



# Robust R Deployments: Building a Pipeline from RStudio to Production

David Maguire, Senior Data Scientist



# R is bad for production



## Misconceptions

- R is great for analysis but we need something more robust for production
- R code is ad hoc and untested
- R code is not scalable
- No package & dependency management

# R is bad for production



## Misconceptions

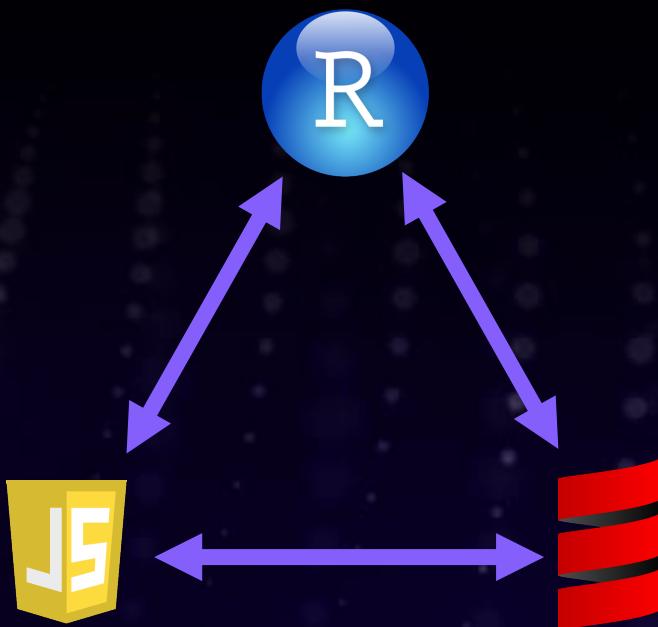
- R is great for analysis but we need something more robust for production
- R code is ad hoc and untested
- R code is not scalable
- No package & dependency management





# Tape Cracker: Customer Facing Machine Learning

- Web application powered by machine learning in R
- Models available 24/7
- Instant results

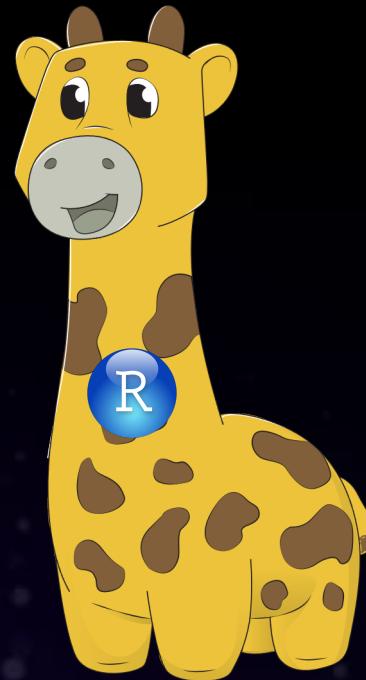


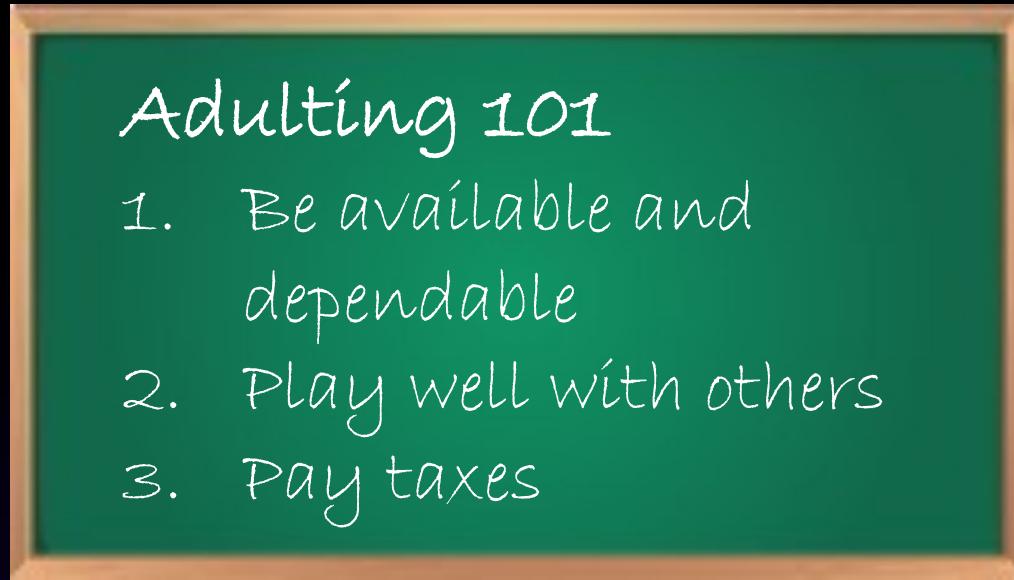
The screenshot shows a web-based data wrangling interface for a machine learning project named "Tape Cracker". The interface includes:

- A top navigation bar with tabs for "FIELD MATCHING" and "DATA WRANGLING".
- A main table titled "CURRENT ACTUAL BALANCE" showing data for various loans, including columns for "Current Actual Balance", "# of loans", "%", "Total Current Balance", and "Current Gross Rate".
- A "Source Values" summary table showing counts for "Valid", "Non-standard", "Invalid", and "Missing" rows.
- A "DATA WRANGLING" section listing fields like "Pingers LTV", "Next Due Date", "CULFICO", and "Hudson LTV" with their corresponding CSV fields and validation status.
- A bottom section showing a list of fields such as "PTTOWNHOUSE", "LPCASHOUT", "NVMT LTV", "Digital LTV", "DOCUM", "PTTOWNHOUSE", "Hudson LTV", "Pingers LTV", and "PTTOWNHOUSE".
- On the right side, there are several horizontal bar charts showing the distribution of values for different fields.

Raising R microservices to thrive in the wild

# Raising R microservices to thrive in the wild





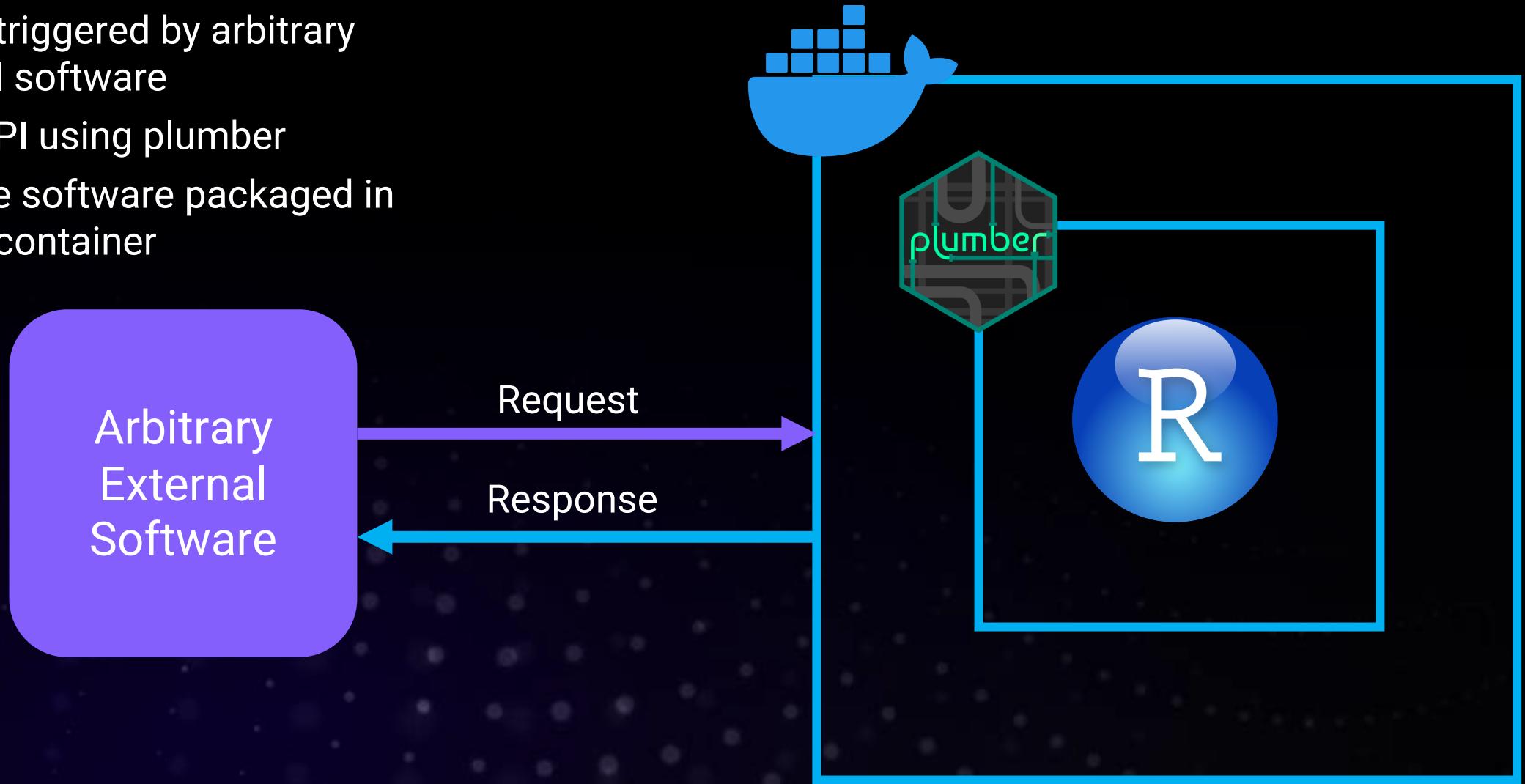
# Raise R microservices to thrive in the wild!

dv01

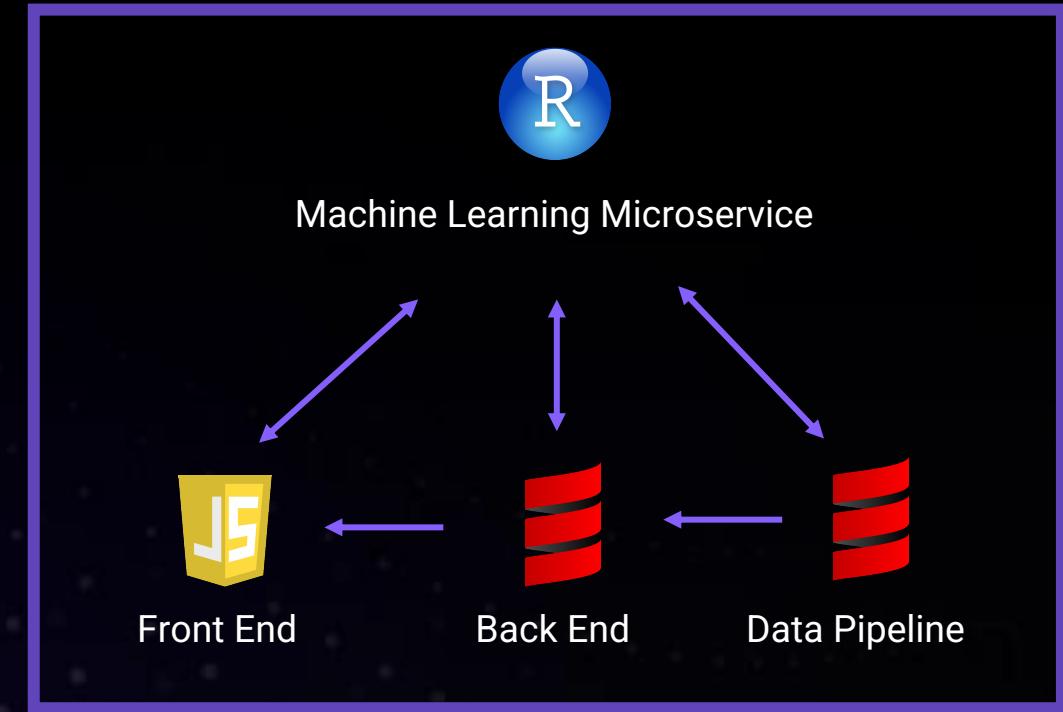
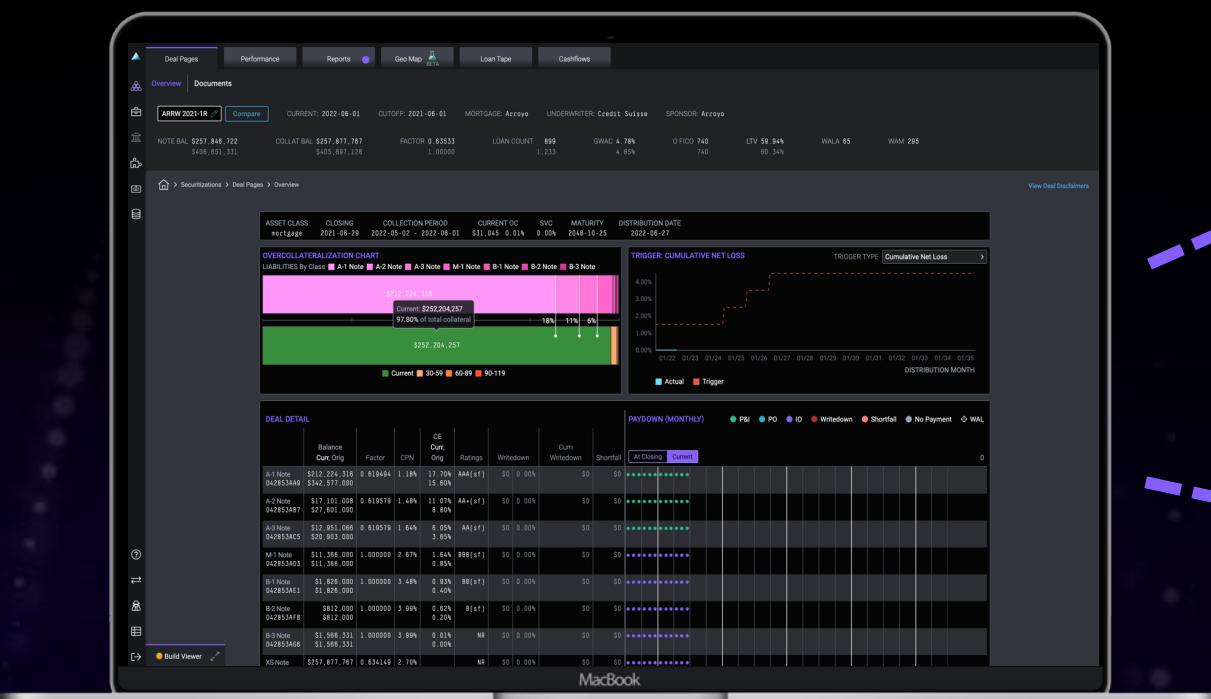


# R microservice <- plumber + docker

- R code triggered by arbitrary external software
- REST API using plumber
- Portable software packaged in docker container



# What is the microservices wild?



# Benefits of Microservice Architecture



**Software engineer builds  
end-to-end system**

**Data scientist builds  
machine learning models**

## Continuous Integration

Build and test R programs

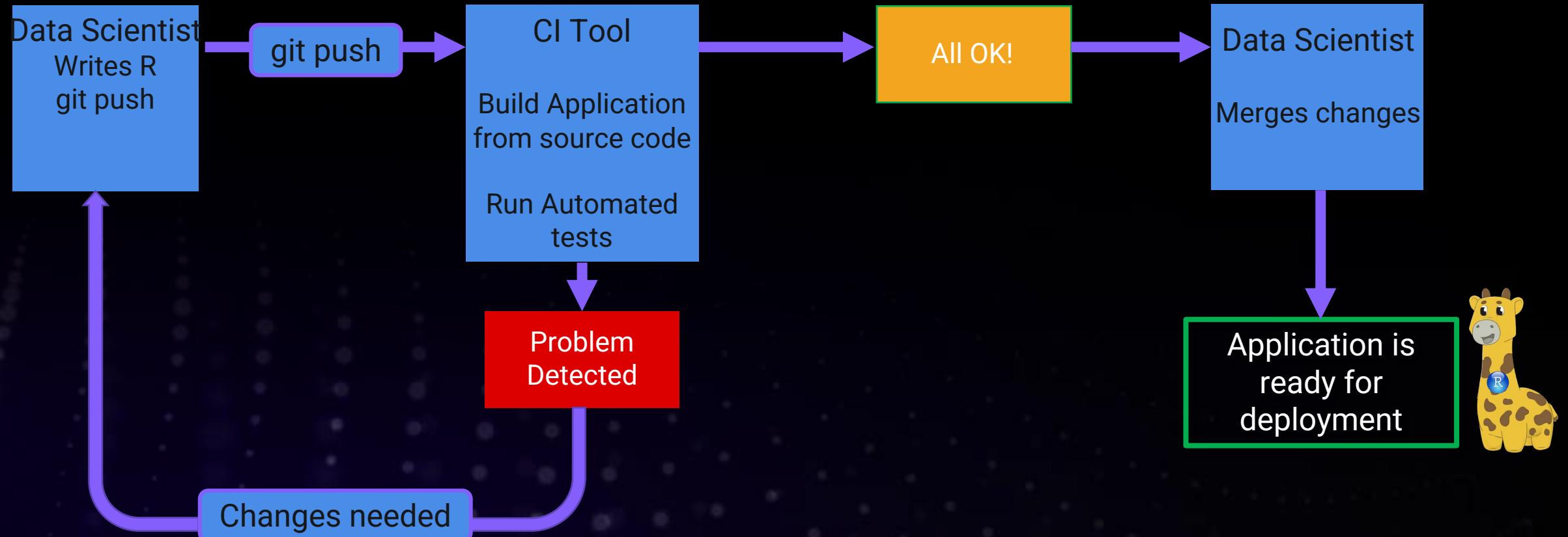
## Continuous Deployment

Release validated software into production

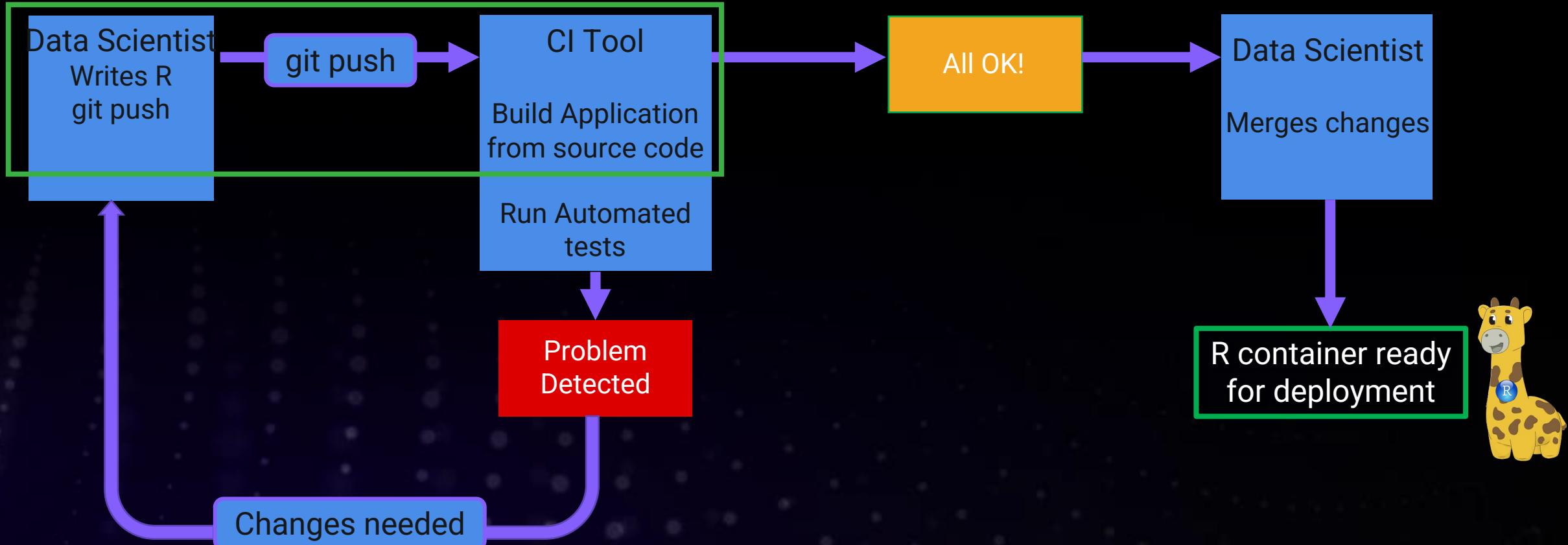
## Cluster Computing

Manage microservice infrastructure in the cloud

# Continuous Integration (“CI”)

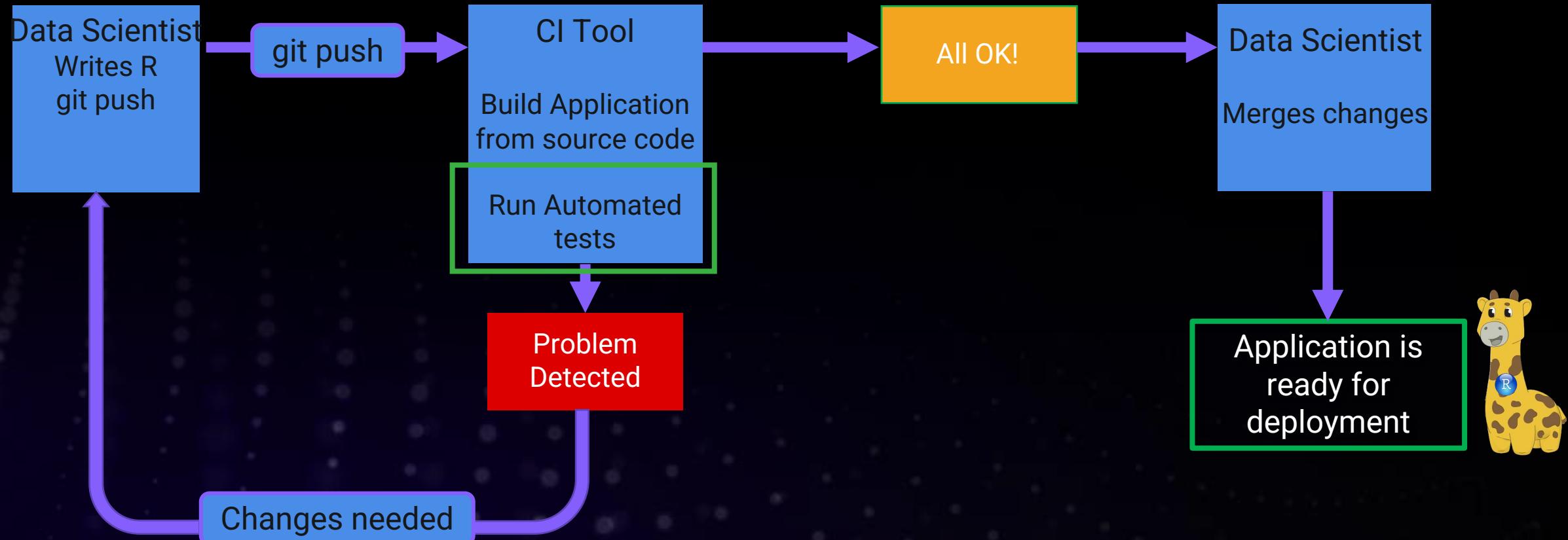


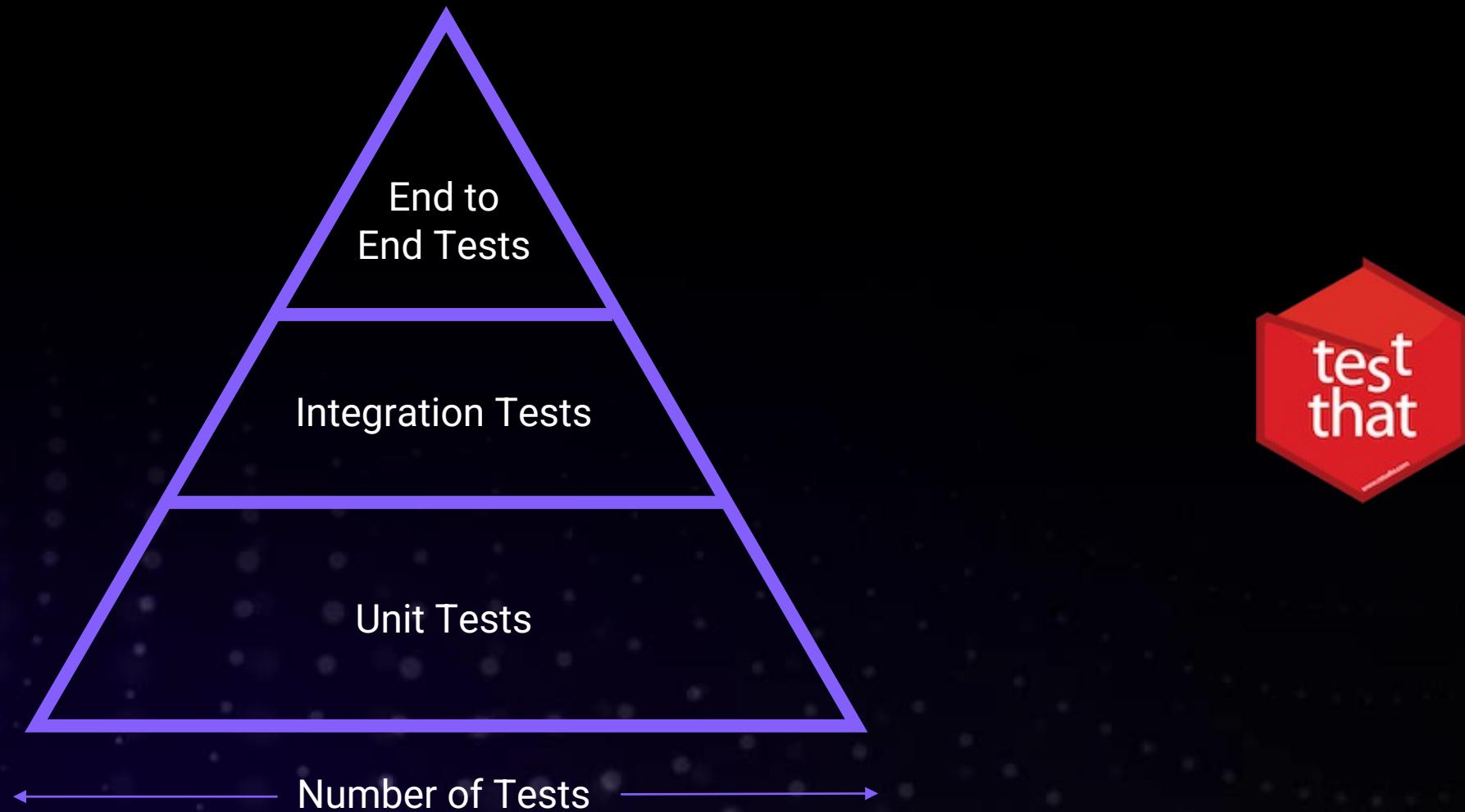
# Continuous Integration (“CI”)

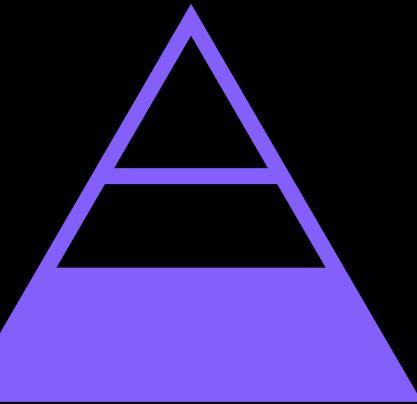




# Continuous Integration (“CI”)





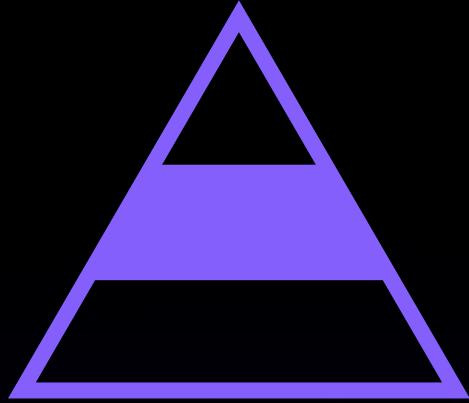


Test individual  
functions and  
lower level code

```
1 library(testthat)
2
3 add_20 <- function(x) x + 20
4
5 test_that("add_20() adds 20", {
6     expect_equal(add_20(10), 30)
7 })
```



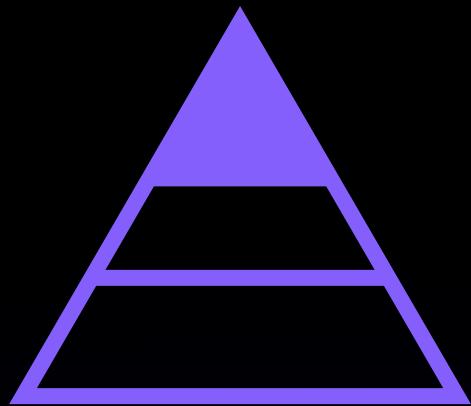
# CI: Integration Tests



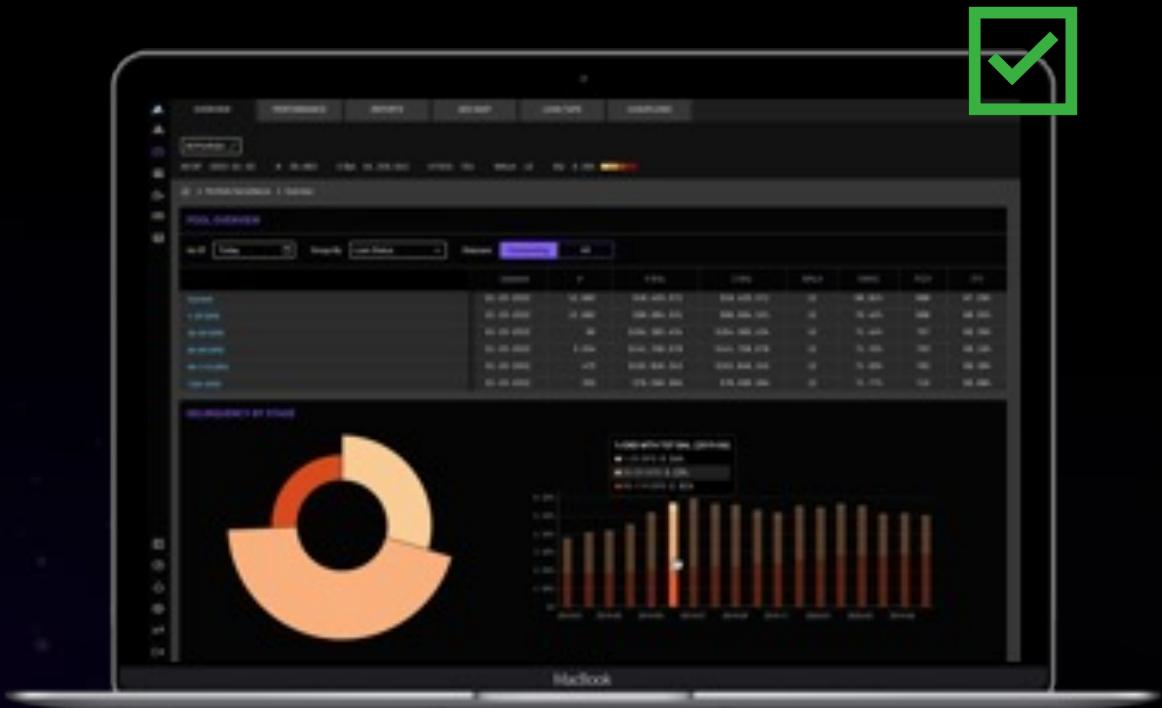
Test R  
container's  
ability to interact  
with other  
services



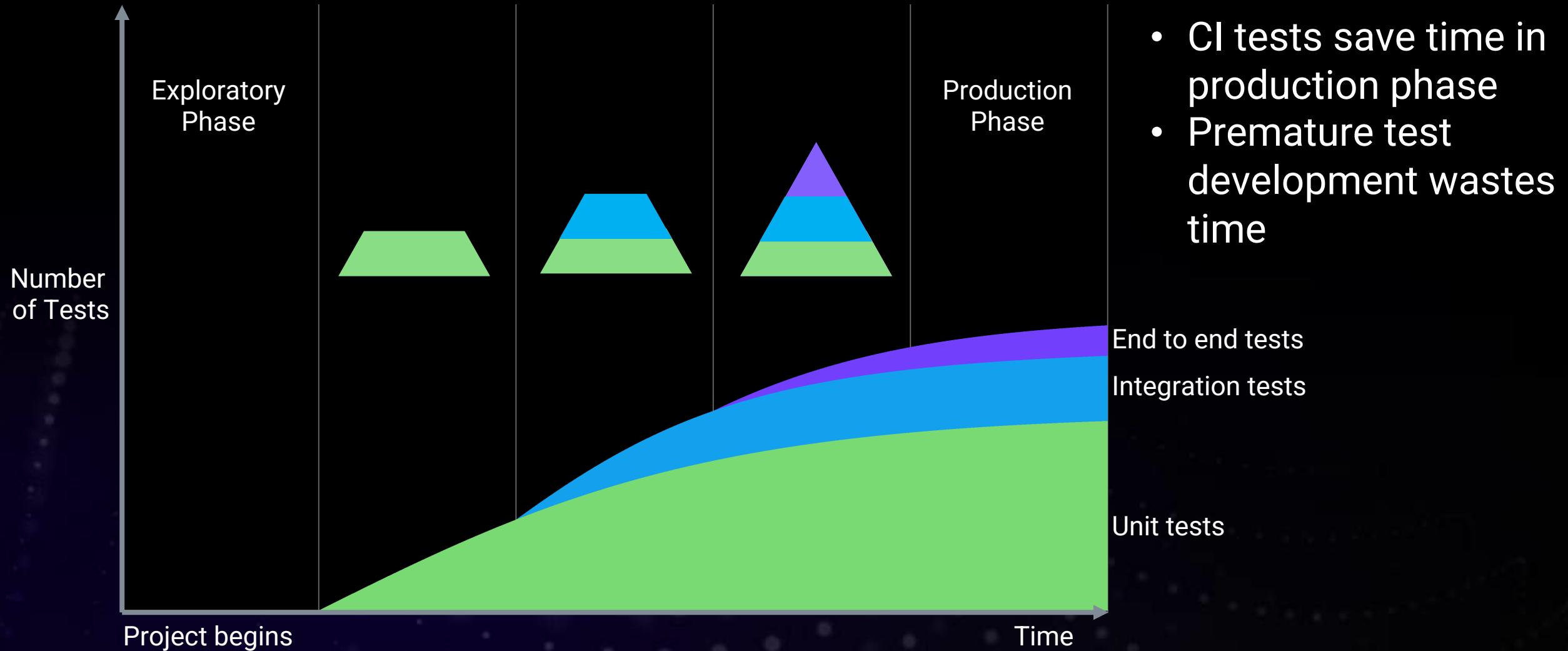
# CI: End-to-End Tests



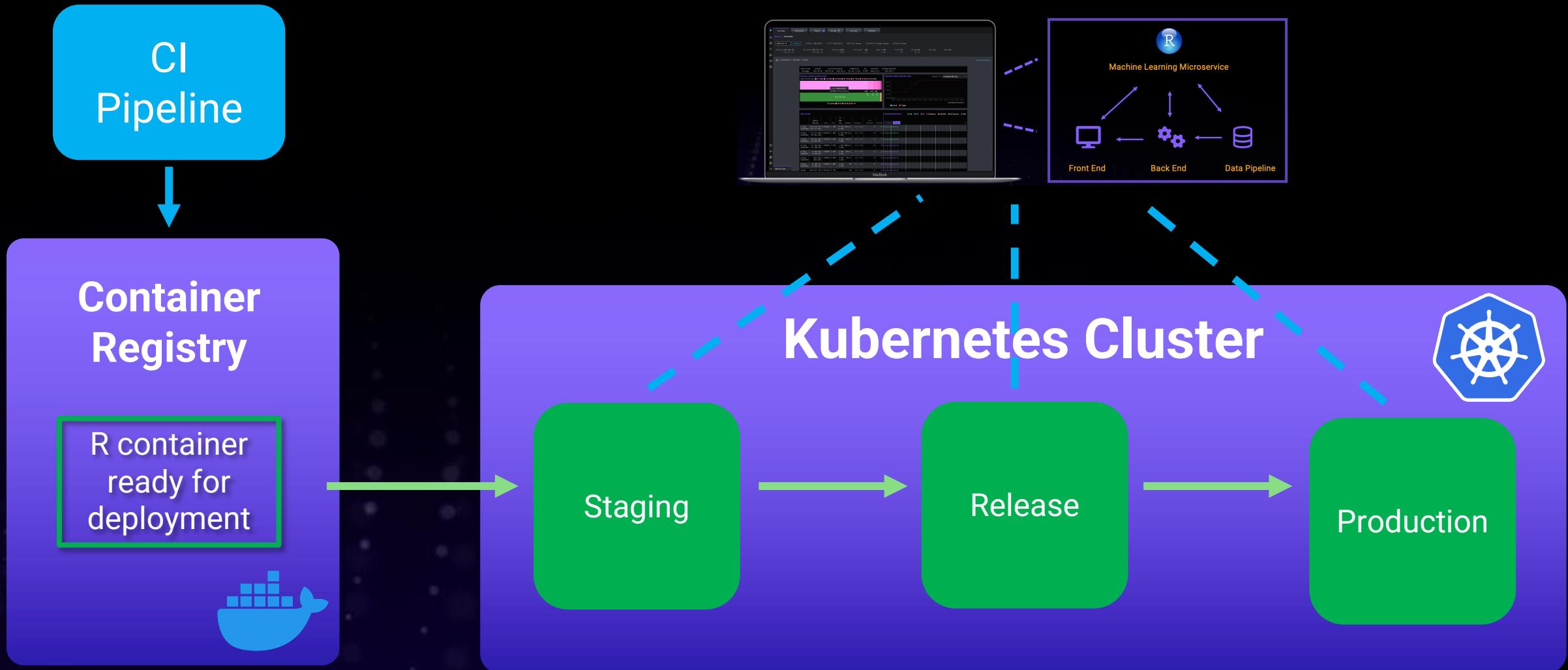
- Full microservice application tested as a whole
- R container tested indirectly



# Write tests at the right time



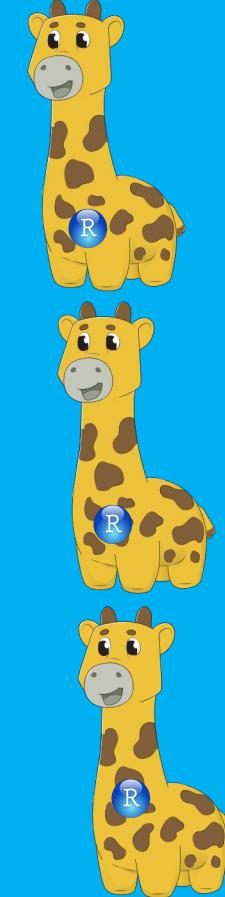
# Continuous Deployment



# R microservices in Kubernetes



Production  
Cluster



# R microservices in Kubernetes



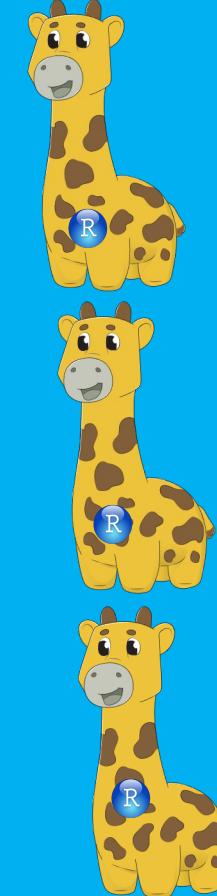
Production  
Cluster



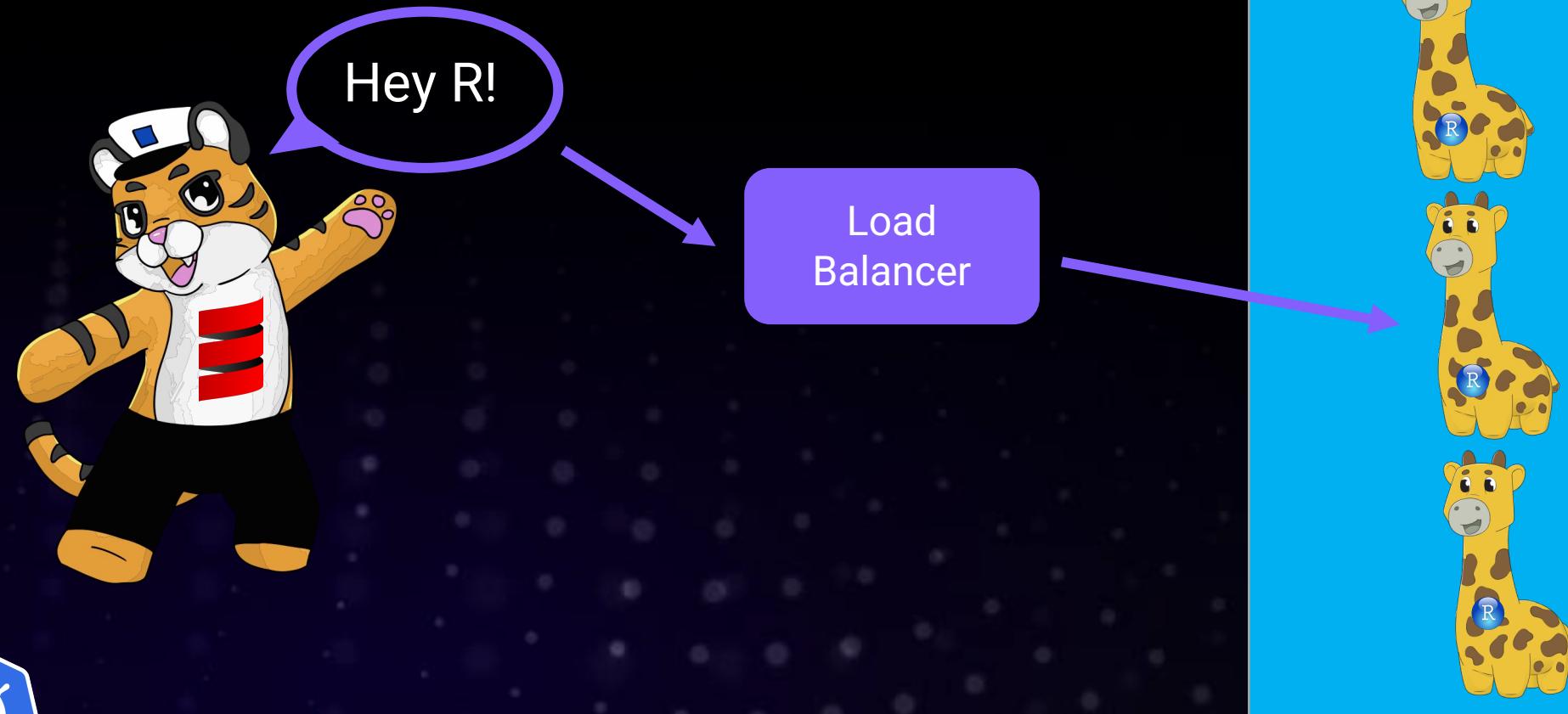
# R microservices in Kubernetes



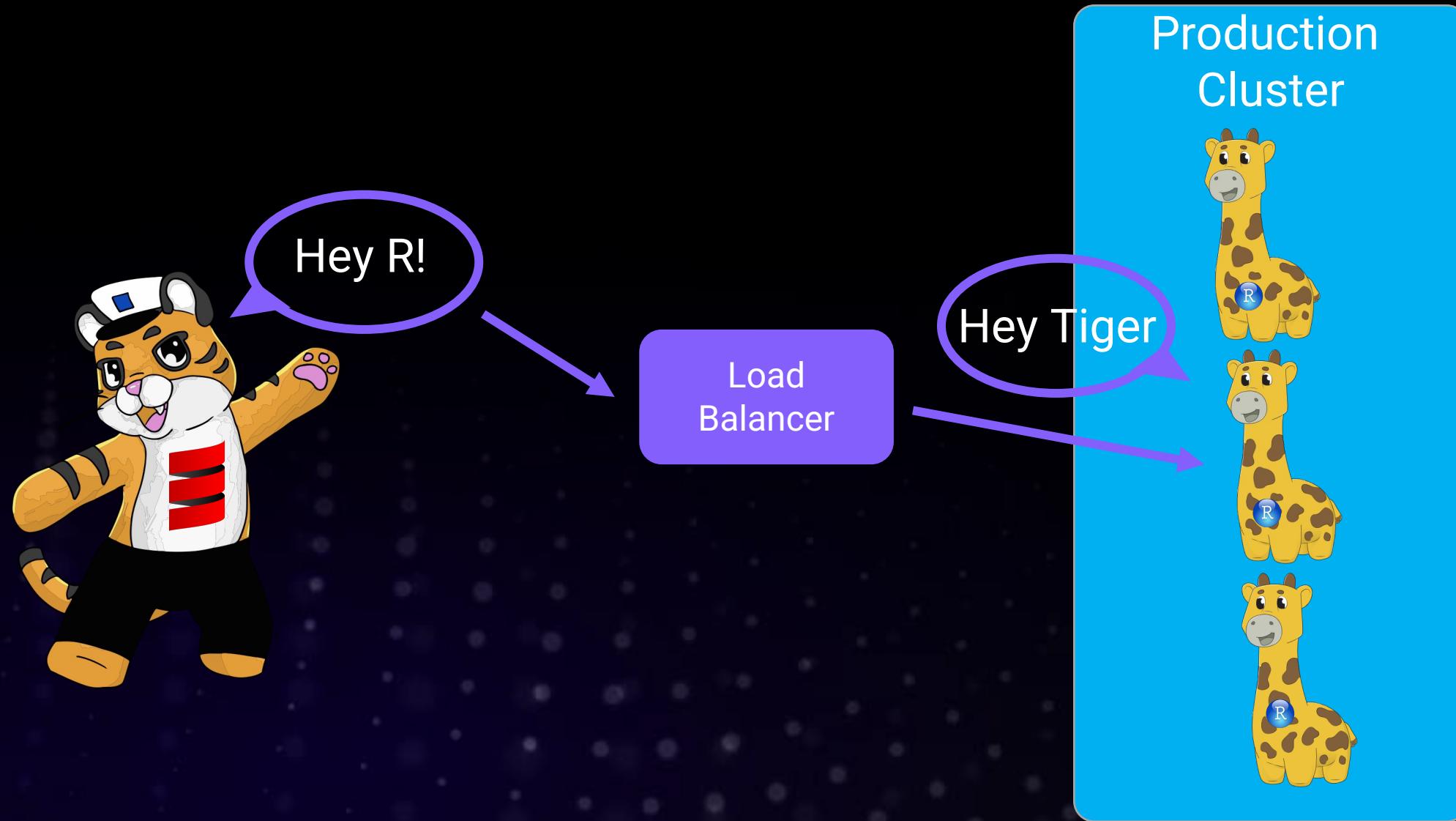
Production  
Cluster



# R microservices in Kubernetes



# R microservices in Kubernetes



# R microservices in Kubernetes



- ReplicaSet of microservice
  - 3 copies always present
  - More copies created when traffic is high
  - If one replica fails, it is immediately replaced with a working copy

Production  
Cluster



# Conclusion



- R is great for production (with CI/CD)
  - **Continuous Integration** ensures R packages are robust and well vetted
  - **Continuous Deployment** automates the release of updates
- R microservices can be built with plumber and docker
- **Plumber** creates REST APIs from R code
- **Docker** creates portable software packages
- **Kubernetes** manages resilient compute cluster

Slides at:





## Questions?

**David Maguire, Sr Data Scientist**



dmaguire@dv01.co



[dv01.co](http://dv01.co)