

Engenharia dos Sistemas de Computação

Objectivos de Aprendizagem

Em sistemas de computação de alto desempenho, muito frequentemente, os recursos disponibilizados excedem largamente os existentes nos sistemas pessoais. Todavia, as aplicações desenvolvidas para correr naqueles sistemas, para além de terem de tomar em consideração as características do equipamento, eventualmente heterogéneo, têm de tomar em consideração o ambiente geral de execução que poderá incluir, em cada momento, outras aplicações e utilizadores que irão concorrer no acesso e utilização dos mesmos recursos.

O que significa que uma aplicação desenvolvida, inicialmente, para correr numa plataforma que corresponde a um modelo abstracto de execução, fica sujeita a um conjunto alargado de condicionantes que podem ser absolutamente determinantes para o desempenho final da aplicação.

Cumulativamente, porque as aplicações têm normalmente, um tempo de vida muito superior ao do equipamento que além disso evolui muito rapidamente, criam-se facilmente ambientes cada vez mais heterogéneos que agravam as condições de execução das aplicações.

Avaliação Individual e em grupo

Com vista a explorar de forma extensiva os atuais ambientes de *clustering*, no desenvolvimento das propostas de trabalhos que seguem, os estudantes terão necessidade de explorar: ferramentas de administração de recursos e escalonamento de tarefas, tais como o PBS, ferramentas de desenvolvimento e utilitários para a compilação, edição de texto, criação/análise de perfis de execução, ou aplicações como o GNUplot, ou o GDB, etc.

Relatórios: Não está *a priori* definido nenhum formato especial para a elaboração do(s) relatório(s). Todavia sugere-se a utilização de formatos convencionais de produção de relatórios e artigos científicos (1 ou 2 colunas), com resumo, introdução, corpo principal (divisão em secções), uma secção de discussão dos resultados e eventualmente uma secção de trabalho futuro.

I - O ambiente de execução em *clustering*

Este exercício cria as condições para um primeiro contacto dos estudantes com as facilidades gerais de um ambiente de computação cooperativa - *cluster* -para compilar, submeter e executar aplicações e visualizar os resultados das execuções.

Em particular será usado o pacote **NAS Parallel Benchmarks (NPB)** afim de avaliar a performance relativa de programas. O pacote compreende um conjunto de testes de referência constituídos por *kernels* e simulações, escritos em linguagem C/Fortran, em três versões distintas: Sequencial, Memória Partilhada/Memória Distribuída.

NPB: <http://www.nas.nasa.gov/Software/NPB/>

Pretende-se que os estudantes, selecionem e executem alguns dos testes da série NBP no cluster SeARCH (<http://search.di.uminho.pt>), fazendo a comparação entre os resultados produzidos pelas diferentes versões disponibilizadas, nomeadamente, a partir da visualização gráfica dos respectivos resultados.

Especial importância deverá ser dada a repetição dos testes, nas diferentes classes de arquiteturas de nós existentes, usando: as versões sequenciais e as paralelas: memória distribuída e memória partilhada, Open-MPI e Open-MP, diferentes

compiladores (Intel e GNU), diferentes opções de compilação, diferentes tecnologias de comunicação e, ainda, diferentes dimensões (classes) de dados.

O estudo do impacto no desempenho dos testes deverá realizado através de medições precisas dos tempos de execução/ócio/bloqueio, da ocupação de memória, da comutação do tempo de E/S, e de outras métricas disponibilizada pelas ferramentas de monitorização).

As execuções deverão ser executadas em ambiente de processamento em lotes, recorrendo ao sistema PBS, sendo os resultados processados, recorrendo eventualmente a ferramentas de *scripting* e o *gnuplot*, de forma a produzir gráficos que servirão de base para a interpretação de resultados

II - Utilitários de Monitorização

Há uma vasta gama de programas, ferramentas e utilitários que podem ser usados para examinar o estado de um sistema de operação ou de uma aplicação. Este trabalho tem a vista a familiarização com alguns deles através da utilização dos seus parâmetros mais importantes e mais utilizados.

Sistema:

1. No exercício anterior o objectivo era obter os melhores valores para a duração e o desempenho das aplicações, modificando configurações de compiladores/compilação/memória e dos diferentes nós do Search.

Assim, usando uma seleção de testes NPB, sugere-se ir aumentando gradualmente a carga de computação para compreender a forma como é afectado o estado do sistema, em termos de utilização: dos cpus, da memória, dos discos, da rede etc.

Em particular, pretende-se exercitar comandos, tais como: *iostat*, *vmstat*, *mpstat*, *ps*, *top*, *lsof* e outros derivados, tais como *dstat*, para através dos respectivos relatórios, obter: estatísticas de utilização, estudos de eficiência e análises dos limites de utilização.

III - Armazenamento

Muitos sistemas de operações são afinados para terem um bom desempenho de E/S de dados em algumas das aplicações mais utilizadas, mas pode não funcionar bem para outras aplicações. Outras vezes, os padrões de acesso aos dados podem mudar, por exemplo passando de leituras/escritas sequenciais para leituras/escritas com padrões aleatórios, o que se pode traduzir em aumentos significativos dos tempos esperados para as operações nos sistemas de E/S.

IOZONE:

Neste exercício o objectivo é usar uma ferramenta de referência para fazer uma análise ampla dos diferentes tipos de sistema de discos existentes no Search apresentando o resultado do desempenho para uma multiplicidade de operações, incluindo entre outras: Ler/escrever, reler/reescrever, ler/escrever para trás, etc. Tal estudo deverá ser acompanhado pela apresentação das características físicas de cada um dos tipo de discos identificados (dica, usar o comando *dmesg*, ou o comando *udevadm*)

IV – Análise de traçados

O **strace** é uma ferramenta para a depuração de programas cujos traçados quando filtrados e analisados podem também ser usados para estudar os padrões de execução das aplicações.

2. Considere o comando `strace -T -ttt -o strace.out <aplicação>` usado sobre a aplicação *iozone*, numa configuração que contemple vários tamanhos de blocos e os seguintes modos `-i0 -i1 -i2 -i5`. Seguidamente use o utilitário *strace_analyzer* (e.g. `strace_analyzer_ng_0.09.pl --dat strace.out > st.log`) sobre os traçados. Para facilitar a utilização daquele comando considere a possibilidade de usar o comando auxiliar `refazStFd.py`. Ambos os comandos estão acessíveis na pasta `/share/apps/IOAPPS` do servidor `search6`

A partir das estatísticas geradas pretende-se:

- a) Obter um relatório de desempenho equivalente ao que é produzido diretamente pela aplicação *iozone* na configuração acima referida.
- b) Produzir gráficos que representem os padrões temporais de acesso aos dados, para as operações de leitura/escrita e busca (`lseek`)

Nota: Para os comandos envolvidos use as opções que considere mais relevantes, tal como: `-e trace=file` para reduzir a dimensão dos traçados.