

Comunicação em Datacenters

Gerência de Redes

Leandro Souza da Silva
Luís Felipe Mattos

IC - Unicamp

06 de Dezembro de 2016



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



- Com o crescimento da computação em nuvem, os datacenters passaram a receber funções novas.
- Certas aplicações necessitam de certos requisitos:
 - ▶ Escalabilidade
 - ▶ Tolerância a Falhas
 - ▶ Latência
 - ▶ Capacidade da Rede
 - ▶ Virtualização



Sumário

- 1 Introdução
- 2 **Motivação**
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



Motivação

O consumo de dados pelos usuários está crescendo exponencialmente a cada ano.

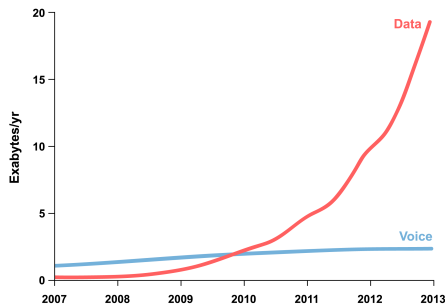


Figure: Consumo de dados e voz



Motivação

Por causa disso, o número de servidores em Data Centers deve crescer exponencialmente para acompanhar a demanda, o que traz dificuldades em desenvolver redes eficientes e de baixo custo.

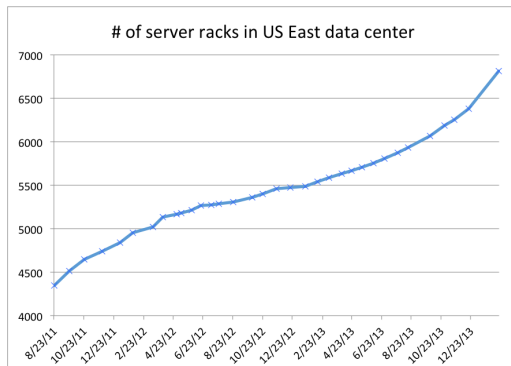


Figure: Número de servidores racks



Disponibilidade de dados e segurança se tornaram aplicações críticas.



Motivação

Por outro lado, a criação de novas tecnologias faz com que o custo dos componentes seja cada vez menor.

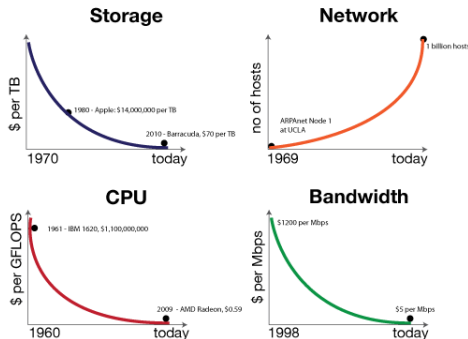


Figure: Custo de tecnologias



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias**
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias**
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



Topologias Tradicionais

- Baseadas em Árvores:

- ▶ CLOS
- ▶ Basic Tree
- ▶ Fat-Tree
- ▶ VL2

- Recursivas:

- ▶ Dcell
- ▶ Bcube
- ▶ FiConn
- ▶ FlatNet
- ▶ SprintNet



Topologias Tradicionais

Topologias baseadas em árvores



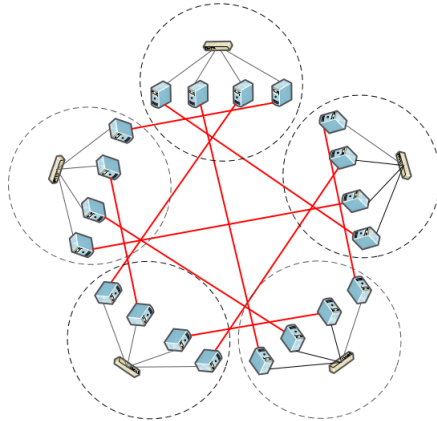
Topologias Tradicionais

Topologias recursivas



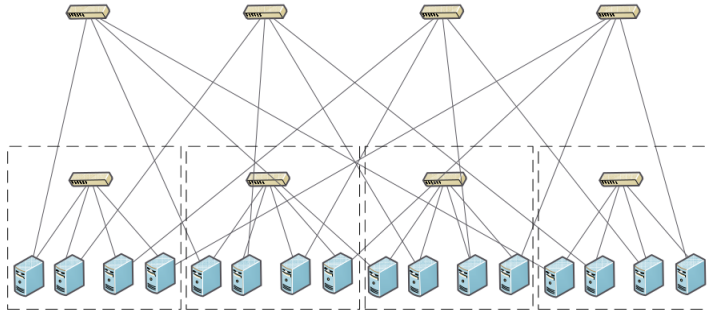
Topologias Recursivas: Dcell

Baseada em células interligadas entre servidores



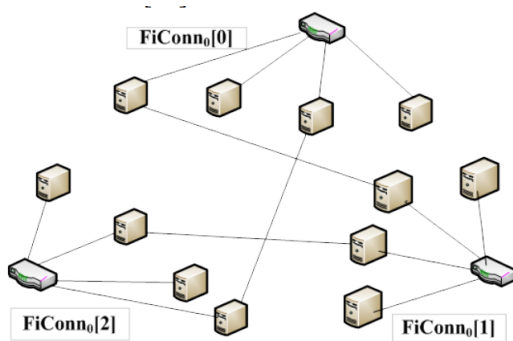
Topologias Recursivas: Bcube

Baseada em células interligadas entre switches



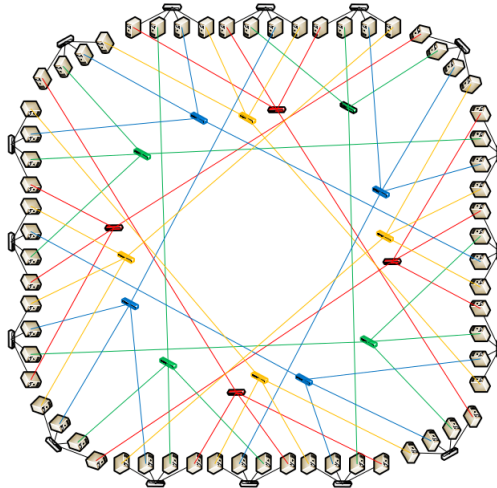
Topologias Recursivas: FiConn

Semelhante à Dcell, mas o grau de cada célula é sempre 2



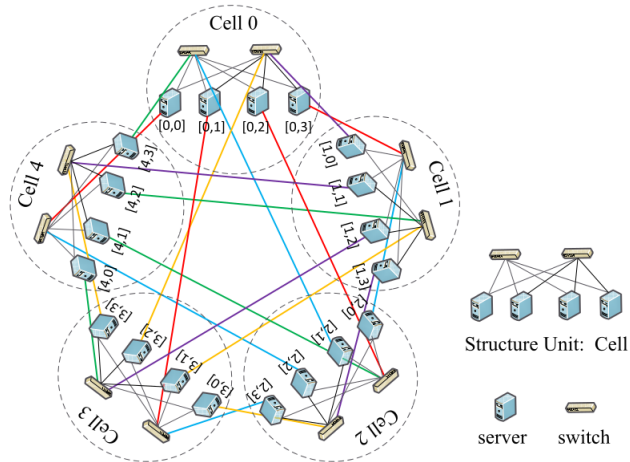
Topologias Recursivas: FlatNet

Semelhante ao BCube, porém é mais escalável



Topologias Recursivas: SprintNet

Semelhante à DCell, porém as células são compostas por 4 servidores e 2 switches de 6 portas



Comparação

	Fat Tree (3 layers)	VL2 (3 layers)	DCell (2 layers)	BCube (2 layers)	FlatNet (2 layers)	SprintNet (2 layers)
Servers Number	$\frac{n^3}{4}$	$\frac{(n-2)n^2}{4}$	$n(n+1)$	n^2	n^3	$(\frac{c}{c+1})^2 n^2 + \frac{c}{c+1} n$
Links Number	$\frac{3n^3}{4}$	$\frac{(n+2)n^2}{4}$	$\frac{3n(n+1)}{2}$	$2n^2$	$2n^3$	$\frac{c^2 n^2}{c+1} + cn$
per Server	3	$\frac{n+2}{n-2}$	$\frac{3}{2}$	2	2	≥ 2
Switches Number	$\frac{5n^2}{4}$	$\frac{n^2}{4} + \frac{3n}{2}$	$n+1$	$2n$	$2n^2$	$\frac{c^2}{c+1} n + c$
per Server	$\frac{5}{n}$	$\frac{n+6}{n^2-2n}$	$\frac{1}{n}$	$\frac{2}{n}$	$\frac{2}{n}$	$\frac{c+1}{n}$
Bisection Bandwidth	$\frac{n^3}{8}$	$\frac{n^2}{4}$	$\frac{n^2}{4} + \frac{n}{2}$	$\frac{n^2}{2}$	$\frac{n^3}{4}$	$\frac{c^2 n^2}{2(c+1)^2} + cn$
per Server	$\frac{1}{2}$	$\frac{1}{n-2}$	$\approx \frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{2} + \frac{(2c+1)(c+1)}{2(cn+c+1)}$
Network Diameter	6	6	5	4	8	4



Sumário

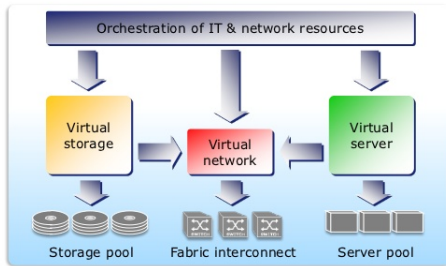
- 1 Introdução
- 2 Motivação
- 3 **Topologias**
 - Tradicionais
 - **SDN**
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



SDN em Datacenters

Com o avanço do SDN, a ideia mais básica é definir servidores virtualizados e criar uma rede virtualizada

SDN Inside The Data Center



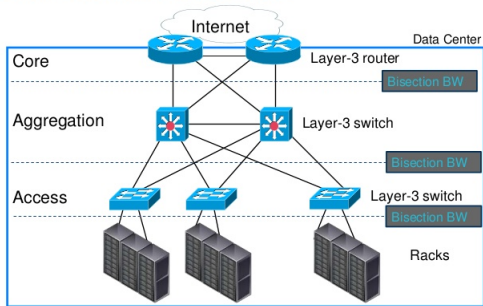
SDN adds missing piece to the virtualization puzzle: Network virtualization.



SDN em Datacenters

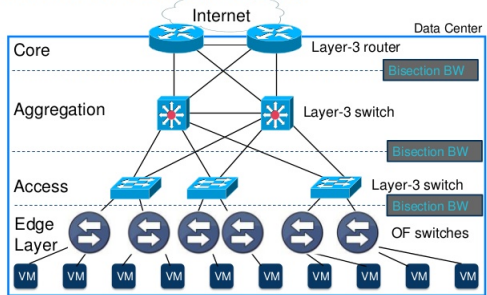
Esta técnica já é utilizada atualmente (PayPal por exemplo)

DATACENTER ARCHITECTURE



PayPal™

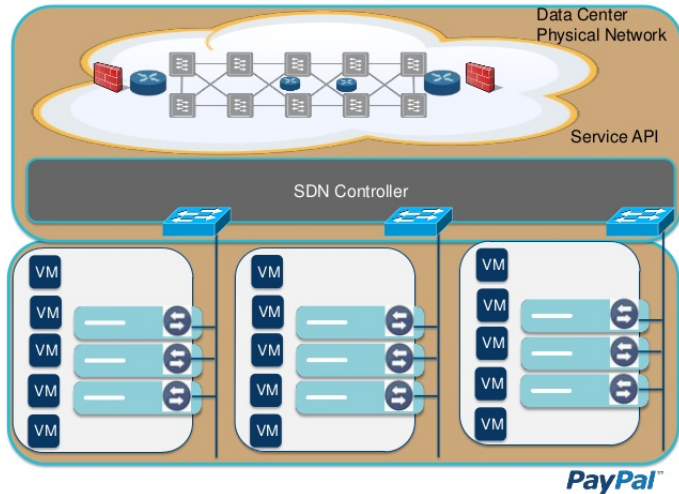
DATACENTER ARCHITECTURE WITH OPENFLOW VSITCHES



PayPal™



SDN em Datacenters



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos**
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



Protocolos



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



- Deadline-Agnostic:

- ▶ *DCTCP*
- ▶ *MPTCP*
- ▶ *ICTCP*

- Deadline-Aware:

- ▶ D^3
- ▶ D^2TCP
- ▶ *DeTail*
- ▶ *PDQ*



Protocolos Deadline-Agnostic



Deadline-Aware: D^3

- Recebe a informação do tamanho do fluxo e o deadline
- Algoritmo guloso para tentar cumprir o máximo de deadlines possíveis
- Roteadores tentam alocar uma taxa adequada para cada fluxo
- Emissor envia dados com a mínima taxa alocada no próximo RTT
- A fonte periodicamente requisita uma nova taxa baseada no deadline e o tamanho do fluxo restante



Deadline-Aware: D^3

- Switches precisam de modificações para lidar com as requisições
- Não é compatível com o TCP tradicional, por causa da alocação de largura banda baseado em prioridades sem a informação do deadline no header
- Alocação com algoritmo guloso pode alocar largura de banda para fluxos com deadlines distantes ao invés de deadlines próximos, o que pode causar maior perda de deadlines
- Requisições constantes de taxas tem um overhead



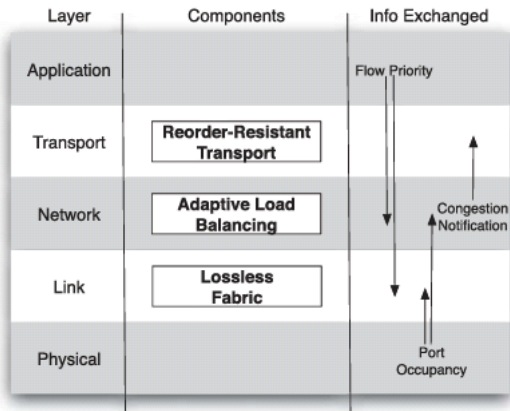
Deadline-Aware: D^2TCP

- Baseado no DCTCP, mas leva o deadline em consideração para redimensionar a janela de congestionamento
- Se a maioria dos deadlines são próximos, ainda pode ocorrer congestionamento
- Se a maioria dos deadlines são distantes, ocorrerá a sub-utilização da rede e baixa taxa de transferência
- Se todos os deadlines são próximos, estão competindo pela largura de banda e nenhum fluxo é adiado, todos os deadlines não serão cumpridos. Caso um fluxo seja adiado, todos os outros podem ser cumpridos.
- O problema é saber qual fluxo sacrificar para satisfazer o máximo de deadlines possíveis



Deadline-Aware: *DeTail*

- Fluxos são associados com prioridades e os switches usam filas de prioridades nas portas de saída e de entrada
- Cada camada da abstração da rede tem uma função



Enlace:

- Controle de fluxo “hop-by-hop”
- Tenta amenizar o bloqueio de “head-of-line” (HOL)
- Recebe informações da camada de rede em relação ao balanço de carga adaptativo
- Recebe informações da camada de transporte sobre o estado do ECN



Rede:

- Balanço de carga adaptativo baseado em pacotes de acordo com o nível de congestionamento
- Pode transmitir pacotes por caminhos que estão com pouca carga



Rede:

- Usa o ECN para marcar fluxos de baixa prioridade quando os bytes transmitidos para o destino ultrapassam um certo limiar
- Previne congestionamento persistente



Deadline-Aware: *DeTail*

Aplicação:

- Seleciona as prioridades de cada fluxo baseado na sensibilidade de latência



- Semelhante ao D^3 , mas ao contrário do D^3 , escalona taxas de acordo com a criticalidade dos fluxos ao invés da política “first-come first-serve”
- Duas políticas de alocação, implementadas de forma totalmente distribuída:
 - ▶ EDF (Earliest Deadline First)
 - ▶ SJF (Shortest Job First)



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências**
- 6 Conclusão
- 7 Pergunta



Ponha aqui seu texto



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão**
- 7 Pergunta



Ponha aqui seu texto



Sumário

- 1 Introdução
- 2 Motivação
- 3 Topologias
 - Tradicionais
 - SDN
- 4 Protocolos
 - Roteamento
 - Comunicação
- 5 Tendências
- 6 Conclusão
- 7 Pergunta



Ponha aqui seu texto

