Estudo comparativo de escalonadores de tarefas para grades computacionais

Candidato
Alvaro Henry Mamani Aliaga*

Orientador
Alfredo Goldman

Instituto de Matemática e Estatística
Departamento de Ciência da Computação
Universidade São Paulo

alvaroma@ime.usp.br

13 de Dezembro de 2010

*O aluno recebe apoio financeiro do CNPq, processo 133147/2009-6

Roteiro

- Introdução
  - Computação em grade
  - Escalonamento
  - Motivação e objetivos
- Escalonadores
  - OAR
  - OurGrid
  - Condor
  - PBS
  - Maui
- Algoritmos de Escalonamento
  - Tarefas independentes
  - Tarefas dependentes
- Simulador
  - SimGrid
- Resultados Iniciais
- Plano de Trabalho e Cronograma
- Conclusões

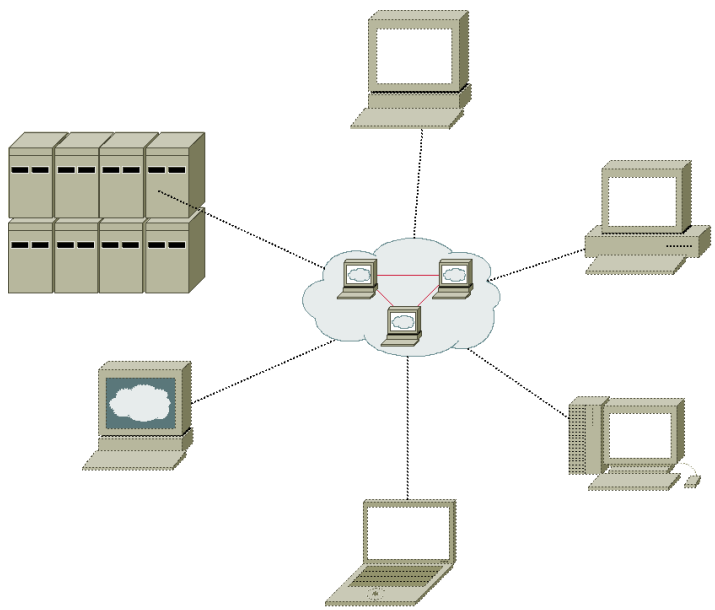
Introdução

- Necessidade de poder computacional: mineração de dados, previsão do tempo, processamento de imagens médicas, ...
- Aumento na disponibilidade de computadores poderosos e na interligação de redes de alta velocidade
- Computação em grade

  Uma alternativa para obter grande capacidade processamento
- Características da computação em grade:
  - Heterogeneidade
  - Dinamicidade

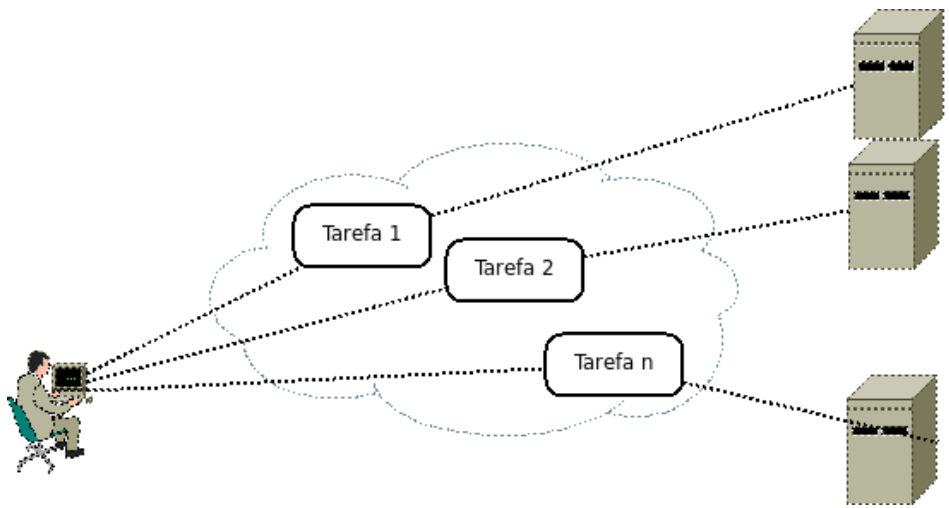
Computação em grade

- Compartilhamento coordenado e dinâmico de recursos por diversas instituições
- *Middlewares*: *Globus*, *Legion*, *InteGrade* e *OurGrid*
- Estes já permitem que coleções heterogêneas distribuídas em aglomerados interconectados através da Internet trabalhem em conjunto

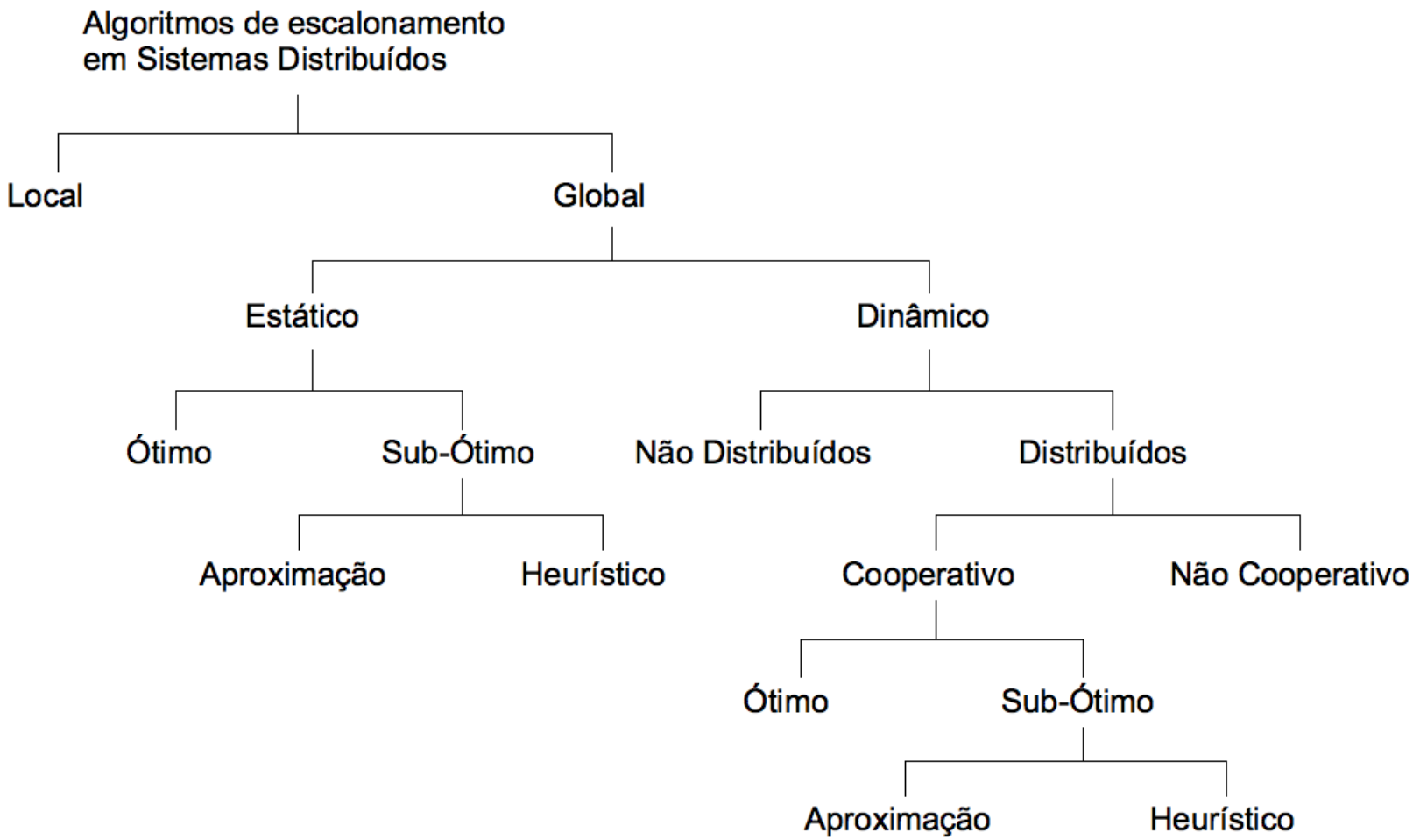


Escalonamento

- O problema de escalonamento: NP-Completo
- Atribuição de tarefas no tempo aos recursos
- Os principais objetivos
  - Maximizar a utilização dos recursos computacionais disponíveis
  - Minimizar os custos relativos à comunicação

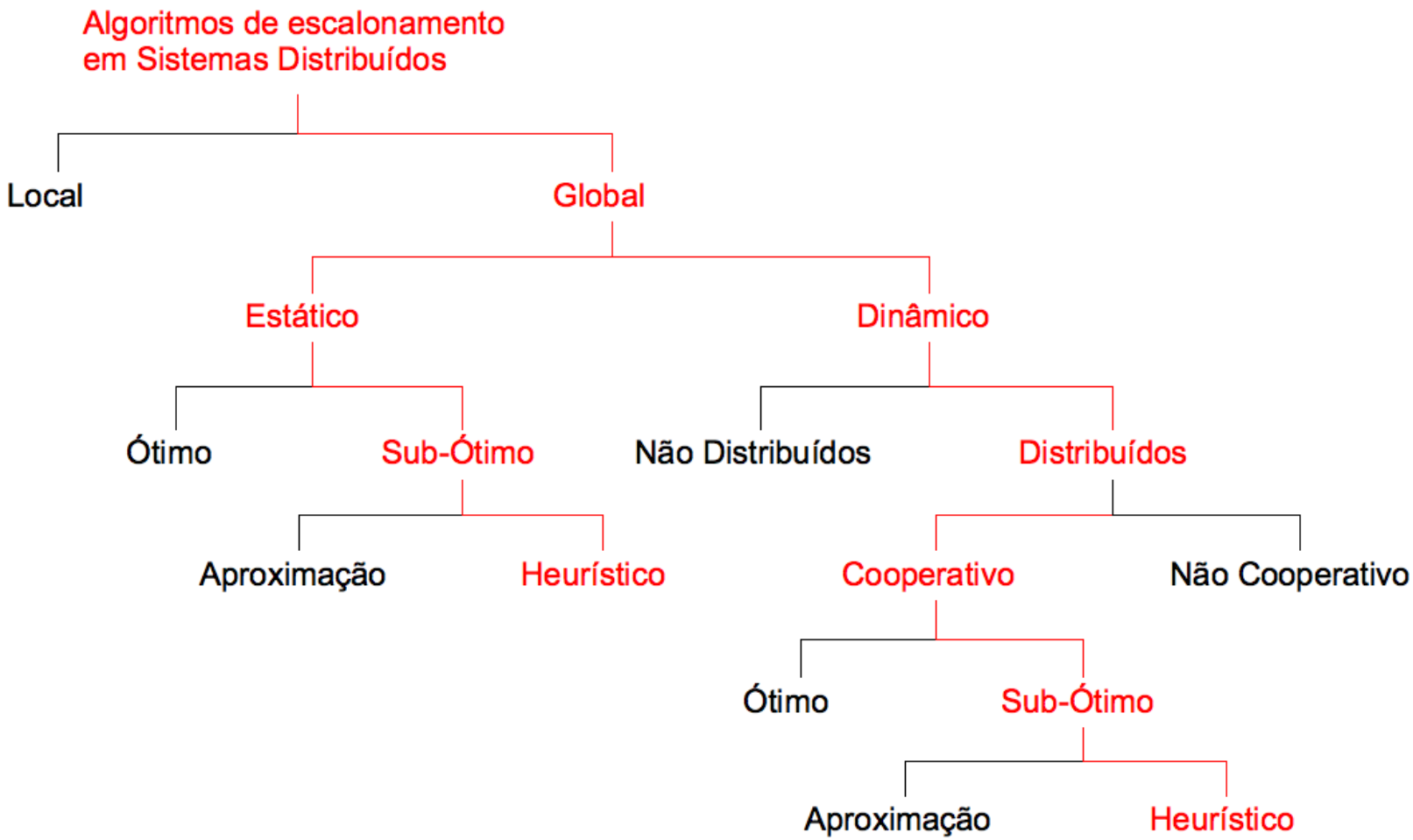


Classificação dos métodos de escalonamento



Casavant, T. L. and Kuhl, J. G., A taxonomy of scheduling in general-purpose distributed computing systems, IEEE Trans. Softw. Eng., 1988.

Classificação dos métodos de escalonamento

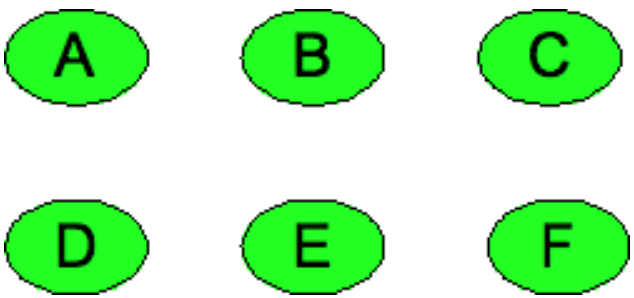


Casavant, T. L. and Kuhl, J. G., A taxonomy of scheduling in general-purpose distributed computing systems, IEEE Trans. Softw. Eng., 1988.

Escalonamento de tarefas

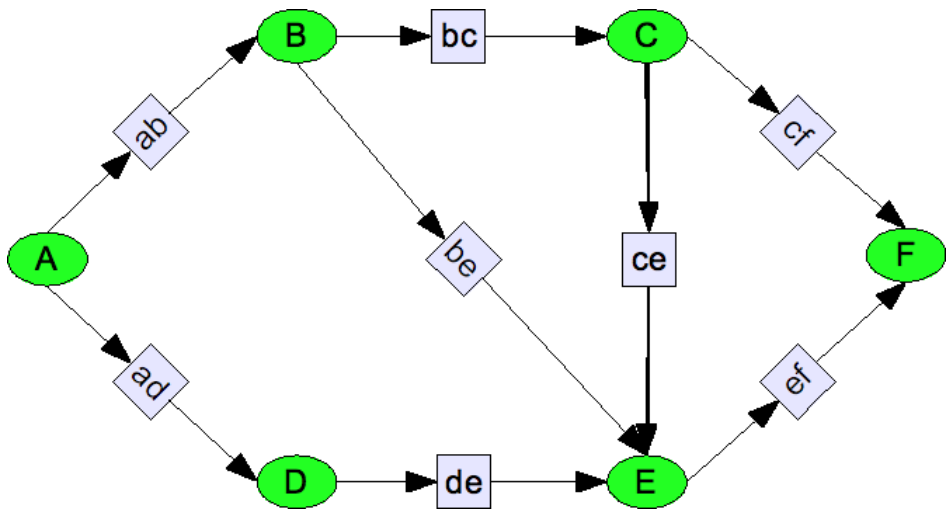
Tarefas Independentes

- Quando as dependências não existem, as tarefas formam grafos vazios
- Os grafos que não possuem arestas Bag-of-Tasks (BoT)



Tarefas Dependentes

- As tarefas que compõem uma aplicação podem ter dependências entre si
- Quando existem dependências, são representados por DAGs



Motivação do trabalho

- Necessidade de grande capacidade de processamento
- Uso correto da capacidade do processamento
- Problema de escalonamento de tarefas: NP-Completo
- Dentro da grade, ainda mais completo
  - Dinamicidade e heterogeneidade
  - Recursos fisicamente distantes uns dos outros

Objetivos

Objetivos geral

- Análise comparativa de algoritmos de escalonamento para tarefas dependentes, com diversos workloads reais.

Objetivos específicos

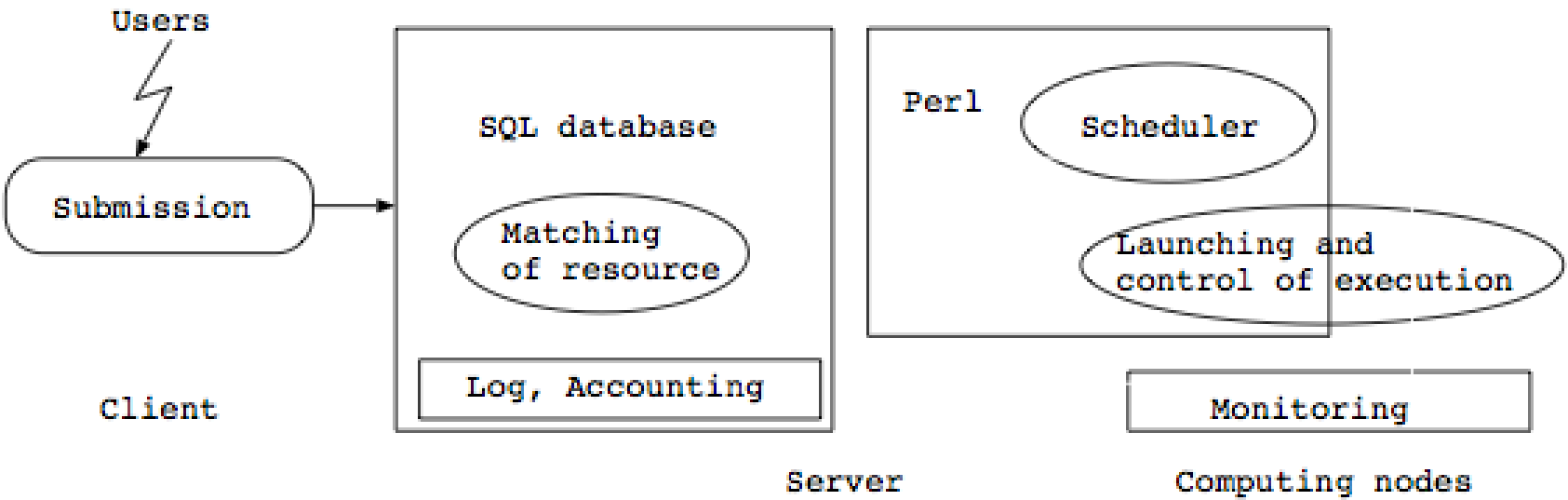
- Estudo dos diferentes escalonadores
- Estudo dos algoritmos de escalonamento
- Simulação dos algoritmos
- Estudo de métricas de comparação
- Análise comparativo com diferentes *workloads* reais

Roteiro

OAR

- Desenvolvido no Instituto Politécnico Nacional de Grenoble na França
- Código livre com licença GPL
- Banco de dados (MySQL ou PostgreSQL)
- Linguagens *Script* (Perl, Ruby)
- Outros componentes *Script* (SSH, Taktuk, . . .)
- Principais características
  - Suporte para multi-escalonadores (FIFO simples e FIFO com emparelhamento)
  - Multi-filas com prioridade
  - Propriedade de preempção
  - Mecanismos de políticas de *Backfilling*
  - Mecanismos de "reserva" avançada

Arquitetura do OAR

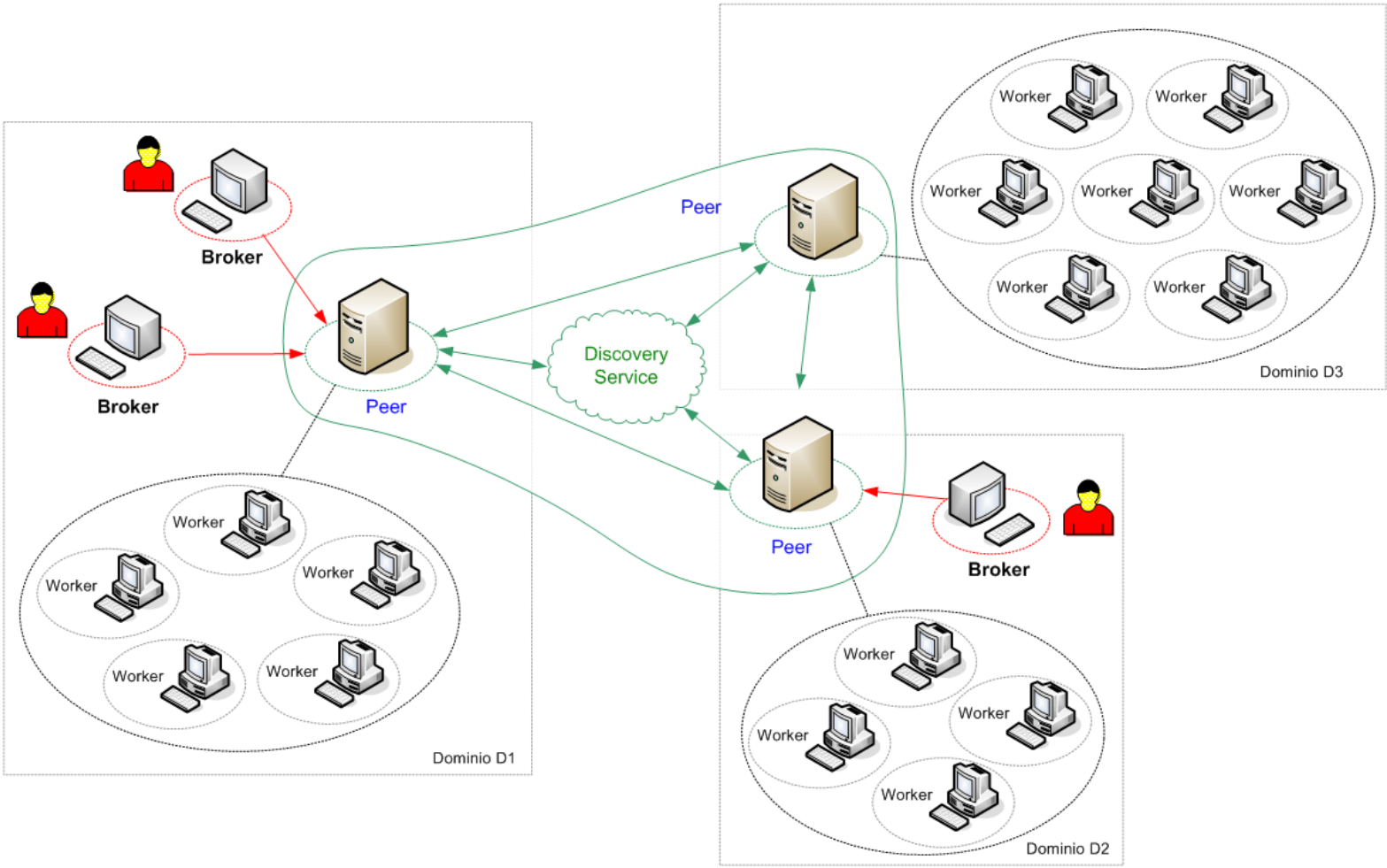


Capit, N et Al., A batch scheduler with high level components, CCGRID, 2005.

OurGrid

- Desenvolvido no Universidade de Campina Grande
- É um projeto de software livre com licença GPL
- Executa aplicações de tipo *Bag-of-Tasks*
- Escrito em *Java*
- Componentes:
  - *MyGrid*
  - *OurGrid peer*
  - *SWAN (Sandboxing Without A Name)*
- Algoritmos de escalonamento:
  - *Workqueue*
  - *Workqueue with Replication*
  - *Storage Affinity*

Arquitetura do OurGrid



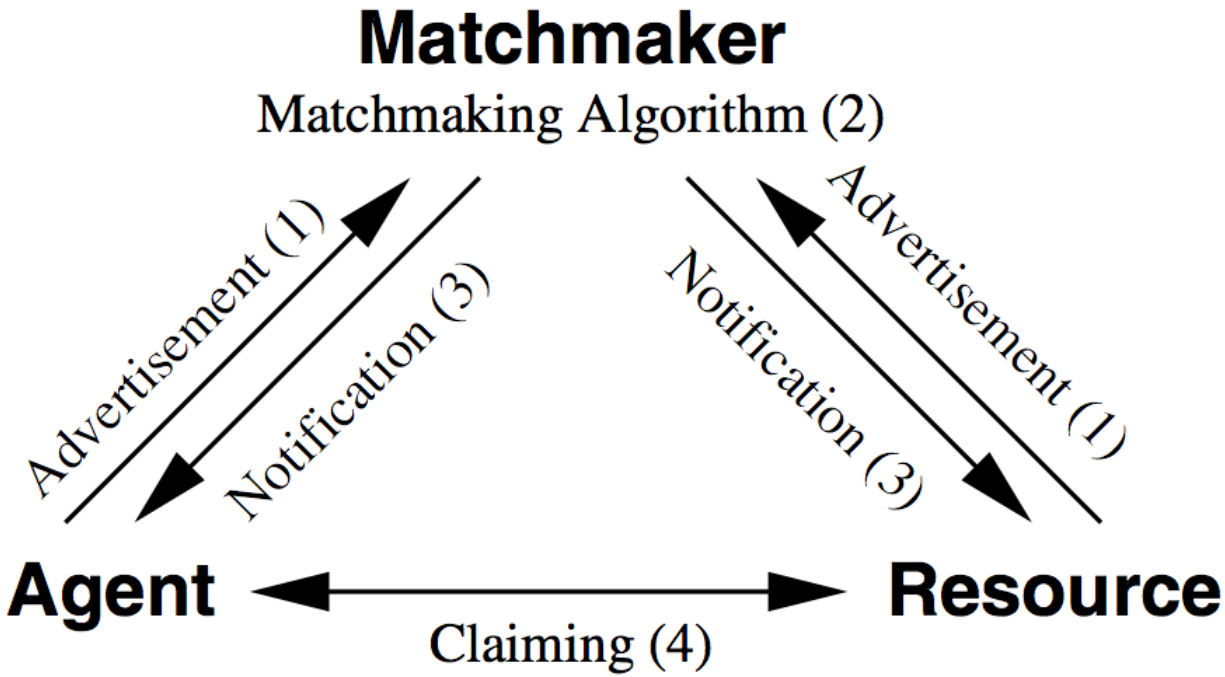
Cirne, Waldredo et Al., Labs of the World, Unite!!!, Journal of Grid Computing, 2006.

Condor

- Desenvolvido pela equipe Condor na Universidade de *Wisconsin-Madison*
- É um dos sistemas pioneiros na área da computação oportunista, lançado em 1984
- É software livre, possui licença Apache versão 2.0
- Provê mecanismos de enfileiramento e priorização de aplicações
- Os usuários podem submeter aplicações paralelas ou seriais ao Condor
- Políticas de escalonamento e monitoração de recursos
- Principais características:
  - *Matchmaking*;
  - *ClassAds*;

Matchmaking no Condor

- O escalonamento no Condor é feito através de *matchmaking*, decide quando, onde e como será executada uma determinada tarefa



Douglas Thain and Todd Tannenbaum and Miron Livny, Distributed computing in practice: the Condor experience, Concurrency - Practice and Experience, 2005.

ClassAds no Condor

- *Classified advertisements*,
  Cada recurso e tarefa, anunciam suas respectivas existências a entidades de *matchmaker*

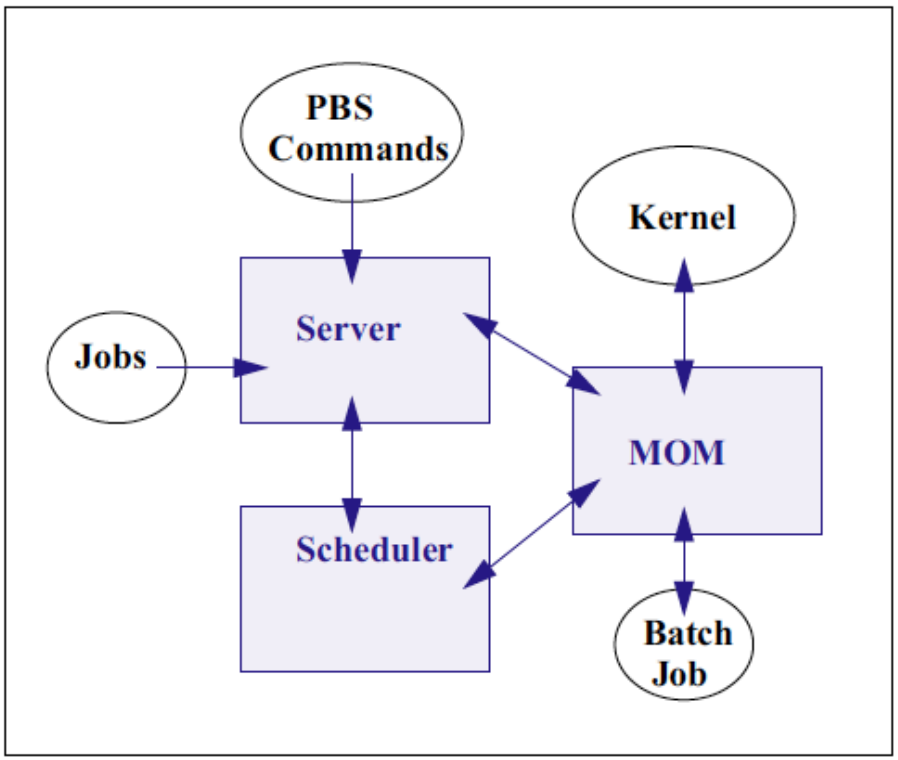
Job ClassAd	Machine ClassAd
[	[
MyType = "Job"	**MyType** = "Machine"
TargetType = "Machine"	**TargetType** = "Job"
Requirements =	**Machine** = "nostos.cs.wisc.edu"
((other.Arch=="INTEL" &&	**Requirements** =
other.OpSys=="LINUX")	(LoadAvg <= 0.300000) &&
&& other.Disk > my.DiskUsage)	(KeyboardIdle > (15 * 60))
Rank = (Memory * 10000) + KFlops	**Rank** = other.Department==self.Department
Crod = "/home/tannenba/bin/sim-exe"	**Arch** = "INTEL"
Department = "CompSci"	**OpSys** = "LINUX"
Owner = "tannenba"	**Disk** = 3076076
DiskUsage = 6000	]
]	

Douglas Thain and Todd Tannenbaum and Miron Livny, Distributed computing in practice: the Condor experience, Concurrency - Practice and Experience, 2005.

PBS

Portable Batch System

- Deselvolvido pela *Veridian Systems* para a NASA.
- *Veridiam Systems* foi adquirida pela *Altair Engineering*
- *Altair Engineering* distribui duas versões do PBS:
  - *PBS Professional*, versão comercial
  - *OpenPBS*, distribuição livre
- Um derivado do OpenPBS e é ativamente desenvolvido, suportado e mantido pela *Cluster Resources Inc.*, chamado Torque



Torque

Terascale Open-Source Resource and QUEue Manager

- Deselvolvido pela *Cluster Resources Inc.*
- Licença OpenPBS(Portable Batch System) v2.3
- Mais de 1200 linhas de código modificadas
- Algumas características inseridas ao OpenPBS pelo Torque são:
  - Tolerância a falhas
  - Interface de escalonamento
  - Escalabilidade
  - Usabilidade

Maui

- Deselvolvido pela *Cluster Resources Inc.*
- Licença *End User Open Source* de *Cluster Resources Inc.*
- Surgiu com o propósito de auxiliar algumas carências *IBM LoadLeveler*
- Características do Maui:
  - Priorização de tarefas
  - Reserva de recursos
  - Políticas de *Backfill*
  - Suporte de diagnóstico
  - Modo de teste

Roteiro

Algoritmos de Escalonamento

Tarefas Independentes

- WQR
- Sufferage
- Storage Affinity

Elizeu Santos-Neto *et Al.*, *Exploiting Replication and Data Reuse to Efficiently Schedule Data-Intensive Applications on Grids*, Workshop on Job Scheduling Strategies for Parallel Processing, 2004

Algoritmos de Escalonamento

Tarefas Independentes

- WQR
- Sufferage
- Storage Affinity

Elizeu Santos-Neto *et Al.*, *Exploiting Replication and Data Reuse to Efficiently Schedule Data-Intensive Applications on Grids*, Workshop on Job Scheduling Strategies for Parallel Processing, 2004

Tarefas Dependentes

- *Heterogeneus Earliest Finish Time*
- *Critical Path On a Processor*
- *Path Clustering Heuristic*

HEFT (Heterogeneus Earliest Finish Time)

Priorização de tarefas

- Atribuir prioridade às tarefas
- Calculo da prioridade, baseado na média dos custos de computação e custos de comunicação
- lista das tarefas

Seleção de recursos

- Selecionar a tarefa t_i da lista com maior prioridade
- Para cada recurso $r \in R$ é calculado o EST e EFT de cada tarefa t_i
- r_j é alocada ao recurso que minimiza o EFT da tarefa t_i

Topcuouglu, Haluk et Al., Performance-Effective and Low-Complexity Task Scheduling for Heterogeneous Computing, IEEE Trans. Parallel Distrib. Syst., 2002.

CPOP (Critical Path On a Processor)

Priorização de tarefas

- Atribuir prioridade às tarefas
- Calculo das prioridades baseados no custo de computação e comunicação
- $|CP|$ é o caminho crítico

Seleção de recursos

- *PCP* (*critical-path processor*)
- Se a tarefa selecionada está no caminho crítico, então é escalonada no recurso de caminho crítico
- ela é atribuída a um recurso que minimiza o EFT

Topcuouglu, Haluk et Al., Performance-Effective and Low-Complexity Task Scheduling for Heterogeneous Computing, IEEE Trans. Parallel Distrib. Syst., 2002.

PCH (Path Clustering Heuristic)

Seleção de tarefas e agrupamento

- seleciona tarefas que formarão cada *cluster* que serão escalonadas no mesmo recurso
- A primeira tarefa que compõe um *cluster* cls_k é a tarefa não escalonada com maior prioridade
- A partir dessa tarefa, o algoritmo faz uma busca em profundidade

Bittencourt, Luiz F et Al., Uma Heuristica de Agrupamento de Caminhos para Escalonamento de Tarefas em Grades Computacionais, SBRC, 2006.

PCH (Path Clustering Heuristic)

Seleção de recursos

- A seleção de recursos se dá através do cálculo de valores
- qual recurso terminará a execução do *cluster* em menor tempo
- O fator que determina em qual recurso um *cluster* será escalonado é o *EST* do sucessor da último tarefa do *cluster* considerado

Bittencourt, Luiz F et Al., Uma Heuristica de Agrupamento de Caminhos para Escalonamento de Tarefas em Grades Computacionais, SBRC, 2006.

Roteiro

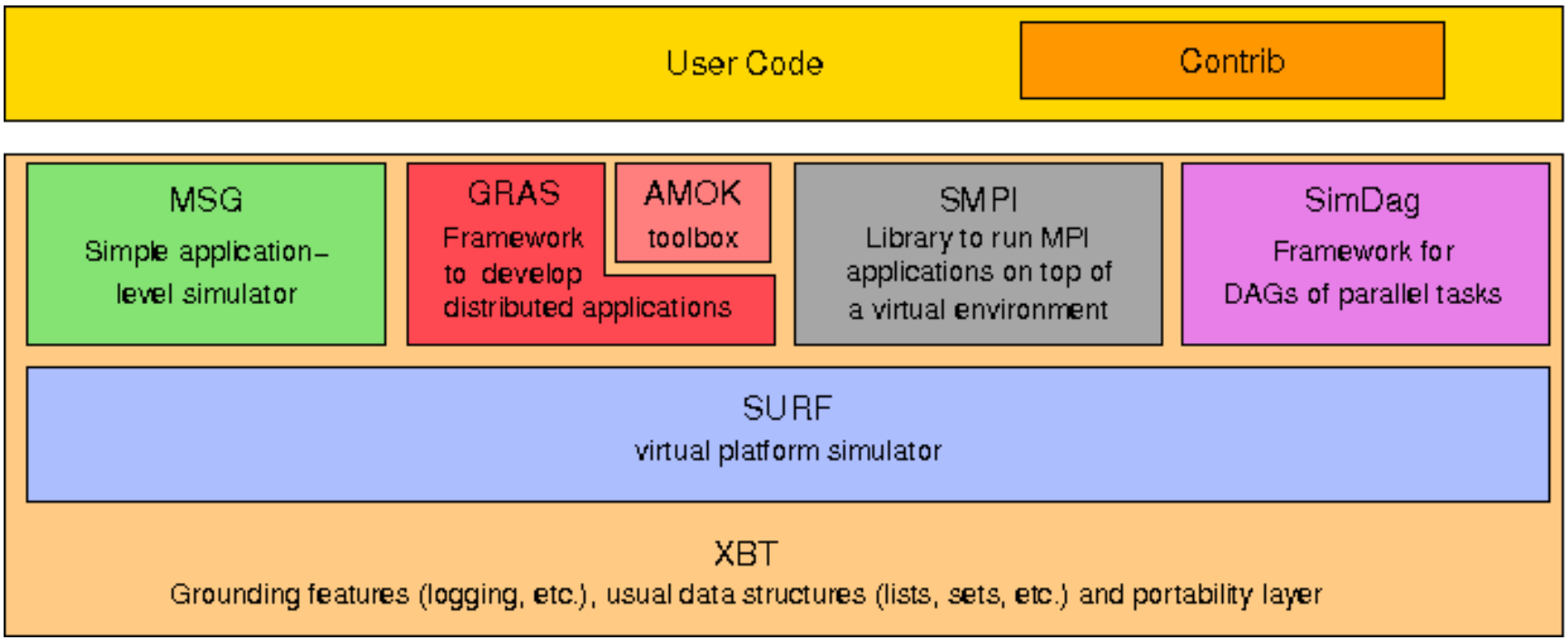
Simulador

Principais Simuladores

- *Bricks*, ferramenta empregada para simular sistemas distribuidos, cliente-servidor, ...
- *Optorsim*, criado especificamente para estudar replicação de dados
- *GridSim*, permite modelagem e simulação de entidades em sistemas de computação paralela e distribuída
- *SimGrid*,
  - Fornece importantes funcionalidades para a simulação de aplicações distribuídas em ambientes heterogêneos
  - Possui uma comunidade ativa

Arquitetura do SimGrid

Componentes



Casanova, Henri and Legrand, Arnaud and Quinson, Martin, SimGrid: a Generic Framework for Large-Scale Distributed Experiments, IEEE Computer Society Press, 2008.

Modelagem da Plataforma e os Workloads

Arquivo XML-Plataforma

```
<?xml version='1.0'?>
<!DOCTYPE platform SYSTEM "simgrid.dtd≫
<platform version="2≫
<host name="C1-00" power="1E8"/>
<host name="C1-01" power="2E8"/>
...
<link name="1" bandwidth="1E6"
                  latency="1E-5"/>
<link name="2" bandwidth="1E6"
                  latency="1E-5"/>
...
<route src="C1-00" dst="C1-01≫
  <link:ctn id="1"/>
  <link:ctn id="2"/>
</route>
</platform>
```

Arquivo XML-Workload - DAX

```
<?xml version='1.0'?>
<adag xmlns="http://pegasus.../DAX" ...>
  <job id="ID00" namespace="montage"...>
    <uses file="region.hdr" size="304"/>
    ...
  </job>
  <job id="ID24" namespace="montage"...>
    <uses file="mosaic.fits" size="18611"/>
    <uses file="shrunken.fits" size="18611"/>
    ...
  </job>
  ...
  <child ref="ID24≫
    <parent ref="ID00"/>
  </child>
</adag>
```

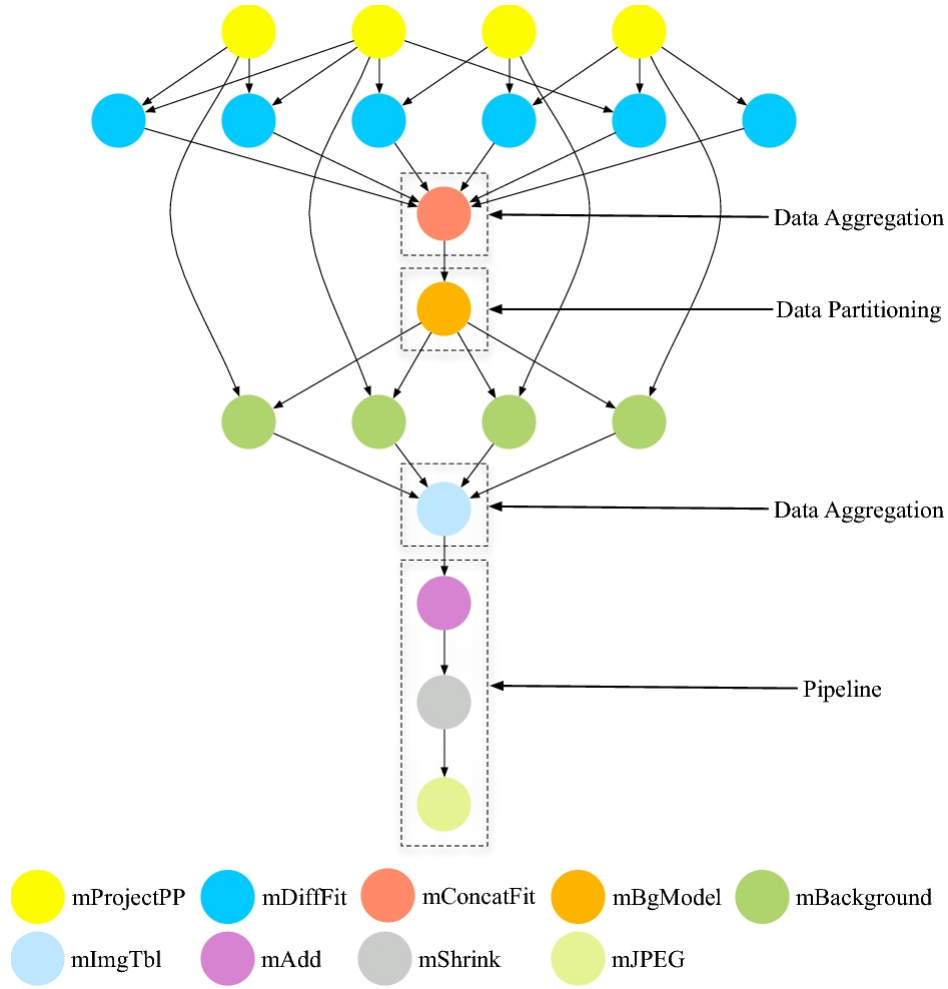
SimDag

- Simulação de DAGs
- Tarefa paralela `SD_task_t`
- Dependência `SD_task_dependency`
- Carregador para DAX (*Directed Acyclic Graph in XML*)
  - `SD_daxload( dax_file.xml )`
  - `SD_dotload( dot_file.xml )`
- Recursos computacionais
  - Recurso `SD_workstation_t`, poder computacional
  - Enlace (*link*), largura de banda e latência

Roteiro

Os workloads reais avaliados neste trabalho

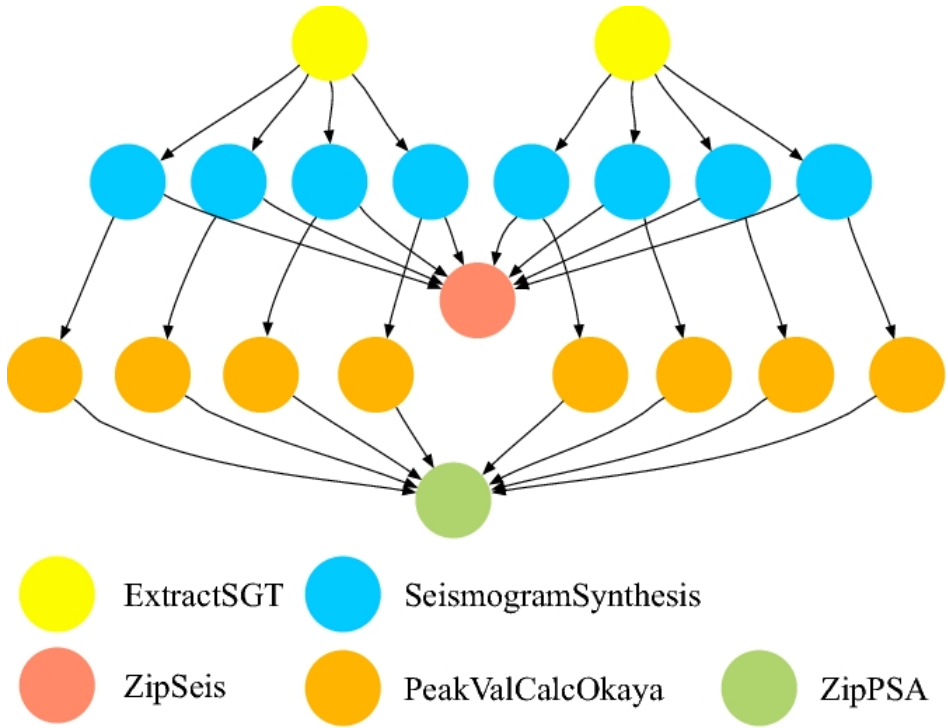
- O *workflow Montage*



Shishir Bharathi et Al., Characterization of Scientific Workflows, Workshop on Workflows in Support of Large-Scale Science, 2008.

Os workloads reais avaliados neste trabalho

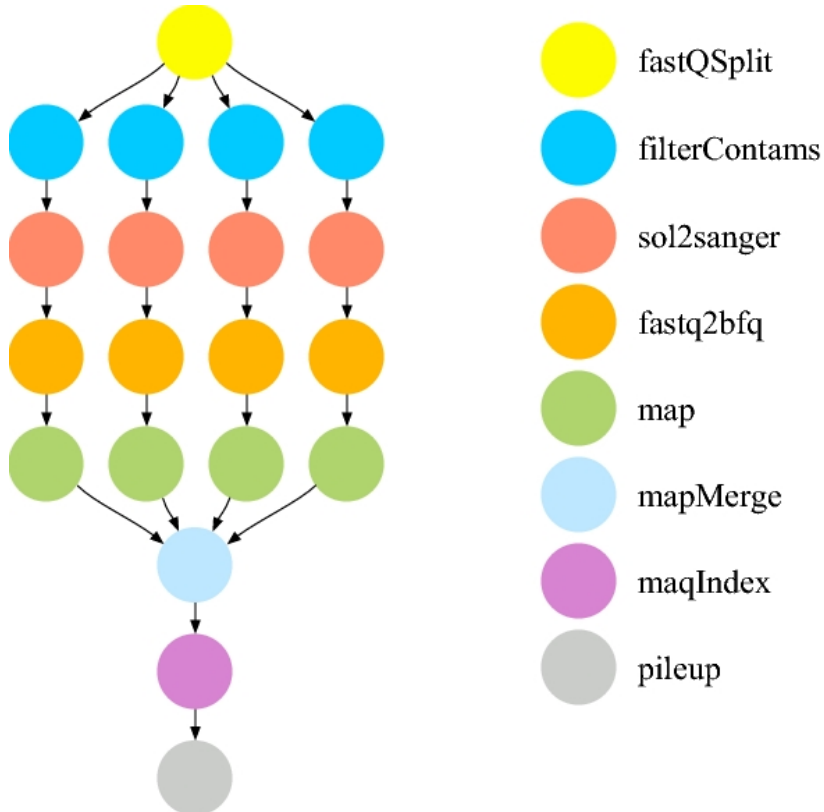
- O *workflow CyberShake*



Shishir Bharathi et Al., Characterization of Scientific Workflows, Workshop on Workflows in Support of Large-Scale Science, 2008.

Os workloads reais avaliados neste trabalho

- O *workflow Genome*



Shishir Bharathi et Al., Characterization of Scientific Workflows, Workshop on Workflows in Support of Large-Scale Science, 2008.

Descrição dos Cenários

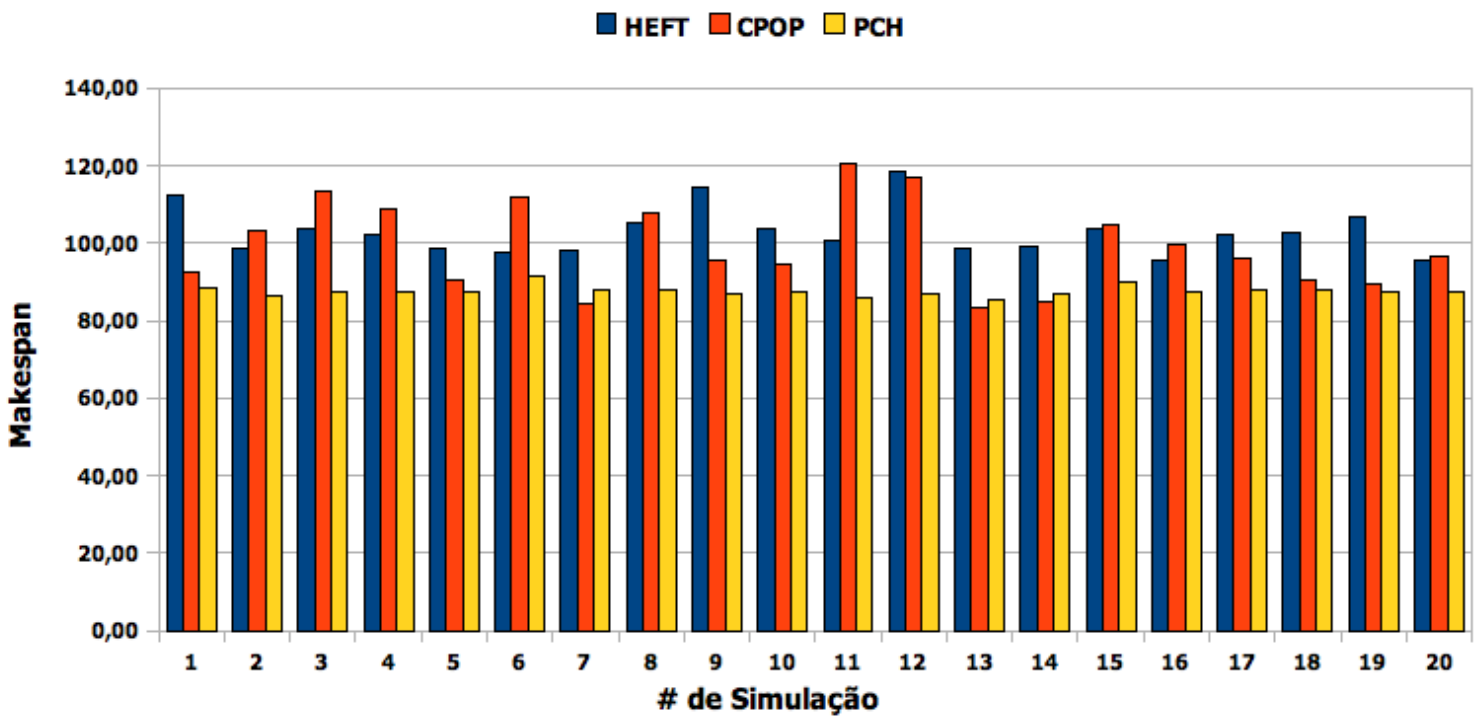
- Heterogeneidade dos Tamanhos das Tarefas
- Escalabilidade do Workload
- Heterogeneidade da Grade

Heterogeneidade dos Tamanhos das Tarefas

Resultados das simulações com 50 tarefas

	HEFT	CPOP	PCH
Média	102,89	99,29	87,61
Desvio Padrão	6,23	11,13	1,26

Tabela: Média e desvio padrão de 20 simulações com 50 tarefas do Montage

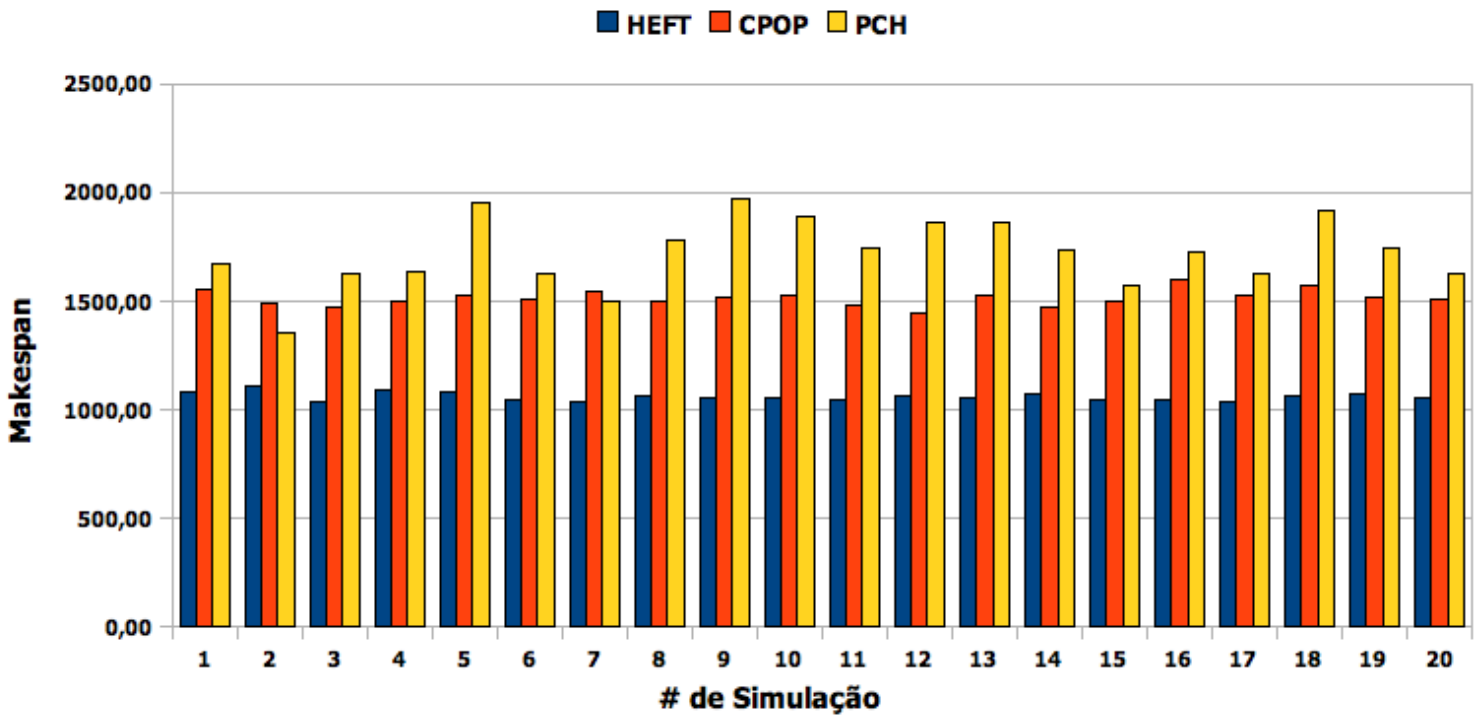


Heterogeneidade dos Tamanhos das Tarefas

Resultados das simulações com 1000 tarefas

	HEFT	CPOP	PCH
Média	1061,69	1515,22	1721,68
Desvio Padrão	19,10	35,48	158,46

Tabela: Média e desvio padrão de 20 simulações com 1000 tarefas do Montage



Escalabilidade do Workload

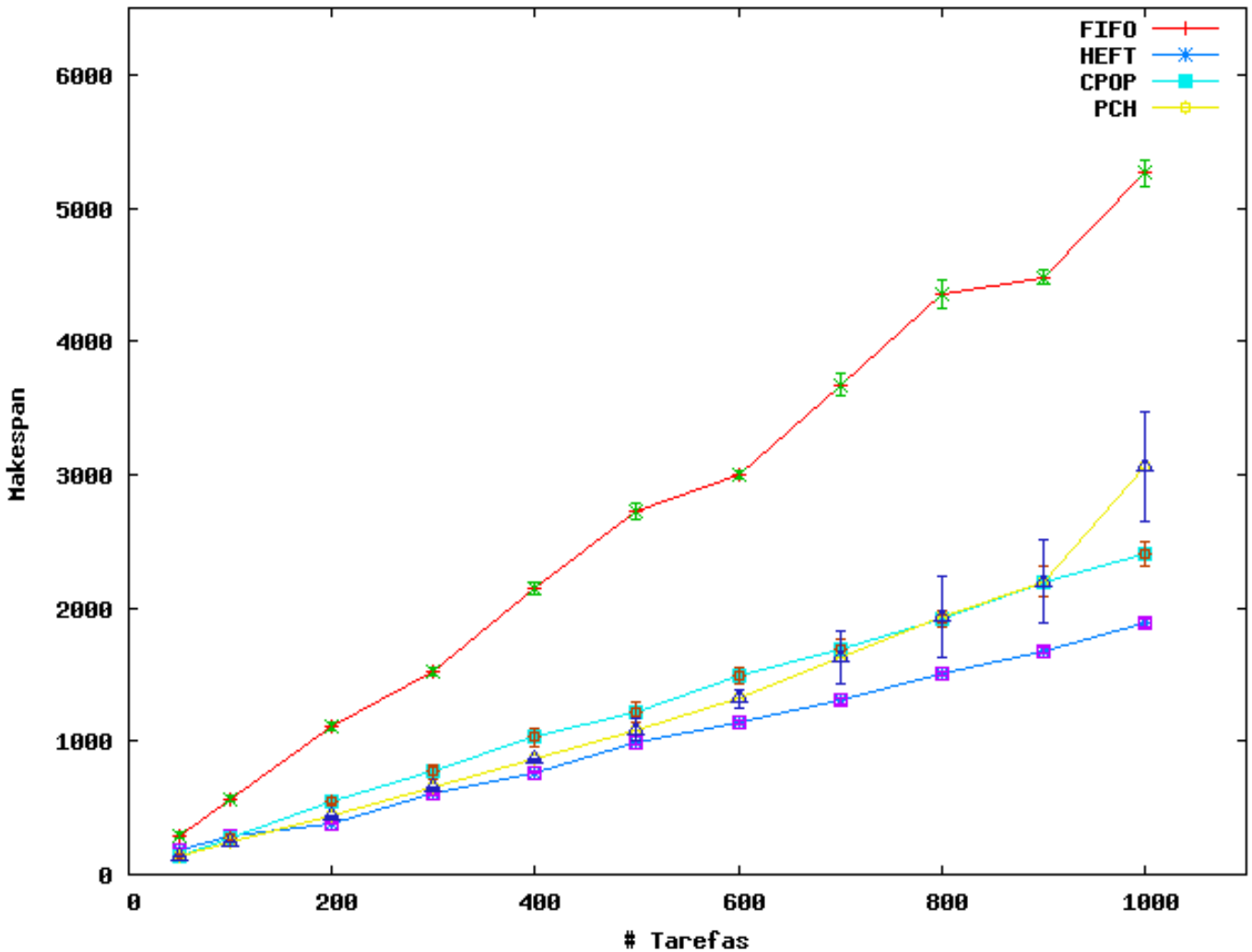
Plataforma utilizada neste cenário

Id	Poder Comp. (*MFlops/s*)
C1-00	100
C1-01	100
C1-02	100
C1-03	100
C1-04	100
C2-05	500
C2-06	500
C2-07	500
C2-08	500
C2-09	500

Tabela: Id das máquinas, poder computacional de cada uma

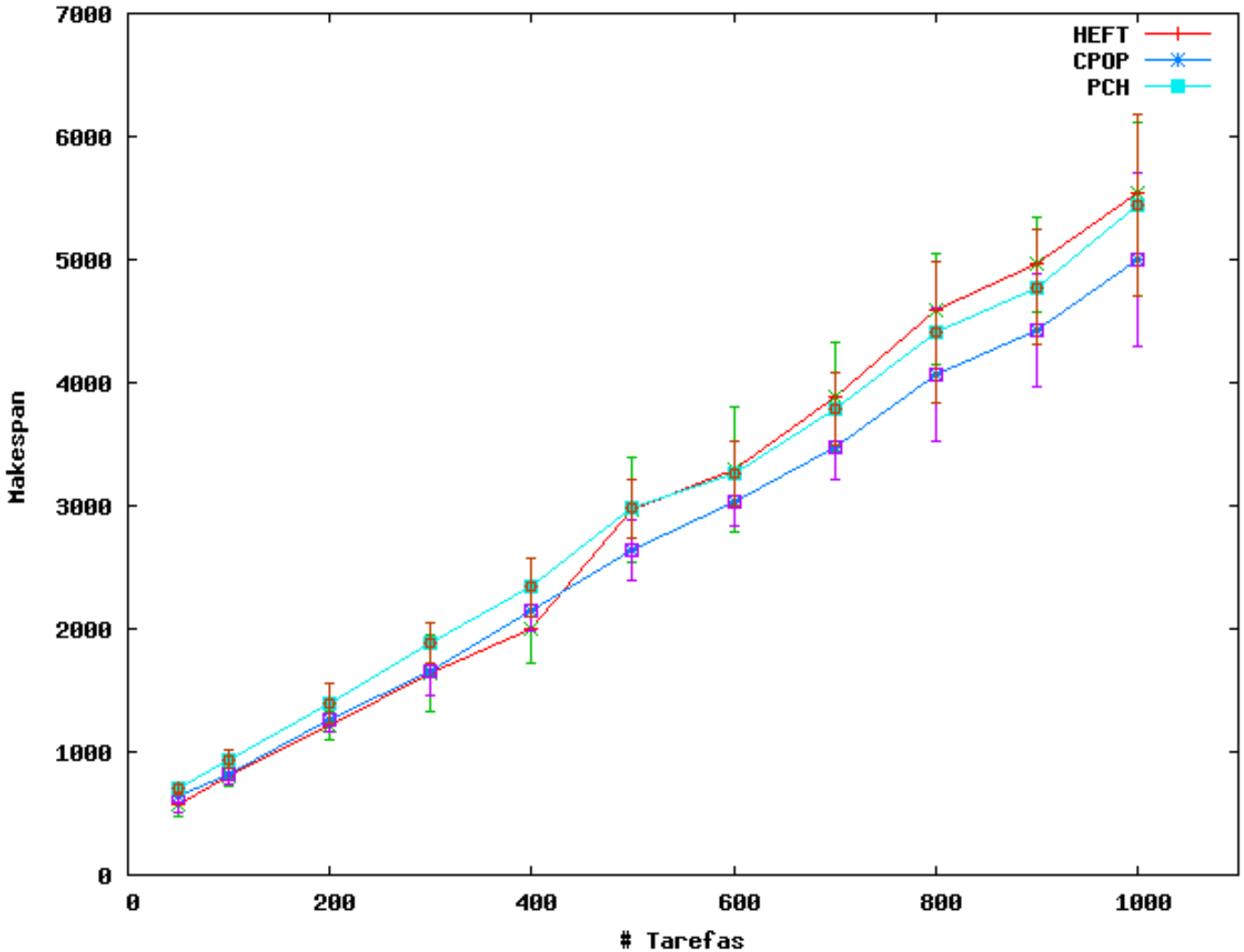
Escalabilidade do Workload

Escalabilidade do Workload Montage



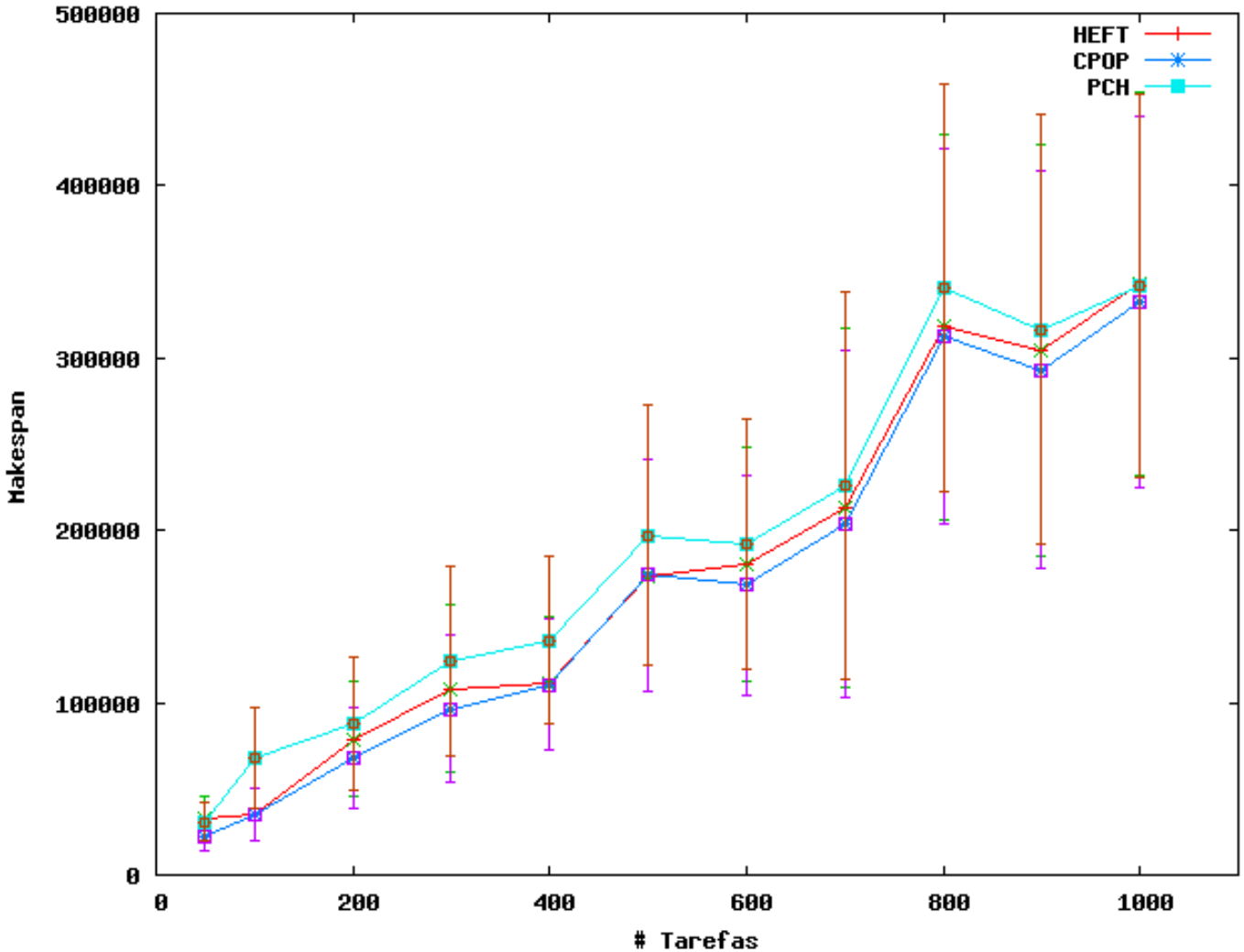
Escalabilidade do Workload

Escalabilidade do Workload Cybershake



Escalabilidade do Workload

Escalabilidade do Workload Genome



Heterogeneidade da Grade

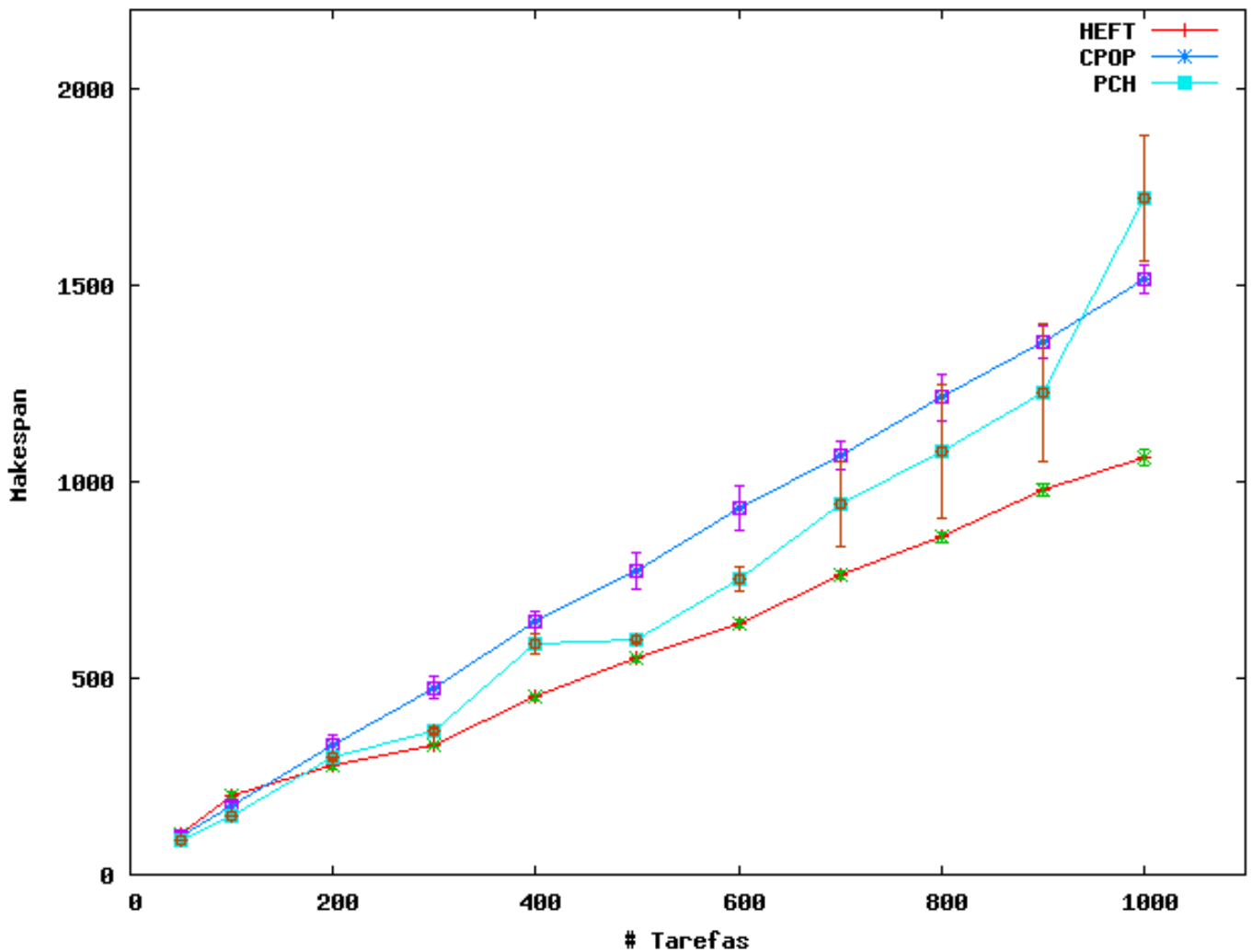
Plataforma utilizada neste cenário

Id	Poder Comp. (*MFlops/s*)
C1-00	100
C1-01	200
C1-02	300
C1-03	400
C1-04	500
C2-05	600
C2-06	700
C2-07	800
C2-08	900
C2-09	900

Tabela: Id das máquinas, poder computacional de cada uma

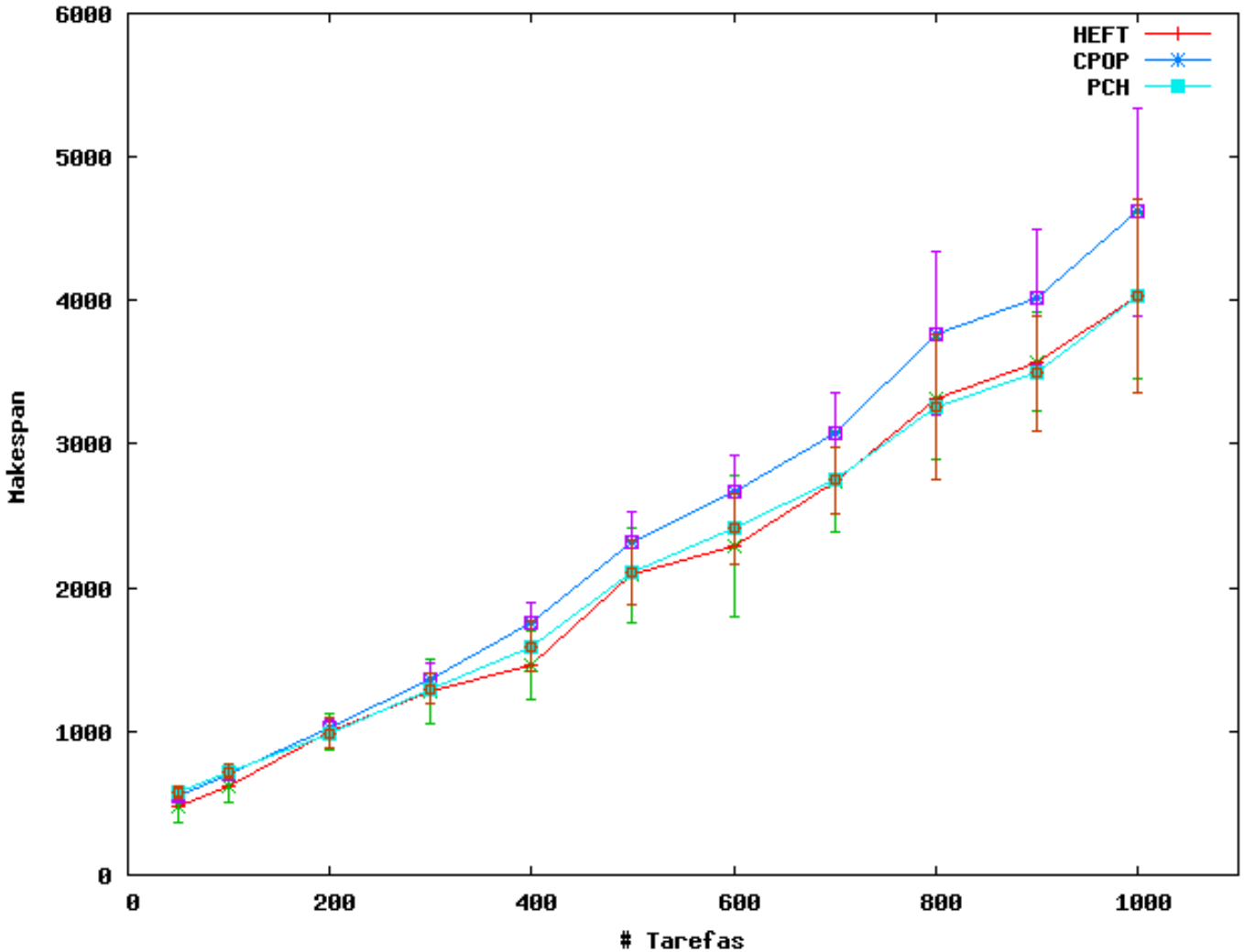
Escalabilidade do Grade

Escalabilidade do Grade no workload Montage



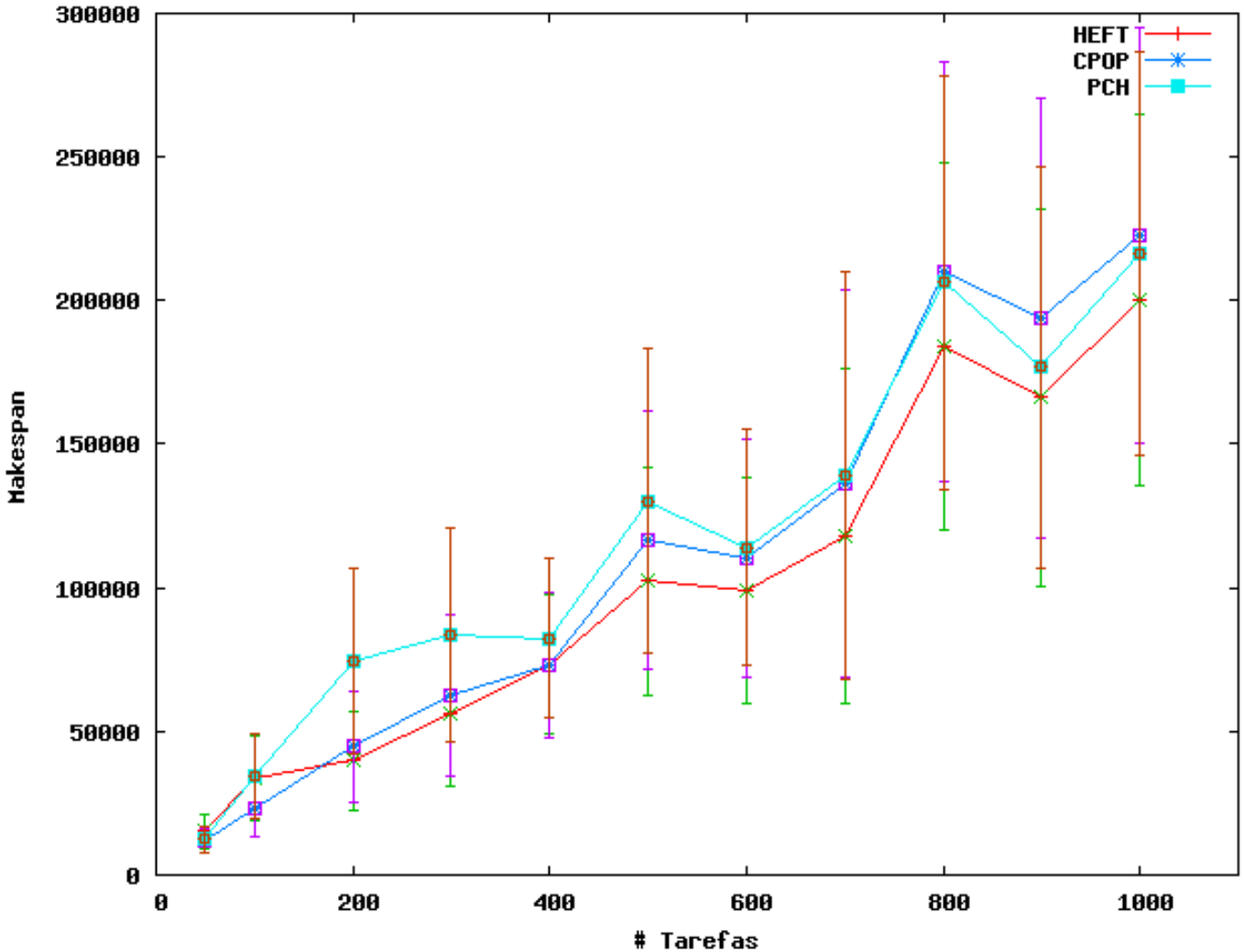
Escalabilidade do Grade

Escalabilidade do Grade no workload Cybershake



Escalabilidade do Grade

Escalabilidade do Grade no workload Genome



Roteiro

Plano de Trabalho e Cronograma

Atividades	**Anos e Semestres**							
	'09	**2010-2011**						
		1º	2º	Dez	Jan	Fev	Mar	Abr
Disciplinas obrigatórias	x	x						
Levantamento Bibliográfico	x	x						
Estudo - diversos escalonadores	x	x						
Análise - ambientes d simulação	x	x	x					
Comparação dos escalonadores			x	x				
Implementação dos algoritmos			x	x				
Estudo - métricas d comparação			x	x	x			
Estudo com workloads reais				x	x			
Suporte para mais workloads					x	x	x	
Artigos			x		x	x	x	x
Redação da dissertação e defesa						x	x	x

Tabela: Cronograma de atividades

Roteiro

Conclusões

Considerações Finais

- Neste estudo, são avaliados os algoritmos de escalonamento para grades computacionais:
  - O *Path Clustering Heuristic* (PCH)
  - O *Critical Path on a Processor* (CPOP)
  - O *Heterogeneuos Earliest Finish Time* (HEFT)
  - Além deles foi implementado um escalonamento simples de tipo FIFO
- A heurística HEFT apresenta bom desempenho a medida que o número de tarefas foi acrescentado, tanto o PCH quanto o CPOP não apresentaram bom desempenho com relação ao HEFT
- O uso de um algoritmo de escalonamento especializado, é fundamental para obter um "bom" desempenho no escalonamento

Muchas Gracias!!!