

Guia de Utilização do SeARCH

Preâmbulo

As sessões de Prática-Laboratorial de Arquitectura de Computadores usam como recursos computacionais um conjunto dos nós de computação do *cluster* SeARCH. Este documento faz uma breve introdução à utilização desses recursos no contexto destas sessões.

Descrição

O SeARCH (*Services and Advanced Research Computing with HTC/HPC clusters*) disponibiliza uma gama alargada de recursos de computação, armazenamento de dados e comunicação de alto desempenho. Este *cluster* está fisicamente instalado no Departamento de Informática da Universidade do Minho e suporta projectos de Investigação e Desenvolvimento dos Centros de Investigação em Ciências da Computação, Matemática e Física.

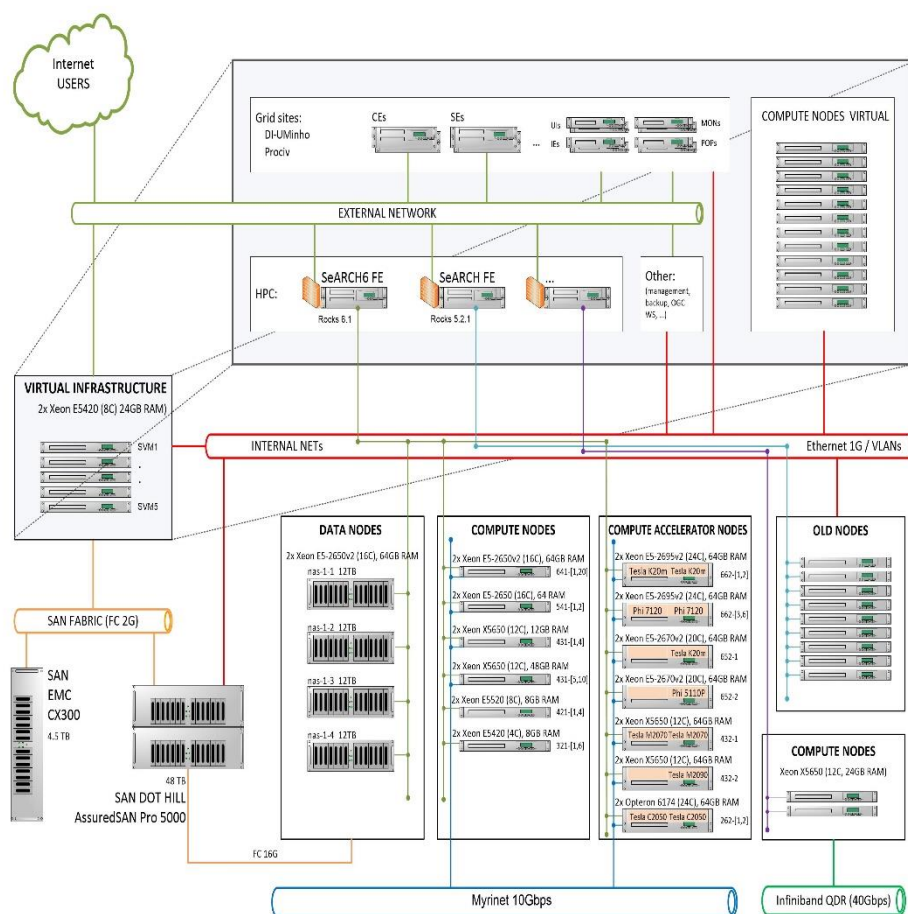


Ilustração 1 - Organização do SeARCH (2016)

A Ilustração 1 apresenta a organização actual do SeARCH – mais informação em <http://search.di.uminho.pt>. No contexto desta Unidade Curricular são relevantes:

- os utilizadores ligam-se ao SeARCH através de nós visíveis na Internet e designados por *frontend*; em particular, os alunos de Arquitectura de Computadores ligar-se-ão ao SeARCH através do search1.di.uminho.pt (apresentado na Ilustração 1 como SeARCH FE) – esta ligação é feita por *ssh* (*secure shell*);
- o *frontend* é a máquina através da qual o utilizador interage com o *cluster*; é neste nó que os ficheiros são editados e os programas compilados;

- quando um programa está pronto para execução deve ser submetido ao sistema de gestão de *jobs* e filas (*queues*) do SeARCH; este sistema designa-se por PBS e o comando que permite submeter *jobs* para execução é o `qsub` – os *jobs* são submetidos em filas de espera com determinadas características: em Arquitectura de Computadores é usada a *queue* `acomp`;
- o *job* será executado quando houver recursos disponíveis; o estado do *job* pode ser consultado usando o comando `qstat`; em Arquitectura de Computadores é usado apenas o nó designado por *compute nodes* 641-1; este nó está reservado apenas para os alunos e equipa docente de Arquitectura de Computadores (*queue* `acomp`);
- em cada momento é executado apenas um *job* na máquina 641-1; o acesso à máquina é portanto feito em regime de exclusividade, evitando degradações no desempenho originadas pela partilha de recursos entre vários *jobs*;
- para que um único *job* não possa bloquear a máquina, negando acesso aos restantes utilizadores o tempo máximo de execução de 1 *job* é de 60 segundos – se o *job* não terminar ao fim deste intervalo de tempo, então é terminado automaticamente pelo sistema;
- para que um único utilizador não possa monopolizar o acesso à máquina, cada utilizador só pode ter em espera na *queue* um único *job* – a submissão de outro *job* terá sempre que aguardar pela selecção para execução do *job* anterior do mesmo utilizador;
- quando a execução do *job* termina são criados na pasta do utilizador dois ficheiros, contendo respectivamente o `stdout` e o `stderr` do *job*.

Utilização

1. Ligue-se ao `search1.di.uminho.pt` usando a aplicação de `ssh` disponível na sua máquina (Sistemas Unix/Linux/MacOS: comando de linha `ssh` ; Windows `putty`/OpenSSH).
Autentique-se usando o nome de utilizador e a palavra chave que lhe foi enviada por correio electrónico;
2. Após a autenticação estará ligado ao *frontend* e tem acesso à sua pasta. Será aqui que editará ficheiros, criará os executáveis, submeterá *jobs* para execução e terá acesso aos resultados;
3. Os ficheiros necessários para cada módulo prático de Arquitectura serão disponibilizados na directoria `/share/acomp`; copie os ficheiros apropriados para a sua directoria, seguindo as instruções do respectivo guião;
4. Ao longo do semestre usaremos a versão 5.3.0 do compilador `gcc` e a versão 5.4.1 da biblioteca `PAPI`; estes módulos devem ser carregados explicitamente para compilar e para executar os programas – no entanto, as `script` e as `Makefile` disponibilizadas já incluem os comandos necessários para carregar estes módulos, pelo que não precisará de o fazer;
5. Para construir o executável escreva use o comando `make`;
6. Para submeter o *job* use o comando `qsub`; cada guião incluirá instruções sobre como usar o `qsub` – na primeira versão do programa `gemm` este necessita de 2 argumentos: o número de linhas da matriz e qual a versão do programa a usar; para uma matriz de 512x512 elementos e usando a versão 1 da função `gemm()` escreva:

```
qsub -N "nomeJob"-F "512 1" gemm.sh
```


O *job* será submetido e o `qsub` imprime o ID que identifica este job;
7. Pode verificar o estado de execução do job escrevendo

```
qstat -u <username>
```


onde `<username>` é o seu nome de utilizador;
8. Quando o *job* termina são criados 2 ficheiros, com os seguintes nomes:

```
nomeJob.o<jobID>
```

 - ficheiro de texto contendo o *output* do programa (`stdout`)

```
nomeJob.e<jobID>
```

 - ficheiro de texto com as mensagens de erro do programa (`stderr`)

Deve consultar estes ficheiros para verificar o resultado do *job*.