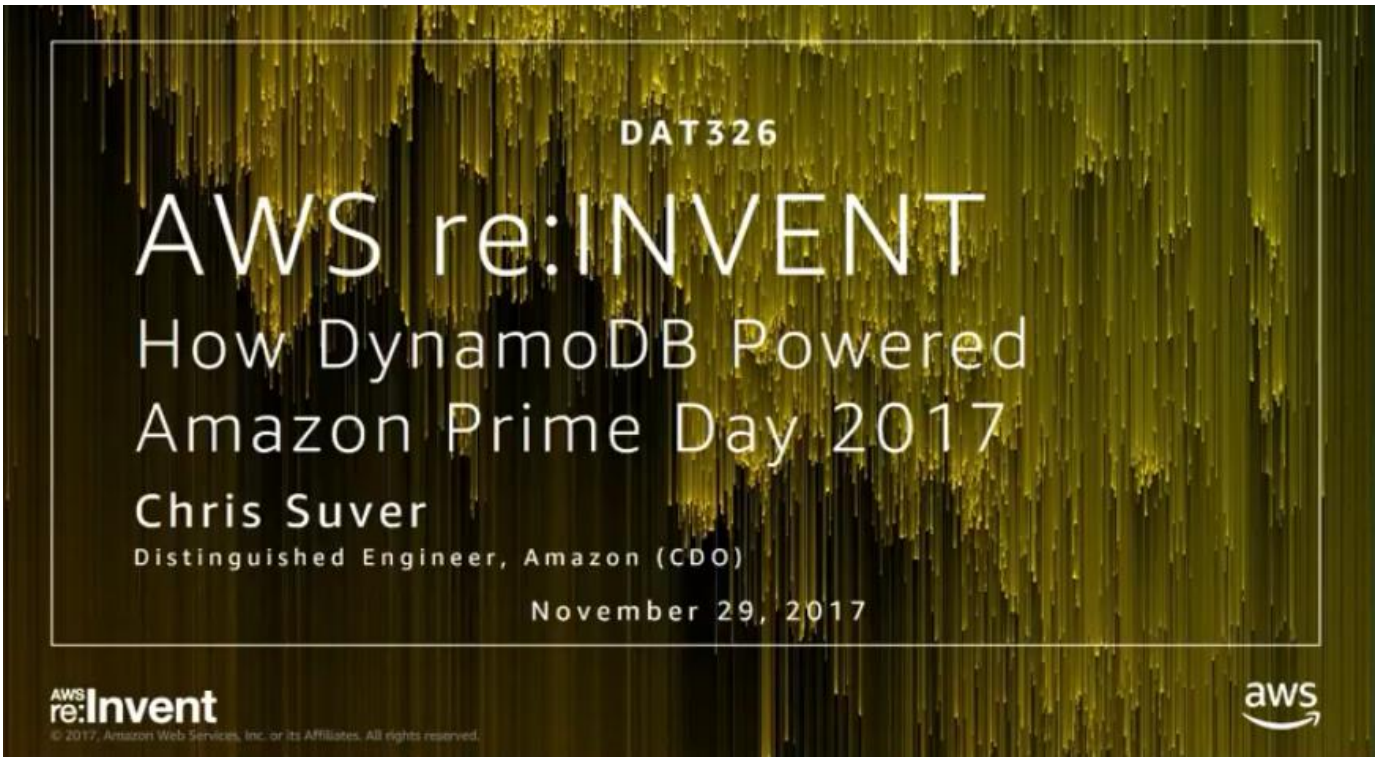
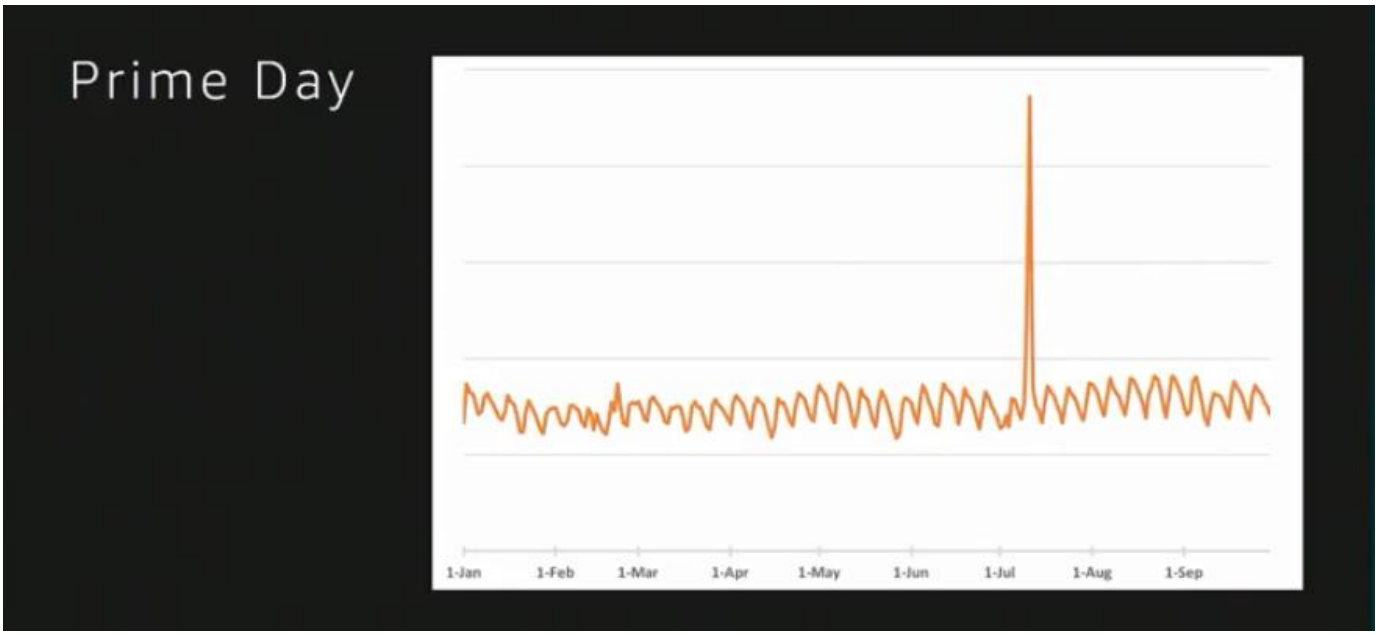
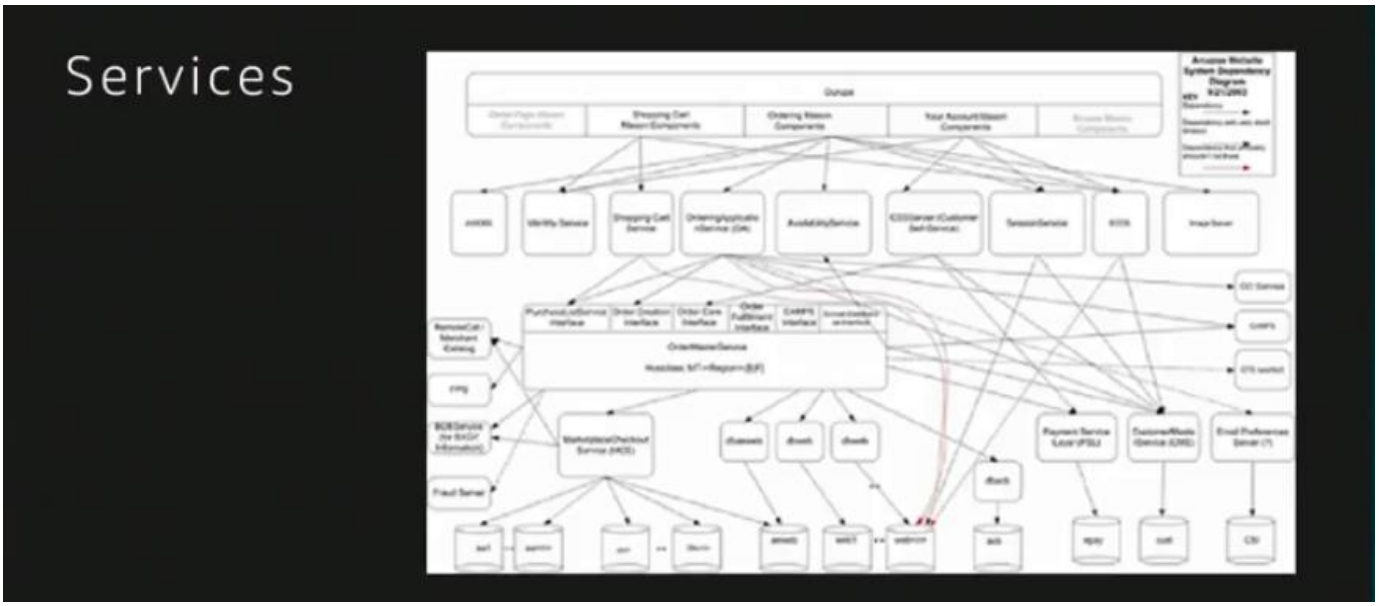
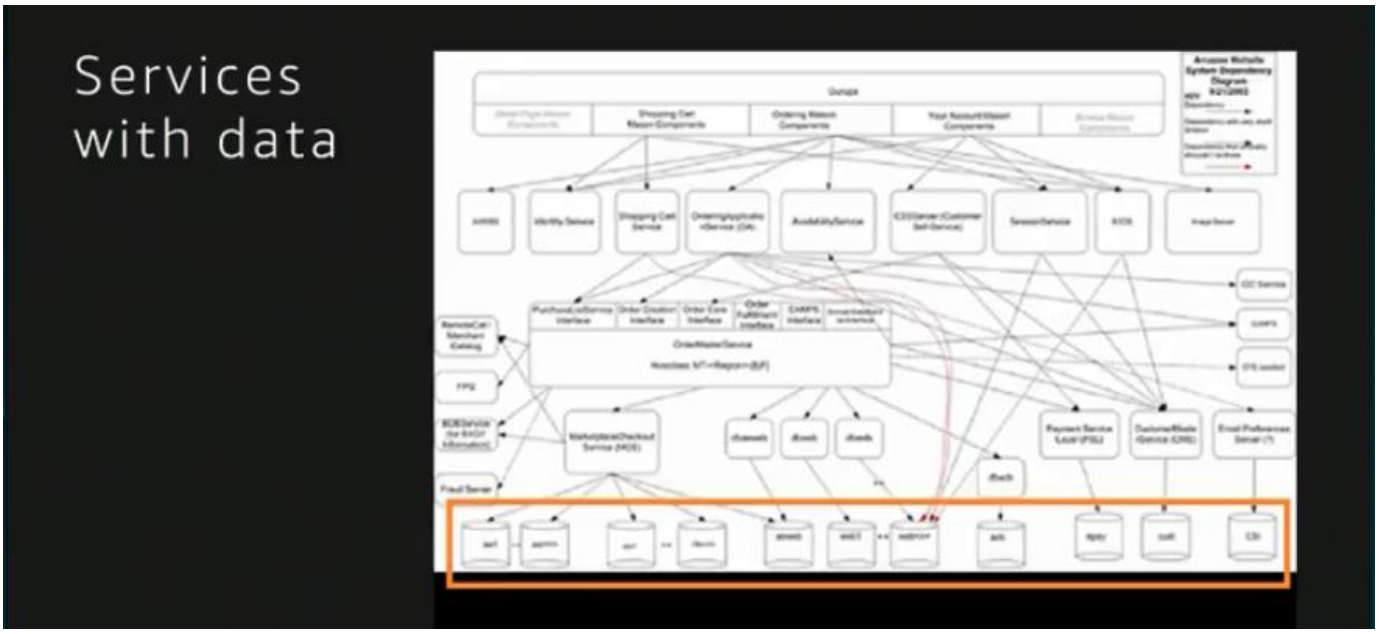



Sales on Prime Day 2017 surpassed Black Friday and Cyber Monday, making it the biggest day ever in Amazon history. An event of this scale requires infrastructure that can easily scale to match the surge in traffic. In this session, learn how AWS and Amazon DynamoDB powered Prime Day 2017. DynamoDB requests from Amazon Alexa, the Amazon.com sites, and the Amazon fulfillment centers peaked at 12.9 million per second, a total of 3.34 trillion requests. Learn how the extreme scale, consistent performance, and high availability of DynamoDB let Amazon.com meet the needs of Prime Day without breaking a sweat.

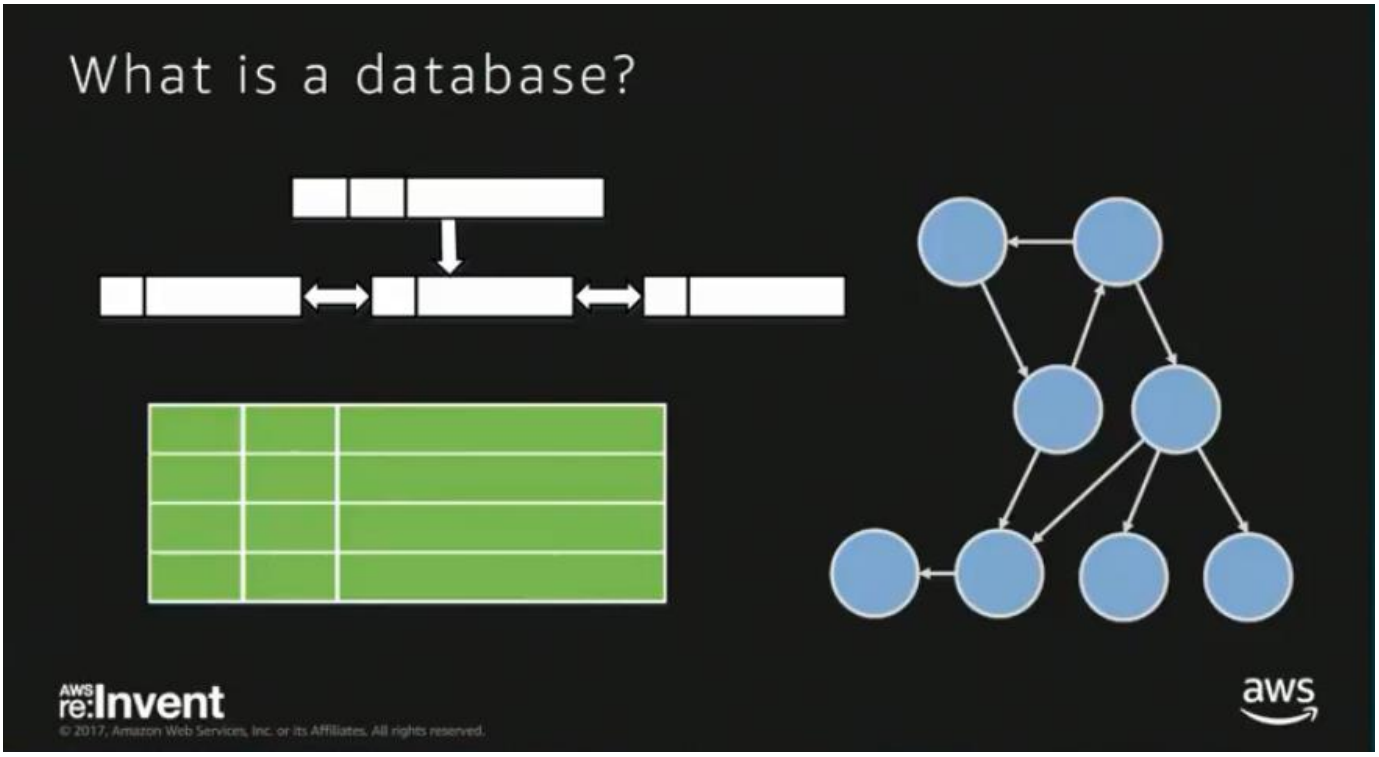


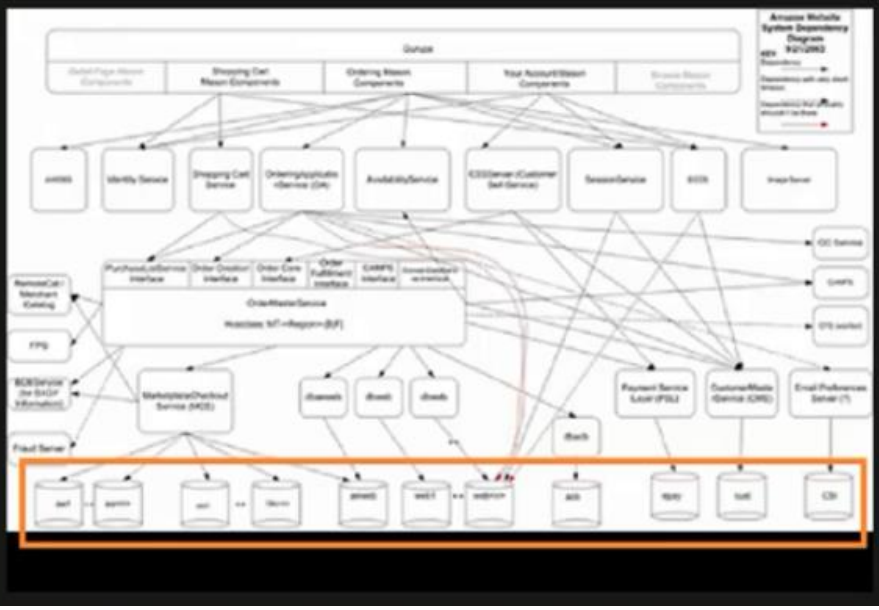

Prime day was run with lots of ***stateless services*** running and several ***databases***



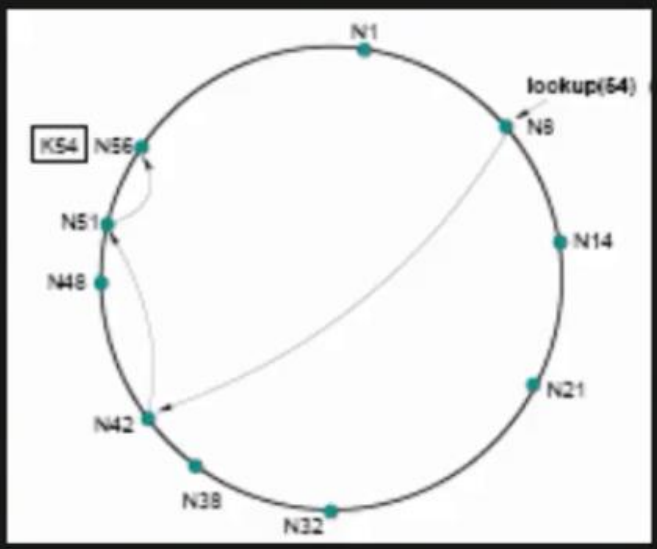

Databases here is ***DynamoDB***!



## Scale (circa 2003)

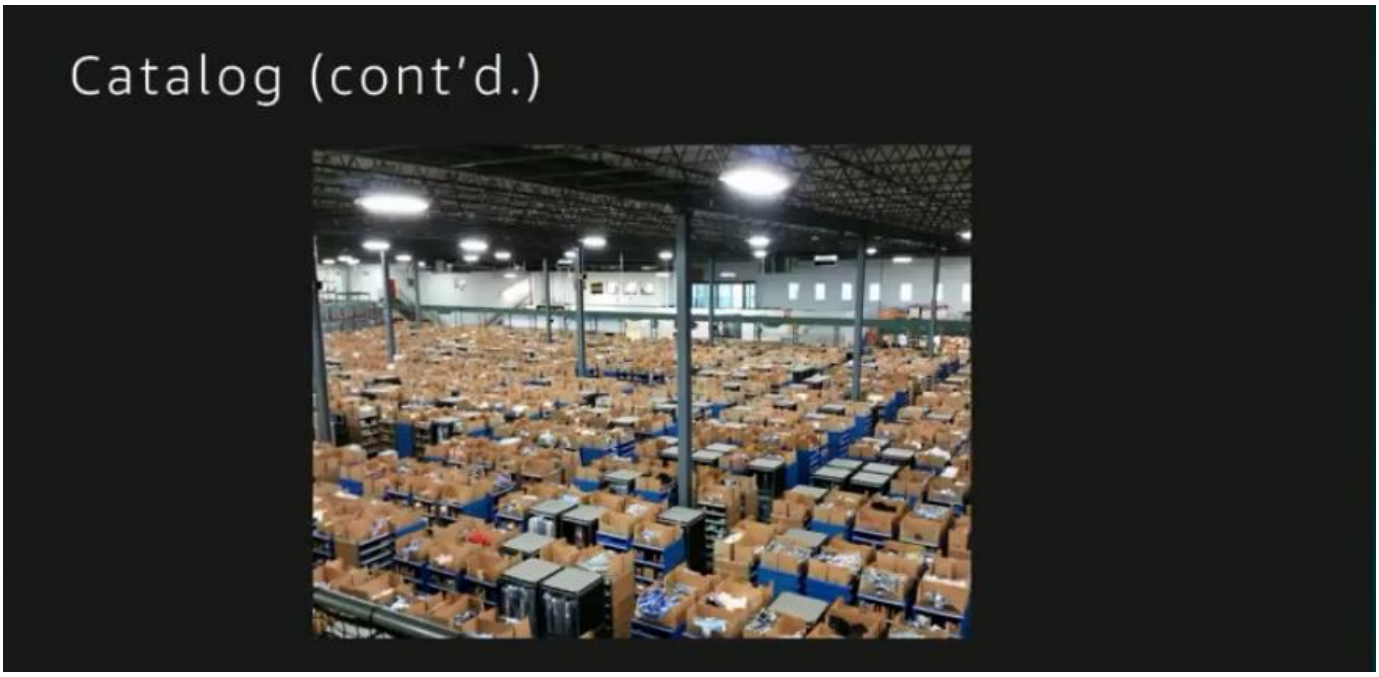


## Scale (circa 2000)




Instead of using just one huge server, we can wire up several servers together and use distributed hash tables for achieving massive scale


Amazon has a ***Catalog service*** of about 600 hundred million items with their individual 7000 attributes data in it




***DynamoDB*** is a ***distributed hash table DHT*** or simply a ***key/value store***. If you work with a DHT, there is no single master or point of failure and you can do read and write data as a single record or operation.

## Data in

Key (hash):
207808519/C00K2DG90

```
{ customer_id: 207808519,
  sku:" 207808519",
  version:2,
  product:{
    can_be_giftwrapped:true,
    binding:shoes,
    brand:"DISCOS",
    bullet_point:[
      {value:"Outdoor-ready runner with mesh",lang:en_US},
      {value:"Rearfoot GEL cushioning",lang:en_US},
      {value:"Removable sockliner accommodates orthotics",lang:en_US}
    ],
    upc:"887649299485",
} }
```

This is a sample catalog data in a simple document, the key here is the ***sku***. The hash algorithm takes the key and computes a hash that is uniform from the document data.

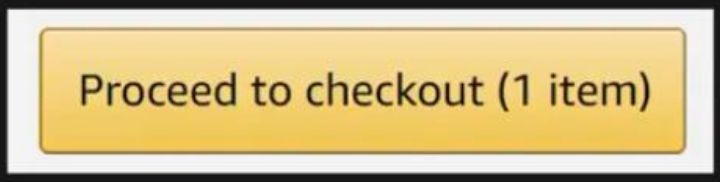
## Data out

Key (hash):
207808519/C00K2DG90

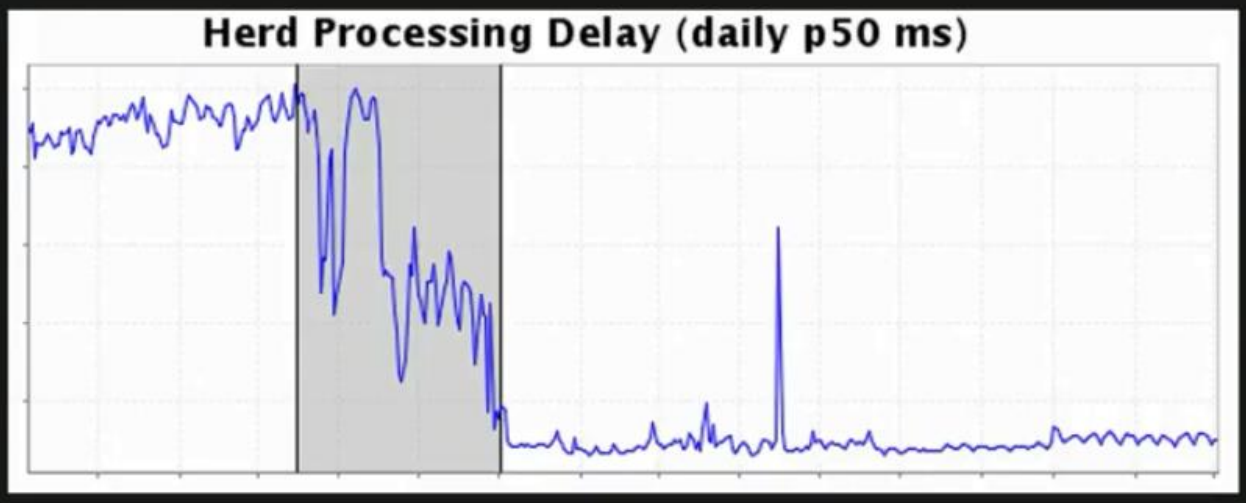
```
{ customer_id: 207808519,
  sku:" 207808519",
  version:2,
  product:{
    can_be_giftwrapped:true,
    binding:shoes,
    brand:"DISCOS",
    bullet_point:[
      {value:"Outdoor-ready runner with mesh",lang:en_US},
      {value:"Rearfoot GEL cushioning",lang:en_US},
      {value:"Removable sockliner accommodates orthotics",lang:en_US}
    ],
    upc:"887649299485",
} }
```

## Anything else?

## Ordering




## Ordering (cont'd.)



# Now…

Auto Scaling

| | Read capacity | | Write capacity | |
|---|---|---|---|---|
| | | | Same settings as read | |
| Target utilization | 50 | % | 50 | % |
| Minimum provisioned capacity | 5 | units | 5 | units |
| Maximum provisioned capacity | 1000 | units | 1000 | units |
| | Apply same settings to global secondary indexes | | Apply same settings to global secondary indexes | |

# Summary

Prime Day is big

Single instance doesn't cut it

DynamoDB for scale—up *and* down

Remember: schema, ACID, streams