# Make Haste Slowly

## Balancing SRE Diligence in Urgency Driven Organizations

Jason Hiltz-Laforge

shopify

# WTF?

- Production Engineering Lead @ Shopify in 
- I manage both systems and humans

*How fast can the humans go without blowing up the systems too often?*



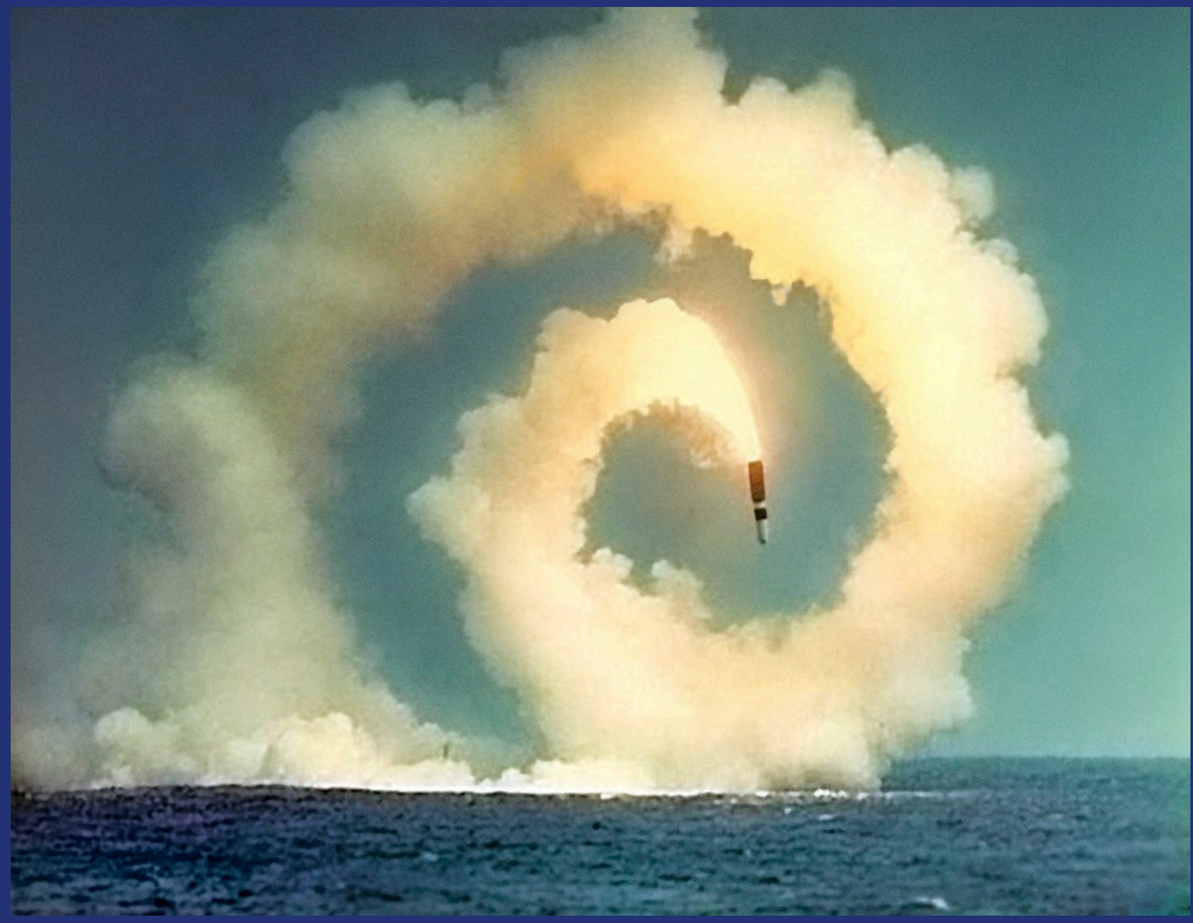

# *Festina Lente*
# (Make Haste Slowly)

**“… activities should be performed with a proper balance of urgency and diligence. If tasks are overly rushed, mistakes are made and good long-term results are not achieved.”**

- “Festina Lente”, *Wikipedia (https://en.wikipedia.org/wiki/Festina_lente)*




VS



# Disaster Matrix

| Database | Impact | Likelihood |
| --- | --- | --- |
| Drop Table | High | Low |
| Drop Permissions | Medium | Medium |
| Writer Failure | Medium | Low |
| Backup Corruption | High | High |

# Disaster(less) Matrix

| Database | Impact | Likelihood |
| --- | --- | --- |
| Drop Table | High | Low |
| Drop Permissions | Low | Medium |
| Writer Failure | Medium | Low |
| Backup Corruption | Medium | Low |

# Disaster(less) Matrix over Time

| Database | Impact | L |
| --- | --- | --- |
| Drop Table | High | |
| Drop Permissions | Low | M |
| Writer Failure | Medium | L |
| Backup Corruption | Medium | Low |

Jason Hiltz-Laforge

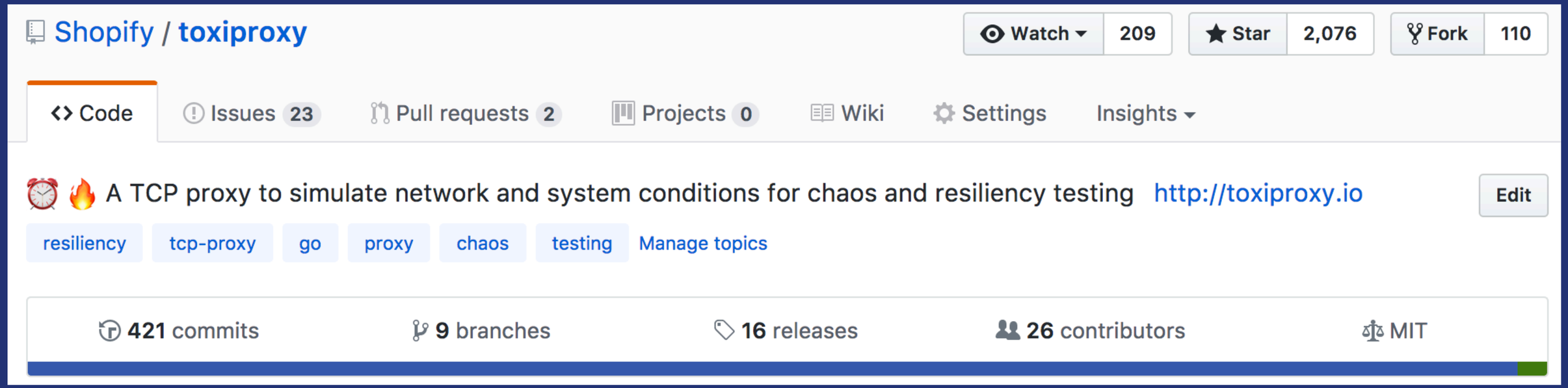
Project Alpha targeted to reduce this to Medium impact in Q3.

Done

# Availability Matrix

| Unavailable | Storefront | Admin | API |
|---|---|---|---|
| **Primary MySQL** | Down | Degraded | Down |
| **Redis** | Up | Down | Up |
| **Elasticsearch** | Degraded | Degraded | Up |
| **Kafka** | Up | Degraded | Degraded |

# Availability Testing



# Availability Testing

For example, to add 1000ms of latency to the response of MySQL from the Ruby client:

```ruby
Toxiproxy[:mysql_master].downstream(:latency, latency: 1000).apply do
  Shop.first # this takes at least 1s
end
```

To take down all Redis instances:

```ruby
Toxiproxy[/redis/].down do
  Shop.first # this will throw an exception
end
```

## Team Speed

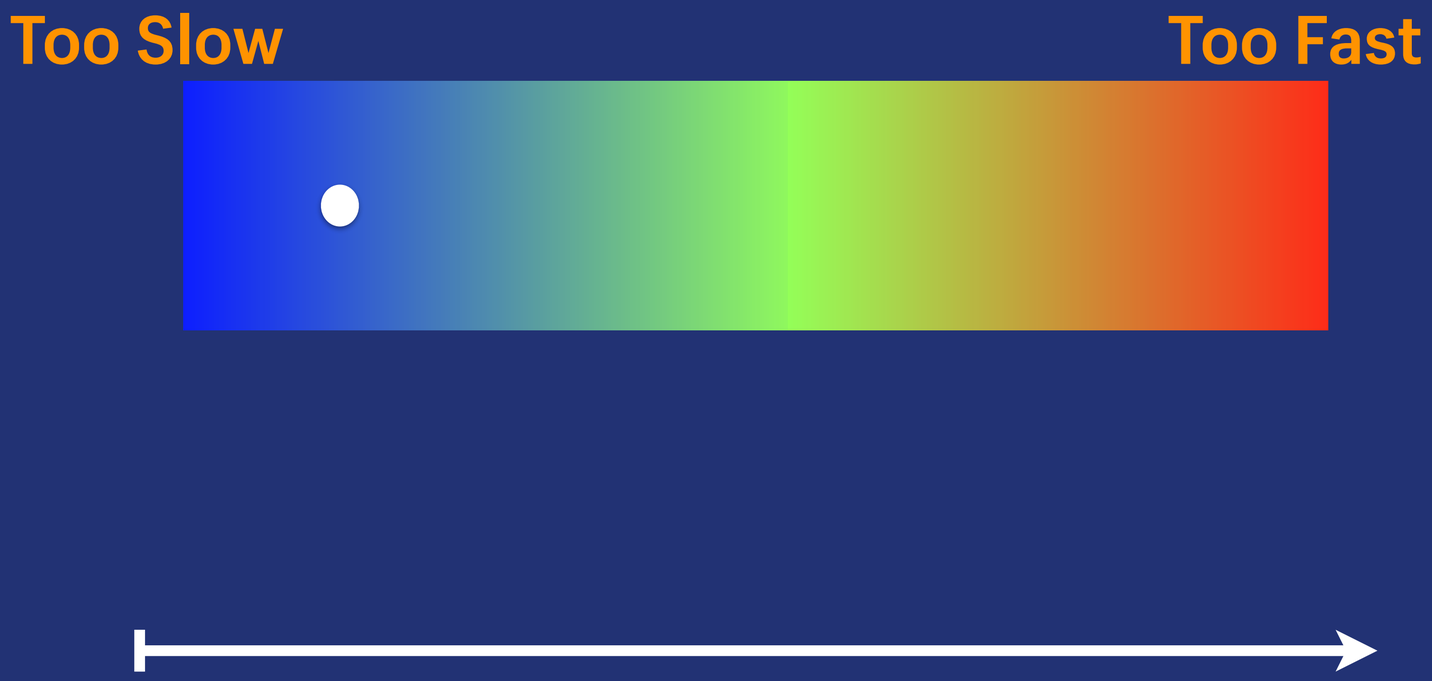
Too Slow

Too Fast

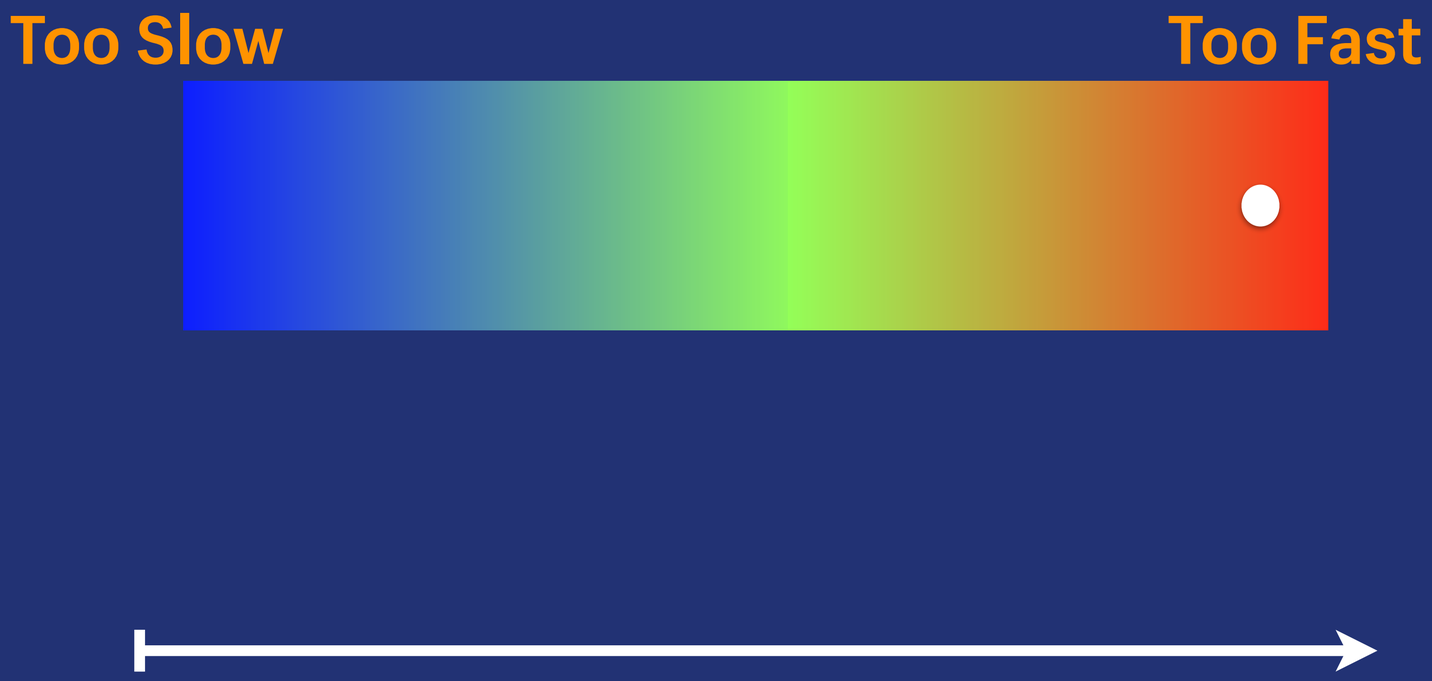


Speed of Iteration

# Not Enough Haste



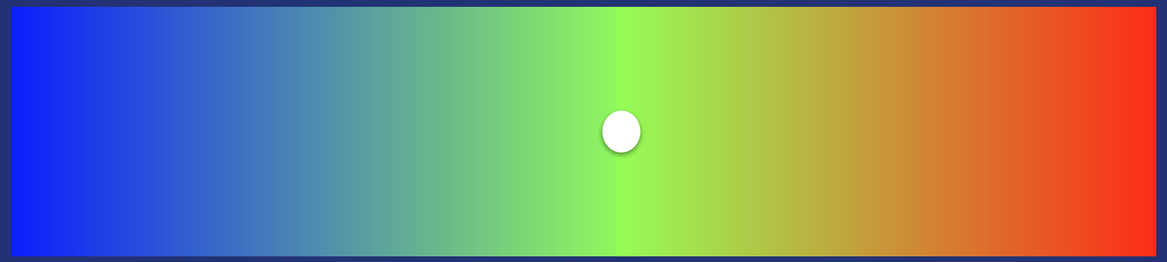
## Not Enough Diligence



## Just Right

Too Slow

Too Fast



*Festina Lente*



Speed of Iteration

# Where’s the Dot?

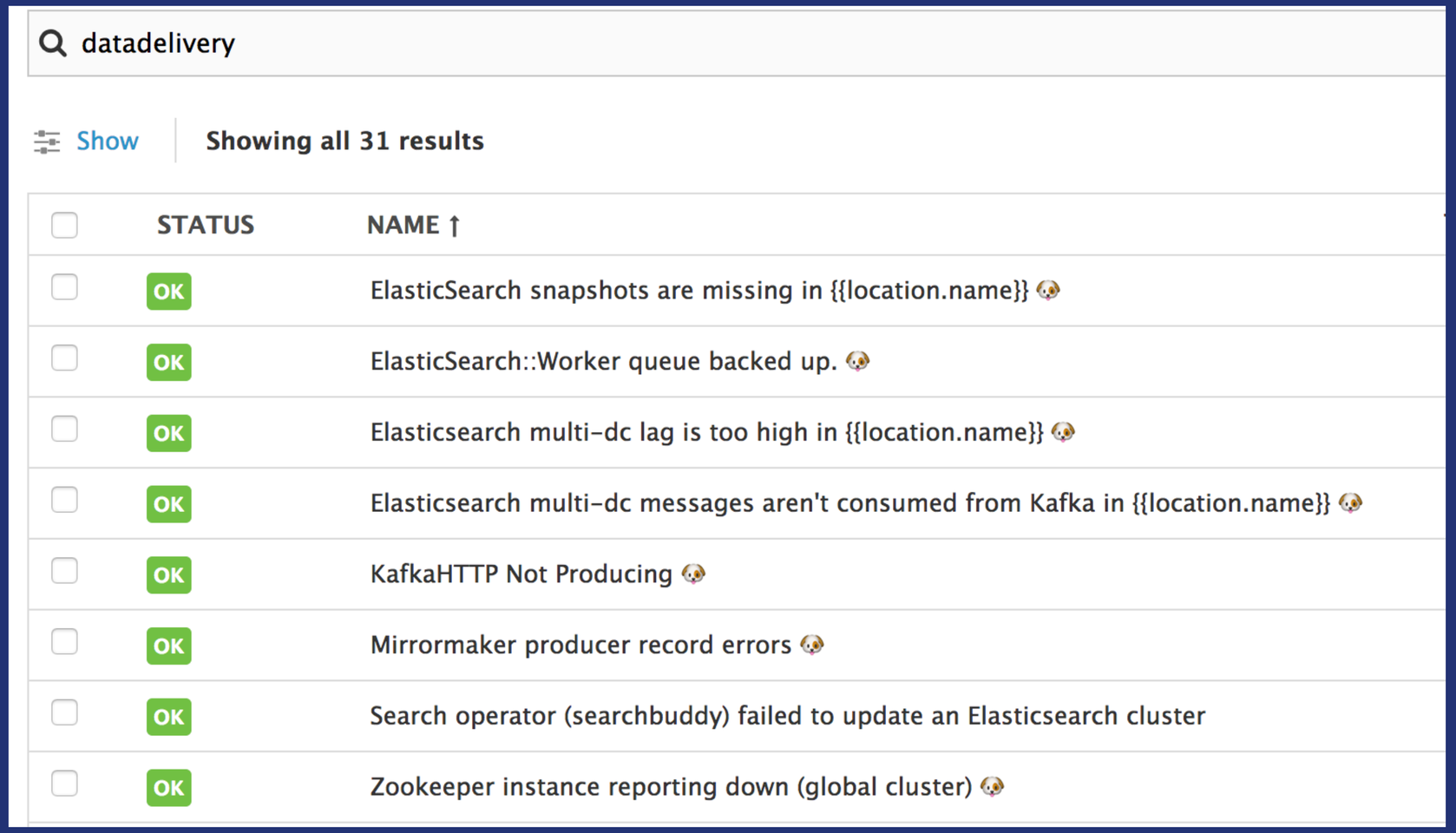
- Disaster Matrix
- Availability Testing
- Error Budgets
- Outage reports

*When the systems do blow up,*
*how do you deal with the humans?*

# Control over Your Environment



# Alerts and Pages



# Failure is OK



**Simon Eskildsen** 10:48 AM

@here Just wanted to recognize some amazing work by @zen and @emil, as we've reached a major milestone. Over the past couple of months, @zen has worked on building more and more confidence in pod failovers. Quite a few people in Production Engineering have now tried to do a failover of pod 1, and a respectable number even production failovers. The dream has always been that we could do `spy failover`, and as of yesterday, we can actually failover a Shopify pod with a single spy command. Check this out:

9 7 3

spy failover shopify pod 1 to chi2

**spy** APP 10:48 AM

ShipitAPI::Error: Unexpected error: 409

6 3 4 3 1

# Root Cause Analysis

Timeline Review (5 minutes)

Good, Bad, and Ugly (15 minutes)

Five Whys (15 minutes)

Action Items (15 minutes)

*How fast can the humans go without blowing up the systems too often?*

*Only as fast as their cost of failure lets them*

*When the systems do blow up, how do you deal with the humans?*

*By giving them control and safety*

# Shopify Talks

**Wednesday**

Standing On the Shoulders of Giants: Unleashing the Power of Scriptable Load Balancers
1:40 pm to 2:30 pm - Emil Stolarsky

Bots Are Fast, Humans Are Smarter—Eliminate Unwanted Traffic and Defend Against DDoS
3:40 pm to 4:10 pm - Felix Glaser

**Thursday**

Resiliency Testing with Toxiproxy
9:50 am to 10:20 am - Jake Pittis

Incident Management and Chatops at Shopify
4:30 pm to 5:00 pm - Daniella Niyonkuru

Six Ways a Culture of Communication Strengthens Your Team's Resiliency (Lightning Talk)
5:00 pm to 6:00 pm - Jaime Woo

**Friday**

Building an On-Premise Kubernetes Cluster For a Large Web Application
11:30 am to 12:00 pm - Daniel Turner

# Thanks!

jason.hl@shopify.com
https://engineering.shopify.com/
@shopifyeng

shopify