

# Grid and Cloud

---

COMPUTAÇÃO GRID 2012/2013

**U.** PORTO

**FC** FACULDADE DE CIÊNCIAS  
UNIVERSIDADE DO PORTO

## 1. Introdução

Com o conceito de “Cloud” surgiu também a ideia de aproveitar estas infraestruturas como serviços para serem rentabilizadas para solução de necessidade de HPC (high performance computing).

No seguimento desta ideia foi feita a análise de dois artigos que tratam da comparação entre os tradicionais clusters privados e serviços de grid. Estes utilizados em geral pela comunidade científica. Os artigos revistos foram:

“A Quantitative Analysis of High Performance Computing with Amazon’s EC2 Infrastructure: The Death of the Local Cluster?” proposto em 2009 por, Zach Hill e Marty Humphrey.

“Combining Grid and Cloud Resources by Use of Middleware for SPMD Applications”, proposto em 2010 por, Brian Amedro, Françoise Baude, Fabrice Huet e Elton Mathias.

O primeiro proposto para revisão pela Professora Dra. Inês Castro Dutra, o segundo procurado pelas palavras chave “Grid”, “Cloud” e “HPC”, no motor de busca GoogleScholar.

## 2. Problema e motivação

O problema surge da comparação de performance entre Cloud e Grid no contexto de utilização para HPC, vantagens e desvantagens dos diferentes serviços e saber qual mais se adequa a uma utilização genérica deste tipo de problemas, debatendo também a questão de custos.

A ter em conta que a bibliografia escolhida não é a mais actual e como o serviço Cloud utilizado nos dois artigos para comparação é o Amazon EC2, uma pesquisa dos custos actuais deste serviço revelou uma diferença entre os preços praticados na altura das publicações e os praticados no presente, no entanto os custos de montagem de um cluster equiparável aos utilizado.

## 3. Discussão

A Comunidade científica pergunta-se sobre comparação de performance entre clusters de ponta e muito mais baratos de montar em laboratório se comparam com EC2?

Para além disso temos problemas de segurança e privacidade pelo que muitos cientistas não expõem o seu trabalho a clusters públicos.

A questão acima foi avaliada para a mais comum das aplicações científicas ou seja programas em MPI. Testes de benchmark mostram que tanto em velocidade de memória e performance MPI para as instâncias Amazon EC2, competem com os clusters locais. No entanto e relativamente a este ponto não são apresentados dados.

Sem surpresa e atentando na latência de comunicação, EC2 mostrou ser lento em mais de uma ordem de grandeza do que os clusters de HPC.

Os sistemas mais usados normalmente são 64 bits, foram ainda comparados dentro da EC2 vários tipos de sistemas que a Amazon oferece. Na avaliação não foi considerado o sistema de I/O (na minha opinião

é um ponto de falha na avaliação pois podemos querer executar tarefas ordenadas por DAG, que para a comunidade científica podem gerar um DAG enorme e a transferência de ficheiros tem um impacto muito grande no tempo de execução final). O foco foi no CPU e na carga de troca de mensagens a que a rede é sujeita.

Quanto aos benchmarks da virtualização, neste caso XEN, a própria XEN disponibiliza estudos, mas estes foram feitos em ambientes de cluster criados por eles próprios o que torna as medidas negligentes. Um projecto chamado "ElasticWulf", contem bibliotecas de MPI e linha de comandos da Amazon, o que permite rapidamente criar um cluster de simulação.

Para assemelhar mais o cluster ao da Amazon as máquinas tem os mesmos processadores Intel Xeon, visto que estes tem uma elevada performance, o que influencia mais nos tempos é o acesso à memória em cada nó e a performance da rede.

Para clusters dedicados a comparar foram usados 'Sunfire' e 'Camillus'.

Logo na primeira tabela podemos encontrar uma discrepância, as máquinas do cluster 'sunfire' usam uma arquitectura de memória anterior à DDR, o que não acontece nas simulações do 'Camillus' e dos tipos do EC2, no entanto os valores do cluster 'Camillus' conseguem ser melhores do que 3 dos 4 tipos de EC2 testados.

Para N Threads o autor considera que as máquinas EC2 fazem frente ao cluster 'Camillus', no entanto e mais uma vez 3 das 4 máquinas são ultrapassadas pelo 'Camillus', sendo a única capaz de produzir mais resultados a C1XLarge, cujo custo por hora é de 0.80\$ o que ao fim de um ano perfaz um total de 6912\$, o que faz com que a questão inicial dependa muito do tipo de aplicação e necessidade de utilização.

A ferramenta utilizada para medir benchmarks de MPI foi a IntelMPIBenchmarks V3.2, apesar de já estar disponível a versão 2 do MPI os autores decidiram fazer os testes usando a versão anterior assumindo que a diferença não seria significativa.

Para estes testes, não foi usado o cluster que melhores resultados obtivera anteriormente o que torna a comparação menos rica.

Como podemos verificar em todos os testes, tratando de latência ou largura de banda de memória, o cluster 'sunfire' supera todos os tipos de EC2, tendo especial atenção que as máquinas deste cluster ainda utilizam memória anterior à DDR, pelo que MPI é completamente desfavorável à utilização em EC2.

O propósito do segundo artigo é fazer uma avaliação comparativa entre cloud e clusters pessoais e apresentar uma solução de middleware para criar uma arquitectura híbrida entre cloud e grid computing para solução de necessidades de HPC.

É referido à partida que este tipo de solução surge da falta de performance de cloud para processamento de HPC, pelo que podemos ter como garantido que a performance de cloud é baixa comparativamente aos clusters tradicionais.

Mais uma vez são usados como referência para cloud os serviços da Amazon EC2, como referido, devido à sua interface configurável e simples.

As instâncias de EC2 usadas são as mesmas do estudo anterior mas as máquinas que as suportam sofreram melhorias da mesma maneira que o cluster utilizado no estudo é relativamente mais poderoso que os dois referidos anteriormente. Portanto assume-se que as condições dos dois testes podem ser equiparáveis dando assim origem a uma comparação mais justa entre os dois trabalhos.

Como podemos verificar o teste de troca de mensagens o cluster privado obtém resultados superiores a todos os testes nas instancias EC2, apenas sendo superado quando a troca de mensagens exige um envio superior a 900MB que é o limite da rede Ethernet do cluster privado, sendo a partir deste ponto a instância EC2 CC1XLARGE mais eficiente, pois as máquinas desta instancia estão ligadas por FastEthernet.

Para teste foram submetidos a todas as arquitecturas testes de performance denominados "Kernel EP", "Kernel CG", "Kernel MG" todos de classe C.

-Kernel EP:

Problema paralelo que quase não envolve comunicação, o foco deste teste é o poder computacional de cada nó, o resultado medido em instrução/unidade tempo (mflops) revela que a instância XLarge CC1 é a mais capaz, isto porque a sua arquitectura é a que dispõe de características mais avançadas, para o um cluster privado isto exigiria um investimento maior.

-Kernel CG:

Envolve um problema de cálculo de gradiente, para qual o foco é a comunicação, como havíamos verificado, para o cluster o delay de comunicação é muito menor, pelo que se destaca neste teste, mas mais uma vez, quando o tamanho da mensagem supera a capacidade da ligação a interface XLarge CC1 volta a ganhar destaque.

-Kernel MG:

Teste que foca poder de computação e comunicação, envolvendo transferências de grandes quantidades de dados e processamento dos mesmos, voltando mais uma vez a destacar-se o cluster, pois o delay de comunicação provoca atrasos muito significativos para os CPU's mesmo que estes sejam rápidos, novamente o problema do cluster local surge quando a troca de comunicações excede a capacidade das mesmas.

No geral podemos concluir destes testes que existe uma diferença fundamental para as duas arquitecturas, numa falta de poder de computação, na outra falta de poder de comunicação. Neste caso a falta de poder de comunicação afecta as duas soluções. Podemos afirmar que seria mais vantajoso um cluster privado, pois sendo este que sofre de falta de poder de comunicação, a aplicação de uma rede FastEthernet ao cluster não seria muito dispendiosa em relação ao ter que melhorar a qualidade do hardware. No entanto decidir qual o mais vantajoso continuar a depender da finalidade de utilização, pelo que falaremos à frente do middleware proposto para tornar estes dois serviços num serviço híbrido.

O middleware utilizado para criar a plataforma híbrida é o Java ProActive, para ser possível a integração com os recursos EC2 é necessária uma conta de Web Service da Amazon.

A ter em atenção que todos os nós do serviço da Amazon são passíveis de comunicação de entrada e saída enquanto que os do serviço grid utilizado (Grid500), são isolados do exterior. Para ser possível utilizar a comunicação multi-dominio do ProActive nesta configuração, o middleware comunica com o Gateway Grid500.

As configurações multi-dominio tem pelo menos dois níveis, no primeiro as aplicações, a correrem nas máquinas físicas ou virtuais no caso da Cloud, no segundo nível os componentes necessários para troca de mensagens multi-dominio. Devido à composição hierárquica, todos os nós estão logicamente ligados, mesmo que indirectamente e conseqüentemente as execuções MPI independentes são acopladas formando uma única aplicação paralela.