The ELK Stack

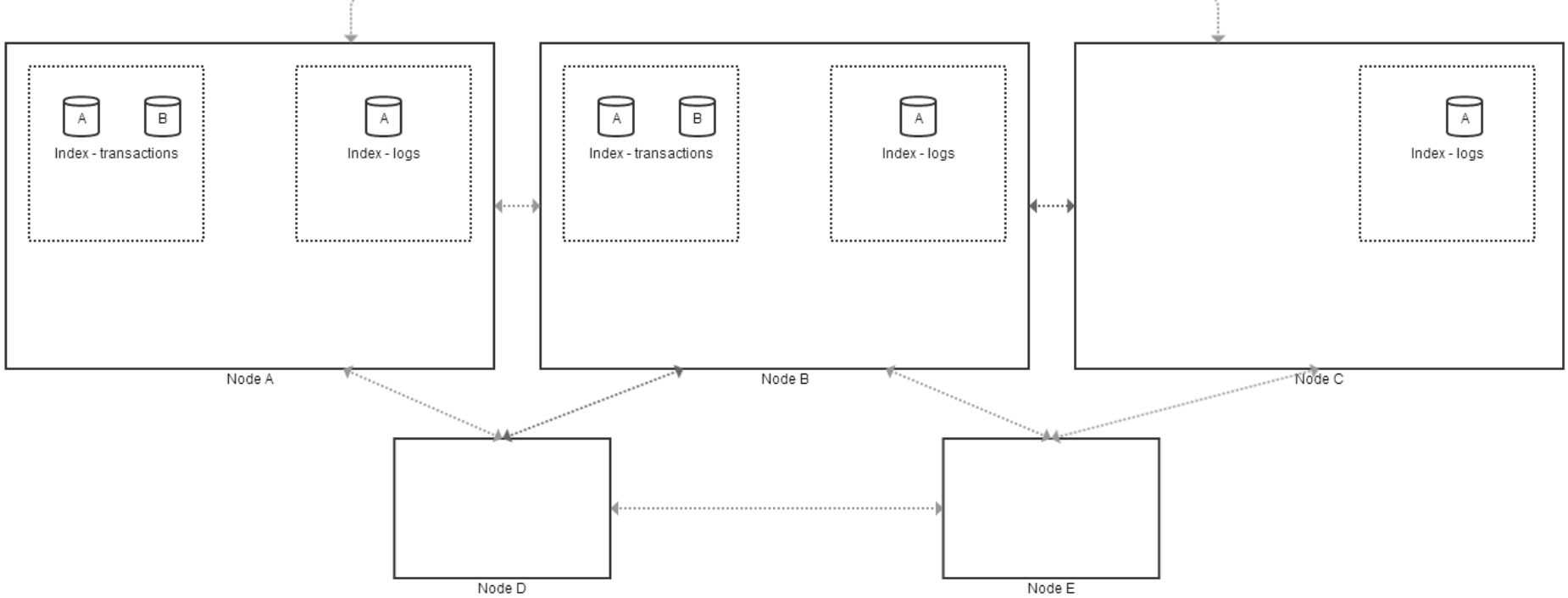
ElasticSearch, LogStash, and Kibana

What is 

- Schema-less
- Distributed
- REST-ful, Document-oriented, and speaks JSON
- For searching and analytics
- and more...

Architecture

- Built on top of Apache Lucene
- Runs on the JVM
- Distributed in nature - cluster can have data, master or load balancing nodes
- Highly available and fault tolerant



Legend

Shard

Index

Node (instance of ElasticSearch process)

ElasticSearch - document oriented

- and schema-less

Movies example…

- index a document (PUT and POST)
- check for existence of a document
- retrieve fields
- delete
- update
- update with optimistic concurrency
- update partial
- upsert

ElasticSearch - Distributed

- Start a node - open Marvel
  - Data is allocated within the node
- Start another node
  - Highlight data being redistributed to the new node
- Discovery mechanisms - multicast vs. unicast
- Master election, sharding

ElasticSearch - RESTful

- Get index stats - number of shards (partitions of data), replicas, state and size
- Get cluster health - overall health status, number of shards and nodes
- Get cluster state - metrics of all indices, settings and mappings of all indices, some metrics, info on all shards in all indices

ElasticSearch - Concepts

- Index - highest level bucket to store documents, indicates some physical storage
- Type

```
Relational DB  ⇒ Databases ⇒ Tables ⇒ Rows      ⇒ Columns
Elasticsearch  ⇒ Indices   ⇒ Types  ⇒ Documents ⇒ Fields
```

- Mapping - the definition of a type (think schema) and how ElasticSearch should analyze, parse and store the fields of this type
- Analysis:
  - first, tokenizing a block of text into individual *terms* suitable for use in an inverted index,
  - then normalizing these terms into a standard form to improve their “searchability” or *recall*.

ElasticSearch - search and analytics

- Search
  - Structured search - working with exact values, between date ranges, numbers, enumerated strings, etc...
  - Full-text search - natural language and other text, relevance is usually concern here instead of exact matches

ElasticSearch - search in depth

- Analyzes all documents and keeps an inverted index data structure for fast matching

Inverted index example

Document: “The quick brown fox jumped over the lazy dog.”

Term	Doc 1
The	x
quick	x
brown	x
fox	x
jumped	x
over	x
the	x
lazy	x
dog	x

Inverted index example

Document: “The quick brown fox jumped over the lazy dog.”

Document: “Quick, the fox, was lazy.”

Term	Doc 1	Doc 2
The	x	
quick	x	
brown	x	
fox	x	x
jumped	x	
over	x	
the	x	x
lazy	x	x
dog	x	
Quick		x
was		x

ElasticSearch - Query DSL

- Simple search
- Compound search
- Query vs. Filters
- Range filter
- Aggregations
- Significant terms (‘the uncommonly common’)

ElasticSearch - search examples

- NFL data - fuzzy description, more like this
- NFL data - bool query
- NFL data - all IND offense
- NFL data - aggregations - average down and distance, 2nd half yard to go

ElasticSearch - search examples

- NFL 2013 data - get touchdowns by quarter
- NFL 2013 data - get significant terms in description by teams

ElasticSearch - Demo Charting App

NFL Viz: https://github.com/mradamlacey/nfl-viz

What is LogStash

- Data import/export tool for time series and log data
- Design inspired by Unix utilities which pipe in/out to each other

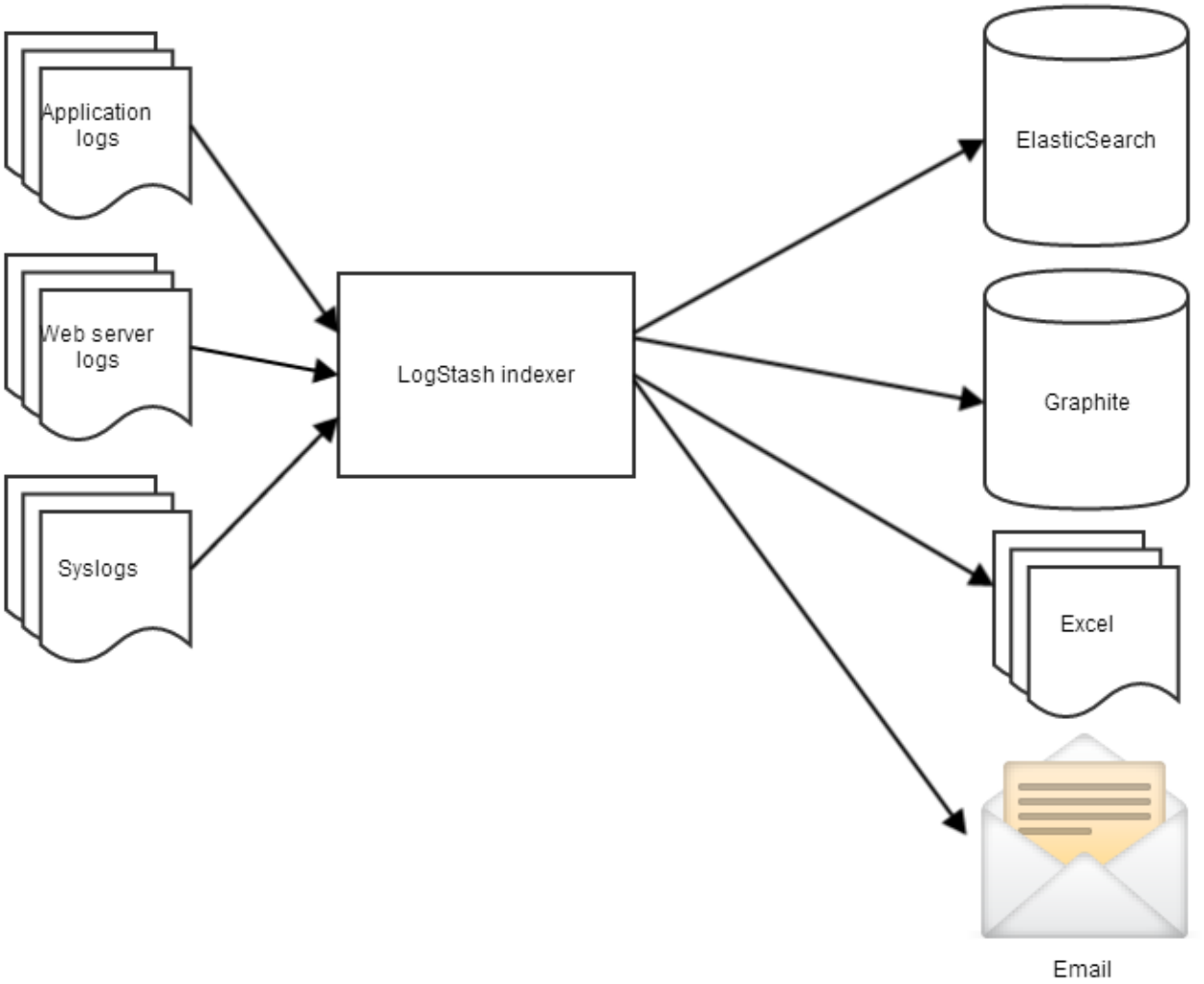
What problem does it solve?

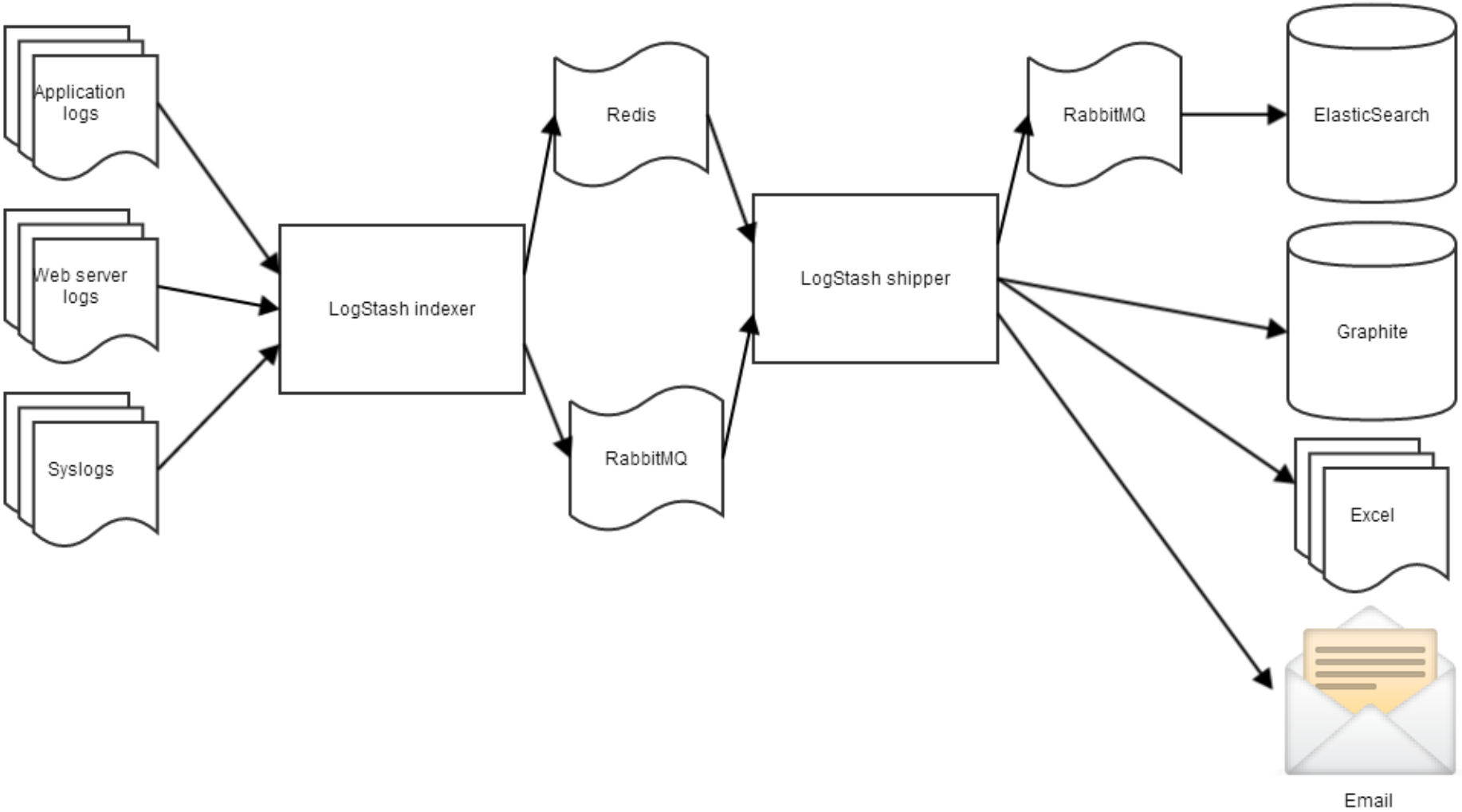
- How to parse and analyze log data from many sources?

```
Mar 12 12:00:08 server2 rcd[308]: Loaded 12 packages in 'ximian-red-carpet|351' (0.01878 seconds)

[2014-05-06 08:04:00.333] [ERROR] - core - bad thing happened

[Wed Oct 11 14:32:52 2000] [error] [client 127.0.0.1] client denied by server configuration: /export/home/live/ap/htdocs/test
```

LogStash - example use cases

- Import of JSON to ES by dropping files into a folder
- Parse webserver access files across multiple servers, calculate response times and chart
- Parse application logs and send emails when an error occurs
- Stream application log data across many servers to a single log dashboard
- Drop a file into a folder to be ingested and aggregated into centralized log database


Application logs
Web server logs
Syslogs
LogStash indexer
ElasticSearch
Graphite
Excel
Email


Application logs
Web server logs
Syslogs
LogStash indexer
Redis
RabbitMQ
LogStash shipper
RabbitMQ
ElasticSearch
Graphite
Excel
Email

LogStash - example configuration

- input
- filter
- output

LogStash - demo

- output CPU load to CSV (load-avg.conf)
- Stream tweets to ElasticSearch (twitter.conf)
- Parse NodeJS server logs to ElasticSearch (mp.conf)

What is Kibana

- Dashboard tool for data in ElasticSearch
- Highly configurable/customizable, build panels with user defined charts, tables, etc...
- Built on AngularJS

Kibana - demos

- NFL stats dashboard
- Tweets dashboard
- ElasticSearch Marvel

https://github.com/mradamlacey/elk-stack-presentation

Sources

ElasticSeach - The Definitive Guide: http://www.elasticsearch.org/guide/en/elasticsearch/guide/current/

LMAO If you don't LogStash: http://tech.paulcz.net/ACUG-Logstash

Quick ELK Demos: https://github.com/kurtado/quick-elk