

# Microservices in a Data Science World

@jhug | @NikosDimos





Workable is  
reinventing hiring for  
**5000+**  
**companies**

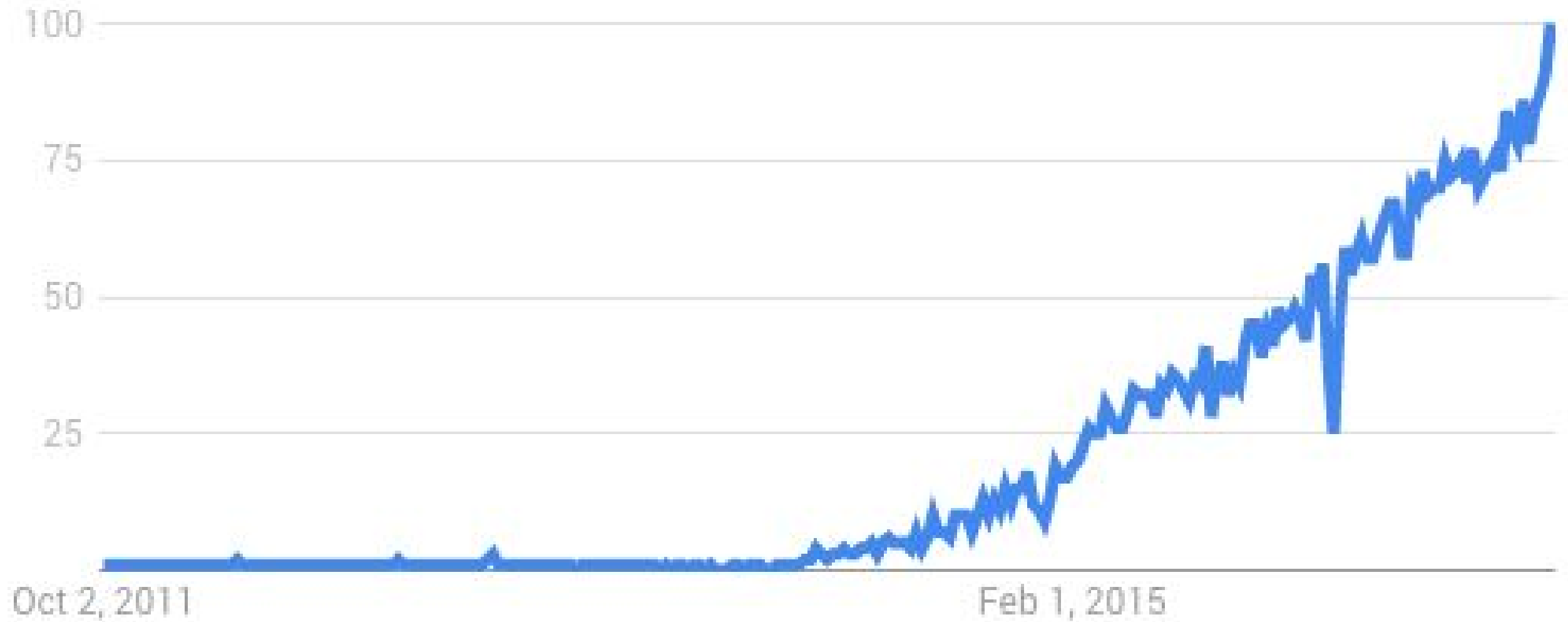
in over **80 countries**. Hundreds  
more are joining every month



## Interest over time

Google Trends

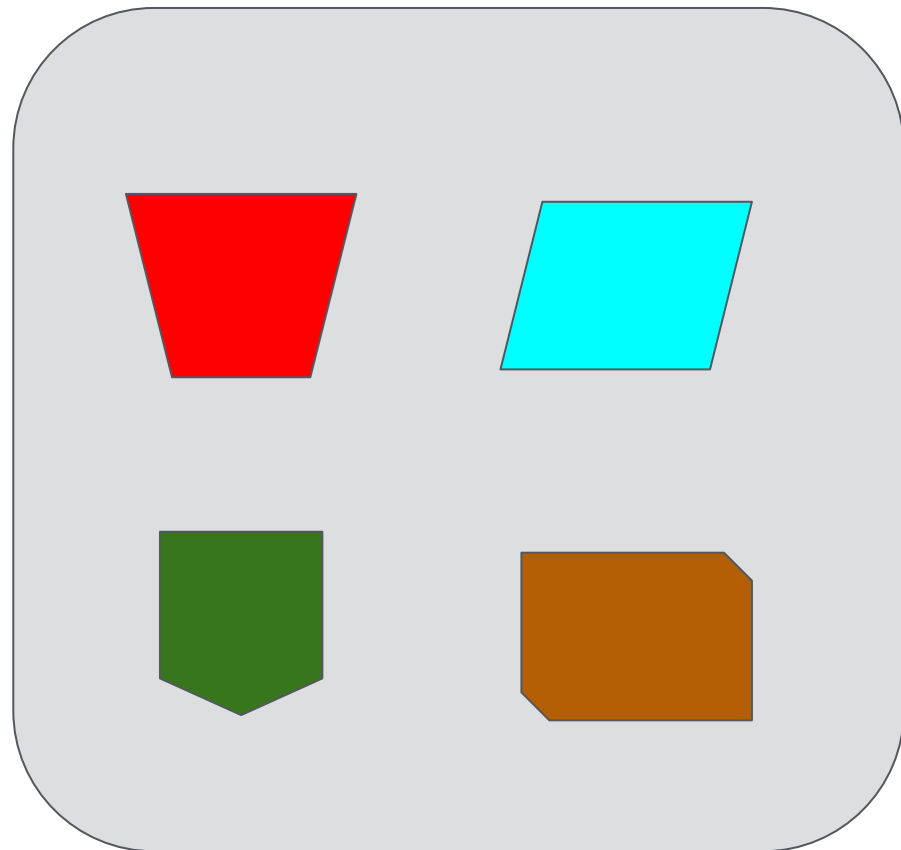
● microservices



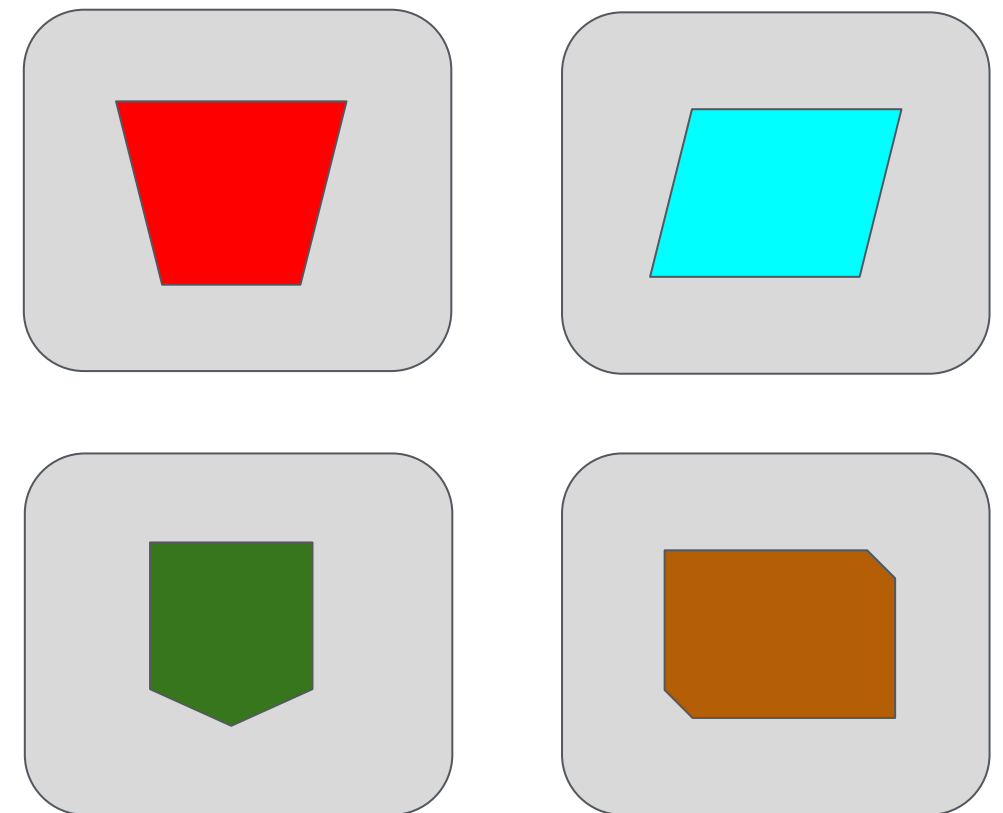
Worldwide. Past 5 years.



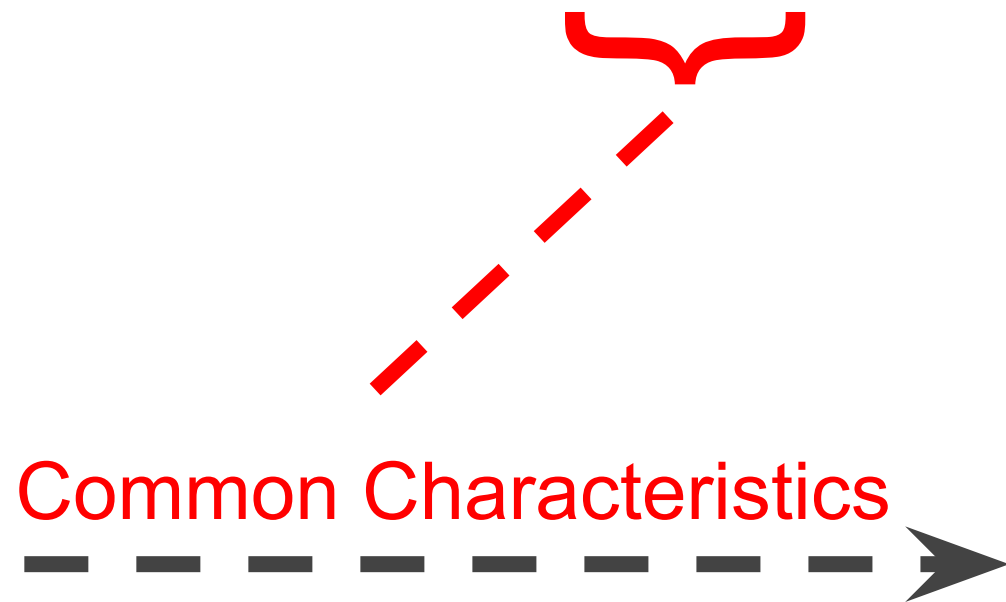
# Monolith



# Microservice



# Definition of Microservices



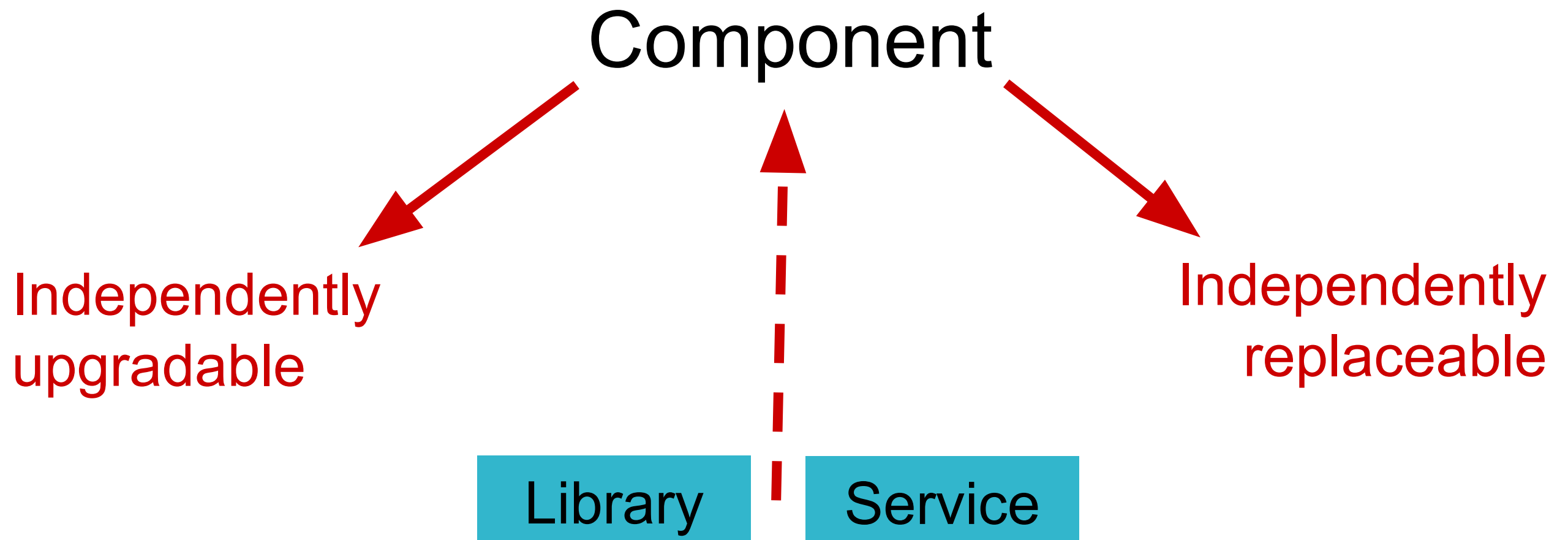
1. Componentization via services
2. Organized around business capabilities
3. Products not Projects
4. Smart endpoints and dumb pipes
5. Decentralized Governance
6. Decentralized Data Management
7. Infrastructure Automation
8. Design for failure
9. Evolutionary Design

*<http://martinfowler.com/articles/microservices.html>*

# Common characteristics

1. Componentization via services
2. Organized around business capabilities
3. Products not Projects
4. Smart endpoints and dumb pipes
5. Decentralized Governance
6. Decentralized Data Management
7. Infrastructure Automation
8. Design for failure
9. Evolutionary Design

# Componentization via Services



# Common characteristics

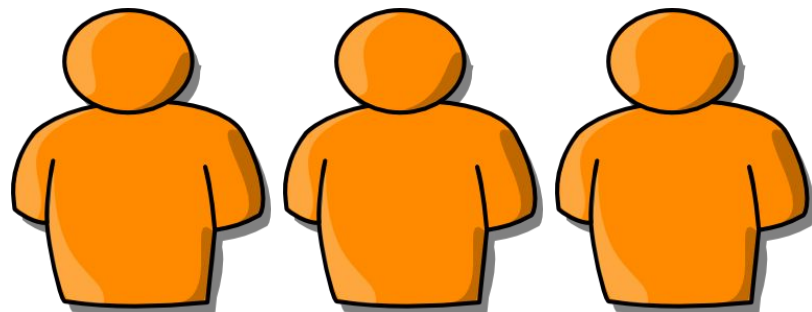
1. Componentization via services
2. Organized around business capabilities
3. Products not Projects
4. Smart endpoints and dumb pipes
5. Decentralized Governance
6. Decentralized Data Management
7. Infrastructure Automation
8. Design for failure
9. Evolutionary Design



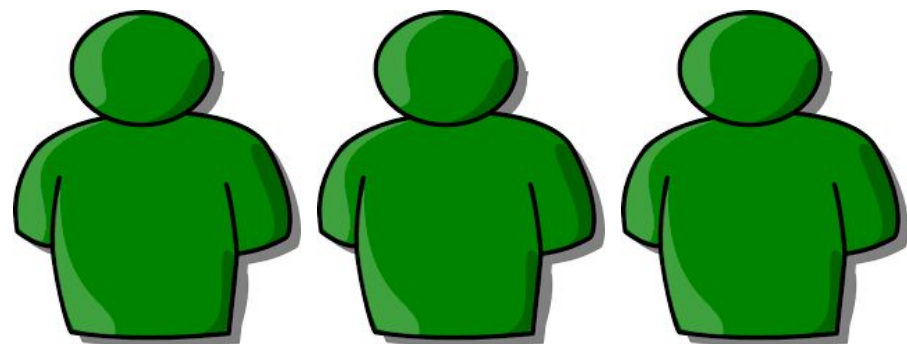
# Organized around business capabilities



UI

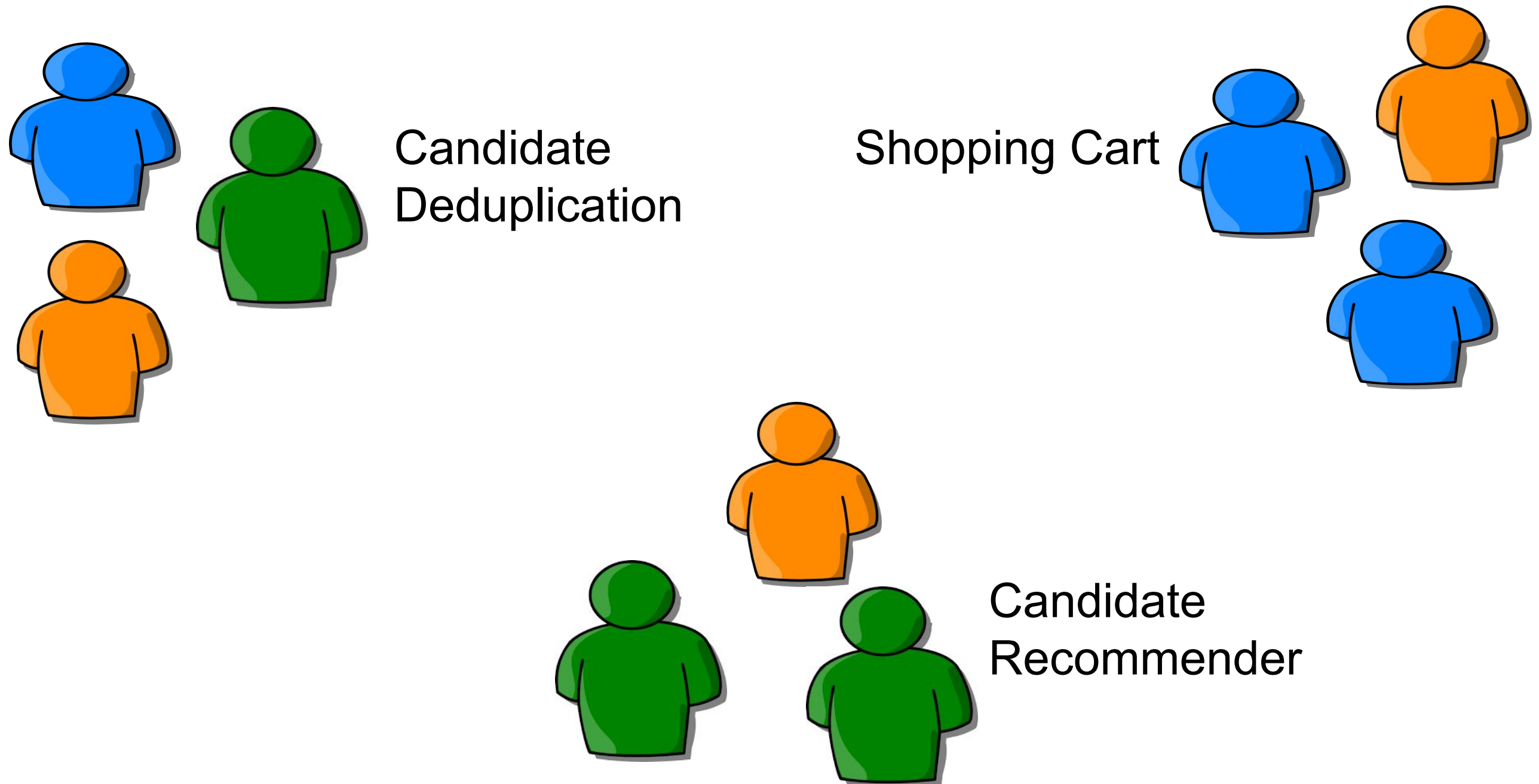


SERVER



DATA SCIENCE

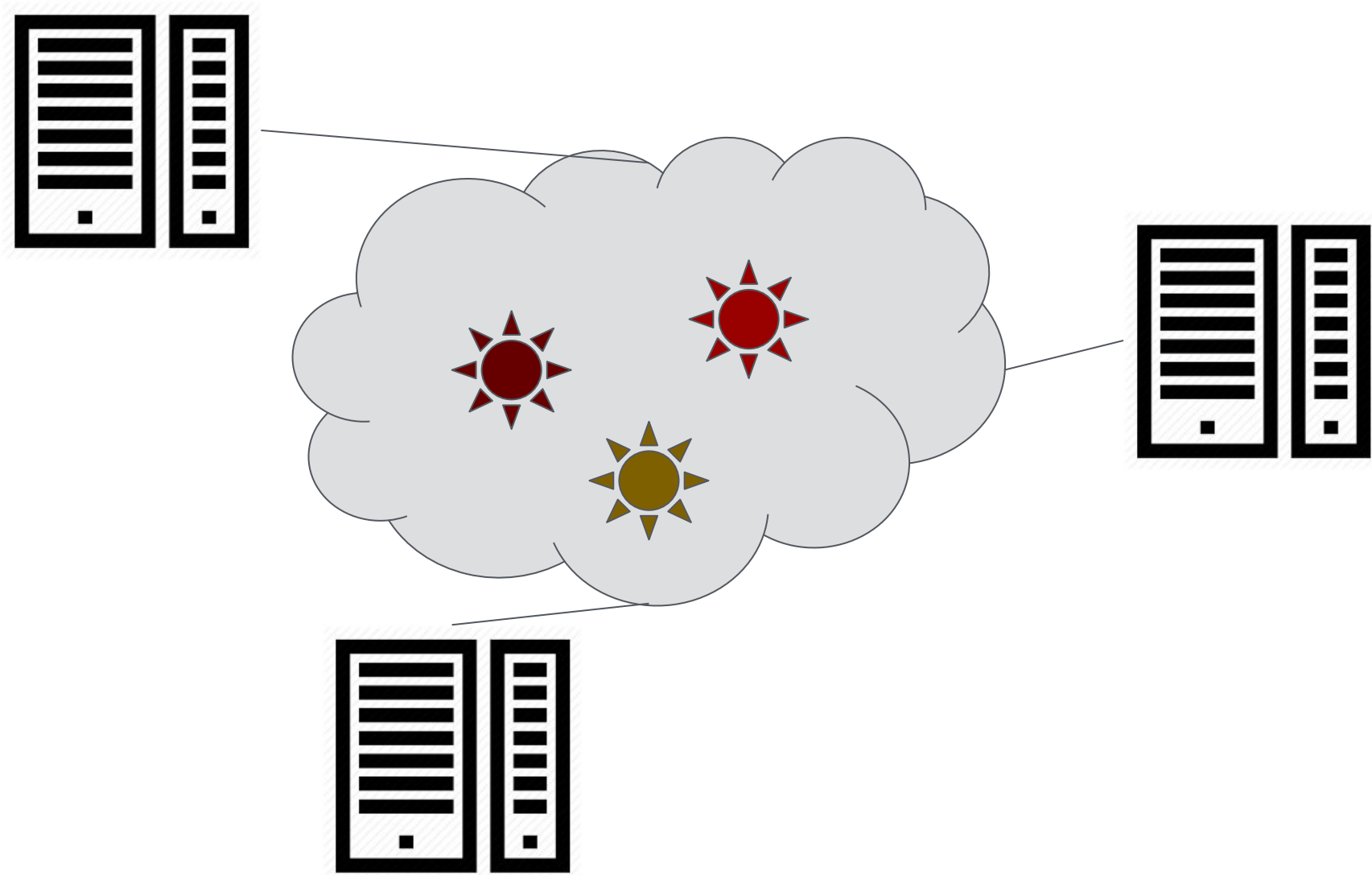
# Organized around business capabilities



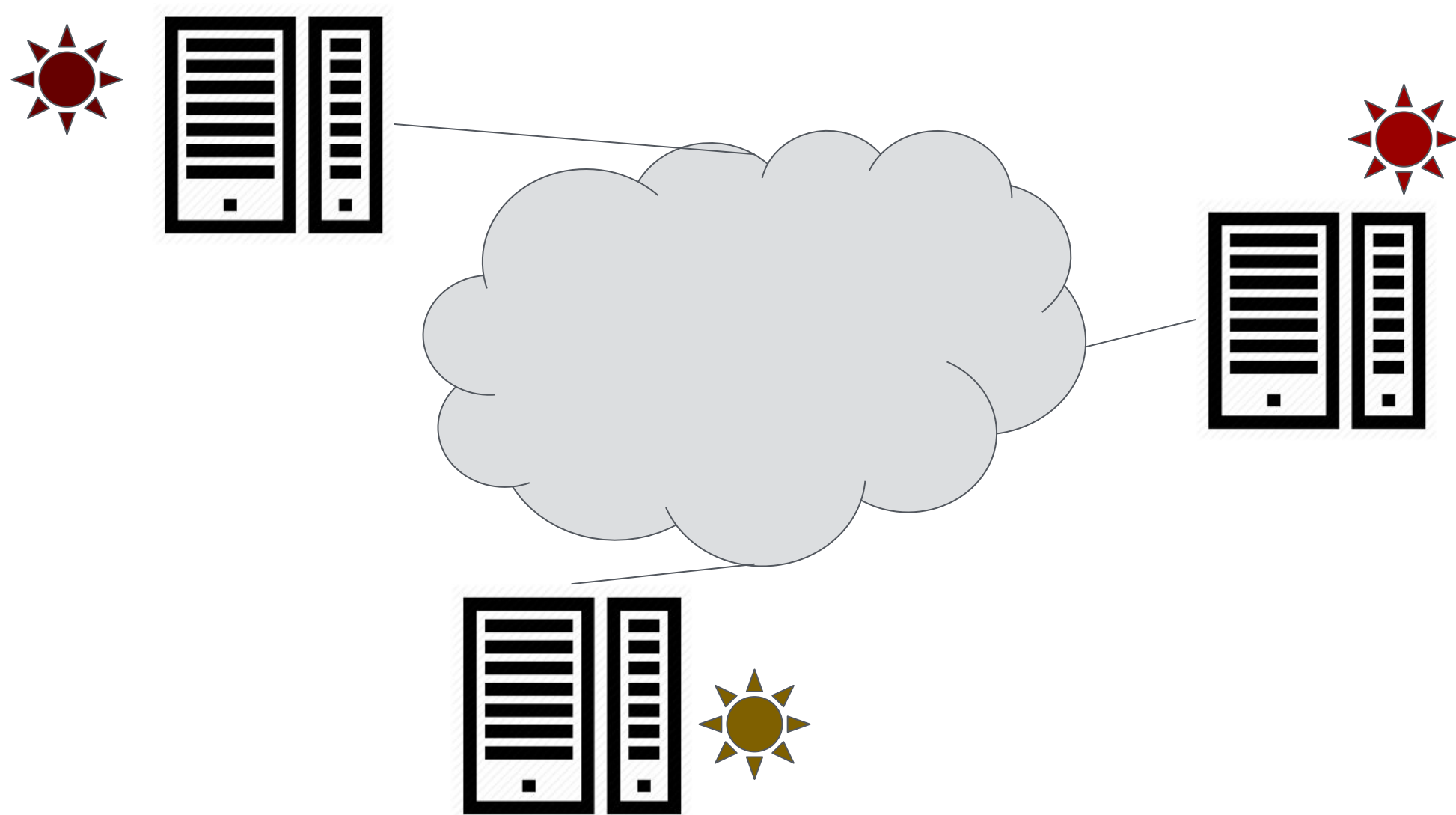
# Common characteristics

1. Componentization via services
2. Organized around business capabilities
3. Products not Projects
4. Smart endpoints and dumb pipes
5. Decentralized Governance
6. Decentralized Data Management
7. Infrastructure Automation
8. Design for failure
9. Evolutionary Design

# Smart Endpoints Dumb Pipes



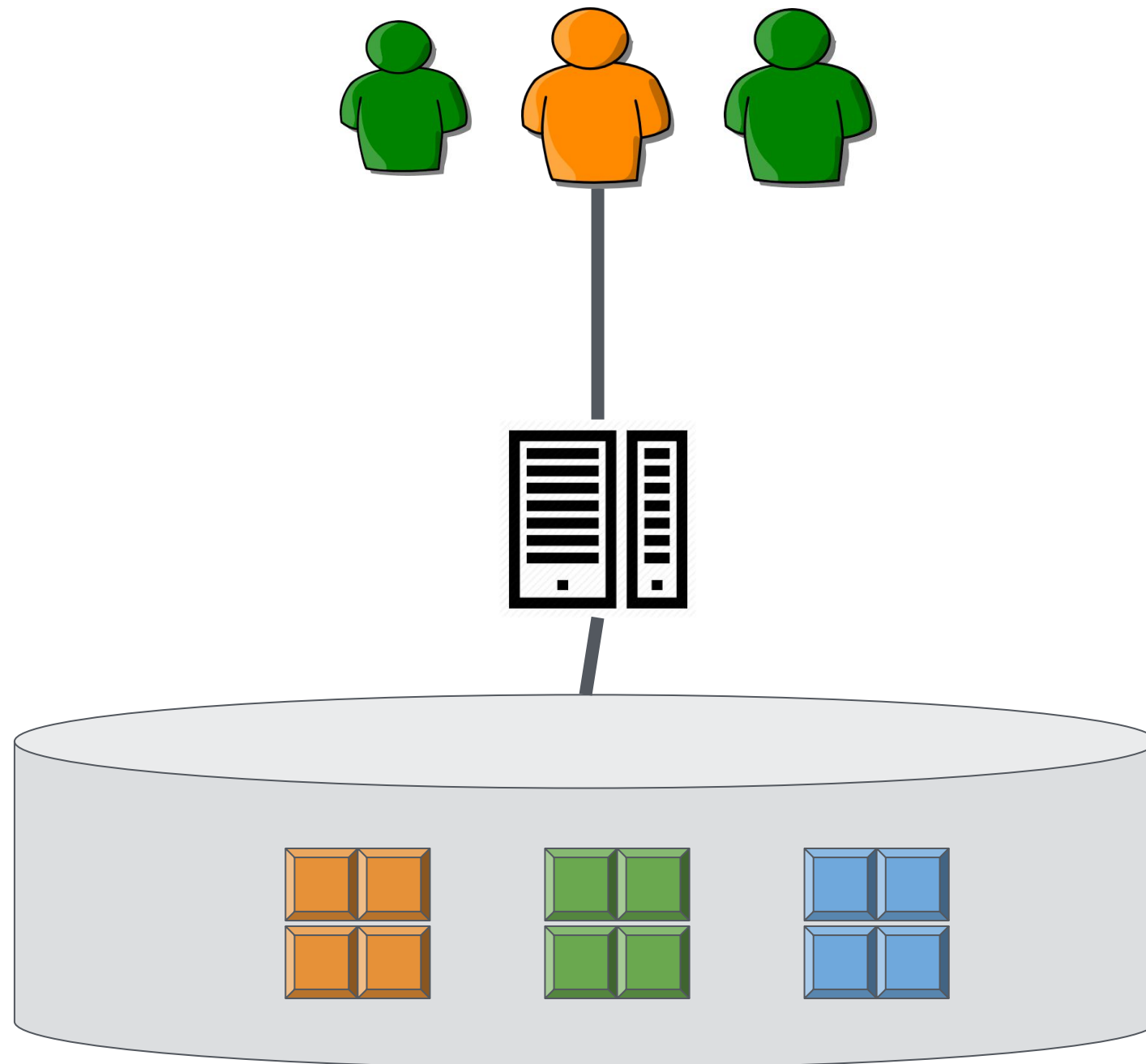
# Smart Endpoints Dumb Pipes



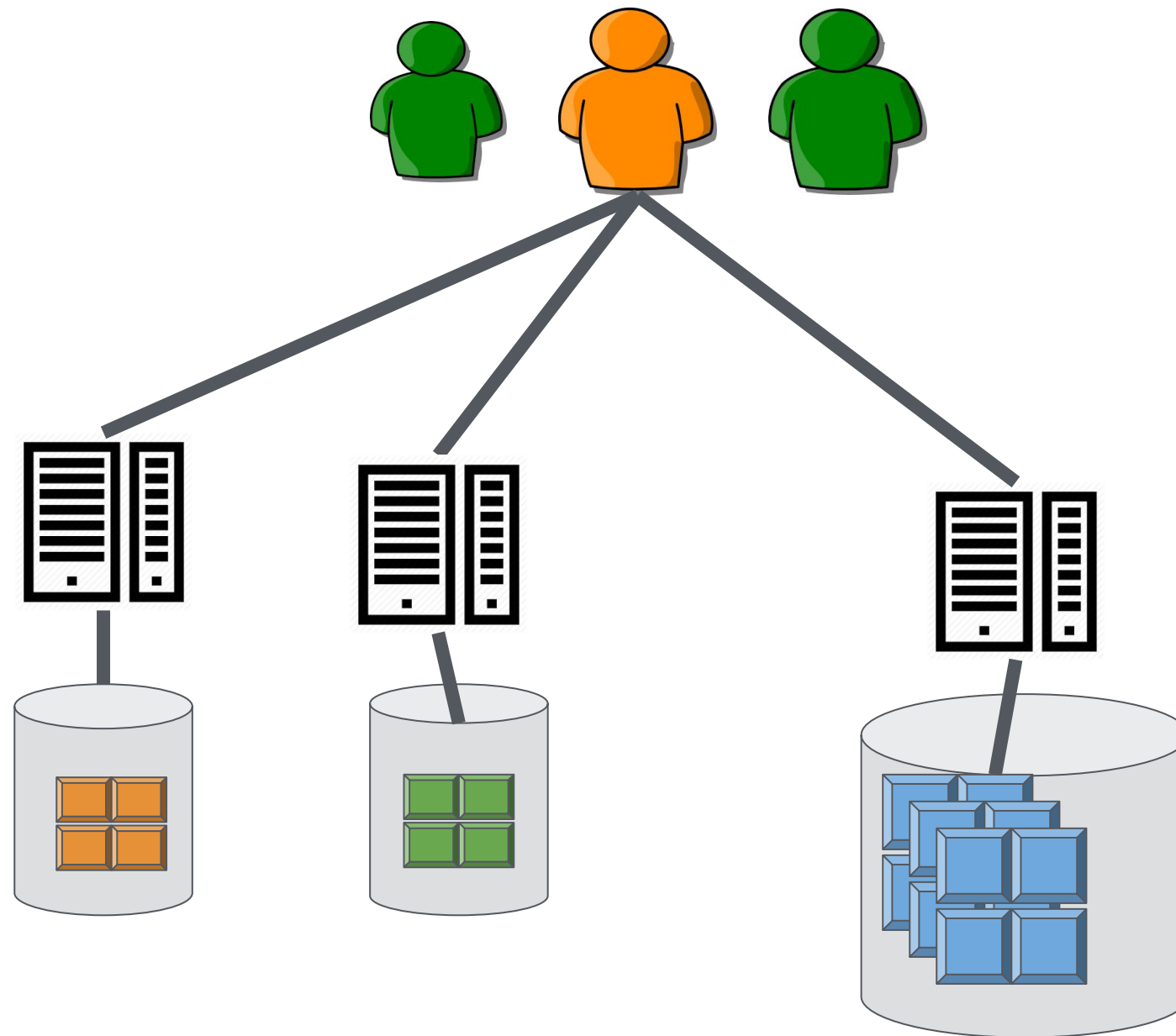
# Common characteristics

1. Componentization via services
2. Organized around business capabilities
3. Products not Projects
4. Smart endpoints and dumb pipes
5. Decentralized Governance
6. Decentralized Data Management
7. Infrastructure Automation
8. Design for failure
9. Evolutionary Design

# Decentralization of Data Management



# Decentralization of Data Management





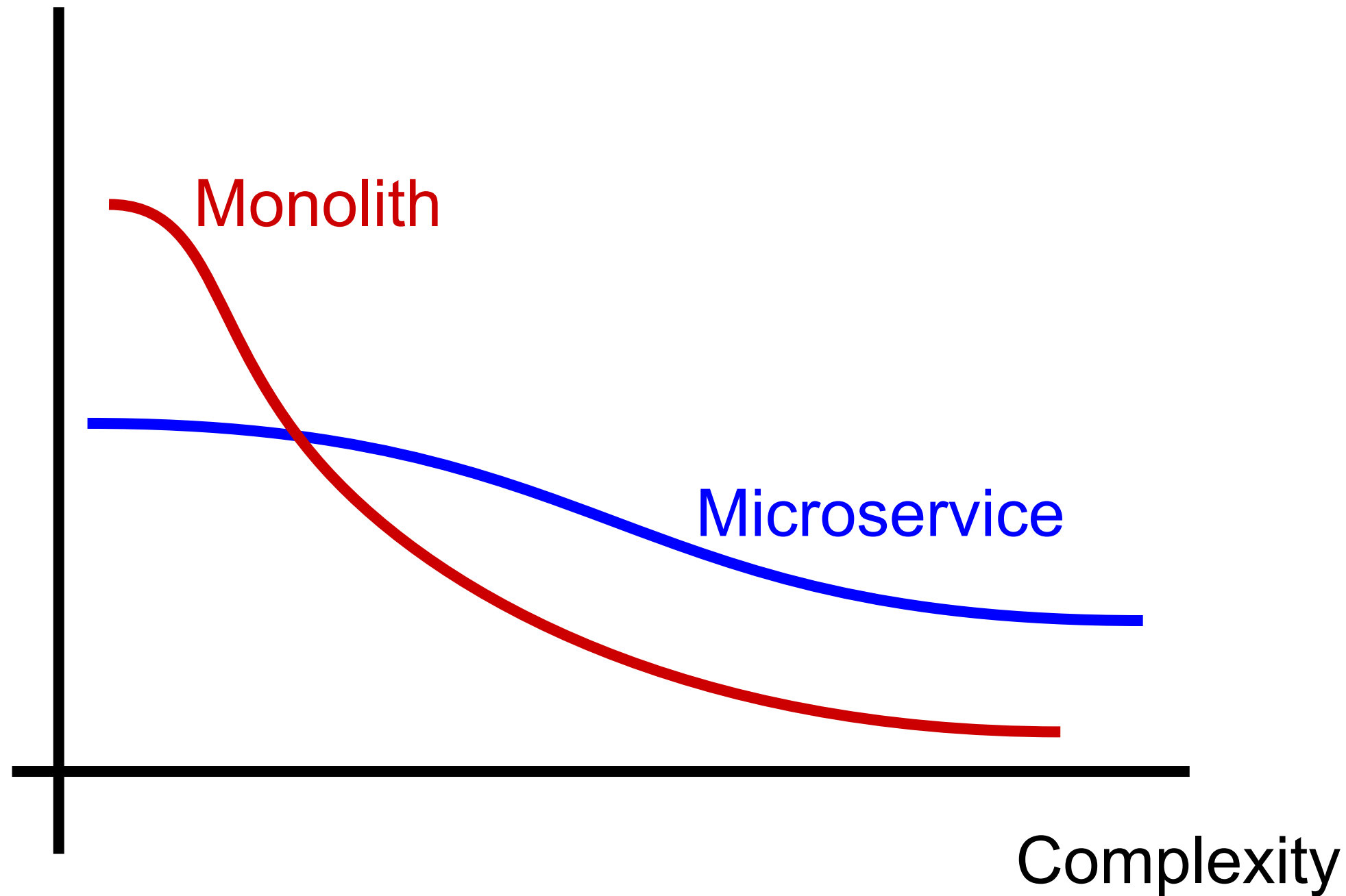
## Benefits

- + Strong Module Boundaries
- + Independent Deployment
- + Technology Diversity

## Costs

- Distribution
- Eventual Consistency
- Operational Complexity

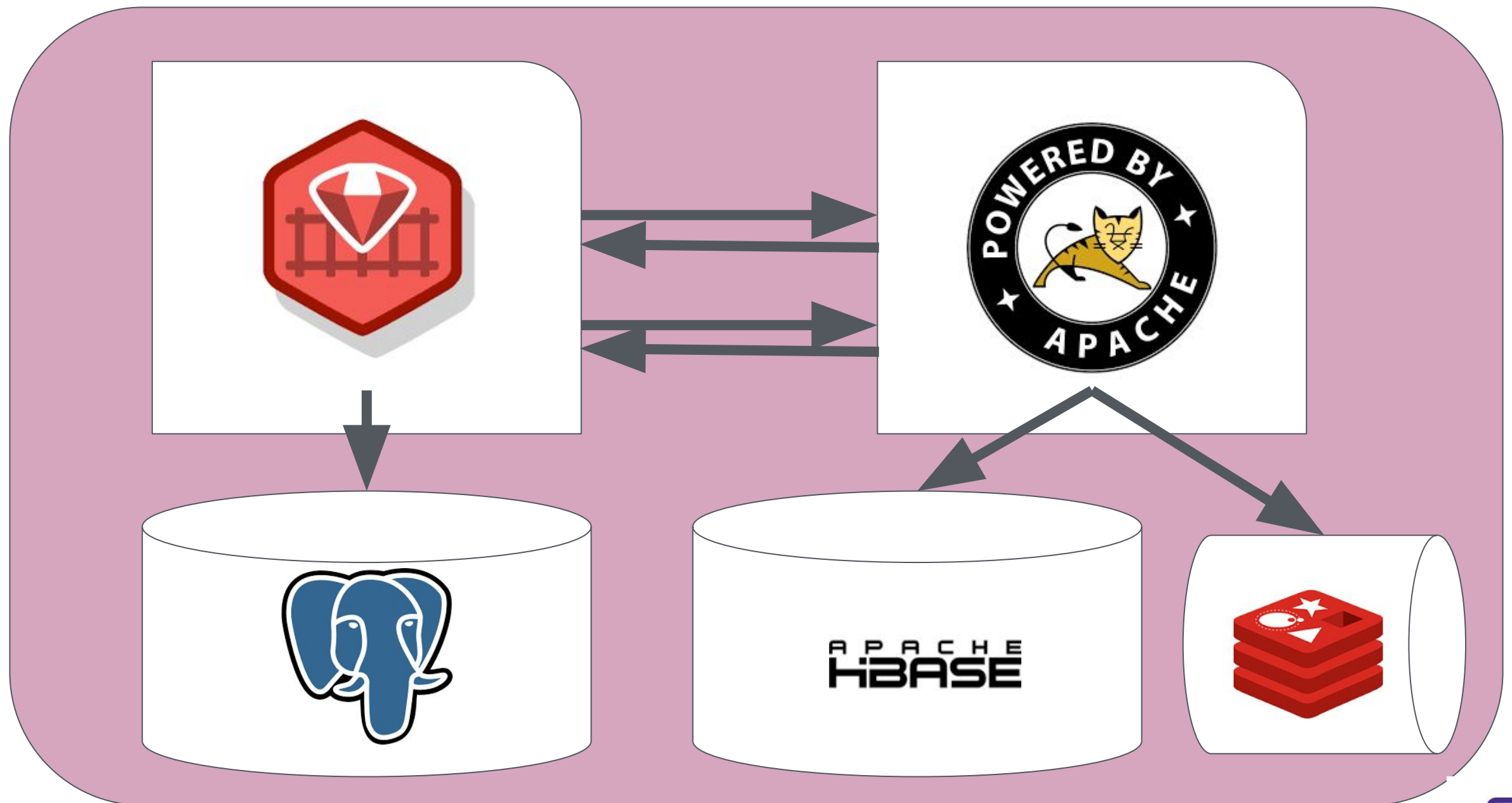
# Productivity



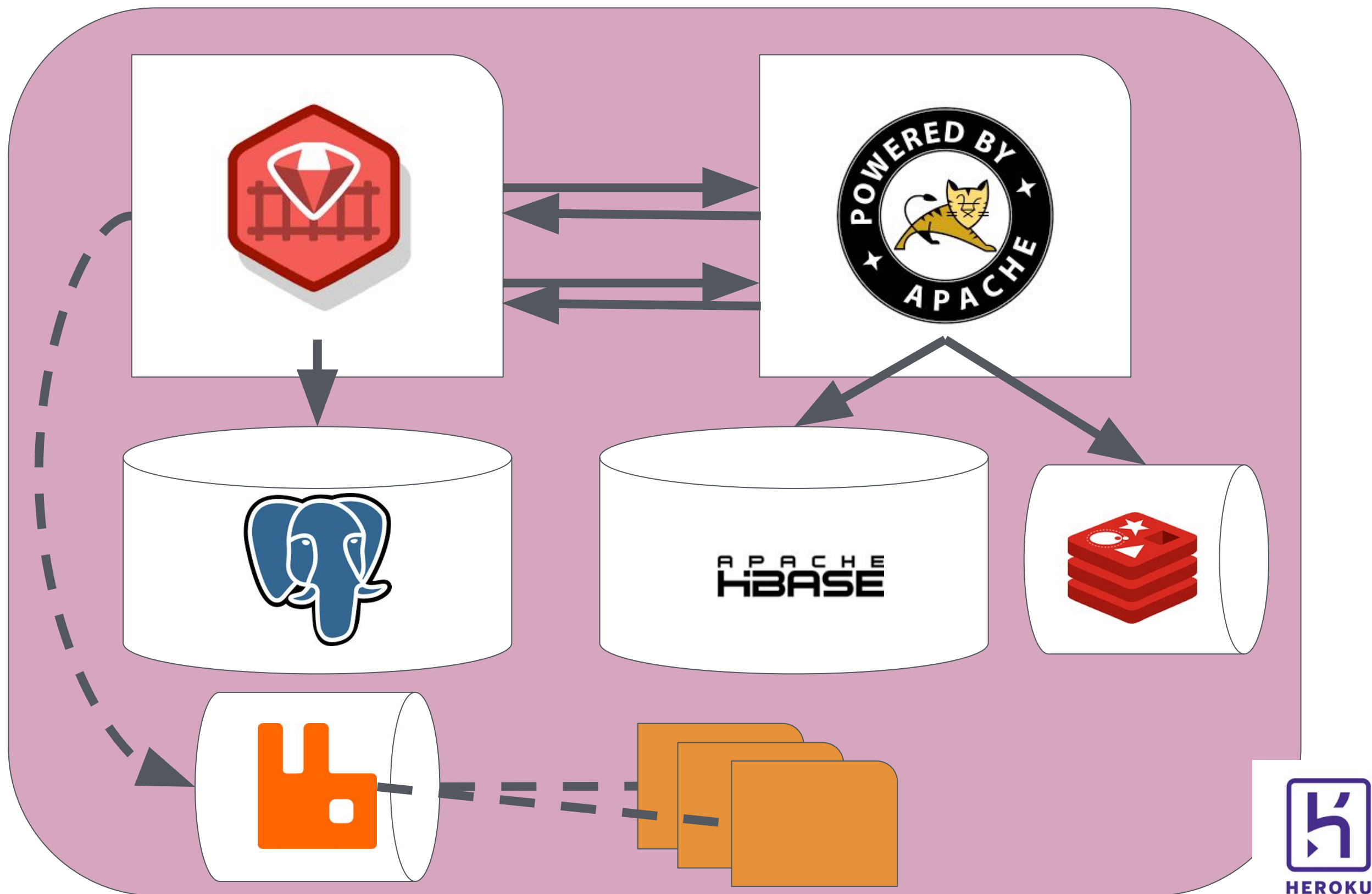
# Workable journey to Microservices



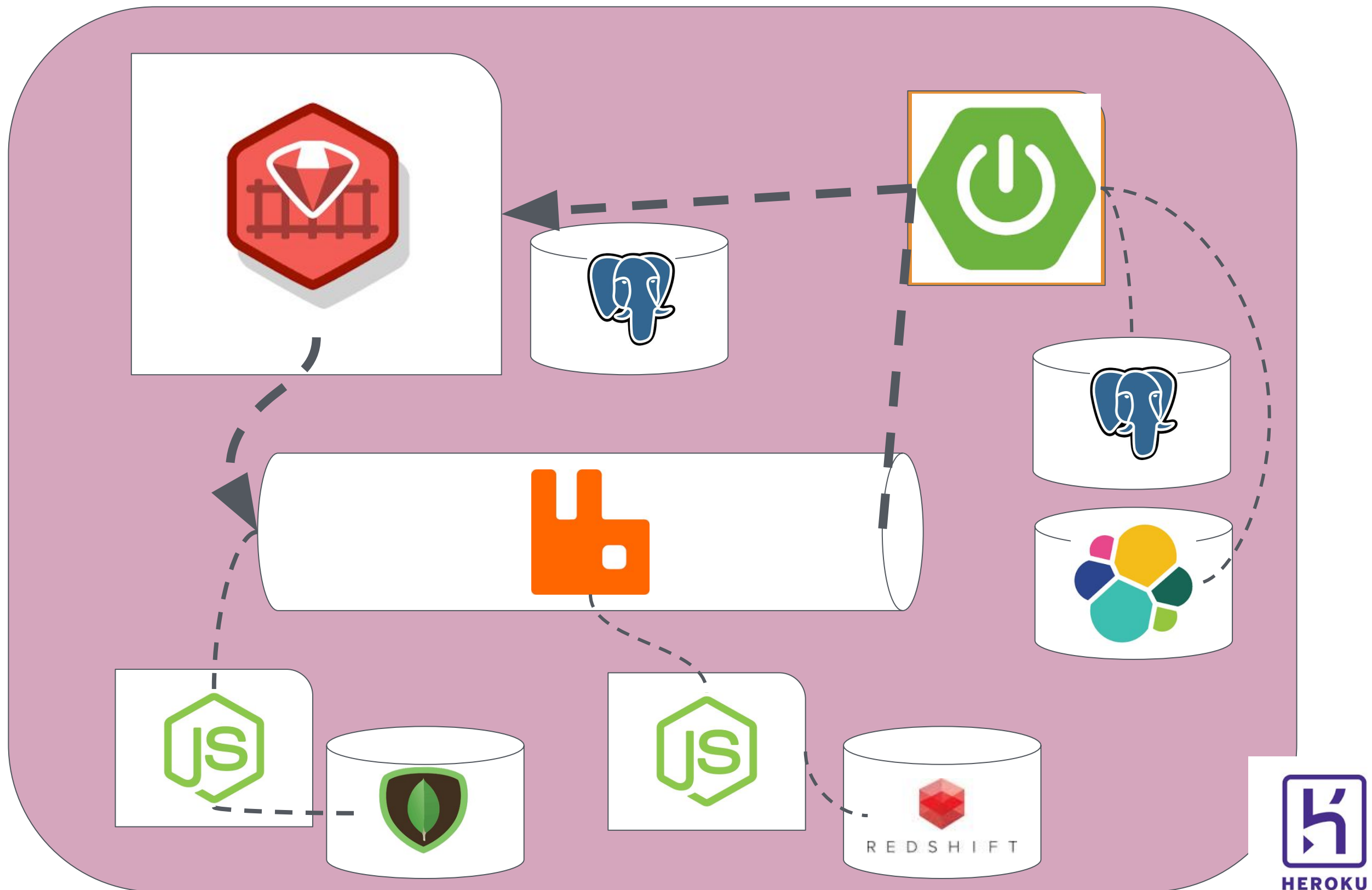
# Workable Gen 1.0



# Workable Gen 1.5

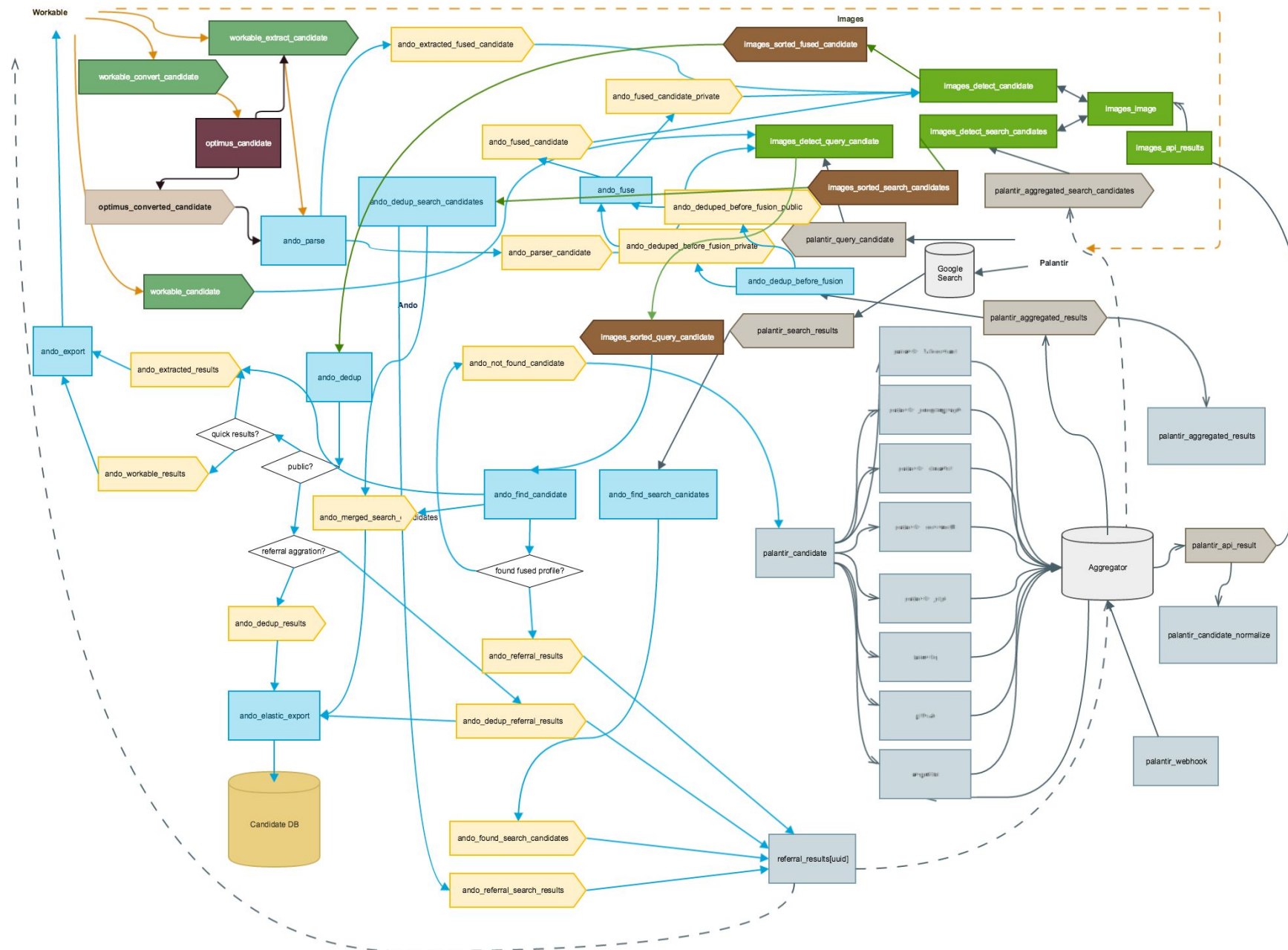


# Workable Gen 2.0



# Workable Gen 2.0

# Until we reached this...



# Workable Lessons Learned

## Things to improve

- Rapid provisioning - ***Heroku***
- Basic Monitoring - ***Newrelic***
- Rapid Application Deployment - ***Heroku***
- Traceable Business Transactions - ***Kafka***
- Devops Culture
- Continuous Delivery
- Product-centered Teams



# Thanks.

@workable | @NikosDimos

