



The Human Side of Microservices
John Billings

@jnb42
billings@yelp.com



Introduction

About me

- PhD in compiling and verifying routing protocols
- Joined Yelp in February 2011
- Initially:
 - Joined <10 person search team
 - 70 - 80 engineers
- Now 300+ engineers

Yelp's Mission

Connecting people with great local businesses.

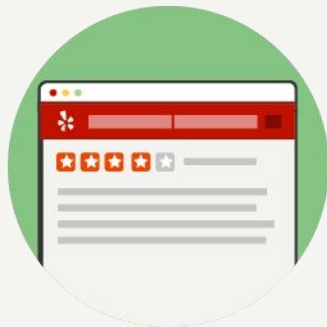


Yelp Stats

As of Q1 2016



90M



102M



70%



32



Best pizza in New York, NY

Showing 1-10 of 11124

Filters

\$

\$\$

\$\$\$

\$\$\$\$


🕒 Open Now

 Order Pickup or Delivery

 Make a Reservation

 Offering a Deal

 All Filters

☐  Order Pickup or Delivery

Pickup ▾


1 Yelp St., San Francisco, CA 94105



Ad Margherita NYC

★★★★☆ 103 reviews
\$\$ · Italian, Pizza

Little Italy
197 Grand St
New York, NY 10013
(212) 226-8391

 Last nite, I stopped by at Margherita because after my last review the owner - Giuseppe messaged me back and apologized regarding the extra service charge and offered me a free pizza... [read more](#)




Ad Mama Mia 44 SW

★★★★☆ 82 reviews
\$\$ · Pizza, Seafood, Italian

Hell's Kitchen, Midtown West
621 9th Ave
New York, NY 10036
(212) 315-4582

 This restaurant takes reservations

[Find a Table](#)

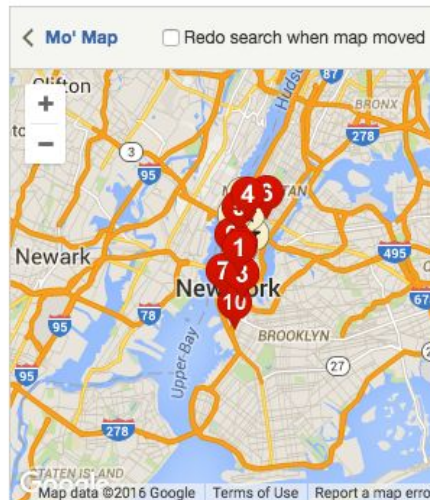
 This place is the best place so far since we arrived here on monday. I am a business owner and I never say anything bad about hard working people that make the leap into small... [read more](#)



1. Lombardi's Pizza

★★★★☆ 4775 reviews
\$\$ · Pizza

Nolita
32 Spring St
New York, NY 10012
(212) 941-7994



Ads

Luzzo's La Pizza Napoletana

★★★★☆ 904 reviews

Lauren B. - I was recommended this place when I first moved into the east village. I... [read more](#)

Zia Maria

★★★★☆ 11 reviews

Zia Maria Brick -Oven Pizza on Chelsea one of the most popular Italian... [read more](#)

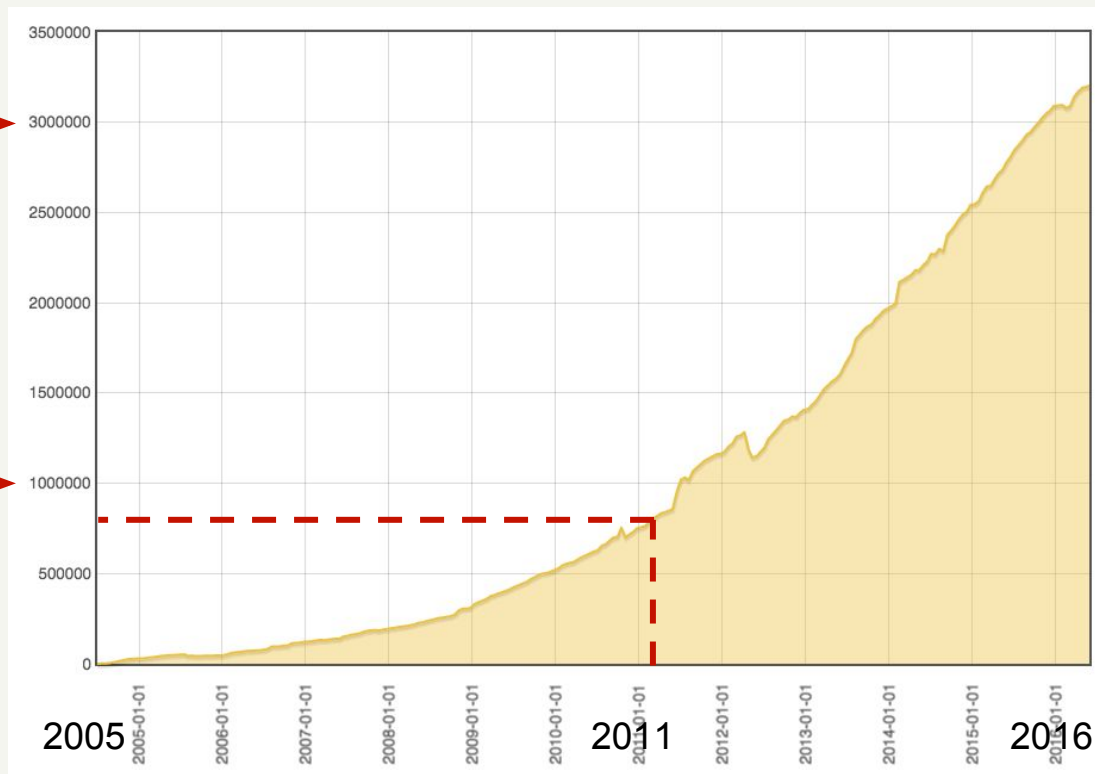


LoC in monolith

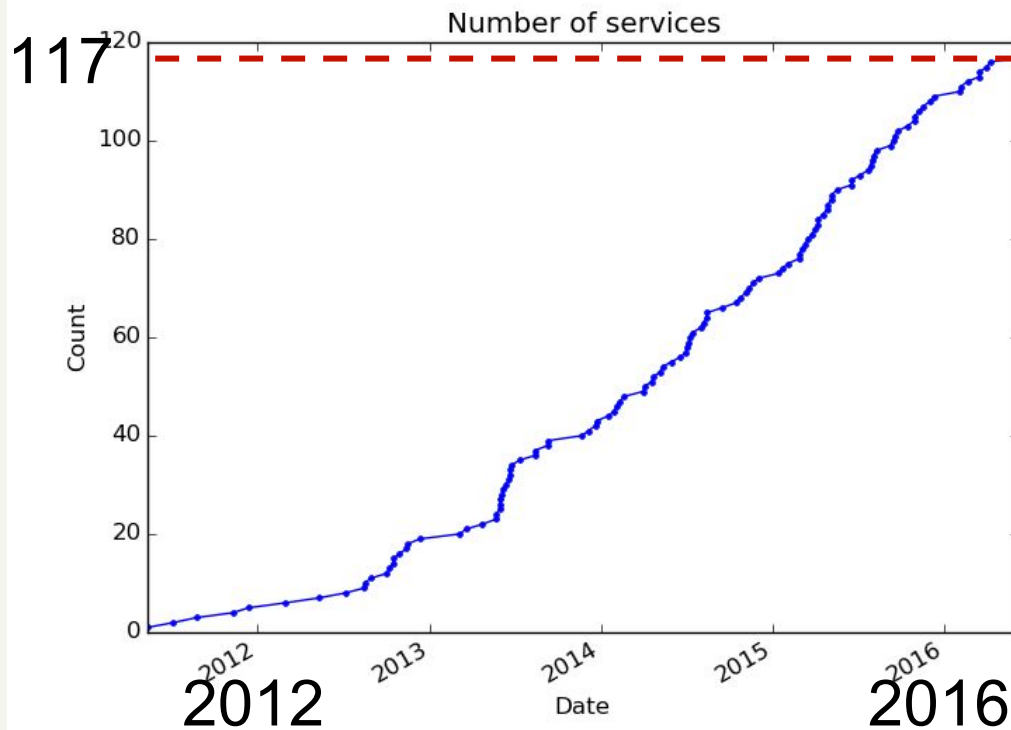
3 million



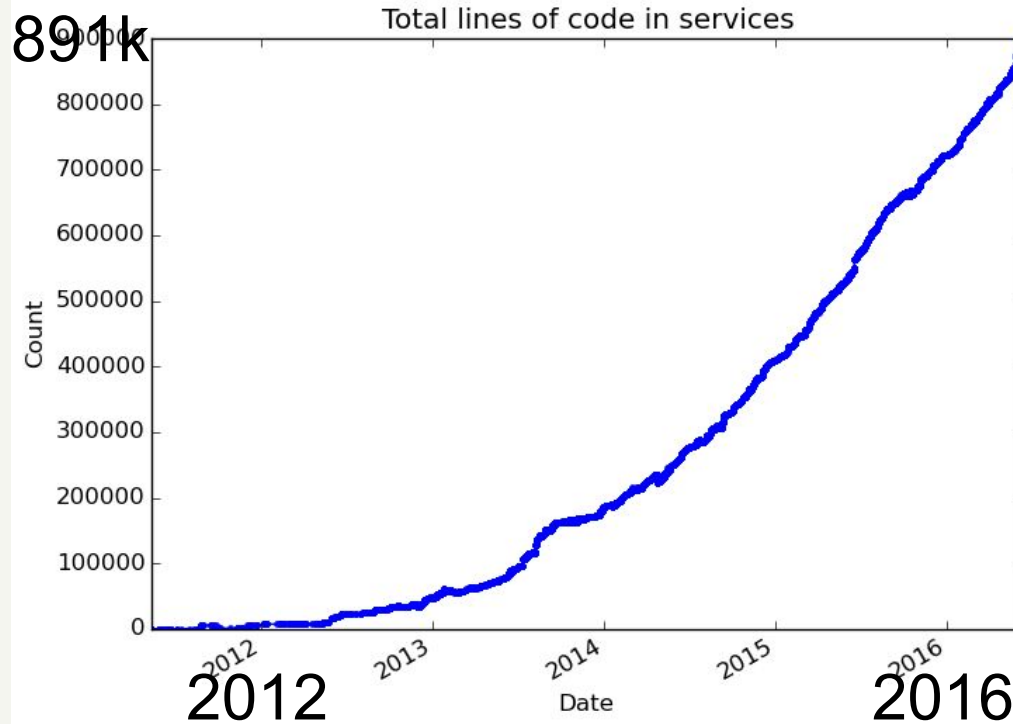
1 million



Number of microservices



LoC in microservices



Microservice adoption



<https://www.flickr.com/photos/24736216@N07/4433492189>

“The Diffusion Process”

As agriculture becomes more complex and problems of adjustment more acute, it becomes increasingly important to know more about the educational processes which lead people to accept new ideas and adapt them to their individual enterprises. Such information is of particular value to the educational programs of the Cooperative Extension Service in Agriculture and Home Economics. It is of equal importance to other groups in industry and government which work and deal with farm people.

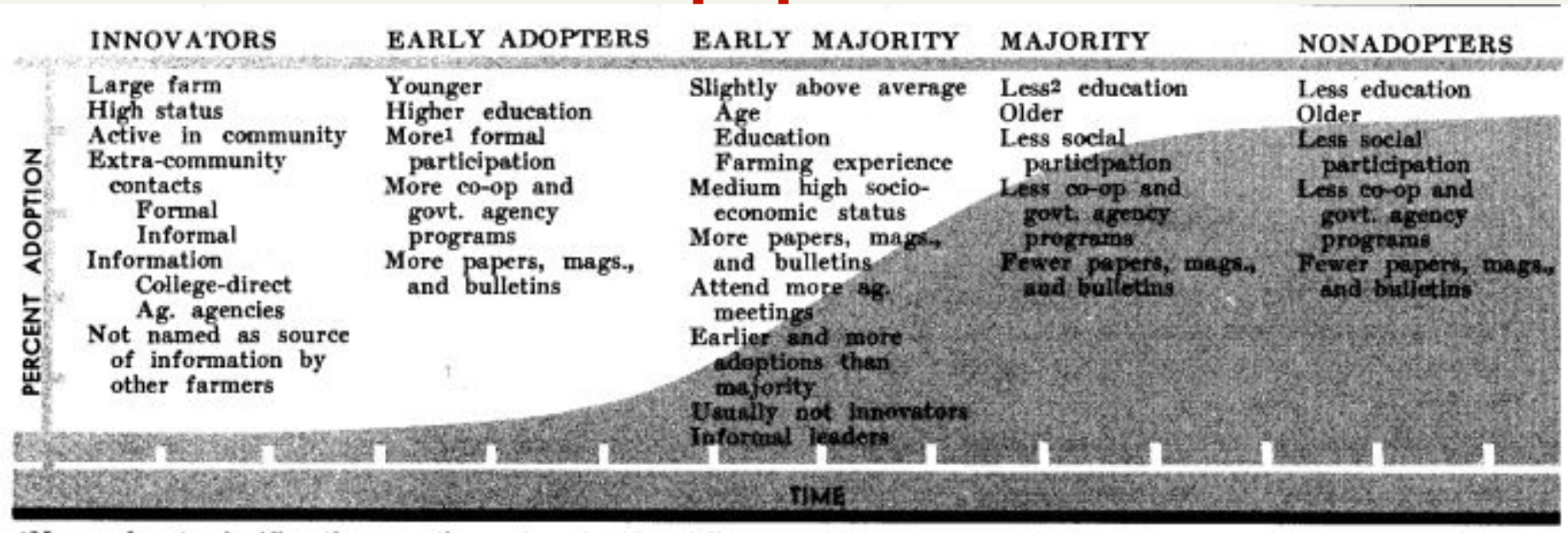
George M. Beal and Joe M. Bohlen, 1957

<http://www.soc.iastate.edu/extension/pub/comm/SP18.pdf>

This paper is a summary of the flannelboard presentation on how farm people accept new ideas. It is based on the findings of 35 research studies conducted during the past 20 years in various parts of the United States, including Illinois, Iowa, Missouri, Wisconsin, New Hampshire, New York, West Virginia, Kentucky and Tennessee.¹ The findings of these studies are presented in a framework which will be useful to people who are faced with the problem of diffusing new ideas and practices.

of the information used by farm people. Practices studied were farm and home practices, such as use of fertilizer, 2,4-D, antibiotics, contouring, new fabrics and deep freezers as well as several others.

Technology adoption curve for a population



The chasm



Innovators

What are some of the characteristics of innovators? They have the larger farms, they usually have a relatively high net worth and—probably more important—a large amount of risk capital. They can afford to take some calculated risks. They are respected and have prestige. They adhere to and represent important community standards. Quite often these innovators come

Innovators

They are active in the community. They have power. They may not hold many offices in the community, but they may act behind the scenes. For instance, they may not be members of the school board, but they have a lot to say about who serves on the board.

CEP39ServiceOrientedArchitecture

Abstract

Break out appropriate parts of our code into REST services with their own codebase and deploy process.

Motivation & Rationale

- Modularize our code
- Trim out unnecessary functions and interfaces
- Allow JavaScript access to the same logic that Python has
- Good platforms allow us to build awesome stuff

The Geocoder Service

```
class ReverseHandler(RequestHandler):
    """
    Interface to reverse geocoder.

    lang -- the language for the response; an ISO 639-1 language code.
    types -- one or more of 'hoods', .., separated by commas.
    lng -- query longitude.
    lat -- query latitude.
    """

    path = r'/reverse/(?P<lang>.*?)/(?P<types>.*?)/(?P<lng>-?[.0-9]+),(?P<lat>-?[.0-9]+)'
    method_name = 'reverse'

    def get(self, lang, types, lng, lat):
        types = list(set(filter(None, types.split(','))))
        lng, lat = float(lng), float(lat)
        locations = []
        for type in types:
            if type == 'countries':
                locations.extend(c for _, c in quadtree.query_by_point(countries_quadtree, (lng, lat)))
```



Trac Archive / TRAC-19564

pick webserver for Geoservices



Trac Archive / TRAC-23035

ganglia monitoring for geoservices



Trac Archive / TRAC-23481

Benchmark geoservices code



Trac Archive / TRAC-24513

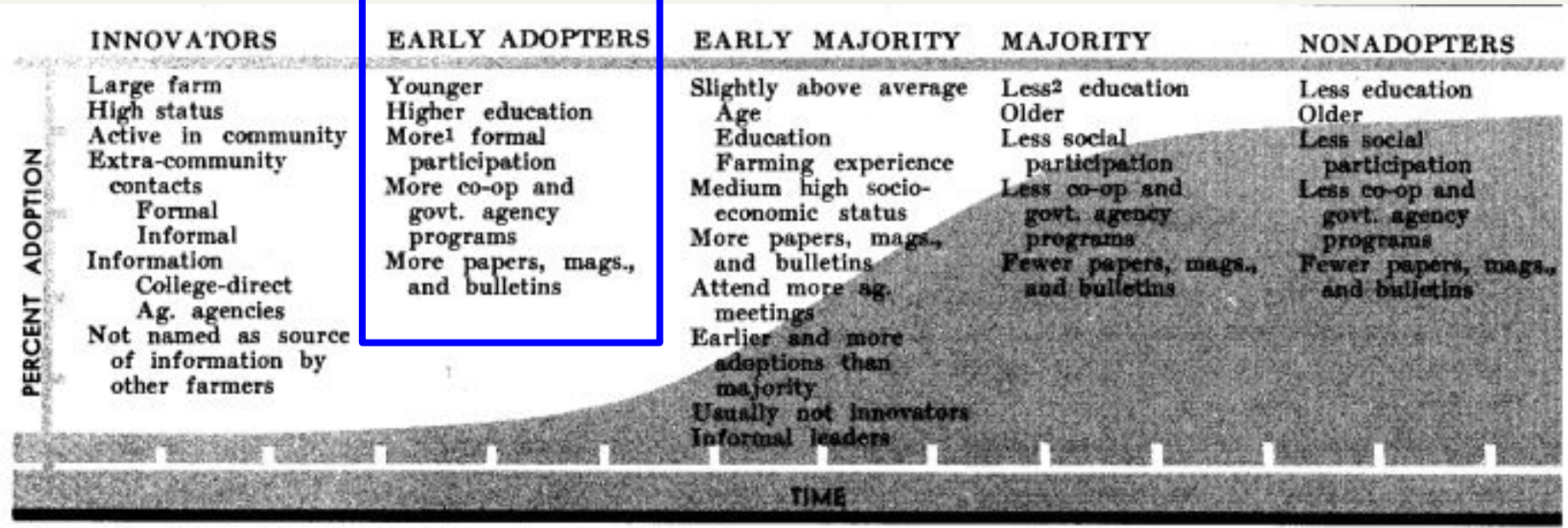
bad cookies make the geocoder cause mondo site issues



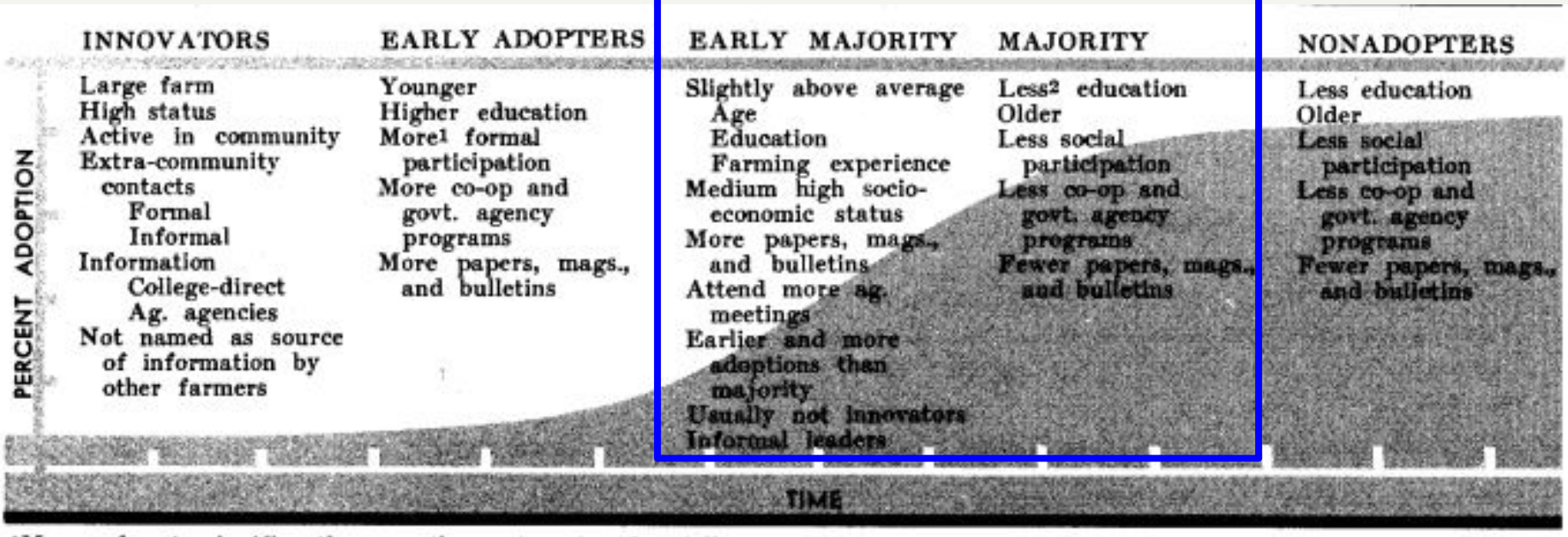
Service Infra / SRV-2

need monitoring that service versions match across the cluster

Early adopters



The majority



Education

1. The **network** is reliable.
2. **Latency** is zero.
3. **Bandwidth** is infinite.
4. The network is **secure**.
5. **Topology** doesn't change.
6. There is one **administrator**.
7. Transport cost is zero.
8. The network is homogeneous.

https://en.wikipedia.org/wiki/Fallacies_of_distributed_computing



Service Principles

- Service Principles
 - Creation
 - Discuss the organization of your overall system first
 - Check whether you can add your feature to an existing service
 - Consider whether your feature is better suited to a library
 - Services are curated by teams, not individuals
 - Services are a long-term commitment
 - Factor in the overhead of deploying a distributed system
 - Prefer larger services
 - Minimize the depth of the service call-graph
 - Minimize the number of services owned by your team
 - Interfaces
 - Interfaces should be easy to understand
 - Interfaces should be robust
 - Changes to interfaces should be backwards compatible
 - Testing
 - Any changes to your service should be able to be tested automatically
 - Your interface is the most important thing to test
 - Operations
 - You are responsible for running your service
 - Guide your clients' expectations
 - Plan for failure
 - Additional Reading

<https://github.com/Yelp/service-principles>



SCFXX: Your Service Name Here

Pronounced "skiff"

This is the template that must be filled out for any new service. This is not a CEP and doesn't replace the role of a CEP. This details the information about your service necessary for Operations to setup your service for deployment.

- Service type (PaaS vs legacy SOA)
- Technology type (Python / Java / daemon)
- Caching
- Downtime scenarios
- Databases
- External services
- Hardware requirements
- Load balancing
- Performance and uptime
- Monitoring
- Security

The Tutorial

- [1. Write a design and SCF](#)
- [2. Create your service repo](#)
- [3. Write your first Pyramid endpoint](#)
- [4. Write your first test](#)
- [5. Test your service continuously with Jenkins](#)
- [6. Use a database](#)
- [7. Swagger](#)
- [8. Proper Testing](#)
- [9. Acceptance Testing](#)
- [10. Resiliency Testing](#)
- [11. Write a healthcheck endpoint](#)
- [12. Zipkin Tracing](#)
- [13. Deployment For PaaS Services!](#)
- [14. Monitoring your service](#)
- [15. Document your service](#)
- [16. Call your service from yelp-main](#)
- [17. Retries and Timeouts](#)
- [18. Add a Batch for PaaS](#)
- [19. Add a Batch for Classic SOA \(Deprecated\)](#)
- [20. Add a Yelp frontend](#)
- [21. Localize your service](#)

Summits and unconferences

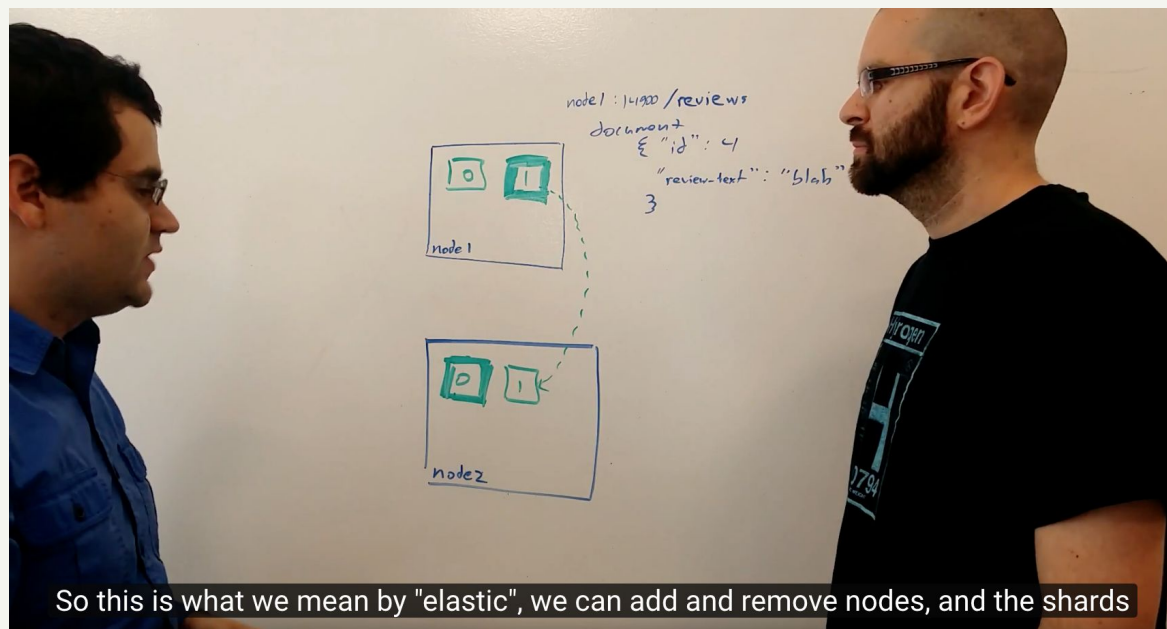
Technical Leadership Summit

Two days of technical sessions to help share knowledge, skills, and talk the direction of engineering at Yelp.

Tech Leadership Unconference 2016

An afternoon unconference of Technical Leaders at Yelp discussing topics relevant to all of Engineering

Talks



Deputy programs

Manual processes are a common form of technical debt afflicting software development teams. Yelp Deputies are members of other teams who are granted extra authority and responsibility over systems owned by teams like [ReleaseEng](#) and [Operations](#). Deputies help fight the scourge of manual processes:

Office hours

Core Backend Office Hours



When

Mon, June 6, 2:30pm – 3:30pm

MySQL/DBA Office Hours



When

Fri, June 17, 3:00pm – 3:50pm

PaaSTA/Service Office Hours



When

Tue, June 7, 2:30pm – 3:30pm

Metrics + Splunk Office hours



When

Wed, June 15, 2pm – 3pm

Backend Sync



When

Wed, June 8, 10:00am – 10:45am

Operations Office Hours



When

Fri, June 10, 3pm – 4pm

Postmortems

postmortem Shared privately

30 of 844 topics (99+ unread) ☆

Members · About ▾

Welcome to the postmortem group, where Yelp's engineering failures and triumphs go to be remembered for all eternity.

"A calm despair, without angry convulsions or reproaches directed at heaven, is the essence of wisdom" - Alfred de Vigny

☆	💬	Mini-Mortem: Seagull sandbox suites taking longer than their usual run time (SEAGULL-1595) (1)	1 post	Jun 6
☆	💬	Postmortem: (KAFKA-1118) Standard Kafka cluster service degraded after 0.9 upgrade (1)	1	Jun 6

Boo!


Boring!

Standards

Tell me
something
interesting!

Zzzz.....!

Interfaces

 **y/swagger** [Old Version](#)

internalapi_2dot0

Internalapi Service

activity_feed : Activity Feed APIs[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

ads : Ad related APIs[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

biz_user : Business User APIs[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

GET

/biz_user/{biz_user_id}/business_ids/v1

(biz_user_business_ids) Returns a list of all the ids of businesses that a biz_user has access to.

GET

/biz_user/{biz_user_id}/has_approved_photo

(has_approved_photo) Return whether or not the biz_user has an approved photo.

GET

/biz_user/{biz_user_id}/last_viewed_businesses

(biz_user_last_viewed_businesses) Get last viewed businesses for given biz user, excludes given business_id.

POST

/biz_user/get_biz_user_id_by_email

(get_biz_user_id_by_email) Get biz id given an email.

biz_user_business : Biz User Business APIs[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

bookmarks : Bookmark related APIs[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

GET

/photos/v2/list

(list) Get a list of biz photos

Response Class (Status 200)

Model | Model Schema

```
[
  {
    "url_prefix": "string",
    "user_id": 0,
    "review_id": 0,
    "uploading_user_type": "user",
    "business_id": 0,
    "time_created": 0,
    "enc_user_id": "string",
    "caption": "string",
    "encrypted_id": "string"
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
photo_ids	<input type="text" value="1"/>	A comma-separated list of photo ids.	query	string

Try it out!

[Hide response](#)

Perform a real query



Tracing

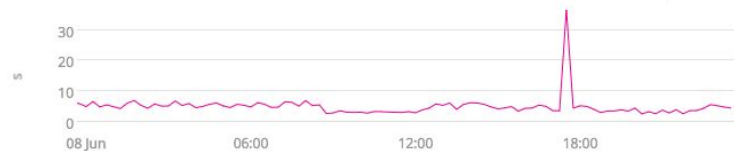
Services	561.284ms	1.123s
- yelp_main/api	2.806s : get /search	-
logic	2.419ms : txn: _is_ip_blocked_from_login	-
core_memcache	341μ : get user_session	-
core_memcache	198μ : get employee_department_cache-20140520	-
+ logic	3.043ms : txn: user_get_basic_and_scout_info	-
logic	6.048ms : txn: by_longitude_latitude	-
core_memcache	1.665ms : set user_latest_search_cache-20110819	-
+ adddelivery	187.000ms : post /v2/pick_ads	-
+ logic	9.087ms : txn: user_get_info	-
- logic	414.998ms : txn: business_search_with_review_and_user_photo	-
core_memcache	463μ : get logic_load_row	-
federator	318.000ms : get /v1/query	-
lucy	24.000ms : post /lucy_jersey/name_query	-
lucy	17.000ms : post /lucy_jersey/name_query	-
lucy	33.000ms : post /lucy_jersey/name_query	-
lucy	19.000ms : post /lucy_jersey/name_query	-
lucy	17.000ms : post /lucy_jersey/name_query	-
lucy	207.000ms : post /lucy_jersey/composite_query	-
lucy	150.000ms : post /lucy_jersey/composite_query	-
lucy	173.000ms : post /lucy_jersey/composite_query	-
lucy	186.000ms : post /lucy_jersey/composite_query	-
lucy	190.000ms : post /lucy_jersey/composite_query	-
core_memcache	7.369ms : get_multi short_highlights_cache20141010	-

Metrics

Kew Core production metrics ▾

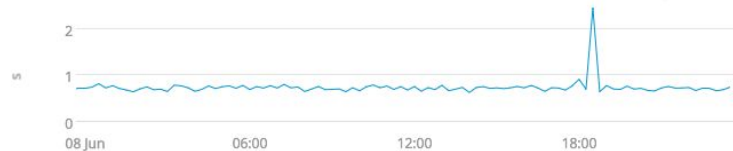
kew_task_async end-to-end latencies

End-to-end latency for kew_task_async calculated using kewala



kew_task_sync end-to-end latencies

End-to-end latency for kew_task_sync calculated using kewala



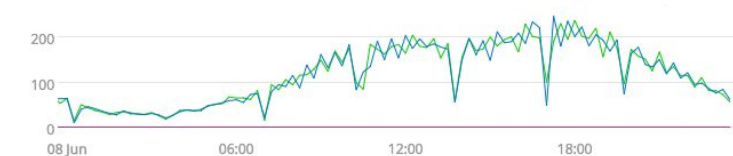
Kafka-scribe deltas

Difference between high watermark and consumer offset for the kew topic partitions in kafka-scribe



Kafka-standard deltas

Difference between high watermark and consumer offset for the kew topic partitions in kafka-standard



Datastores



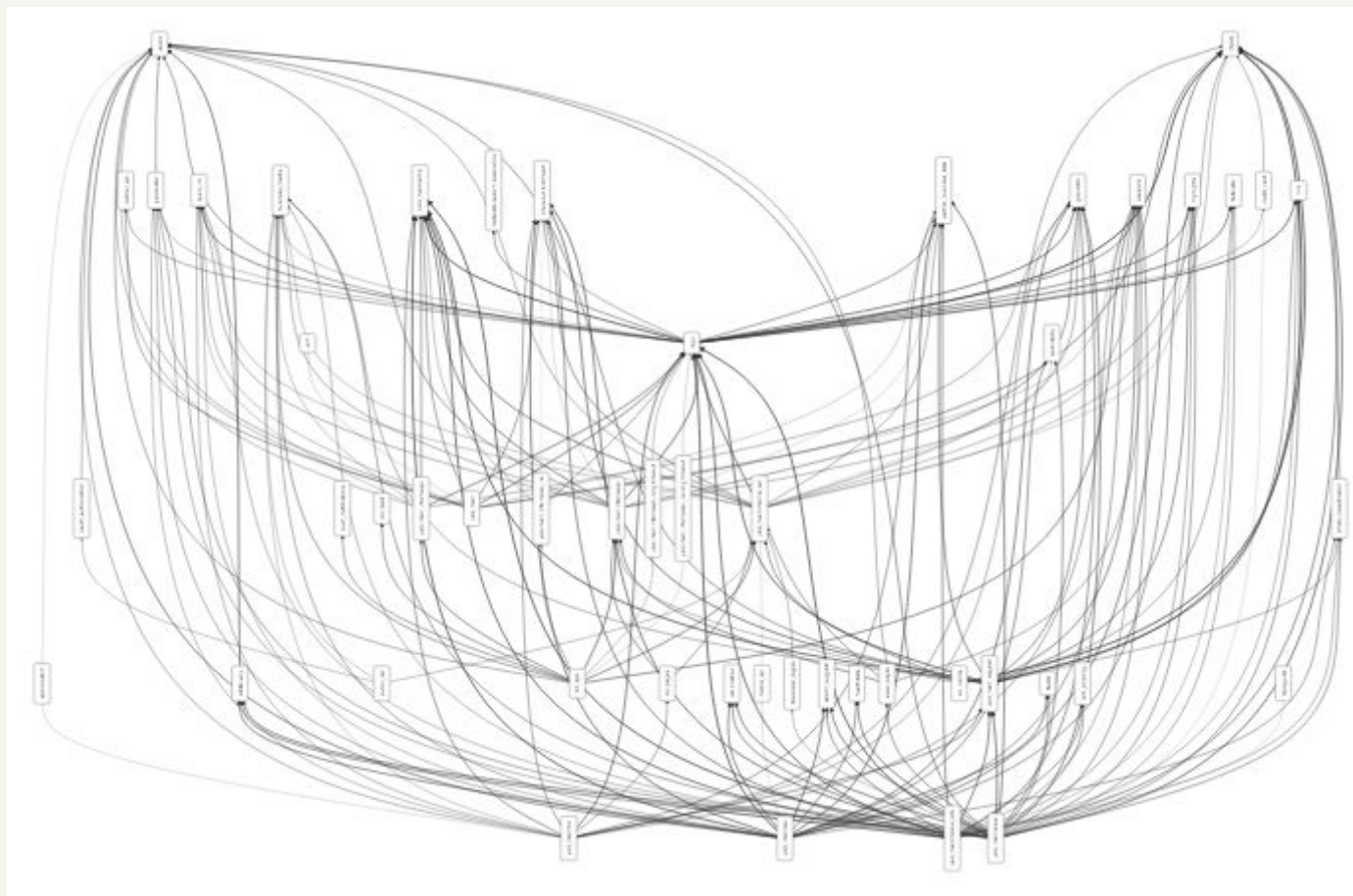
Timeouts and retries

```
C:\>DIR A:
```

```
Not ready reading drive A  
Abort, Retry, Fail?_
```

https://en.wikipedia.org/wiki/Abort,_Retry,_Fail%3F#/media/File:Abort_Retry_Fail.PNG

Ownership



[projects](#) / [yelp-main.git](#) / summary

summary | [shortlog](#) | [log](#) | [commit](#) | [commitdiff](#) | [tree](#)

description	Yelp main site codebase
owner	Gitolite User
last change	Tue, 7 Jun 2016 14:06:06 -0700 (14:06 -0700)

Organizational objectives

Lots of (potentially) conflicting objectives:

- Performance
- Reliability
- Cost
- Security
- Iteration speed

“What gets measured gets improved”

Peter Drucker

Conclusions

Microservices have forced us to build a more robust, decentralized organization

This change has taken a long time

I believe (but cannot prove) that we are shipping
code as quickly as ever

Any questions?