Office 365 + Spark Powering Delve Analytics

- Introductions
- Delve Analytics
- Architecture
- Spark Environment
- Usage Patterns
- Best Practices
- Takeaways

Introductions

Paavany Jayanty

Paavany is a Senior Program Manager in the Office 365 Customer Fabric team at Microsoft, whose mission is to attract, retain, and engage users with the help of their big data products.

Yi Wang

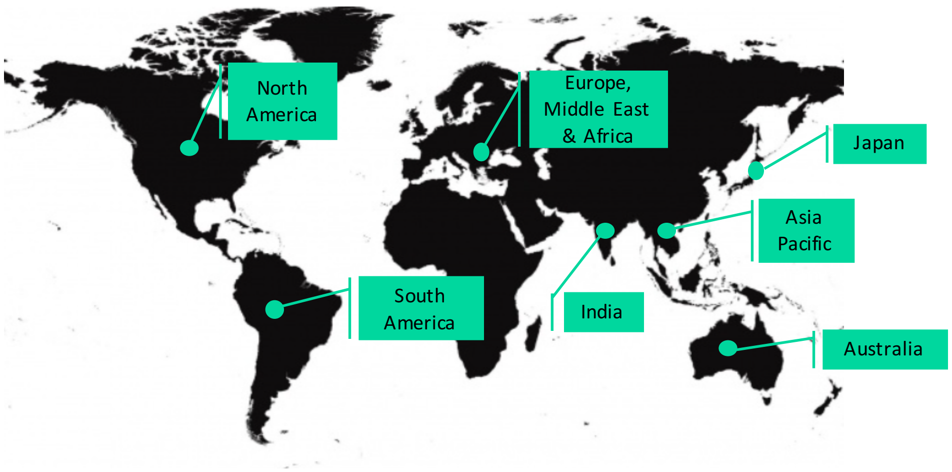
Yi is a Senior Software Engineer in the Office 365 Customer Fabric team at Microsoft working on building a platform to enable big data applications.

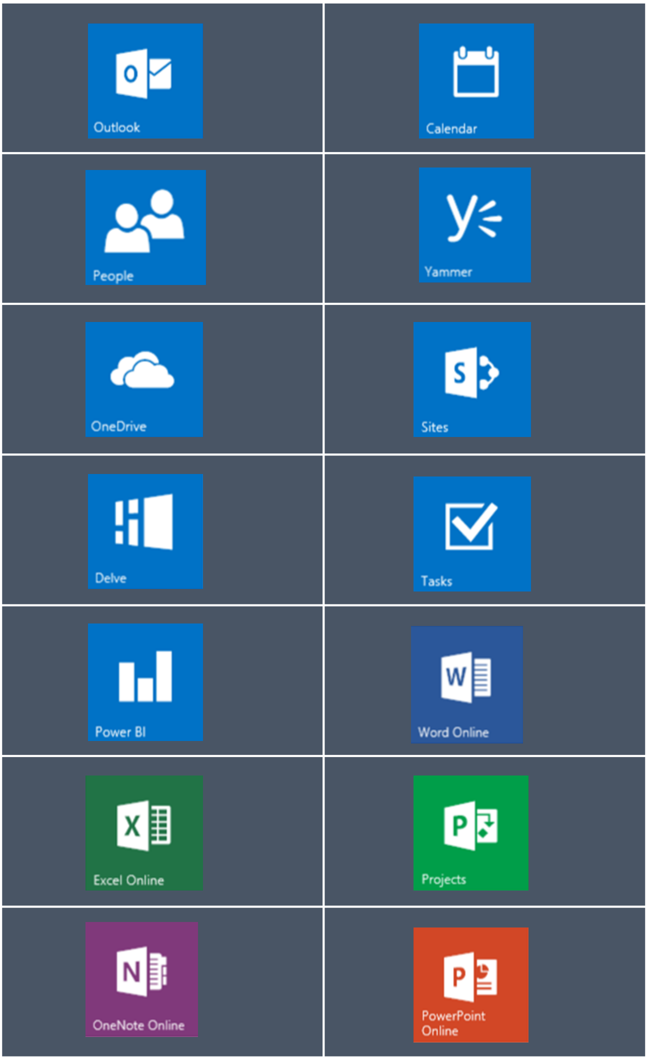
Office 365 - Fun Facts

- 1.6 billion – Sessions per month of phones and tablets
- 59% - Commercial seat growth in FY16 Q2
- 20.6 million - Number of Consumer Subscribers now

- >30 Million – Number of iOS and Android active devices running Outlook
- 80K – Number of partners selling O365
- Continued install base growth across Office, Exchange, SharePoint and Skype for Business

Users across the world served by O365 data centers





Delve Analytics

Reinventing productivity through individual empowerment.

Delve Analytics provides you with insights into two of the most important factors in personal productivity:

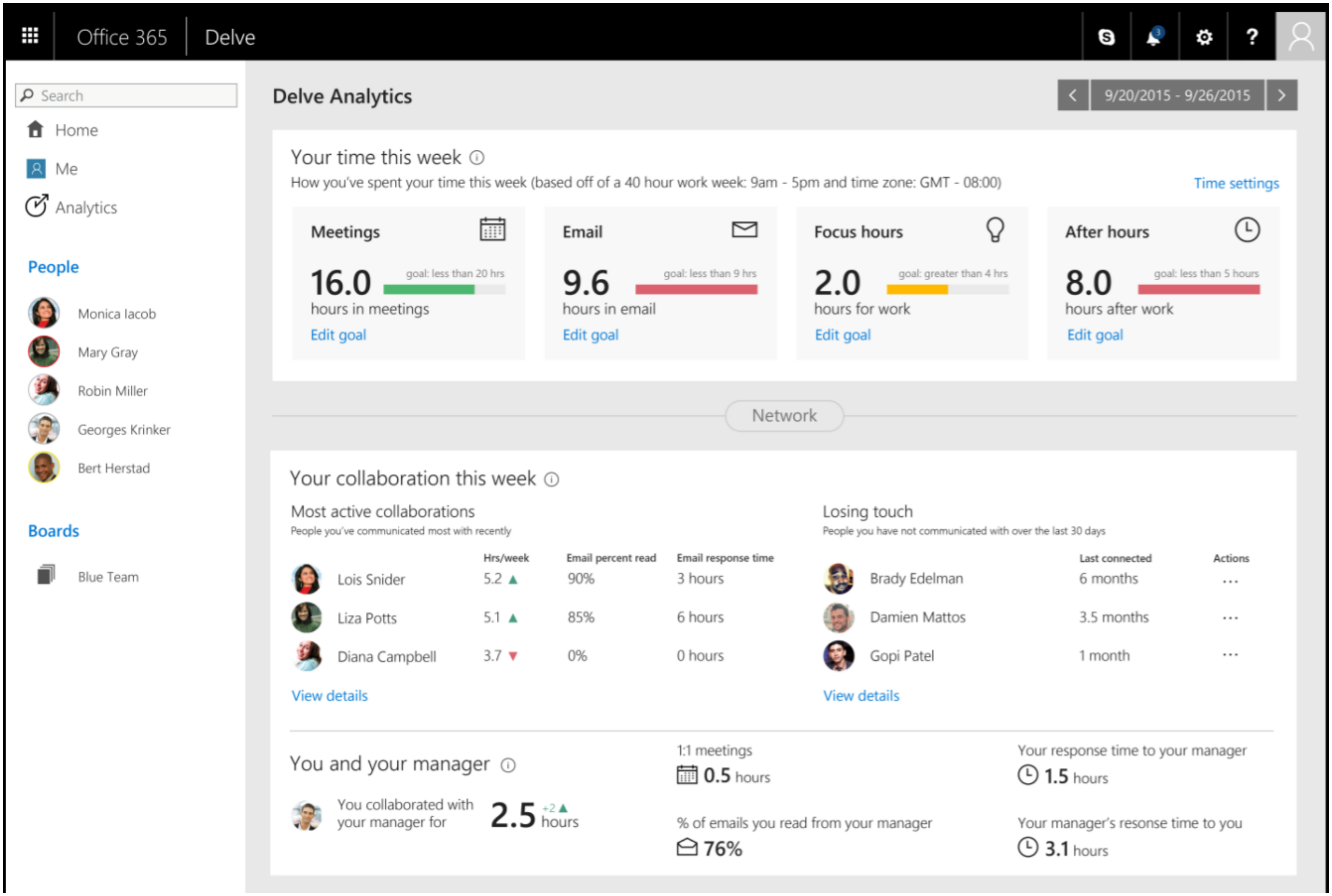
- How you spend your time
- Who you spend your time with

Delve Analytics helps you take back your time and achieve more.

Offered in the E5 SKU, and as an add-on to E1 or E3 subscriptions.

Delve Analytics inherits all Office 365 Security, Privacy and Compliance standards and commitments. Your insights are only available to you, otherwise service metadata is aggregated and anonymized and not personally identifiable.

- How many hours do I spend in Meetings?
- How many hours do I spend working after work?
- How many hours do I spend on email?
- How many hours to I spend on email compared to the rest of the organization?
- What are my most active collaborations?



People

- Monica Iacob
- Mary Gray
- Robin Miller
- Georges Krinker
- Bert Herstad

Boards

- Blue Team

Delve Analytics

9/20/2015 - 9/26/2015

Your time this week

How you've spent your time this week (based off of a 40 hour work week: 9am - 5pm and time zone: GMT - 08:00)

Time settings

Meetings
16.0 hours in meetings
goal: less than 20 hrs
Edit goal

Email
9.6 hours in email
goal: less than 9 hrs
Edit goal

Focus hours
2.0 hours for work
goal: greater than 4 hrs
Edit goal

After hours
8.0 hours after work
goal: less than 5 hours
Edit goal

Network

Your collaboration this week

Most active collaborations

People you've communicated most with recently

	Hrs/week	Email percent read	Email response time
Lois Snider	5.2 ▲	90%	3 hours
Liza Potts	5.1 ▲	85%	6 hours
Diana Campbell	3.7 ▼	0%	0 hours

View details

Losing touch

People you have not communicated with over the last 30 days

	Last connected	Actions
Brady Edelman	6 months	···
Damien Mattos	3.5 months	···
Gopi Patel	1 month	···

View details

You and your manager

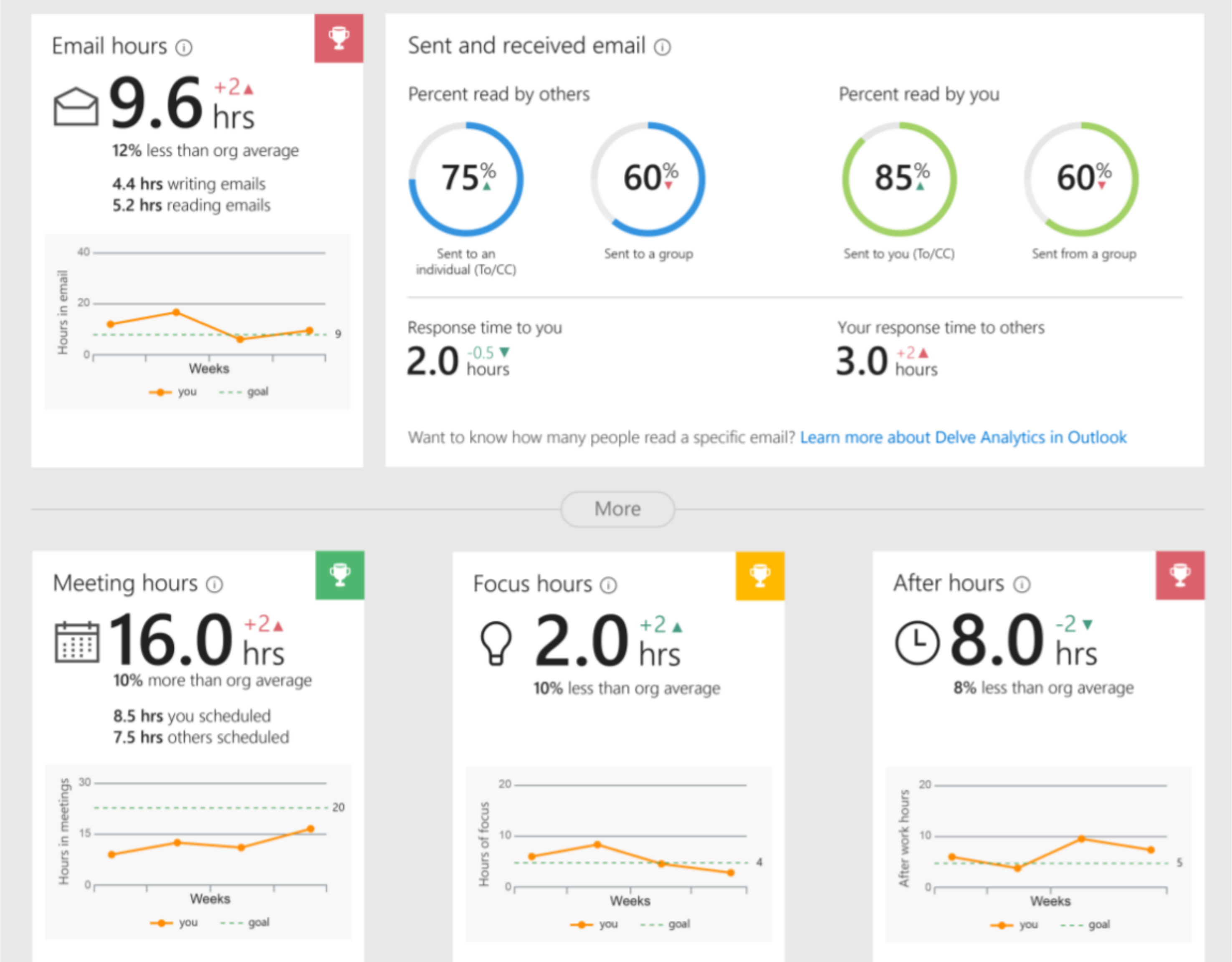
You collaborated with your manager for 2.5 hours +2 ▲

1:1 meetings
0.5 hours

% of emails you read from your manager
76%

Your response time to your manager
1.5 hours

Your manager's resonse time to you
3.1 hours



Email hours ⓘ

9.6 +2▲ hrs

12% less than org average

4.4 hrs writing emails
5.2 hrs reading emails

Sent and received email ⓘ

Percent read by others

75% ▲ Sent to an individual (To/CC)

60% ▼ Sent to a group

Percent read by you

85% ▲ Sent to you (To/CC)

60% ▼ Sent from a group

Response time to you

2.0 -0.5 ▼ hours

Your response time to others

3.0 +2 ▲ hours

Want to know how many people read a specific email? Learn more about Delve Analytics in Outlook

Meeting hours ⓘ

16.0 +2▲ hrs

10% more than org average

8.5 hrs you scheduled
7.5 hrs others scheduled

Focus hours ⓘ

2.0 +2 ▲ hrs

10% less than org average

After hours ⓘ

8.0 -2 ▼ hrs

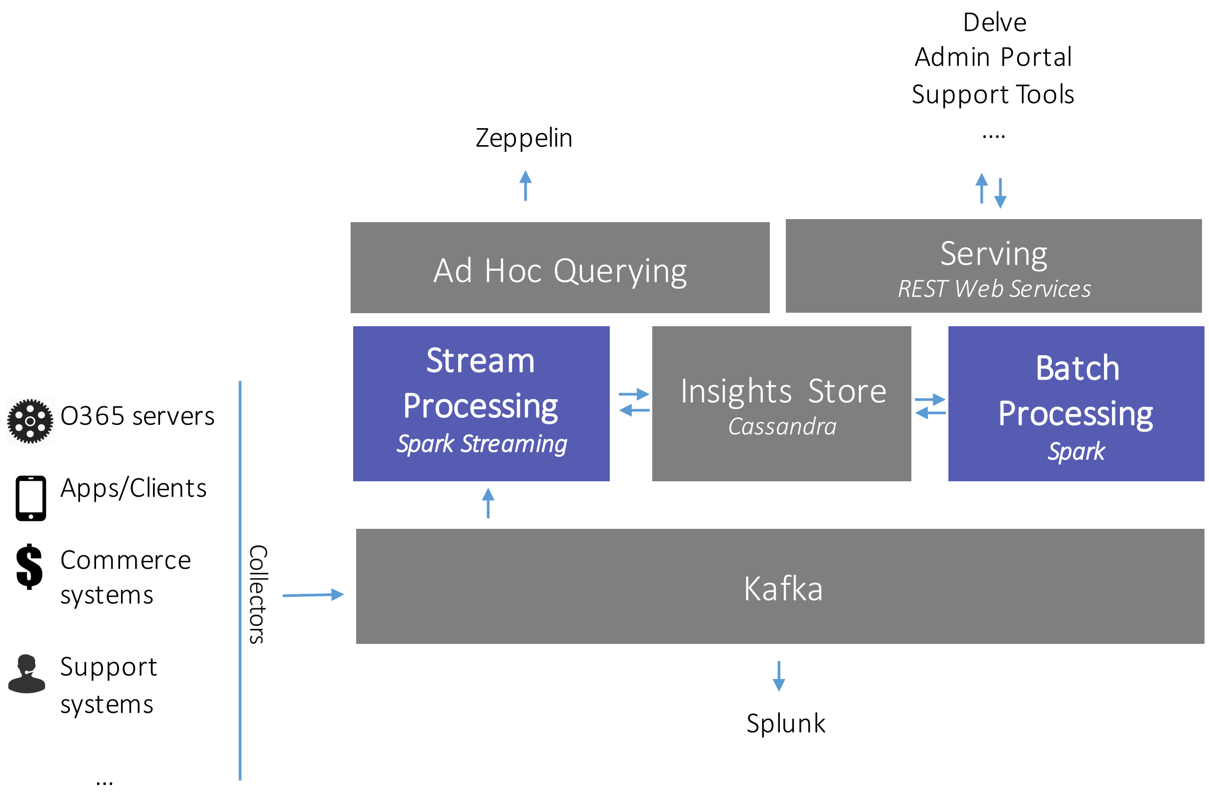
8% less than org average

Sparking Delight!

Keeping millions of users ***happy*** is the way we, at O365, help attract, retain and engage our users.

We do this with a common analytics platform ***powered by Spark*** and shared insight repository that enables anyone to quickly and easily use multi-signal analysis of near real-time O365 data to gain rich insights, deeply understand our customers, and build and deliver customized experiences that ***truly delight!***

Architecture



Zeppelin

Delve
Admin Portal
Support Tools
....

Ad Hoc Querying

Serving
REST Web Services

Stream Processing
Spark Streaming

Insights Store
Cassandra

Batch Processing
Spark

Kafka

Splunk

O365 servers

Apps/Clients

Commerce systems

Support systems

...

Collectors

Key facts

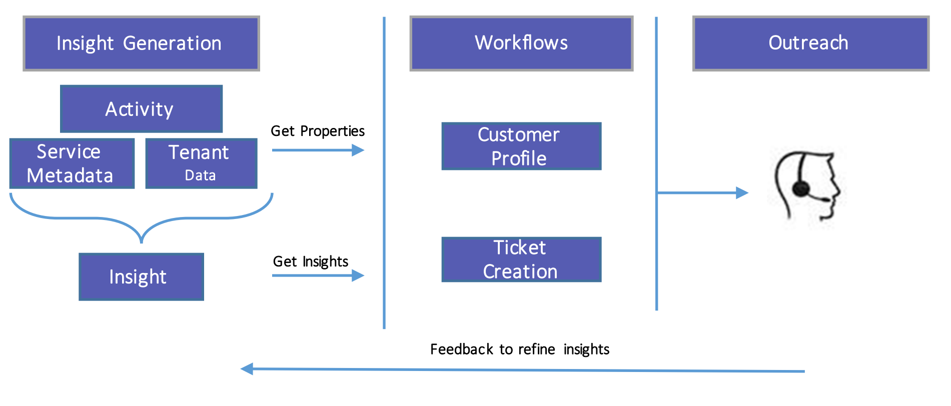
- ● Running in Azure
- ● Highly Scalable
- ● High ingestion rates
- ● Real time analytics
- ● Batch analytics
- ● Machine learning
- ● PII Compliance

Spark Batch Use Case

Support Scenario - Prevent customers who are actively using our service from getting disabled due to expired subscriptions (Dunning).

We decided to win on *satisfaction* with these customers by proactive outreach and helping customers renew the service on time.

Using spark batch analytics we flagged customers who were about to be dunned and automatically created support tickets for our support agents to act on. We also generated customer profiles so that our agents are empowered with targeted information.

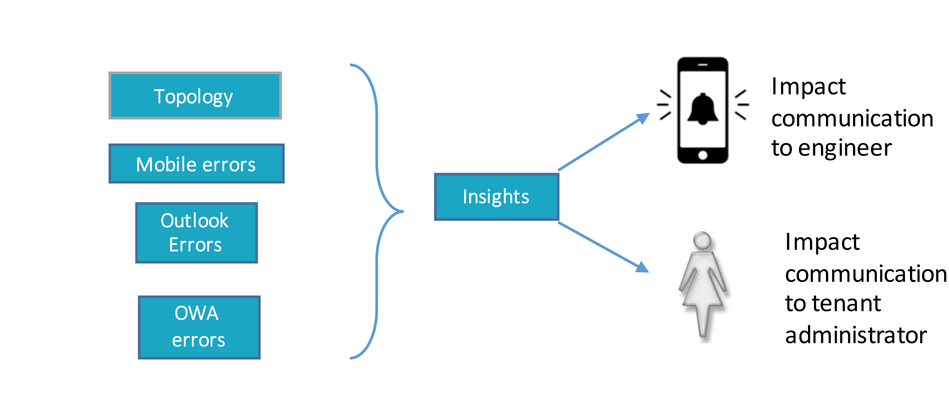


Spark Streaming Use Case

Service Scenario – Detect impact of a service incident in real time and narrowcast status to customers.

The reality of the service world is that it is subject to incidents which impact the user experience. The key is to handle them proactively and in a timely manner: alert before the service availability dips below a threshold, investigate the issue in real time and narrowcast communications to the specific set of impacted users.

Using spark streaming we correlate the error signals with our topology to determine those who were impacted and proactively communicate with them.


Topology
Mobile errors
Outlook Errors
OWA errors
Insights
Impact communication to engineer
Impact communication to tenant administrator

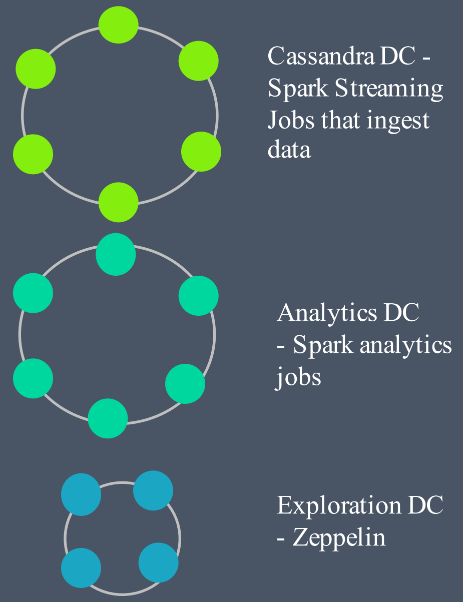
Environment

Production Environment:

- Running on Azure
- Version: DSE 4.8, Spark 1.4.1
- Maintain multiple clusters powering different scenarios
- 3 different DCs in a cluster
  - Cassandra, Replication Factor = 5
  - Analytics, Replication Factor = 5
  - Exploration, Replication Factor = 3
- All DCs for a single cluster are in one Vnet.
- No inbound access is allowed from outside the Vnet.

Machines Used:

- D14: 16 cores;112 gb memory; 3TB attached local SSD
- G4: 16 cores;224 gb memory; 3TB attached local SSD



Cassandra DC - Spark Streaming Jobs that ingest data

Analytics DC
- Spark analytics jobs

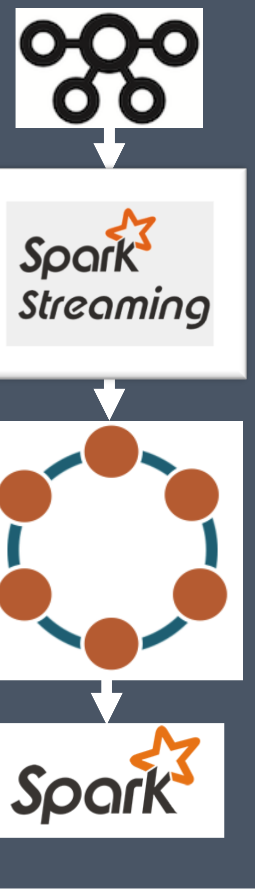
Exploration DC
- Zeppelin

Microsoft Azure

Spark Usage Patterns

We have three Spark usage patterns:

- Near Real-Time Processing
- Batch Processing
- Ad-hoc Querying



Usage Pattern #1: Near Real-Time Processing

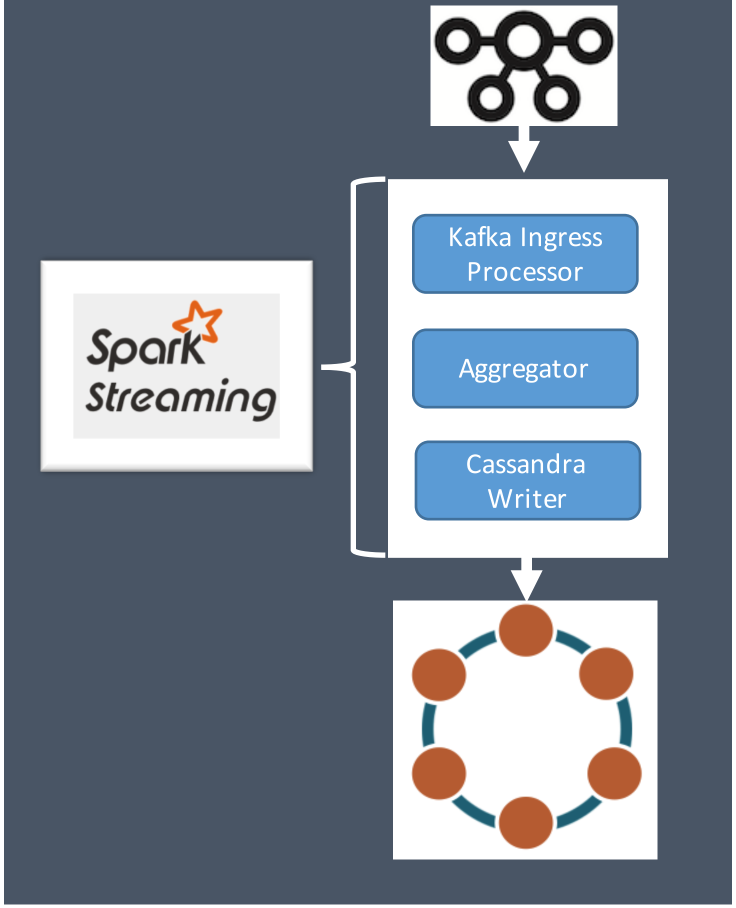
Spark Streaming jobs pipe data from one stage to another in real time.

When do we use this?

- Scenario needs to be completed near real-time
- Event disorders, late events or event drops are accepted
- Don't have a big look back window

Pros: Less data stored in Cassandra; Near Real-time;

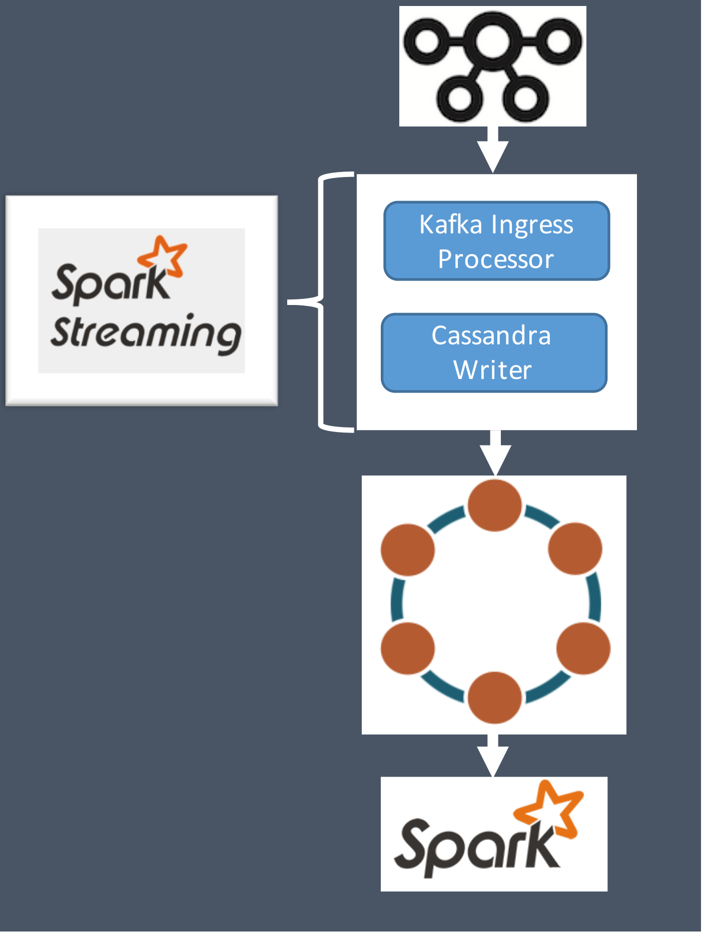
Cons: If system is unhealthy, since the buffering window is small, there is no easy way to recover the data.



Usage Pattern #2: Batch Processing

Spark Streaming jobs move the raw data from Kafka, do simple data conversion and output processed raw data to Cassandra.

Spark Batch Jobs do further aggregations and analysis.



When do we use this?

- Event accuracy and order is very important to the stream
- Need to look back a few days / weeks / months of data for trends
- Provide a common datasets for other jobs to leverage
- Complicated joins with multiple datasets to produce rich insights

Pros: High data accuracy; Can easily recover from issues;

Complicated analytics like TopN become feasible; Allows other jobs to

reuse the common curated datasets

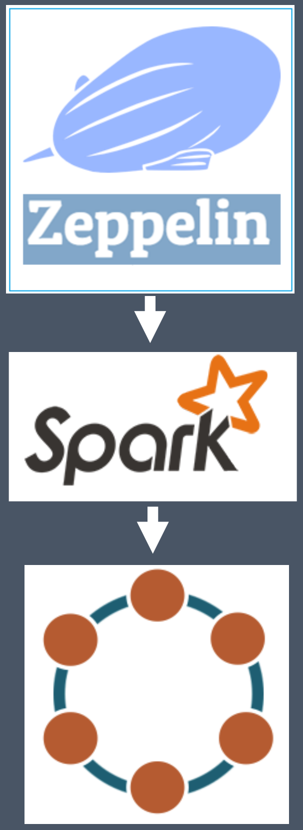
Usage Pattern #3: Ad-hoc Querying

Query data through Zeppelin which supports spark interpreters.

When do we use this?

- Explore valuable existing insights for planning
- Validate data to ensure accuracy
- Ad-hoc data access. Dream up a query and run it!

Pros: Flexible; Democratizes access to rich insights;



Best Practices: Streaming Jobs

Streaming Jobs

- Connection timeout causes streaming job slowness. Solved by increasing Keep_alive_ms
- Cache intermediate result that used frequently to improve performance.
- Direct approach is more efficient than the Receive-Based approach.
- Avoid usage of inserts and updates in streaming jobs.

Best Practices: Batch Jobs

Batch Jobs

- Generate common datasets that can be used by other jobs.
- Tune spark.Cassandra.input.split.size to adjust # of partition size for better job performance.

Key Takeaways

- More nodes with relatively smaller capacity is more performant than few more powerful nodes.
- Streaming Jobs, Batch Jobs and Adhoc Query should lived in separate DCs.
- Use direct approach for spark streaming jobs.
- Improve job performance by increase keep_alive_ms to avoid expensive reconnects to Cassandra.
- Investing in data modelling early on is very important, it will be expensive to change later. Your api and spark access patterns should drive schema design.

Delve Analytics-Video

https://www.youtube.com/watch?v=u1Toq7Y0NPo

Contact US

ywa@microsoft.com

pajayant@microsoft.com

Appendix

Security and Privacy

PRIVACY

The Delve Analytics dashboard surfaces information to you about you and already discoverable to you.

CONTROLS

Tenant and user level opt-in / opt-out settings

COMPLIANCE

All data remains Office 365 compliant.

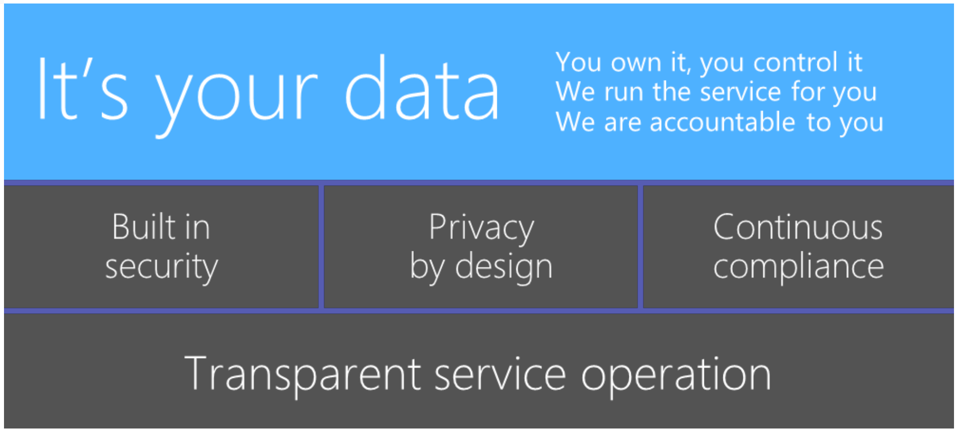
All customer data remains subject to established geographic data boundaries.

Delve Analytics inherits all Office 365 Security, Privacy and Compliance standards and commitments.

WHAT DOES THIS MEAN?

All insights surfaced via Delve Analytics dashboard are already available to you in your inbox and calendar, such as response times, who you meet with and how often.

If not already available to you, then this data is aggregated and anonymized and not personally identifiable



It's your data

You own it, you control it
We run the service for you
We are accountable to you

Built in security

Privacy by design

Continuous compliance

Transparent service operation

Monitoring and Recovery

We have the following in place:

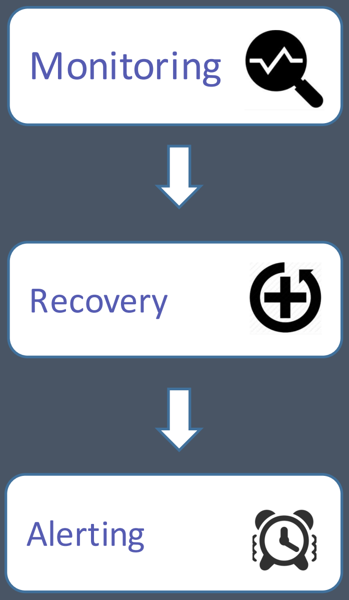
- Kafka Monitors: Monitors the number of Kafka brokers and Zookeepers that are alive.
- Spark Streaming Monitor: Monitors the # of batches pull from Kafka, # of records saved to Cassandra etc.
- Cassandra Monitor: Monitors if Cassandra nodes are healthy using ops center API.
- Spark Batch Job Monitor: Monitors if jobs ran successfully or not.

Recovery:

- If the node is down automatically bring it up.

Learnings:

- If a node is in a bad state, bringing it up might cause more issues.



Monitoring

Recovery

Alerting

Challenges

- Spark History Server
- Detect Failure and auto-recovery
- Scheduling systems: Cron, Azkaban and Oozie
- Debug Diagnostic job failure