



Using distributed tracing to solve performance and operational challenges

Naoman Abbas

October 2, 2018

Naoman Abbas

Engineering Manager

Visibility

Metrics

Log Search

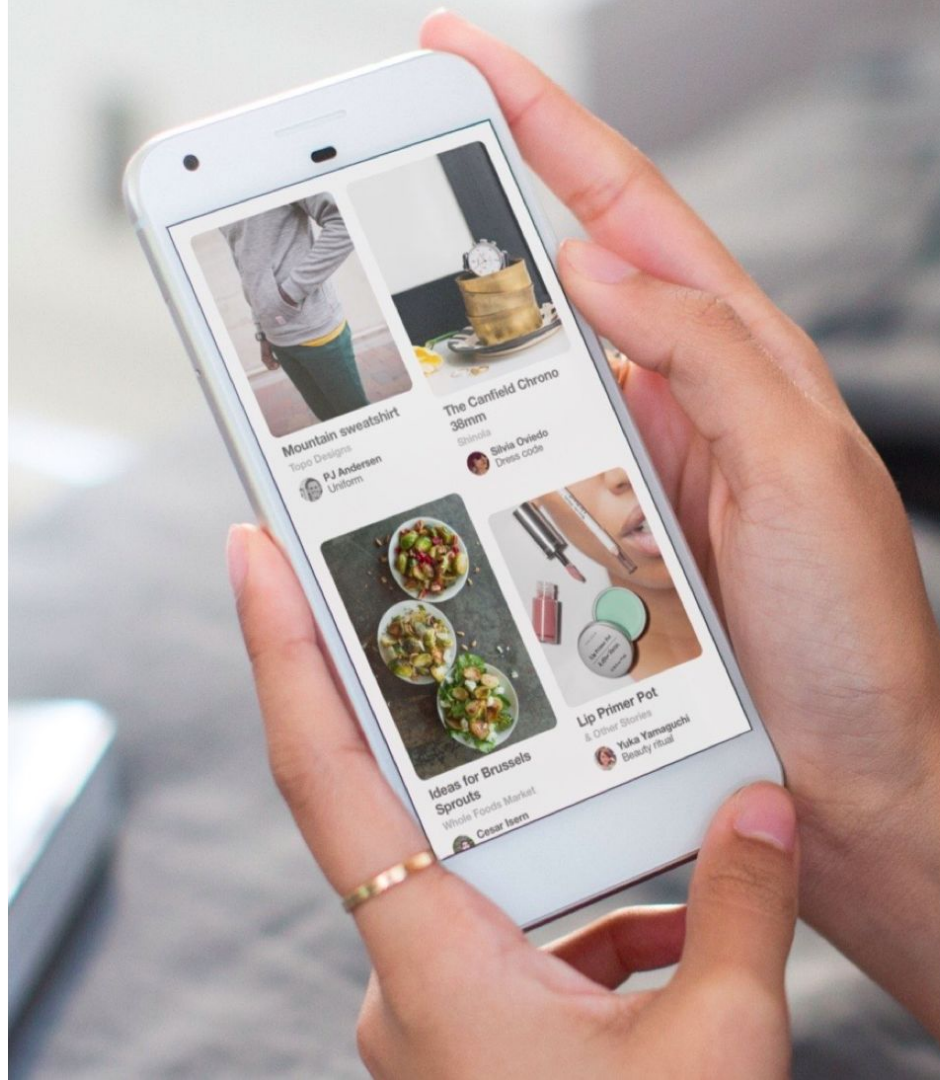
Distributed Tracing



Pinterest

Helping people discover
and do what they love

- **+250M** monthly active users
- **+3T** recommendations per year
- **+100B** pins
- hundreds of services



Beginner

Projects

Pattern

Backpack

Ideas

Tools

Tutorial

Template

Workbench

DIY

Bracelet

Bag

Design

Workshop

Furniture

Belts

Purse



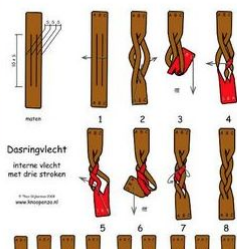
PERSONALIZED Gift, Treasure Chest Credit Card Wallet, Perfect Gift Idea, Unique, Handcrafted Excellence, ADD it to your FAVORITES #011



Mascon Leather
See more



This is it man! The idea I've kept having in mind. Definitely knew it wasn't original. LOL. What's new. // Leather Card Holder
See more
from Instagram



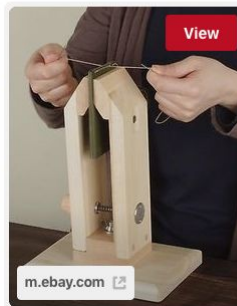
Ultimate Guide to Leather #infographic
Ultimate Guide to Leather #infographic #Leather #LifeStyle
See more



A Textbook of Leathercraft: Projects and Patterns - Leatherworking & Leathercraft - Crafts & Hobbies - PDF Classic Books, Online Bookstore
See more



Foldover Fob
Leather and Brass key holder
See more



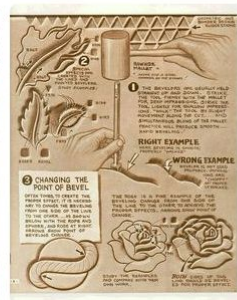
Leathercraft Table Pony Desktop Lacing Pony & Sewing Horse For Stitching Leather
See more



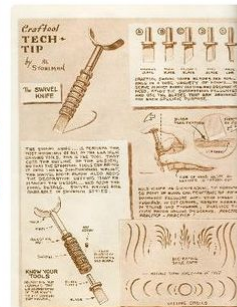
leather working table
See more



A hardened leather thimble to make stitching leather easier!
See more
from Instagram



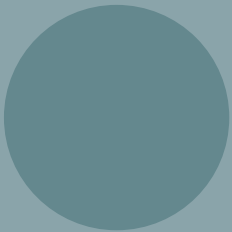
Mert Leathercraft, Leathercraft Pattern, Leatherwork Patterns, Leather Templates, Leather Tooling Patterns, Tooling Ideas, Patrons Leather, Leather Stuff, ...
See more



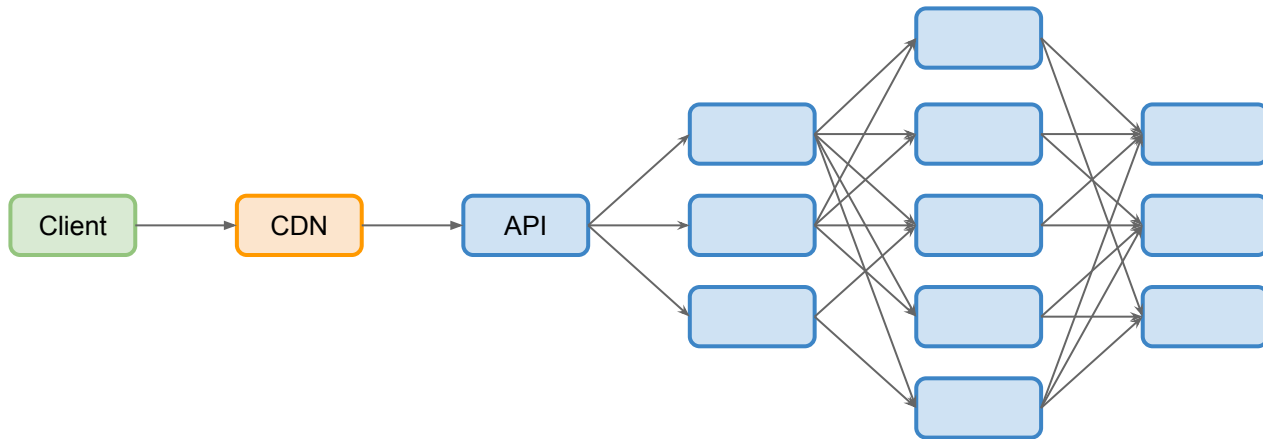
Agenda

- 1 **Intro**
- 2 **Pintrace**
- 3 **Tools**

Intro

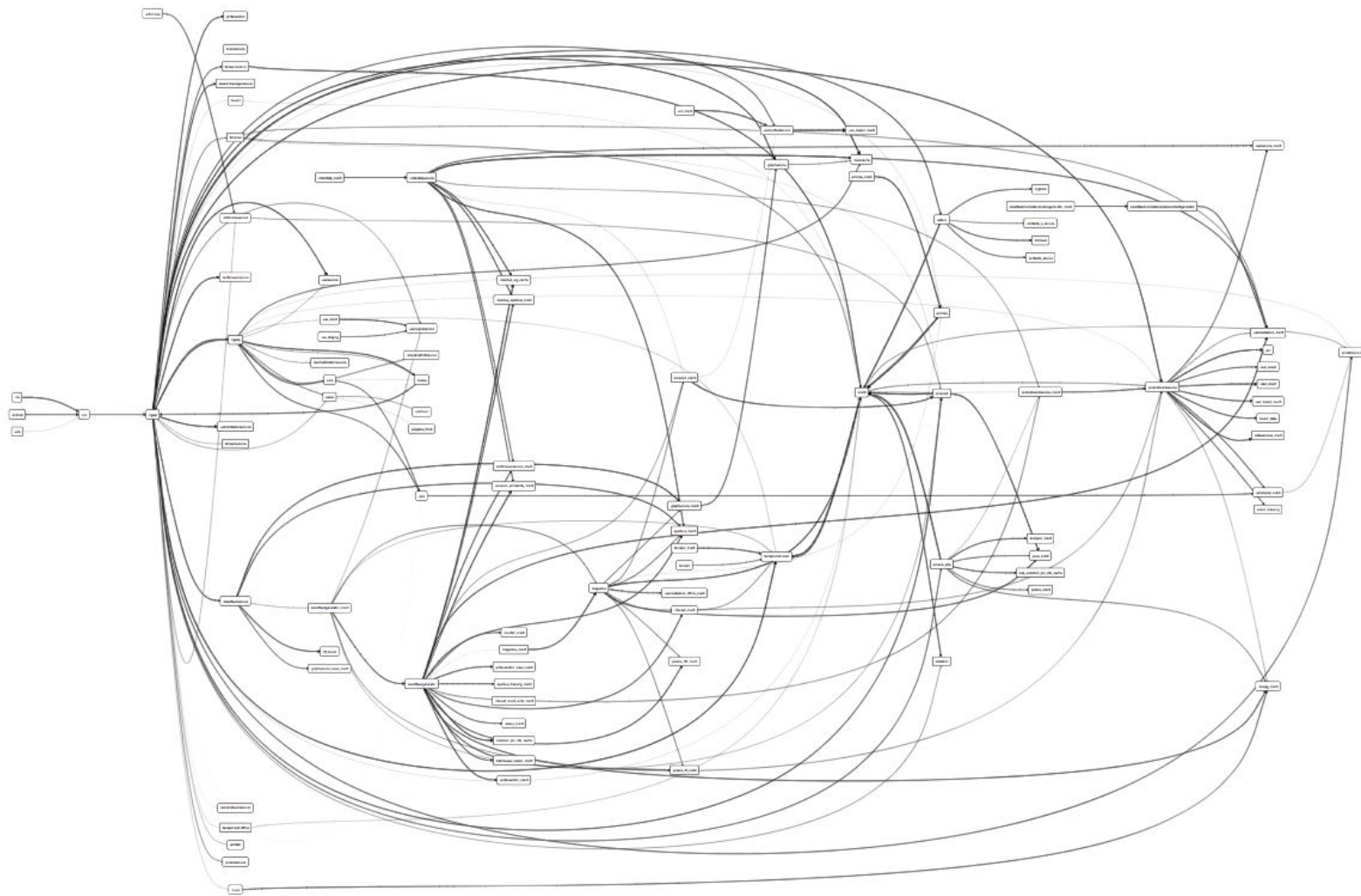


Why Distributed Tracing?



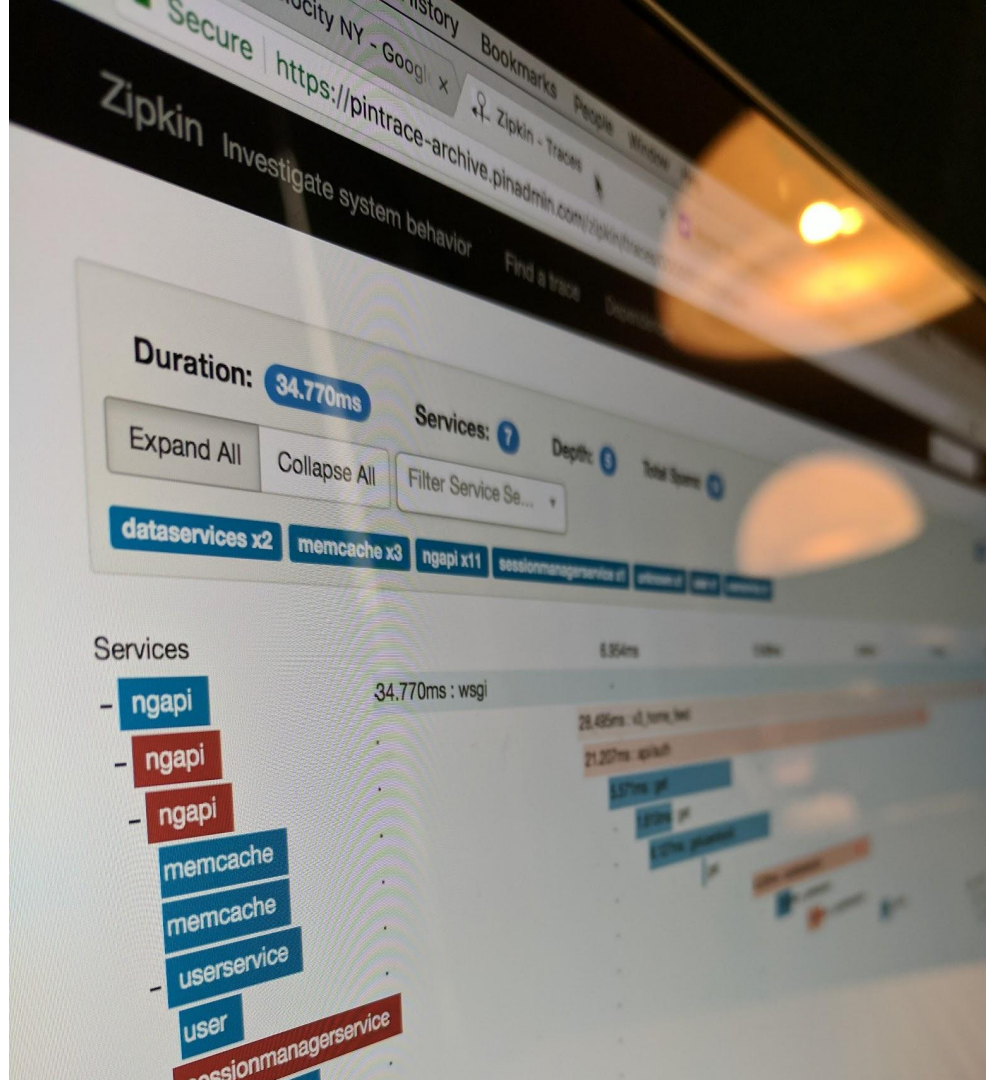
Microservice Architecture





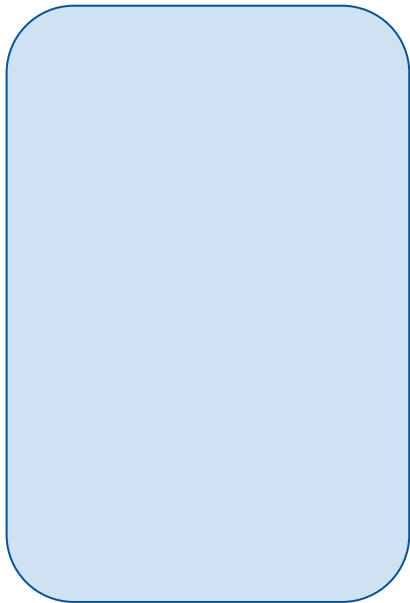
Distributed Tracing

- Performance tuning
- Root-cause analysis
- Operation

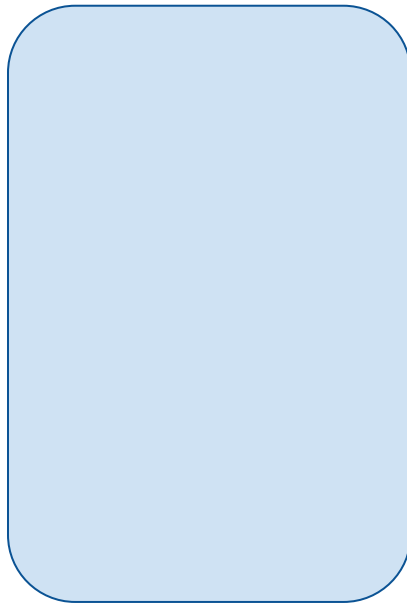


Distributed Tracing

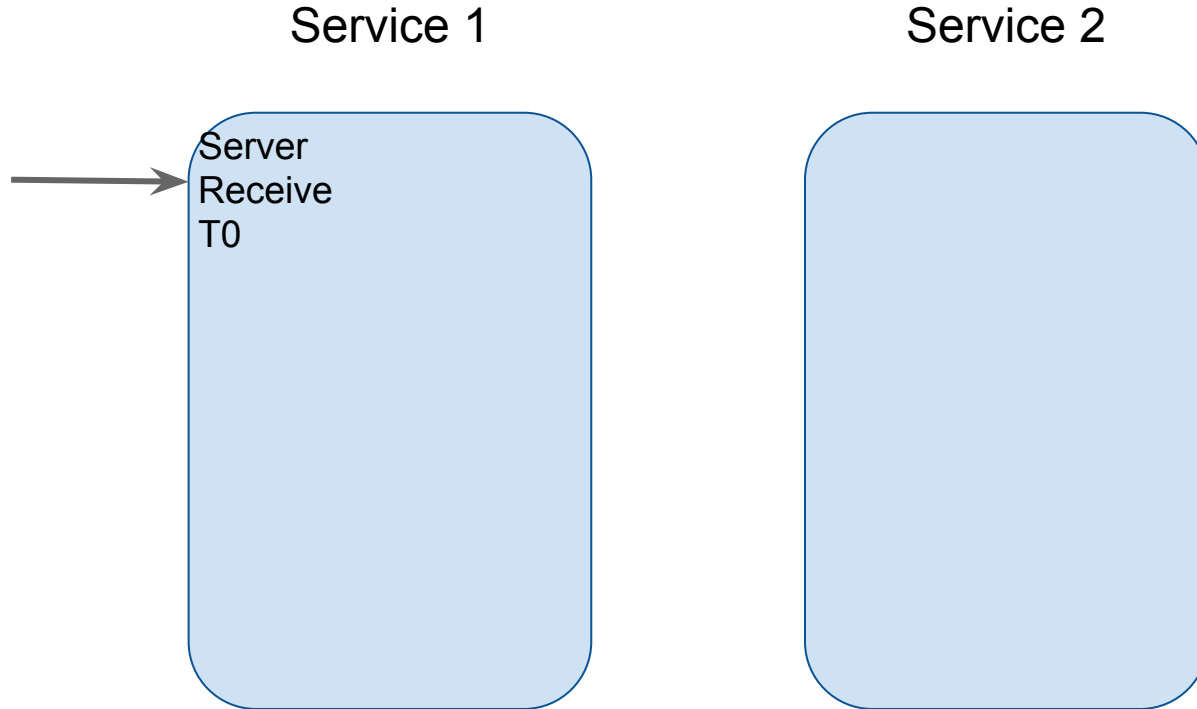
Service 1



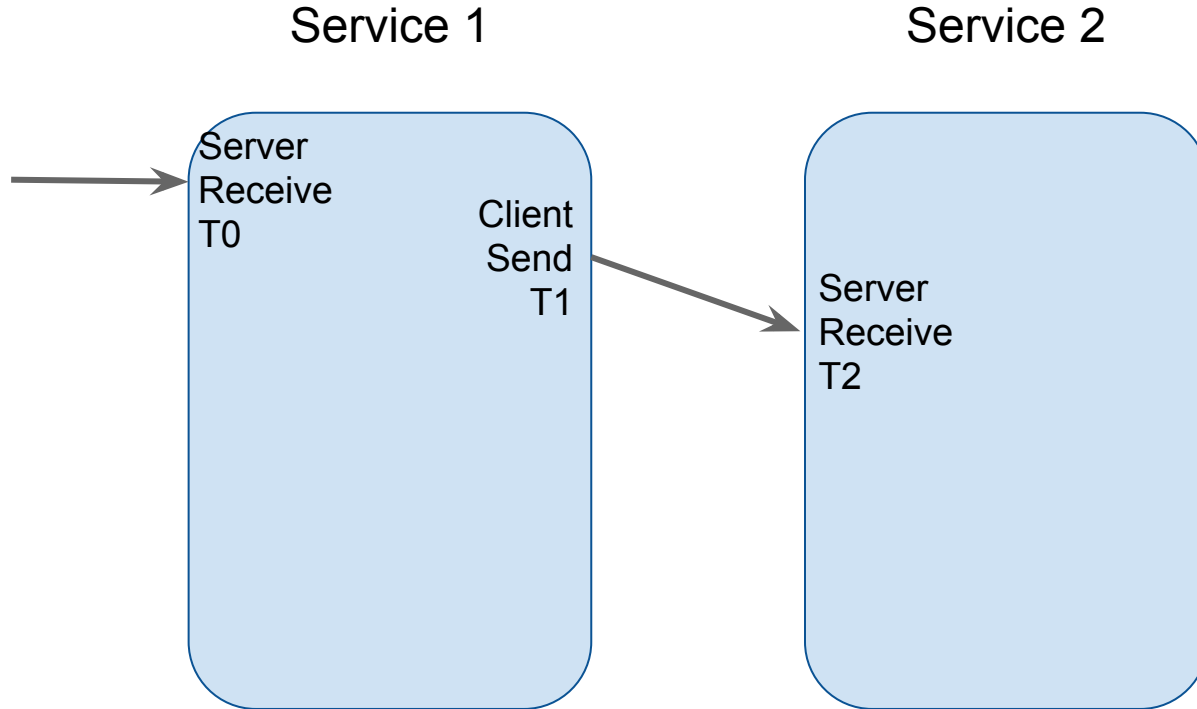
Service 2



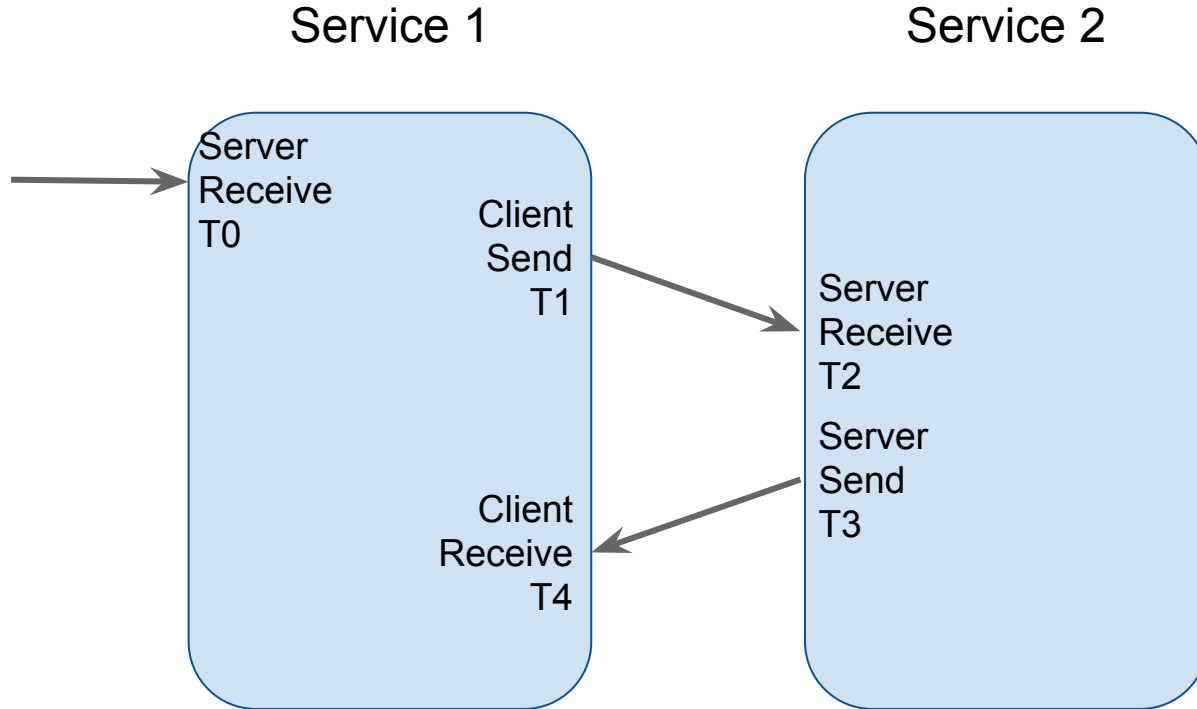
Distributed Tracing



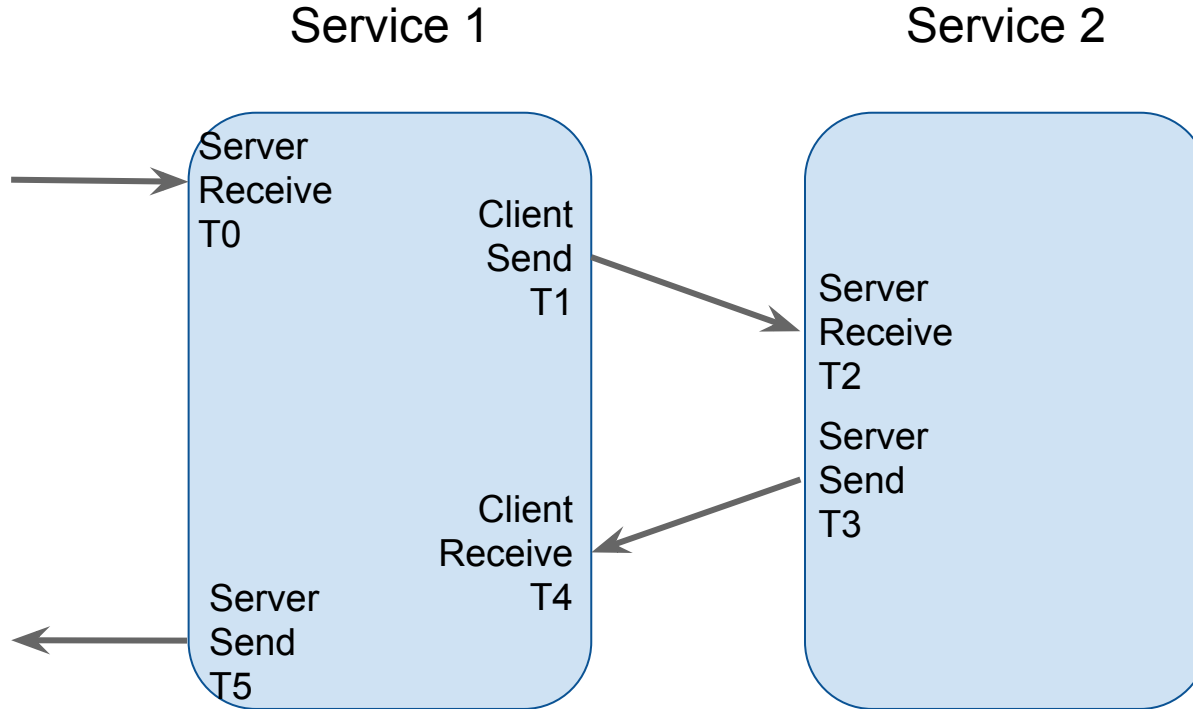
Distributed Tracing



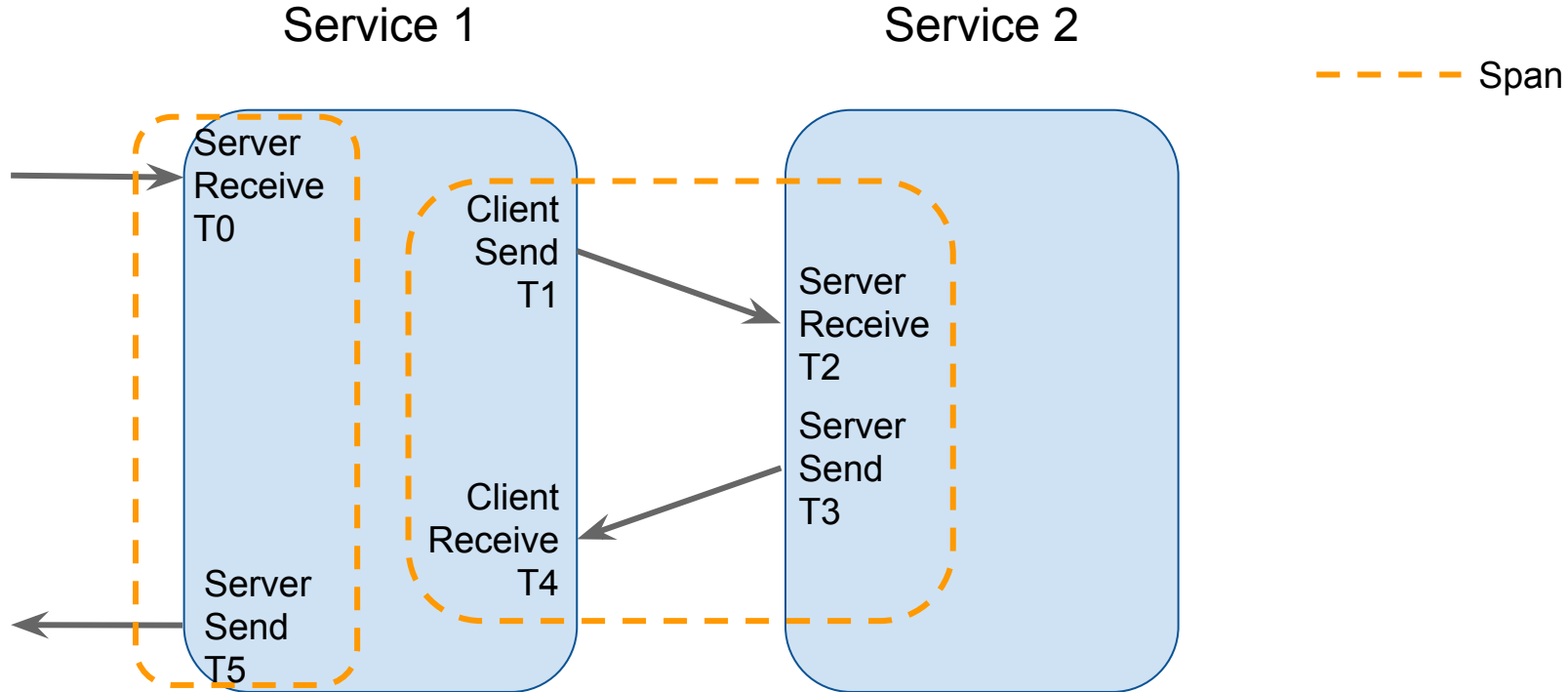
Distributed Tracing



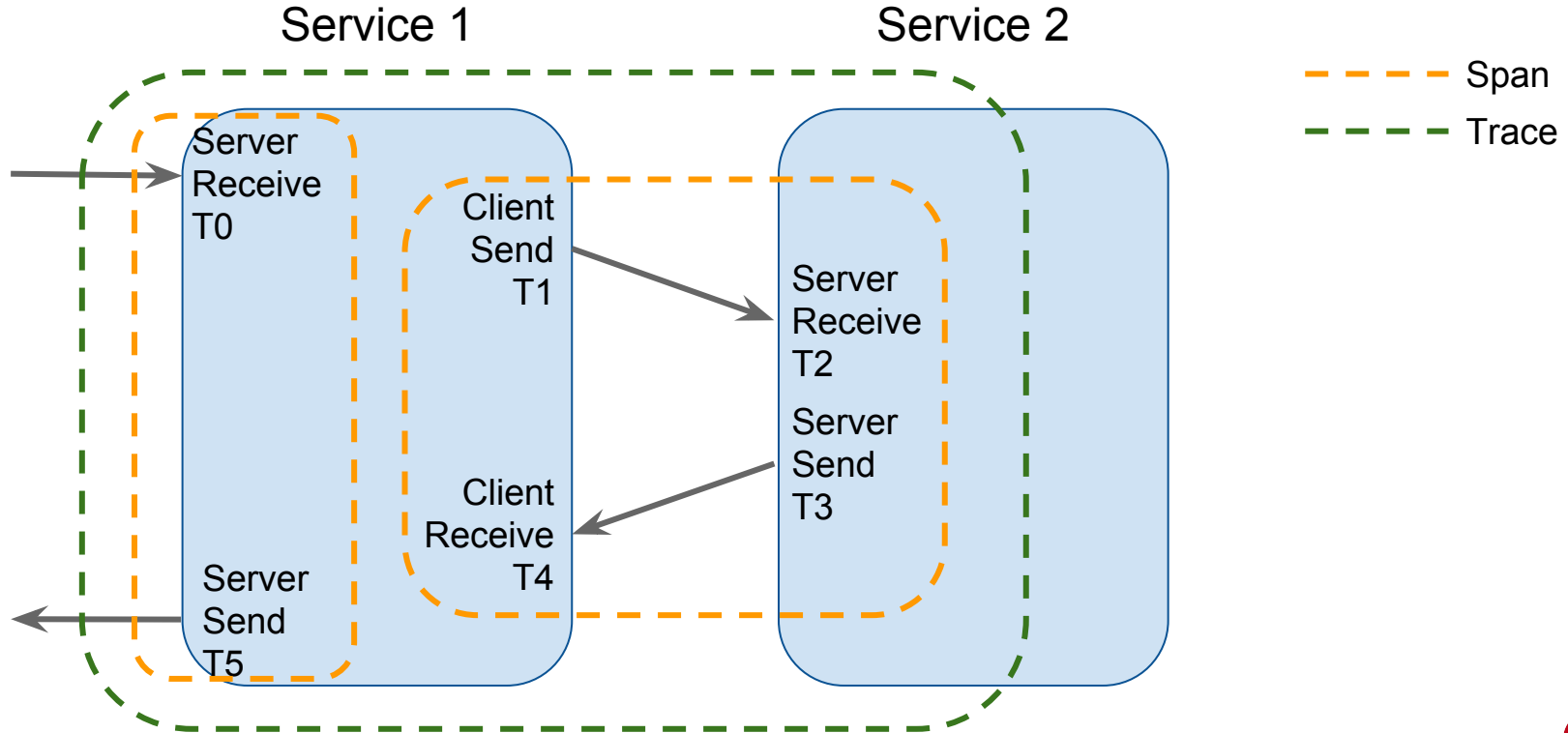
Distributed Tracing



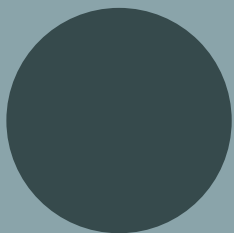
Distributed Tracing



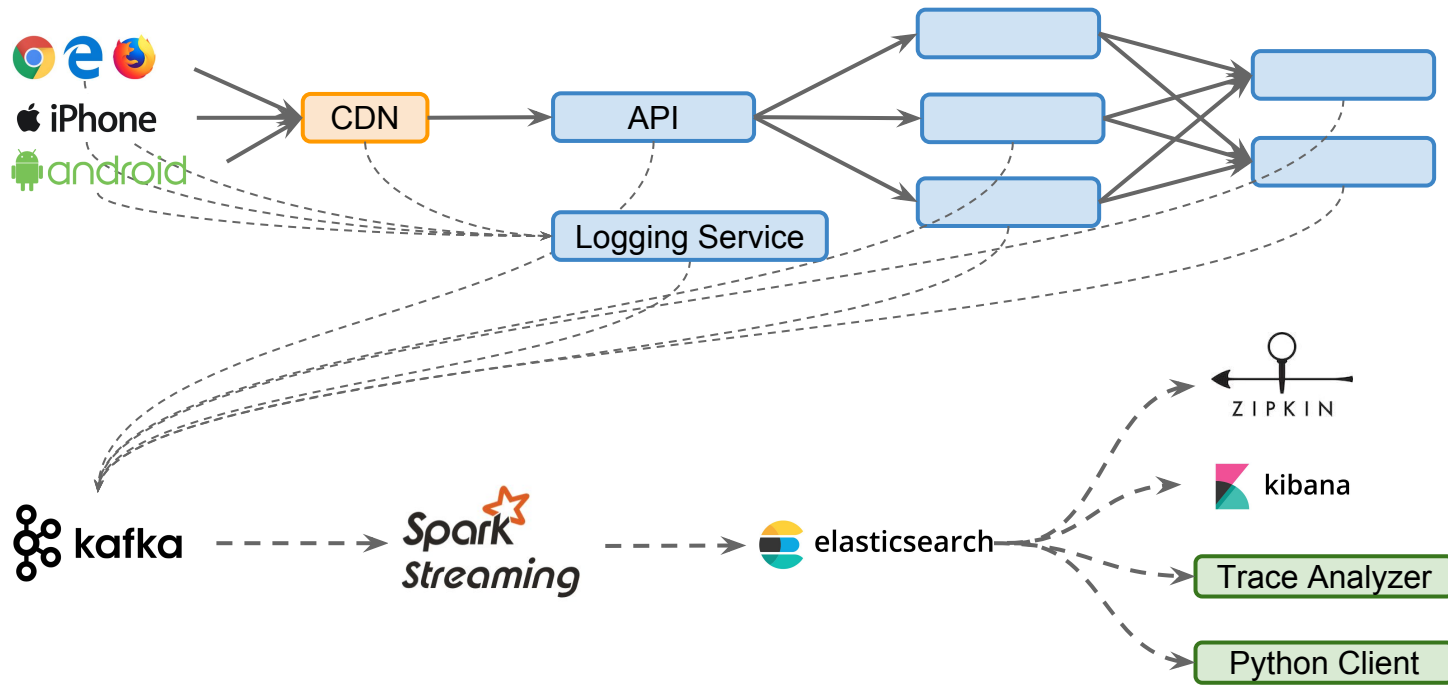
Distributed Tracing



Pintrance



Pintrace Architecture

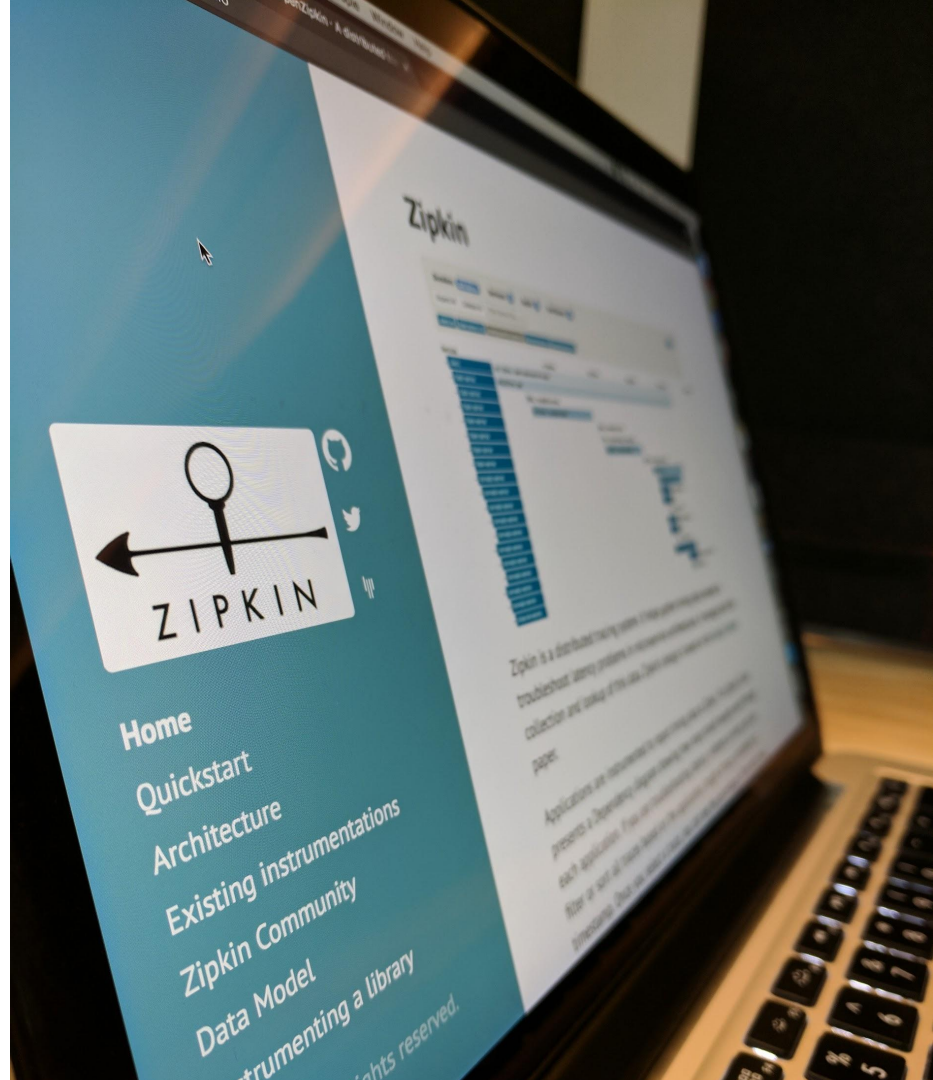


Tools



1. Zipkin UI

- Open source
- Search traces
- View trace details
- Latency analysis



Zipkin

Zipkin Investigate system behavior

Find a trace

Go to trace

ngapi

v3_home_feed

Start time

05-23-2018

14:58

End time

05-23-2018

15:58

Duration (µs) >=

Limit

10

Find Traces

?

Annotations Query (e.g. "finagle.timeout", "error", "http.path=/foo/bar/ and cluster=foo and cache.miss")

Showing: 0 of 0

Sort: Longest First

Services:

JSON

Please select the criteria for your trace lookup.



Duration: 770.000ms

Services: 23

Depth: 7

Total Spans: 68

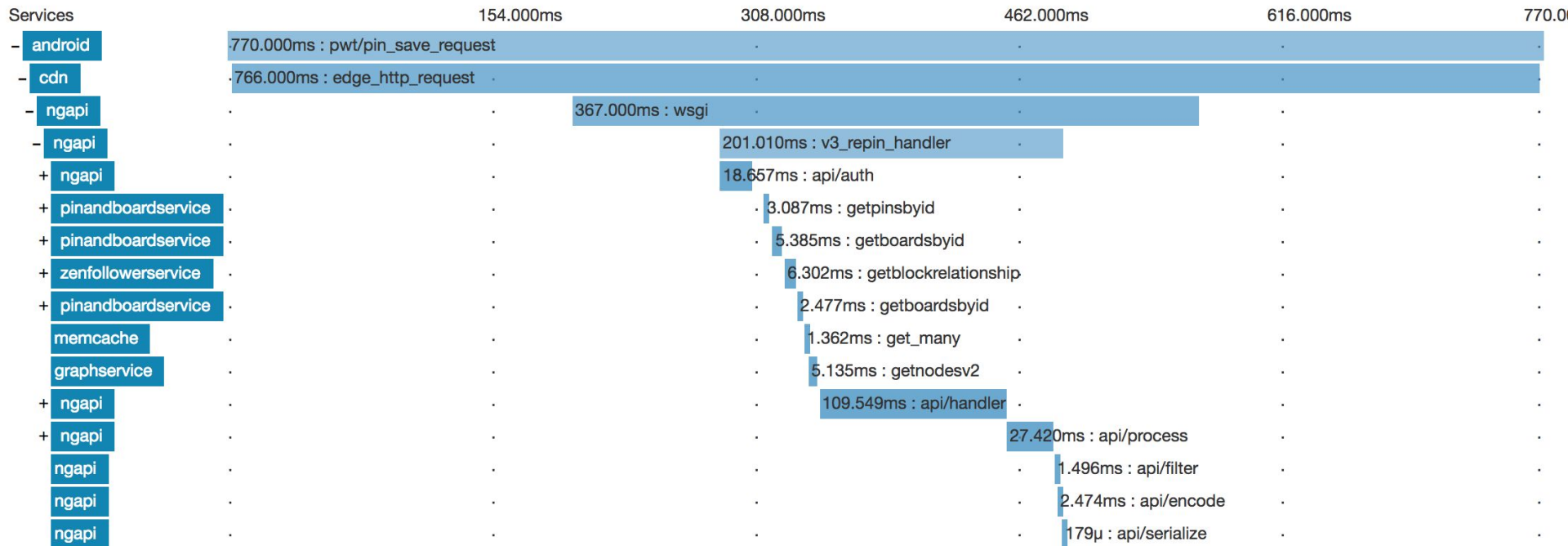
JSON

Expand All

Collapse All

Filter Service Se...

android x2 cdn x2 dataservices x5 dataservices_client x3 dataservices_tis x2 followerservice x1 graphservice x8 memcache x21 ngapi x54 pin x4 pinandboardservice x6
pinandboardservice_tis x6 pinlater x3 pinlaterservice x3 read_board x2 rio x2 unknown x6 user x2 usermetastoreservice x1 userservice x2 userservice_tis x1 zen_client x1
zenfollowerservice x1



Duration: 770.000ms

Services: 23

Depth: 7

Total Spans: 68

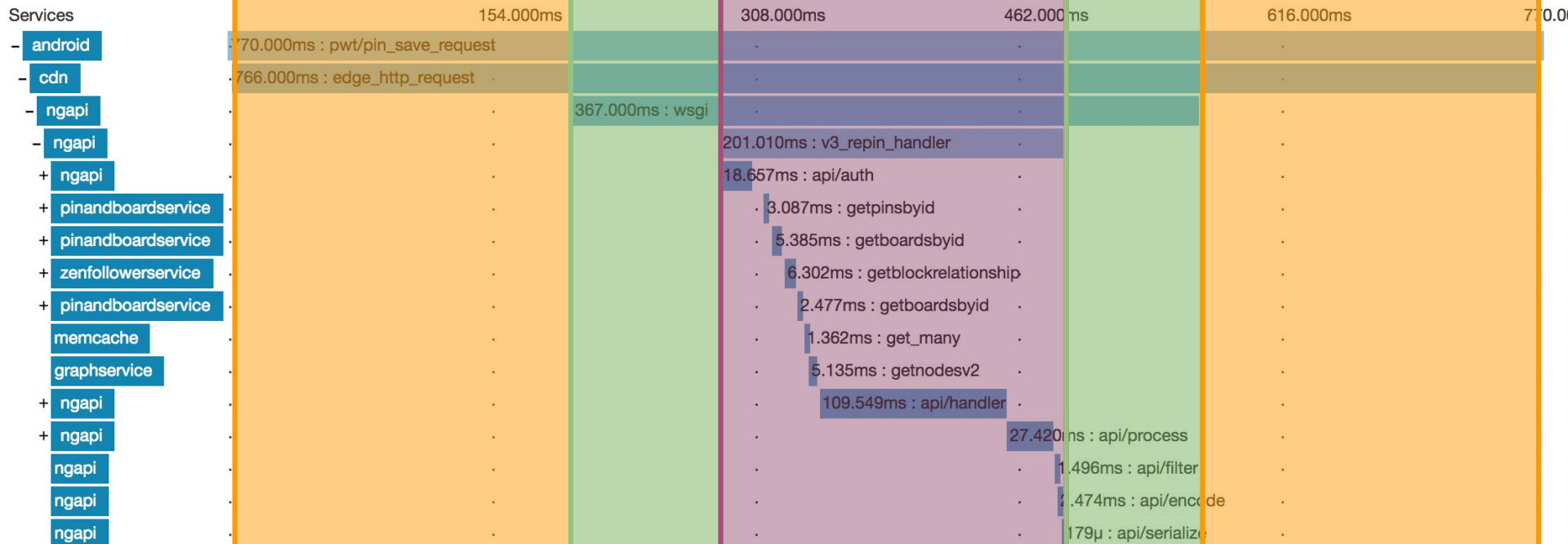
JSON

Expand All

Collapse All

Filter Service Se...

android x2 cdn x2 dataservices x5 dataservices_client x3 dataservices_tis x2 followerservice x1 graphservice x8 memcache x21 ngapi x54 pin x4 pinandboardservice x6
pinandboardservice_tis x6 pinlater x3 pinlaterservice x3 read_board x2 rio x2 unknown x6 user x2 usermetastoreservice x1 userservice x2 userservice_tis x1 zen_client x1
zenfollowerservice x1



Span Details

cdn.edge_http_request: 766.000ms			
AKA: android,cdn			
Date Time	Relative Time	Annotation	Address
5/22/2018, 5:05:09 PM	2.000ms	Client Send	246.6.69.0:0 (android)
5/22/2018, 5:05:09 PM	201.500ms	Server Receive	151.101.216.84:0 (cdn)
5/22/2018, 5:05:10 PM	568.500ms	Server Send	151.101.216.84:0 (cdn)
5/22/2018, 5:05:10 PM	768.000ms	Client Receive	246.6.69.0:0 (android)
Key	Value		
Client Address	190.224.247.0:0 (android)		
cacheLatencyMsec	367		
cache_hit	false		
edge_location	EZE		
edge_provider	fastly		
geo.country	AR		
geo.region	06		



Date Time	Relative Time	Annotation	Address
2/7/2018, 4:33:45 PM	14.751ms	Client Send	10.1.75.27:0 (ngapi)
2/7/2018, 4:33:45 PM	14.756ms	Wire Send	10.1.75.27:0 (ngapi)
2/7/2018, 4:33:45 PM	17.362ms	Wire Receive	10.1.75.27:0 (ngapi)
2/7/2018, 4:33:45 PM	17.442ms	error	10.1.75.27:0 (ngapi)
2/7/2018, 4:33:45 PM	19.694ms	Client Receive	10.1.75.27:0 (ngapi)

Key	Value
PintraceClient	1
clnt/mux/enabled	true
error	GraphServiceException
python.exception	GraphServiceException(errorCode=404, message=u"Node (nodeType:2359 propertyName:key) doesn't exist")
python.traceback	<pre> File "/usr/local/lib/python2.7/dist-packages/thrift_utils/mux/client.py", line 251, in _call_method desc, result.data) File "/usr/local/lib/python2.7/dist-packages/thrift_utils/mux/thrift_codec.py", line 5 0, in decode_call_result return desc.recv_method(self._client) File "/mnt/pinboard/services/graph_service/thrift_libs/GraphService.py", line 1489, in recv_getNodeByUniqueIndexV2 raise result.e </pre>



Duration: 34.770ms

Services: 7

Depth: 5

Total Spans: 14

JSON

Archive

Expand All

Collapse All

Filter Service Se... ▾

dataservices x2

memcache x3

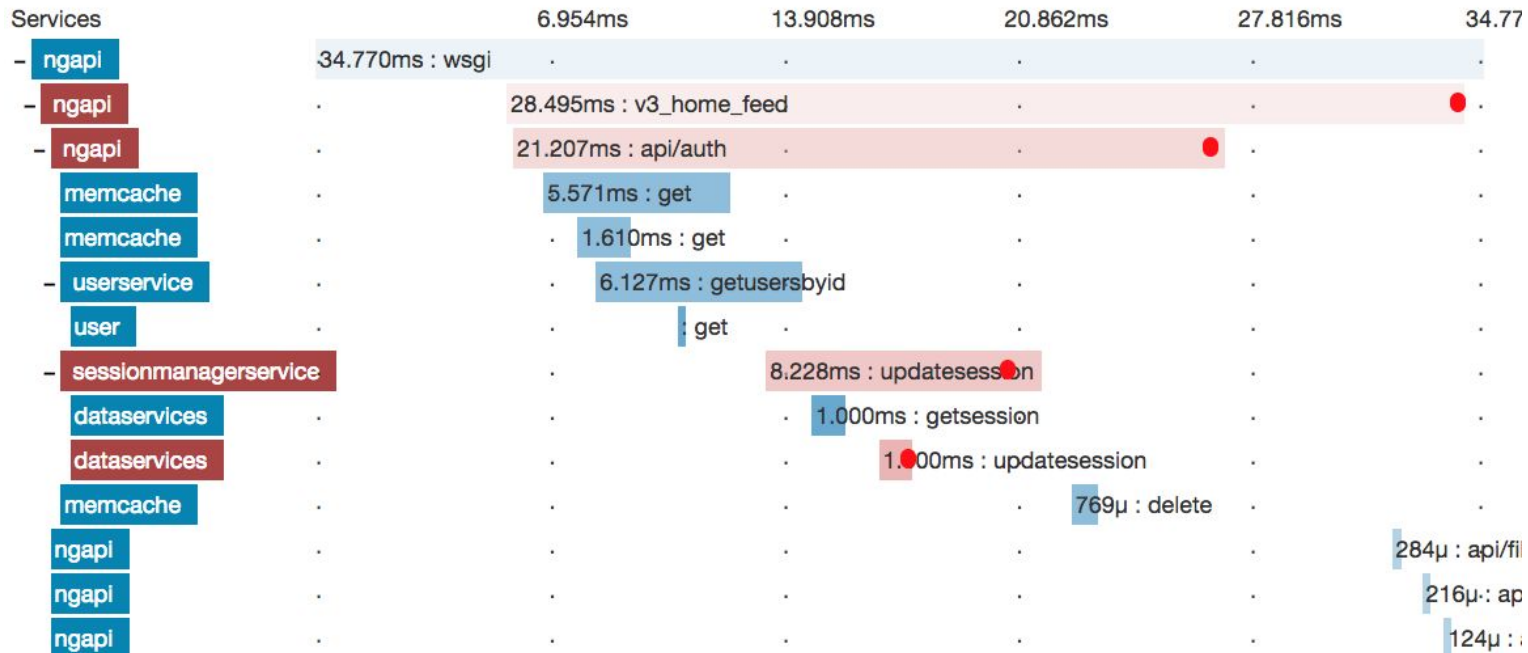
ngapi x11

sessionmanagerservice x1

unknown x1

user x1

userservice x1



313.849ms : fn_encode	.	.
946μ : thrift_getmediaresourcev2	.	.
2.064ms : thrift_getedgesforquery	.	.
2.390ms : thrift_getusermetadata	.	.
1.000ms : get	.	.
5.325ms : thrift_getpintuplesforboard	.	.
2.005ms : thrift_getedgesforquery	.	.
4.560ms : thrift_getusermetadata	.	.
1.000ms : get	.	.
1.916ms : thrift_getedgesforquery	.	.
2.080ms : thrift_getusermetadata	.	.
NaNs : get	.	.
2.133ms : thrift_getedgesforquery	.	.
2.409ms : thrift_getusermetadata	.	.
NaNs : get	.	.
2.127ms : thrift_getedgesforquery	.	.
2.826ms : thrift_getusermetadata	.	.
1.000ms : get	.	.
3.053ms : thrift_getedgesforquery	.	.
2.601ms : thrift_getusermetadata	.	.
NaNs : get	.	.
2.924ms : thrift_getedgesforquery	.	.
6.502ms : thrift_getusermetadata	.	.
1.000ms : get	.	.
2.098ms : thrift_getedgesforquery	.	.
2.287ms : getusermetadata	.	.
1.000ms : get	.	.
2.311ms : thrift_getedgesforquery	.	.
2.772ms : getusermetadata	.	.
1.000ms : get	.	.
2.078ms : thrift_getedgesforquery	.	.
2.823ms : thrift_getusermetadata	.	.



2. Kibana

- Visualize Elasticsearch data
- Finding trends
- Operations

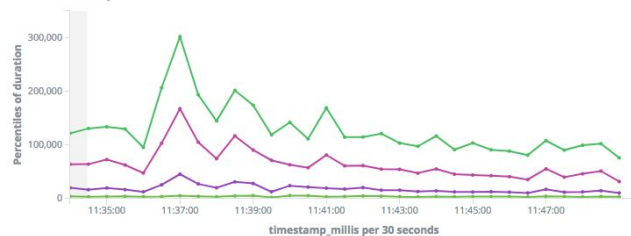


Kibana Dashboards

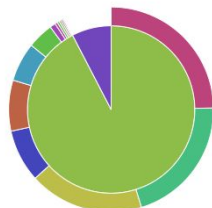
Pintrace - Spans per Service



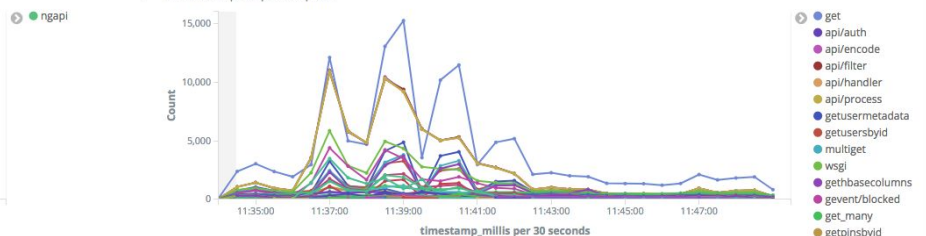
Pintrace - Latency Percentile



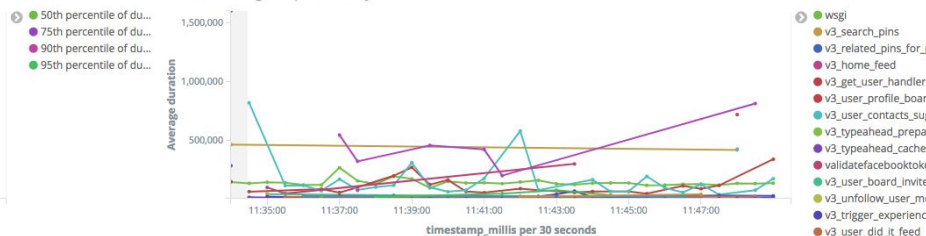
Pintrace - Dependencies



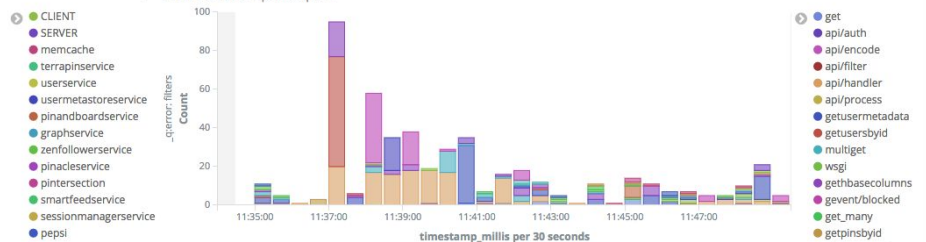
Pintrace - Spans per Endpoint



Pintrace - Average Endpoint Latency

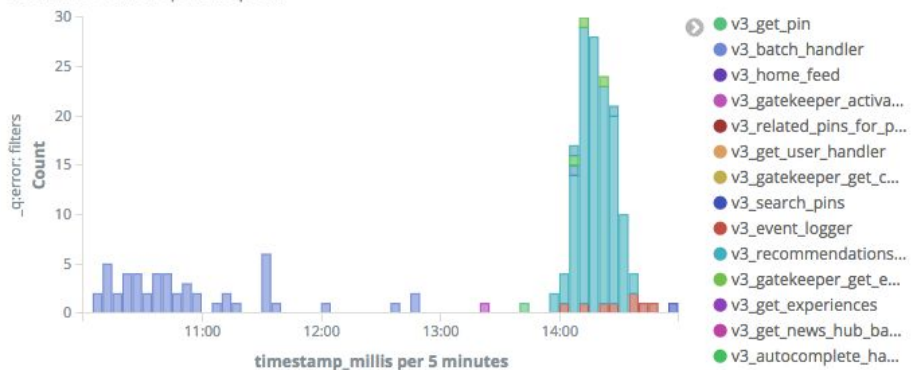


Pintrace - Errors per Endpoint

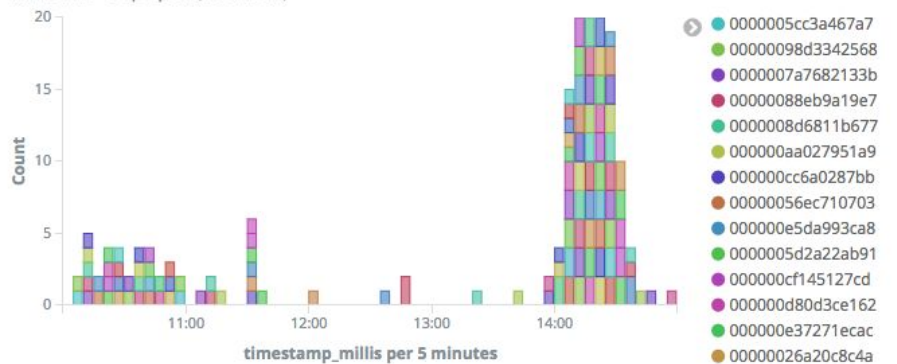


API Errors Dashboard

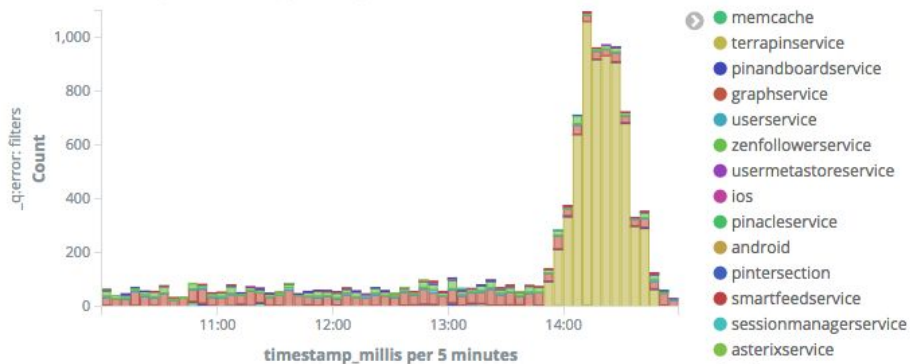
NGAPI V3 - Errors per Endpoint



NGAPI V3 - Top Span (V3 errors)

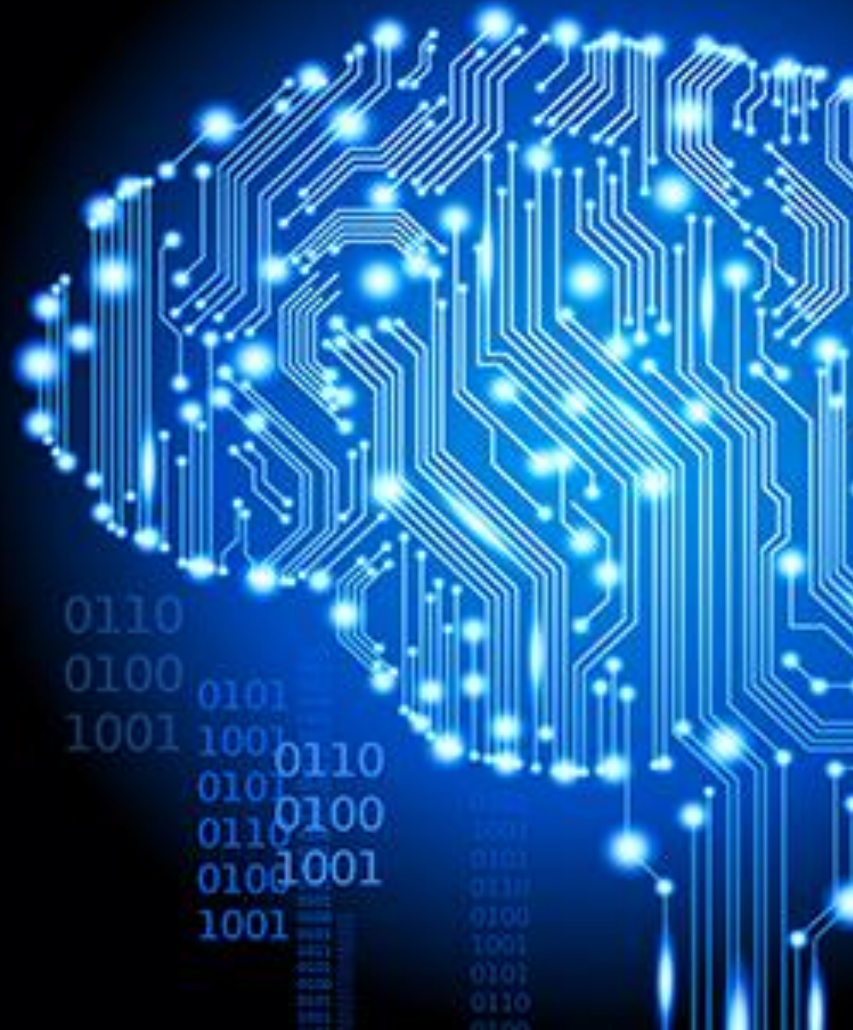


Pintrace - Errors per Service (reported by client)

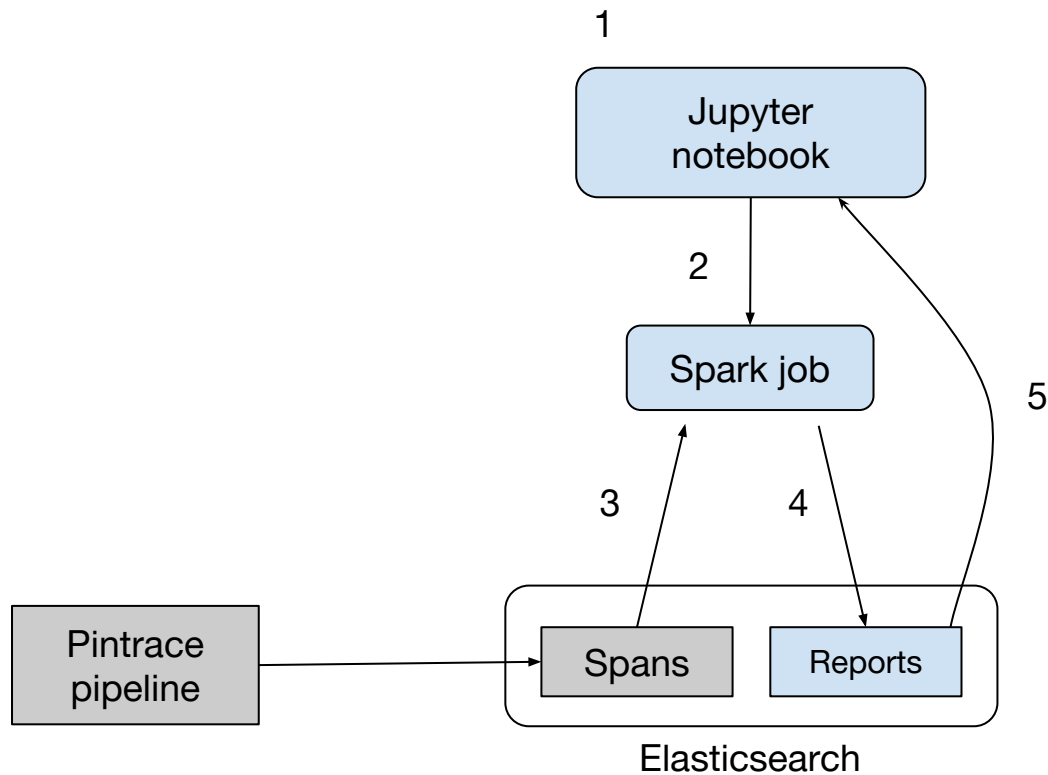


3. Trace Analyzer

- Offline analysis
- Aggregate view of traces
- Diff between sets of traces
- Root cause analysis



Architecture



Trace Analyzer - Input Parameters

E-mails	<input type="text"/>	Email addresses to send the report to.
---------	----------------------	--

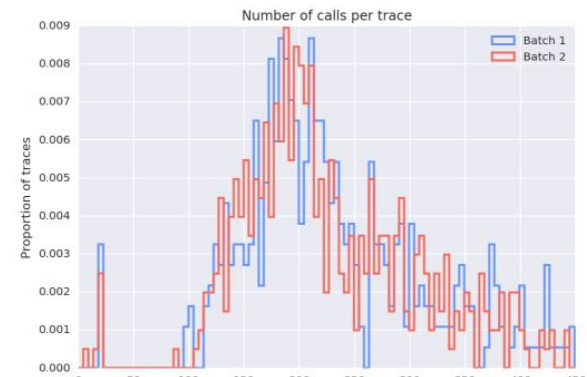
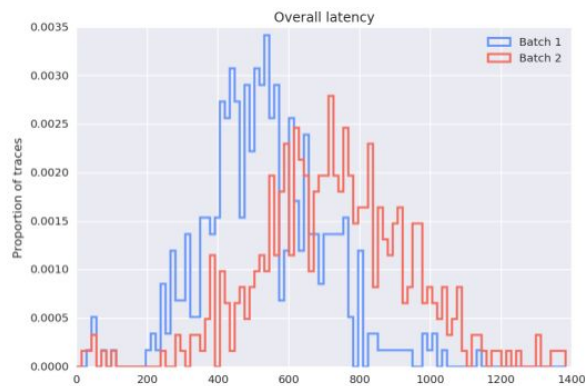
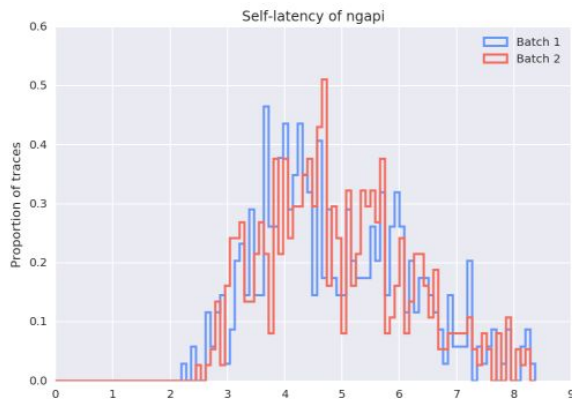
input_start	<input type="text" value="2018-02-06 13:00:00"/>	<input type="checkbox"/>	datetime*	Start time for first time window. Defaults to 1 day + 5 minutes ago.
input_end	<input type="text" value="2018-02-07 13:00:00"/>	<input type="checkbox"/>	datetime*	End time for first time window. Defaults to 1 day ago.
input_endpoint	<input type="text" value="v3_home_feed"/>		string	Optional NGAPI endpoint to filter down the first set of traces, such as "v3_home_feed".
input_binary	<input type="text" value="api.client.id.name,iPhone"/>		string	Optional "key,value" pair to filter down the first set of traces, based on annotation. For example "api.client.id.name,iPhone".
input_start2	<input type="text" value="2018-01-31 13:00:00"/>	<input type="checkbox"/>	datetime*	Start time for second time window. Defaults to 5 minutes ago.
input_end2	<input type="text" value="2018-02-01 13:00:00"/>	<input type="checkbox"/>	datetime*	End time for second time window. Defaults to now.
input_endpoint2	<input type="text" value="v3_home_feed"/>		string	Optional NGAPI endpoint to filter down the second set of traces, such as "v3_home_feed".
input_binary2	<input type="text" value="api.client.id.name,iPhone"/>		string	Optional "key,value" pair to filter down the second set of traces, based on annotation. For example "api.client.id.name,iPhone".
time_zone	<input type="text" value="US/Pacific"/>	<input type="checkbox"/>	string*	Time zone of input_start, input_end, etc.

* required field



Trace Analyzer - Report Summary

	First traces	Second traces	Difference	Percent difference
Number of traces	429	459	30	6.99301
Number of services	14	14	0	0
Average overall latency	573.322	806.25	232.928	40.6278
Calls per trace	244.07	235.928	-8.14183	-3.33586
Total number of calls	101902	107186	5284	5.18537
Average self-latency	4.71562	4.67538	-0.0402365	-0.853259



Trace Analyzer - Report

Downstream Services Latency

	First traces	Second traces	Difference
Service1	99.3975	118.445	19.0478
Service2	88.9196	92.2861	3.36652
Service2	67.0345	89.5526	22.5181
...	65.3212	52.185	-13.1362
...	10.4688	10.6408	0.172027
...	10.4142	10.8673	0.453135
...	9.80483	10.8275	1.02263
...	9.07438	8.88087	-0.193505
...	7.60317	8.72428	1.12111

Downstream Services Calls

	First traces	Second traces	Difference
Service1	764	1350	586
Service2	5526	4891	-635
...	1924	1723	-201
...	596	400	-196
...	530	236	-294
...	79	65	-14
...	34	29	-5
...	32	26	-6



4. Python Library

- Ad hoc analysis
- Simplify data access
- Simplify data processing

```
31
32 self.file = None
33 self.fingerprints = set()
34 self.logdups = True
35 self.debug = debug
36 self.logger = logging.getLogger(__name__)
37 if path:
38     self.file = open(os.path.join(path, 'fingerprint.log'), 'a')
39     self.file.seek(0)
40     self.fingerprints.update(e.request_fingerprint(request) for e in requests)
41
42 @classmethod
43 def from_settings(cls, settings):
44     debug = settings.getbool('debug')
45     return cls(job_dir(settings), debug)
46
47 def request_seen(self, request):
48     fp = self.request_fingerprint(request)
49     if fp in self.fingerprints:
50         return True
51     self.fingerprints.add(fp)
52     if self.file:
53         self.file.write(fp + os.linesep)
54
55 def request_fingerprint(self, request):
56     return request_fingerprint(request)
```


Python Library

query spans

```
end_time = datetime.datetime.now()
```

```
start_time = end_time - datetime.timedelta(minutes=1)
```

```
query = EsQuery().set_endpoint("v3_home_feed").set_local_service_name("ngapi").set_start_end_time(start_time, end_time) \
    .set_annotation("api.client.id.name=Android").set_source_fields("traceld")
```

```
es_client = EsClient(ES_ENDPOINT, ES_INDEX)
```

```
scroll_response = es_client.get_with_scroll(query)
```

```
trace_ids = list(map(lambda span: span.trace_id, scroll_response.spans))
```

query traces

```
get_traces_response = es_client.get_traces_for_span_ids(trace_ids)
```

group spans by trace id

```
spans = get_traces_response.spans
```

```
traces = defaultdict(list)
```

```
for span in spans:
```

```
    traces[span.trace_id].append(span)
```

process traces

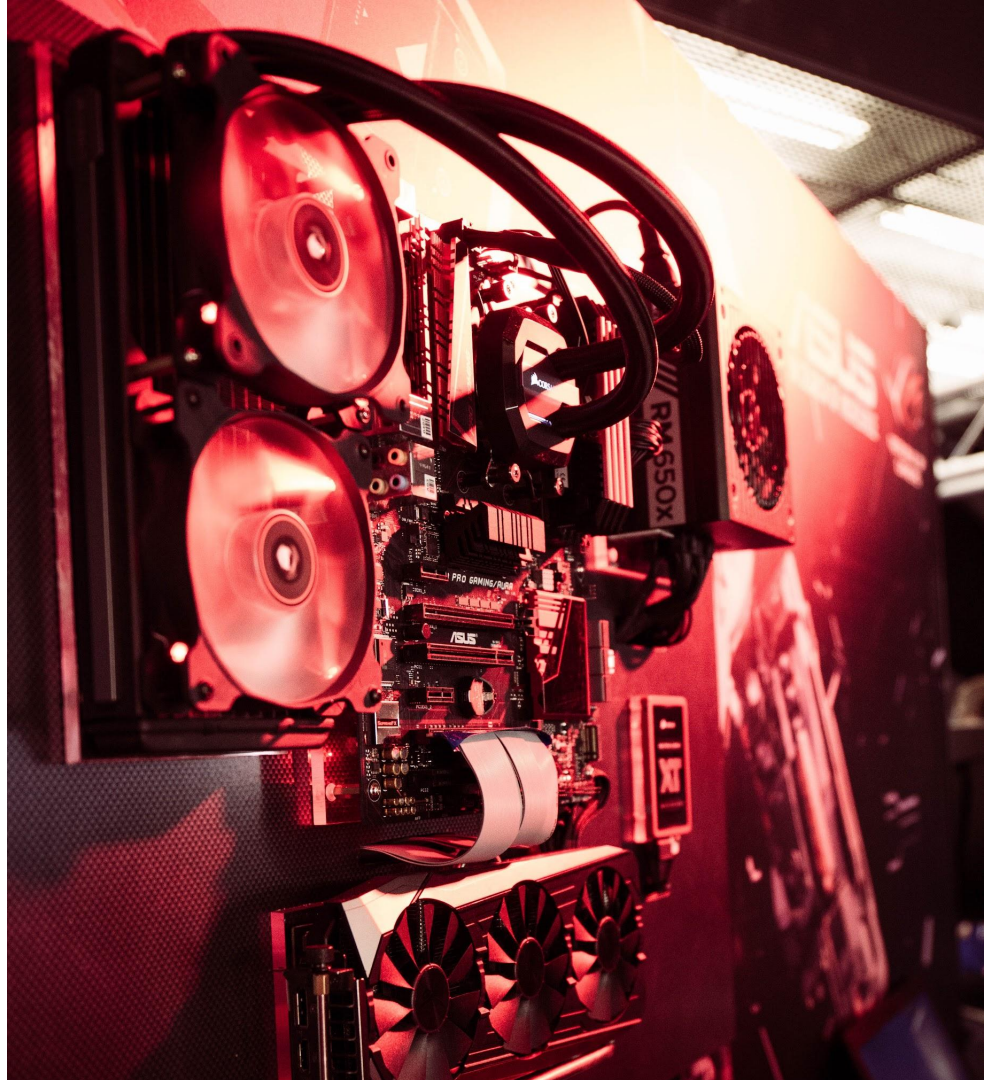
```
for trace_id, spans in traces.items():
```

```
    logger.info("Trace id: %s , span count: %s", trace_id, len(spans))
```

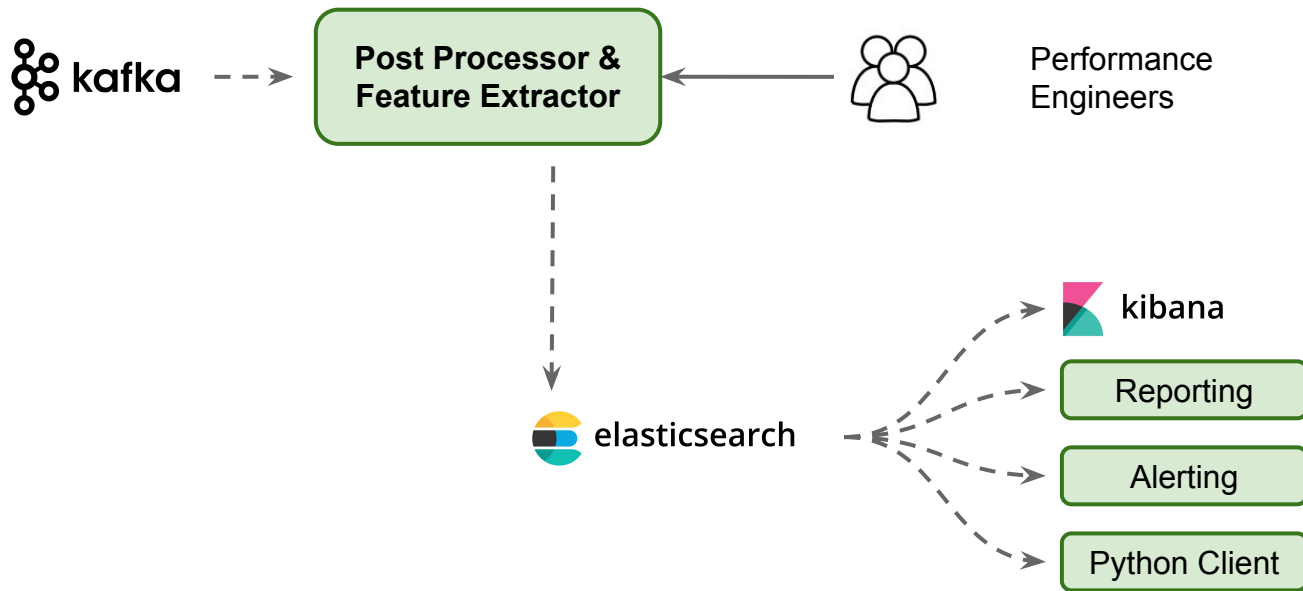


5. Post Processor

- Online analysis at trace level
- Feature extraction
- Flexible storage schema
- Automation (canary analysis, alerting)



Architecture

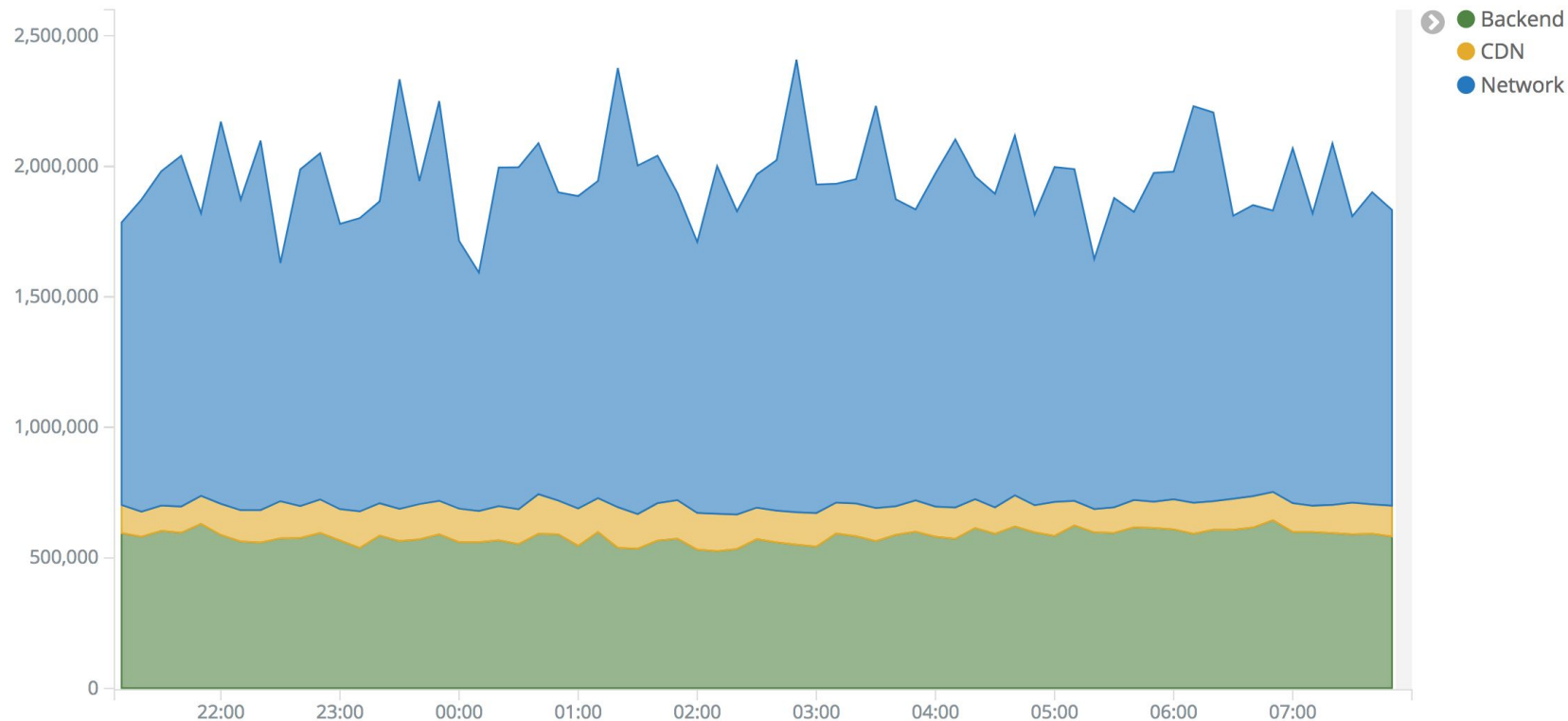


Processed Data

```
{
  "traceSummary": {
    "spanCount": 61,
    "rootService": "android",
    "rootEndpoint": "pwt/pin_save_request",
    "timeStampMillis": 1538094458613,
    "traceId": "3ac5149533006108",
    "networkCalls": 52,
    "completeTrace": true
  },
  ...
  "android": {
    "country": "CA",
    "pwtDuration": 34
  },
  ...
},
....
```

```
    "cdn": {
      "v3ClientToCdnLatency": 145000,
      "v3CdnToAwsLatency": 20551,
      "v3Endpoint": "/v3/pins/37872464/repin/",
      ...
    },
    "ngapi": {
      "v3Endpoints": ["v3_repin_handler"],
      "v3EndpointCount": 1,
      "clientType": "Android",
      "ngapiCluster": "prod",
      "v3Latency": 265449
    },
    ...
  }
}
```

Latency Breakdown



Summary

Zipkin UI	Viewing single trace
Kibana	View trends and metrics
Trace Analyser	Offline analysis
Python Client	Ad hoc analysis
Post Processor	Online analysis and feature extraction



Questions

