

# Primeiro Teste de Computação Grid (Duração: 2 horas)

Data: 3 de Abril de 2013

## 1 Parte teórica

- 1) Defina um RMS (Resource Management System). (máximo 5 linhas)
- 2) Qual é a diferença entre um RMS e um middleware utilizado por uma infraestrutura de grid para submissão e controle de jobs? (máximo 5 linhas)
- 3) Caracterize ambientes de grid e de cluster a nível de diferenças para suporte ao utilizador e diferenças de sistemas de gerenciamento (em outras palavras, quais são os requisitos de clusters e grids segundo o ponto de vista do utilizador e do sistema?) (máximo 5 linhas)
- 4) Assuma que lhe foi pedido para fazer a implementação de um middleware para submeter jobs para diferentes RMS. Estes RMS estão localizados em diferentes domínios administrativos. Faça a especificação técnica da sua implementação incluindo requisitos como: segurança, facilidade de uso, autorização e autenticação, monitoração e alocação de tarefas aos recursos.
- 5) No artigo "The Anatomy of the Grid", Foster, Kesselman and Tuecke argumentam que:  
*"current distributed computing approaches do not provide a general resource-sharing framework that addresses VO - Virtual Organizations - requirements."*  
Que requisitos são estes? Você acha que esta situação persiste nos dias de hoje?
- 6) Qual é a diferença entre um "PKI certificate", como requerido pelo gLite, e um "robot certificate"?

## 2 Parte prática

7) Dada a descrição de tarefa a seguir em JDL (Job Description Language, utilizada no middleware gLite), responda:

- (a) Qual é a função desta descrição?
- (b) Se utilizasse gLite para submeter esta tarefa, que passos e comandos utilizaria? (não omita os preâmbulos de utilização de uma infraestrutura de grid)
- (c) Internamente, o que o gLite faz com esta tarefa?

```
Executable = "yap";
InputSandbox = { "yap", "startup", "*.pl", "aleph.pl", "train.b",
"train.f", "train.n", "test.f", "test.n", "jdl.in.0" };
OutputSandbox = {"err.0", "theory-0.9-5.0", "trace-0.9-5.0"};
StdInput = "jdl.in.0";
StdOutput = "/dev/null";
StdError = "err.0";
```

8) Assuma que a descrição abaixo, escrita usando ClassAds (linguagem de descrição de jobs do Condor), descreve uma tarefa semelhante à da questão anterior (7). Qual é a diferença funcional (ou quais são as diferenças) entre esta descrição e a da questão (7)? Se existirem diferenças, poderia transformar a tarefa JDL na tarefa Condor ou vice-versa?

```
Universe = standard
Executable = yap
Initialdir = .
Log = log
Input = condor.in.$(Process)
Output = /dev/null
Error = /dev/null
Notification = error
Queue 2
```

9) Assuma que precisa correr milhares de simulações e que cada uma delas tem duração de no máximo 24h. Suponha também que não dispõe de recursos locais para correr estas simulações em tempo hábil. Qual seria sua melhor alternativa para correr estas simulações, assumindo que tem à sua disposição um ambiente de grid com o middleware gLite instalado? Escreva um ficheiro de descrição JDL que seja capaz de correr estas simulações em grid da forma mais eficiente possível (assuma que cada simulação recebe um ficheiro de parâmetros como entrada e produz um ficheiro de saída com tamanho maior do que 500 MBytes. É possível completar estas simulações?).

10) No primeiro trabalho desta disciplina, conseguiu submeter o job paralelo (MPI) ao seu RMS? Se não conseguiu, quais foram as razões?