (recursos, usuários)
 - Oferece controle do ambiente para repetição do experimento com diferentes estratégias de escalonamento
- Simulador de Grid: GridSim

Heurísticas

- MCT (Minimum Completion Time)
- MaxMin
 - Mapeia a maior tarefa para o melhor recurso
- XSufferage
 - Baseado no Sufferage
 - Determinar o quanto a tarefa seria prejudicada caso não fosse escalonada para o recurso mais eficiente
 - Utiliza o conceito de domínio de Grid
- Critério de Otimização: Makespan

Workload

- Workload
 - LPC EGEE 2004
 - 244821 Tarefas
 - Tarefas Independentes
 - Tamanho: varia entre 900 e 259200

Ambiente

- 5 Grids
- Cada Grid com 30 máquinas
- Grid 1: Cada máquina com 1 processador de 1000 MIPS
- Grid 2: Cada máquina com 1 processador de 1500 MIPS
- ...
- Grid 1: Cada máquina com 1 processador de 3000 MIPS

Trabalho Realizado e Continuação

- Trabalho Realizado
 - Implementação do ambiente no GridSim
 - Implementação do MaxMin
 - Terminando a implementação do Xsufferage
- Continuação
 - Execução dos Testes
 - Comparação

Dúvidas?

Referências

[CIRNE, 2002] CIRNE, W. Grids Computacionais: Arquiteturas, Tecnologias e Aplicações. Anais do Terceiro Workshop em Sistemas Computacionais de Alto Desempenho, 2002.

[Foster, 2001] FOSTER, I.; KESSELMAN, C.; TUECKE, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal Supercomputer of Applications*, v. 15, n. 3, 2001.

[Murshed, 2002] MURSHED, M. BUYYA, R. Using the GridSim Toolkit for Enabling Grid Computing Education. In: *International Conference on Communication Networks and Distributed Systems Modeling and Simulation*, SCS, 2002. p. 18–24.

MaxMin

```
Enquanto houver Tarefa
  Para cada tarefa(i)
     $MCT(i,j)$  = menor CT da tarefa(i) na máquina(j)
  Fim.Para
  s = Tarefa(i) com maior  $MCT(i,j)$ 
  Marque tarefa(s) para máquina(j)
  //Marca para o Grid que contém a máquina
  Retire a tarefa(s) da lista de Tarefas
Fim.Enquanto
```

XSufferage

Enquanto houver tarefa

Para cada tarefa(i)

Para cada Grid(k)

Calcule o CT da tarefa(i) em na máquina(j) do Grid(k)

Fim.Para

$MCT1(i)$ = menor CT para a tarefa(i) no Grid(k)

$MCT2(i)$ = segundo menor CT para a tarefa(i) no Grid(k)

$Suffer(i) = MCT2 - MCT1$

Fim.Para

s = Tarefa(i) com maior Suffer(i)

Marque a tarefa(s) para o Grid

Retire a tarefa(s) da lista de Tarefas

Fim.Enquanto