

Brigade

Scripting container workflows on Kubernetes

Radu Matei
@matei_radu
Software Engineer, Microsoft Azure





Docs: <https://docs.brigade.sh>

GitHub: <https://github.com/brigadecore>



What is an OS?

An **operating system** is the program that, after being initially loaded into the computer, manages all the other programs.



Kubernetes as a “cloud OS”

An operating system manages
processes on a machine.

Kubernetes manages **containers on a cluster.**



OS shell scripting

Flow control that wraps the execution of **processes**, but is not opinionated about what the **processes** do, or how they run.



OS shell scripting

```
#!/bin/bash

for i in $( ls -1 ); do
    if [[ -d $i ]]; then
        echo "$i is a directory"
    fi
done
```



Cluster shell scripting?

Flow control that wraps the execution of **containers**, but is not opinionated about what the **containers** do, or how they run.



Cluster shell scripting?

```
for i in $( run_some_container ); do
  if $( run_some_other_container );
then
  $( run_yet_another_container )
fi
done
```





CLOUD NATIVE
SANDBOX



What is Brigade?

- framework for event-driven scripting on Kubernetes
- extremely lightweight, Kubernetes native
- chain together containers to create workflows



Cluster shell scripting

```
for i in $( run_some_container ); do
  if $( run_some_other_container );
then
  $( run_yet_another_container )
fi
done
```



Cluster shell scripting

- what programming language?
- how to share data between containers?
- when should we execute scripts?

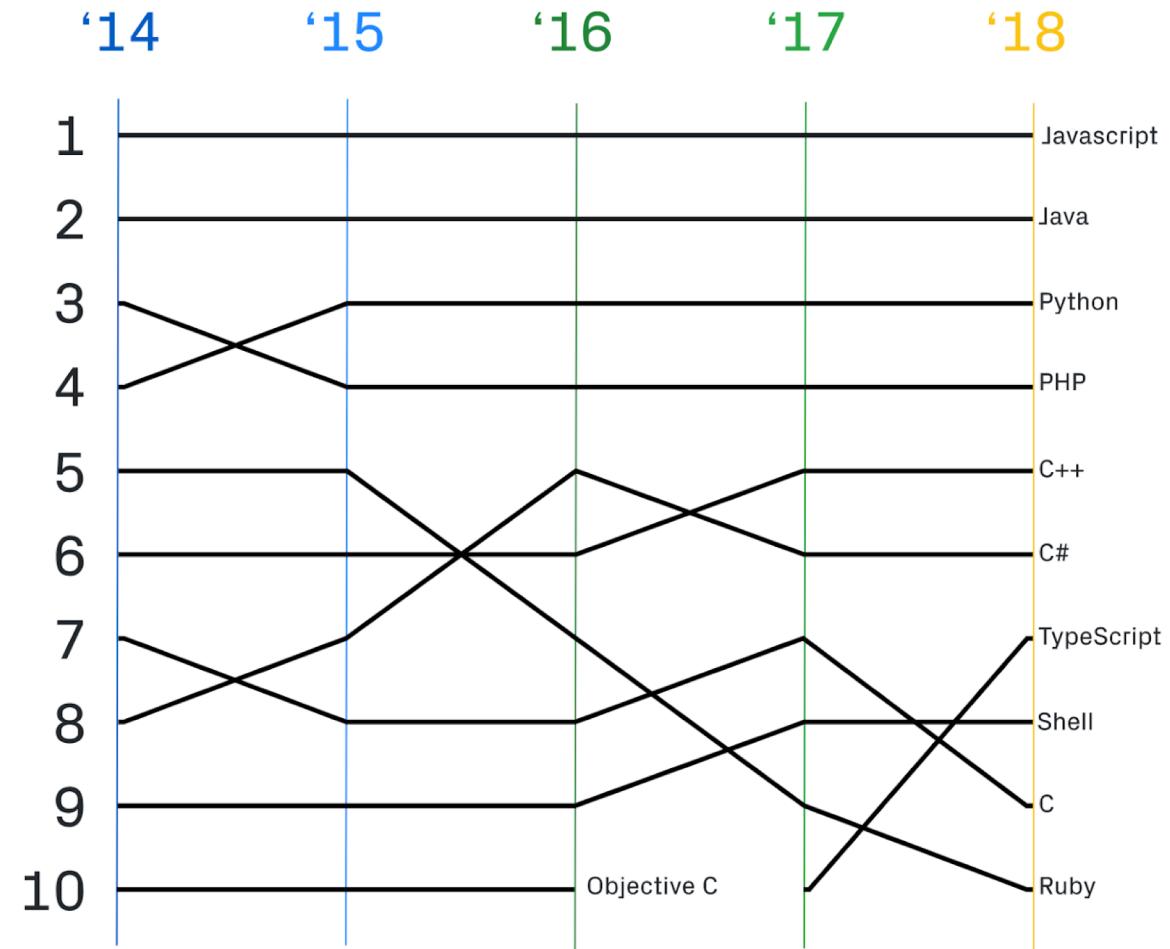


Choose a language, don't write one

Top languages over time

You're coding on GitHub in hundreds of programming languages, but JavaScript still has the most contributors in public and private repositories, organizations of all sizes, and every region of the world.

This year, TypeScript shot up to #7 among top languages used on the platform overall, after making its way in the top 10 for the first time last year. TypeScript is now in the top 10 most used languages across all regions GitHub contributors come from—and across private, public, and open source repositories. *



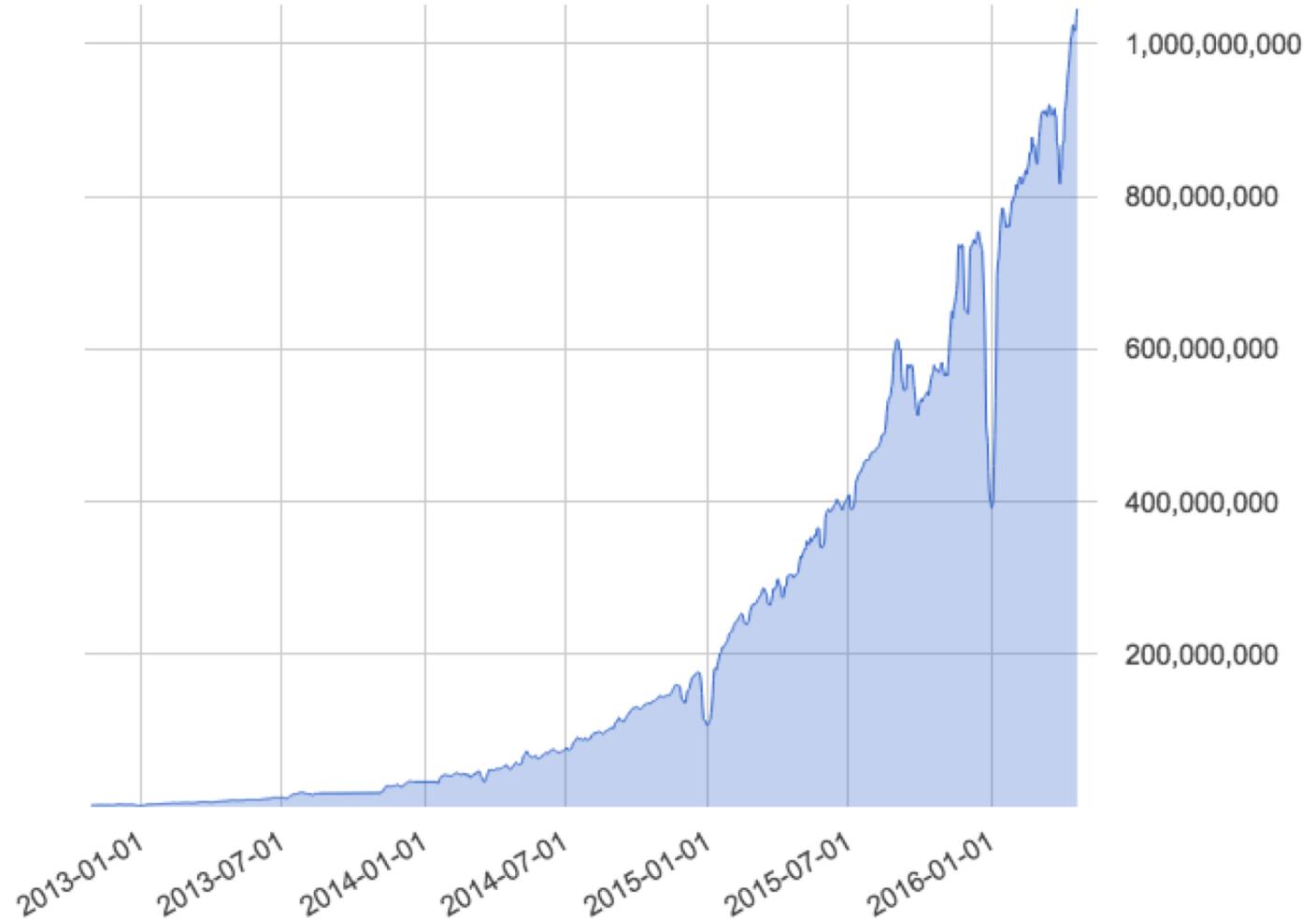
Why JavaScript?

- most popular programming language
- rich ecosystem of tools
- extremely flexible



1 Billion weekly NPM package pulls

Rolling weekly downloads of npm packages



How to share data between containers?

- pass information *TO* the containers (commands, environment variables, files)
- get information *FROM* the containers (STDOUT, exit codes, files)



How to share data between containers?

- JavaScript native
- preserve the existing paradigms for passing data to and from containers



It's just JavaScript

```
var program = new Job("one", "alpine:3.5")
program.tasks = [ "echo hello", "echo goodbye"]
program.env = { "KEY": "value" }

program.run()
```



When do we execute scripts?

- reacting to triggers / events
- event handlers become the entry points for Brigade scripts



Where do events come from?

- HTTP webhooks (internal, 3rd party)
- Git events, container registry events
- Kubernetes events
- any event source can be configured for Brigade - gateways



Reacting to events with Brigade

```
const { events, Job } = require("brigadier");

events.on("someEvent", ( eventData ) => {
  var program = Job("one", "alpine:3.5");
  program.tasks = [ "echo hello", "echo goodbye"];
  program.env = { "KEY": "value" };

  program.run();
}) ;
```



Use cases

- foundation for opinionated CI/CD systems
- application security scanning
- aggregating + analyzing data from multiple systems and building reports
- creating preview environments on Kubernetes for pull requests



Use cases

- processing orders, connecting to external services
- actual CI/CD, integration with GitHub Checks API, BitBucket, GitLab
- any potential container workflow on Kubernetes that would benefit from actual language features (as opposed to being constrained by YAML)



Getting started with Brigade

```
$ helm repo add brigade  
https://brigadecore.github.io/charts
```

```
$ helm install brigade brigade/brigade
```

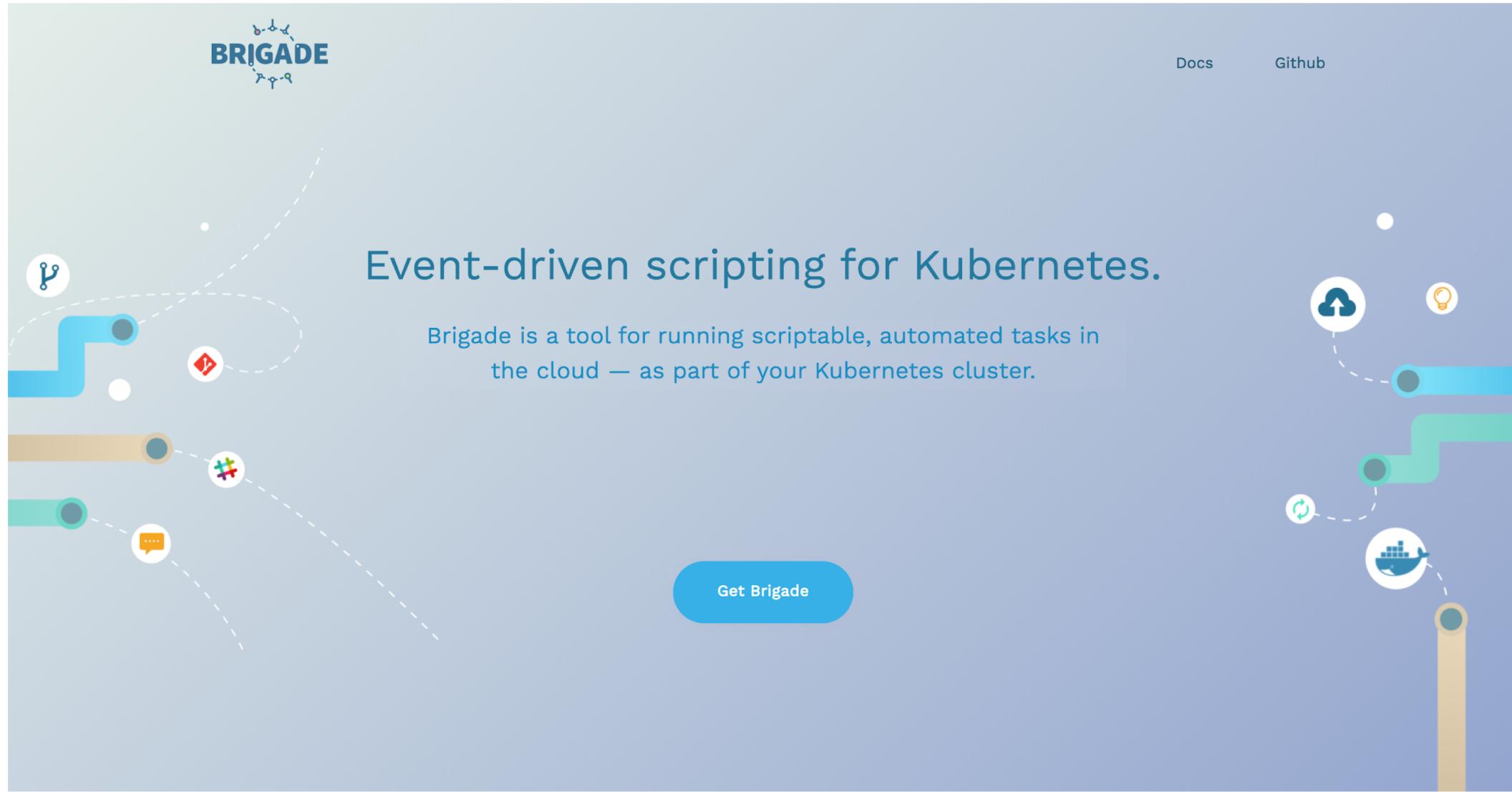
```
$ brig check
```



Demo

Getting started with Brigade

docs.brigade.sh



The image shows the homepage of the Brigade documentation site at docs.brigade.sh. The header features the Brigade logo with the word "BRIGADE" in blue and orange. Navigation links for "Docs" and "Github" are in the top right. The main visual is a light blue background with a white dashed line forming a path. Along this path are several colored icons: a blue gear, a red diamond, a green plus sign, an orange speech bubble, a yellow lightbulb, a green circular arrow, a white cloud with an upward arrow, and a blue container ship. Centered on the page is the text "Event-driven scripting for Kubernetes." Below it is a description: "Brigade is a tool for running scriptable, automated tasks in the cloud — as part of your Kubernetes cluster." A prominent blue button at the bottom center says "Get Brigade". In the top right corner of the slide, there is a small Cloud Native Computing Foundation logo.

Event-driven scripting for Kubernetes.

Brigade is a tool for running scriptable, automated tasks in the cloud — as part of your Kubernetes cluster.

[Get Brigade](#)



Contribute - #good-first-issue

The screenshot shows the GitHub repository page for `brigadecore / brigade`. The repository has 93 issues, 10 pull requests, 1 action, 1 project, and 1.8k stars. It features 5 branches, 29 releases, 1 environment, and 70 contributors under Apache-2.0 license. The commit history lists 1,413 commits by various authors, with the latest commit being `bbfd9bd` 6 days ago.

Author	Commit Message	Date
radu-matei	Move brig README to docs page (#992)	6 days ago
	chore(*): updates per org change to brigadecore	7 months ago
	Move brig README to docs page (#992)	6 days ago
	chore(*): updates per v1.2.1 release (#990)	11 days ago
	Cleanup: fix some typos in code comment (#978)	2 months ago
	use newer go image	5 months ago
	use newer go image	5 months ago
	Cleanup: fix some typos in code comment (#978)	2 months ago
	chore(*): updates per v1.2.1 release (#990)	11 days ago
	Move brig README to docs page (#992)	6 days ago
	e2e first commit (#950)	3 months ago
	filter docker context for each image build	5 months ago
	emphasizing that job timeout is in milliseconds (#925)	5 months ago
	fix(*): ensure all log lines to be fetched regardless of kubelet impl...	12 days ago
	improve the build and release processes	5 months ago
	chore(*): update image org to brigadecore	7 months ago



Get involved!

- #brigade channel in the Kubernetes Slack
- bi-weekly community meetings, Tuesdays, 5PM PST
- github.com/brigadecore
- join us at KubeCon (intro + deep dive)



TL; DR

- Brigade is a cluster scripting environment that allows you to chain multiple containers and create workflows and pipelines
- lightweight, and works on **any Kubernetes cluster**
- write scripts in basic JavaScript
- Brigade is a Kubernetes application - you can manage and monitor it the same way as any other application
- Brigade is a CNCF Sandbox Project, stable at v1.2



FAQ – Where do I learn about X?

- advanced language features guide (async/await, try/catch) -
https://docs.brigade.sh/topics/scripting_advanced/
- gateways - <https://docs.brigade.sh/topics/gateways/>
- security - <https://docs.brigade.sh/topics/security/>
- how to package up dependencies -
<https://docs.brigade.sh/topics/dependencies/>
- testing Brigade scripts - <https://docs.brigade.sh/topics/testing/>



FAQ – Declarative?

- Brigade is a cluster scripting environment - so a real programming language is required
- but... it is easy to create pipelines with Brigade
- there are already multiple implementations of declarative support on top of Brigade (at least 3 that we know of)



FAQ – Windows containers?

[WIP] Experimental Windows containers support #838

Draft

radu-matei wants to merge 1 commit into `brigadecore:master` from `radu-matei:windows-entrypoint` 

Conversation 0

Commits 1

Checks 2

Files changed 1



radu-matei on 8 Mar

Member



...

What this PR does / why we need it:

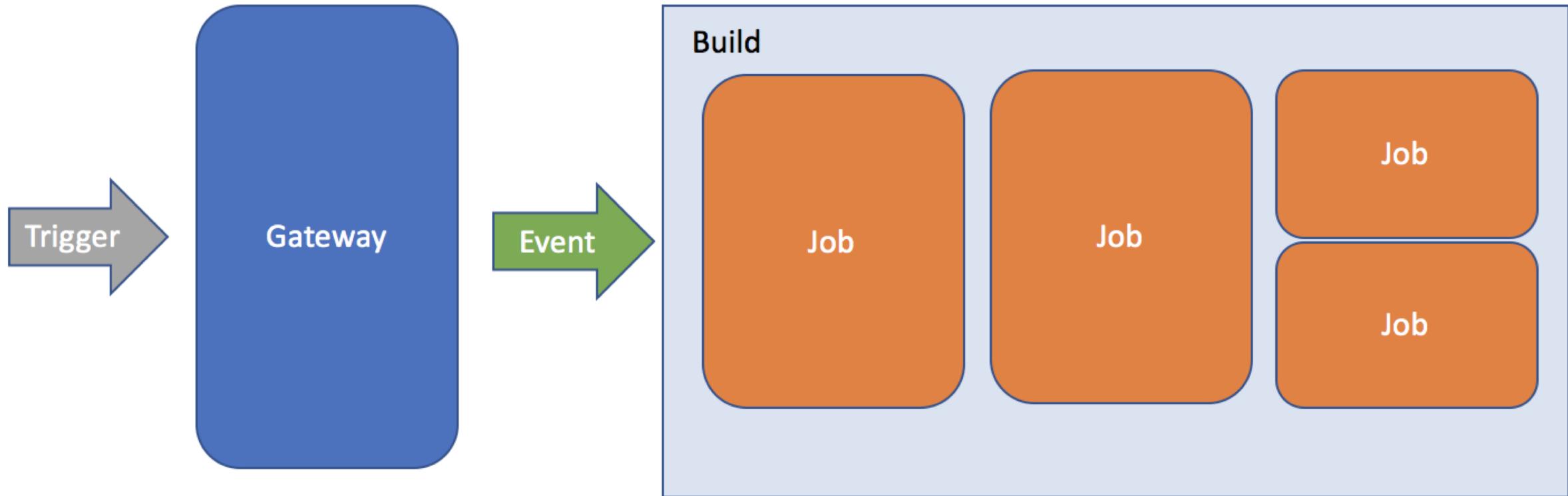
This PR adds very early experimental Windows Server containers support.

It has been tested with a Virtual Kubelet node backed by Azure Container Instances, so the following are current known limitations:

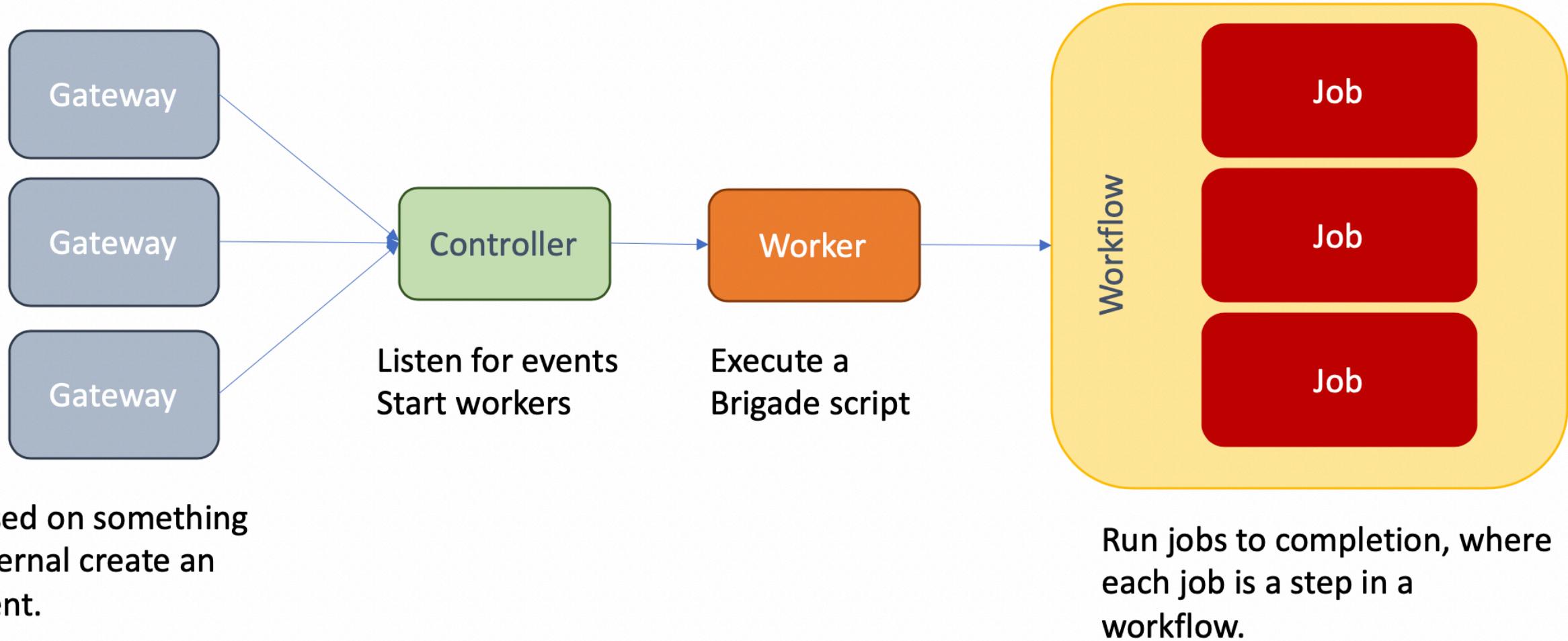
- job and build caches are not supported (limitation of volumes in ACI)
- the Git sidecar is not supported (limitation of volumes in ACI)
- the entrypoint is passed directly to the command arguments, not constructed as a script (limitation of volumes in ACI)



FAQ – how does Brigade work?



FAQ – how does Brigade work?



FAQ – how does Brigade work?

