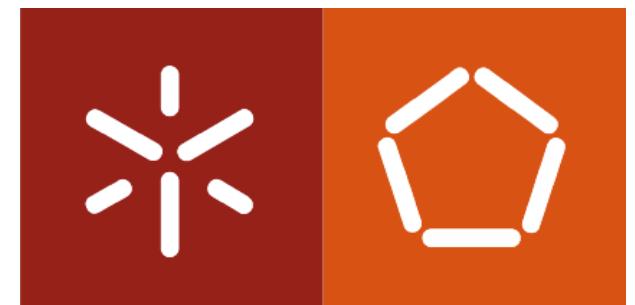


# Cloud Computing Applications and Services

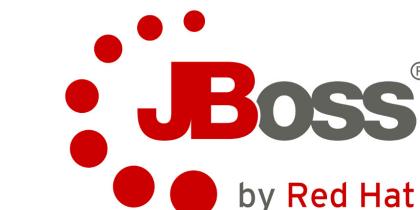
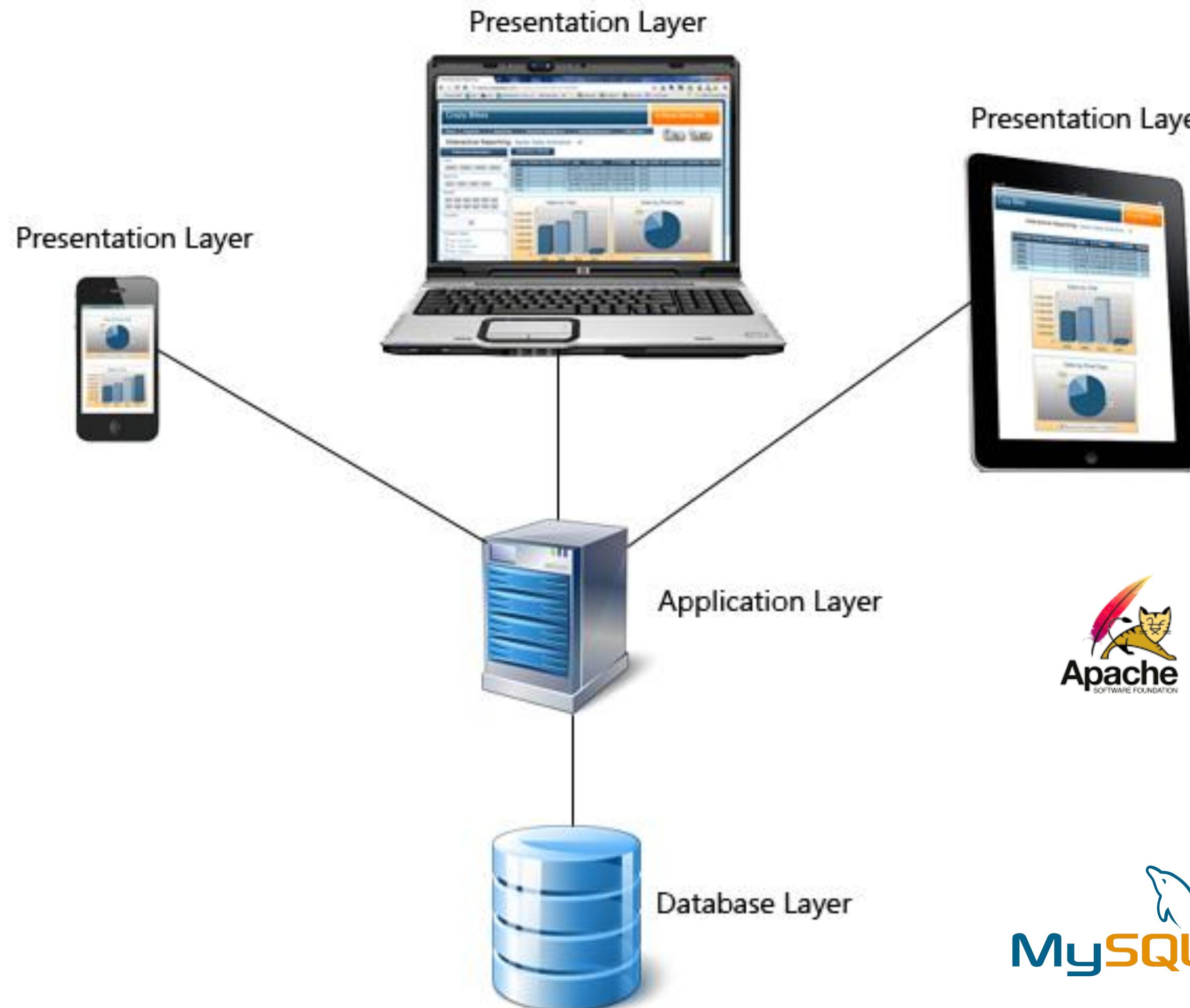
(Aplicações e Serviços de Computação em Nuvem)

## Introduction

University of Minho  
2024-2025

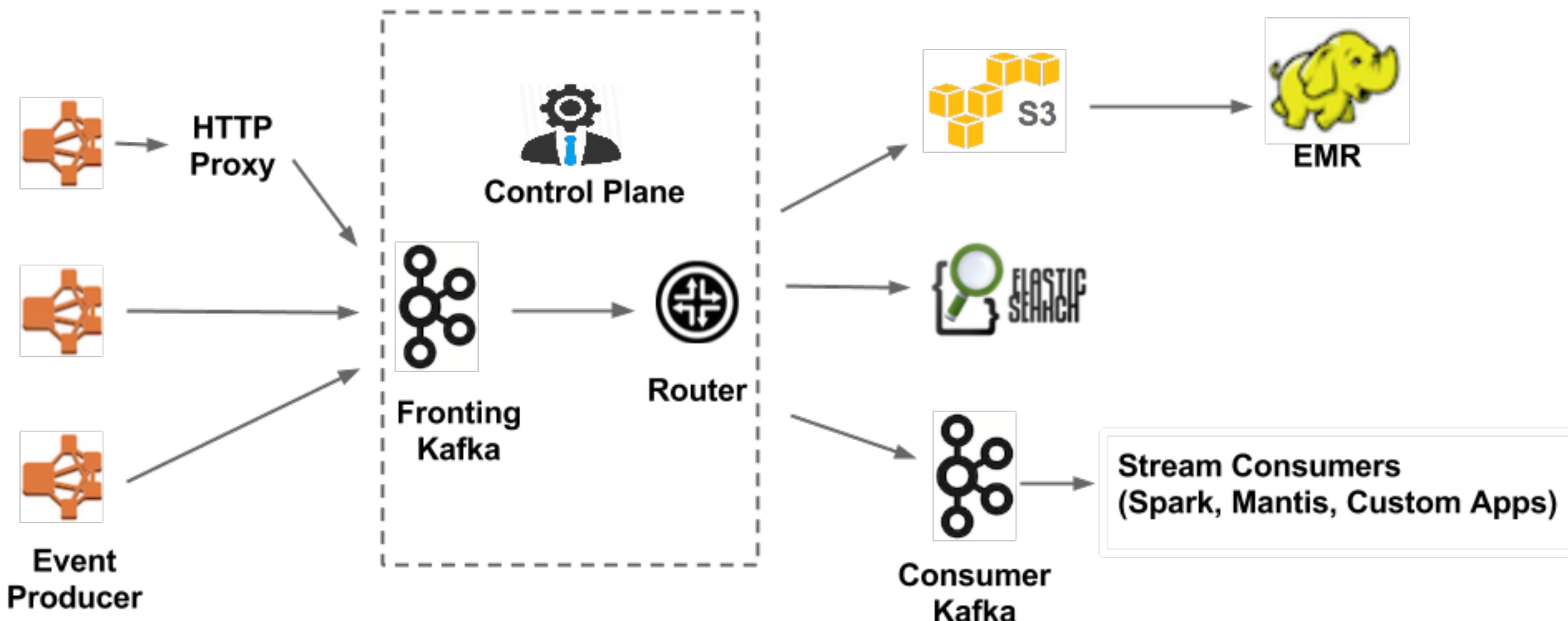


# A “Simple” Application



# Complex Applications

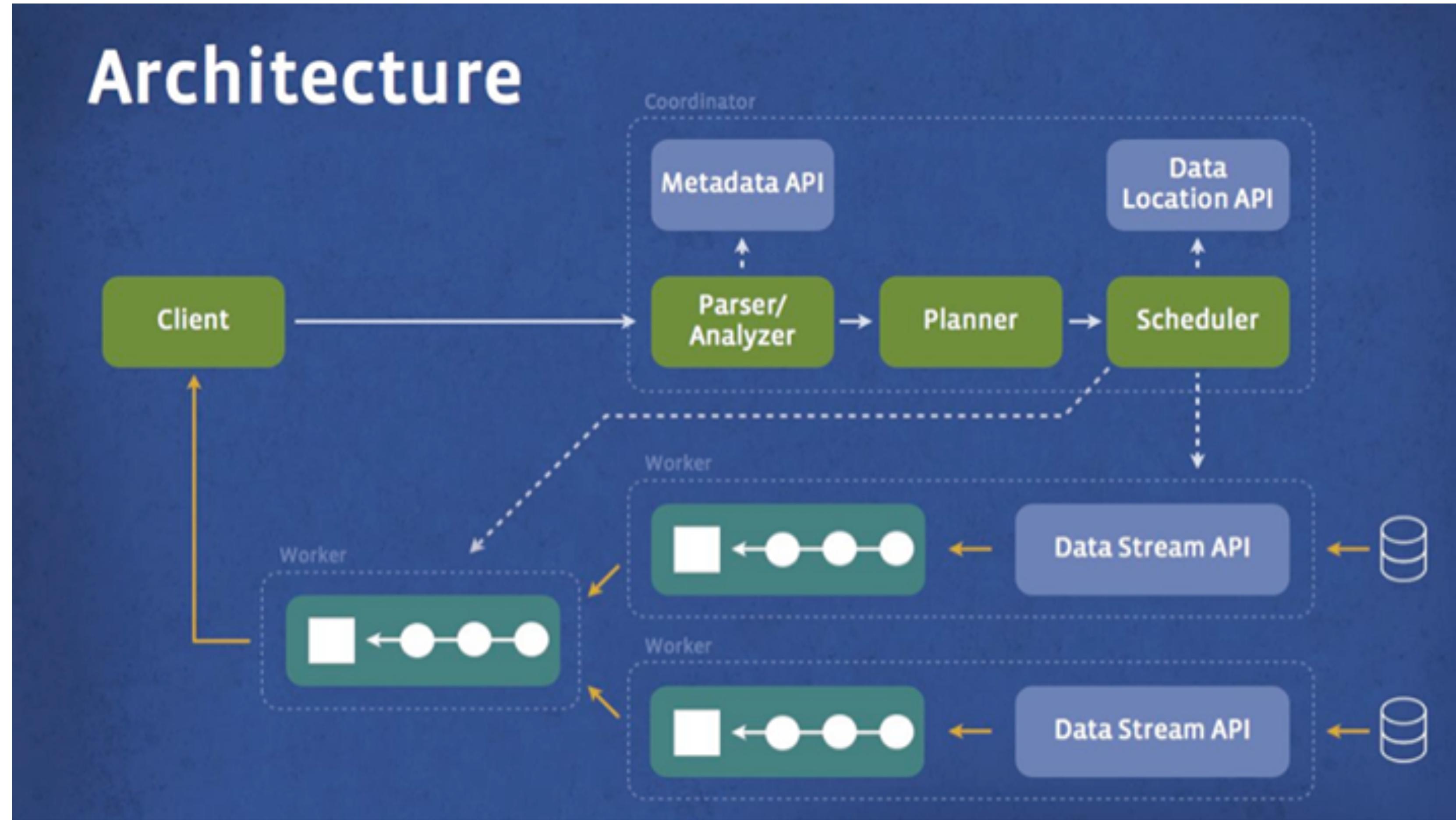
## Netflix – Keystone Data Pipeline



Source: <https://medium.com/netflix-techblog/evolution-of-the-netflix-data-pipeline-da246ca36905>

# Complex Applications

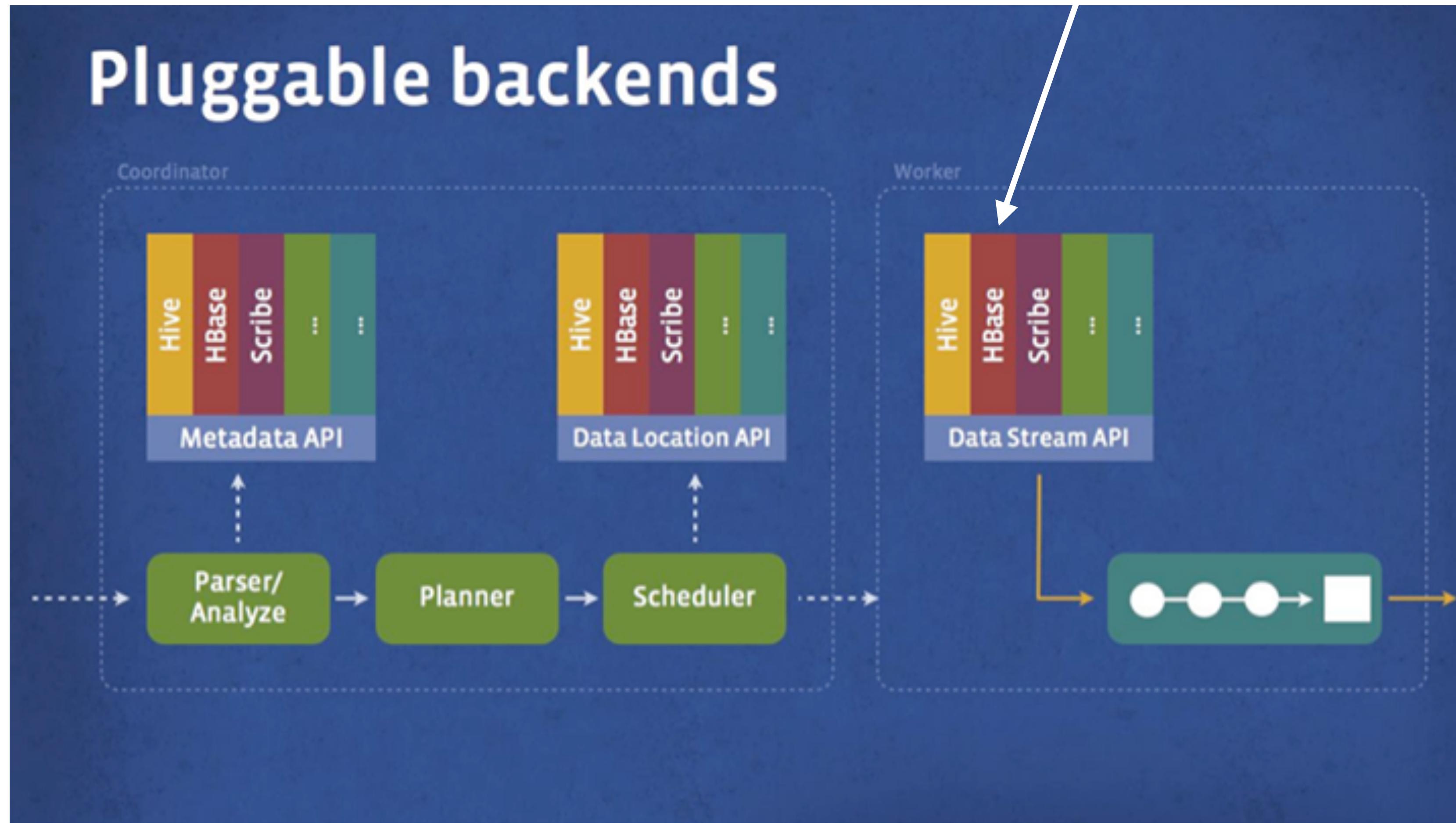
## Facebook – Presto SQL Engine



Source: <https://www.facebook.com/notes/facebook-engineering/presto-interacting-with-petabytes-of-data-at-facebook/10151786197628920/?s=keen-io>

# Complex Applications

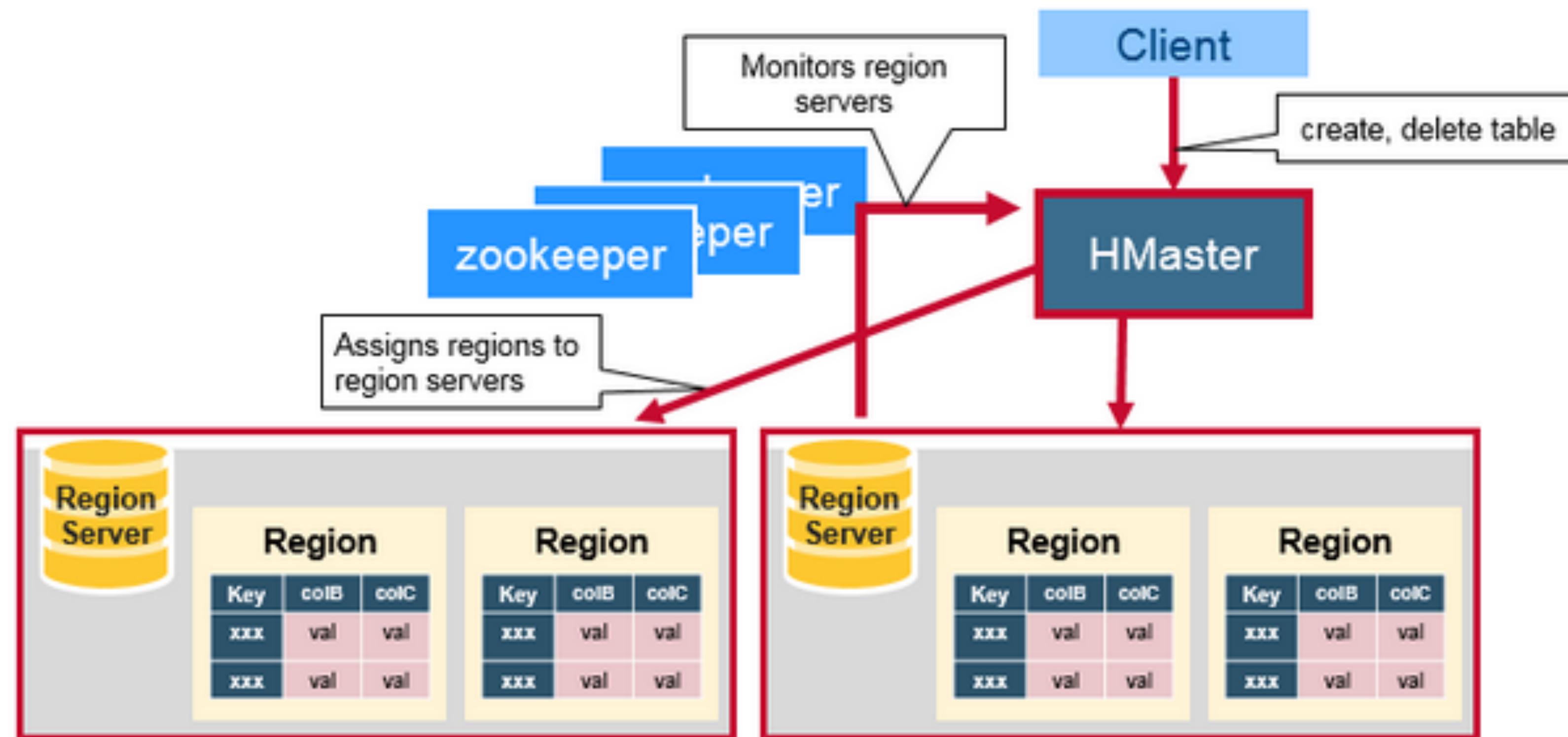
## Facebook – Presto Backends



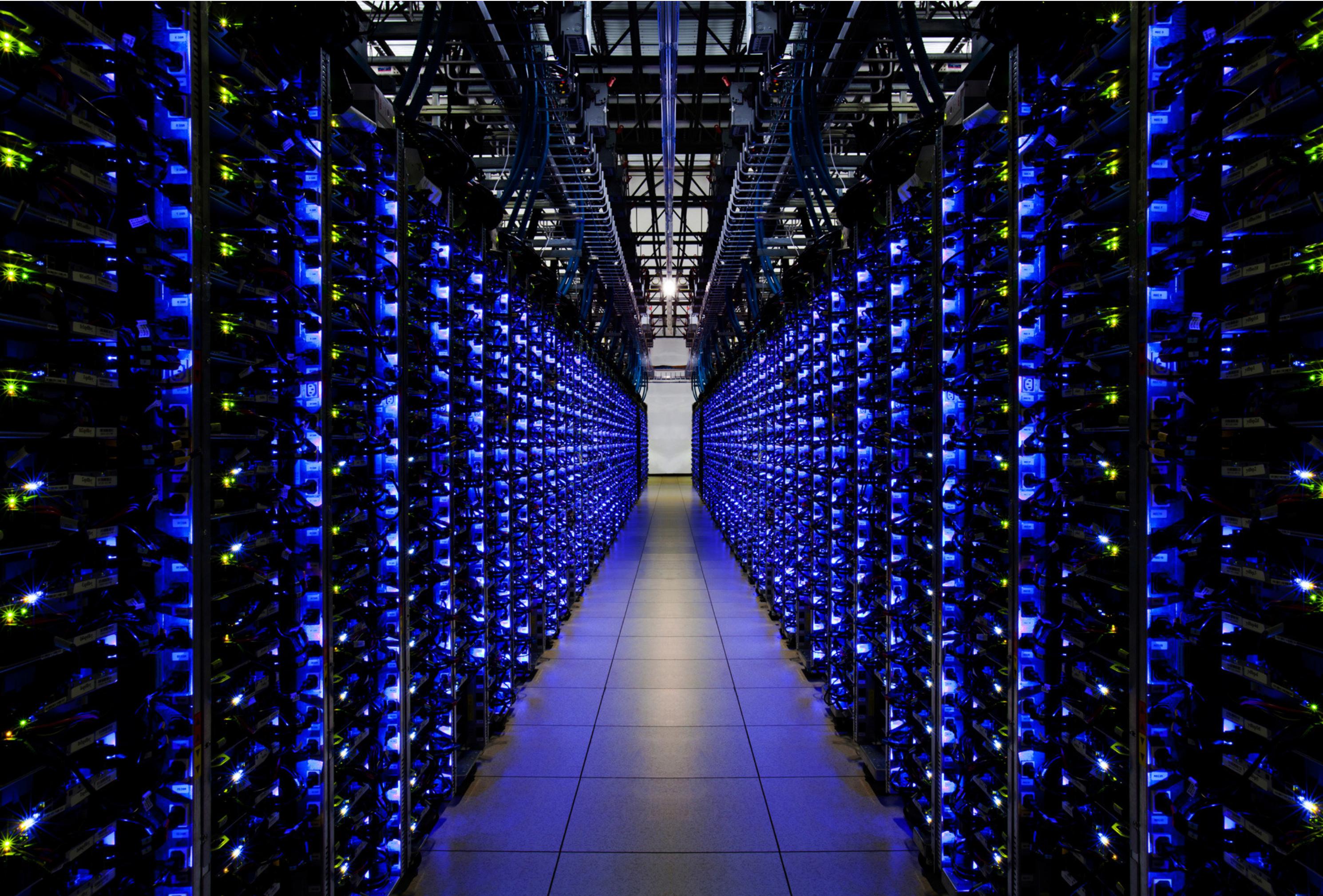
Source: <https://www.facebook.com/notes/facebook-engineering/presto-interacting-with-petabytes-of-data-at-facebook/10151786197628920/?s=keen-io>

# Complex Applications

## HBase



# Complex Infrastructures



# Challenges

## ● Deployment

- ▶ Applications have multiple components with different requirements (e.g., hardware, libraries)
- ▶ Heterogeneous hardware, software and services (e.g., HPC services, cloud computing services)



## ● Management & Configuration

- ▶ Specific to each software/hardware component
- ▶ Optimal configurations will change over time...



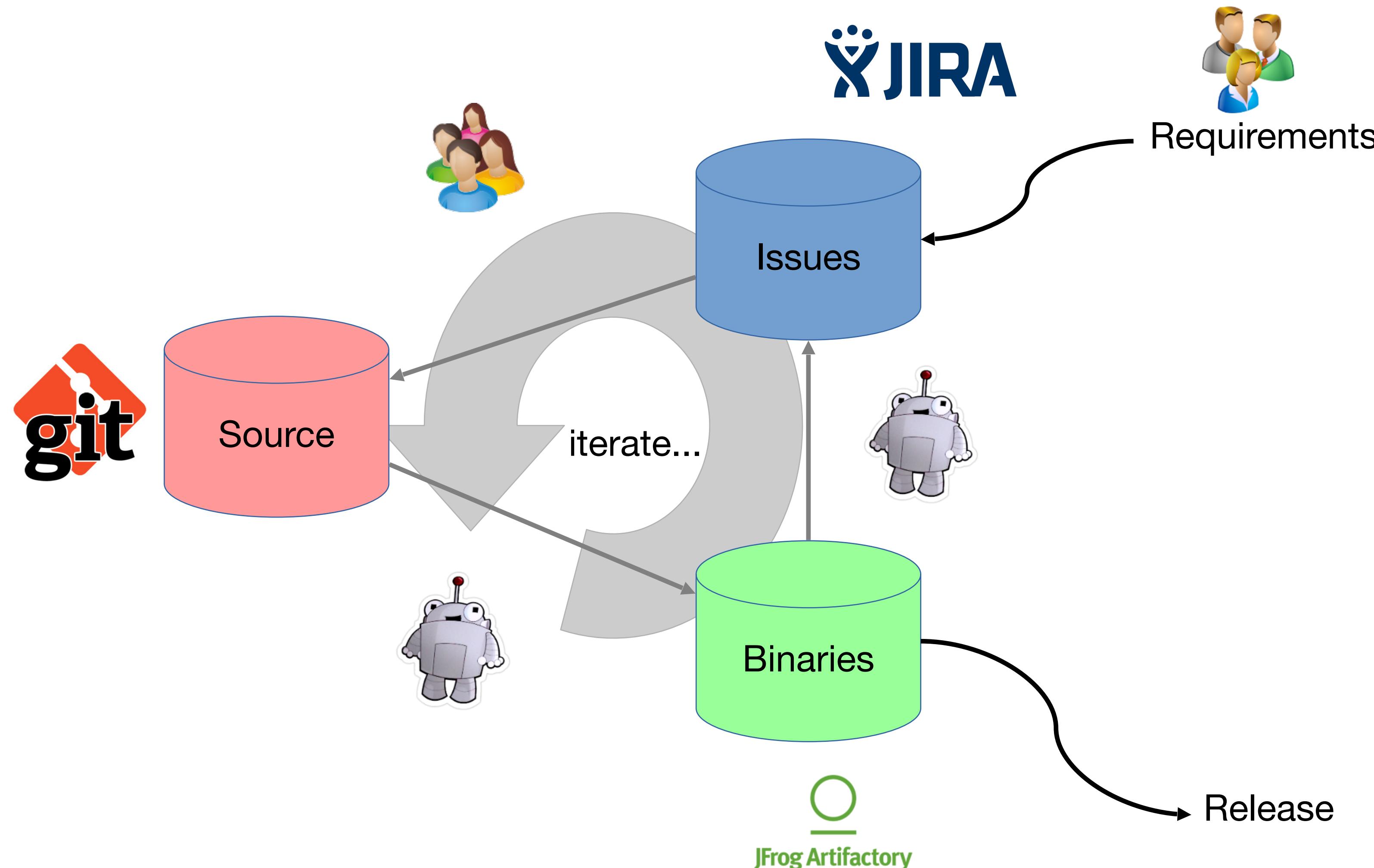
Google Cloud Platform

## ● Monitoring and Benchmarking

- ▶ Finding anomalies (performance, failures, ...) in complex applications and infrastructures

# Yet another example...

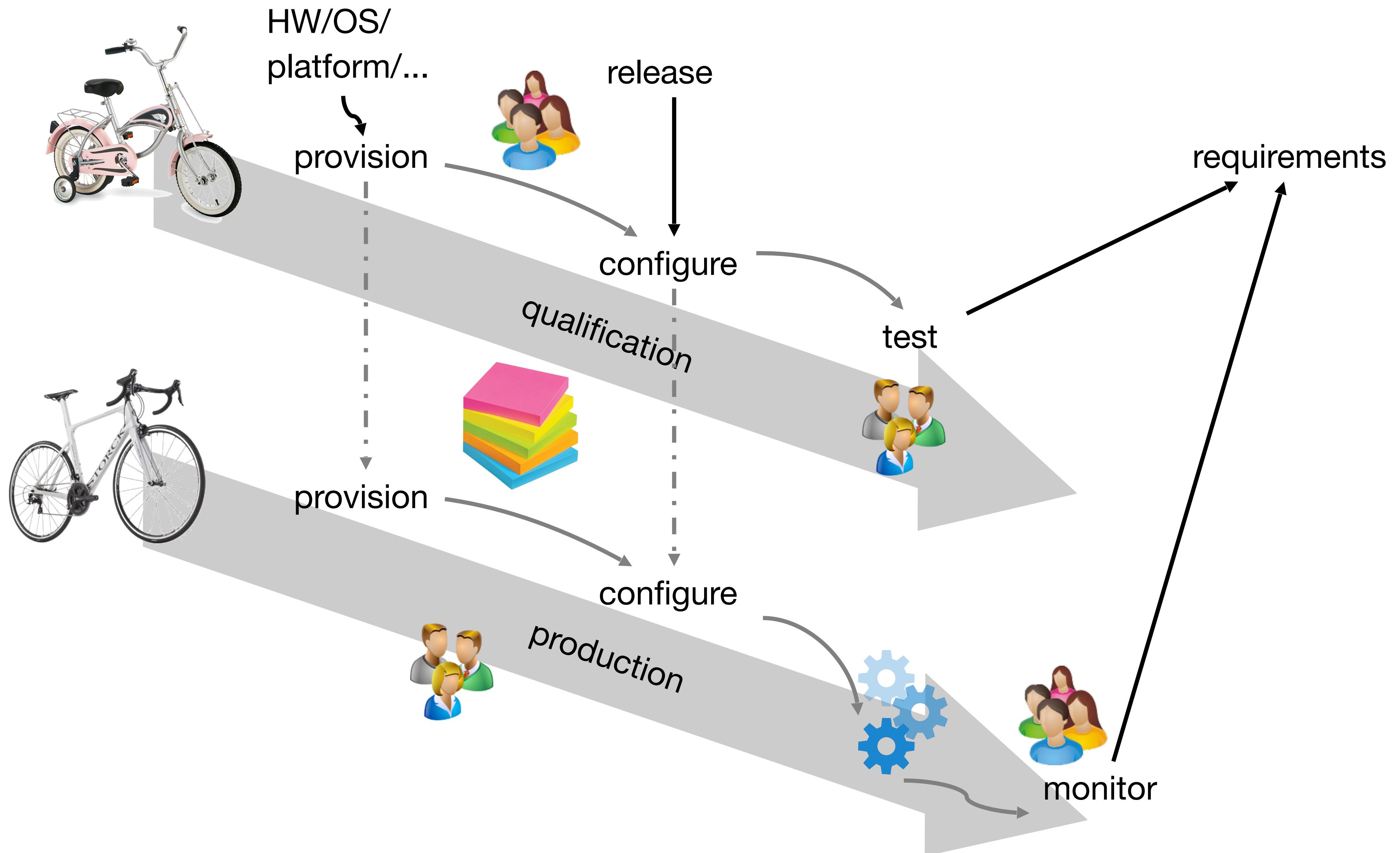
# Agile development



# Agile development

- Rests on formal (versioned) source, binary and documentation repositories
  - ▶ Unambiguous current state
  - ▶ Ability to back track
- Automated build and test
  - ▶ Fast feedback
- Fast iterations
- **Quick reaction to frequent small changes!**

# Operations



# Operations

- Manual provisioning and configuration
- Informal communication between qualification and production stages
- Consequences:
  - ▶ Not repeatable or reproducible
  - ▶ Configuration drift leading to “Snowflake” servers  
(don't even look at it...)
  - ▶ Subjective monitoring

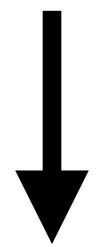
# Infrastructure as Code

- Hardware provisioning with scripts
  - ▶ In contrast to: physically unboxing, plugging and configuring
- Software provisioning with scripts
  - ▶ In contrast to: clicking through setup wizards
- Configuration with scripts
  - ▶ In contrast to: clicking through control panels
- Include infrastructure scripts in agile process!

# Agile development

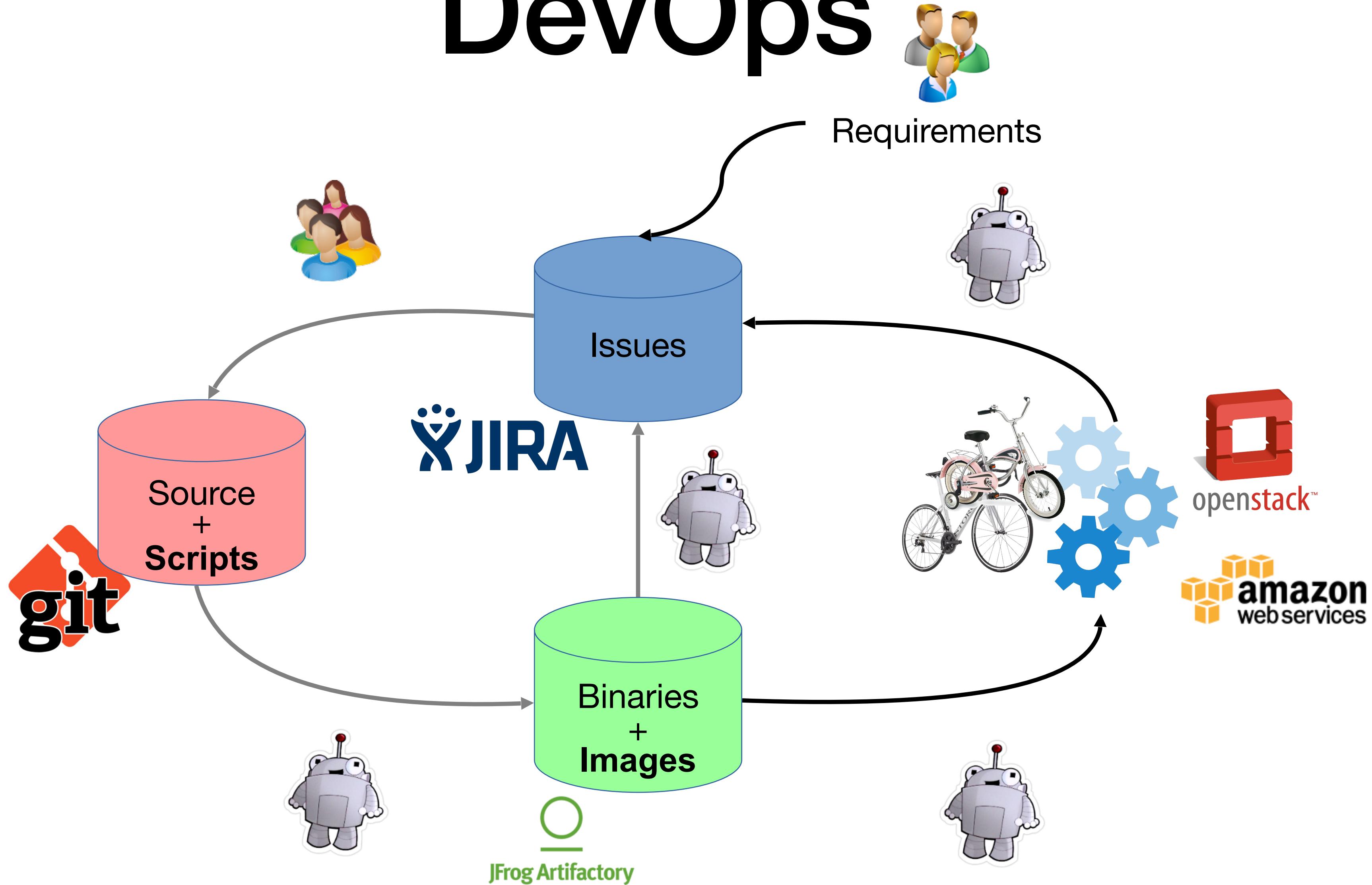
+

# Infrastructure as Code



# DevOps

# DevOps



# DevOps

- Development and deployment are self-documenting and versioned
- Deployment is reproducible and repeatable
- Servers are disposable and consistent
- **Supports fast, frequent, small changes!**

# Roadmap

- Distributed applications architectures & deployment

- ▶ Common distribution patterns
- ▶ Reliability and scalability
- ▶ Automatic provision and configuration

- Cloud computing

- ▶ Services and abstractions
- ▶ Infrastructure management
  - virtualization, storage, ...

- Monitoring and evaluation

- ▶ Monitoring frameworks and metrics
- ▶ Benchmarking

# Assessment

- Project (50%) – minimum grade: 10 values
  - ▶ Checkpoint #1: 27/10/2024
  - ▶ Checkpoint #2: 01/12/2024
  - ▶ Final Report and Submission: 03/01/2025
  - ▶ Presentation: 06-11/01/2025 (week)
- Written exam (50%) – minimum grade: 8 values
  - ▶ 10/12/2024

# Project

- Fill the following form until **September 20th**

- ▶ <https://forms.gle/sVq1aKJrXQWpWdeC6>
- ▶ We will use this information to add you to ASCN's GitHub organization
- ▶ Check your e-mail for an invitation after this date!

- ASCN Github organization

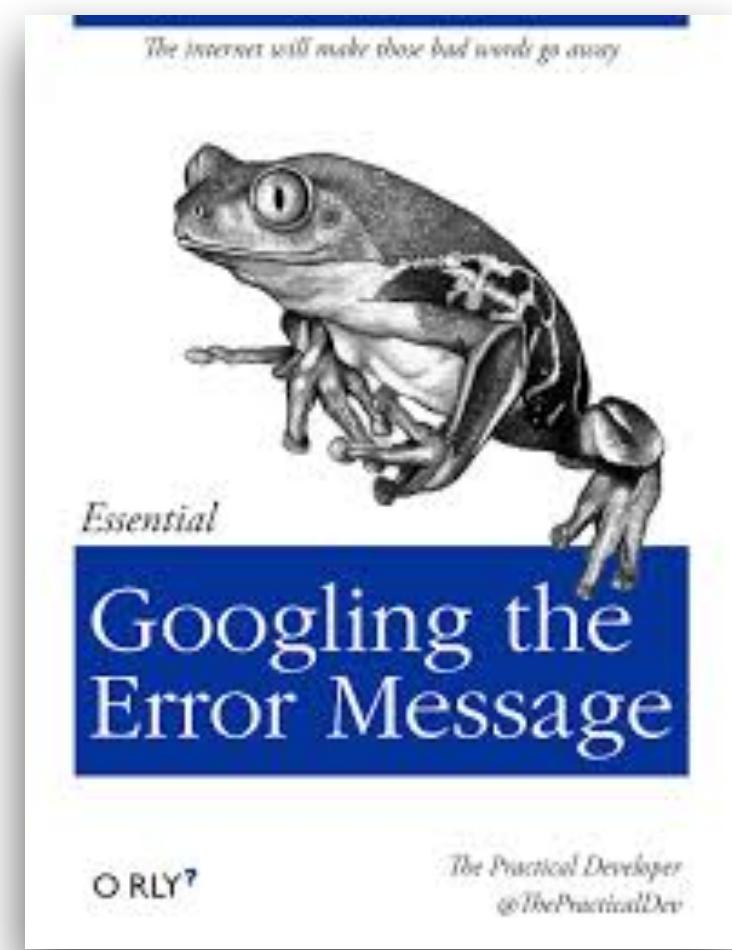
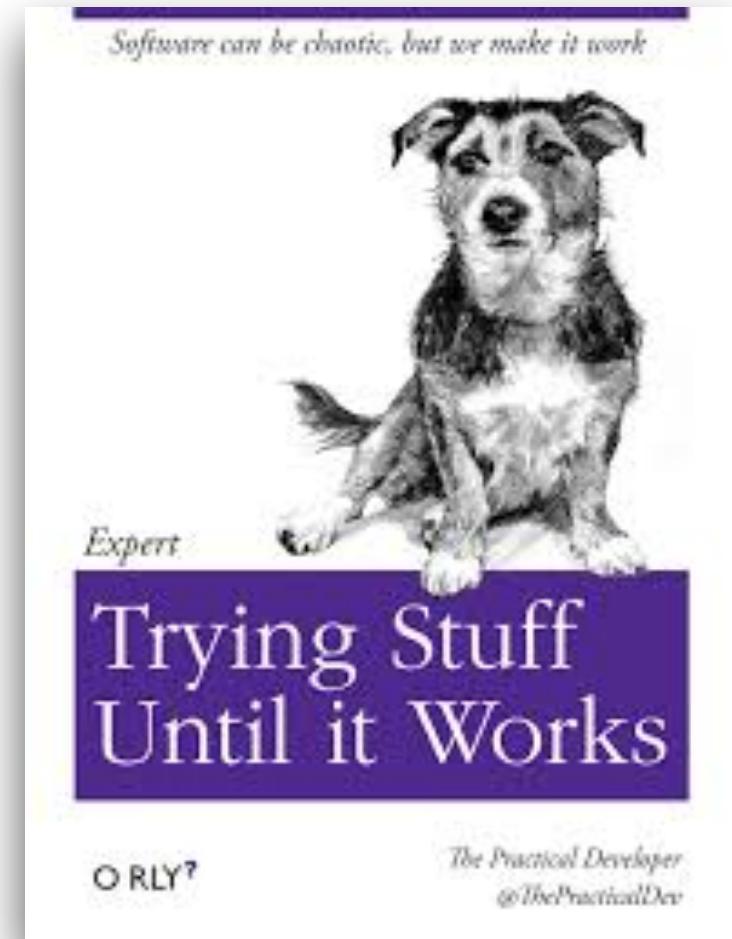
- ▶ When you are granted access **carefully read** the “How to use the ASCN-UM organization” at the **Instructions discussion thread**

- Each group should have **5 elements**

- ▶ Start talking with your colleagues to find a group!
- ▶ The organization has a **Groups discussion thread** to find members/groups

# Main References

- K. Morris. *Infrastructure as Code: Managing Servers in the Cloud*. O'Reilly, 2016
- M. Kleppmann. *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*. O'Reilly, 2017
- T. Erl, R. Puttini e Z. Mahmood. *Cloud Computing: Concepts, Technology and Architecture*. Prentice Hall, 2013
- R. Jain. *The art of computer systems performance analysis - techniques for experimental design, measurement, simulation, and modeling*. Wiley, 1991



# Team

- João Paulo – [jtpaulo@di.uminho.pt](mailto:jtpaulo@di.uminho.pt)
- Francisco Maia – [fmaia@di.uminho.pt](mailto:fmaia@di.uminho.pt)
- Tânia Esteves – [d12729@di.uminho.pt](mailto:d12729@di.uminho.pt)
- Cláudia Brito – [d13235@di.uminho.pt](mailto:d13235@di.uminho.pt)
- Pedro Moreira – [d14112@uminho.pt](mailto:d14112@uminho.pt)



# Complementary Classes

- Advanced Computing (AC)
- Distributed Systems (SD)
- Applications Engineering (EA)
- Cryptography and Information Security (CSI)

# Questions?