

Scaling Shopify's multi-tenant architecture across multiple datacenters

FLORIAN WEINGARTEN

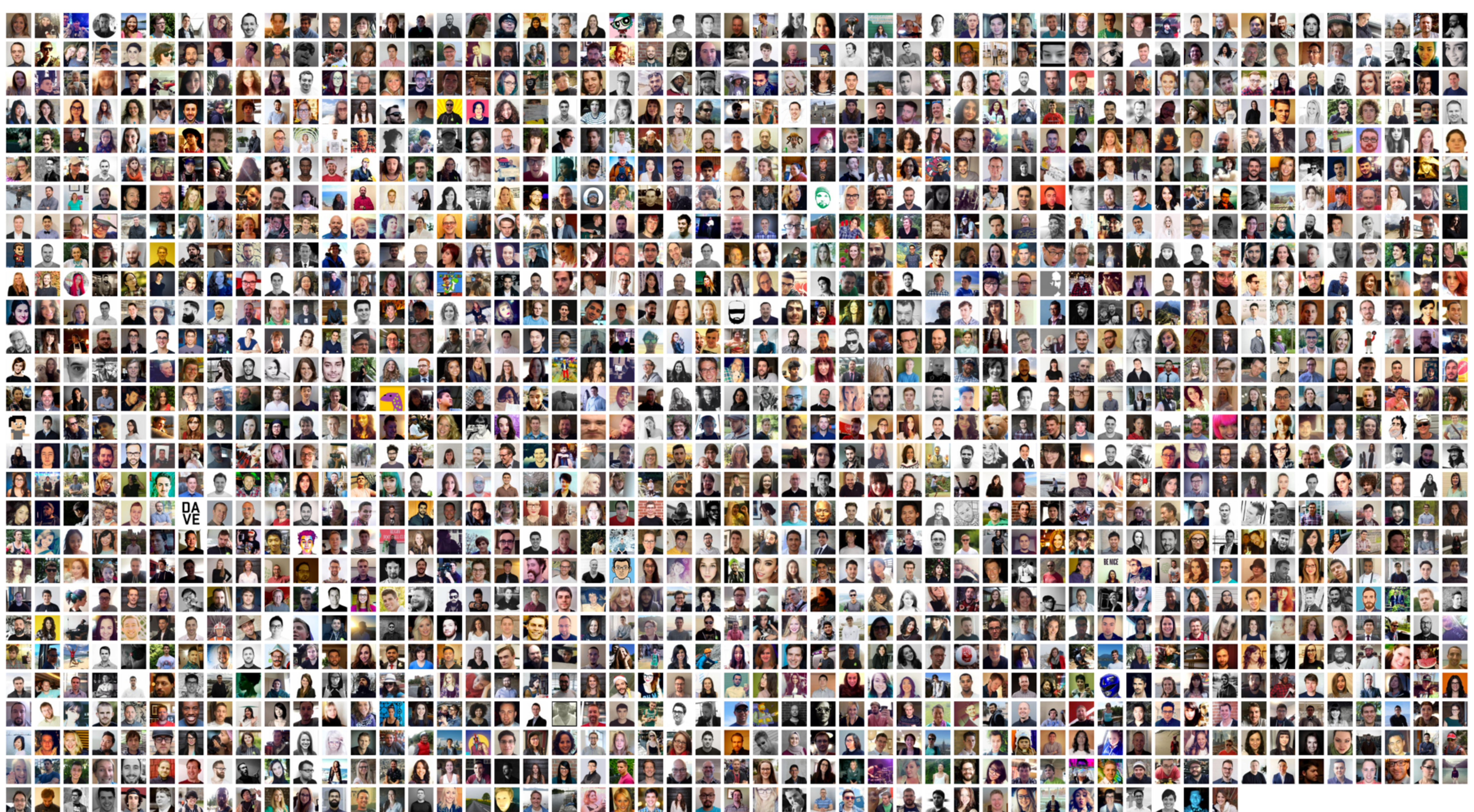
flo@shopify.com

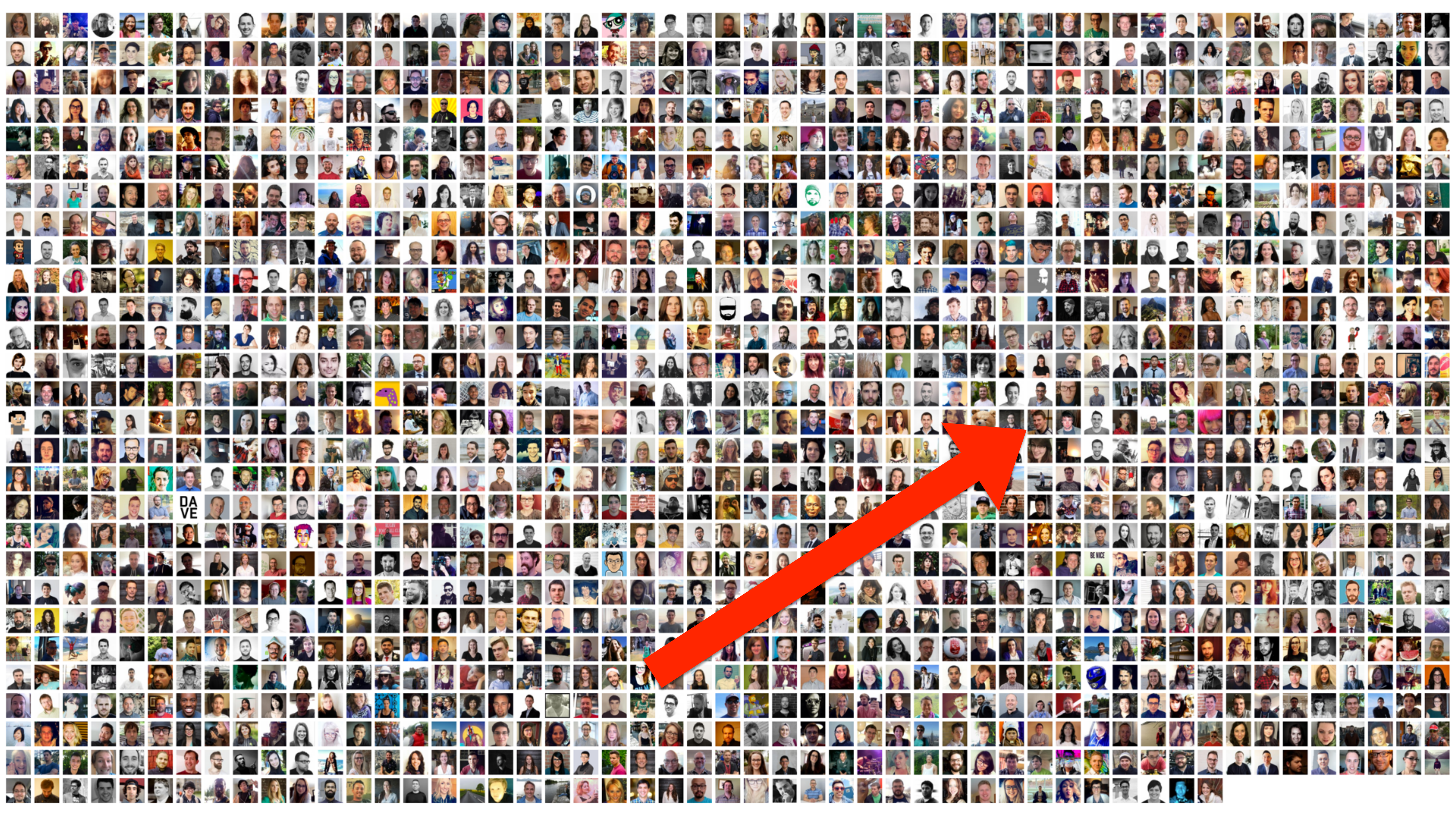
@fw1729











DAVE

BE NICE

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



Kylie Jenner ✓
@KylieJenner

Follow

Get my absolute favorite shade Exposed right now on KylieCosmetics.com



RETWEETS
1,804

LIKES
10,898



9:33 PM - 24 Jun 2016

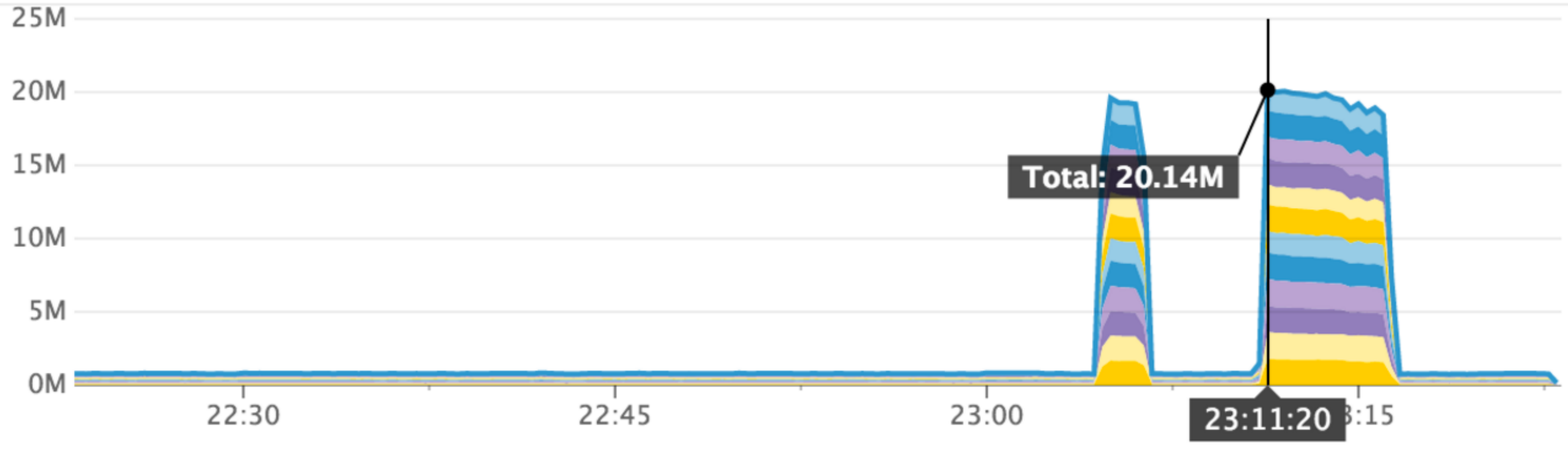
11K

FOLLOWERS
16.6M






Throughput by LB



“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

Share nothing	?	Share everything
Little capacity		Huge capacity
Bad utilization		Great utilization
Flash sale problem		Great for flash sales
Crazy expensive		Cheap
Full isolation and resiliency		No isolation or resiliency
Horizontal scale is easy		Horizontal scale can be hard

“Shared everything” is not good enough!

Spectrum of multi-tenant architectures

Share nothing



2004

Share everything



2005-2012

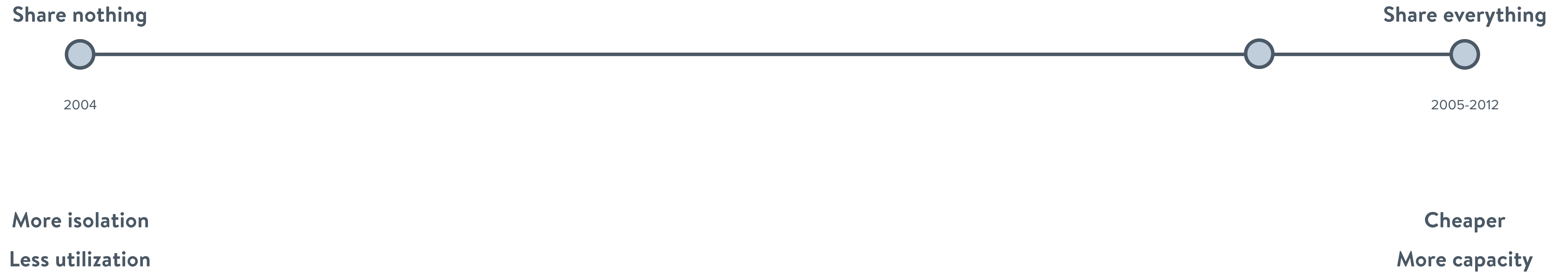
More isolation

Less utilization

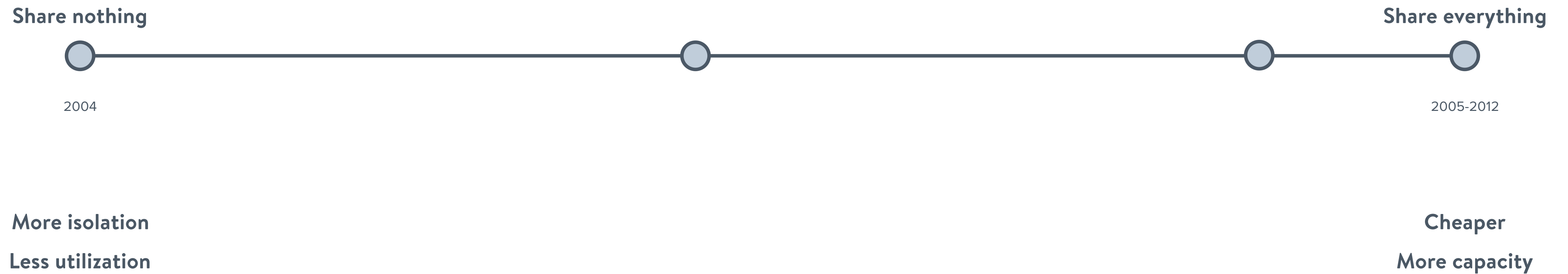
Cheaper

More capacity

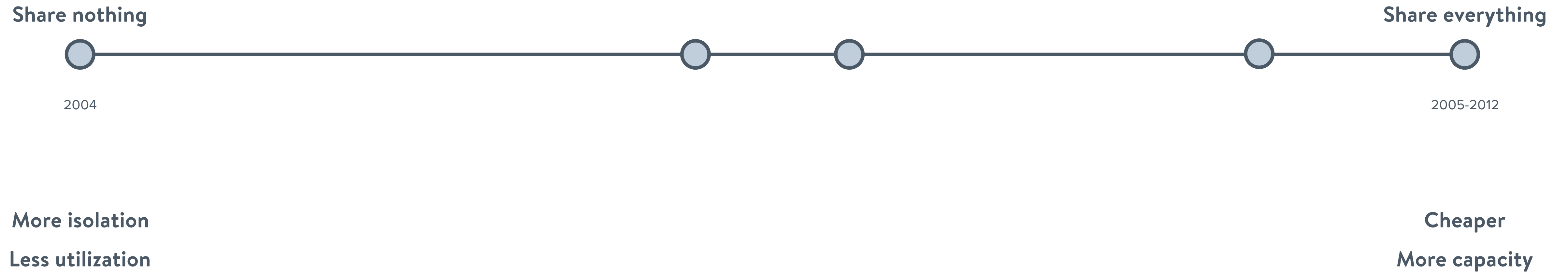
Spectrum of multi-tenant architectures



Spectrum of multi-tenant architectures



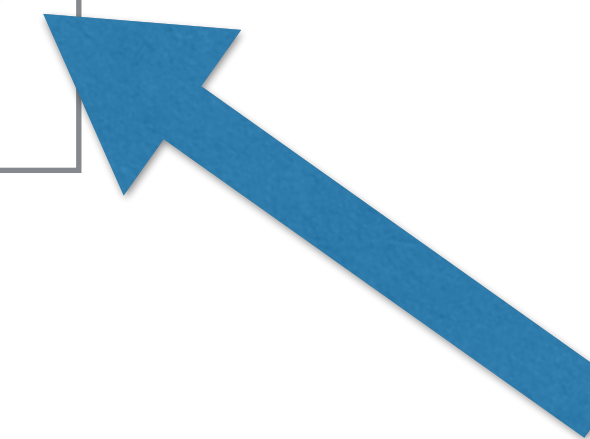
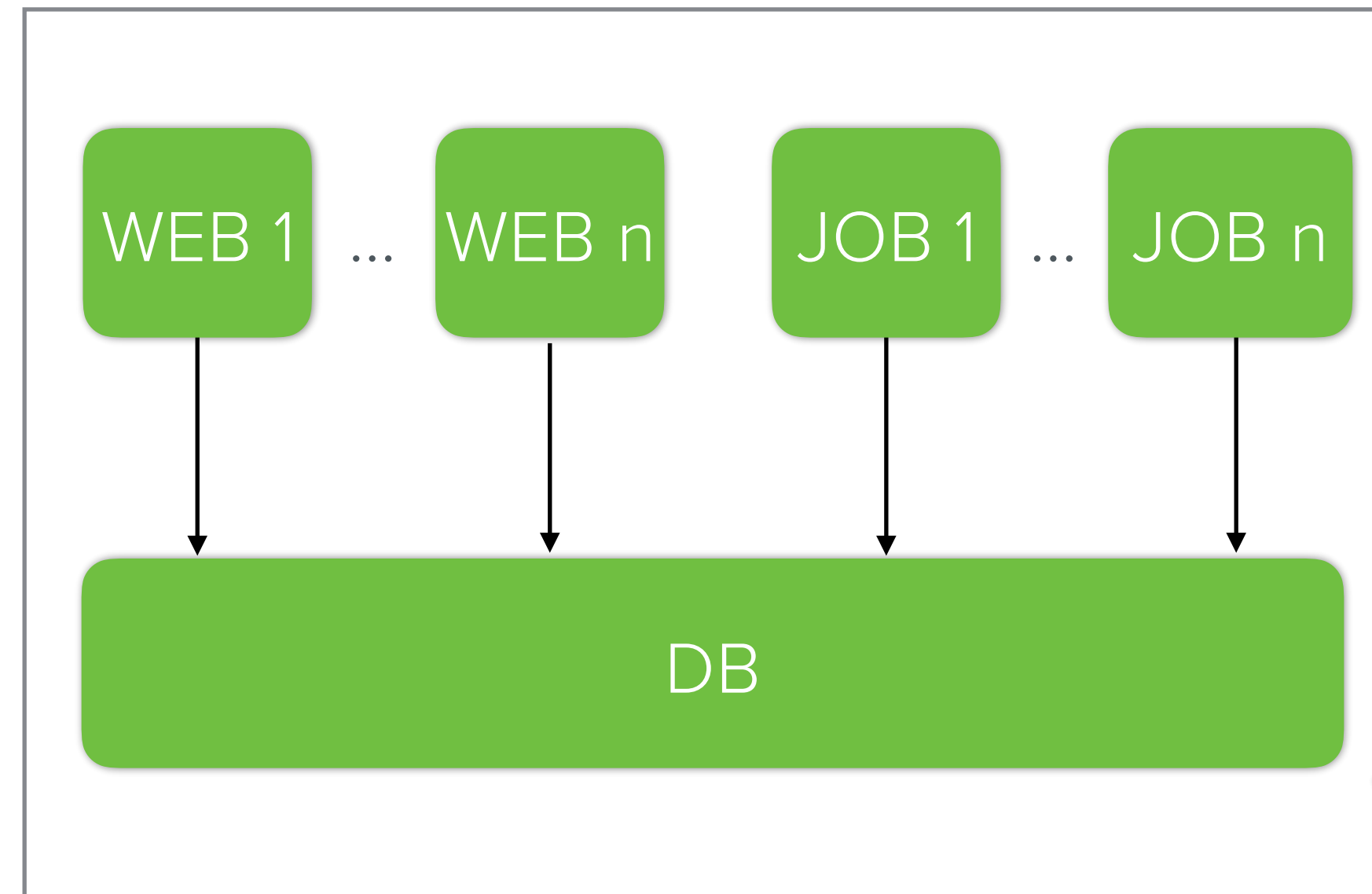
Spectrum of multi-tenant architectures



Spectrum of multi-tenant architectures

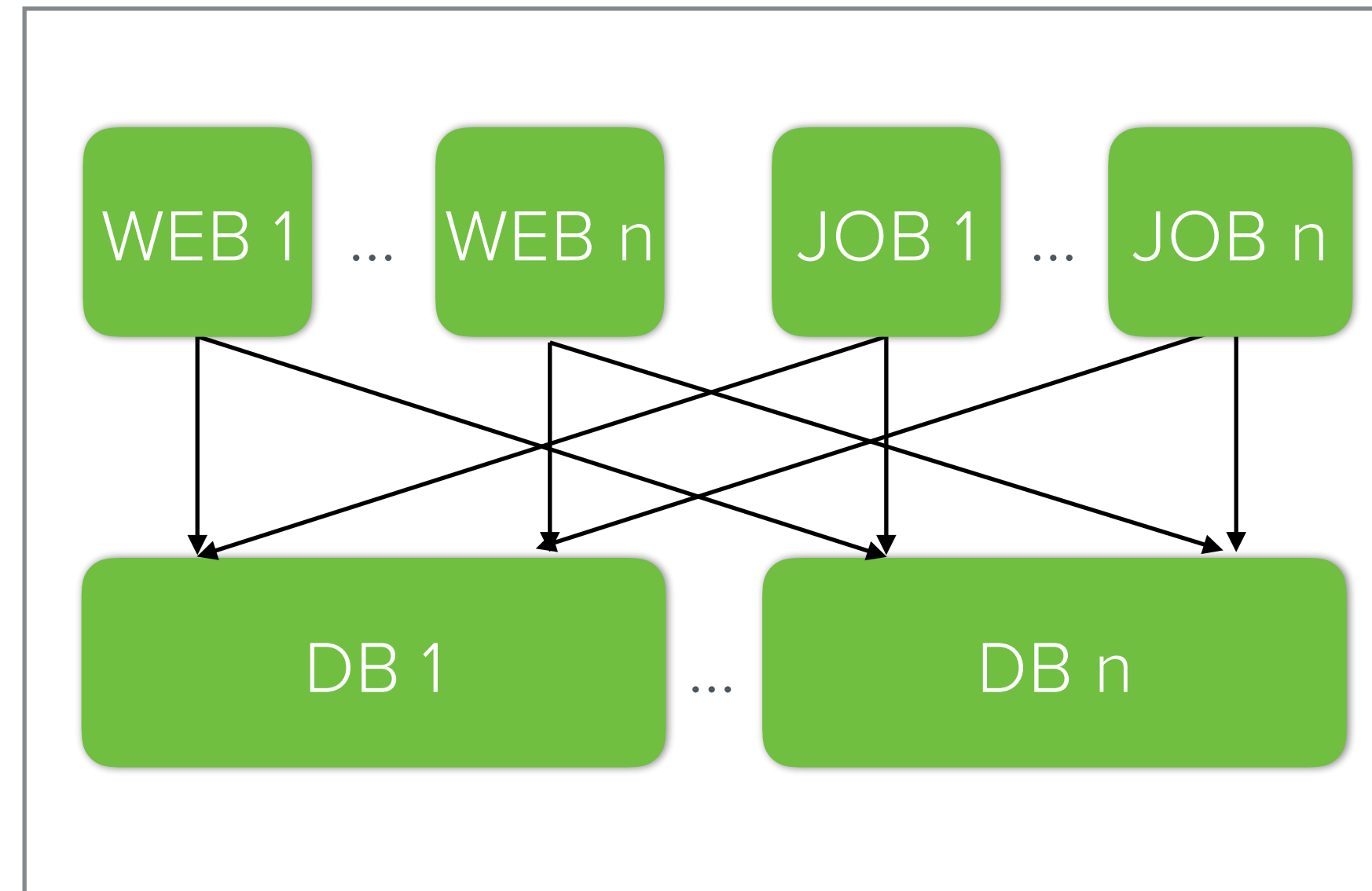


Shared everything



Big, expensive, SPOF

Database isolation

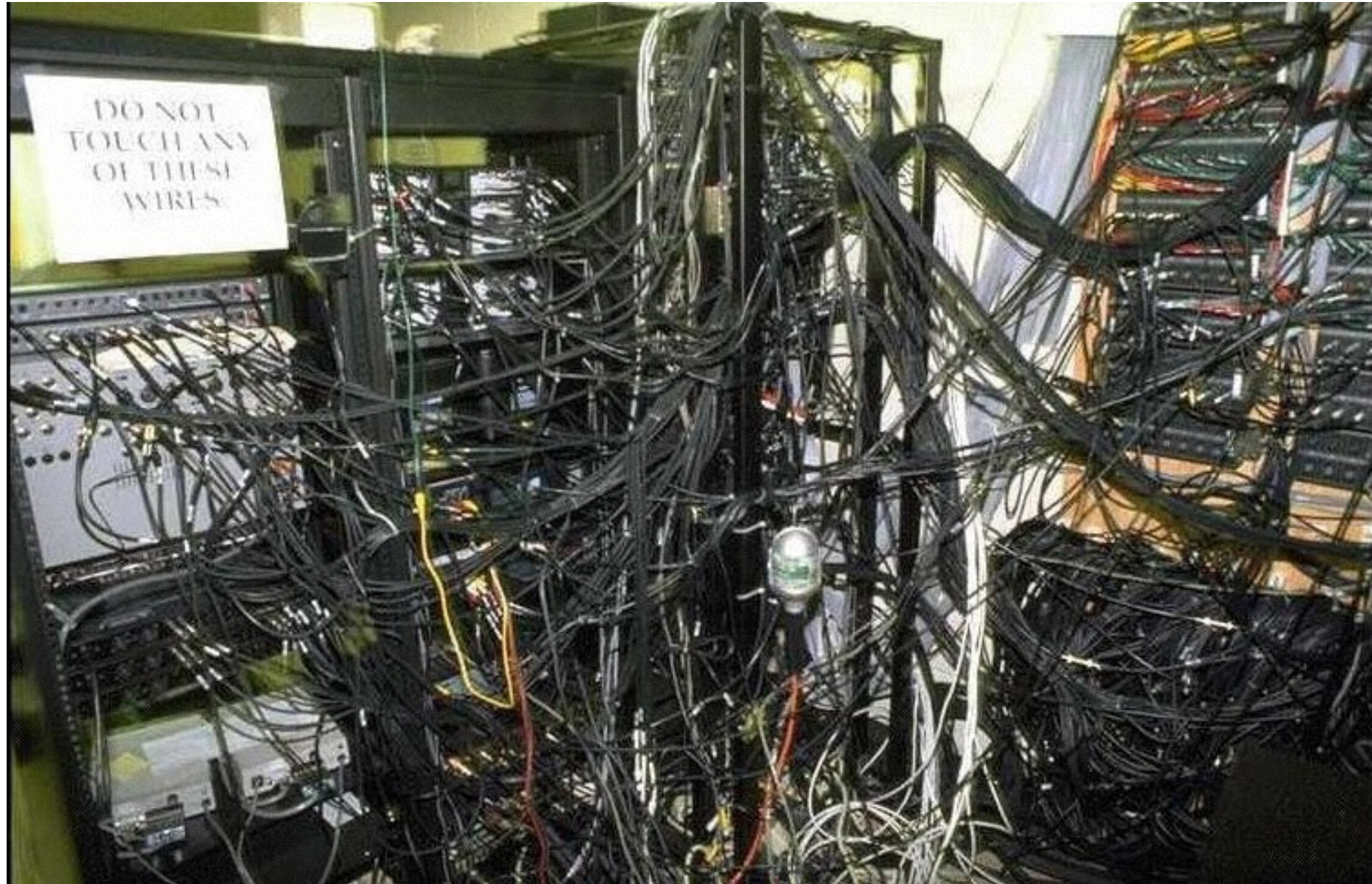




MULTI-DC

BACKUP SITE

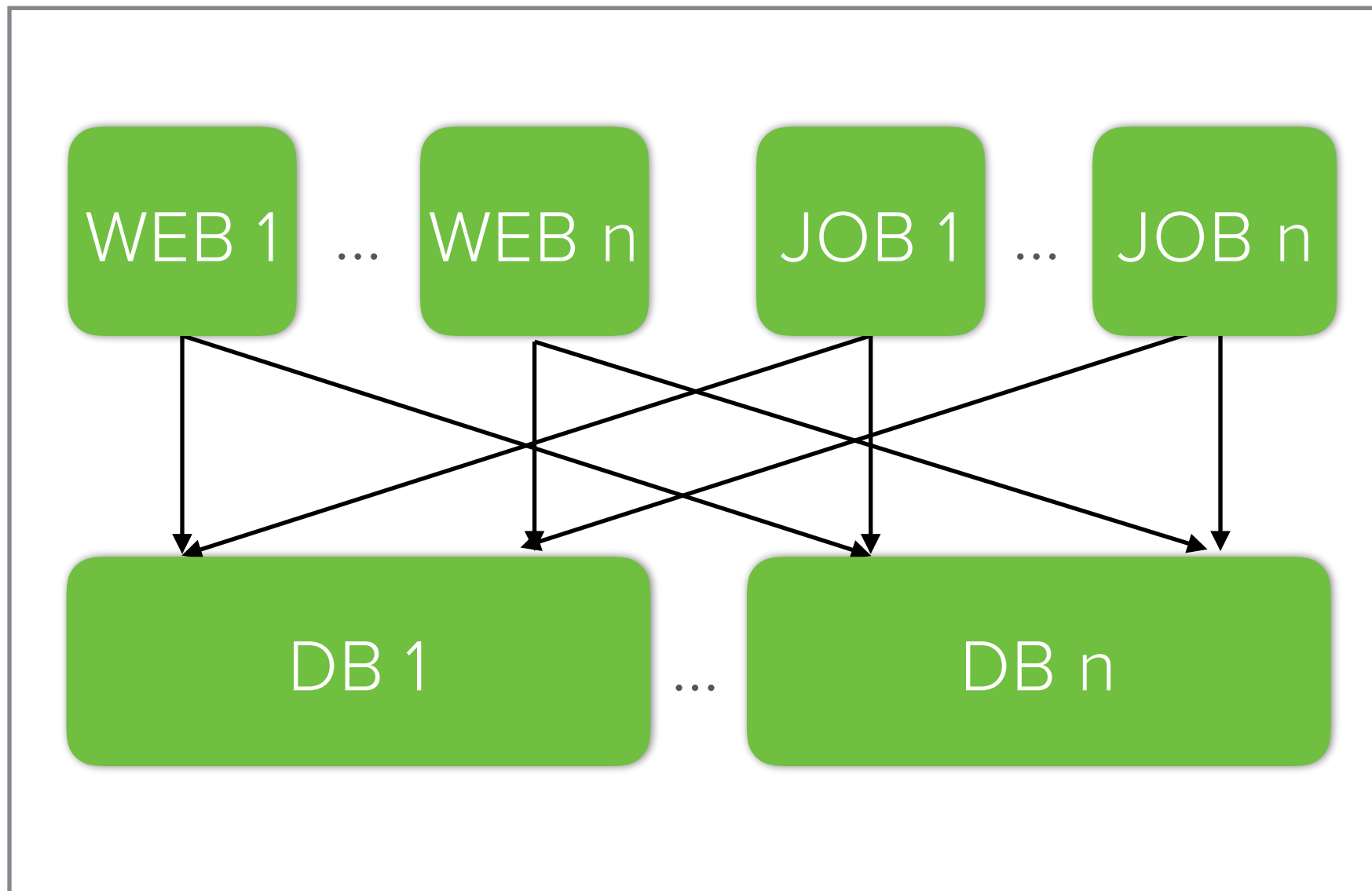
Why?



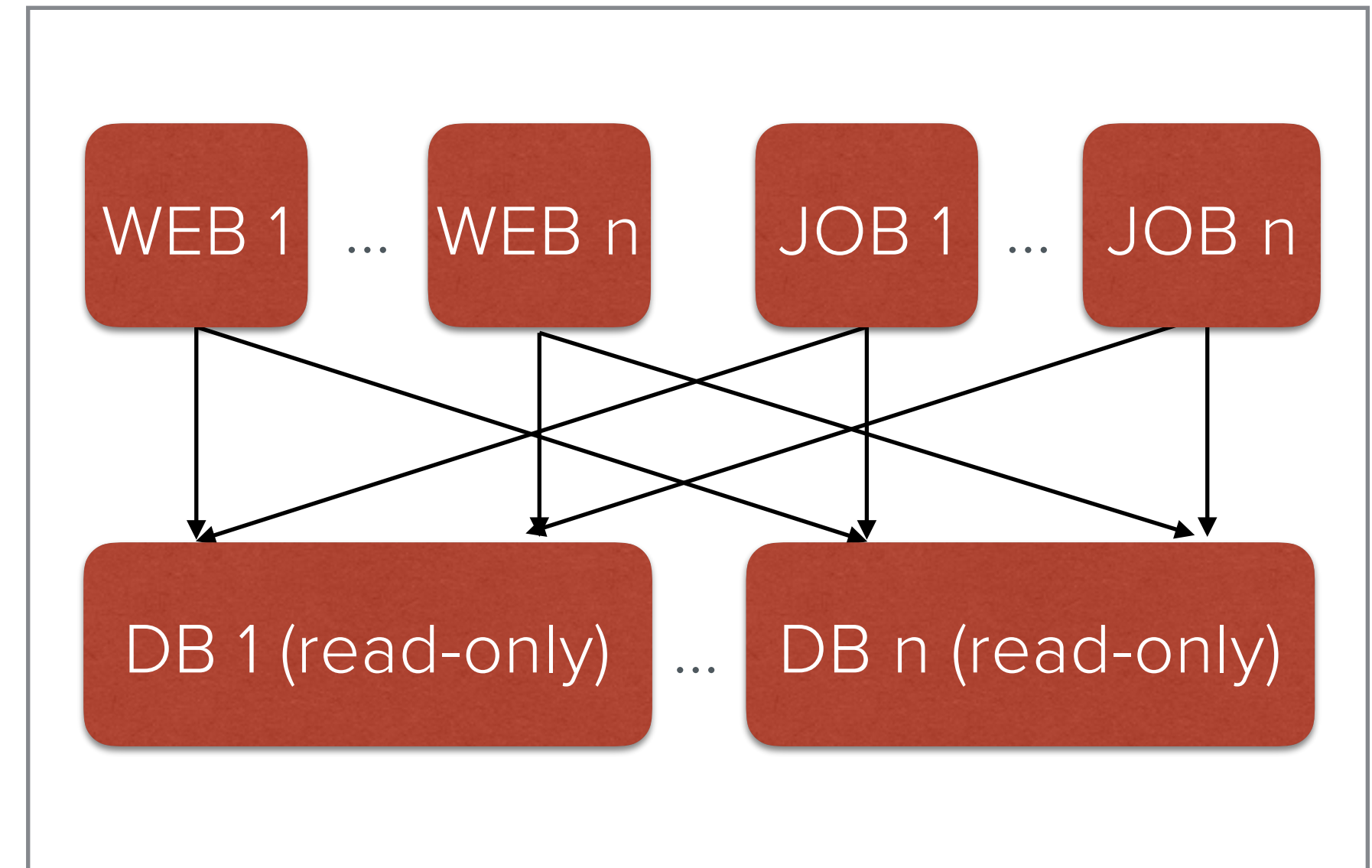
Maintenance



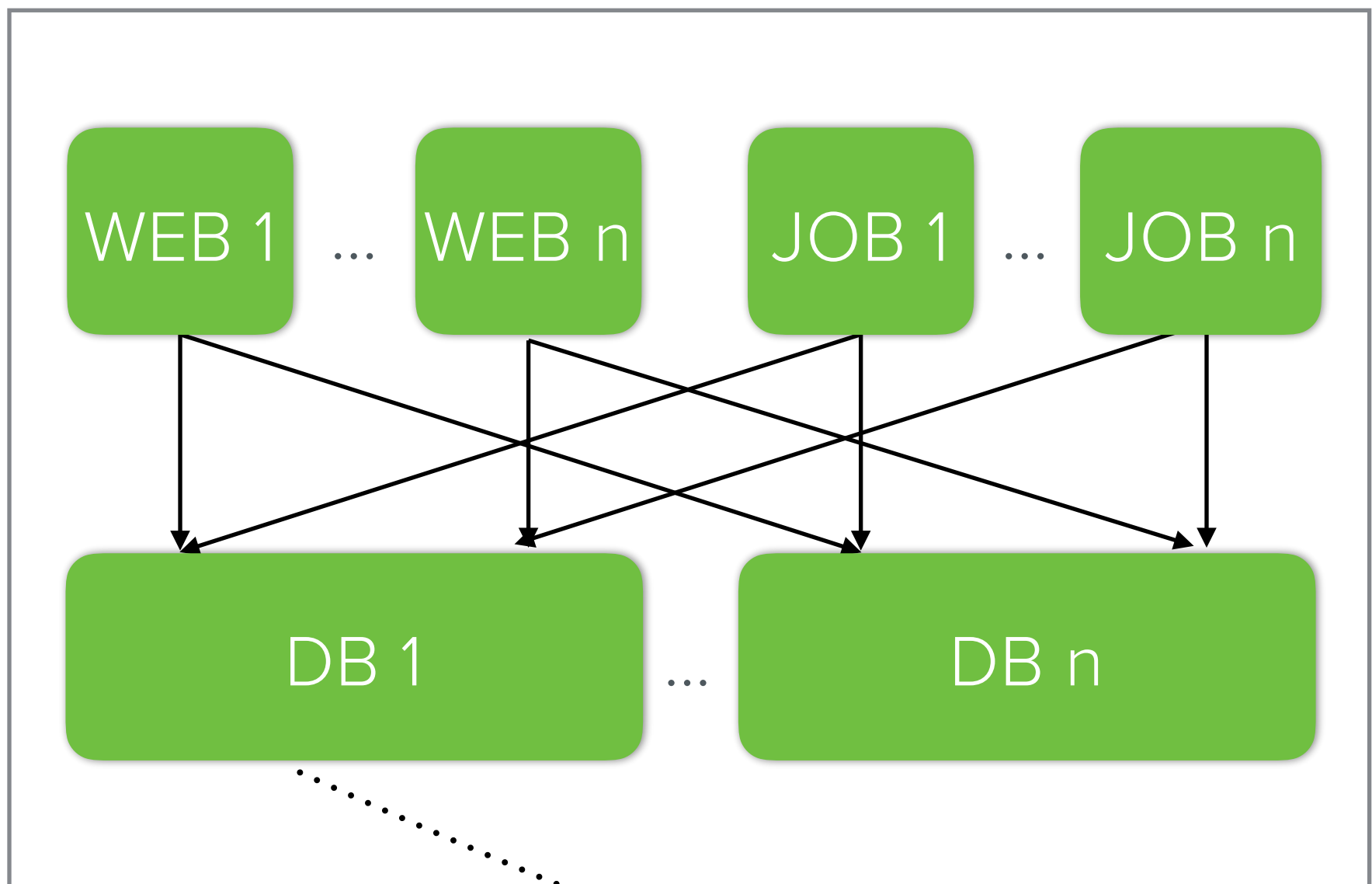
Redundancy and disaster recovery



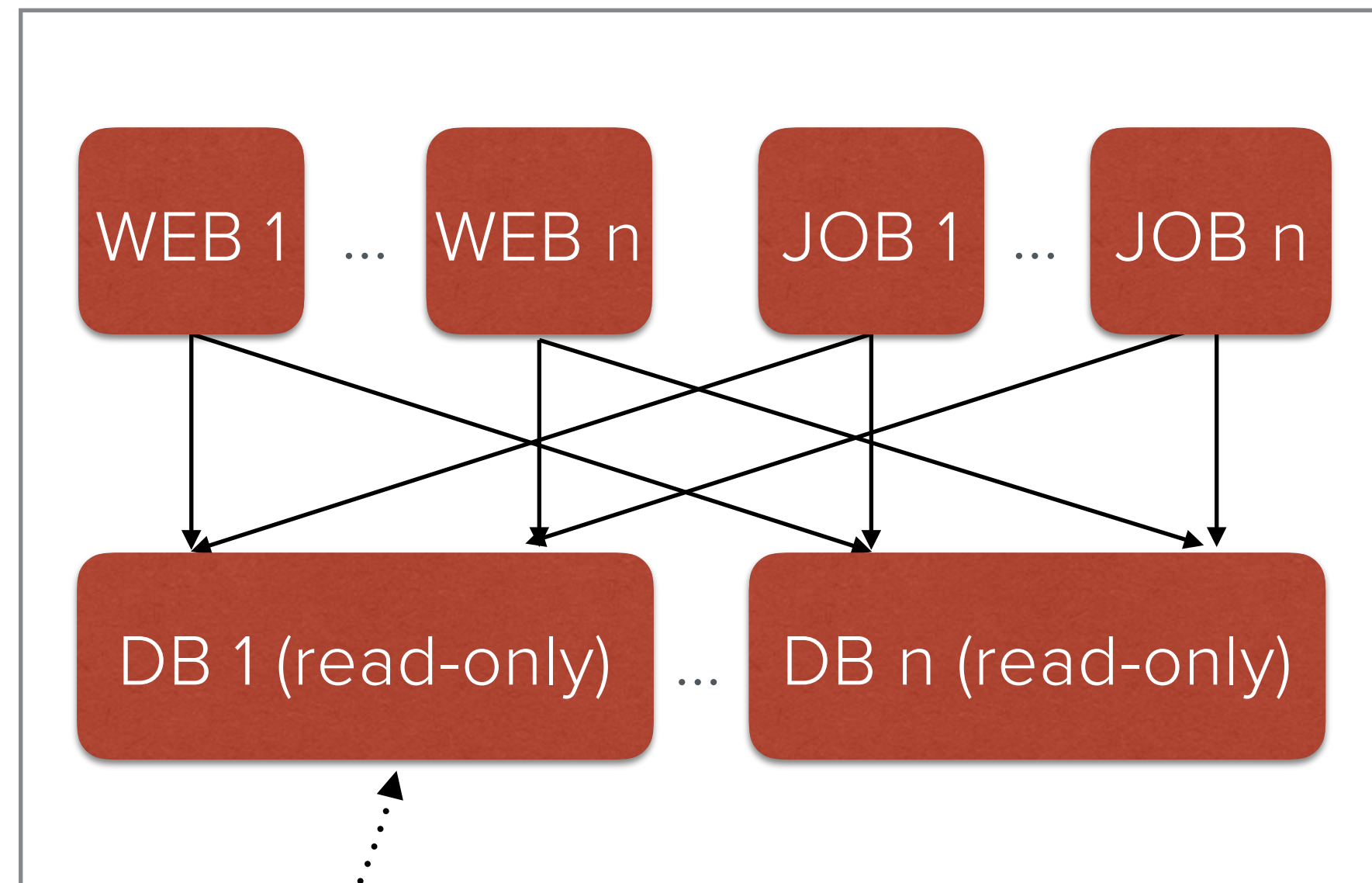
Active datacenter
All traffic goes here



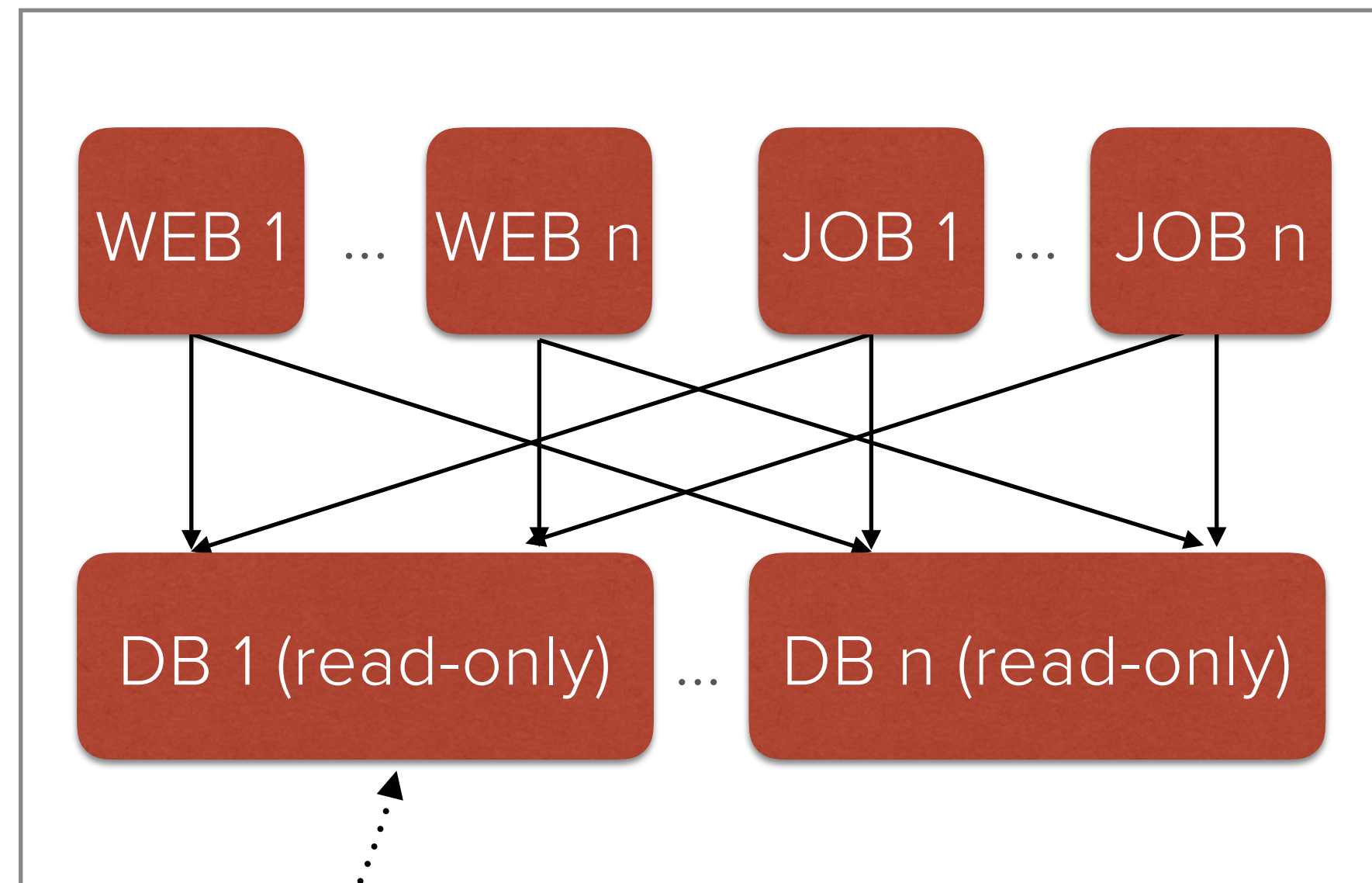
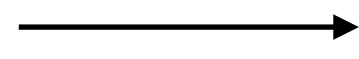
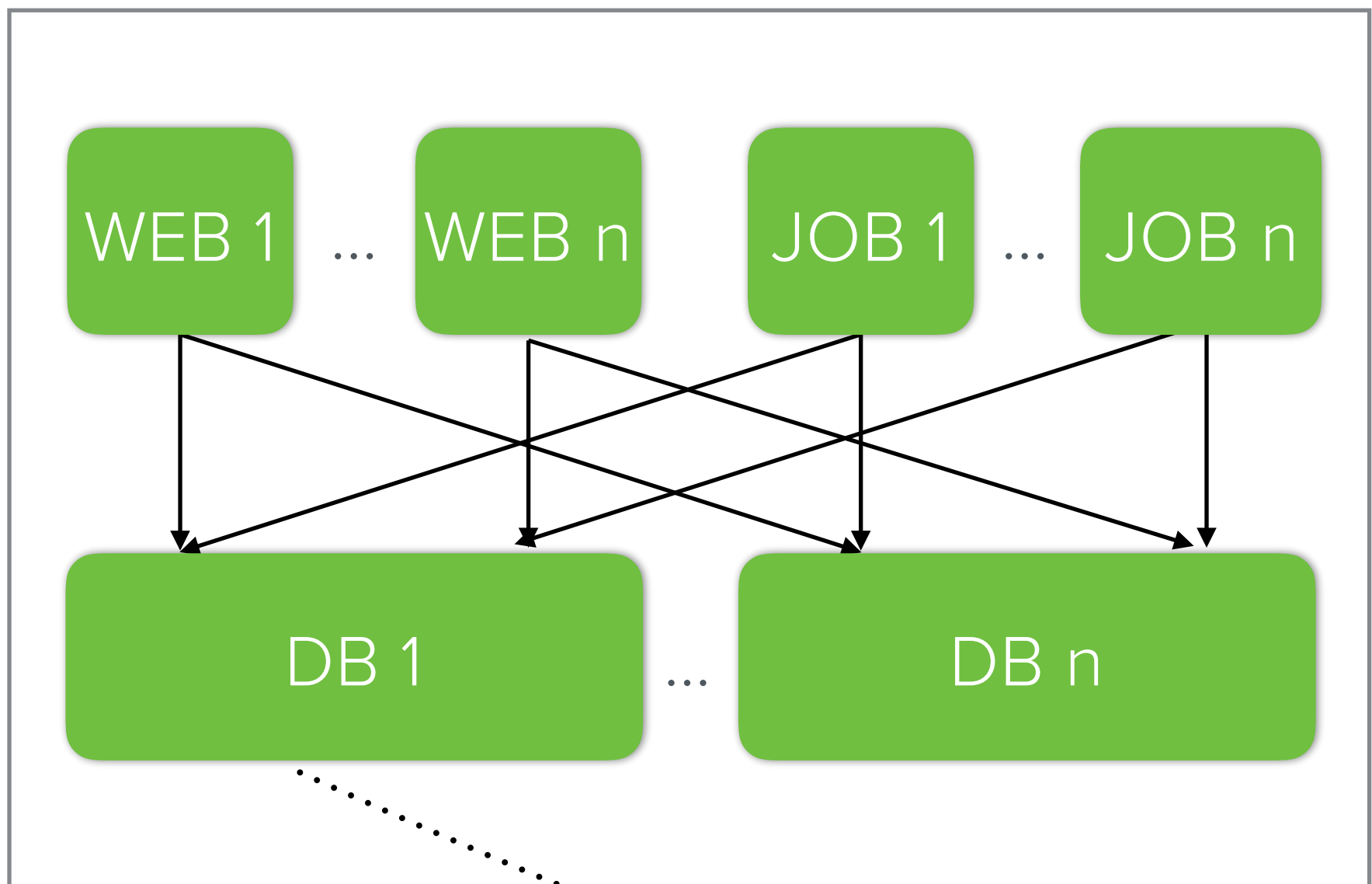
Backup datacenter
All databases are read-only



Active datacenter
All traffic goes here



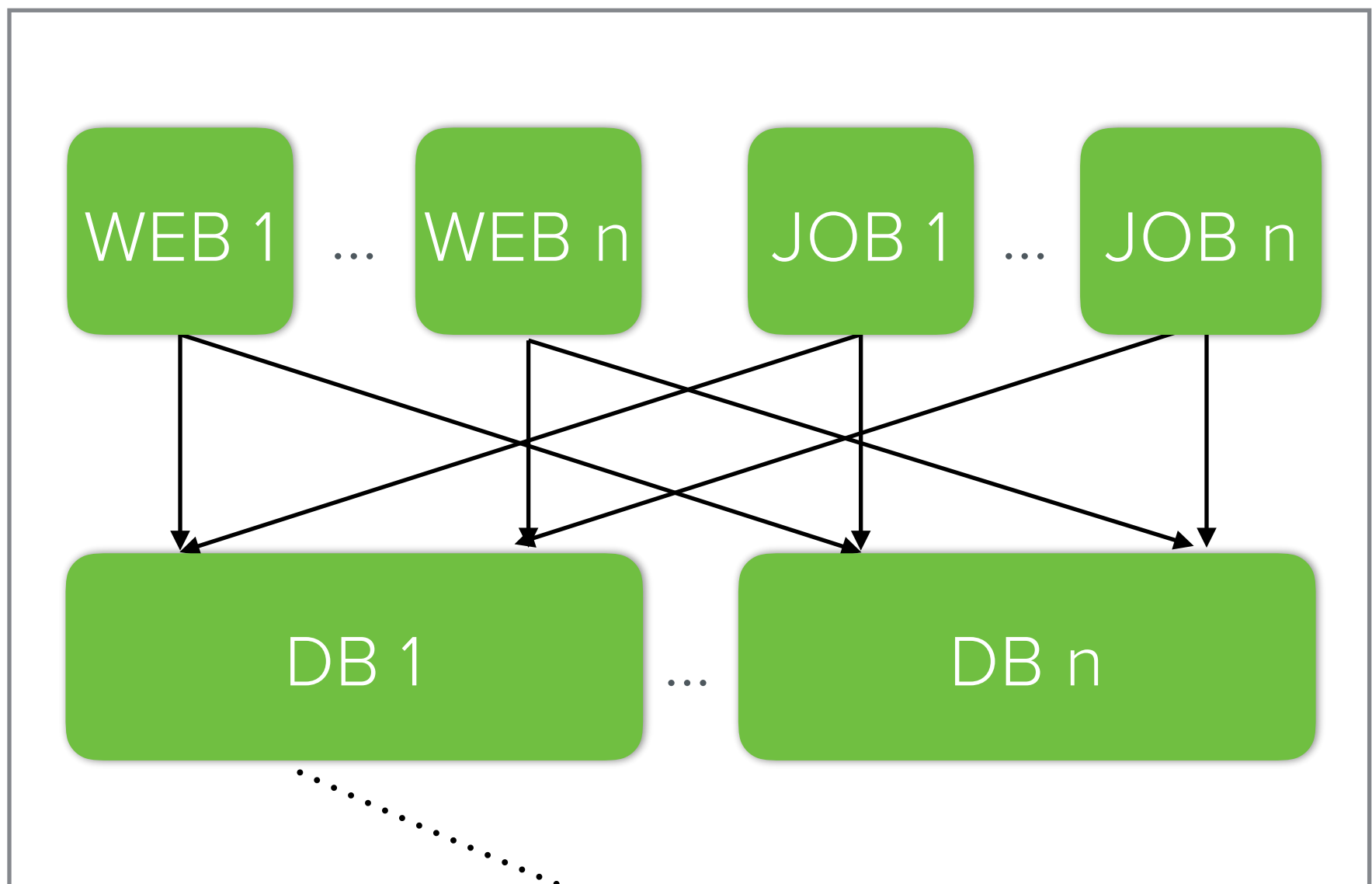
Backup datacenter
All databases are read-only



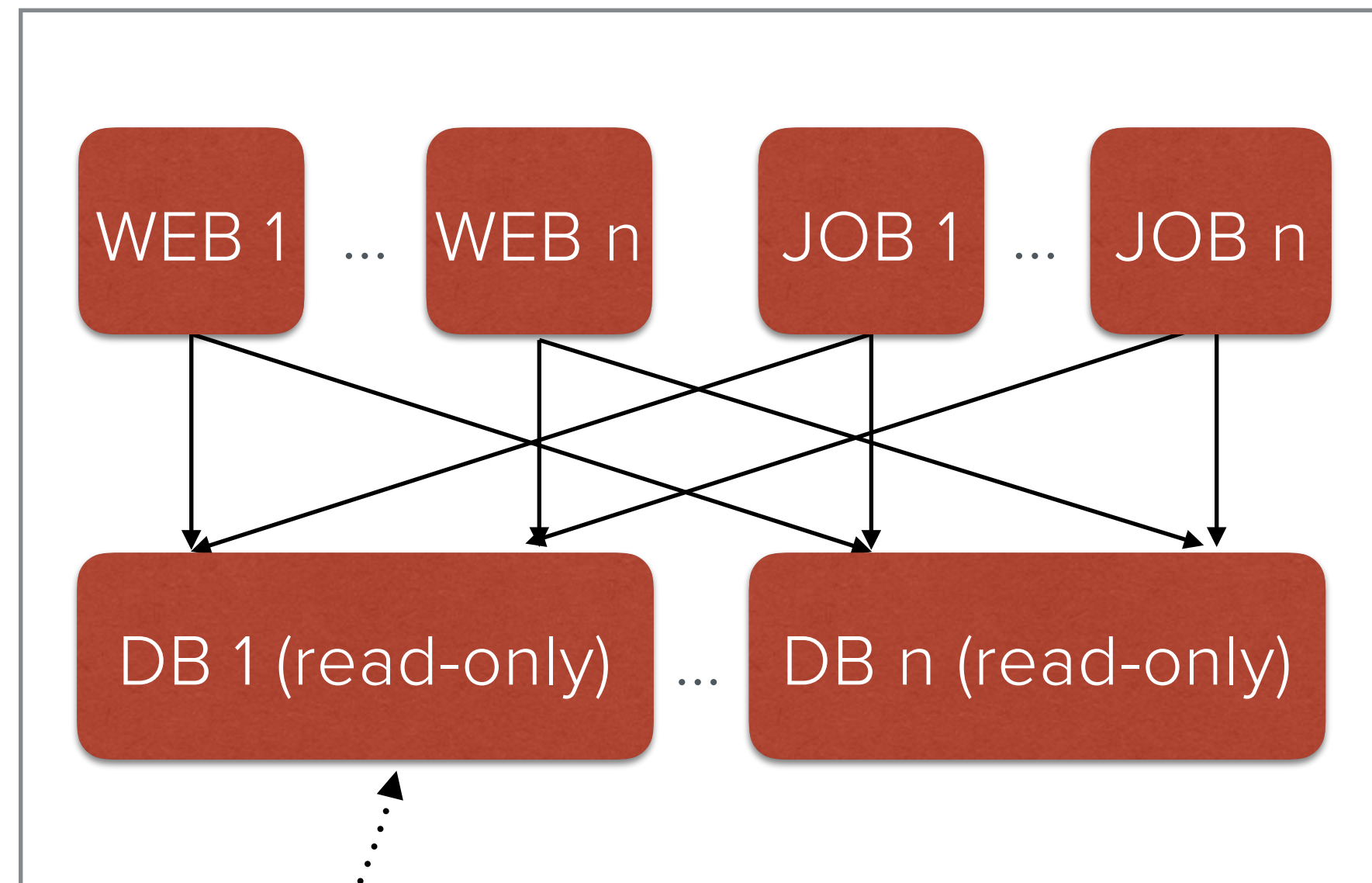
Replication

Active datacenter
All traffic goes here

Backup datacenter
All databases are read-only

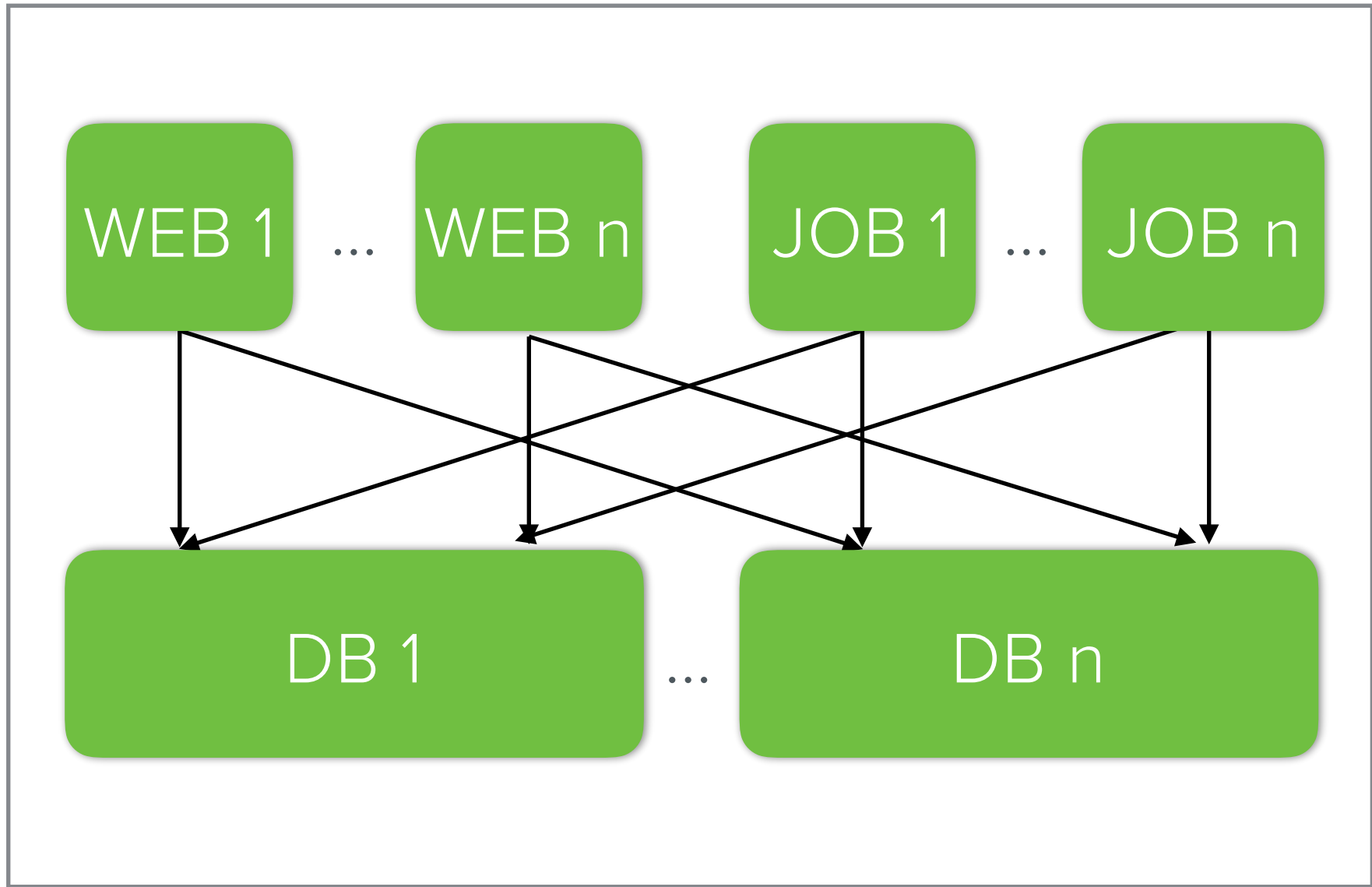
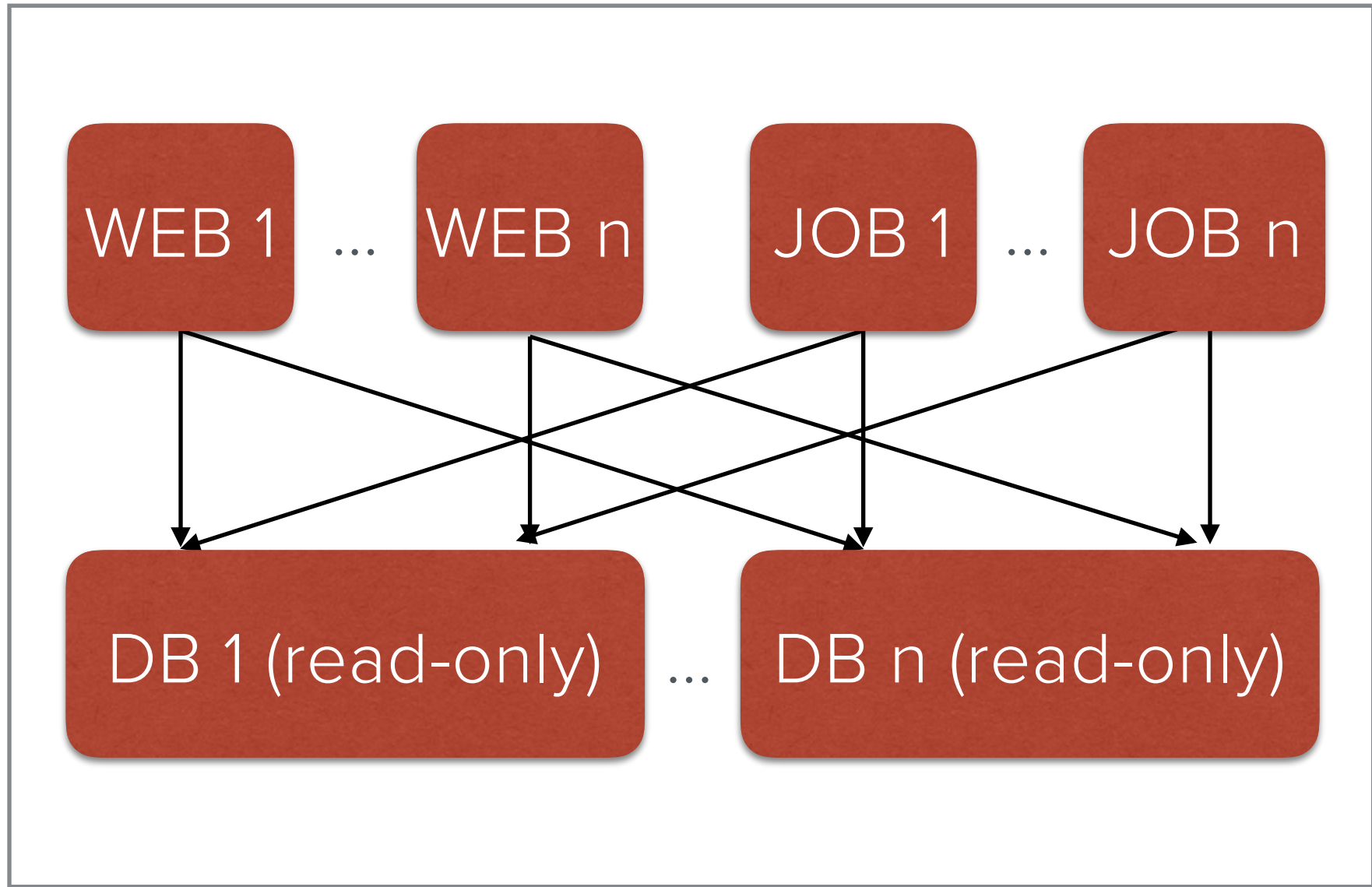


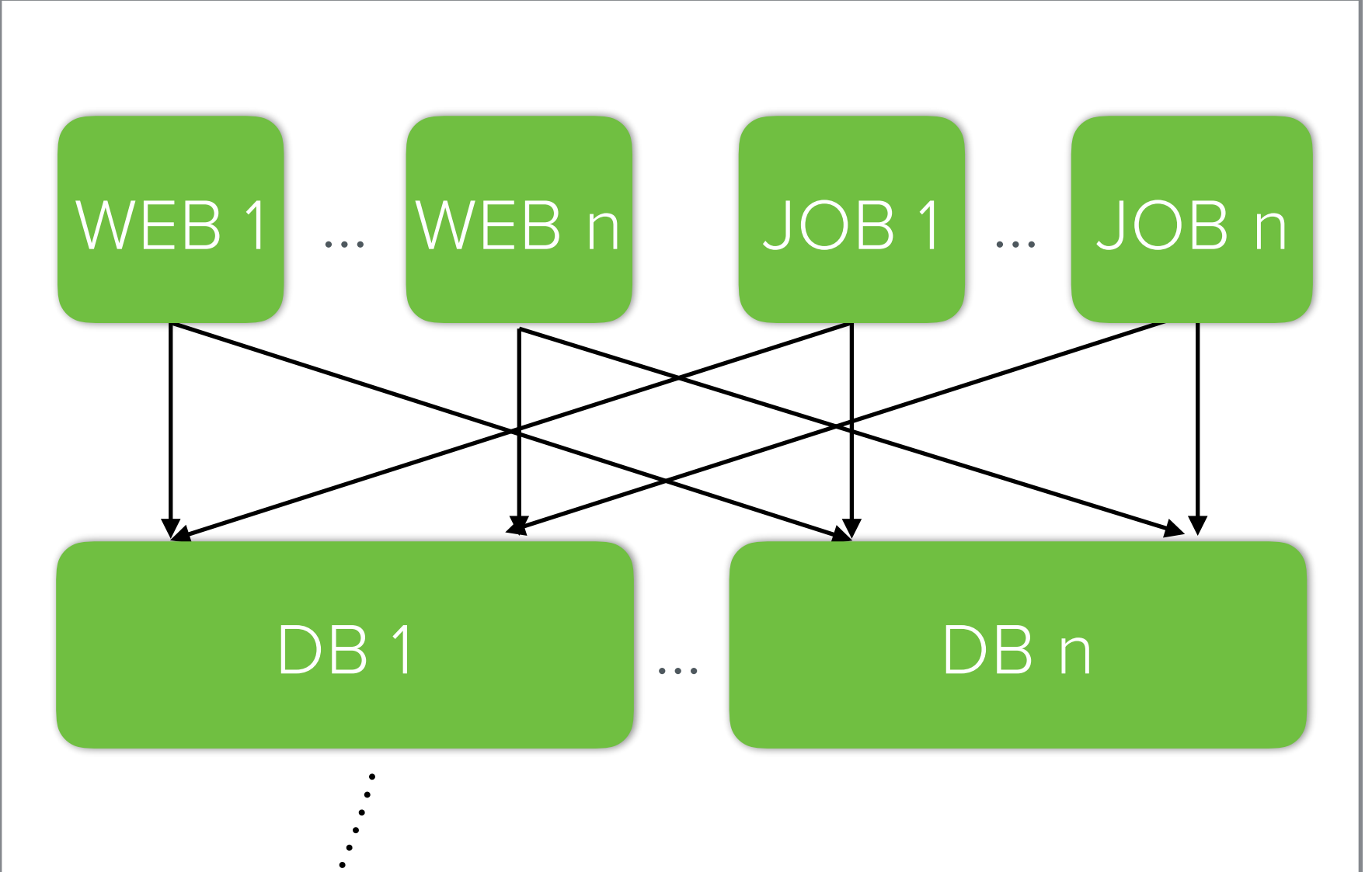
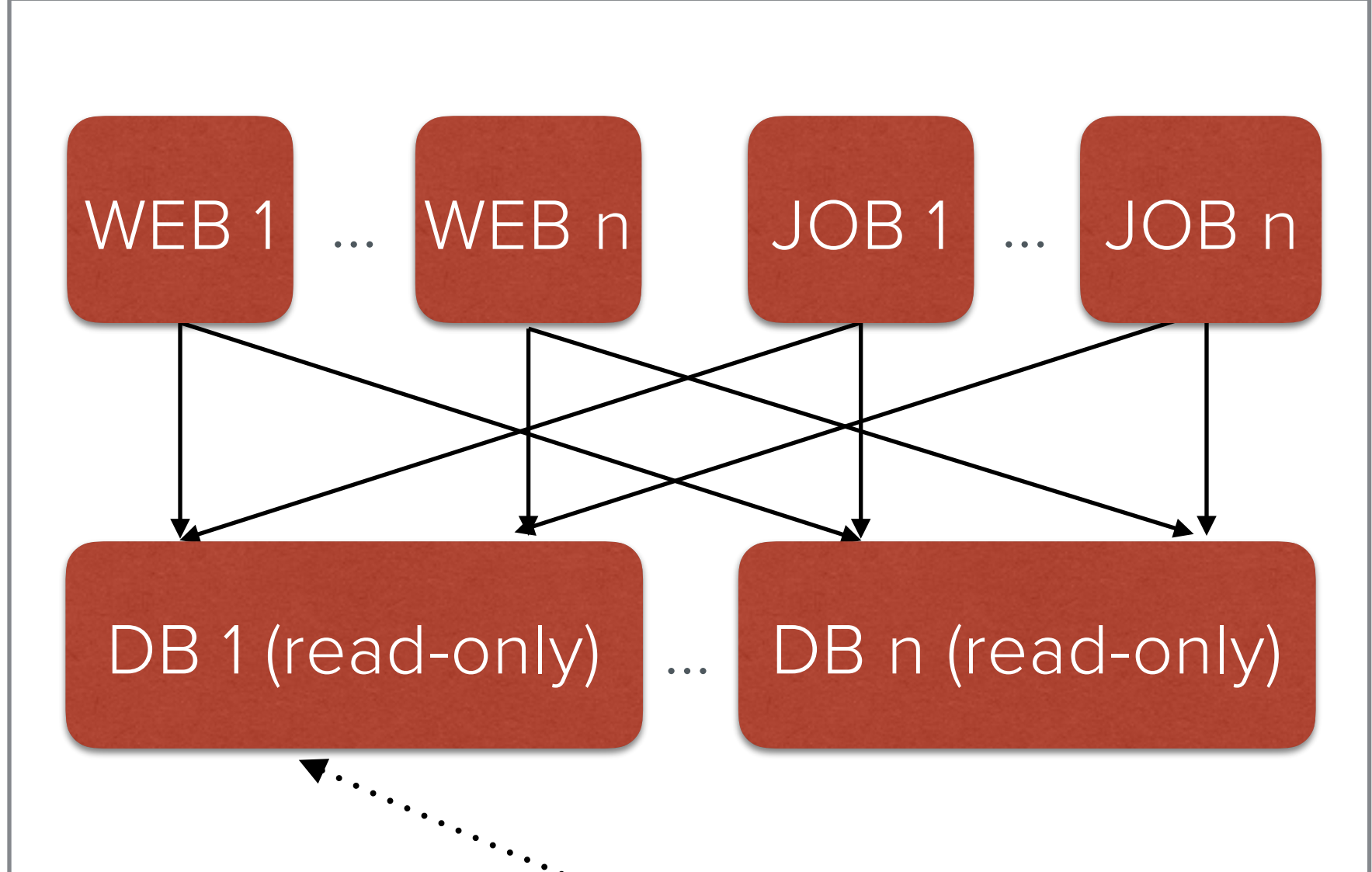
Failover



Active datacenter
All traffic goes here

Backup datacenter
All databases are read-only





Replication

How we used to do failovers



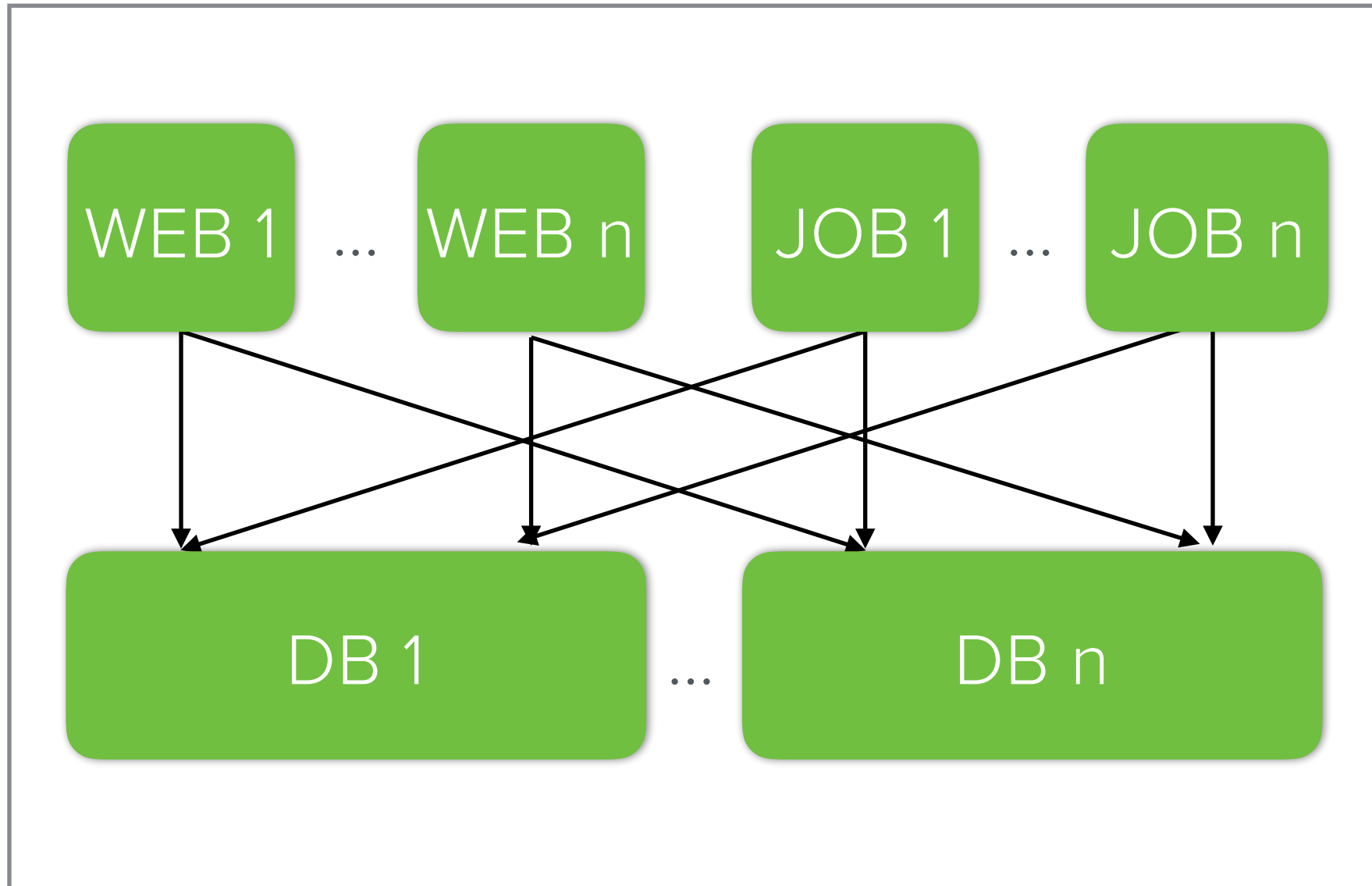
How we do failovers now

```
$ bin/dc-failover
```

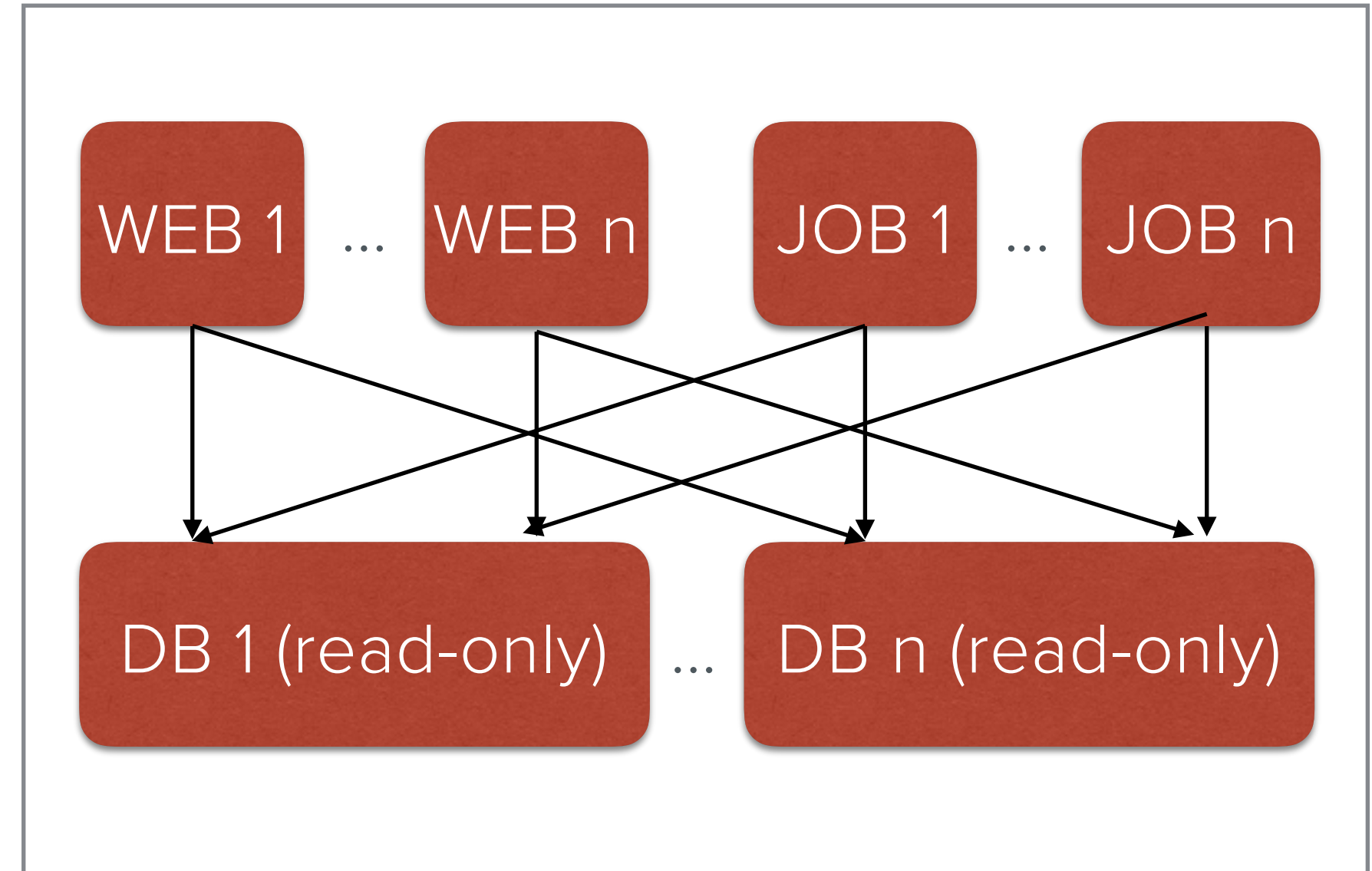

A dimly lit, modern office interior. In the foreground, there are several grey armchairs. In the middle ground, a long wooden table with several metal chairs is set up, resembling a bar or a meeting area. In the background, several people are sitting at long tables, working on laptops. The ceiling is high with exposed pipes and modern lighting fixtures. Large windows are visible on the left side, showing a view of a city. The overall atmosphere is professional and collaborative.

MULTI-DC

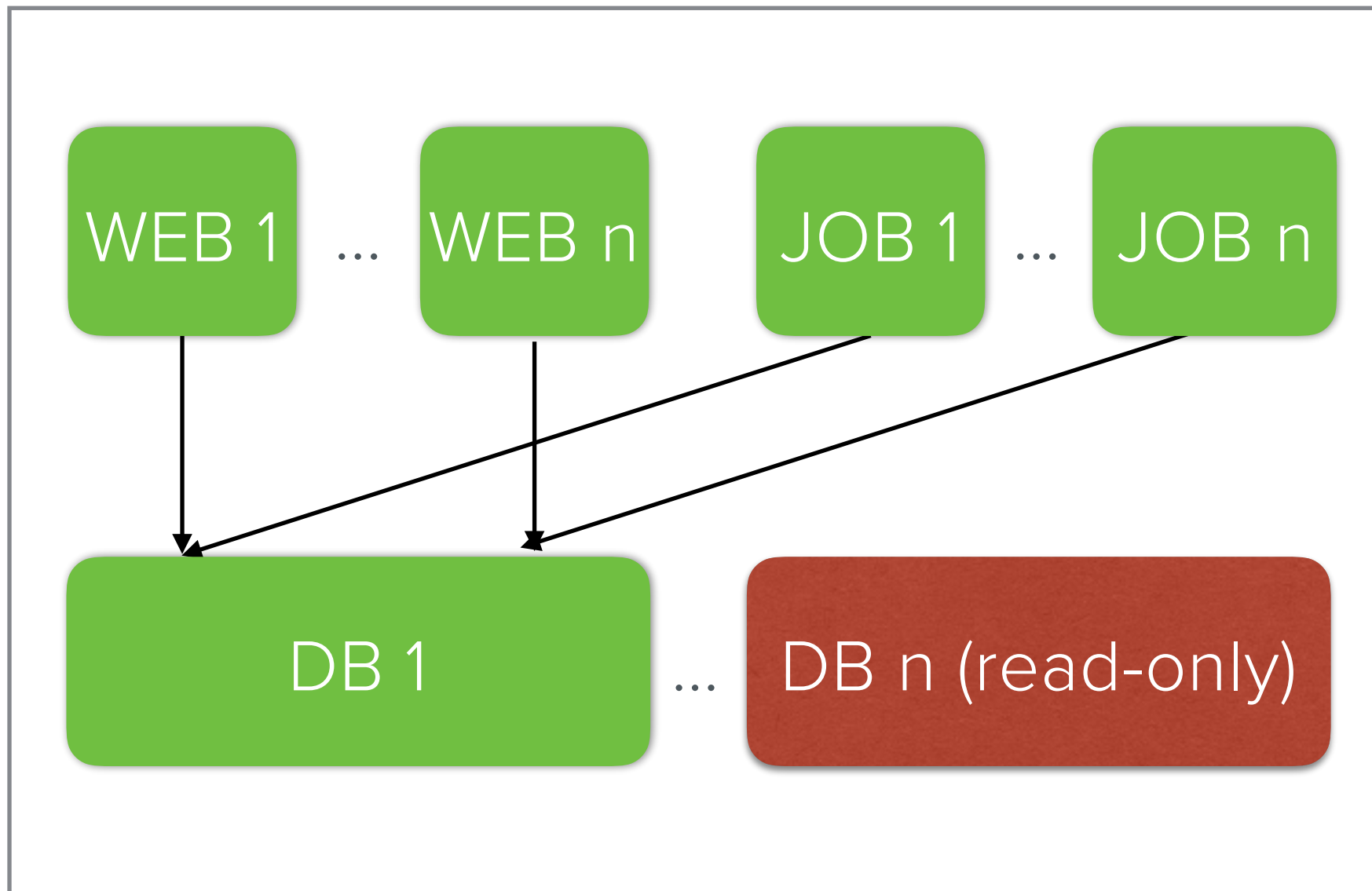
PODS



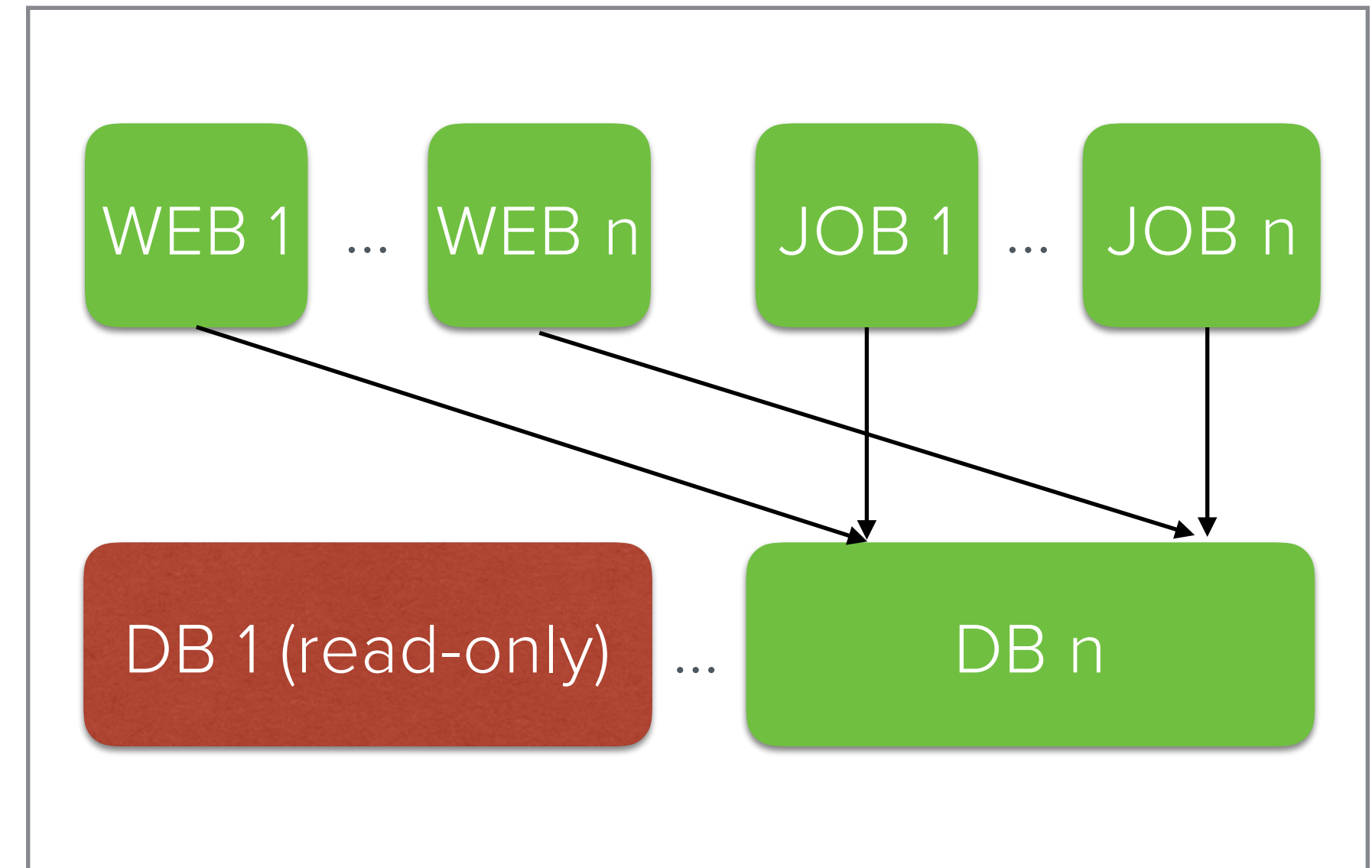
Active datacenter



Passive backup datacenter

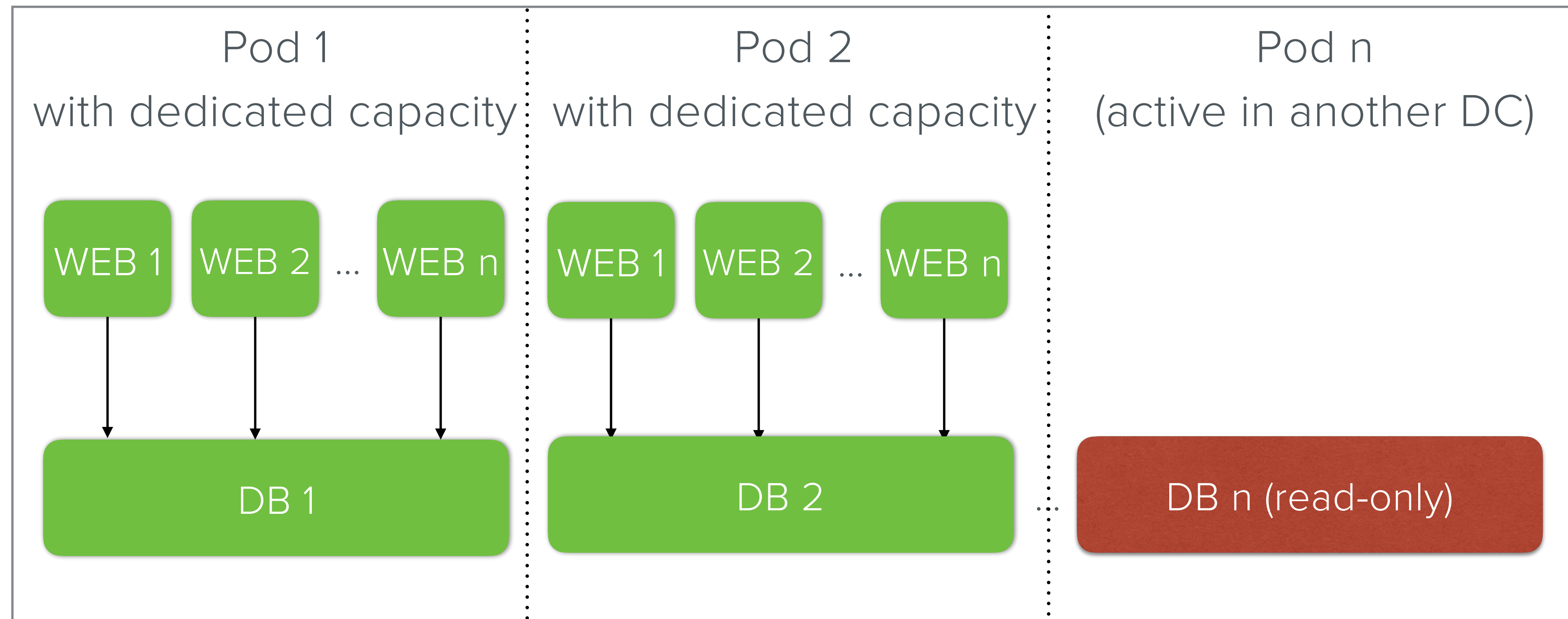


(Partially) active datacenter 1



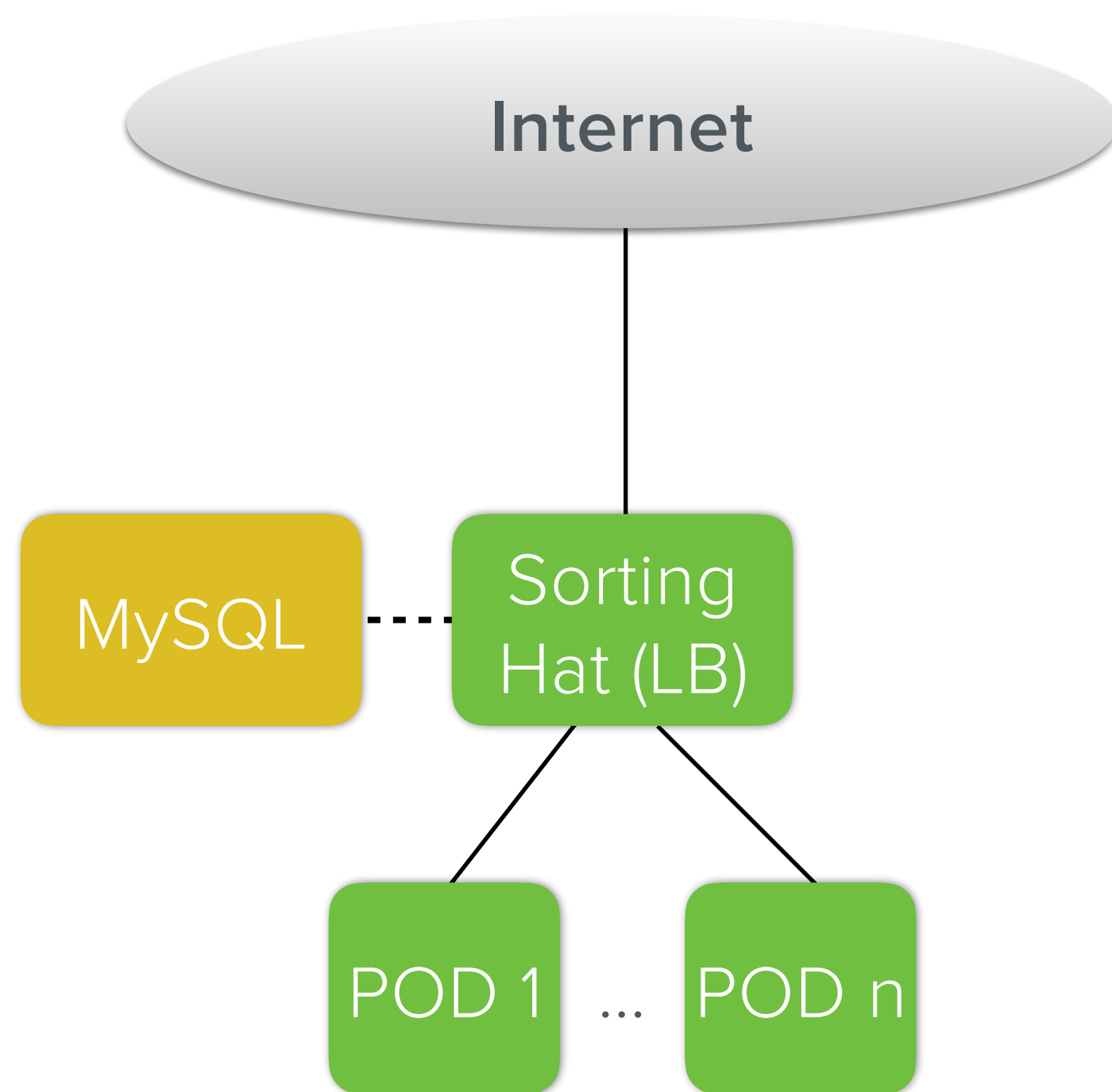
(Partially) active datacenter 2

Podding



**How to route requests
to the right pod?**

Sorting Hat

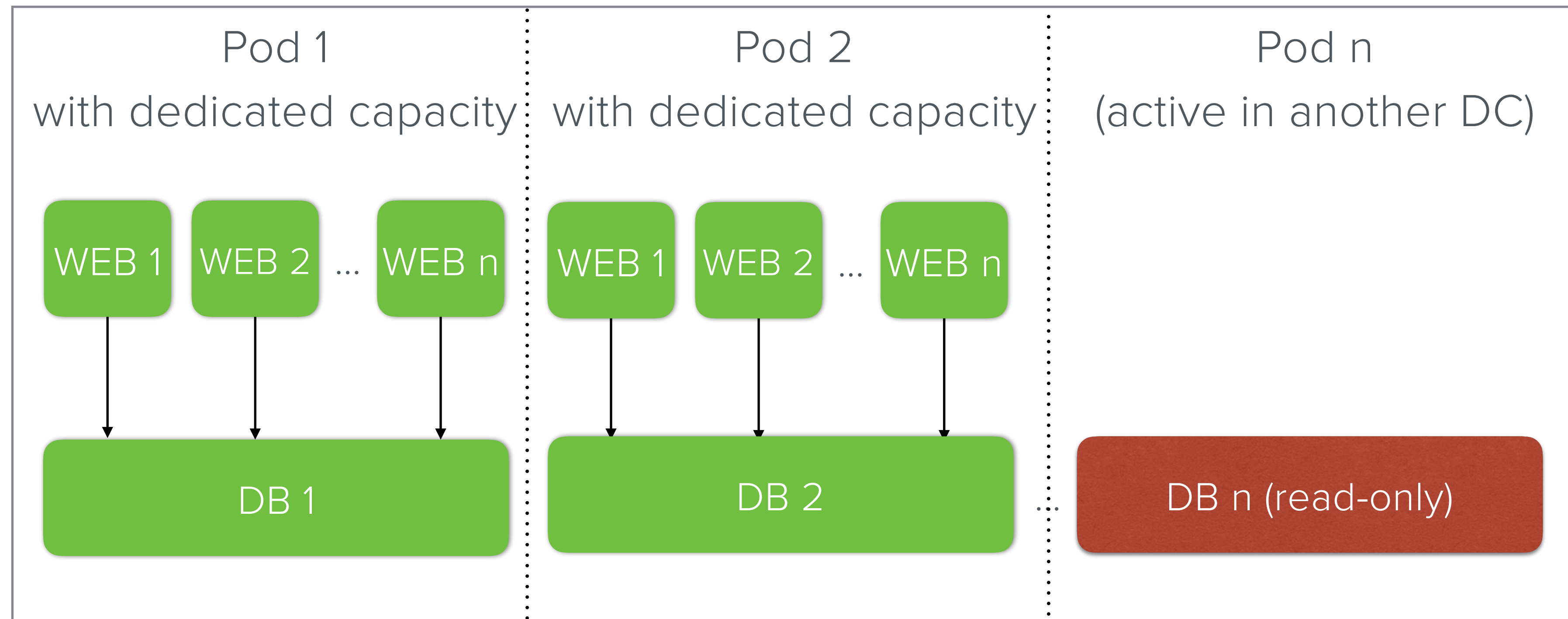


- **Sorting Hat:** Lua application that runs in our nginx load balancer.
- **MySQL:** `domain=bobs-shop.com` → `pod_id=5`
- `ngx.balancer`: API for defining dynamic upstream balancers.
- Other cool stuff: Kafka logger, edge caching, throttling, SSL certs from MySQL, ...

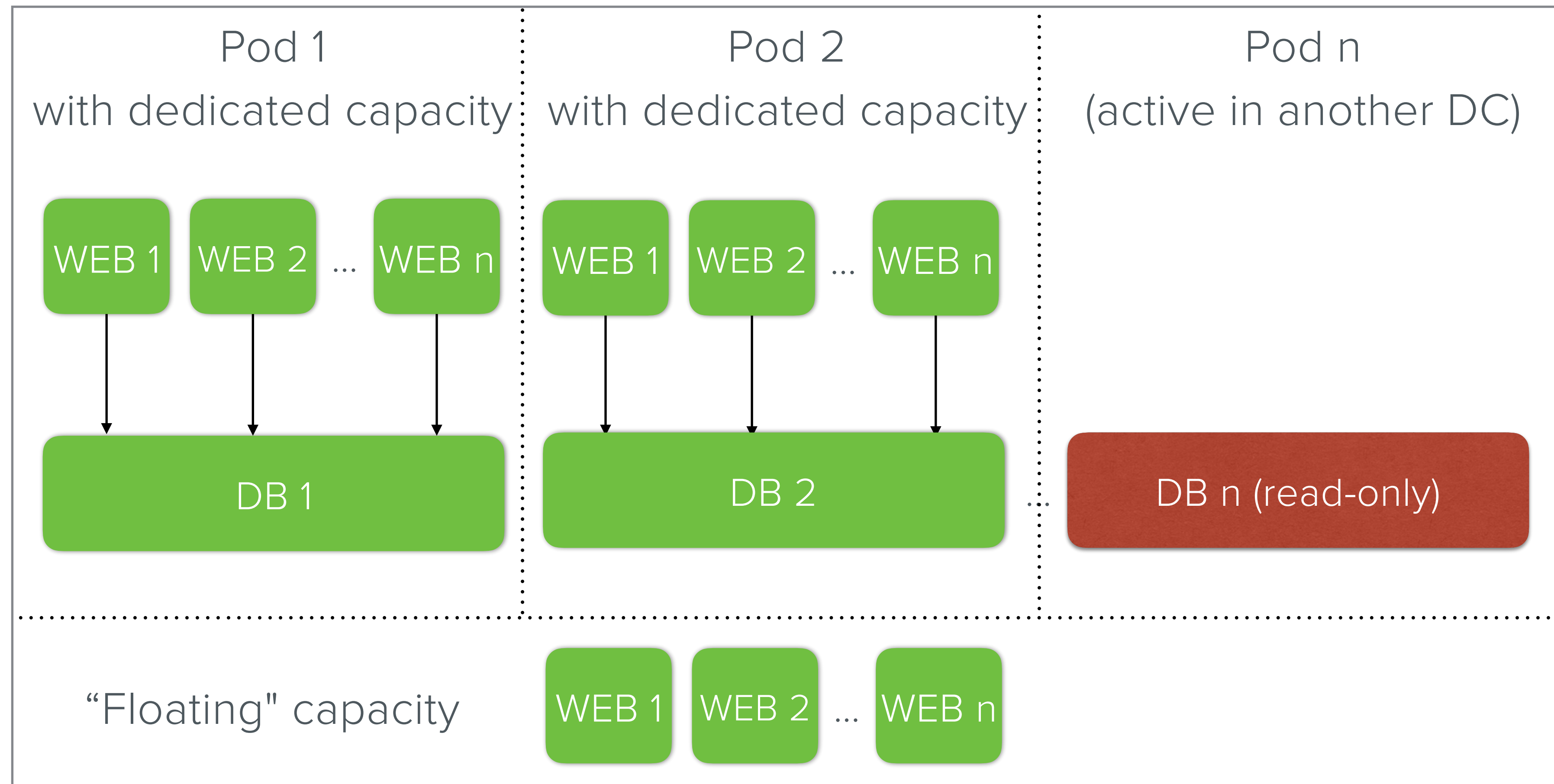
FLOATING CAPACITY

ISOLATION VS. UTILIZATION

Podding



Pods with floating capacity



Multi-tenant architectures

Share nothing	?	Share everything
Little capacity		Huge capacity
Bad utilization		Great utilization
Flash sale problem		Great for flash sales
Crazy expensive		Cheap
Full isolation		No isolation
Horizontal scale is easy		Horizontal scale can be hard

Multi-tenant architectures

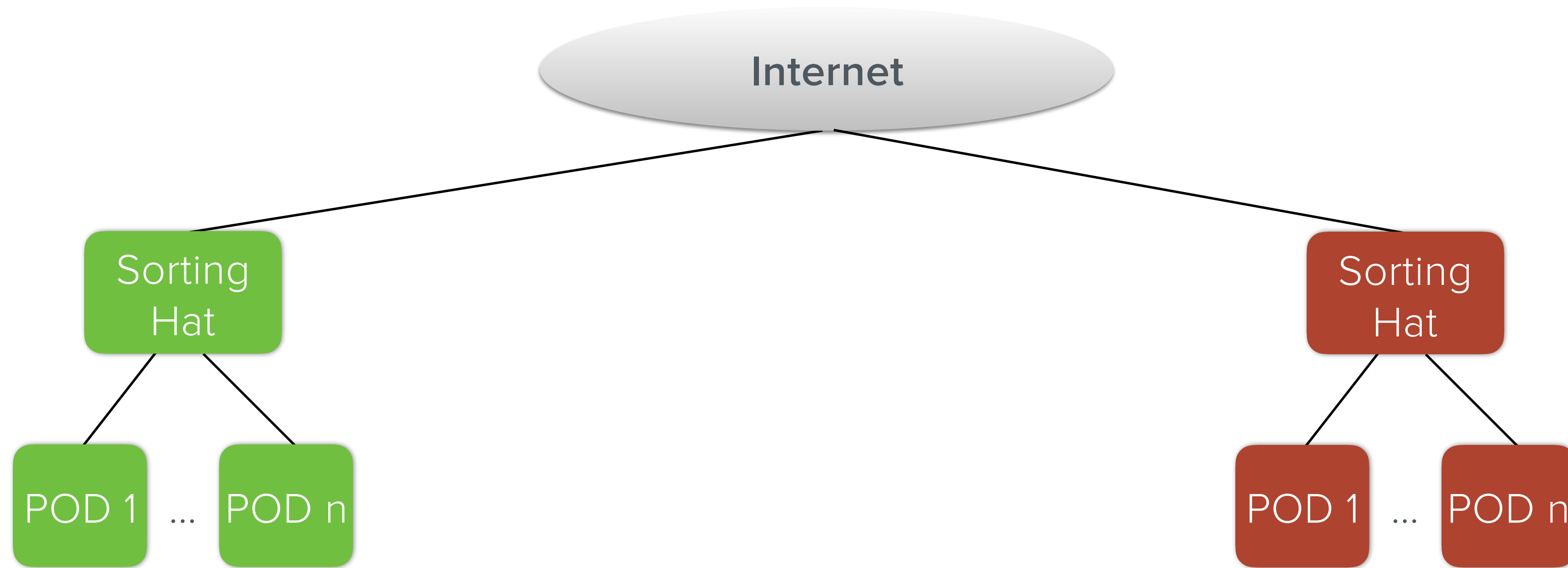
Share nothing	Pods with floating capacity	Share everything
Little capacity	Good capacity	Huge capacity
Bad utilization	Good utilization	Great utilization
Flash sale problem	Great for flash sales	Great for flash sales
Crazy expensive	Cheap	Cheap
Full isolation	Isolated pods	No isolation
Horizontal scale is easy	Horizontal scale is easy	Horizontal scale can be hard

A dimly lit living room with a sofa, a dog, and a fireplace. The scene is dark, with the main focus on the text overlaid on the image. The background shows a living room with a sofa, a dog lying on the floor, and a fireplace with a brick surround. There are plants and a framed picture on the wall.

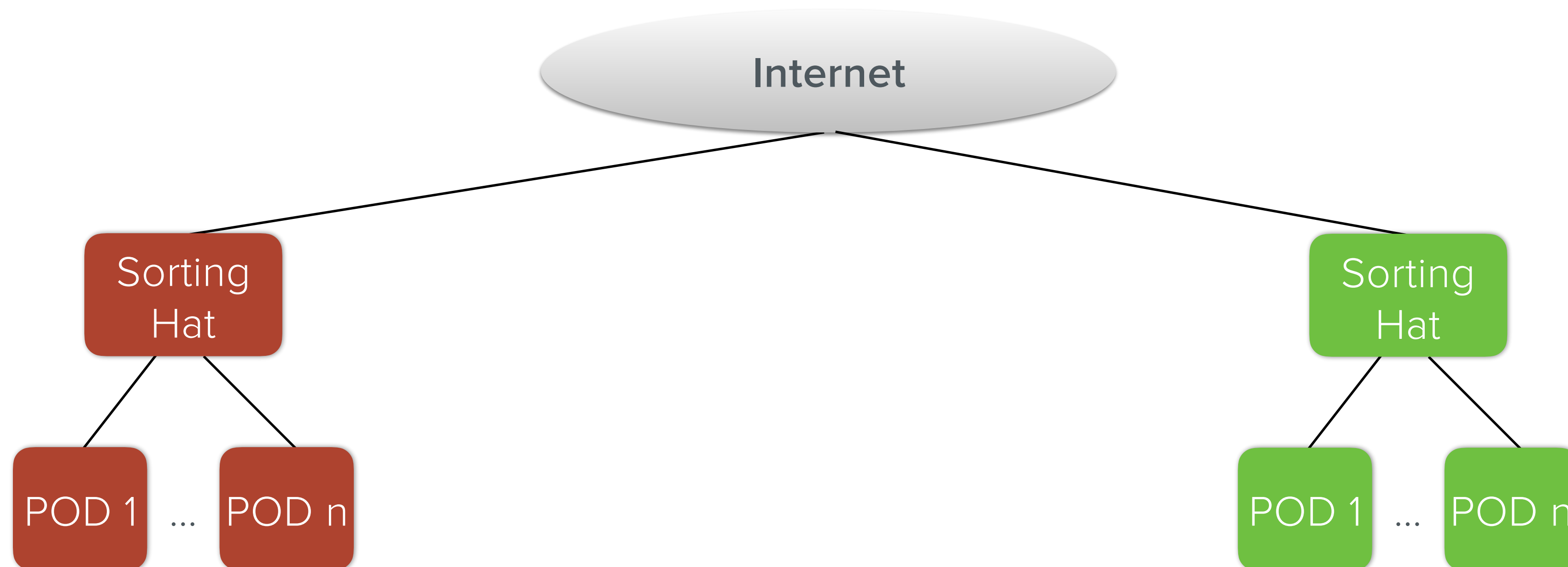
MULTI-DC ROUTING

POINTS OF PRESENCE AND HIGH AVAILABILITY

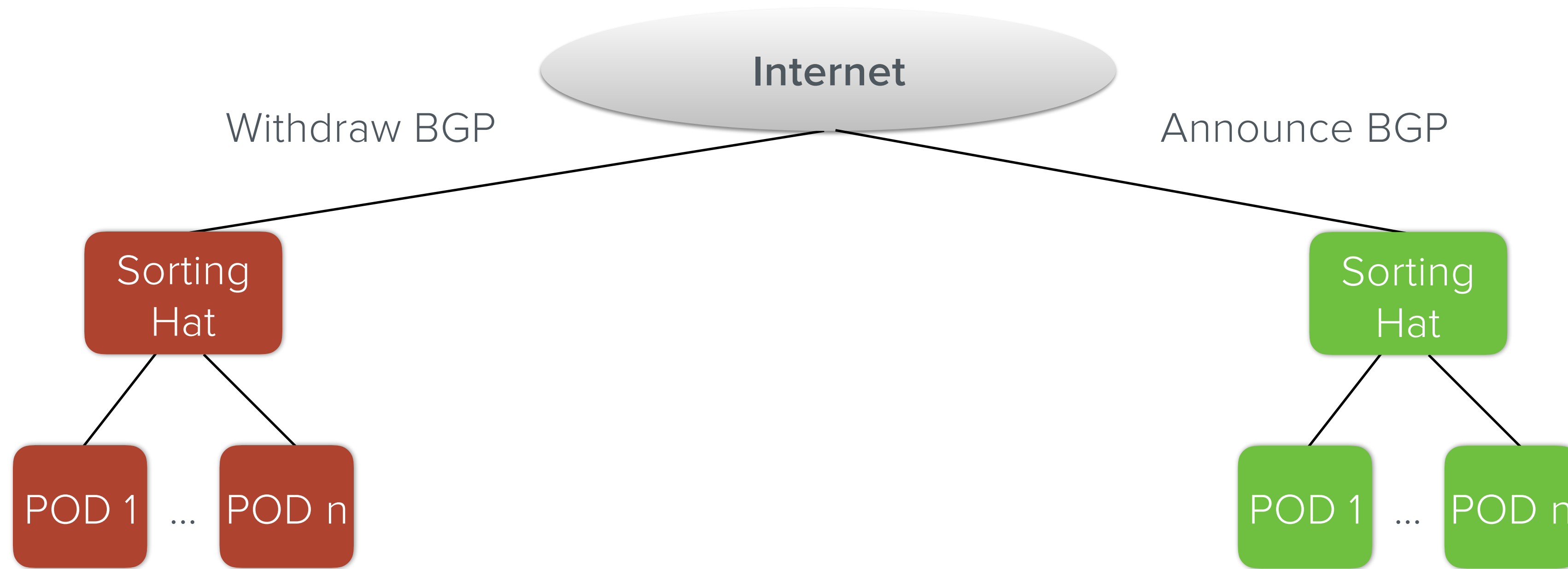
Datacenter failover



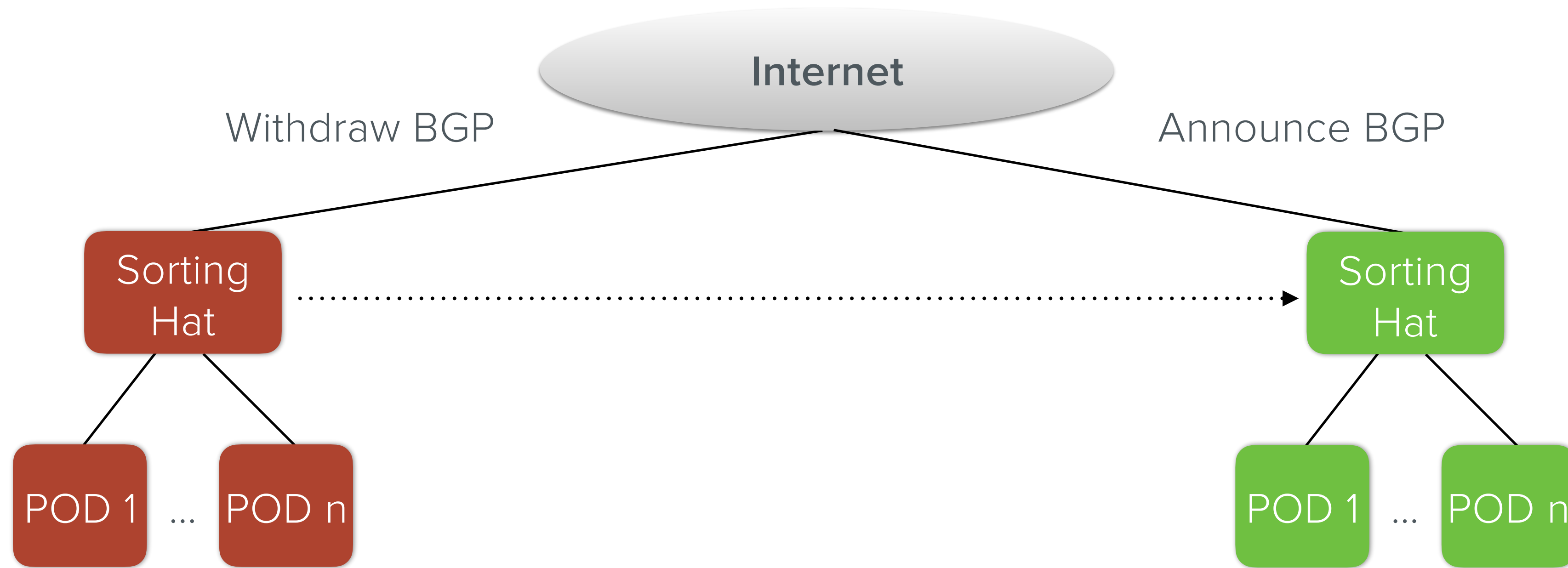
Datacenter failover



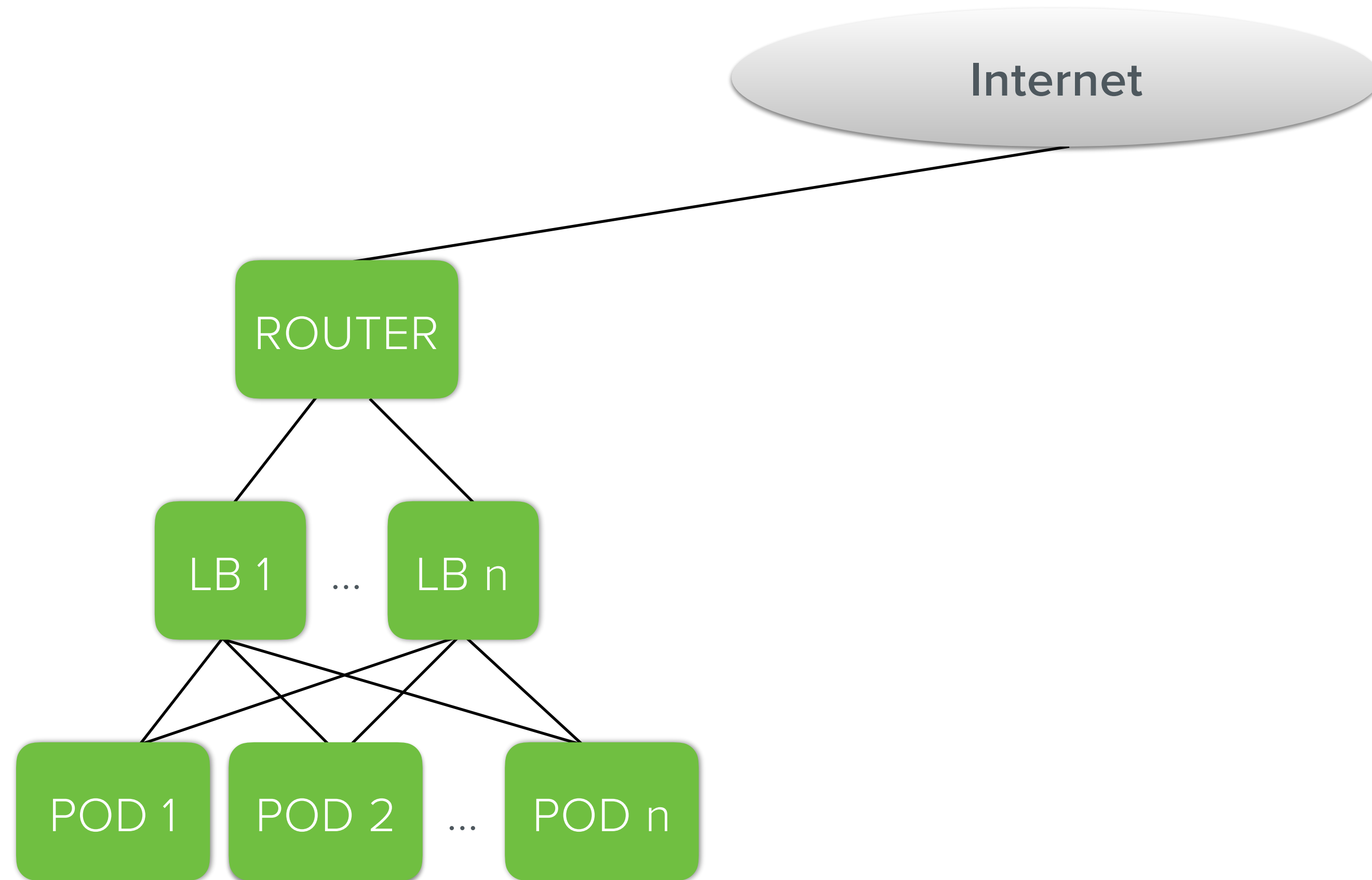
Datacenter failover



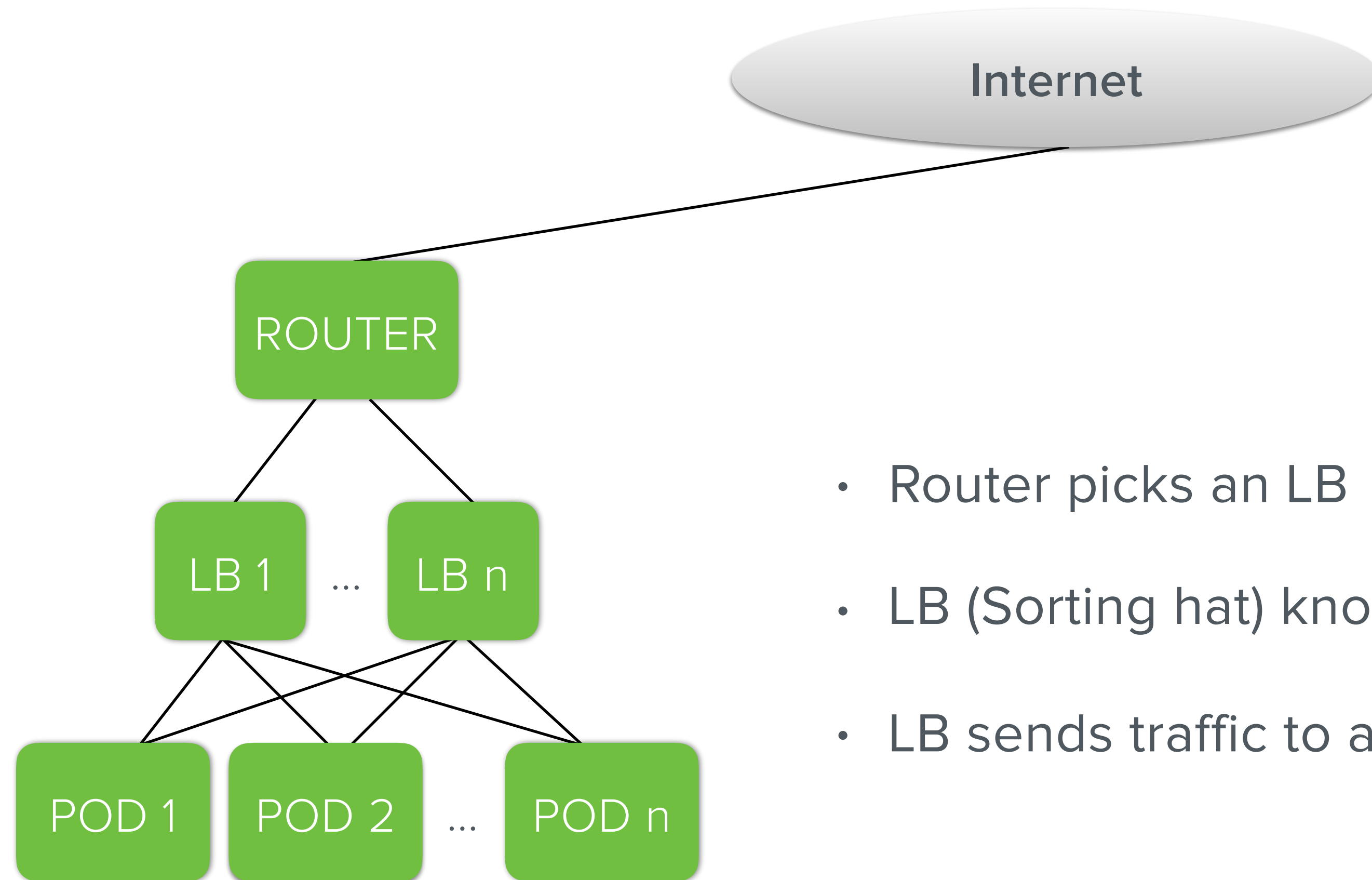
Datacenter failover



Scaling the front door

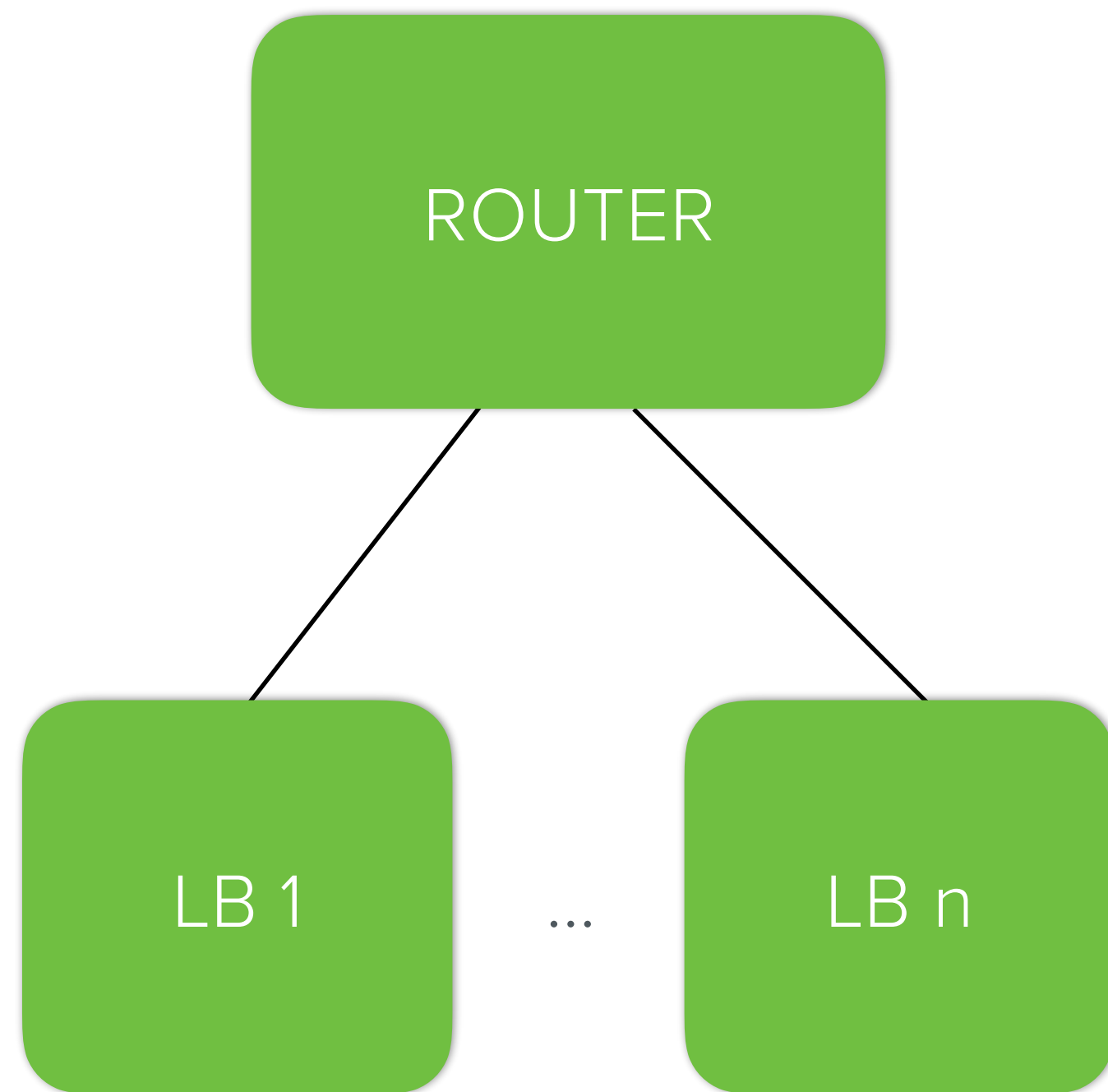


Scaling the front door

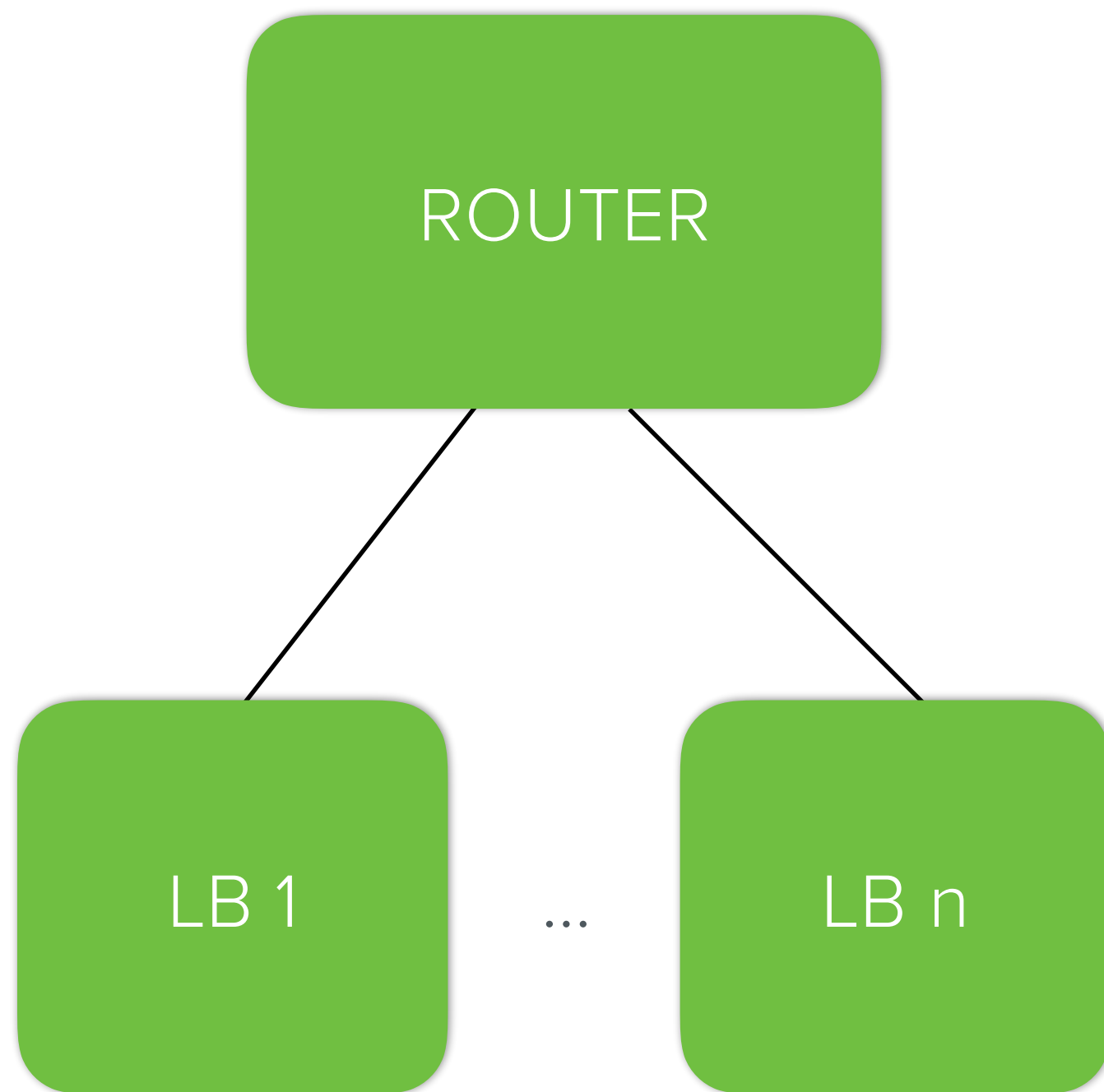


- Router picks an LB
- LB (Sorting hat) knows that `bobs-shop.com` is in pod 2.
- LB sends traffic to a pod 2 upstream.

Load balancing the load balancers

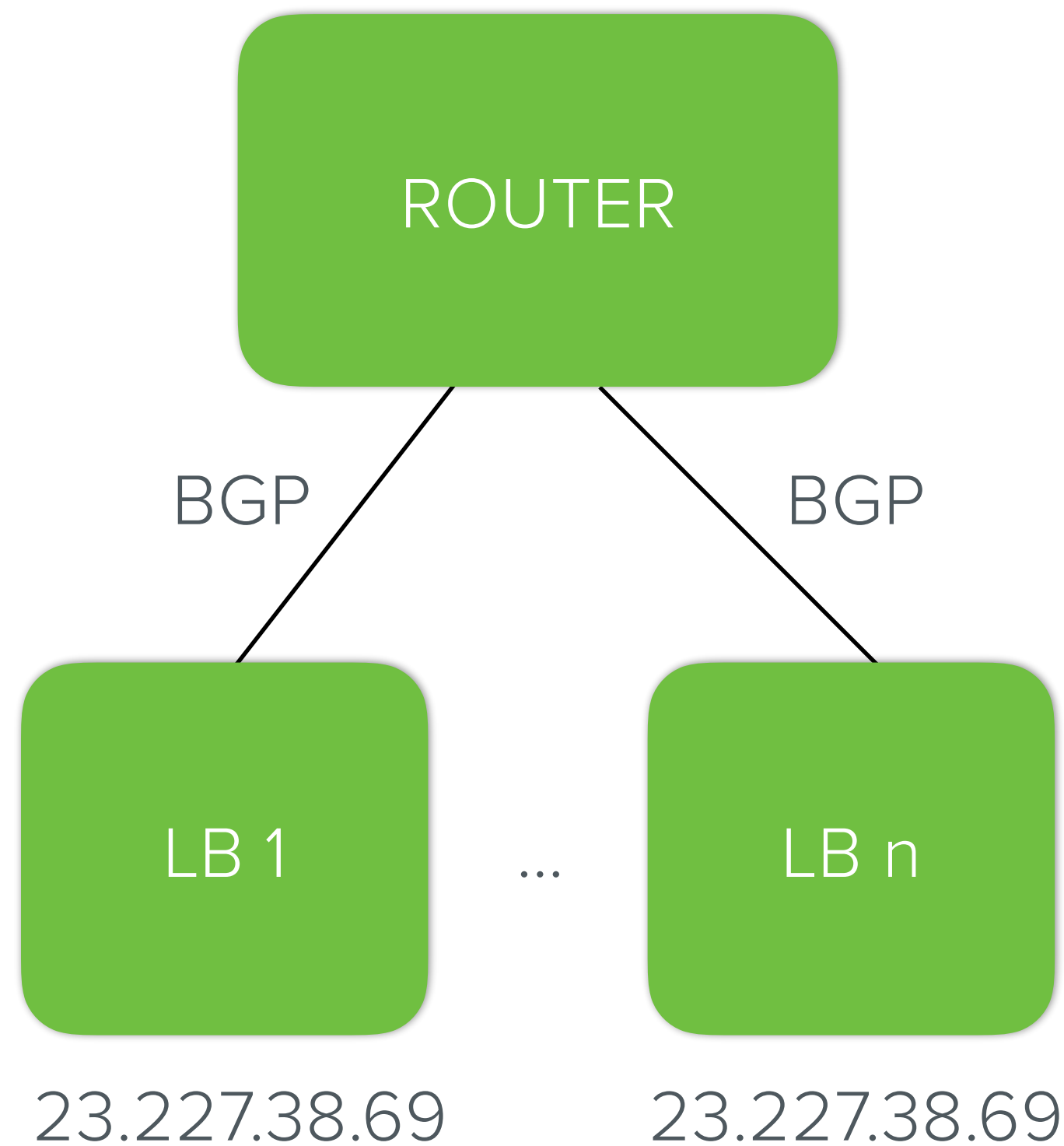


Load balancing the load balancers



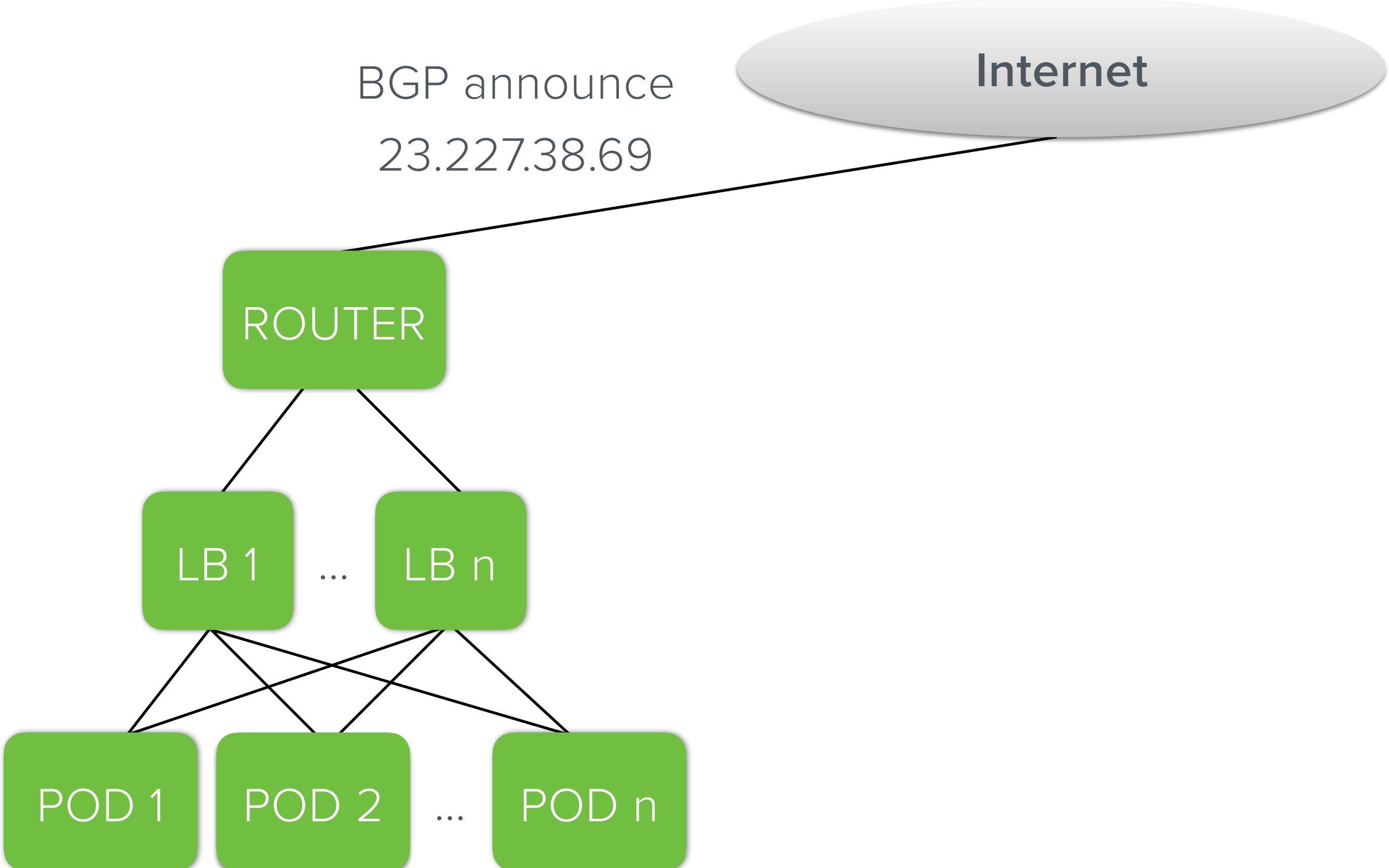
- Multiple LBs for redundancy and load distribution
- How to distribute? Which request goes to which LB?
- Active/backup? One LB per IP?

Load balancing the load balancers



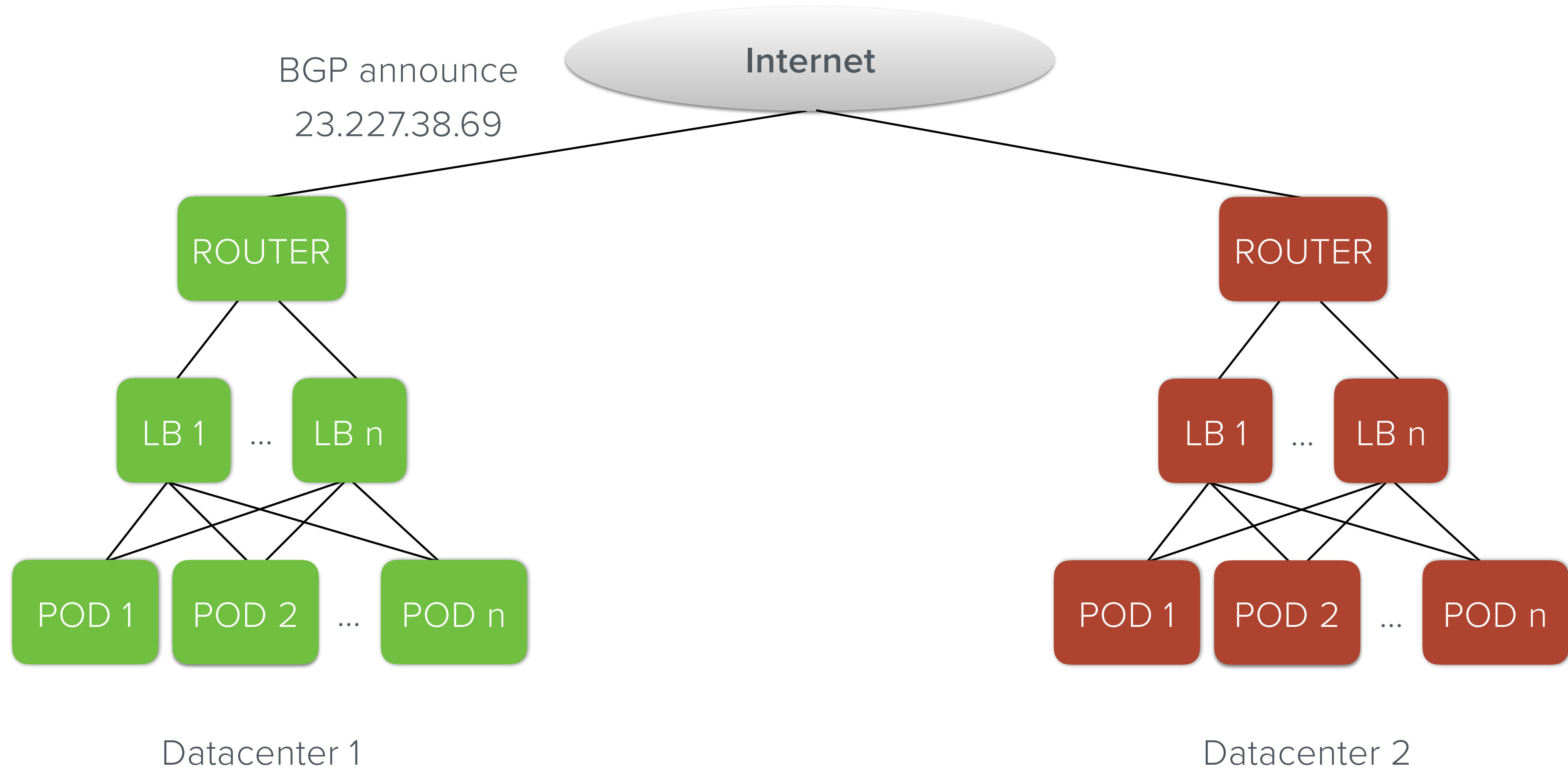
- Multiple LBs for redundancy and load distribution
- How to distribute? Which request goes to which LB?
- Active/backup? One LB per IP?
- **Equal-cost multi-path routing (ECMP)**
- Consistent hashing based on TCP flow
- BGP with health-checks

The front door

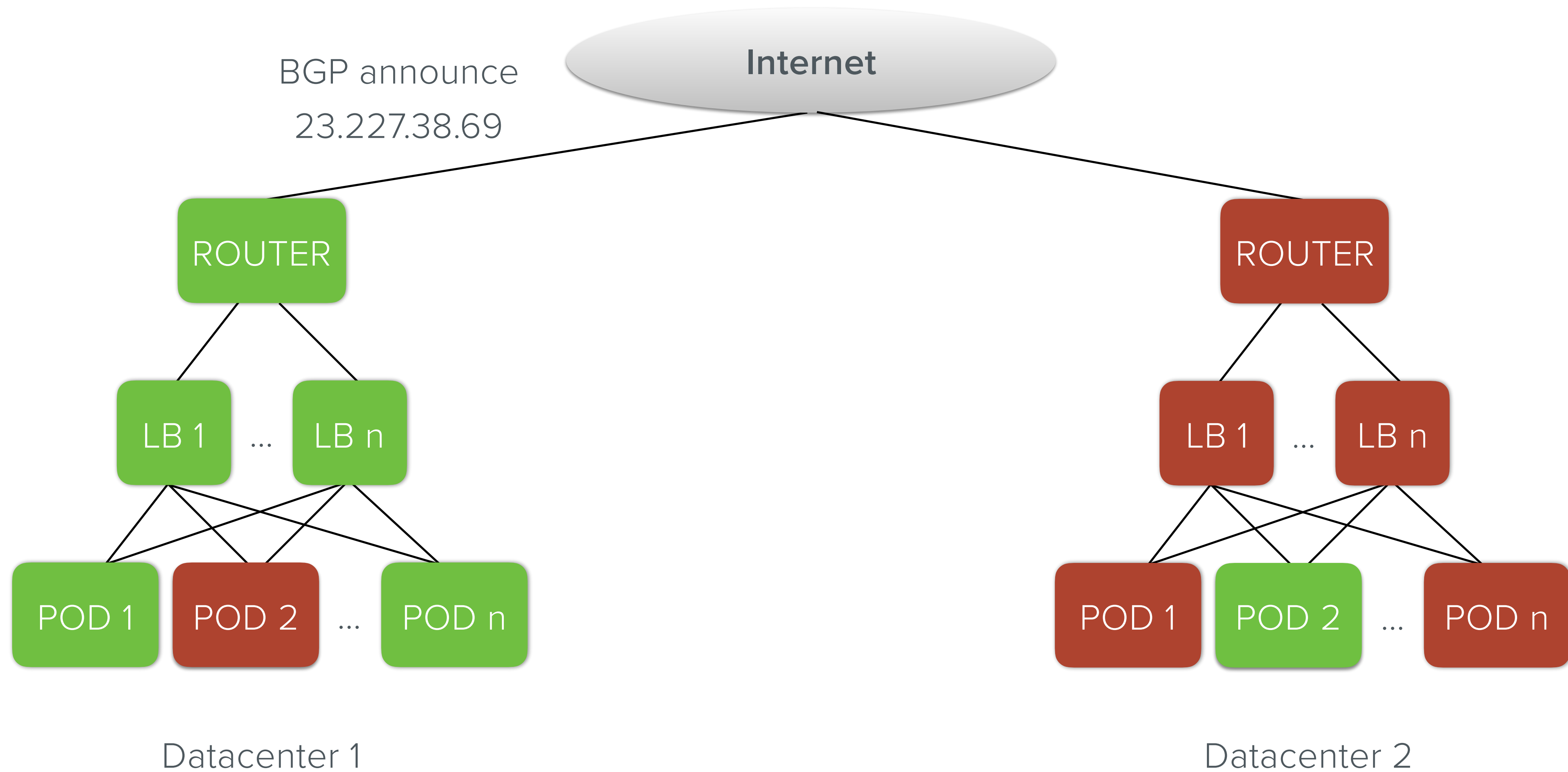


Datacenter 1

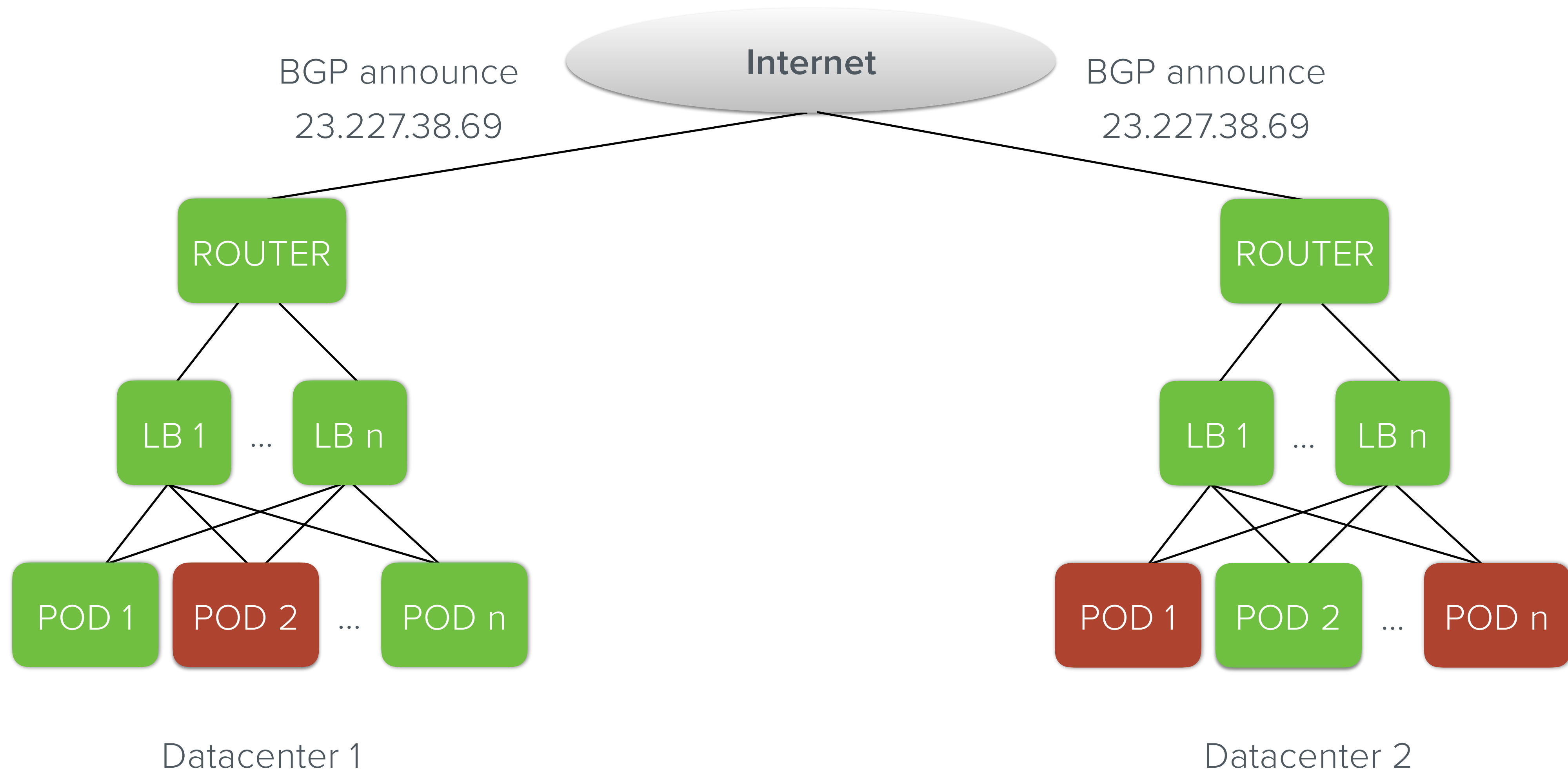
The front door



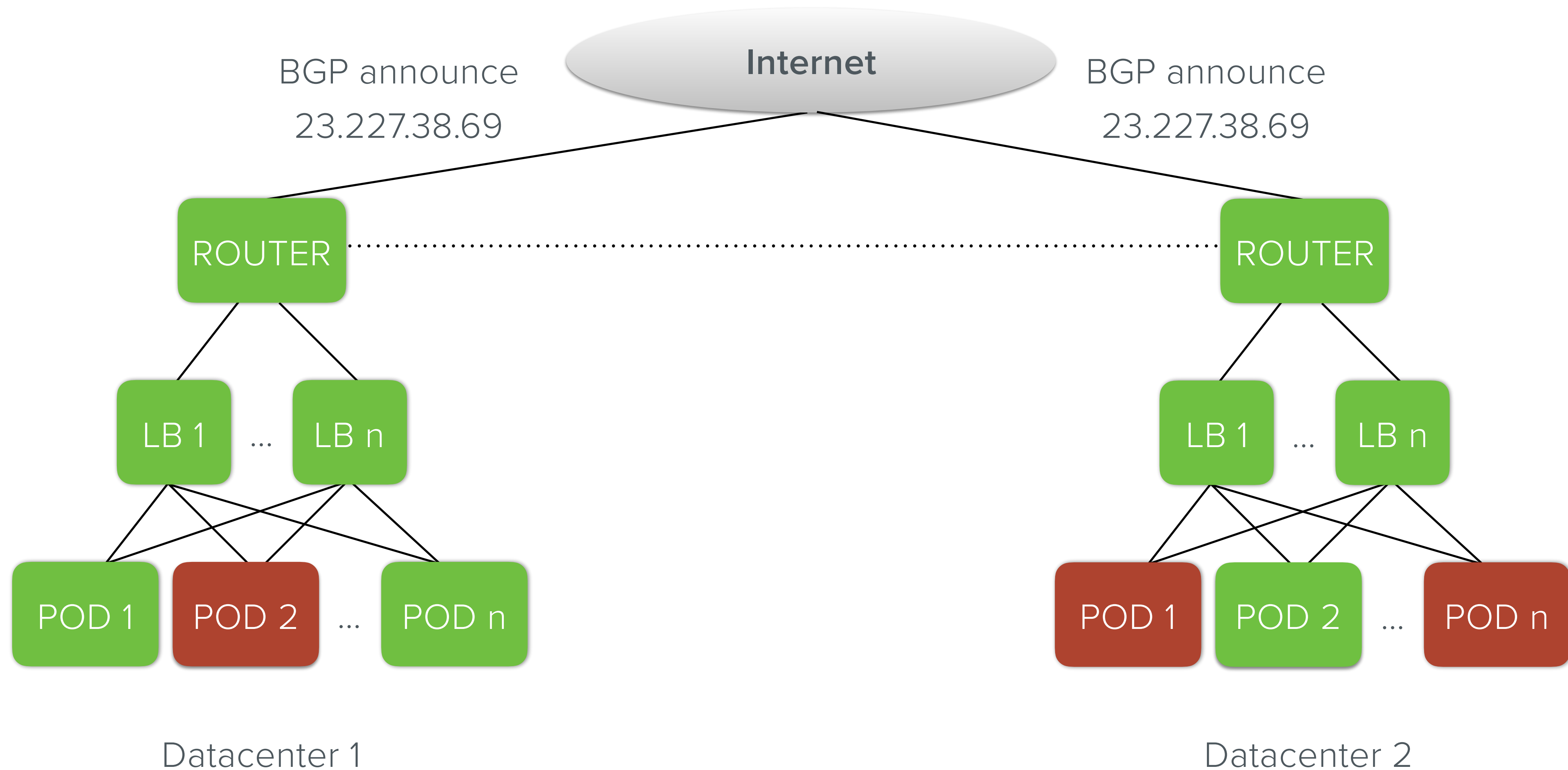
The front door



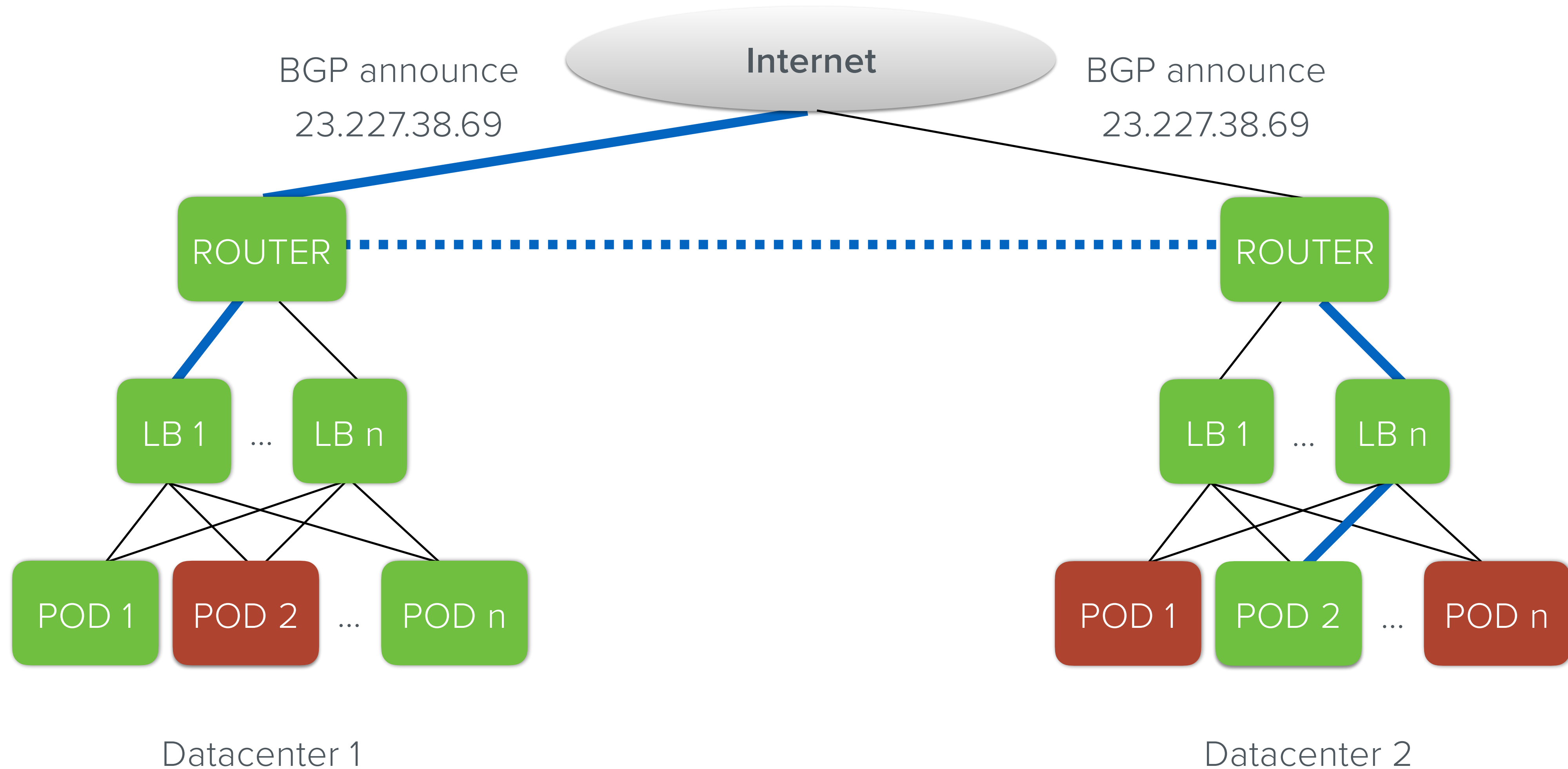
BGP Anycast



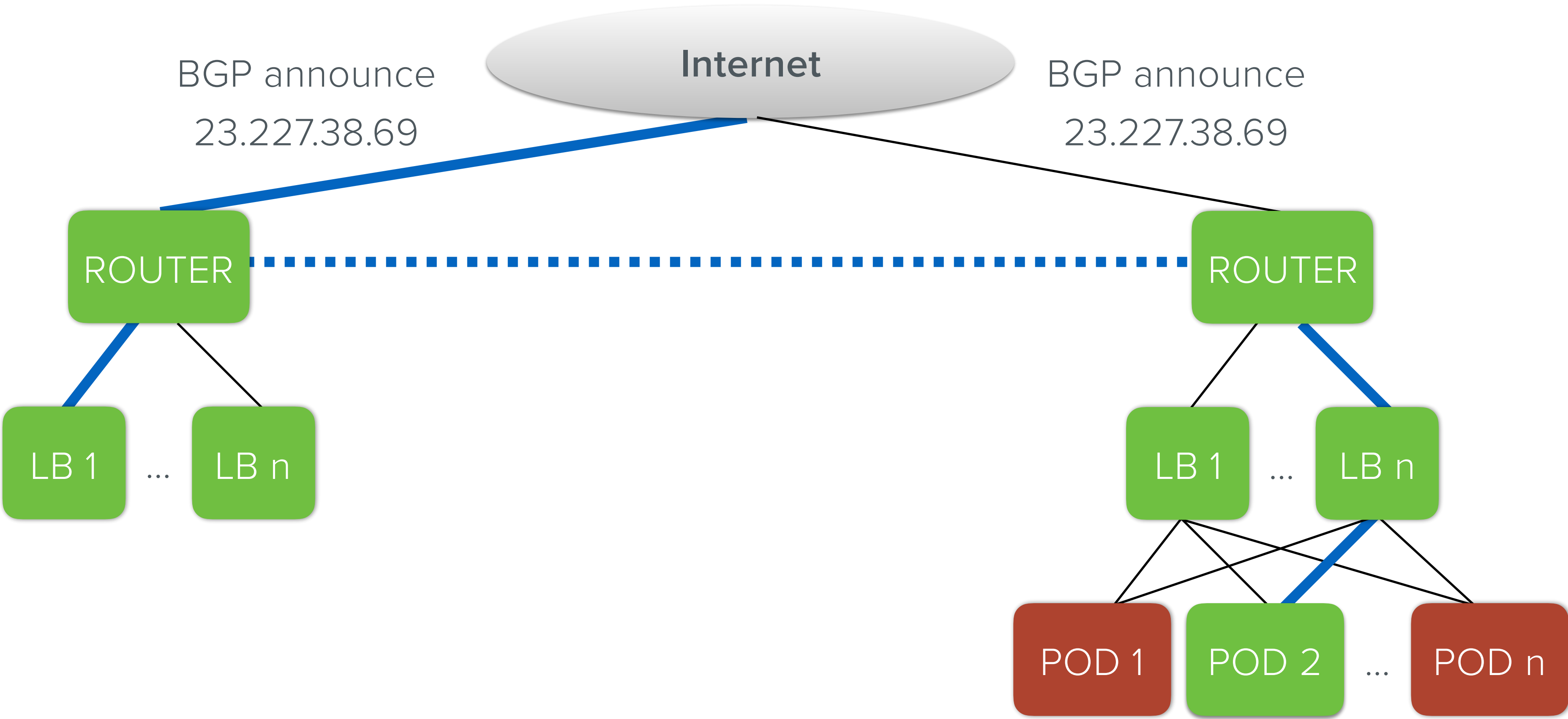
BGP Anycast and Sorting Hat




BGP Anycast and Sorting Hat



Point of presence



A dimly lit living room with a large window, a sofa, and a coffee table. The room is dark, with the primary light source being the window, which shows a view of trees and a building. The furniture is dark and blends into the background. The text 'TL;DR' is overlaid in the center in a bright, white, sans-serif font.

TL;DR

SUMMARY AND KEY TAKEAWAYS

Isolation vs. capacity

Spectrum of multi-tenant architectures

Share nothing



2004

Share everything



2005-2012

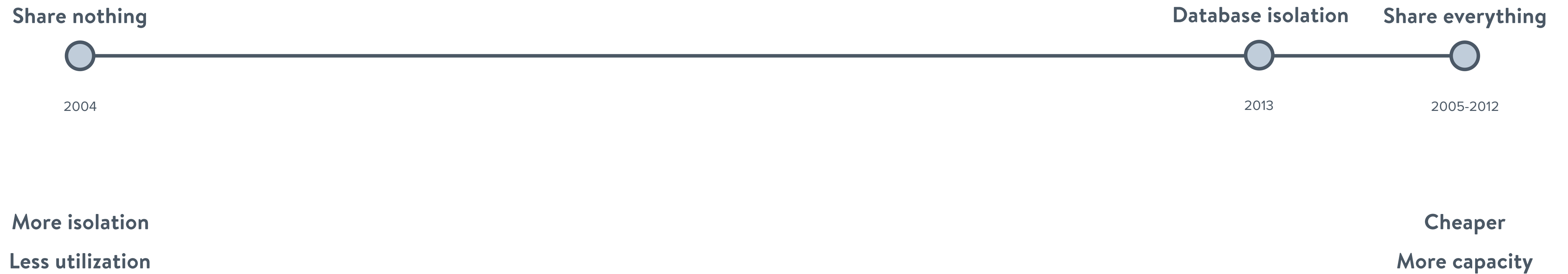
More isolation

Less utilization

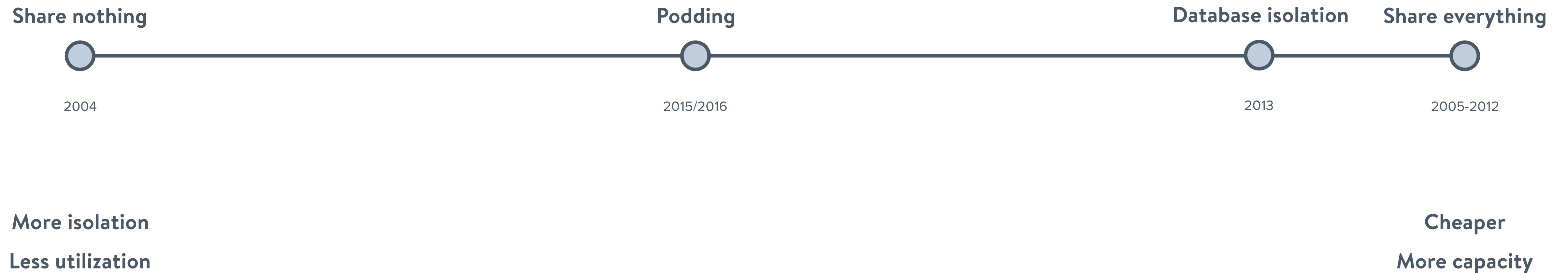
Cheaper

More capacity

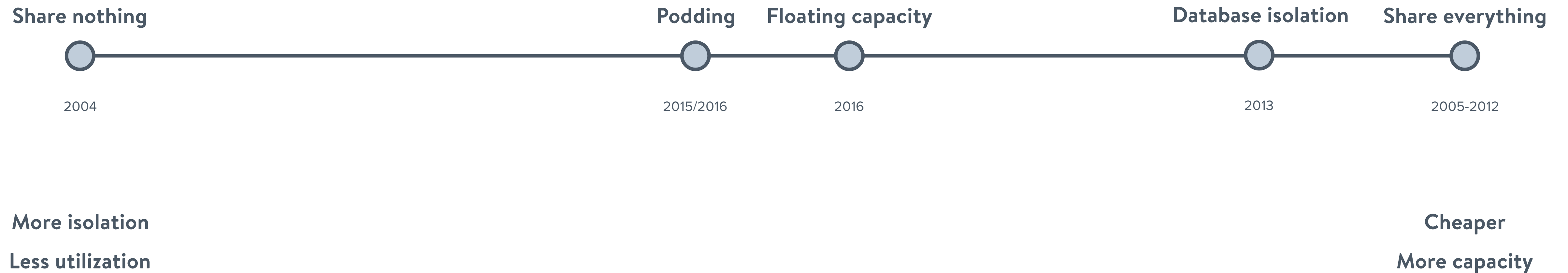
Spectrum of multi-tenant architectures



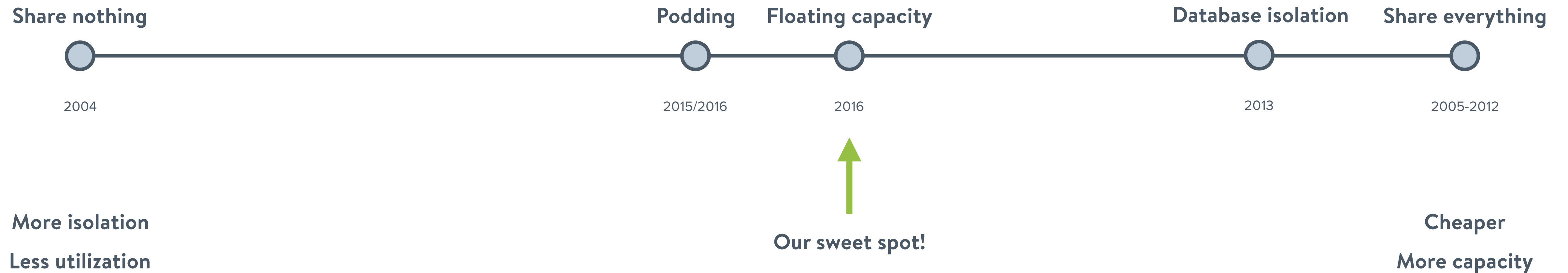
Spectrum of multi-tenant architectures



Spectrum of multi-tenant architectures



Spectrum of multi-tenant architectures



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`

