

Wasm Whirlwind

Bailey Hayes, Cosmonic



Introduction

Bailey Hayes

- Director at Cosmonic
- wasmCloud maintainer
- Wasm enthusiast


`asm.js`


2012



2015



 @baihay

 @ricochetcode

Neither Web, nor Assembly



Open W3C Standard

Open and widely supported standard



Safe & Secure

Deny by default secure sandbox, featuring capability driven permissions



Efficient and fast

Small size and near-native execution speed



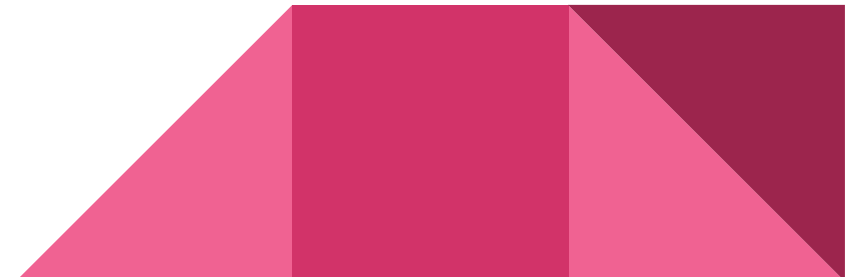
Polyglot

Choice of deployment language means ability to reuse existing libraries



Portable

WebAssembly runs in all major browsers



Neither Web, nor Assembly



Open W3C Standard

Open and widely supported standard



Safe & Secure

Deny by default secure sandbox, featuring capability driven permissions



Efficient and fast

Small size and near-native execution speed



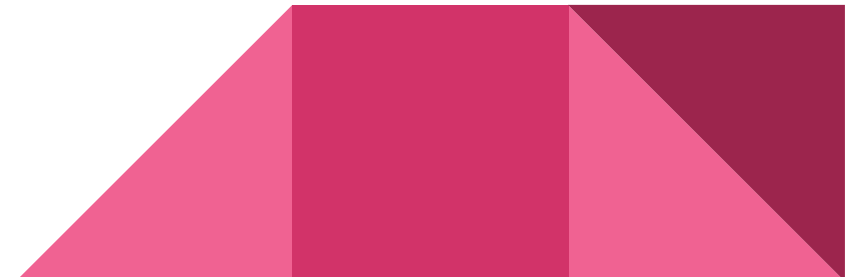
Polyglot

Choice of deployment language means ability to reuse existing libraries



Portable

WebAssembly runs in all major browsers

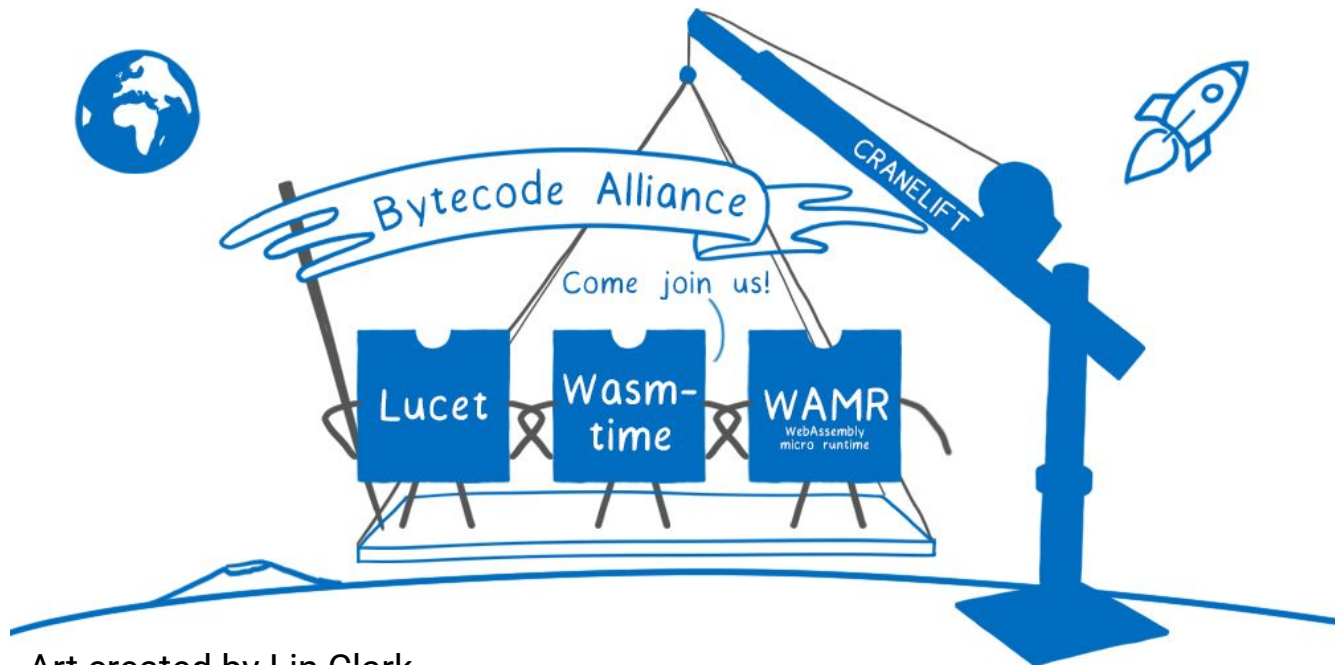


2019



WebAssembly MVP Recommended **W3C**® Standard 🎉

 **BYTECODE**
ALLIANCE formed!



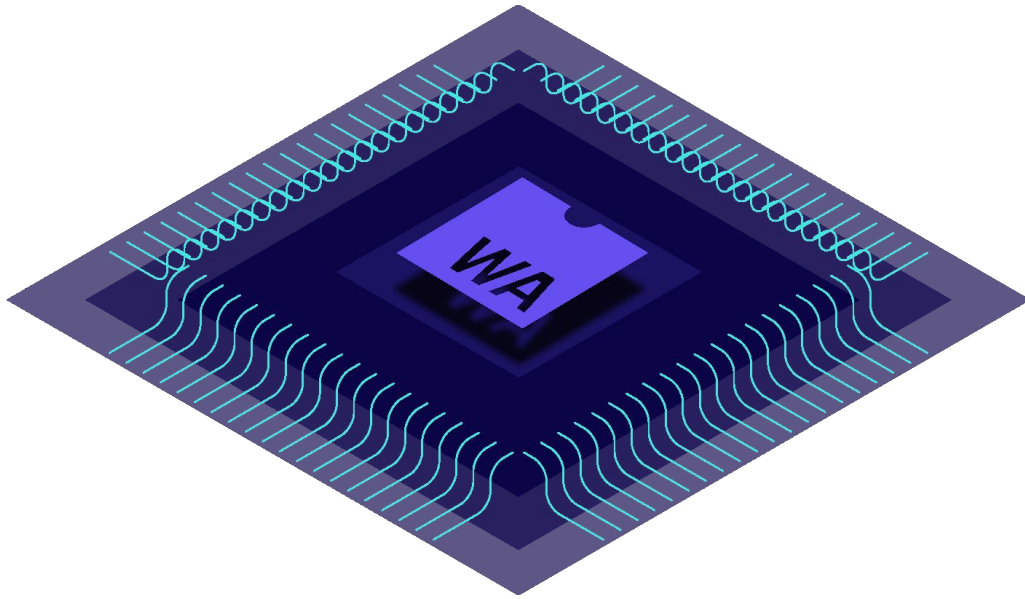
Art created by Lin Clark

How to run a .wasm



Wasm Runtime like **Wasmtime** or **WAMR** (and many more)

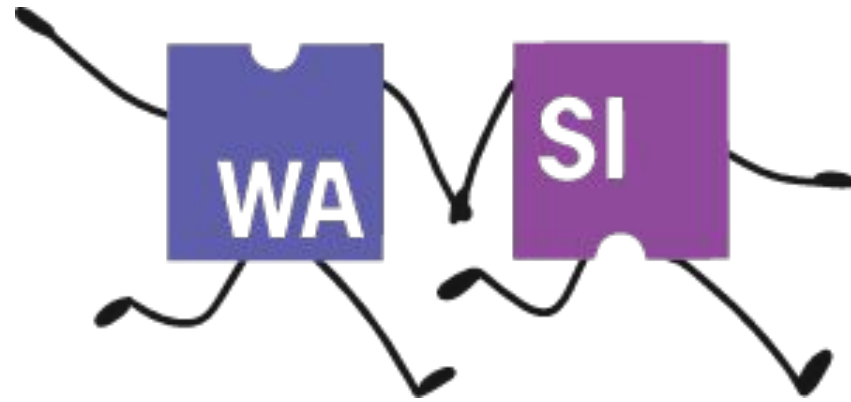
Many app platforms that embed a Wasm runtime



containerd/runwasi

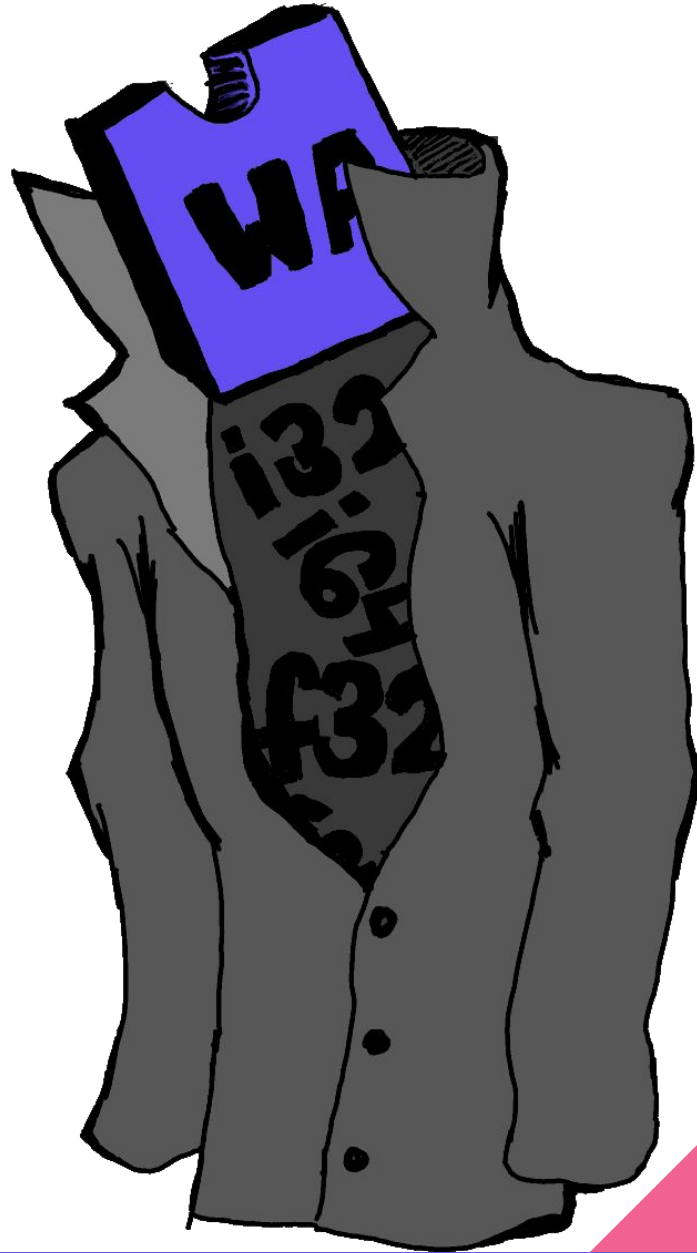


WebAssembly ~~Systems~~ Interface (WASI) ^Standard ^s



Wasm Modules

- One .wasm for entire app
- Built from one target language
- deny-by-default but really no function to call by default

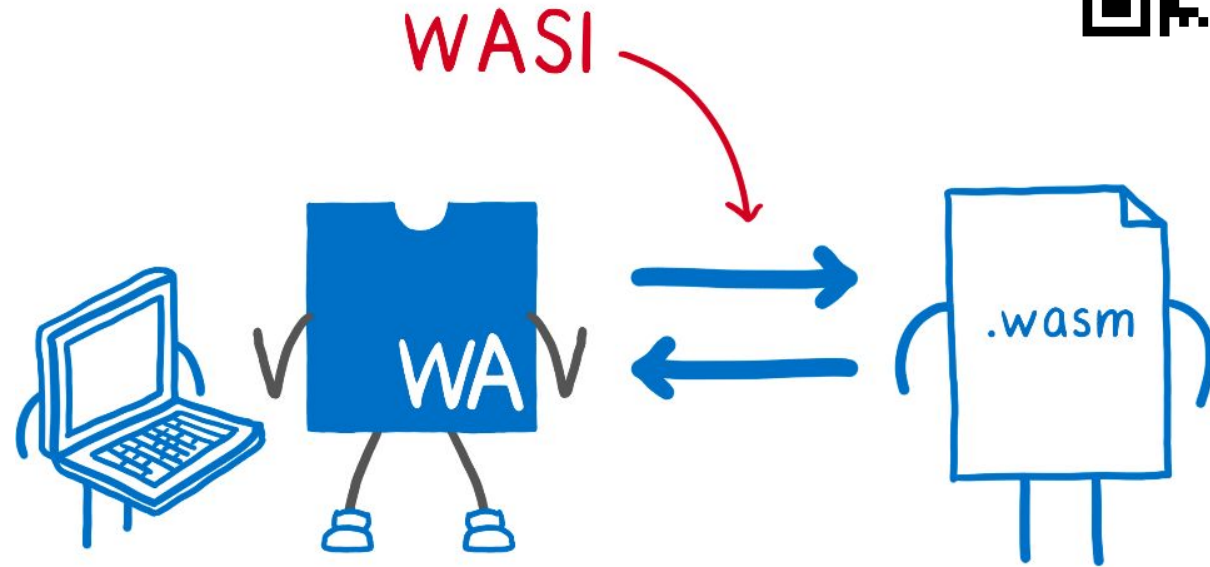


Wasm Modules with WASI



- WASI APIs include:

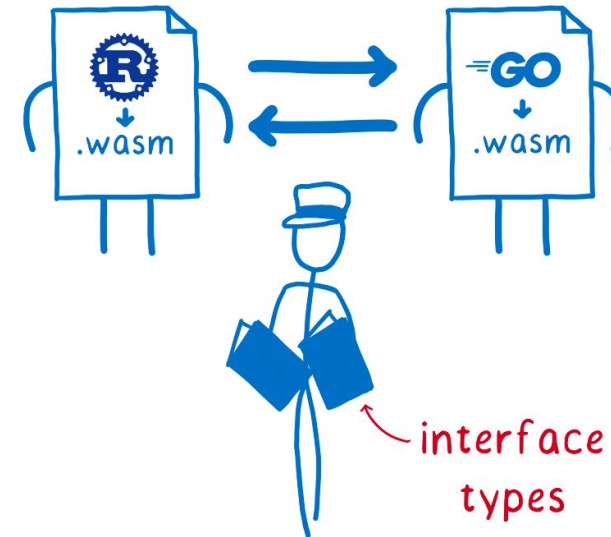
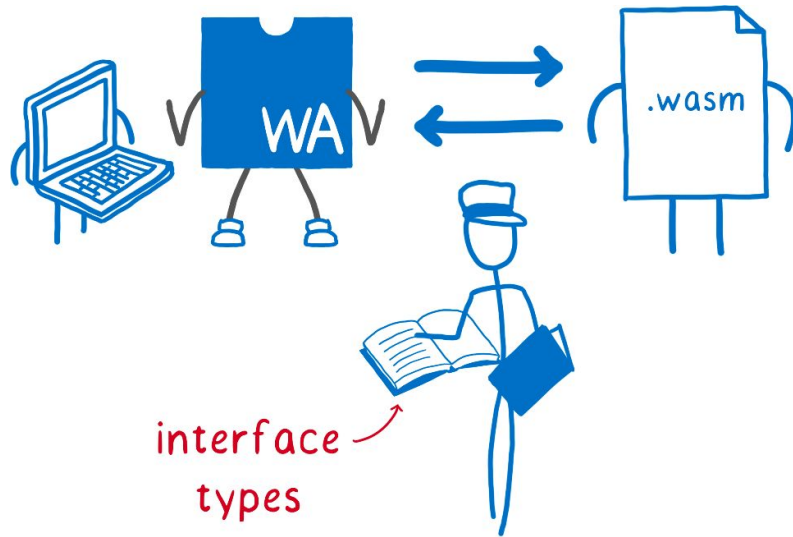
- wasi-http
- wasi-sockets
- wasi-io
- wasi-clocks



Art by Lin Clark

Wasm Components

with the Component Model



- Linking with other Wasm components
- High-level types and an IDL called **WIT**
- Platform virtualization/layering

Art by Lin Clark

Component Model

- Wasm components available with WASI Preview 2
- But very much under construction



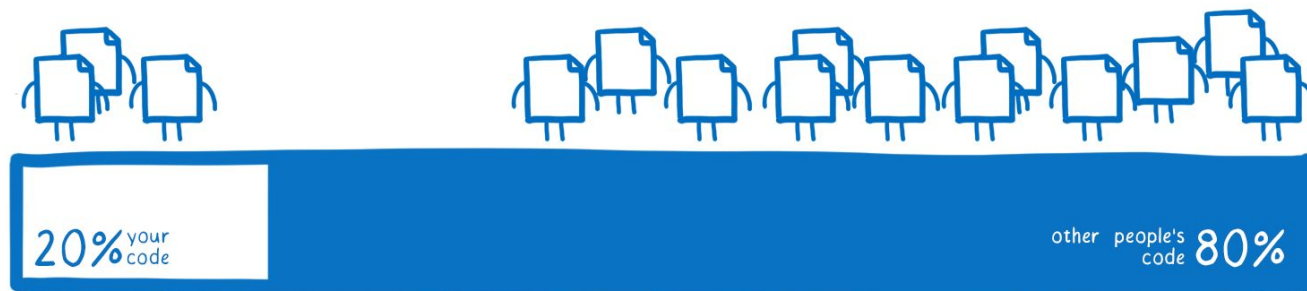
Why is this a DevOps Day Talk?

- 95% of apps are Non-Functional Requirements
 - Other people's code
- 87% of Container Images in Production Have Critical or High CVE

composition of an
average code base



[Sysdig 2023 Cloud-Native Security and Usage Report](#)



Art by Lin Clark

4 Horsemen of Development

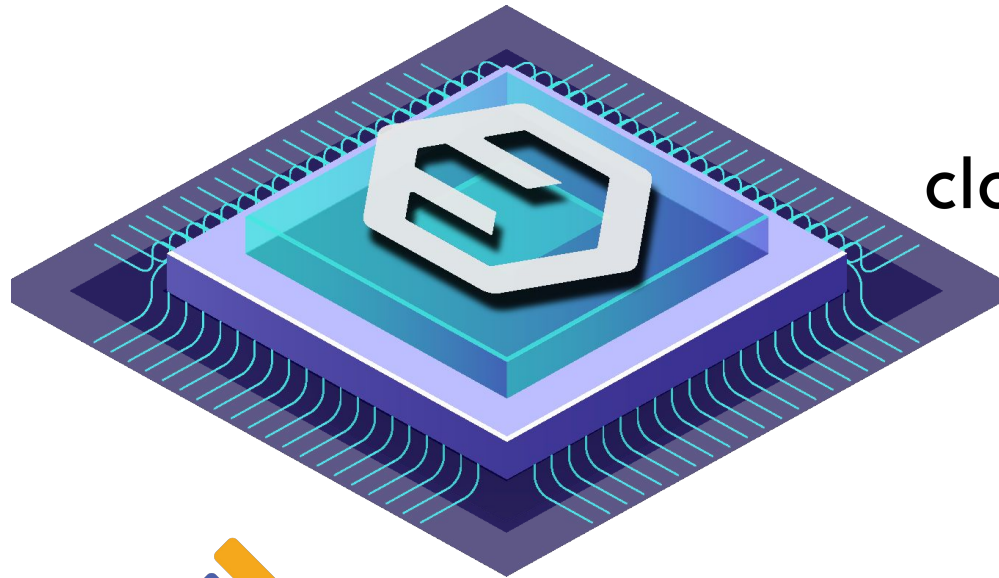
- 1) construction
- 2) cognitive
- 3) maintenance
- 4) deployment



wasmCloud Application Runtime



Open
Application
Model

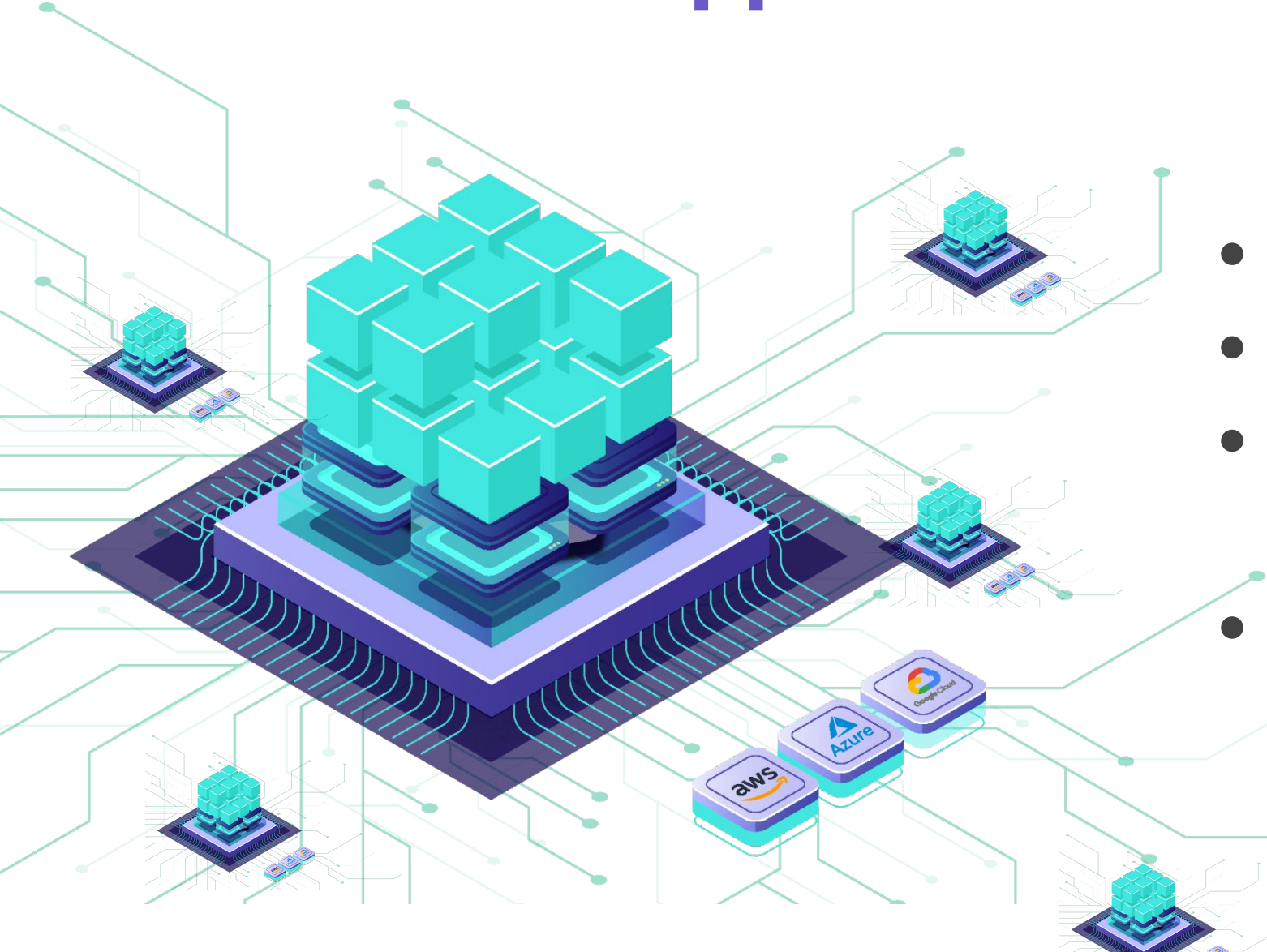


cloudevents

- Run and orchestrate Wasm
- Secure access to capabilities
- Horizontally and vertically scalable, stateless actors
- Manage networking, failover, request routing



wasmCloud Application Runtime



- Run and orchestrate Wasm
- Secure access to capabilities
- Horizontally and vertically scalable, stateless actors
- Manage networking, failover, request routing

Writing a Wasm Component

~/repos/ricochet/wasmio-2023/pingpong git:(main) (0.051s)
bat src/lib.rs

```
1 wit_bindgen::generate!({
2   path: "../wit",
3   world: "pingpong",
4 });
5
6 use crate::{
7   consumer::subscribe,
8   messaging_types::{open_broker, Channel, EventParam},
9   producer::publish,
10 };
11
12 use handler::Event;
13 struct Pingpong;
14
15 impl handler::Handler for Pingpong {
16   fn on_receive(e: Event) -> Result<(), u32> {
17     let data = e.data.unwrap();
18     let msg = String::from_utf8(data).unwrap();
19
20     match msg.as_str() {
21       "ping" => {
22         let ret_msg = format!(
23           "pong {:?}",
24           match keyvalue::get("increment") {
25             Ok(curr_val) => {
26               let increment_count = curr_val.parse::<i32>().unwrap() + 1;
27               keyvalue::set("increment", &increment_count.to_string());
28               increment_count
29             }
30             Err(_) => {
31               keyvalue::set("increment", "1");
32               1
33             }
34           }
35         );
36
37         let new_event = EventParam { data: Some(ret_msg.as_bytes()), subject: Some("pong"), };
38         publish(broker, messaging_types::Channel::Topic("pong"), new_event)?;
39       }
40       _ => {
41         println!("unknown command: {}", msg);
42       }
43     }
44     Ok(())
45   }
46 }
47
48 export_pingpong!(Pingpong);
```

```
keyvalue::set("increment",
&increment_count.to_string());
```

```
publish(broker,
messaging_types::Channel::Topic
("pong"), new_event)?;
```



Running a component in wasmCloud

on Cosmonic PaaS

```
~/repos/ricochet/wasmio-2023/pingpong git:(main) (0.129s)
```

```
cargo build --target wasm32-wasi --release
```

```
Finished release [optimized] target(s) in 0.05s
```

```
~/repos/ricochet/wasmio-2023/deploy git:(main) (3.878s)
```

```
cosmo launch --launch-only
```

```
Launching actor...
```

```
🚀 Actor launched!
```

Launch an Actor ⓘ

OCI Reference ⓘ

Enter an OCI reference URL

ghcr.io/ricochet/pingpong:0.1.0

Instances

Enter the number of actor instances you'd like to launch

1

Host

Constellation

Find Host

Find hosts with the specified labels ⓘ

Key

Value

Add

Find Hosts

Select host from the auction

Select a host to start the resource on

empty-river-8153

Close

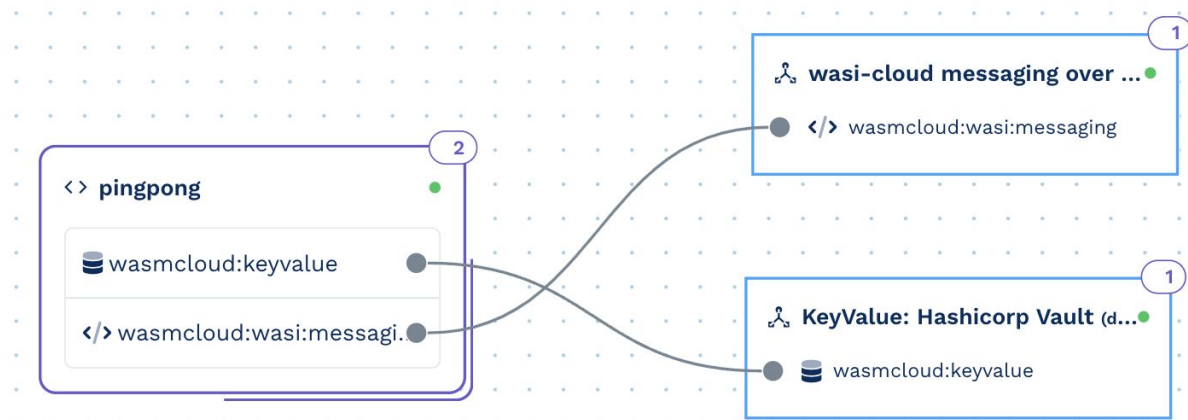
Launch Actor

Linking Capabilities

NATS Jetstream KeyValue vs Redis KeyValue



Or Vault and many more



Make this a thing of the past



Eliminating Churn

with Wasm Components and wasmCloud



- 1) construction => eliminate boilerplate
- 2) cognitive => focus on business logic
- 3) maintenance => update when logic changes
- 4) deployment => deploy from the edge to the cloud, at any scale



Join our community Slack and check out our GitHub!

<https://slack.wasmcloud.com>



<https://github.com/wasmCloud/wasmCloud>

