Big Data Analytics on Container-Orchestrated Systems

Gerard Casas Saez

University of Colorado Boulder

July 20th

Outline

Introduction

Background

Problem statement

Approach

Implementation

Questions?

Introduction

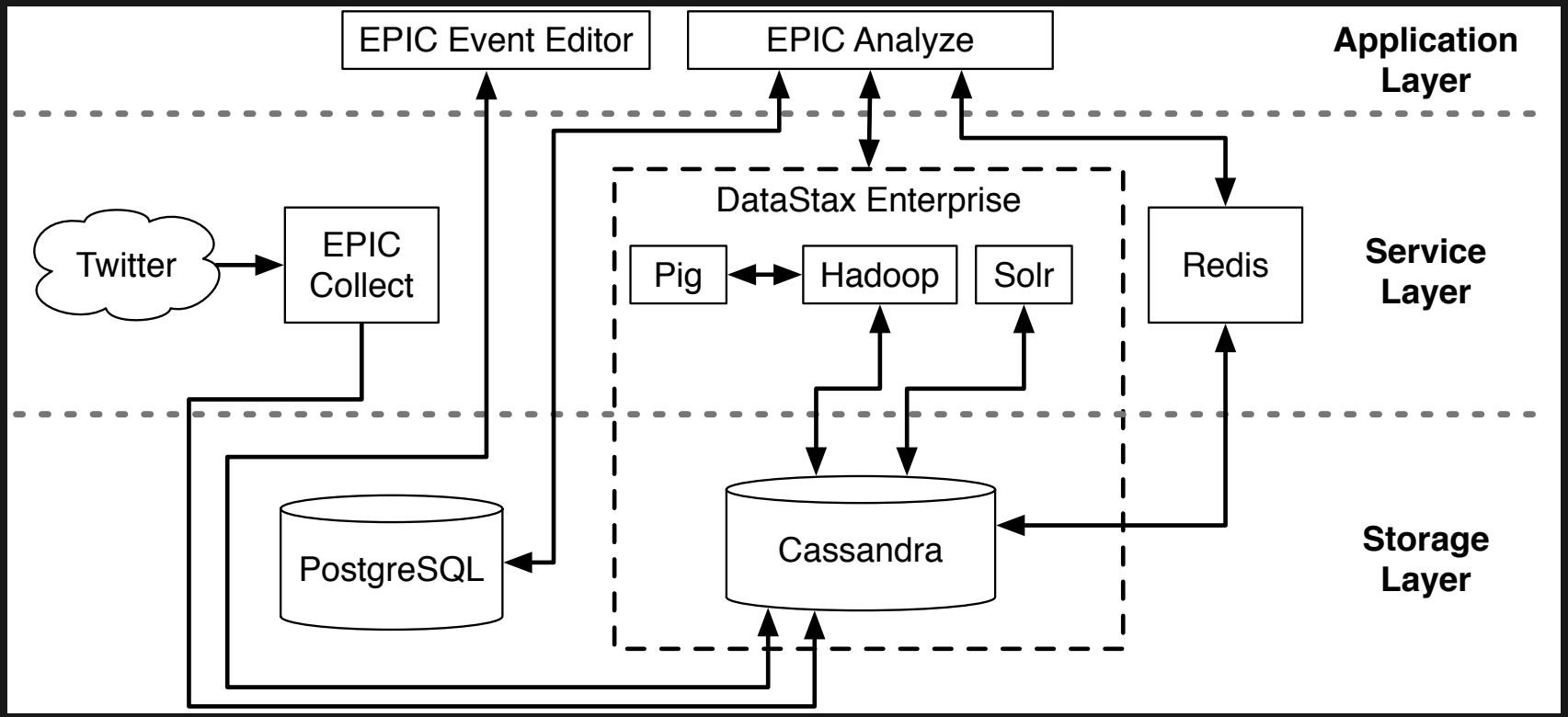
- Exponential increase in data generation
  - Current: 72 petabytes/month
  - Prediction 2021: 232 petabytes/month.
- Need to scale Big Data Analytics System
- Keeping mantainance at low cost
- Migrate Project EPIC architecture to new technologies

Background

Project EPIC

- EPIC Collect
- EPIC Analyze
- EPIC Analytics (additional machine)

Project EPIC



Microservices Architecture

- Small & specific
- Highly iteractive
- Loosely-coupled & highly-cohesive
- Independent development and scalability

Coreography vs Orchestration

Containerization

- Operating-system-level virtualization
- Use host machine system resources
- Development microservices
- **Docker** most used alternative

Container-orchestration systems

- Container interaction abstraction
- More mutable architectures
- Microservices deployment
- Apache Mesos vs **Kubernetes**
- **Google Cloud**: managed Kubernetes cluster

Problem statement

1. Advantages and/or limitations from the new Project EPIC infrastructure
   - 1.1 More reliable?
   - 1.2 More scalable?
2. Lower maintenance costs than the existing infrastructure?
   - 2.1 Easier to deploy?
   - 2.2 Easier to upgrade?
   - 2.3 More resilient to failures?

Approach

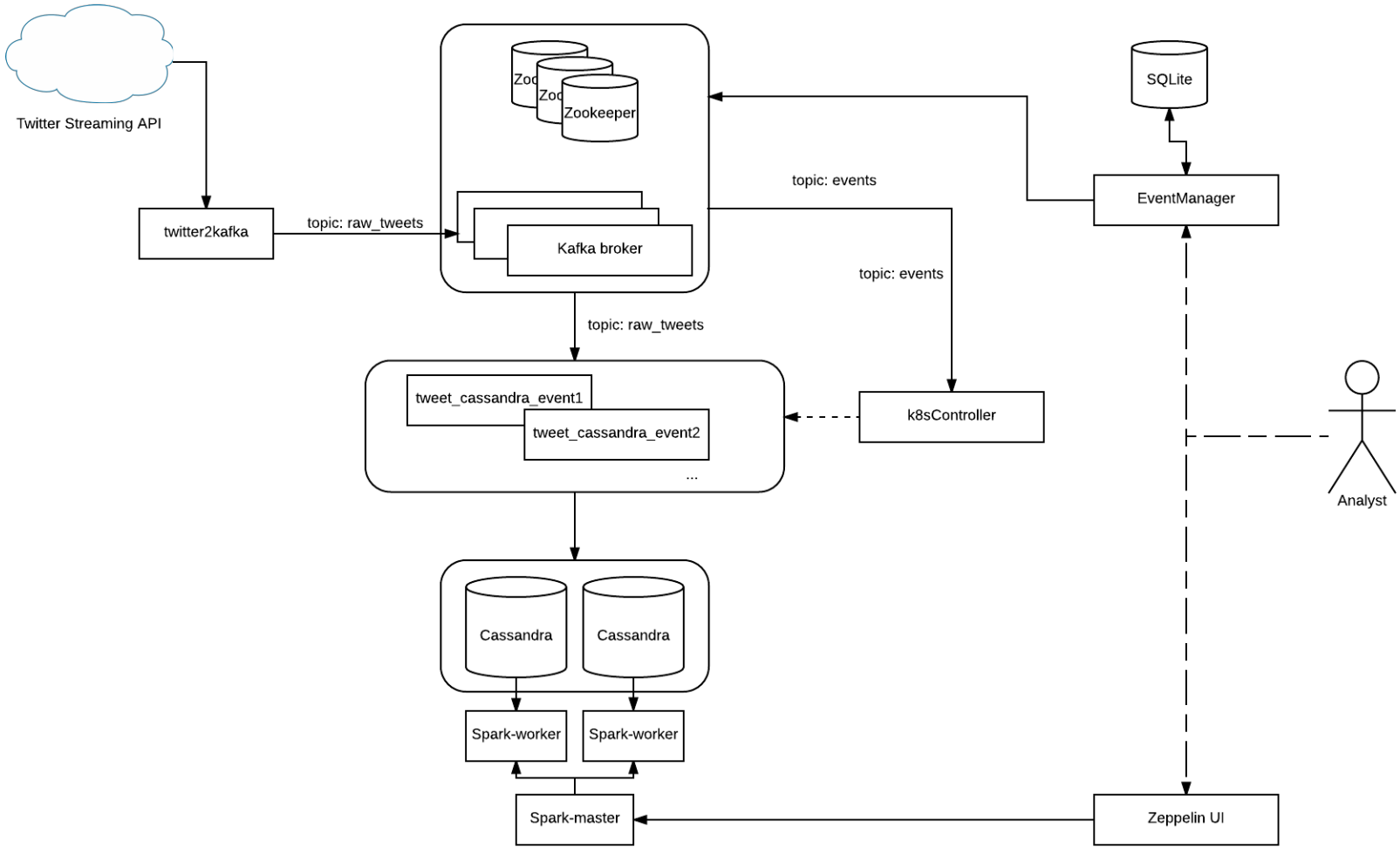
Features

- Event management
- Real-time collection of streaming Twitter data
- Real-time classification of incoming tweets
- Data Analysis

Custom components

- **Event Manager:** CRUD UI for events
- **Infrastructure Controller:** Changes infrastructure on demmand
- **Twitter Tracker:** Twitter streaming client
- **Twitter Normalizer:** JSON to cassandra row

Architecture



Implementation

Let’s track an event...

Event Manager UI

...and analyze it!

Zeppelin Notebook

Questions?