A Scalable, Commodity Data Center Network Architecture

Grupo 1
Francisco Caeiro, **47823**
Bruno Andrade, **47829**
António Estriga, **47839**

Qual é o problema que os autores tentam resolver?

Os *data centers* podem conter dezenas de milhares de computadores e exigir uma largura de banda agregada significativa. Para construir estes *clusters* de larga-escala existem duas opções: usando *hardware* e protocolos de comunicação especializados ou usar *commodity* Ethernet *switches* e *routers* para conectar as várias máquinas do cluster. A primeira opção permite-nos escalar para uma grande quantidade de nós e com grande banda larga, mas não é compatível para aplicações TCP/IP nem permite a utilização de commodity hardware, sendo assim mais caro. A segunda opção, apesar de mais barata, não escala bem a largura de banda agregada com o tamanho do cluster, podendo ser possível contornar este problema com soluções non-commodity, originando custos não lineares.

O que os investigadores deste artigo pretendem é alcançar uma **largura de banda de intercomunicação escalável**, onde seja possível um host arbitrário comunicar com qualquer outro na mesma rede à largura de banda disponível total, uma **economia de escala**, onde switches Ethernet baratos se tornem a base de redes de data center de larga escala, e **compatibilidade regressiva**, onde o sistema permite aos hosts correr Ethernet e IP, sem sofrerem modificações.

Este problema é relevante?

Na altura deste artigo, o principal *bottleneck* na escalabilidade dos cluster de larga escala é a largura de banda na comunicação inter-nós. Os investigadores decidiram então realizar um estudo para identificar quais são as melhores práticas, por volta de 2008, das redes de data center. Perceberam que a topologia usada era no formato de *tree* com 2 ou 3 níveis e os switches nas *leaves* da árvore têm portas GigE (48-288) juntamente com *uplinks* 10 GigE para uma ou mais camadas da rede, enquanto que nos níveis mais altos da topologia são usados switches com portas 10 GigE (32-128). É usado *oversubscription* para diminuir o custo total do sistema, ou seja, conectam-se vários hosts à mesma porta do switch, diminuindo assim a largura de banda agregada.

Para estudar o custo de "construção" de um cluster, assumiram que o custo de um switch das leaves é de $7.000 e que os restantes switches têm um custo de $700.000, sem contar com a cablagem necessária. Perceberam que o hardware necessário para oferecer a uma rede com 20.000 hosts uma largura de banda de 1 Gbps seria necessário 37 milhões de dólares. Ao aumentar o oversubstiption, o custo diminui mas, é usando a arquitetura explicada neste artigo, o custo é menor ($8,64M) e, como veremos à frente, o desempenho é igual ou melhor.

Qual é a sua solução? Que novas técnicas foram usadas?

Os investigadores decidiram, estudando uma topologia desenhada por Charles Clos há mais de 50 anos para as redes de telefone, adotar uma instância especial, chamando-lhe *fat-tree*. Tal como na rede de Clos, a fat-tree permite alcançar altos de níveis de largura de banda mas usando pequenos commodity switches. A arquitetura desenhada, *K-ary fat-tree*, é um topologia de três níveis: edge, aggregation e core. Cada pod consiste em $(k/2)^2$ servidores e 2 camadas de k/2 switches. Cada edge switch conecta-se a k/2 servidores e a k/2 aggregation switches. Cada aggregation switch conecta-se a k/2 edge switches e k/2 edge switches. Existem $(k/2)^2$ core switches onde cada um se conecta a k pods.

Esta arquitetura permite largura de banda idêntica para cada bisseção e cada camada tem a mesma largura de banda agregada. Um dos fatores decisivos para a acessibilidade desta arquitetura é o possível, e recomendado, uso de switches baratos; apesar de oferecerem capacidade uniforme (1 GigE), cada porta suporta a mesma velocidade de ligação que o end host e todos os dispositivos podem transmitir pacotes à total velocidade de ligação se forem distribuídos uniformemente pelos caminhos disponíveis. Oferece ainda uma grande escalabilidade, pois uma fat-tree com k-port switches geralmente suporta $k^3/4$ hosts.

Para evitar a concentração de tráfego e, diretamente, o mau uso dos paths existentes pelos protocolos de routing, os investigadores implementaram nos switches tabelas de routing de dois níveis. Estas tabelas espalham o tráfego baseando-se nos *low-order bits* dos endereços IP de destino: o primeiro nível é um lookup do prefixo, usado para route down (para os servidores). enquanto que o segundo nível é um lookup do sufixo, usado para route up (para os switches core). Este último nível permite manter o *packet ordering* utilizando as mesmas portas para flows idênticos.

Foi também implementado um esquema especial de endereçamento que, apesar de ser um pouco desperdício de espaço de endereçamento, simplifica as tabelas de routing e escala até 4,2M de hosts.

Foram ainda implementadas duas técnicas opcionais para routing dinâmico: flow classification, de modo a evitar a congestão local e assegurando uma distribuição justa do tráfego, distribuindo-o à base per-flow em vez de per-host; e flow scheduling, prevenindo flows de longa duração de partilhar os mesmos caminhos e designando-os para outros, eliminando a congestão global. Definiram ainda técnicas de tolerância a falhas, devido à grande redundância dos caminhos disponíveis entre cada par de host.

Na avaliação realizada, onde foram estudados 5 maneiras de comunicação entre hosts, perceberam que a arquitetura desenhada origina sempre percentagens de largura de banda de bisseção ideal melhores que as atuais e, ainda, que as técnicas opcionais de routing apenas podem a melhorar. Mostram ainda que o uso de commodity switches resulta em menos consumo de energia e desperdício de calor.

Como é que se destaca de trabalhos anteriores?

Foi desenhada uma arquitetura onde é possível usar clusters de commodity switches sem perder desempenho na rede. Com isto, os investigadores ambicionam que aconteça o mesmo que aconteceu com os SMPs ou os MPPs, promovendo menos gastos e melhor desempenho nas soluções de alto-nível.

Quais são os pontos mais fortes deste artigo? E os seus pontos fracos?

Este artigo está muito bem escrito, sem definições complicadas. Sem dúvida que promoveu um ainda maior uso de commodity switches, reduzindo o custo de data centers. Sempre que o custo de algo na tecnologia diminui, a ciência tende a crescer ainda mais rápido, e isso só traz vantagens.

Como seria uma extensão deste trabalho?

Uma extensão deste trabalho poderia ser a avaliação da implementação do sistema num grande data center, para que, não usando multiplexagem, pudesse ser testado como é realizado no mundo real. Apesar dos dados teóricos e as experiências realizadas apresentarem pontos positivos para este sistema, não temos garantias totais até a sua implementação para produção.