

# Fast track cloud development with JenkinsX

**Juan Peredo**  
[linkedin.com/in/juanperedotech](https://www.linkedin.com/in/juanperedotech)  
[@JuanPeredoTech](https://twitter.com/JuanPeredoTech)

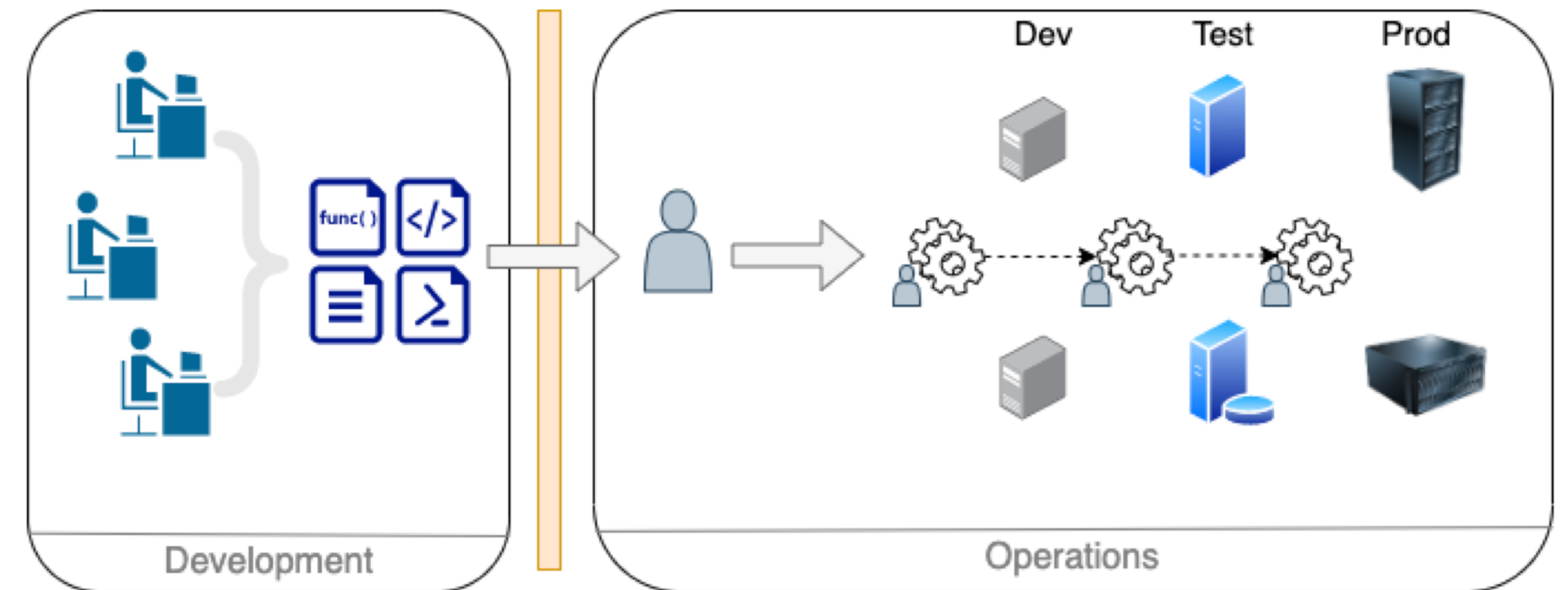
**DEVOPS  
WORLD**  
by CloudBees



# Traditional deployments

## Custom scripts nightmare

- Multiple custom scripts used for deployments
- Multiple teams involved
- Hard to update and debug
- No visibility





---

Process may have  
caused a little  
stress...





# Classic CI/CD

---

- General purpose tools
- Automated
- Standardized pipeline steps
- Better visibility



# Jenkins



# Cloud CI/CD

---

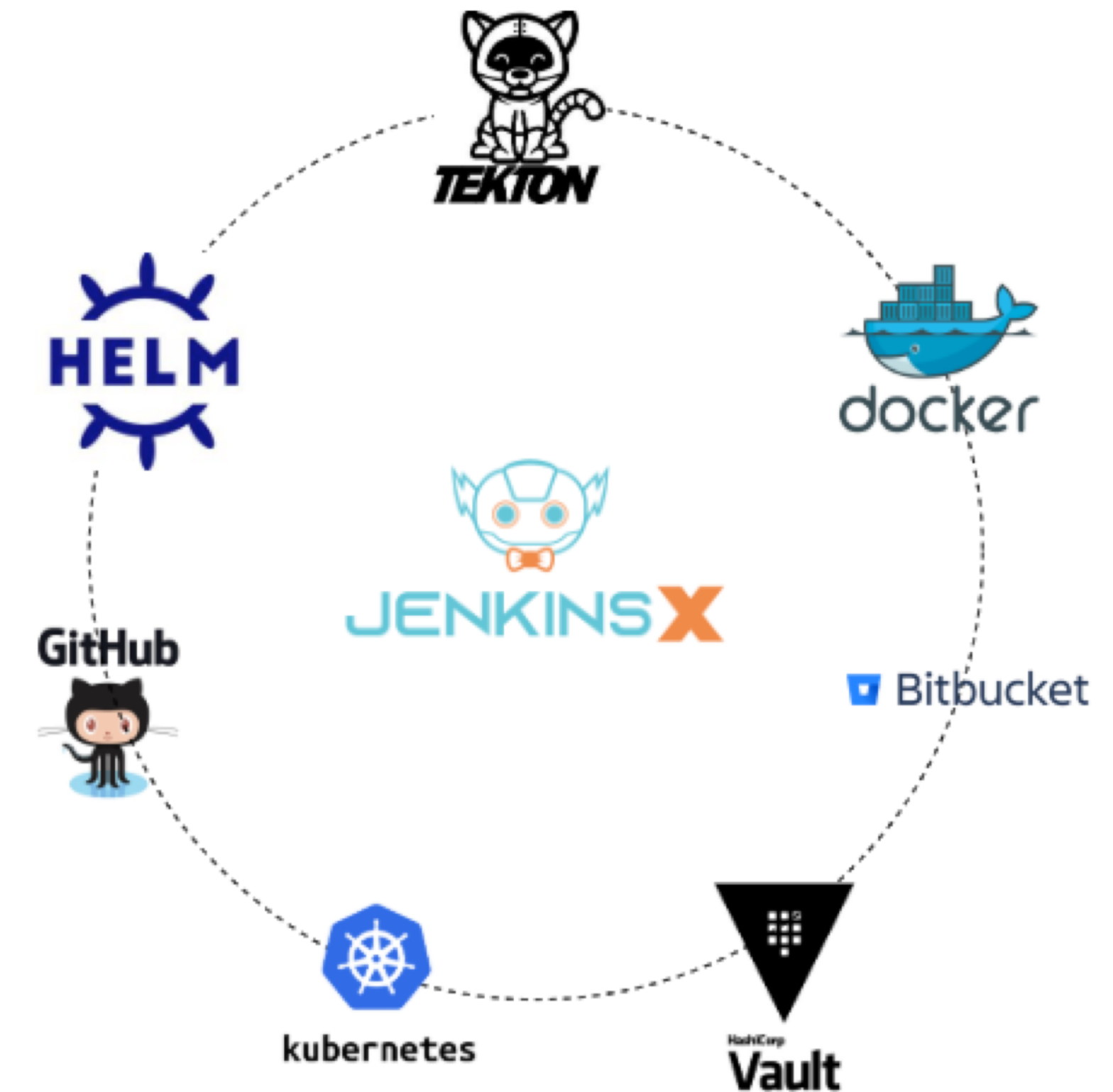
- Automated
- Great traceability and visibility
- Built for re-usability
- Cloud focus





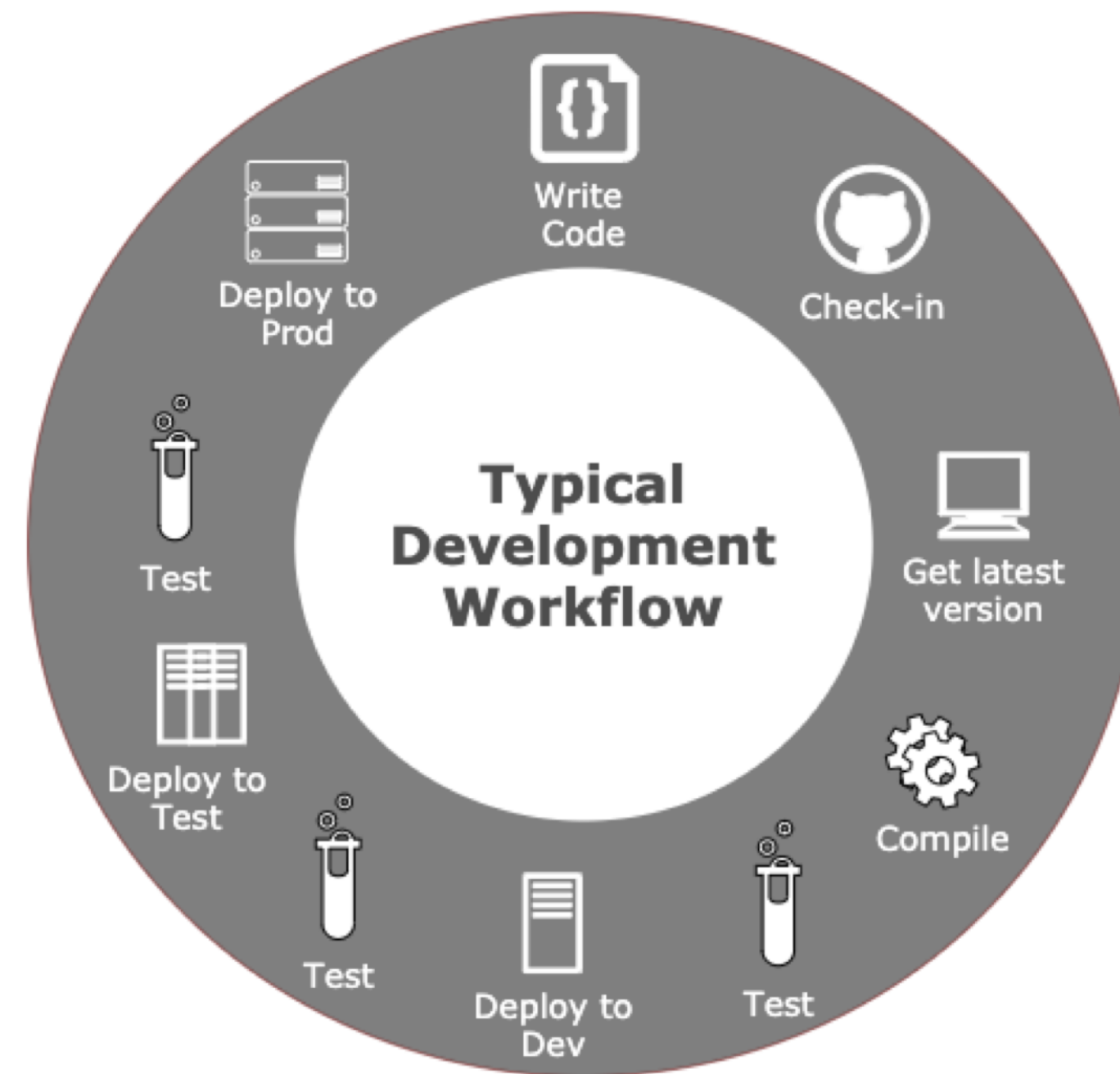
# JX add milk to shopping list...

- Jenkins X is more than just pipelines:
  - Ecosystem to simplify everyday tasks
  - Opinionated (In a good way)
  - Cloud Native
  - Flexible and customizable
  - Combine with Terraform for cluster as code



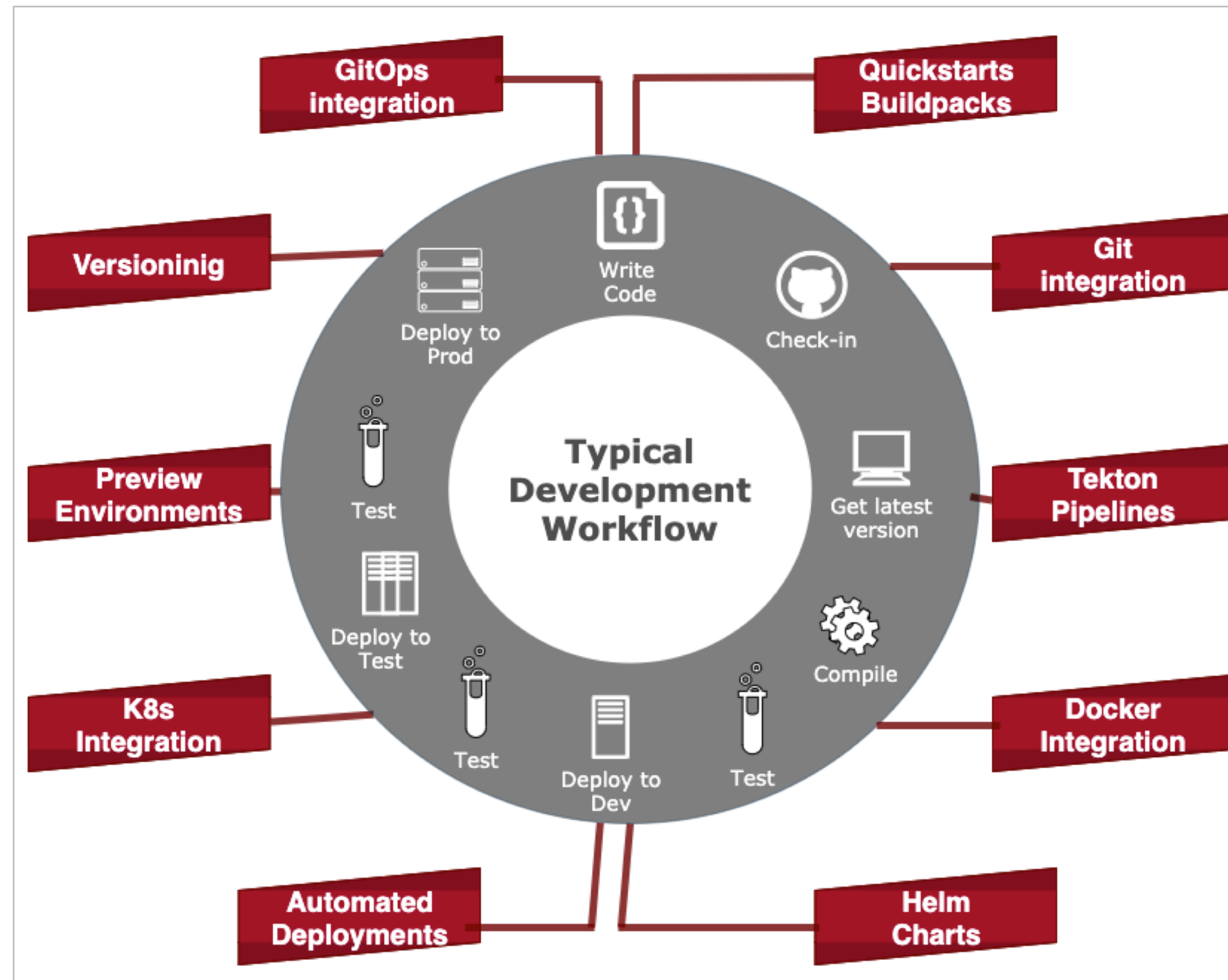


# Classic dev workflow





# Jenkins benefits





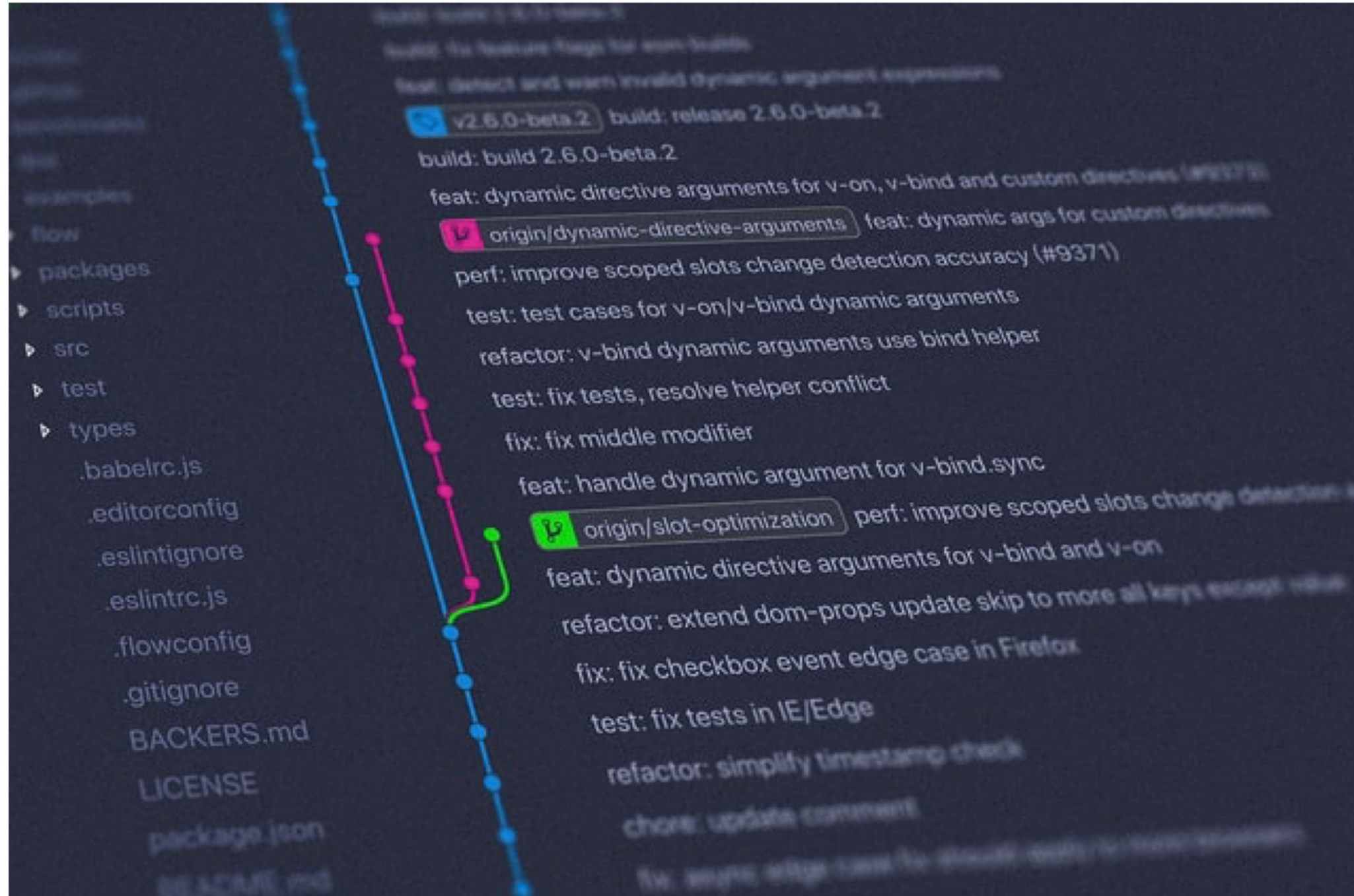
# Setup Jenkins X

Git Provider	GCP	Terraform	Jenkins X	Domain & TLS
<ul style="list-style-type: none"><li>- Create an account (or use existing account)</li></ul>	<ul style="list-style-type: none"><li>- Create new project</li><li>- Setup billing</li><li>- Enable Kubernetes API</li></ul>	<ul style="list-style-type: none"><li>- Install Terraform</li><li>- Create a project folder</li><li>- Create empty main.tf file</li></ul>	<ul style="list-style-type: none"><li>- Install Jenkins X CLI</li></ul>	<ul style="list-style-type: none"><li>- Optional but highly recommended step</li><li>- TLS will secure your internet communications</li></ul>
<ul style="list-style-type: none"><li>- Create organization to hold Jenkins X repos</li></ul>	<ul style="list-style-type: none"><li>- Install Kubectl component</li><li>- Add bucket to hold Terraform state</li></ul>	<ul style="list-style-type: none"><li>- Get jx gcp module from terraform registry</li><li>- Use module instructions to update main.tf</li></ul>	<ul style="list-style-type: none"><li>- Install Helm v2</li><li>- Clone the JX Boot repo</li></ul>	<ul style="list-style-type: none"><li>- Obtain top level domain from internet registrar</li><li>- Configure cloud provider to use the domain</li></ul>
<ul style="list-style-type: none"><li>- Create account to perform Jenkins X actions ("bot")</li></ul>	<ul style="list-style-type: none"><li>- Set default user credentials</li><li>- Add storage Admin role to user</li></ul>	<ul style="list-style-type: none"><li>- Initialize Terraform</li><li>- Generate terraform requirements.yml file</li><li>- Create the Cluster</li></ul>	<ul style="list-style-type: none"><li>- Copy requirements.yml to JX Boot repo</li><li>- Run JX Boot to configure Jenkins X</li></ul>	<ul style="list-style-type: none"><li>- Configure Terraform to use the domain</li><li>- Configure Terraform to use the certificates</li></ul>

\* Illustrative, refer to official install procedures for full install instructions. Setup on Mac OS, using GCP as cloud provider



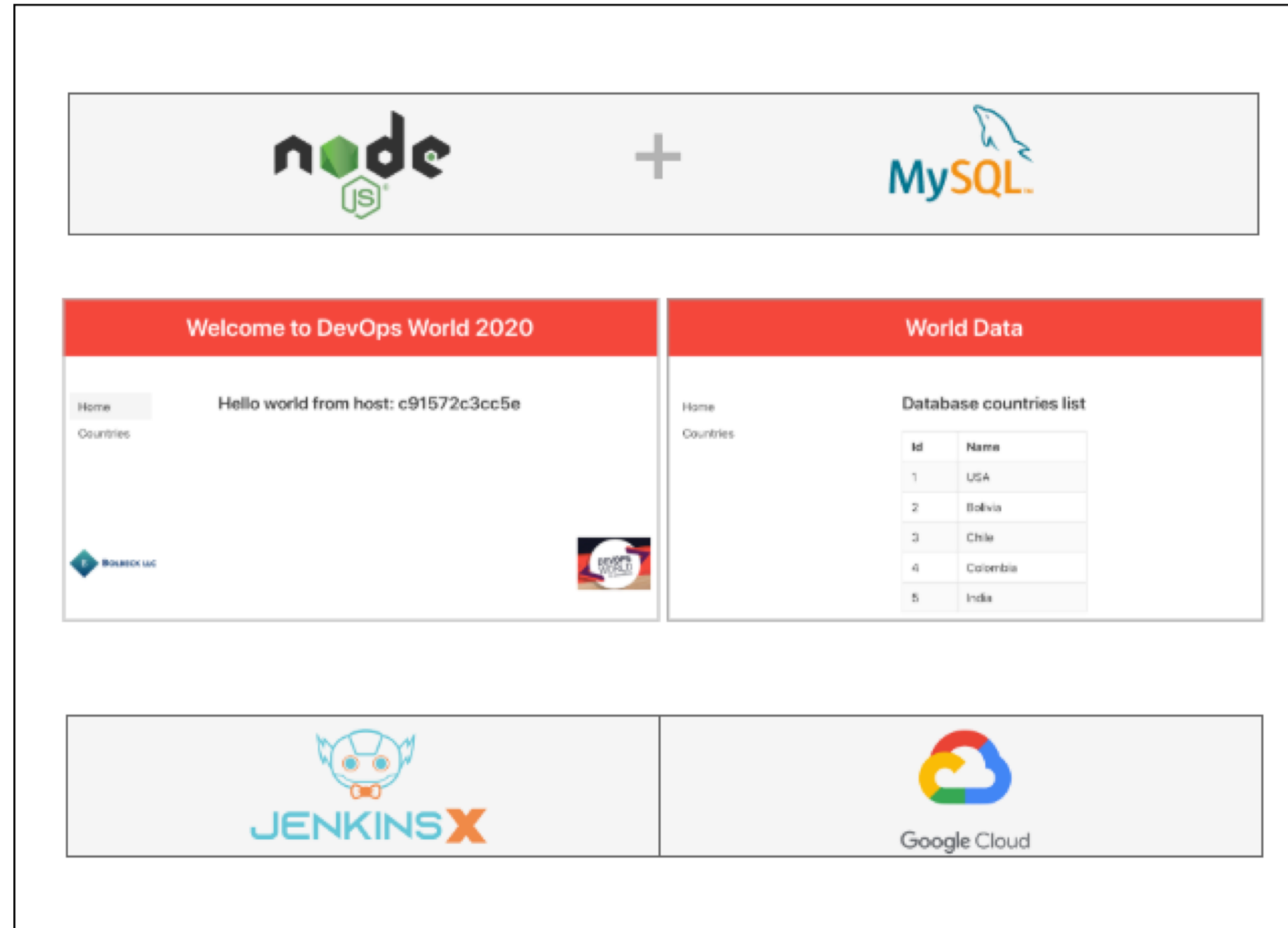
# Demo



- **Goals:**
  - Zero to Production
  - Automate every day tasks
  - Everything as code



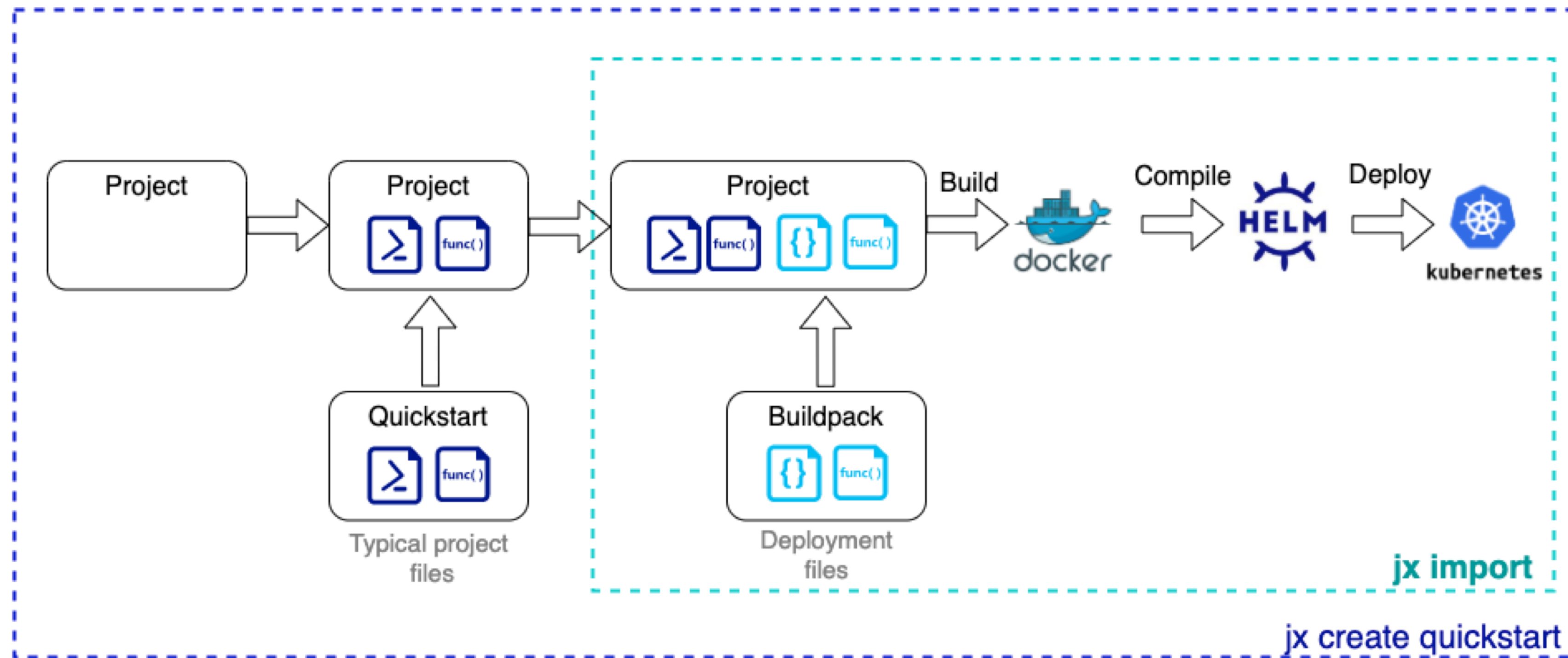
# Our application





# Initialize project

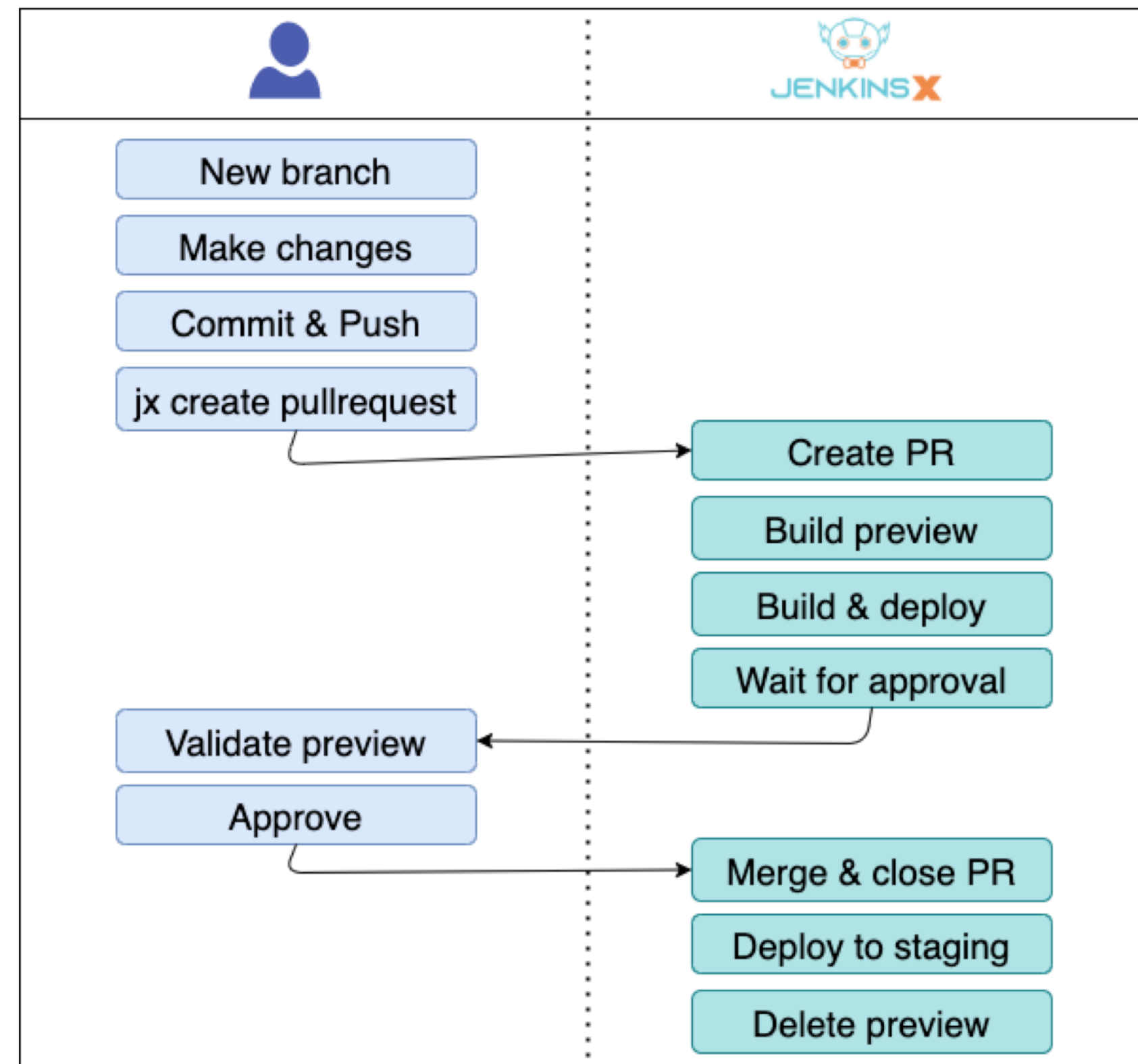
jx create and jx import





# Change Management

jx create pullrequest





# Deployment management










---

`jx promote`

- Deploy manually to prod by default
- Promote a version of app to any environment with `jx promote`
- Rollback by promoting previous version
- Can be configured to do different types of deployments



# Summary

App Code			
     			
Initial application			
Init project & first deploy	jx import	Create / update repo	 Dev Stage
Pull request	jx create pullrequest	<ul style="list-style-type: none"><li>• Create PR</li><li>• Run pipeline</li><li>• Deploy Preview</li><li>• Wait for approval</li><li>• Merge PR</li><li>• Delete Preview</li></ul>	 Dev Stage Preview
Push to production	jx promote	<ul style="list-style-type: none"><li>• Create &amp; close PR</li><li>• Create release tag</li></ul>	 Dev Stage Prod



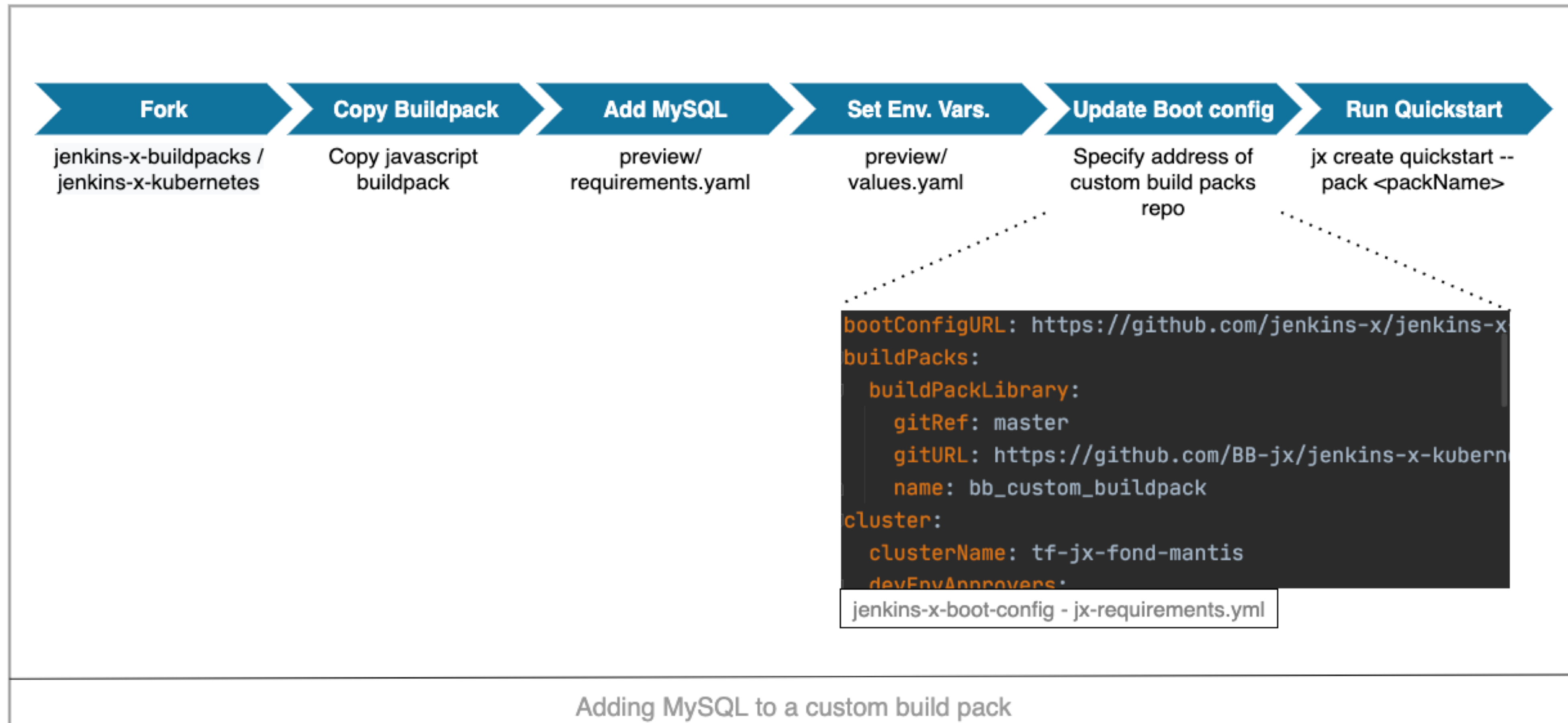
# Inherited pipelines

our-project/jenkins-x.yml

```
jenkins-x.yml
1 buildPack: javascript
2
pipeline.yaml
1 extends:
2   import: classic
3   file: javascript/pipeline.yaml
4 pipelines:
5   pullRequest:
6     build:
7       steps:
8       - sh: export VERSION=$PREVIEW_VERSION && skaffold build -f ska
9         name: container-build
10    postBuild:
11      steps:
12      - sh: dx step post build --image $DOCKER_REGISTRY/$ORG/$APP NA
```

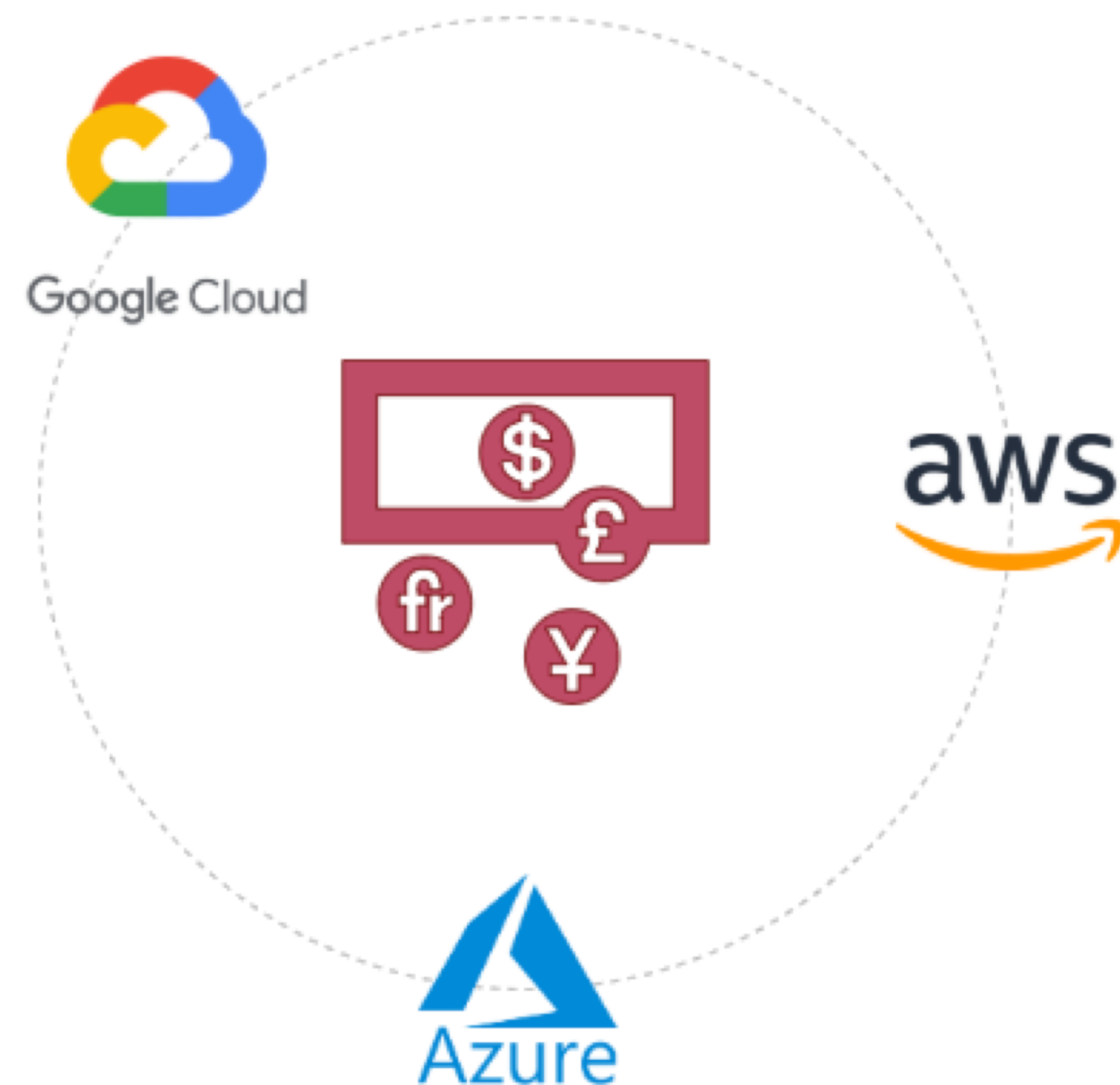
jenkins-x.kubernetes/javascript//pipeline.yaml

# Custom build packs





# Managing your cloud cost



Cloud deployments are priced per resource usage, time and storage

- Removing an application from Jenkins X

`jx delete application`

- Removing nodes from cluster

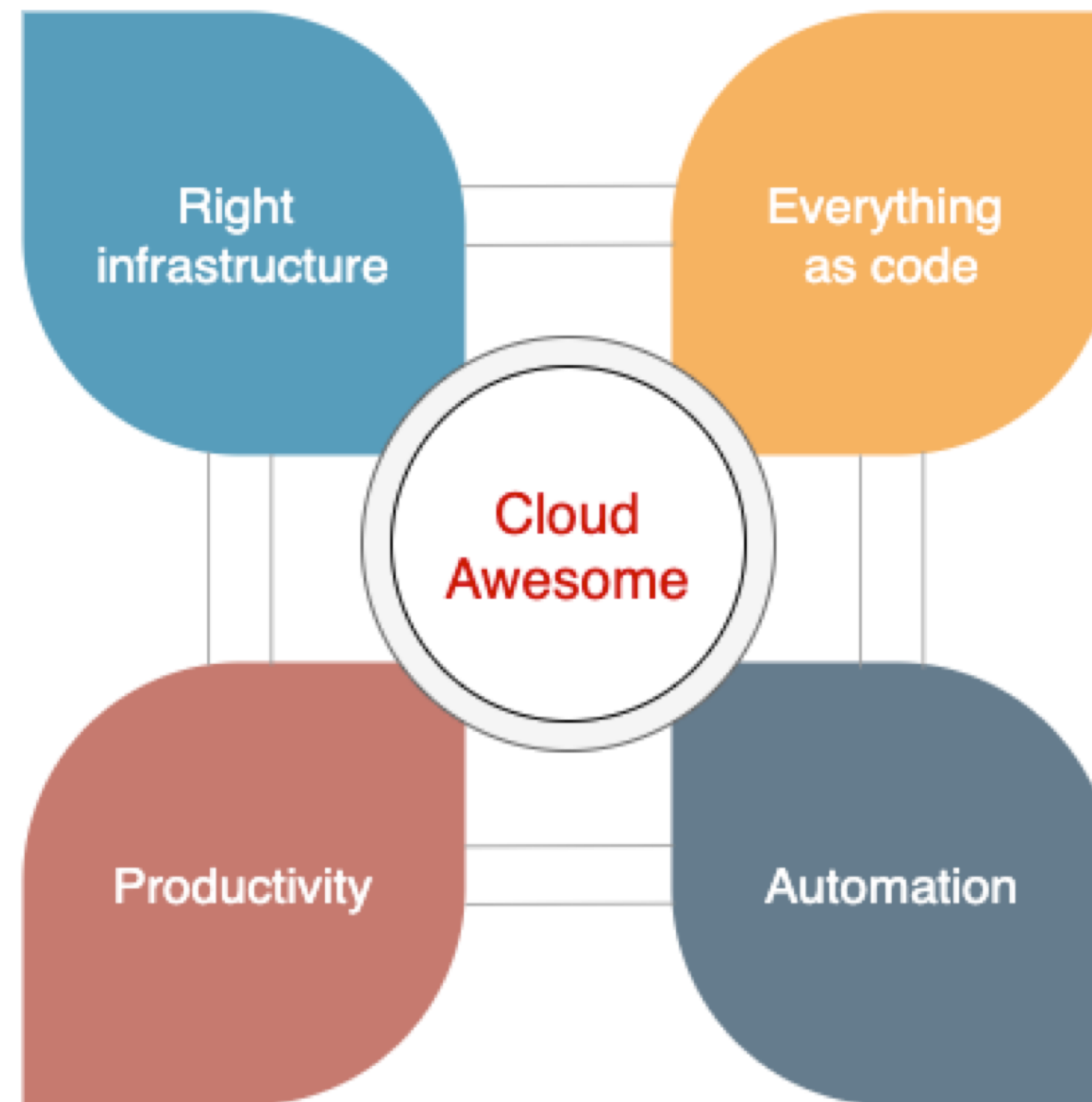
`gcloud container clusters resize`

- Destroying the cluster

`Terraform destroy`

# Key take aways

---





# Questions ?

---



Thanks to CloudBees & all the conference sponsors



# Fast track cloud development with JenkinsX

**Juan Peredo**  
[linkedin.com/in/juanperedotech](https://www.linkedin.com/in/juanperedotech)  
[@JuanPeredoTech](https://twitter.com/JuanPeredoTech)

**DEVOPS  
WORLD**  
by CloudBees






---

# Appendix

# Photos

---

-  Photo by [Andre Guerra](#) on [Unsplash](#)
-  Photo by [JESHOOOTS.COM](#) on [Unsplash](#)
-  Photo by [Yancy Min](#) on [Unsplash](#)