

CLAP: Uma Ferramenta para a Implantação de Aplicações de HPC nas Nuvens Computacionais

Otávio Oliveira Napoli¹, Edson Borin¹

¹Instituto de Computação – Universidade Estadual de Campinas (UNICAMP)
Campinas – SP – Brasil

otavio.napoli@gmail.com, borin@unicamp.br

Abstract. *With the huge increase in the use of cloud computing in several areas of research, new challenges for the high-performance computing community emerge. In this way, tools for automation of the entire workflow, including the deployment and execution of HPC applications on virtual machines and the retrieval of results from them are increasingly necessary. In this context, we propose CLAP (Cloud Application Platform), a tool designed to automate the creation of clusters in multiple computational clouds and the deployment of high-performance applications in them, in a simple, fast and efficient way, using current state-of-the-art technologies.*

Resumo. *Com o grande aumento do uso da computação em nuvem em diversas áreas de pesquisa, novos desafios para a comunidade de computação de alto desempenho emergem. Desta forma, ferramentas para automação de todo o fluxo de trabalho, incluindo a implantação e execução das aplicações em máquinas virtuais e a coleta de resultado das mesmas se mostram cada vez mais necessárias. Neste contexto, introduzimos o CLAP (Cloud Application Platform), uma ferramenta projetada para automatizar a criação de aglomerados em múltiplas nuvens computacionais e a implantação de aplicações de alto desempenho nos mesmos, de maneira simples, rápida e eficiente, utilizando tecnologias atuais do estado da arte.*

1. Introdução

A computação de alto desempenho, ou HPC¹, tem um papel muito importante em diversas aplicações científicas e de engenharia. No entanto, equipamentos de HPC são custosos e a aquisição deste tipo de equipamento pode ser proibitiva. Sendo assim, a nuvem computacional se tornou um recurso alternativo para tais aplicações em relação à aquisição de aglomerados de processamento, devido ao seu modelo de negócio, onde o usuário paga apenas pelo uso do sistema, sem custo de aquisição ou manutenção. Além disso, provedores de serviços de nuvem disponibilizam acesso a sua infraestrutura através da *Internet* e oferecem uma grande gama de opções de recursos computacionais, permitindo que a plataforma seja adaptada às necessidades das aplicações.

A automação da criação e manutenção da infra-estrutura, juntamente com a automação da configuração de máquinas para execução de aplicações na nuvem computacional torna-se cada vez mais necessária visto que tais operações, quando realizadas de

¹do inglês: *High Performance Computing*.

maneira manuais, podem ser tarefas demoradas e sujeita a erros, especialmente quando utilizando um grande numero de máquinas. Neste contexto, ferramentas como *elasticsearch*², Ansible³ e OpenStack⁴ tornam-se cada vez mais populares para solucionar tais problemas, simplificando fluxos de trabalhos complexos de maneiras eficiente.

O CLAP (*Cloud Application Platform*)⁵ é uma ferramenta de automação para criação e manutenção de aglomerados em múltiplos provedores de nuvem e fornece meios para configuração destes aglomerados por meio de um sistema de grupos, simplificando a realização de ações em múltiplas máquinas após o aglomerado ter sido devidamente configurado. O CLAP também fornece uma interface programática (API) na linguagem *Python*, que permite a extensão dos mecanismos existentes da ferramenta e a interação com as máquinas desde sua criação até sua contínua reconfiguração de forma dinâmica, adequado para trabalhos que precisam tomar decisões baseados no estado das instâncias ou de ações realizadas [Okita et al. 2018, Brunetta and Borin 2019].

2. Ferramentas de Automação para a Criação de Aglomerados Computacionais

Aplicações de HPC frequentemente fazem uso de múltiplas unidades de processamento, ou nós computacionais, para acelerar sua execução. Desta forma, ferramentas para a automação da criação e manutenção de aglomerados e ferramentas para a automação da implantação das aplicações nos mesmos se tornam cada vez mais necessárias. O **elasticsearch** é uma ferramenta de código aberto utilizada para automação da criação e manutenção de instancias (máquinas virtuais) em nuvens públicas e privadas. Sua arquitetura funciona de maneira simples através de arquivos de configuração definidos em um diretório conhecido que contém informações de como acessar um serviço de nuvem específico, incluindo informações de identificação de acesso e chaves necessárias.

Um fluxo de execução típico para a criação e configuração de um aglomerado para executar uma aplicação (*e.g.* escalonador de tarefas SLURM) consiste em três etapas. Primeiro, o usuário escreve um pequeno arquivo de configuração que descreve seu aglomerado e os grupos em que cada instancia deve pertencer. Os grupos designam os papéis que cada instância deve desempenhar (*e.g.*, grupo Hadoop *master* e grupo Hadoop *worker*). Então, a ferramenta se conecta ao provedor utilizando a configuração informada, inicia as máquinas virtuais e por fim, as instâncias são configuradas apropriadamente para pertencer aos grupos em questão, utilizando uma ferramenta de automação de configuração conhecida como Ansible.

Tais ferramentas de automação de configurações são especialmente essenciais para o gerenciamento de aglomerados grandes, visto que: (I) a realização de tarefas manuais é cansativa e sujeita a erros; (II) para ambientes heterogêneos pode ser complicado manter as máquinas devidamente configuradas e; (III) em processos manuais, algumas máquinas podem ficar fora de sincronia ou até mesmo configuradas incorretamente.

O Ansible é uma ferramenta de automação de configuração de código aberto utilizada para a implantação de aplicações, orquestração de serviços e provisionamento de

²<https://github.com/elasticsearch/elasticsearch>

³<https://www.ansible.com/>

⁴<https://www.openstack.org/>

⁵<https://github.com/lmcad-unicamp/CLAP>

recursos utilizando livros de receitas em arquivos no formato YAML. Para tanto, a ferramenta não necessita de nenhum pacote adicional para seu funcionamento e também não requer nenhuma habilidade especial de codificação, o que normalmente permite uma rápida curva de aprendizado em relação a outras ferramentas de automação como o Puppet⁶, o Chef⁷ e o SaltStack⁸, onde conhecimentos de linguagens como *Ruby* ou *Python* são necessárias para sua operação e/ou extensão. Além disso, a ferramenta dispensa o uso de agentes instalados nos clientes para sua execução.

Desta forma, visando a modularidade, o Ansible pode ser segmentado em quatro componentes, a saber: (I) um inventário, que diz quais máquinas devem ser gerenciadas; (II) os módulos, que provê meios de controlar recursos locais ou remotos e pode abranger uma variedade de funções, incluindo reiniciar as máquinas, copiar arquivos, entre outras; (III) as tarefas, que são responsáveis por invocar os módulos com parâmetros específicos e; (IV) o livro de receitas, no formato YAML que descreve quais tarefas cada conjunto de máquinas de um inventário deve executar.

Para configuração de aglomerados, o elasticcluster gerencia o inventário que será utilizado pelo Ansible, o que é muito útil quando os inventários são dinâmicos (*e.g.* instâncias com mudanças de IP). Entretanto, algumas desvantagens com a ferramenta incluem: a) necessidade que o usuário tenha conhecimentos com livros de receita em Ansible para realizar ações depois que as instâncias já estejam configuradas; b) não foi desenvolvida para possuir uma API para trabalhar com a ferramenta e; c) não possui suporte para a criação de aglomerados mistos (com máquinas em diferentes zonas de disponibilidade).

3. CLAP – Cloud Application Platform

O CLAP é uma ferramenta de automação para o criação e configuração de instâncias em nuvens. Similarmente ao elasticcluster, o CLAP baseia-se em arquivos de configurações para criação de instâncias e composição de aglomerados. Entretanto, o CLAP permite criar e configurar instancias de maneira individual e atribuir rótulos as mesmas. Assim, um aglomerado consiste em um conjunto de instâncias com um mesmo rótulo, permitindo a criação de aglomerados mistos. Tal característica pode ser desejável em trabalhos que necessitam operar sobre instâncias de diferentes provedores simultaneamente.

O CLAP também permite a configuração automatizada das instâncias utilizando Ansible, além de permitir que instâncias sejam adicionadas a grupos e que ações sejam executadas em grupo. Quando uma instância é adicionada a um grupo, diz-se que essa instância pode desempenhar um papel específico (*e.g.* instâncias adicionadas ao grupo Hadoop, conseguem executar aplicações Hadoop). Para tanto, cada grupo define um conjunto de ações associadas a ele (*e.g.* iniciar uma aplicação, verificar seu estado, etc.), que são os livros de receitas que devem ser executados quando uma determinada ação é invocada pela linha de comando.

O funcionamento para a criação de aglomerados, baseado em arquivos de configurações é ilustrado na Figura 1. O usuário escreve os arquivos de configuração que descreve as instâncias e os aglomerados (1). Depois (2), máquinas virtuais são instanciadas e então (3) rótulos são adicionados as instâncias, determinando aglomerados. Logo

⁶<https://puppet.com/>

⁷<https://www.chef.io/>

⁸<https://www.saltstack.com/>

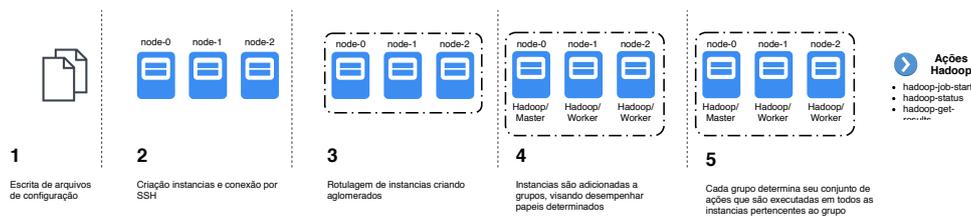


Figura 1. Funcionamento utilizado pelo CLAP para a criação de aglomerados

(4), as instâncias de cada aglomerado são adicionadas aos seus respectivos grupos e, por fim (5), ações definidas por cada grupo podem ser tomadas nas instâncias que pertencem a aquele grupo.

Por fim, o CLAP fornece uma API simples na linguagem *Python* que permite o gerenciamento de instâncias e a tomada de ações em grupo. O uso da API é opcional, mas permite a extensão dos comandos providos pela ferramenta e a realização de trabalhos programáticos sobre as instâncias, isto é, trabalhos que precisam tomar decisões baseados no estado das instâncias ou de ações realizadas. Por exemplo, trabalhos recentes [Brunetta and Borin 2019, Tavares et al. 2019] utilizam técnicas de execução parcial para determinar quais instâncias melhor se adequam à execução de aplicações MPI, visando obter um menor custo monetário final. Neste, um monitoramento periódico seguido de uma análise de custo é realizada dinamicamente e então intervenções são realizadas para otimizar o sistema (*e.g.* desligando-se máquinas virtuais).

4. Considerações Finais

O CLAP é uma ferramenta de automação de infra-estrutura e configuração visando prover uma interface simples, rápida e eficiente para o levantamento de aglomerados em múltiplos provedores de nuvem computacional e implantação de aplicações de HPC nos mesmos. Para tanto, o CLAP se baseia em ferramentas atuais para realização destas operações provendo a possibilidade de criação de aglomerados mistos, maneiras simplificadas para tomada de ações em grupos de máquinas diversos e uma API na linguagem *Python* para trabalhar com as máquinas e estender a ferramenta.

Os autores agradecem a FAPESP (CCES 13/08293-7), o CNPq (140653/2017-1) e a Petrobras pelo apoio financeiro e o Laboratório Multidisciplinar de Computação de Alto Desempenho do Instituto de Computação da Unicamp pelo suporte computacional.

Referências

- Brunetta, J. R. and Borin, E. (2019). Selecting efficient cloud resources for hpc workloads. In *Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing*, pages 155–164.
- Okita, N., Coimbra, T., Rodamilans, C., Tygel, M., and Borin, E. (2018). Using spits to optimize the cost of high-performance geophysics processing on the cloud. In *First EAGE Workshop on High Performance Computing for Upstream in Latin America*, volume 2018, pages 1–5. European Association of Geoscientists & Engineers.
- Tavares, W., Reis, L., Brunetta, J., and Borin, E. (2019). Aplicação da técnica paramount iteration nas aplicações blast e dnn-rom na nuvem computacional. In *Anais do XX Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 228–239. SBC.