

# Universidade Federal do Rio Grande do Sul

## Instituto de Informática

### *Determinação da Disponibilidade do Recurso para Sistemas de Global Computing*

**Autor: Eder Stone Fontoura**

**Orientador: Cláudio Fernando Resin Geyer**

**WSPPD - 08/2007**

# Súmula

- Introdução
- Modelo
- Motivação
- Proposta
  - Arquiteturas (Server, *Worker*)
  - Disponibilidade
- Experimentos e Resultados
- Conclusões
- Trabalhos Futuros
- Referências

# Introdução

- Global Computing, Volunteer Computing, Desktop Grids
  - recursos requisitam jobs ao servidor
  - servidor entrega os jobs solicitados
  - recurso processa os jobs e transfere os resultados
- XtremWeb e BOINC são exemplos de sistemas que seguem o modelo de Global Computing
- Recursos altamente voláteis
  - podem entrar e sair do sistema livremente
  - pode causar a necessidade de um novo escalonamento do job
- Considerar a disponibilidade dos recursos participantes pode ser uma boa alternativa para a realização do escalonamento de jobs

# Modelo

- **Modelo característico de Global Computing:**

- Elemento central (Servidor)
- Recursos voluntários (Workers)
- Recurso inicia o escalonamento
- Recursos altamente heterogêneos, distribuídos e em grande quantidade

- **Aplicação:**

- Composta por jobs(tarefas) sem comunicação ou precedência (bag-of-tasks)
- Tempo total de execução da aplicação = horário de entrega do resultado do último job – horário de submissão do 1º job

- **Disponibilidade do recurso:**

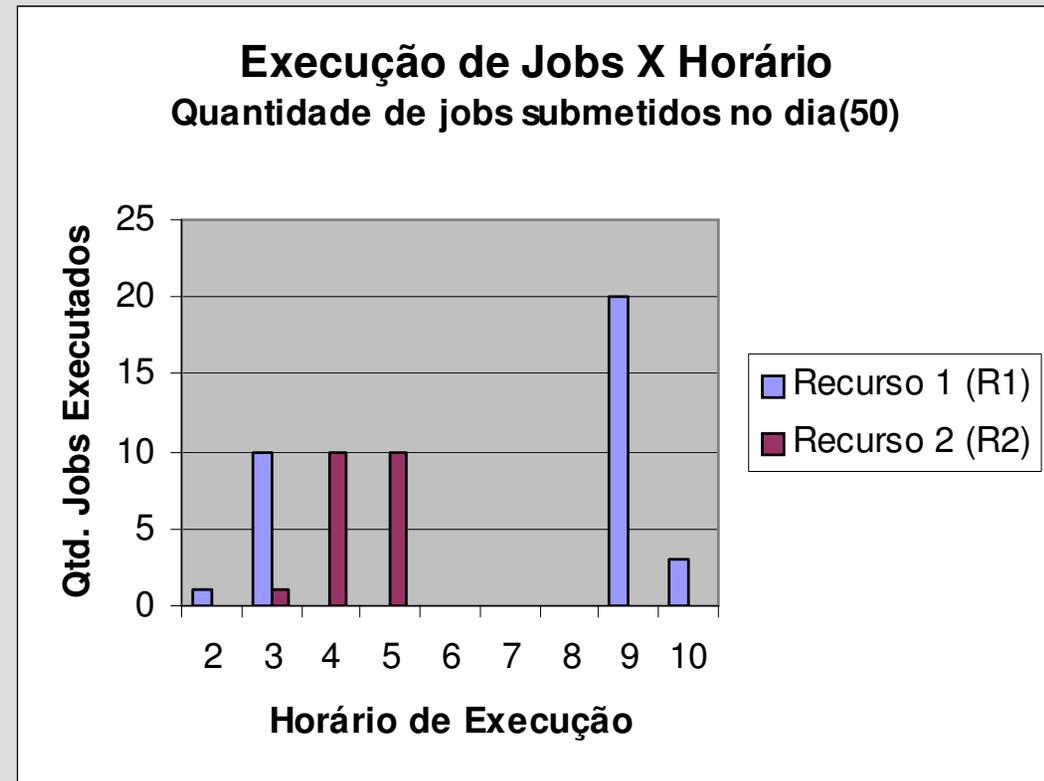
- neste contexto é considerada como os momentos em que o recurso pode executar jobs do sistema de Global Computing

# Motivação

- Medidas de disponibilidade apresentadas em [Byun, Choi, Baik e Hwang 2005], [Taufer, Teller, David e Brooks 2005], [Estrada, Flores, Taufer, Teller, Kerstens, e Anderson 2006] e [Choi, Baik, Hwang, Gil e Yu 2004] elevaram o *throughput* do sistema e reduziram o tempo total de execução de uma aplicação
- Problema das propostas:
  - Qual recurso estaria disponível em um determinado momento(Ex: 2:15)?
  - O servidor tem que manter registradas as disponibilidades para cada recurso para poder determinar o mais disponível
    - Não prejudica as atividades do servidor?
- Disponibilidade =  $\text{qtd\_jobs\_executados} / \text{qtd\_jobs\_submetidos}$

# Motivação

- Recurso mais disponível:
  - $R1 = 34/50 = 0,68^*$
  - $R2 = 21/50 = 0,42$
- Recurso mais disponível em um intervalo de tempo:
  - $R1(3;3:59) = 10/15 = 0,66...^*$
  - $R2(3;3:59) = 1/15 = 0,066...$



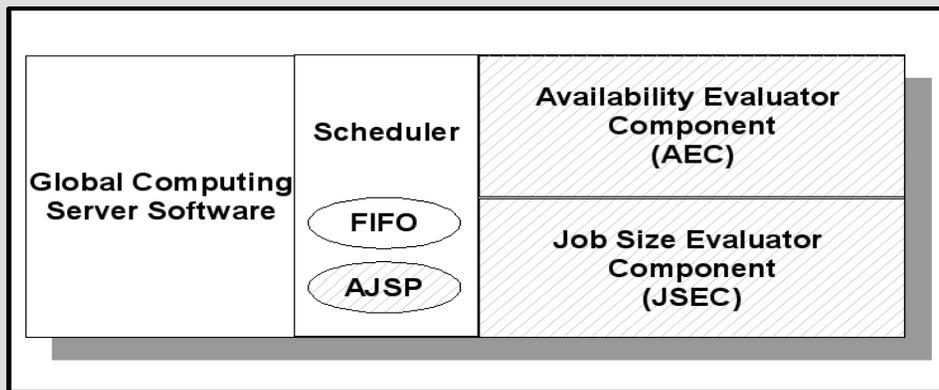
\* Recurso mais disponível segundo [Taufer, Teller, David e Brooks 2005]

# Proposta

- Apresentar um conjunto de componentes que permita a elaboração de uma medida de disponibilidade para uso em escalonadores de Global Computing;
- Obter a disponibilidade através do padrão de utilização do recurso (Duração):
  - *Os usuários de computadores apresentam um comportamento padrão no que diz respeito à utilização de seu recurso [Begole, Tang, Smith and Yankelovich 2002], Mutka [Mutka 1991] e Taufer [Taufer, Teller, David e Brooks 2005]*

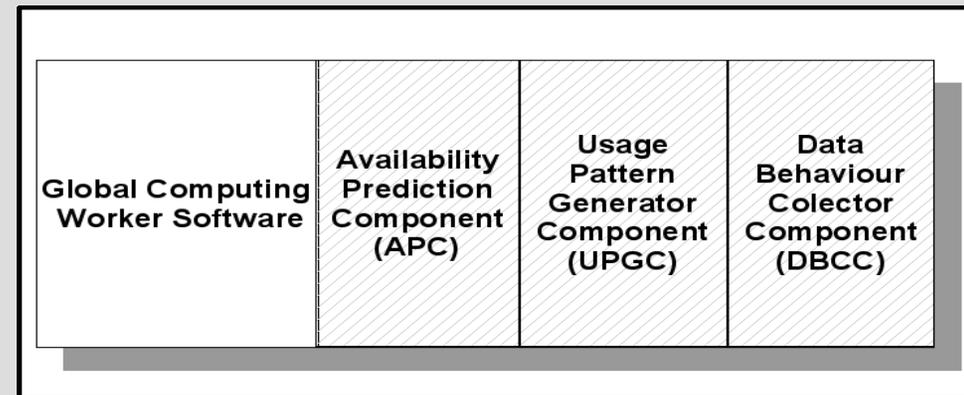
# Proposta – Arquitetura

## Server



- **JSEC** – determina o rank de tamanho de job
- **AEC** – determina o rank de disponibilidade do recurso
- **AJSP** – usa o AEC e o JSEC para determinar o escalonamento

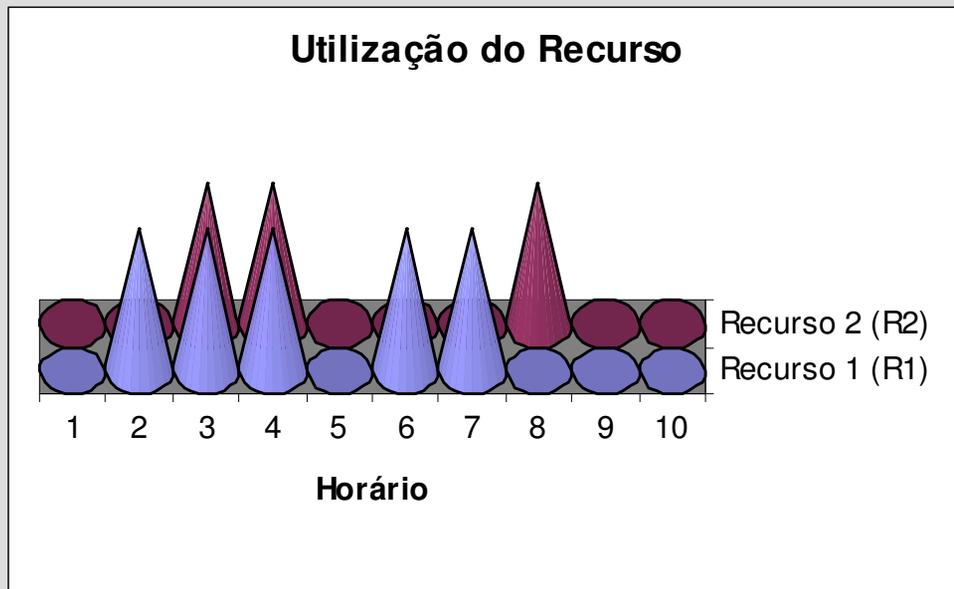
## Worker



- **DBCC** – coleta de dados de utilização do recurso
- **UPGC** – geração do padrão de utilização do recurso
- **APC** – geração da disponibilidade do recurso

# Proposta - Disponibilidade

- Detectar os períodos de disponibilidade do recurso:

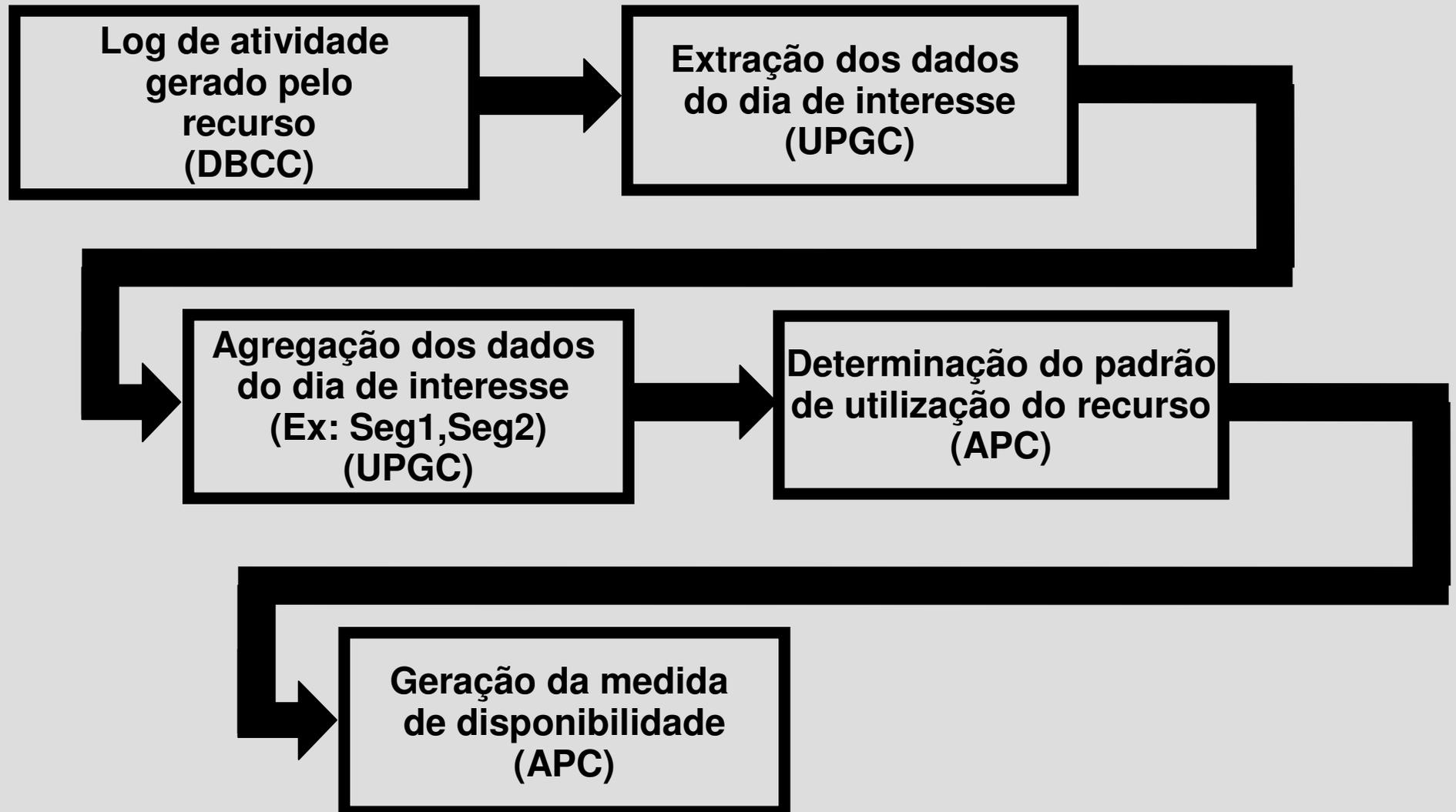


Considerando histórico de utilização

Elaboração do padrão de utilização

Geração da medida de disponibilidade  
(duração)

# Proposta - Disponibilidade



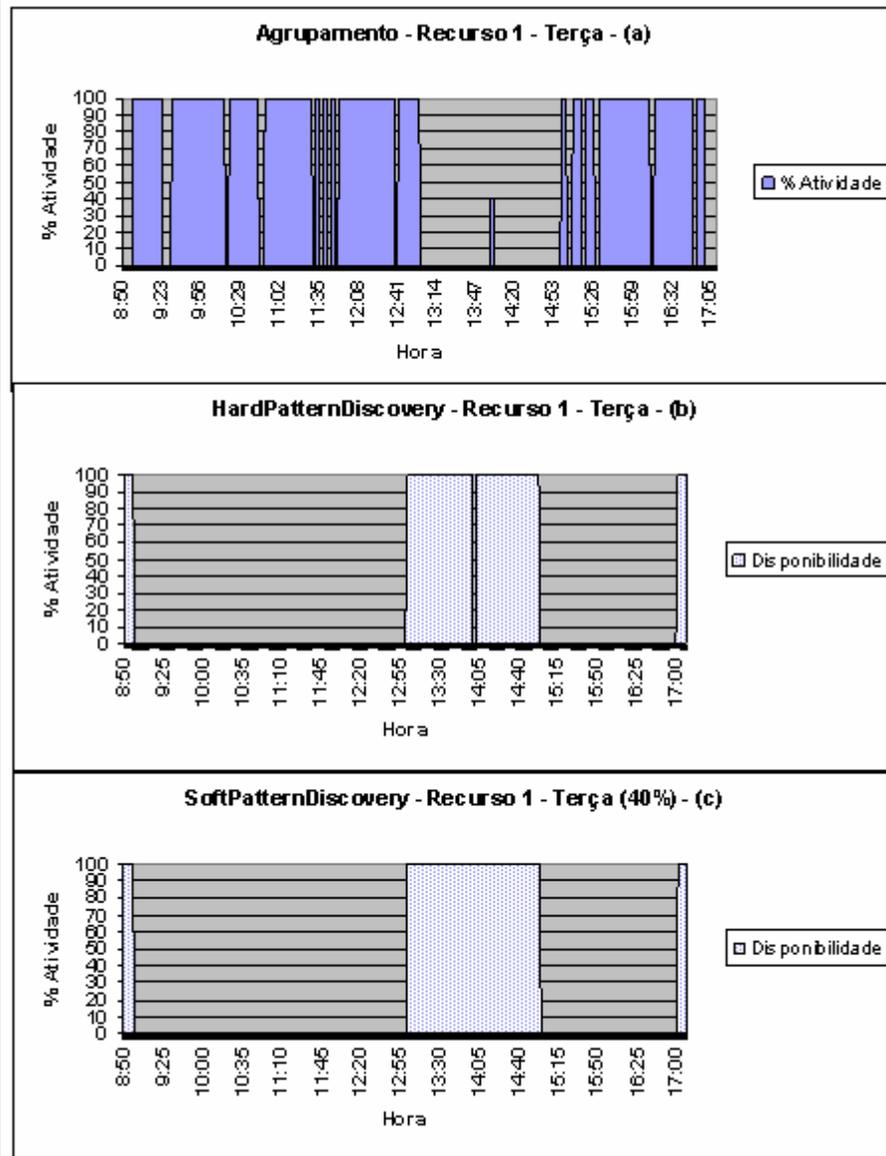
# Experimentos e Resultados

- Simulador de usuário:
  - Comporta-se como um usuário utilizando teclado/mouse, instalado no recurso voluntário e executa um comportamento diferente para cada dia da semana
- DBCC, UPGC e APC (Worker):
  - Window (determina o menor período de disponibilidade), HardPatternDiscovery e SoftPatternDiscovery (determinar o padrão de disponibilidade). Configuração: Window = 20 minutos e SoftPatternDiscovery com 40%
  - Percentual de uso para cada minuto:

$$Pu(xy) = \frac{((A_{xya} \cdot 1) + (A_{xyb} \cdot 2) + (A_{xyc} \cdot 3) + (A_{xyd} \cdot 4))}{10} \cdot 100$$

- Coleta de dados:
  - coletados dois dias de utilização de um recurso (com simulador de usuário)
  - Foram inseridos/retirados minutos de atividade dos dados para simular pequenas alterações comportamentais

# Resultados



HardPatternDiscovery		SoftPatternDiscovery	
Horário	Dur.	Horário	Dur.
00:00 - 8:59	539	00:00 - 8:59	539
13:01 - 13:59	58	13:01 - 14:58	118
14:03 - 14:59	56	17:01 - 23:59	419
17:01 - 23:59	419		

- HardPatternDiscovery
  - 4 períodos de disponibilidade
  - mais tempo para identificar novos períodos de disponibilidade
- SoftPatternDiscovery
  - 3 períodos de disponibilidade
  - mais tempo para identificar que um período de disponibilidade deixou de existir

# Conclusões

- Modelo proposto não atribui ao servidor o estabelecimento do padrão de utilização dos recursos
- Padrão de utilização do recurso é determinado através de métricas simples com processamento leve
- Flexibilidade para determinação do padrão de utilização (heurísticas e parâmetro window)
- Esta forma de determinação da disponibilidade poderia ser usada também em:
  - Migração de processos
  - Fornecer garantias de execução

# Trabalhos Futuros

- Desenvolvimento dos componentes do servidor
- Realização de testes com o modelo de escalonamento
- Objetiva-se também a realização de testes em um ambiente de *Global Computing com grande quantidade de recursos (com simulação de usuário)*
  - Construção de protótipo com o XtremWeb
  - Testes Grid5000?

# Referências

- Anderson, D. P. (2004) “BOINC: A System for Public-Resource Computing and Storage”, Proceedings Fifth IEEE/ACM International Workshop on Grid Computing, Novembro 2004.
- Begole, J. B., Tang, J. C., Smith, R. B. and Yankelovich, N. (2002) “Work Rhythms: Analyzing Visualizations of Awareness Histories of Distributed Groups”. In Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW), 2002.
- Byun, E., Choi, S., Baik, M., Hwang, C. (2005) “Scheduling Scheme based on Dedication Rate in Volunteer Computing Environment”, Proceedings of the 4th International Symposium on Parallel and Distributed Computing (ISPDC'05).
- Choi, S., Baik, M. e Hwang, C., Gil, J., Yu, H. (2004) “Volunteer Availability based Fault Tolerant Scheduling Mechanism in Desktop Grid Computing Environment”, Proceedings of the Third IEEE International Symposium on Network Computing and Applications (NCA' 04).

# Referências

- Fedak, G., Germain, C., Néri, V. e Cappelo, F. (2001) “XtremWeb: A Generic Global Computing System”, Proceedings of the IEEE International Symposium on Cluster Computing and the Grid (CCGRID01).
- Mutka, M. W. (1991) “An Examination of Strategies for Estimating Capacity to Share Among Private Workstations”, Proceedings of the 1991 SIGSMALL/PC, Symposium on Small Systems, ACM Press.
- Sotrup, C. U. e Pedersen, J. G. (2005) “Developing Distributed Computing Solutions Combining Grid Computing and Public Computing”, <https://uimon.cern.ch/twiki/pub/LHCAtHome/LinksAndDocs/ChristianSoettrupBOINCthesis.pdf>, Setembro.
- Taufer, M., Teller, P., David P. A. e Brooks C. L. (2005) “Metrics for Effective Resource Management in Global Computing Environments”, e-Science and Grid Computing, First International Conference on Volume , Issue , 5-8 Dec.
- Estrada, T., Flores, D. A., Taufer, M., Teller, P. J., Kerstens, A. e Anderson D. P. (2006) “The Effectiveness of Threshold-Based Scheduling Policies in BOINC Projects”, Proceedings of the Second IEEE International Conference on e-Science and Grid Computing.