Scaling Shopify’s multi-tenant architecture across multiple datacenters

FLORIAN WEINGARTEN
flo@shopify.com
@fw1729
Evolution of our platform

• ~2004: Snowdevil (single-tenant)
• ~2005: Shopify (multi-tenant)
• 2005-2012: Platform grows, flash sales, ...
• 2013/2014: Database isolation
• 2015: Backup datacenter for disaster recovery
• 2016: Multi-DC podding
FLASH SALES
MAKING MILLIONS WITHIN MINUTES
Get my absolute favorite shade Exposed right now on KylieCosmetics.com
“The Flash Sale Problem”

• Unpredictable. Not scheduled. No notice in advance.

• **Compared to our regular baseline, we *always* need to be massively over-provisioned.**

• Provisioning resources on demand is way too slow.

• Flash sales come and go within minutes.
MULTI-TENANT ARCHITECTURES
## Nothing vs. everything

<table>
<thead>
<tr>
<th>Share nothing</th>
<th>?</th>
<th>Share everything</th>
</tr>
</thead>
<tbody>
<tr>
<td>Little capacity</td>
<td></td>
<td>Huge capacity</td>
</tr>
<tr>
<td>Bad utilization</td>
<td></td>
<td>Great utilization</td>
</tr>
<tr>
<td>Flash sale problem</td>
<td></td>
<td>Great for flash sales</td>
</tr>
<tr>
<td>Crazy expensive</td>
<td></td>
<td>Cheap</td>
</tr>
<tr>
<td>Full isolation and resiliency</td>
<td></td>
<td>No isolation or resiliency</td>
</tr>
<tr>
<td>Horizontal scale is easy</td>
<td></td>
<td>Horizontal scale can be hard</td>
</tr>
</tbody>
</table>
“Shared everything” is not good enough!
Spectrum of multi-tenant architectures

- Share nothing
  - 2004
  - More isolation
  - Less utilization

- Share everything
  - 2005-2012
  - Cheaper
  - More capacity
Spectrum of multi-tenant architectures

Share nothing

2004

Share everything

2005-2012

More isolation
Less utilization

Cheaper
More capacity
Spectrum of multi-tenant architectures

- Share nothing (2004)
  - More isolation
  - Less utilization

- Share everything (2005-2012)
  - Cheaper
  - More capacity
Spectrum of multi-tenant architectures

Share nothing

Share everything

2004

More isolation
Less utilization

2005-2012

Cheaper
More capacity
Spectrum of multi-tenant architectures

Share nothing

2004

Share everything

2005-2012

More isolation
Less utilization

Our sweet spot!

Cheaper
More capacity
Shared everything

WEB 1 ... WEB n  JOB 1 ... JOB n

DB

Big, expensive, SPOF
Database isolation
Why?

Maintenance

Redundancy and disaster recovery
Active datacenter
All traffic goes here

Backup datacenter
All databases are read-only
Replication

Active datacenter
All traffic goes here

Backup datacenter
All databases are read-only
Active datacenter
All traffic goes here

Backup datacenter
All databases are read-only
Active datacenter
All traffic goes here

Backup datacenter
All databases are read-only

Failover
Replication
Replication
How we used to do failovers
How we do failovers now

$ bin/dc-failover
Active datacenter

Passive backup datacenter
(Partially) active datacenter 1

(Partially) active datacenter 2
Podding

Pod 1
with dedicated capacity

WEB 1  WEB 2  ...  WEB n

DB 1

Pod 2
with dedicated capacity

WEB 1  WEB 2  ...  WEB n

DB 2

Pod n
(active in another DC)

WEB 1  WEB 2  ...  WEB n

DB n (read-only)
How to route requests to the right pod?
• **Sorting Hat**: Lua application that runs in our nginx load balancer.

• **MySQL**: `domain=bobs-shop.com → pod_id=5`

• `nginx.balancer`: API for defining dynamic upstream balancers.

• Other cool stuff: Kafka logger, edge caching, throttling, SSL certs from MySQL, ...
FLOATING CAPACITY

ISOLATION VS. UTILIZATION
Podding

Pod 1 with dedicated capacity

Pod 2 with dedicated capacity

Pod n (active in another DC)

WEB 1 → DB 1
WEB 2 → DB 1...
WEB n → DB 1

WEB 1 → DB 2
WEB 2 → DB 2...
WEB n → DB 2

WEB 1
WEB 2
WEB n

DB n (read-only)
Pods with floating capacity

- **Pod 1** with dedicated capacity
  - WEB 1
  - WEB 2
  - ... WEB n
  - DB 1
- **Pod 2** with dedicated capacity
  - WEB 1
  - WEB 2
  - ... WEB n
  - DB 2
- **Pod n** (active in another DC)
  - DB n (read-only)

- "Floating" capacity
  - WEB 1
  - WEB 2
  - ... WEB n
## Multi-tenant architectures

<table>
<thead>
<tr>
<th>Share nothing</th>
<th>?</th>
<th>Share everything</th>
</tr>
</thead>
<tbody>
<tr>
<td>Little capacity</td>
<td></td>
<td>Huge capacity</td>
</tr>
<tr>
<td>Bad utilization</td>
<td></td>
<td>Great utilization</td>
</tr>
<tr>
<td>Flash sale problem</td>
<td></td>
<td>Great for flash sales</td>
</tr>
<tr>
<td>Crazy expensive</td>
<td></td>
<td>Cheap</td>
</tr>
<tr>
<td>Full isolation</td>
<td></td>
<td>No isolation</td>
</tr>
<tr>
<td>Horizontal scale is easy</td>
<td></td>
<td>Horizontal scale can be hard</td>
</tr>
</tbody>
</table>
## Multi-tenant architectures

<table>
<thead>
<tr>
<th>Share nothing</th>
<th>Pods with floating capacity</th>
<th>Share everything</th>
</tr>
</thead>
<tbody>
<tr>
<td>Little capacity</td>
<td>Good capacity</td>
<td>Huge capacity</td>
</tr>
<tr>
<td>Bad utilization</td>
<td>Good utilization</td>
<td>Great utilization</td>
</tr>
<tr>
<td>Flash sale problem</td>
<td>Great for flash sales</td>
<td>Great for flash sales</td>
</tr>
<tr>
<td>Crazy expensive</td>
<td>Cheap</td>
<td>Cheap</td>
</tr>
<tr>
<td>Full isolation</td>
<td>Isolated pods</td>
<td>No isolation</td>
</tr>
<tr>
<td>Horizontal scale is easy</td>
<td>Horizontal scale is easy</td>
<td>Horizontal scale can be hard</td>
</tr>
</tbody>
</table>
MULTI-DC ROUTING

POINTS OF PRESENCE AND HIGH AVAILABILITY
Datacenter failover
Datacenter failover
Datacenter failover

Internet

Withdraw BGP

Sorting Hat

POD 1 ... POD n

Announce BGP

Sorting Hat

POD 1 ... POD n
Datacenter failover

Internet

Withdraw BGP

Sorting Hat

POD 1  ...  POD n

Announce BGP

Sorting Hat

POD 1  ...  POD n
Scaling the front door

Internet

ROUTER

LB 1   ...   LB n

POD 1   POD 2   ...   POD n
Scaling the front door

- Router picks an LB
- LB (Sorting hat) knows that bobs-shop.com is in pod 2.
- LB sends traffic to a pod 2 upstream.
Load balancing the load balancers
Load balancing the load balancers

- Multiple LBs for redundancy and load distribution
- How to distribute? Which request goes to which LB?
- Active/backup? One LB per IP?
Load balancing the load balancers

- Multiple LBs for redundancy and load distribution
- How to distribute? Which request goes to which LB?
- Active/backup? One LB per IP?

- Equal-cost multi-path routing (ECMP)
- Consistent hashing based on TCP flow
- BGP with health-checks
The front door

Datacenter 1

Datacenter 2

Internet

BGP announce
23.227.38.69
The front door

BGP announce
23.227.38.69

Internet

Datacenter 1

Datacenter 2

ROUTER

LB 1 ... LB n

POD 1 POD 2 ... POD n

ROUTER

LB 1 ... LB n

POD 1 POD 2 ... POD n
BGP Anycast

Internet

BGP announce
23.227.38.69

 ROUTER

 LB 1  ...  LB n

 POD 1  POD 2  ...  POD n

Datacenter 1

BGP announce
23.227.38.69

 ROUTER

 LB 1  ...  LB n

 POD 1  POD 2  ...  POD n

Datacenter 2
BGP Anycast and Sorting Hat

Datacenter 1

Datacenter 2

Internet

BGP announce 23.227.38.69

ROUTER

LB 1 ... LB n

POD 1 POD 2 ... POD n

ROUTER

LB 1 ... LB n

POD 1 POD 2 ... POD n
BGP Anycast and Sorting Hat

Internet

BGP announce
23.227.38.69

Datacenter 1

Datacenter 2

ROUTER

LB 1

... LB n

POD 1

POD 2

... POD n

ROUTER

LB 1

... LB n

POD 1

POD 2

... POD n
Point of presence

Internet

BGP announce
23.227.38.69

ROUTER

LB 1  ...  LB n

POD 1  POD 2  ...  POD n

BGP announce
23.227.38.69

ROUTER

LB 1  ...  LB n
TL;DR

SUMMARY AND KEY TAKEAWAYS
Isolation vs. capacity
Spectrum of multi-tenant architectures

- Share nothing (2004)
  - More isolation
  - Less utilization

- Share everything (2005-2012)
  - Cheaper
  - More capacity
Spectrum of multi-tenant architectures

- Share nothing (2004)
  - More isolation
  - Less utilization

- Database isolation (2013)

- Share everything (2005-2012)
  - Cheaper
  - More capacity
Spectrum of multi-tenant architectures

**Share nothing**
- 2004
- More isolation
- Less utilization

**Podding**
- 2015/2016

**Database isolation**
- 2013

**Share everything**
- 2005-2012
- Cheaper
- More capacity
Spectrum of multi-tenant architectures

Share nothing  Podding  Floating capacity  Database isolation  Share everything

- More isolation
- Less utilization
- Cheaper
- More capacity
Spectrum of multi-tenant architectures

More isolation
Less utilization

Our sweet spot!

Cheaper
More capacity
nginx is awesome.
BGP and ECMP

within your network!
Find your own flash sale problem.

Embrace it!
Thanks! Questions?

github.com/openresty/lua-nginx-module

github.com/Exa-Networks/exabgp

tools.ietf.org/html/rfc2992

FLORIAN WEINGARTEN
flo@shopify.com
@fw1729