April 14 -17 2014

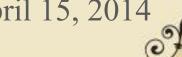




Seth Eliot Principal Knowledge Engineer, Test Excellence



Microsoft April 15, 2014



#### **About Seth**



















A/B Testing of Services

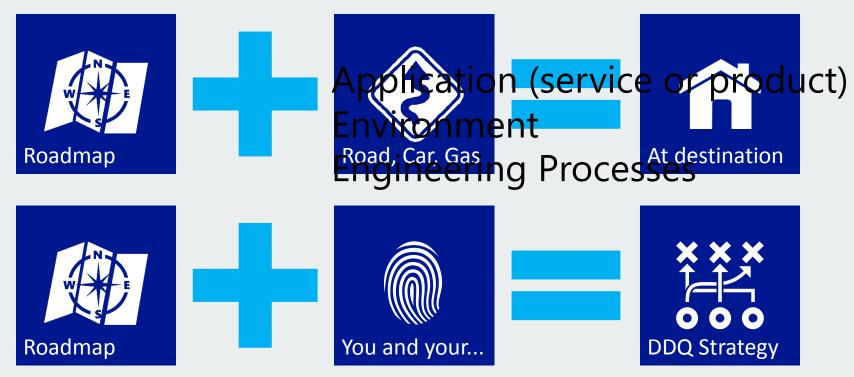
Petabytes Processed Services and Cloud and **Data-Driven Engineering** and...





#### In This Session, You'll Learn...

...how to create your "roadmap" to DDQ





#### Data-Driven Quality

What is it? Why is it important?



#### Data, where have we been?

#### The HiPPO



<u>Highest Paid</u> <u>Person's Opinion</u>

#### **Engineering Data**

Test pass/fail results

Bug counts

Delivery cadence

Code coverage

Code Churn



## Data, where are we going?

Engineering data

Test results -> Scoring engines using Bayesian analysis

Production-quality data
You may have heard of TiP
Not as difficult as you might think
Lots of solutions for lots of application types













### Virtuous cycle of DDQ

Deploy

Detect

Deploy Fix Detect



## The roadmap

Determine your questions

Design for production-quality data

Select your data sources

Use the right data tools

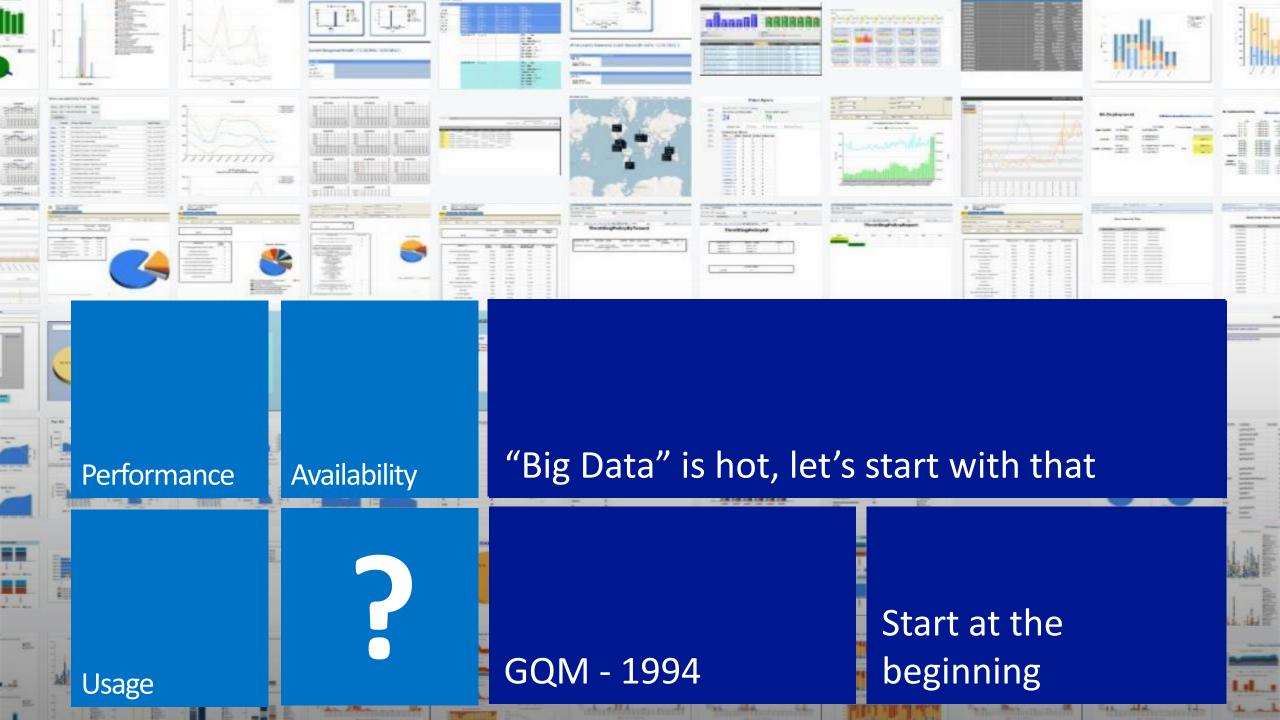
Get answers to your questions

Learn new questions

Repeat









#### Determine your questions

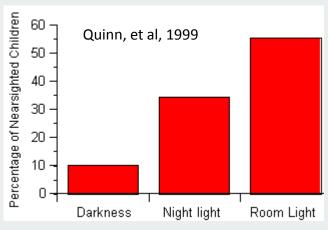
What do you need to know about quality?



# Why not just get data and look for answers?



Does sunscreen increase chance of drowning?



Do night-lights cause near-sightedness in children?



## What questions does EXO ask?



Microsoft Exchange Online Dedicated Plans Version Service Level Agreement (SLA)

| Monthly    | Service |
|------------|---------|
| Uptime     | Credit  |
| Percentage |         |
| < 99.9%    | 25%     |
| < 99%      | 50%     |
| < 95%      | 100%    |

"Downtime" is defined as any period of time when users are unable to send or receive email via all supported mailbox access



#### Is it available?

#### Often a Pri 0

Is the application (product, service) there for the user?



"Dialtone"



Critical work stoppage for user

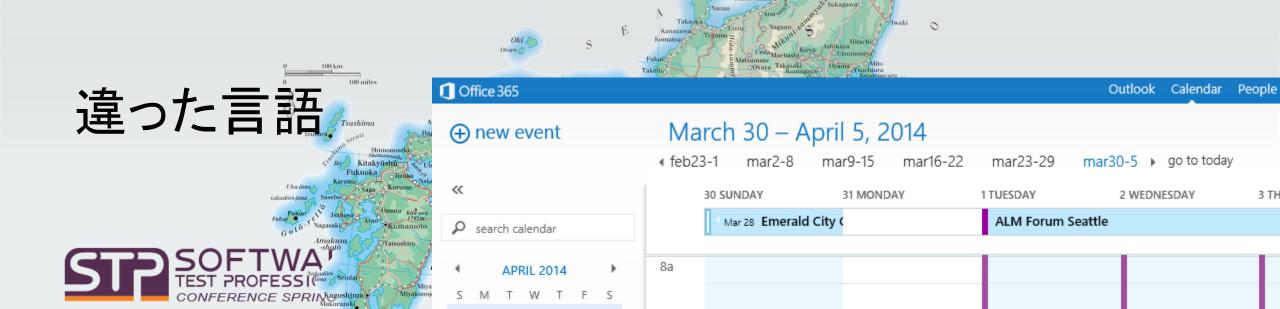




How do users perceive availability?

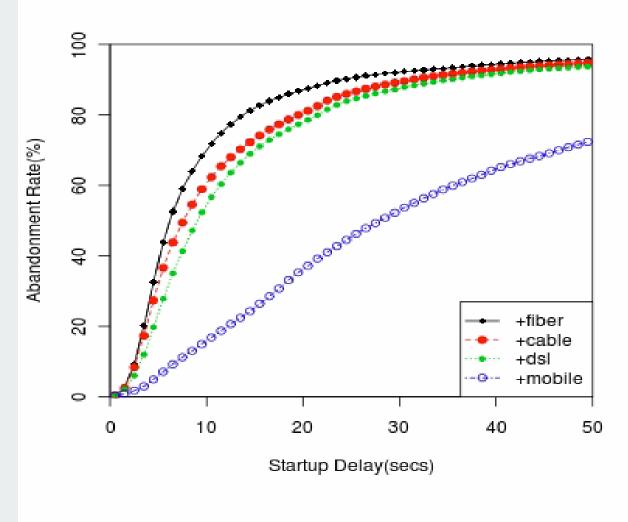
Application works, but feature of the local point o

Occasionally does not render/load look properly

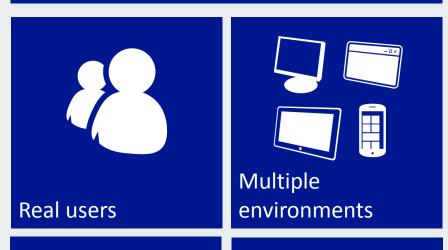


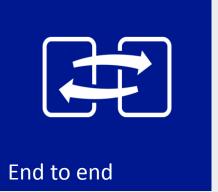
#### How is performance?

#### Huge Impact



## Power of Production Data





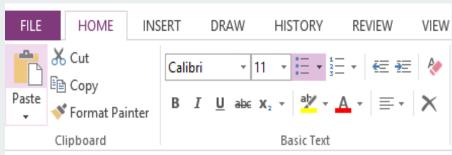


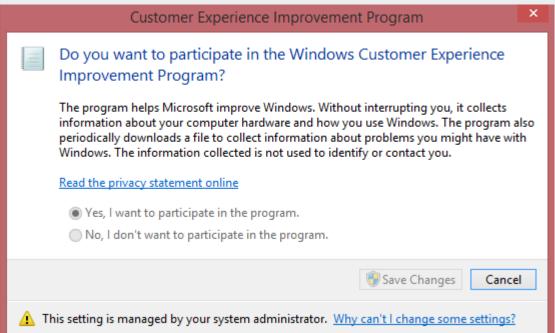
# What do users do?

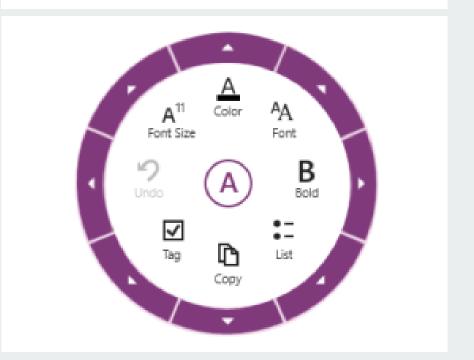




Customer Experience Improvement Program (CEIP)

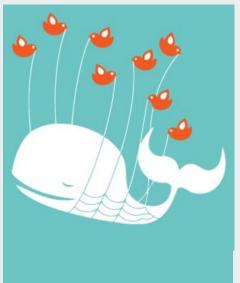






#### Your turn - Determine your questions





What type of answer are you looking for?

Availability, performance, usage?
Prioritize

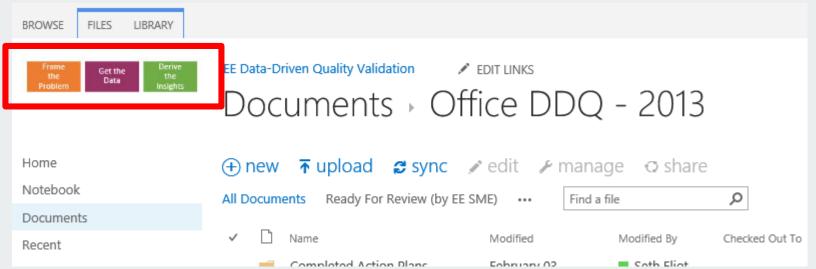
When does availability NOT come first?



#### Your turn - Determine your questions



What are the key scenarios for the type you selected?





What is high priority?





#### Design for production-quality data

Get data from or near production



## Two types of data to acquire

Active: synthetic

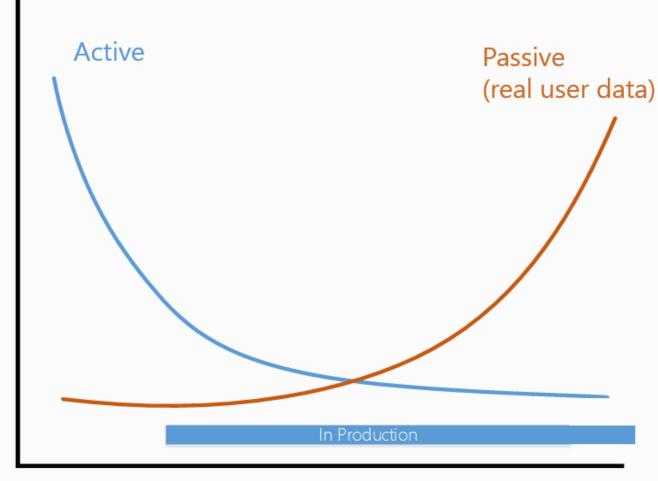
Passive: real (RUM)

Active data in prod

For services only?

Client: is the service there?







## Staged data acquisition mitigates risk

#### Service

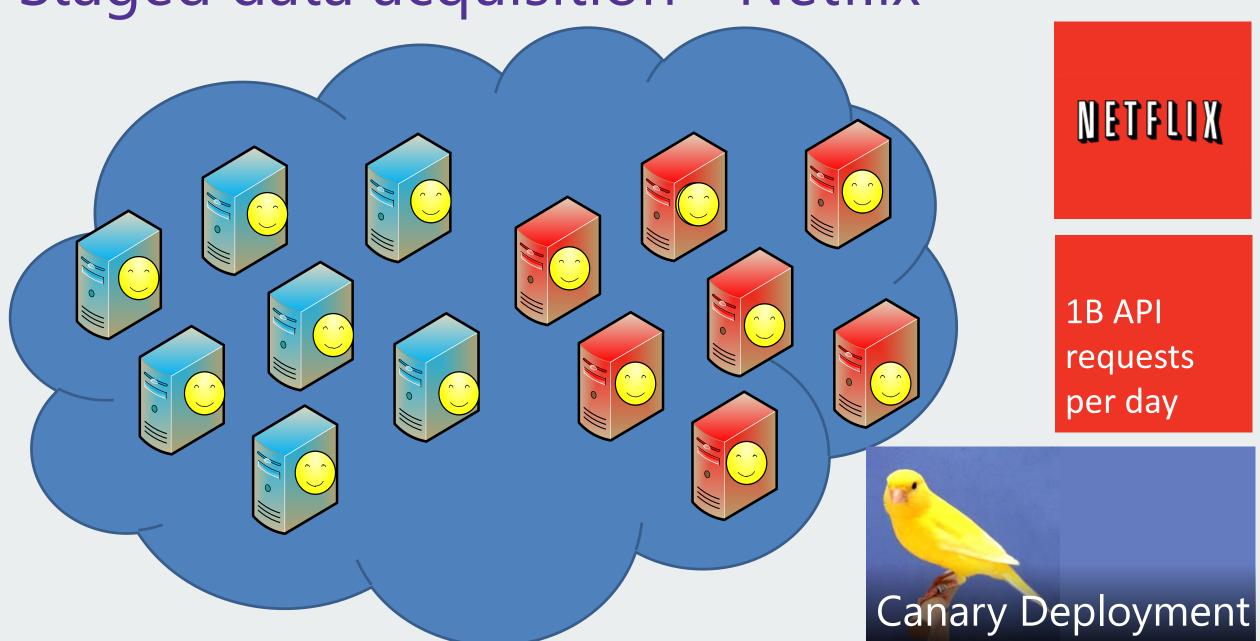


#### Product (client, on-prem server)

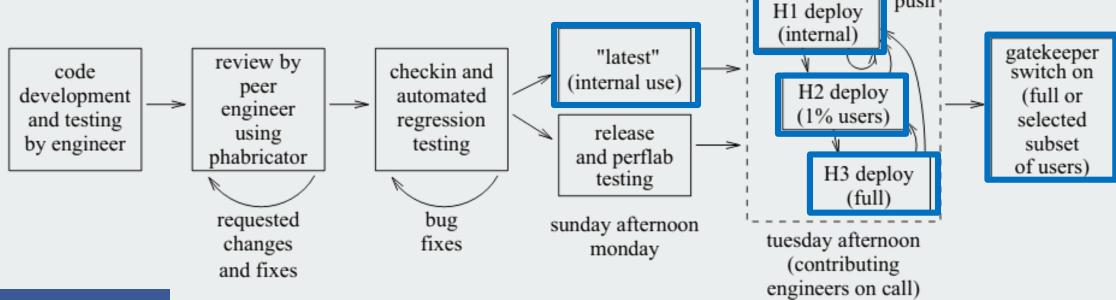




Staged data acquisition - Netflix



Staged Data Acquisition - Facebook



push

facebook.

Dogfood

In prod, no users (except internal ones)

Some servers in Production

World-wide deployment

Feature flags



## Speed of deployment

Deploy

Usually easy for services

Client apps may have a deployment capability

Detect

Or may make use of feature flags

EaaSy - Everything is now connected and thus updatable



#### Data acquisition for clients

Filtering and aggregation at client

Be kind to the client

Don't abuse user resources:

Bandwidth

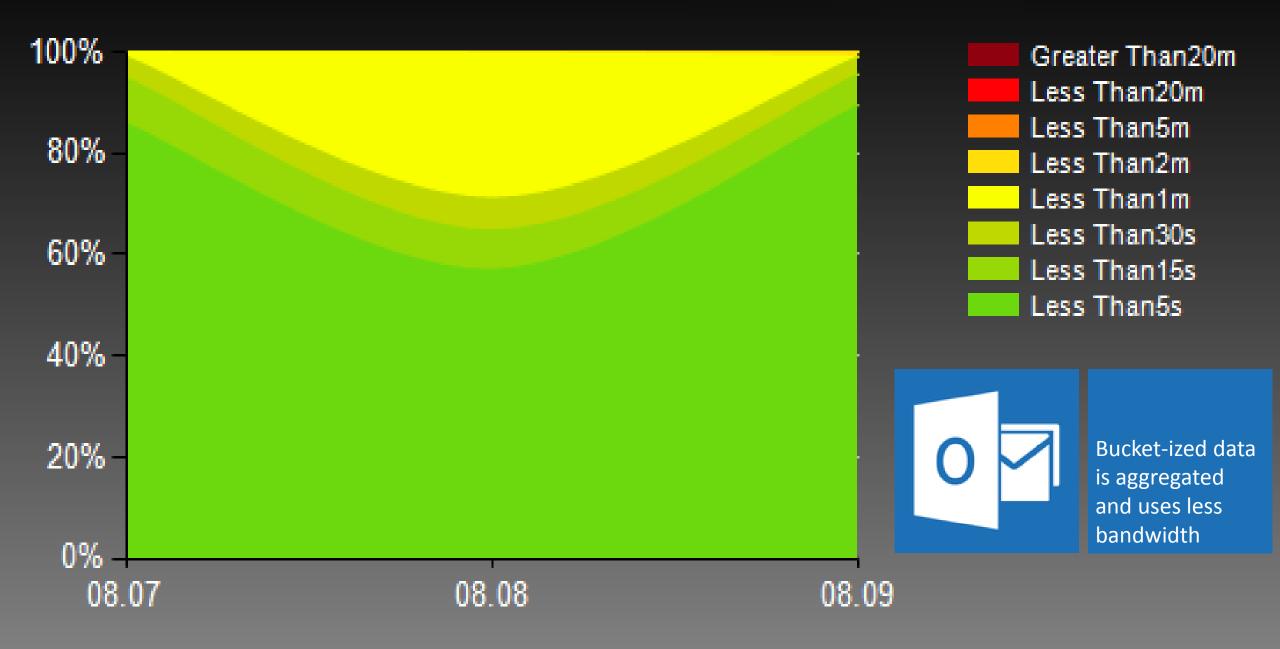
**Battery** 

Disk





#### Initial Connect: HTTP

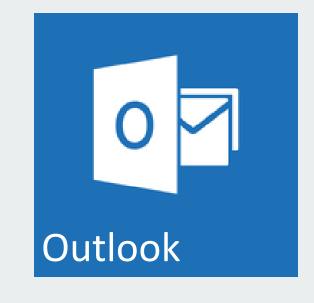


#### Data acquisition for clients

Filtering and aggregation at client
Be kind to the client

Pipeline to collect and process data Make it easy

Staged Data Acquisition







Outlook Team



MS Office Team



Microsoft



Customers

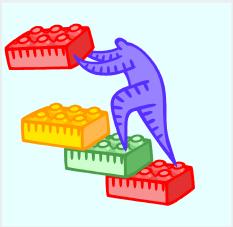


**Scale Validation** 

## Your turn - Design for productionquality data



What might be your stages for risk mitigated data acquisition?



Role of active and passive monitoring?

How can you engineer for EaaSy deployment?

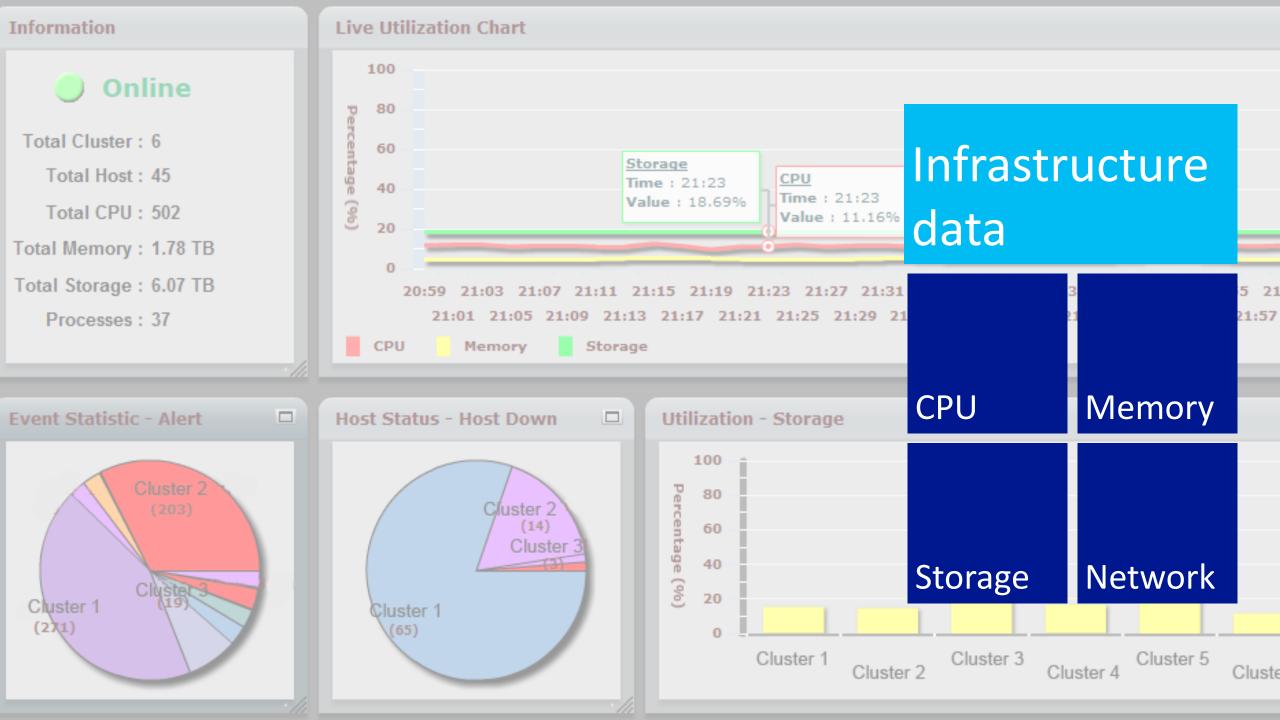




#### Select your data sources

## Determine the data necessary to answer your questions





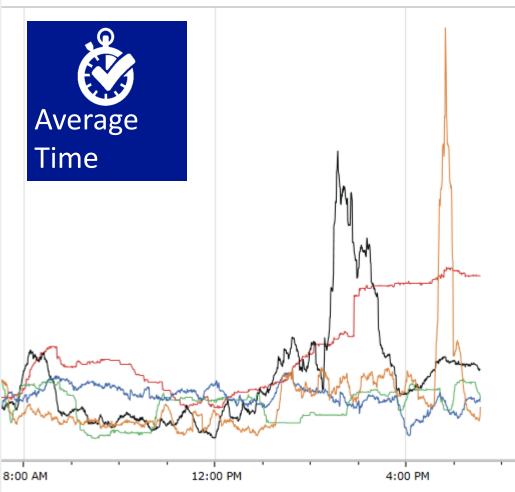
## Application data



New and Open

Download Template







## Hang and crash data

Specialized application data

Most frequently encountered conditions

Bucket the Big Data and find the offenders

Get offending code and function calls (stack)

#### But it worked in Dogfood

Culprit was a old version add-in Harden against this







## Usage data



Client-side instrumentation

Proprietary

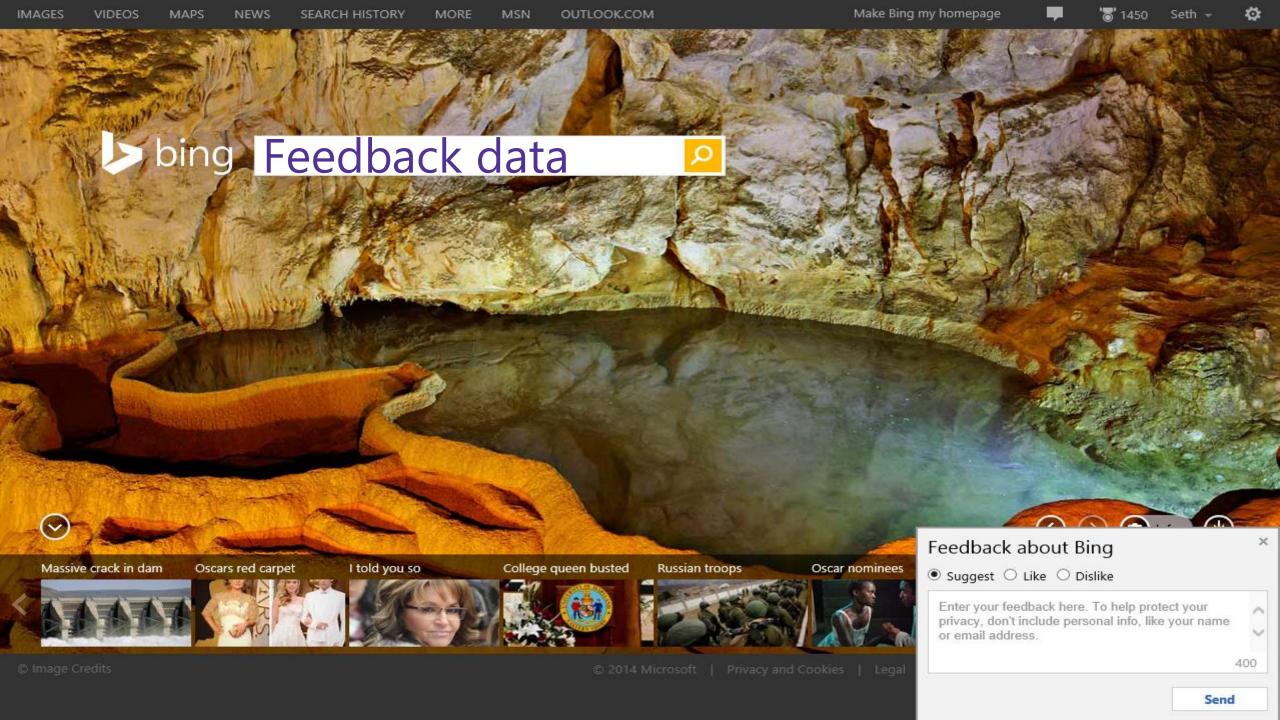
Javascript: clicks, hovers [web apps]

Get 1x1 GIF: Page Views [web apps]

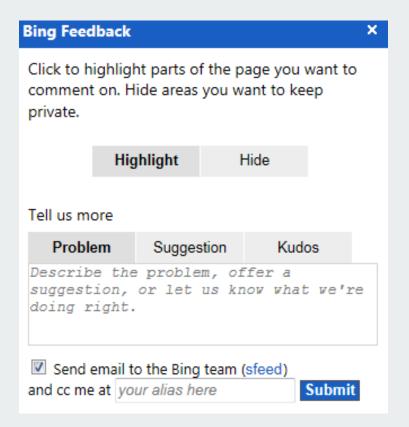
100% **Users** 50% 50% Users Users Control: **Treatment: Existing System Existing System** with Feature X Users interactions instrumented, analyzed & compared Analyze at the end of the experiment

Combine into more complex scenarios How did user get to shopping cart checkout?

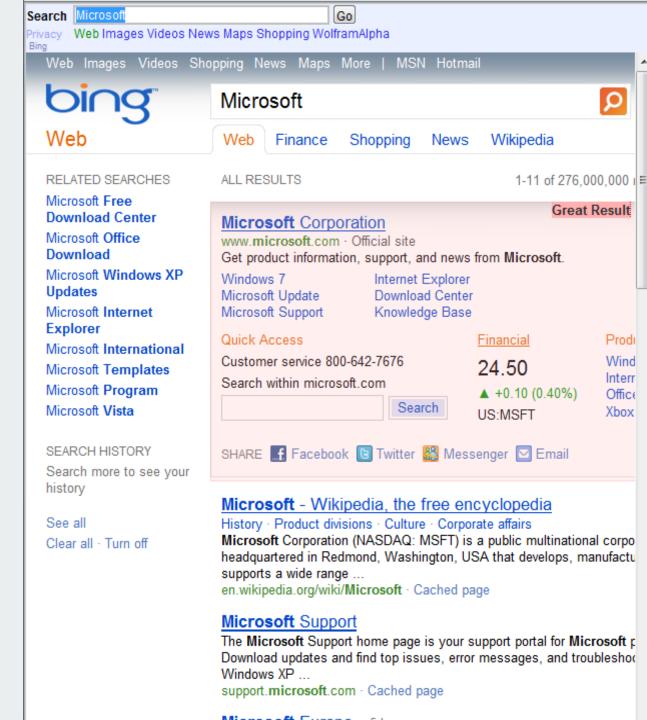




#### Feedback data







#### Feedback data also includes...

Customer Support Data





| Service         | Severity | Status              |
|-----------------|----------|---------------------|
| Lync Online     | Sev1     | Service Degradation |
| Exchange Online | Sev2     | Investigating       |



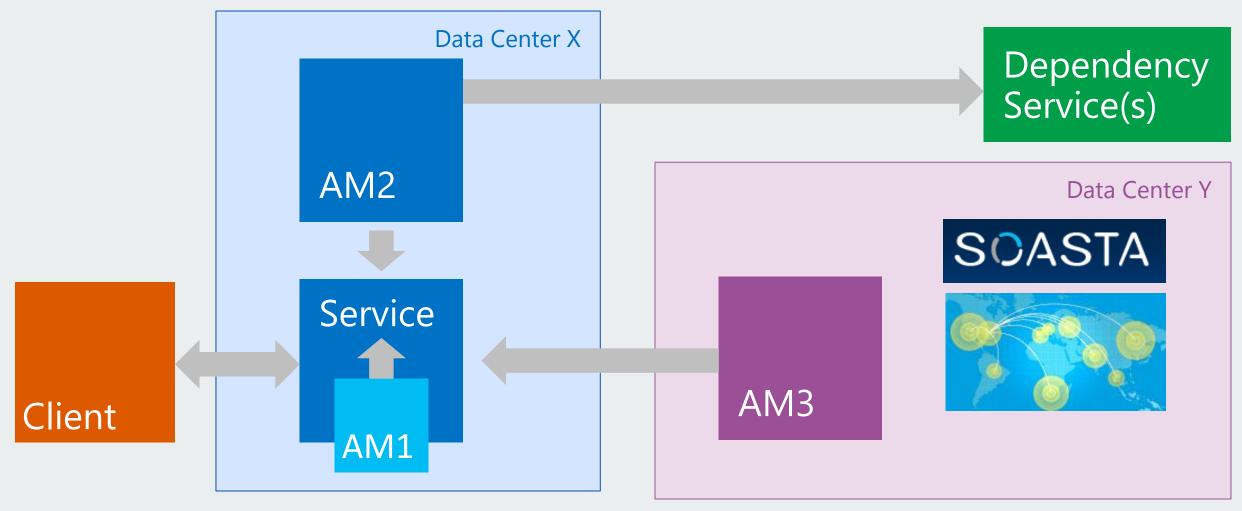
Social Media Mining







# Active monitoring data





#### Ease of detection

Deploy

Detect

Again services have it easy

Many clients are always connected

On prem servers (enterprise) require partnerships

EaaSy – Rich near real-time telemetry



# Data handling - privacy

Transparency and Control Collection and Retention

Depends on Type
anonymous Data
pseudonymous
personally identifiable info (PII)
sensitive PII

Depends on Purpose
provide the service
improve the current service
improve a future version service
improve non-associated services
content personalization
ad targeting



# Data handling – non-service products

Client and on-prem server considerations

**User owned resources:** 

bandwidth

battery

disk, cpu, etc...

Correlations

end-to-end across clients and services

by user, by session



### Your turn - Select your data sources





Infrastructure data

Application data

Hang and crash data

Usage data

Feedback data

Active monitoring data

Design to handle your data





### Use the right data tools

A 50,000 foot view



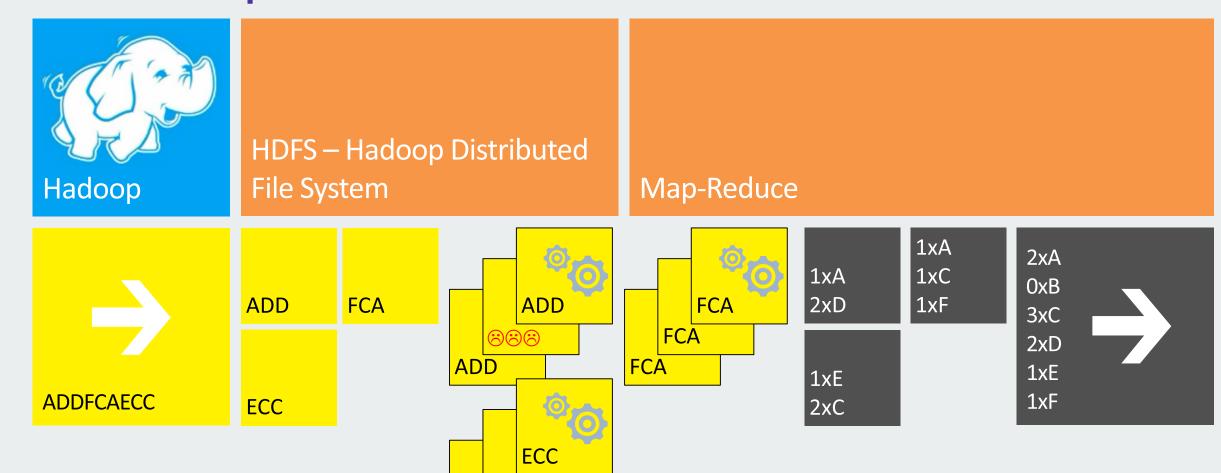


## Data storage and processing systems

| Database                                     | Table storage<br>(SQL)            | Optimized for CRUD –<br>(create, read, update,<br>delete) of single records |
|--|-----------------------------------|---|
| Data<br>Warehouse                            | Table storage<br>(SQL)            | Table structure optimized for queries and bulk insert                       |
| OLAP Cubes<br>(Online Analytical Processing) | Multidimensional<br>Cube (MDX)    | Aggregations of measures for multiple dimensions                            |
| Hadoop /<br>Map-Reduce                       | Distributed File<br>System (HDFS) | Big Data TB-PB-EB<br>Unstructured Data                                      |



#### Hadoop in 60 Seconds

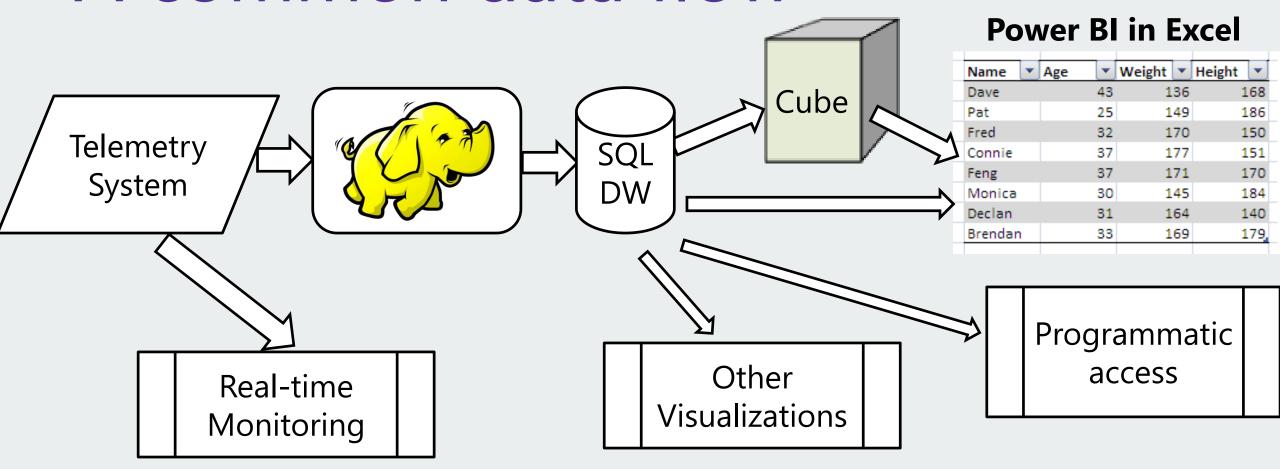


ECC

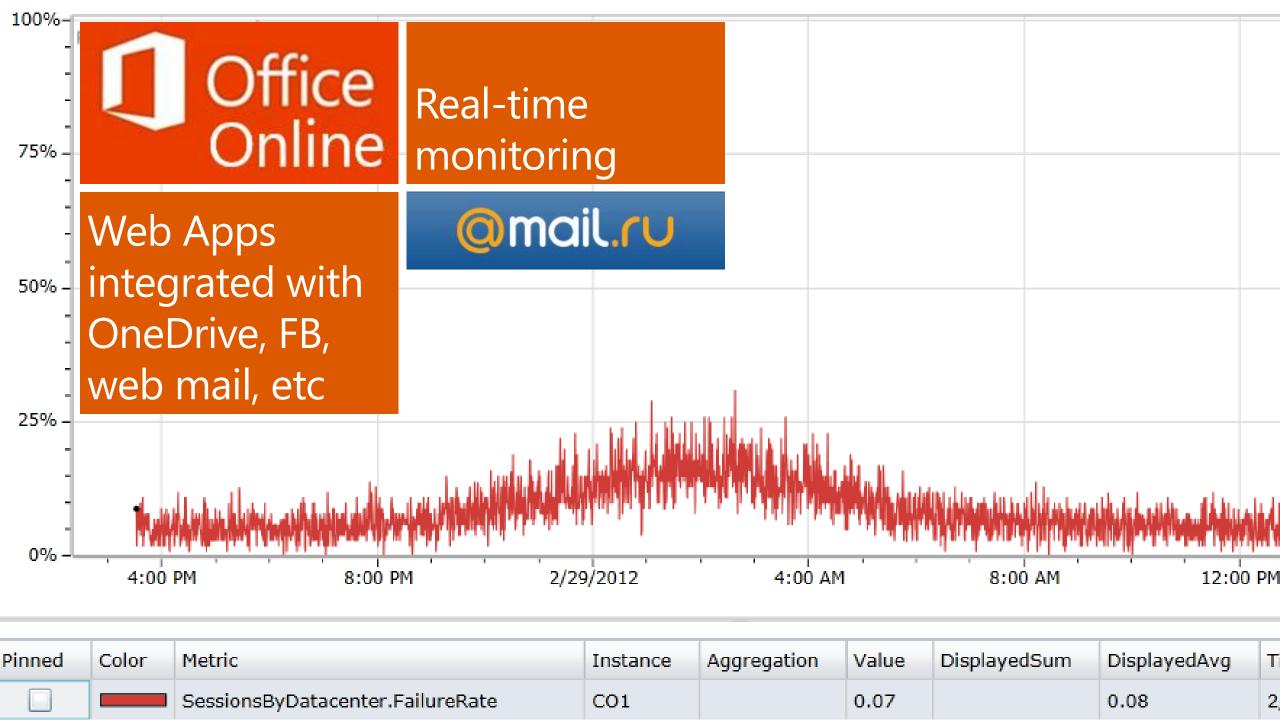
ECC

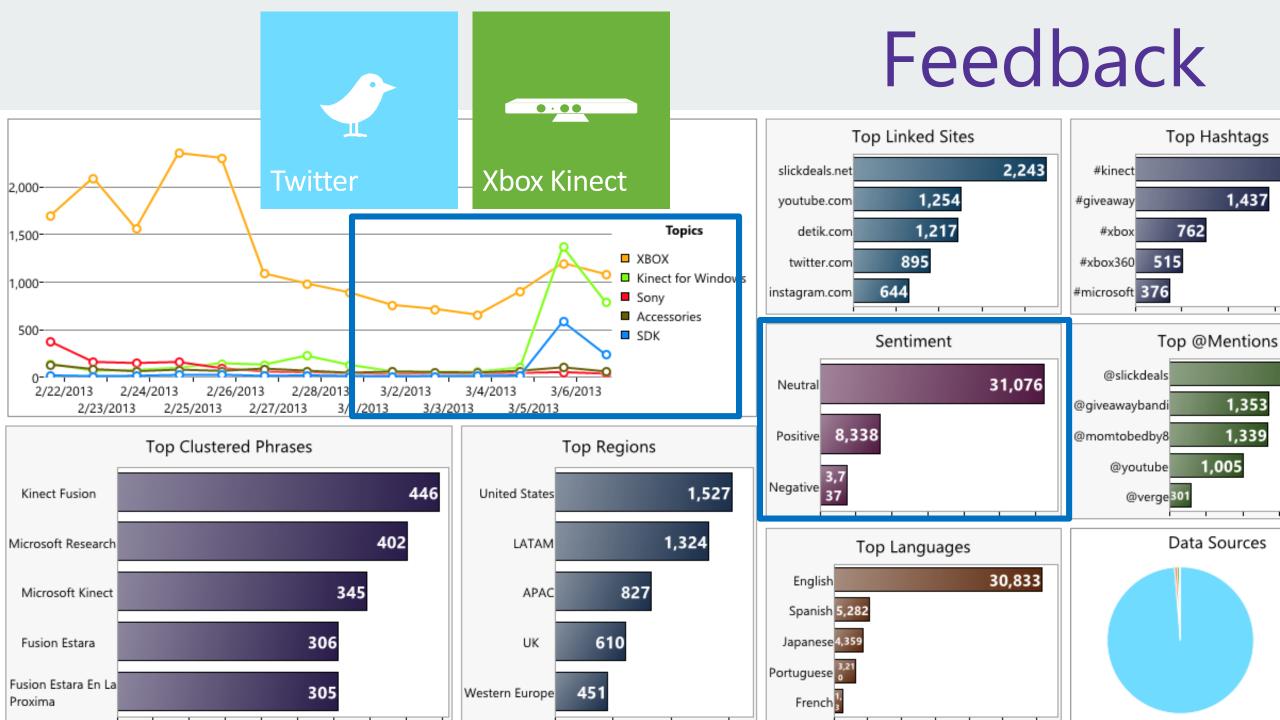


#### A common data flow









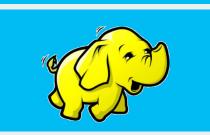
# Your turn - Use the right data processing tools



DB, DW, Cube, Big-Data platforms

Put it all together





Do you need real-time monitoring?

...sentiment analysis?





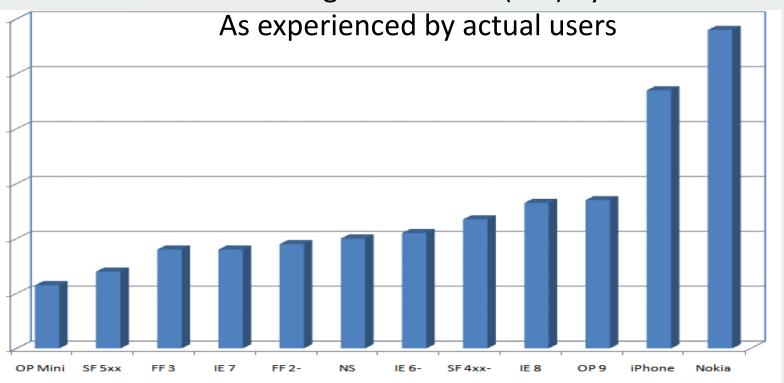
# Get answers to your questions

and learn new questions



## Outlook.com prioritizes performance

View Inbox – Page Load Time (PLT) by Browser



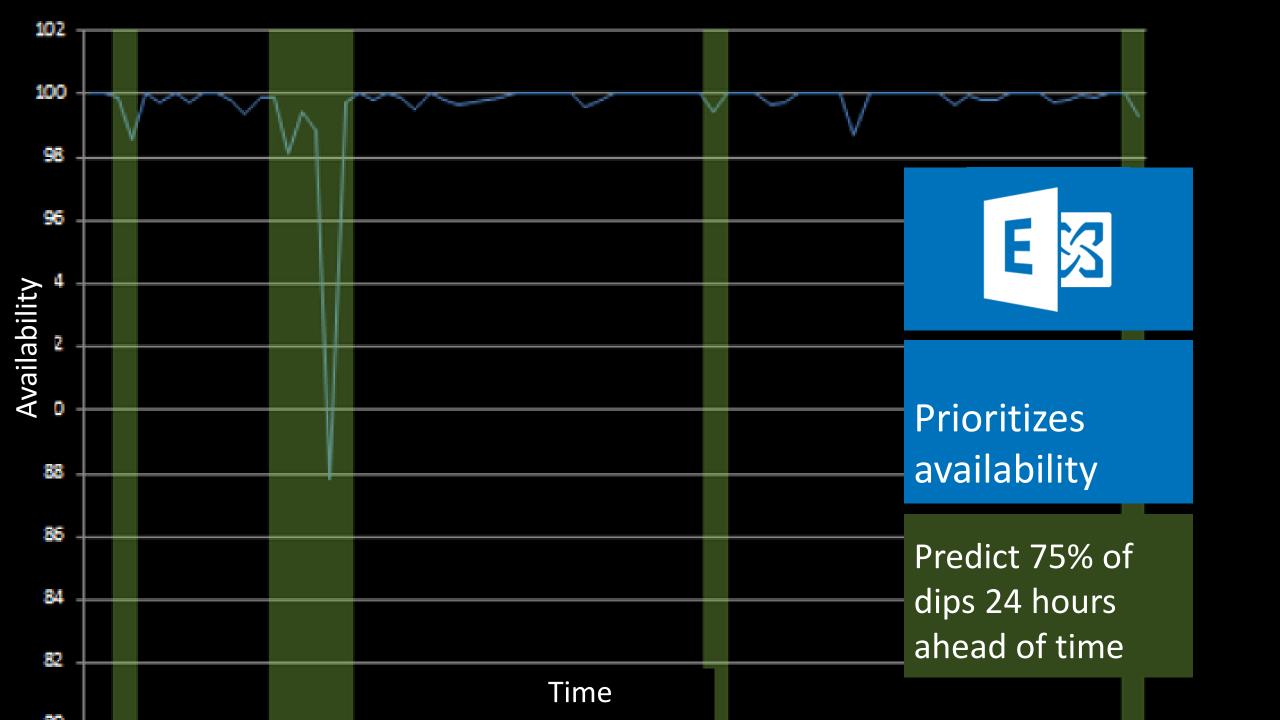


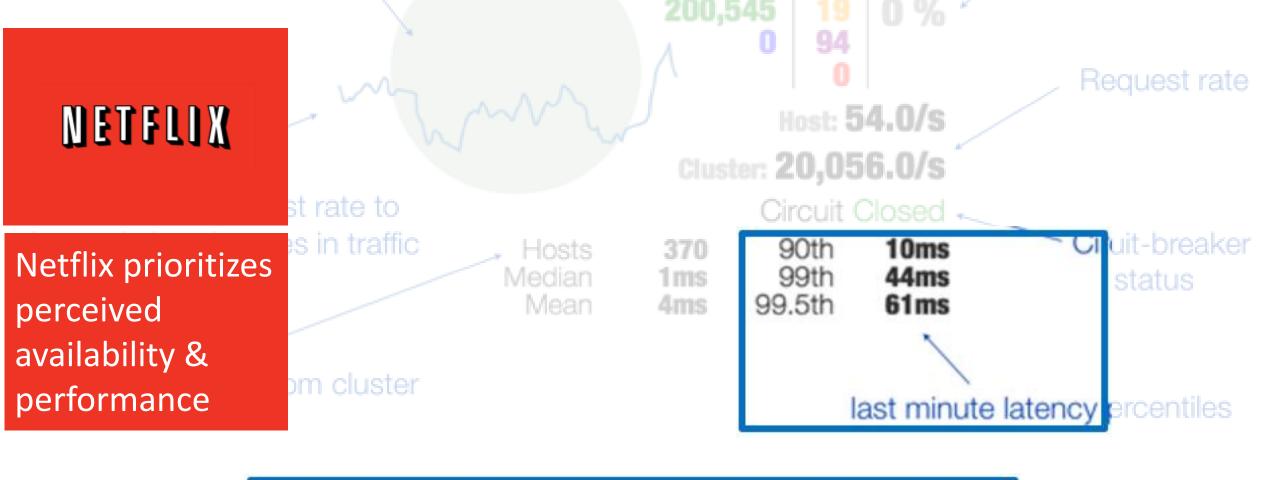
JSI
JavaScript
Instrumentation

500 Million measurements per month









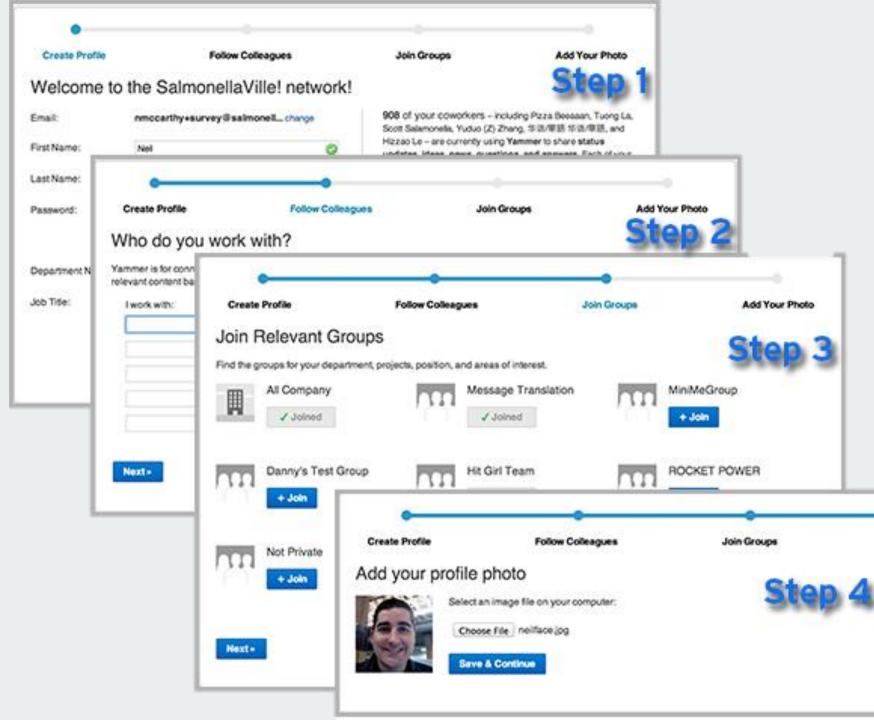




Yammer prioritizes usage

What happens to new user retention when you shorten the signup flow?





# Find new questions and repeat



Try algorithm

Collect data

Adjust algorithm

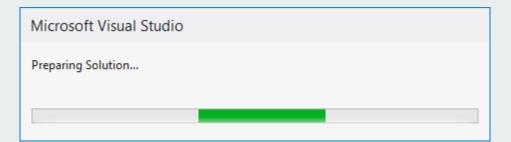
Collect data

Repeat





# Find new questions and repeat



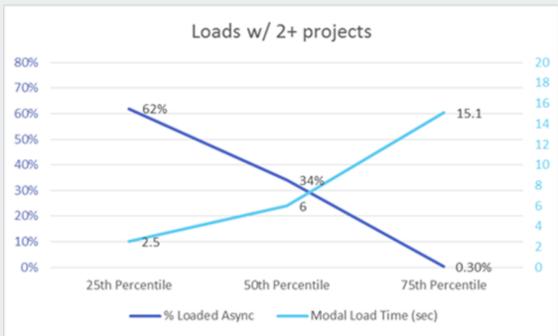


In Visual Studio 2012 we added asynchronous loading for *solutions* 



We also added Telemetry

We can see how it works





# Your turn – Get answers; find more questions



Enjoy the rest of STPCon :-)

Get back to work



Implement your roadmap

Get answers to your quality assessment questions



## Summary

Determine your questions Design for production-quality data Select your data sources Use the right data tools Get answers to your questions Learn new questions Repeat

The most recent version of this deck can be found at <a href="http://setheliot.com">http://setheliot.com</a>



## Special thanks to these folks

Ravi Vedula

Andrea Jesse

Bill Hodghead

Danny Thayer

Joseph Sefair

Kitty Thomas

Amanda Reinke

Heather Lader

**David Brooks** 

Mike Tholfsen

Jodie Draper

Brian Mueller

Tara Roth

Dror Cohen

Nathan Halstead

Lori Oviatt

Monica Catunda

Lynette Skinner

Joe Schumacher

Donny Luu

John Hoegger

Alain Anyouzoa

Any questions?



## Please fill out the survey



Create Your Roadmap to Data-Driven Quality
Session 101

Seth Eliot

