FERMYON WORKSHOP

Getting Started with Serverless Wasm with Spin

What will you take away?

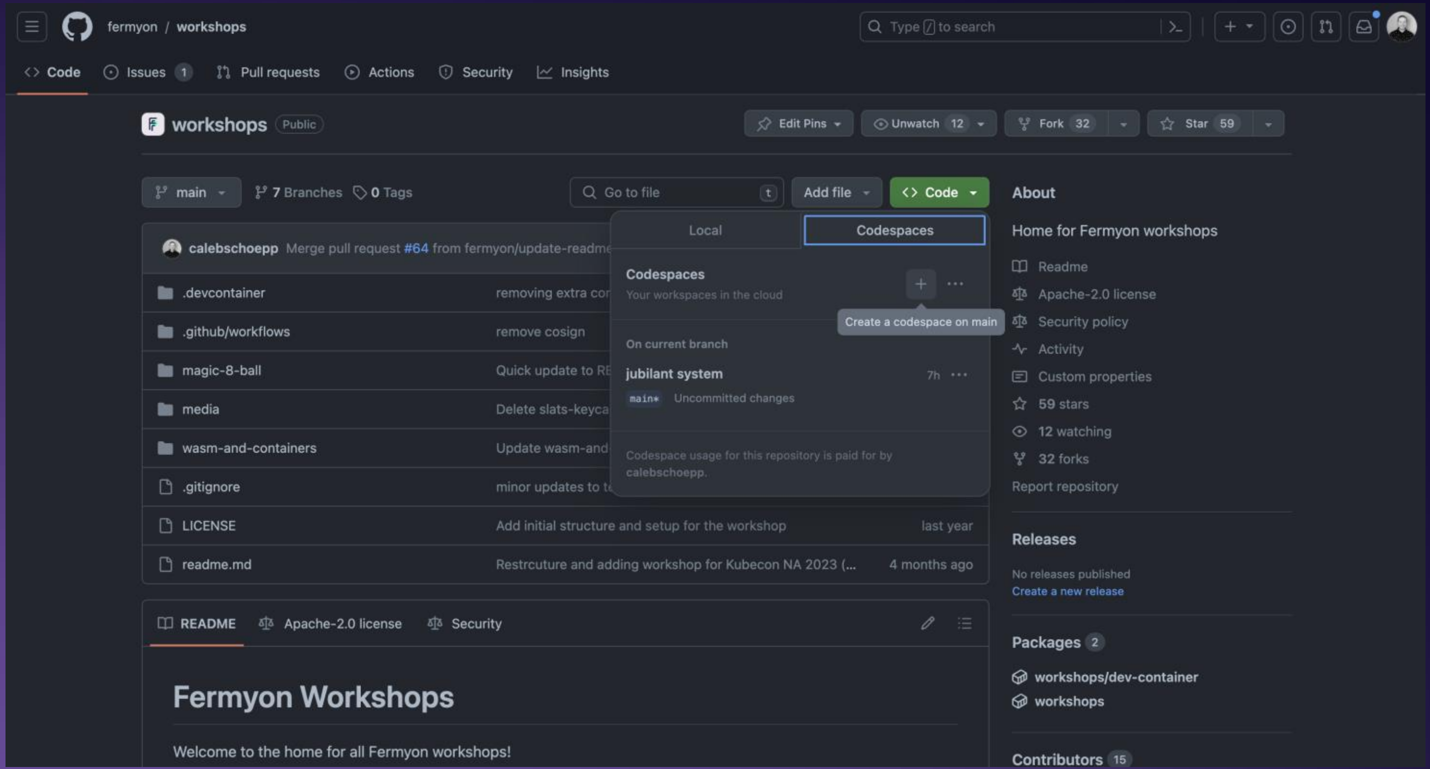
First-hand experience using server-side Wasm

- Familiarity with Spin framework features including
    - HTTP Routing
    - Serverless AI
    - Key Value Store
    - SQLite / NoOps DB
- Deployment of Spin application to multiple targets including
    - Locally
    - Fermyon Cloud

Agenda

- Introduction to WebAssembly
- Introduction to Spin
- Overview of the workshop
- Live-coding walkthrough of the workshop

Setting up a codespace



calebschoepp Merge pull request #64 from fermyon/update-readme

.devcontainer	removing extra cor	
.github/workflows	remove cosign	
magic-8-ball	Quick update to RE	
media	Delete slats-keyca	
wasm-and-containers	Update wasm-and	
.gitignore	minor updates to t	
LICENSE	Add initial structure and setup for the workshop	last year
readme.md	Restrcuture and adding workshop for Kubecon NA 2023 (...	4 months ago

Codespaces
Your workspaces in the cloud

On current branch

jubilant system 7h
main* Uncommitted changes

Codespace usage for this repository is paid for by calebschoepp.

About

Home for Fermyon workshops

- Readme
- Apache-2.0 license
- Security policy
- Activity
- Custom properties
- 59 stars
- 12 watching
- 32 forks

Report repository

Releases

No releases published
Create a new release

Packages 2

- workshops/dev-container
- workshops

Contributors 15

README Apache-2.0 license Security

Fermyon Workshops

Welcome to the home for all Fermyon workshops!

Workshop: github.com/fermyon/workshops

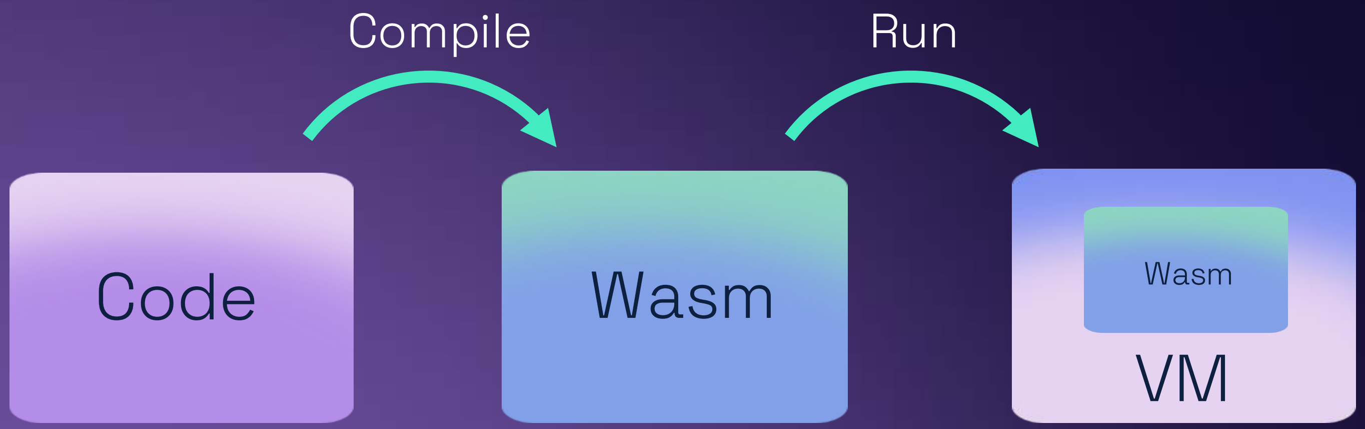
Poll: Have you used WebAssembly before?

- Yes, a lot!
- I’ve tried it out, but that’s about it.
- Not at all.

What are the three most important things to know about WebAssembly?

- It is a specification
- Of a binary instruction format, designed as a portable compilation target, and…
- Wasm is an abbreviation for it (also it's not an acronym ☺)

Compile and Run



Compile

Run

Code

Wasm

Wasm

VM

FERMYON

Four reasons to care about Wasm

- Security — Sandboxed execution environment.
- Performance — Near native execution speed.
- Polyglot — Supports a wide array of languages.
- Portability — Cross platform and cross architecture.

WASI: a system interface for the Wasm platform

Make it portable

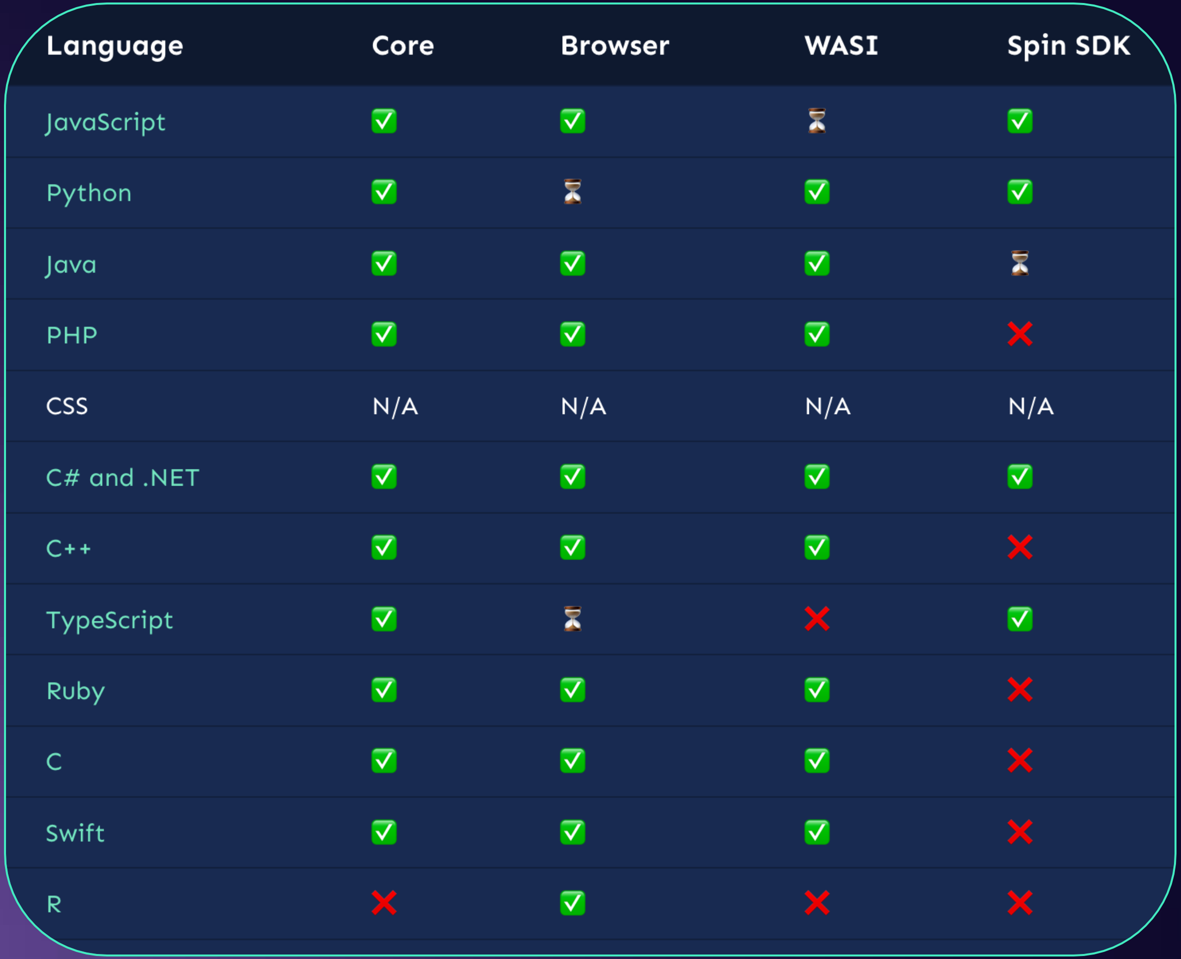
- Code outside of a browser needs a way to talk to a conceptual operating system.
- E.g., files, sockets, clocks, random numbers and many more higher-level types of resources

Make it secure

- Preserve in-browser security model through WASI’s Capability based security model

Language Support

- Guide tracks support for compiling a language to WebAssembly.
- Three sections:
  - Support for the top 20 languages
  - WebAssembly-specific languages
  - Other notable languages.



Language	Core	Browser	WASI	Spin SDK
JavaScript	✅	✅	⏳	✅
Python	✅	⏳	✅	✅
Java	✅	✅	✅	⏳
PHP	✅	✅	✅	❌
CSS	N/A	N/A	N/A	N/A
C# and .NET	✅	✅	✅	✅
C++	✅	✅	✅	❌
TypeScript	✅	⏳	❌	✅
Ruby	✅	✅	✅	❌
C	✅	✅	✅	❌
Swift	✅	✅	✅	❌
R	❌	✅	❌	❌

https://developer.fermyon.com/wasm-languages/webassembly-language-support/

JavaScript runtimes

Designed to complement and run alongside JS to share functionality between JS and Wasm

	V8	Spidermonkey
Browser		
Server-side	node	

WASI runtimes

Designed to be independent of browsers, so it doesn't depend on Web APIs or JS, and isn't limited by the need to be compatible with JS

Wasmtime

What are good use-cases for Wasm?

Cloud

Functions-as-a-Service Frameworks

Extensibility with the component-model

Plug-ins

User-Defined Functions for databases

Bring-your-own-code in SaaS platforms

IoT

System resource usage

No dependencies to carry along

Developer and Operator experiences

Quick start-up time
Size of workload
Security model
Portability

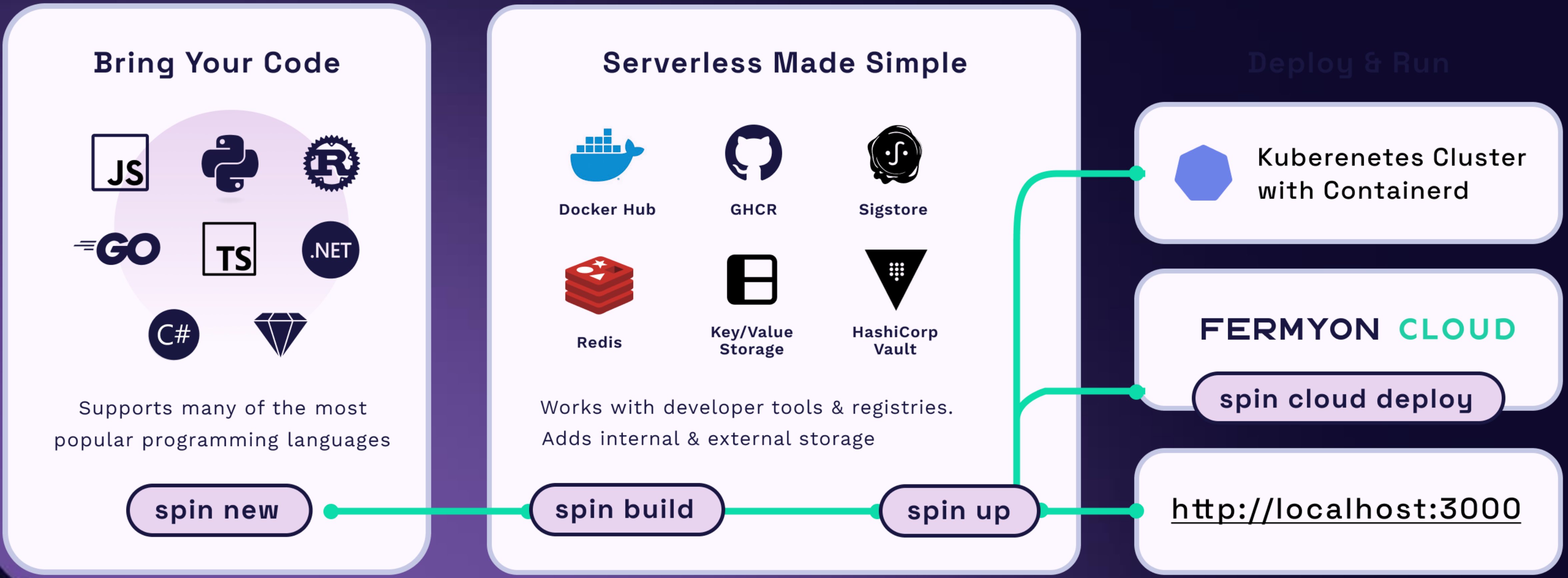
INTRO

SPIN

Spin is the developer tool for building serverless Wasm applications

SPIN 1.0

The Developer Tool for Serverless WebAssembly



Bring Your Code

JS TS .NET C#

Supports many of the most popular programming languages

spin new

Serverless Made Simple

Docker Hub

GHCR

Sigstore

Redis

Key/Value Storage

HashiCorp Vault

Works with developer tools & registries.
Adds internal & external storage

spin build

spin up

Deploy & Run

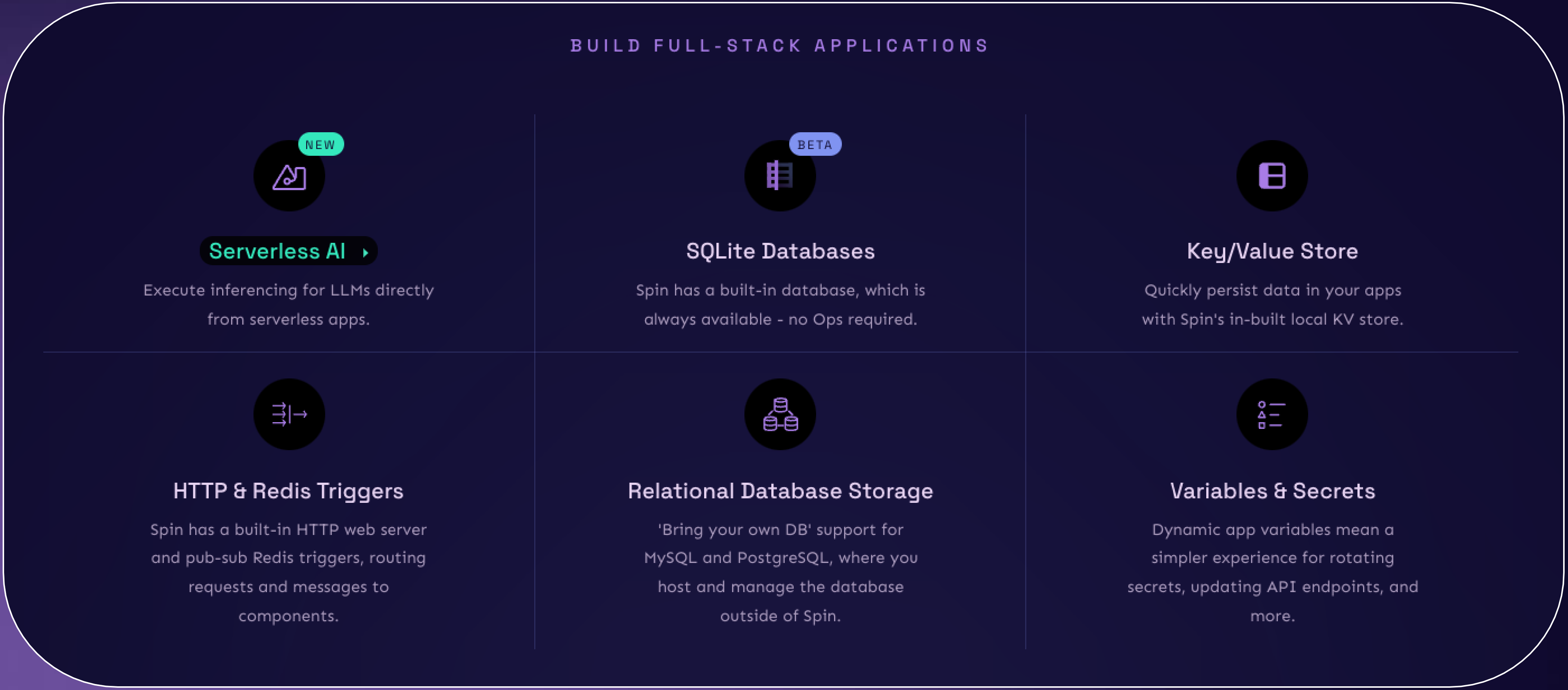
Kuberenetes Cluster with Containerd

FERMYON CLOUD

spin cloud deploy

http://localhost:3000

What's new in Spin?



BUILD FULL-STACK APPLICATIONS

Serverless AI ▸

Execute inferencing for LLMs directly from serverless apps.

SQLite Databases

Spin has a built-in database, which is always available - no Ops required.

Key/Value Store

Quickly persist data in your apps with Spin's in-built local KV store.

HTTP & Redis Triggers

Spin has a built-in HTTP web server and pub-sub Redis triggers, routing requests and messages to components.

Relational Database Storage

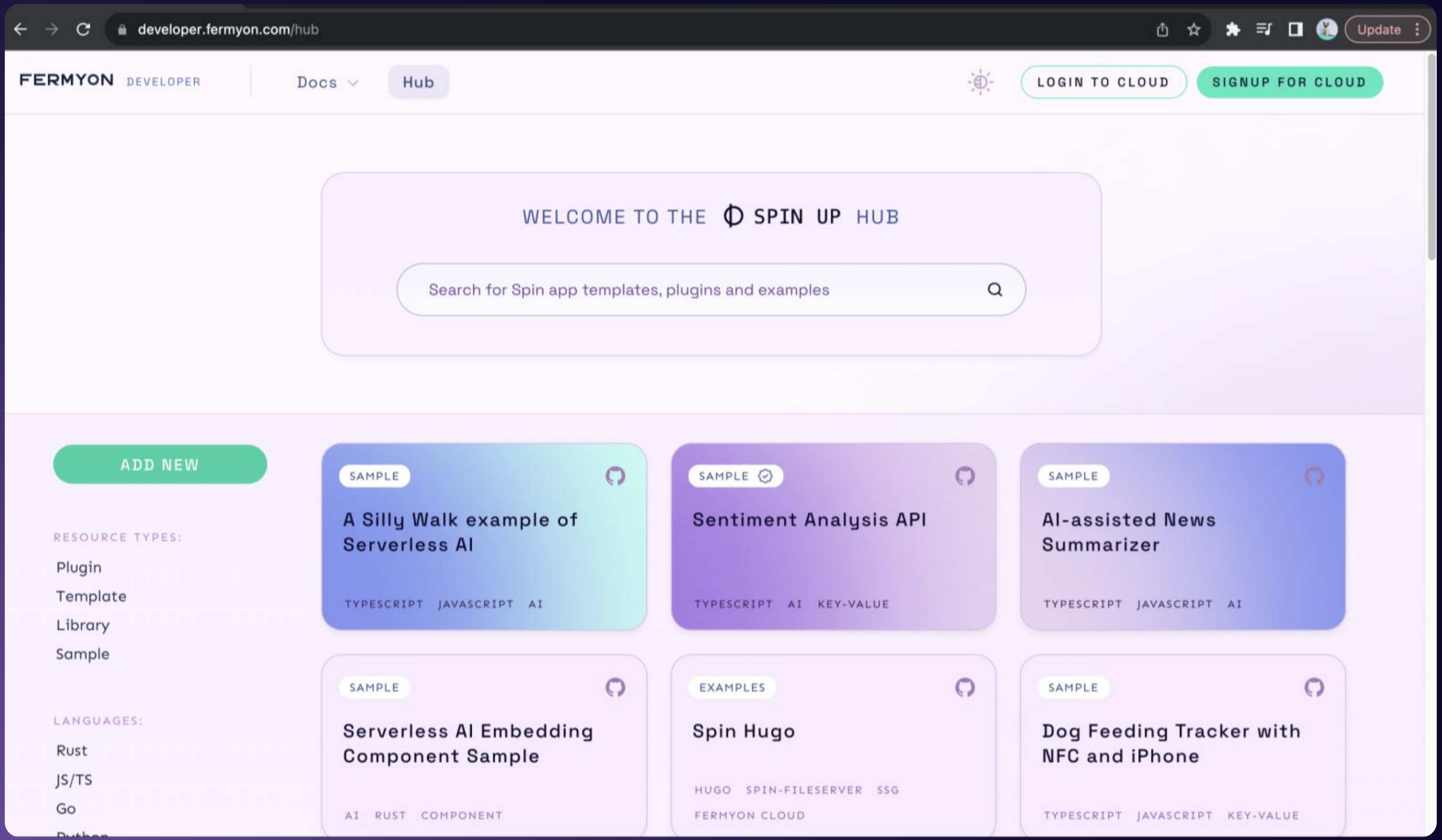
'Bring your own DB' support for MySQL and PostgreSQL, where you host and manage the database outside of Spin.

Variables & Secrets

Dynamic app variables mean a simpler experience for rotating secrets, updating API endpoints, and more.

FERMYON

Discover and share content on Spin Up Hub



WELCOME TO THE SPIN UP HUB

Search for Spin app templates, plugins and examples

SAMPLE
A Silly Walk example of Serverless AI
TYPESCRIPT JAVASCRIPT AI

SAMPLE
Sentiment Analysis API
TYPESCRIPT AI KEY-VALUE

SAMPLE
AI-assisted News Summarizer
TYPESCRIPT JAVASCRIPT AI

SAMPLE
Serverless AI Embedding Component Sample
AI RUST COMPONENT

EXAMPLES
Spin Hugo
HUGO SPIN-FILESERVER SSG FERMYON CLOUD

SAMPLE
Dog Feeding Tracker with NFC and iPhone
TYPESCRIPT JAVASCRIPT KEY-VALUE

developer.fermyon.com/hub

FERMYON CLOUD

The quickest way to deploy and manage your serverless WebAssembly apps.

Architected for sub-millisecond cold starts and high-volume time-slicing of instances.

FULL-STACK PLATFORM

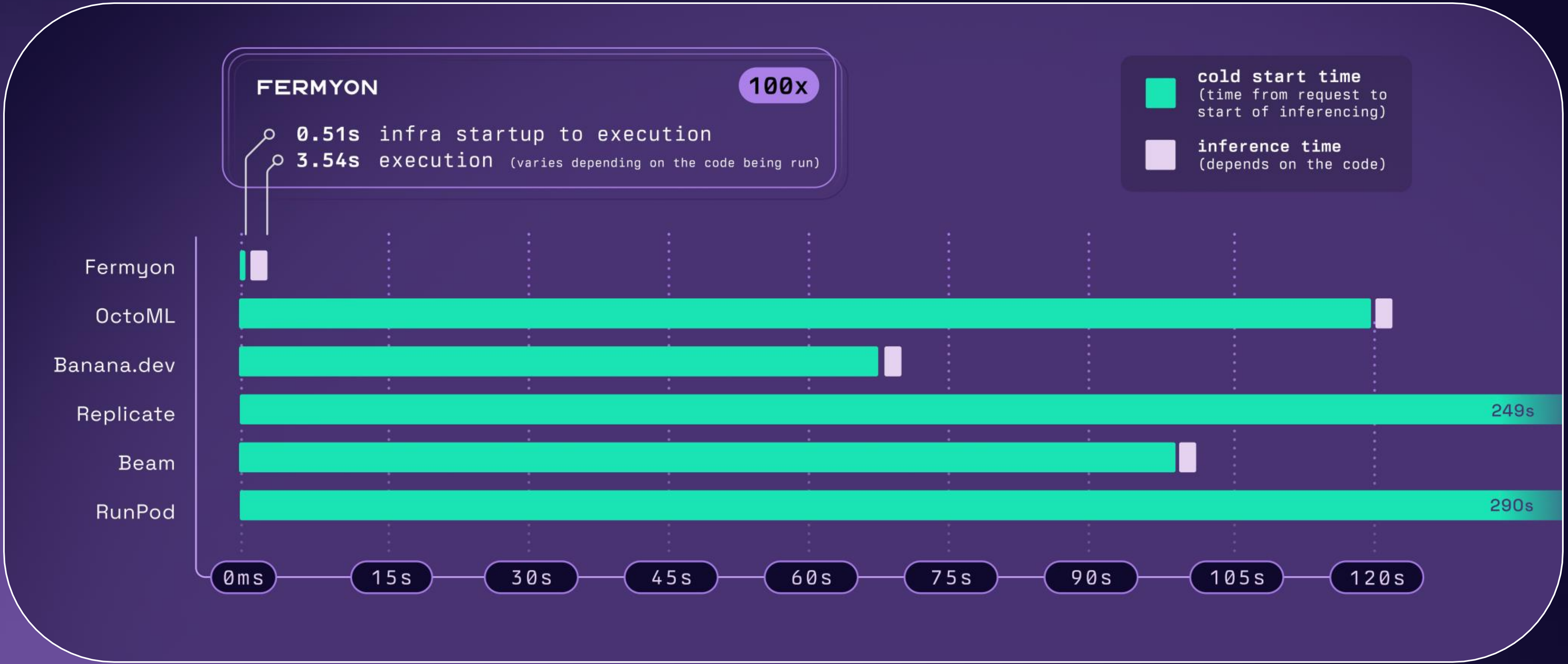
STORAGE

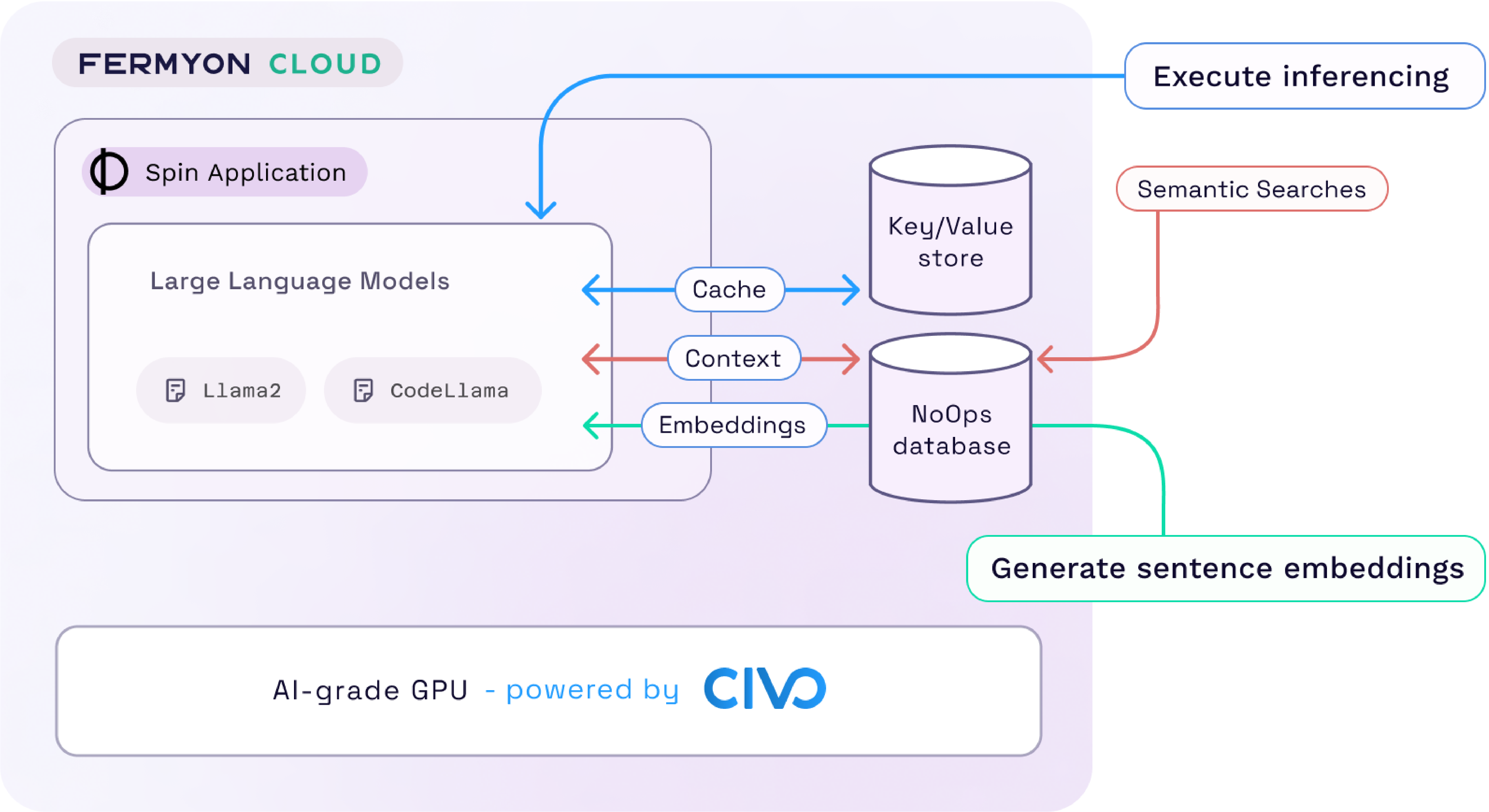
DOMAINS

GPU COMPUTE

FREE

Fermyon Serverless AI

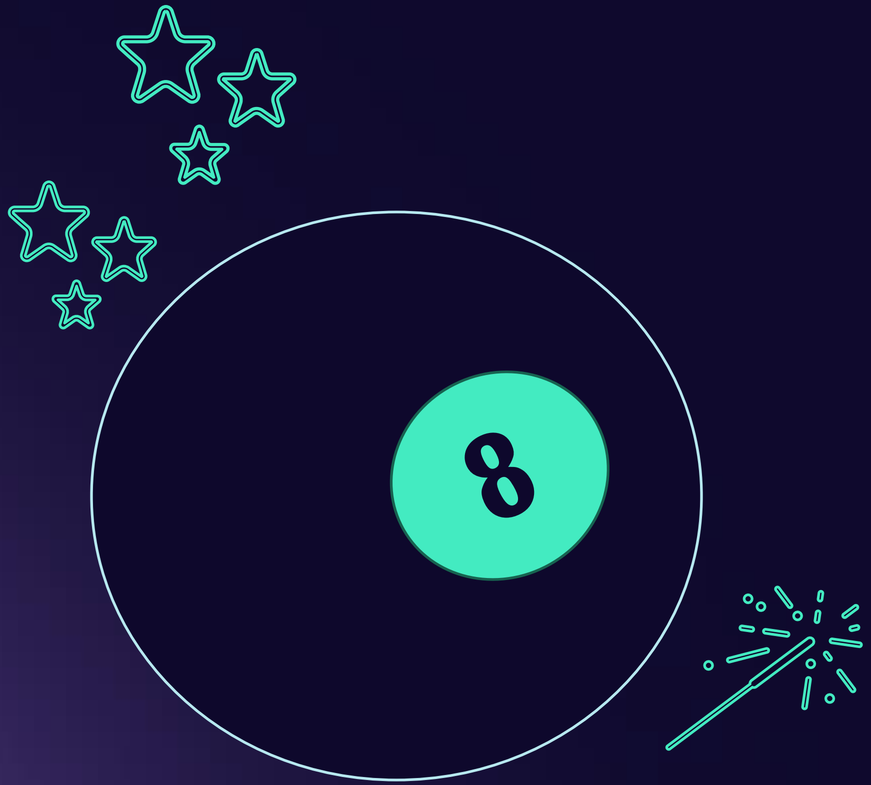



FERMYON CLOUD
Execute inferencing
Spin Application
Key/Value store
Semantic Searches
Large Language Models
Cache
Context
Llama2
CodeLlama
Embeddings
NoOps database
Generate sentence embeddings
AI-grade GPU - powered by CIVO

WORKSHOP

FERMYON

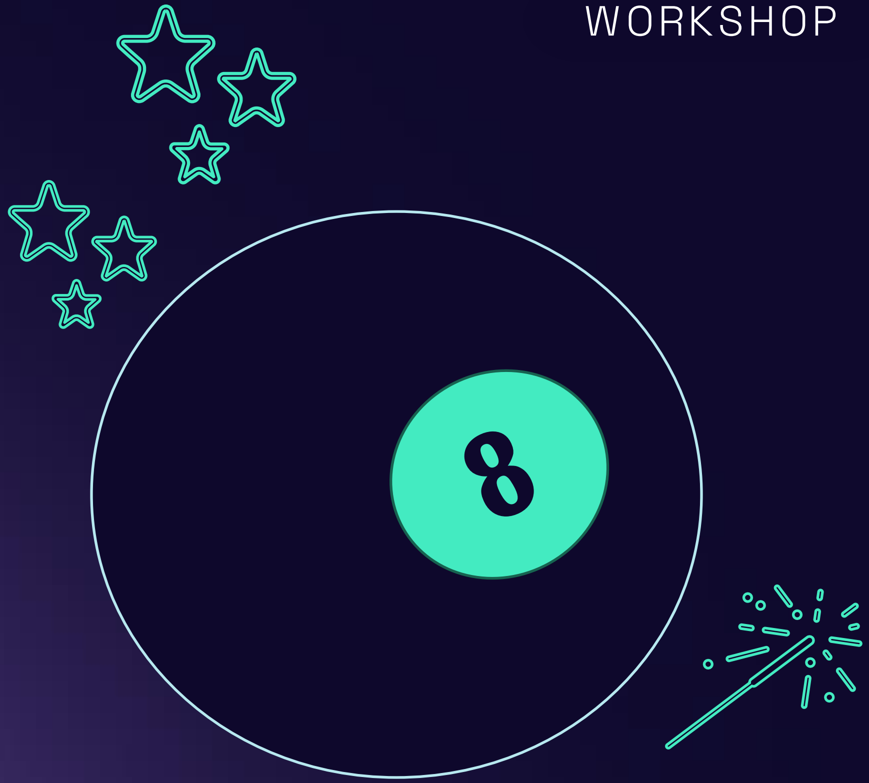
Let’s get hands-on



FERMYON

WORKSHOP

Spinning a Magic 8 Ball



A Magic 8 Ball

Returns a random response to a question

A MAGIC 8 Ball

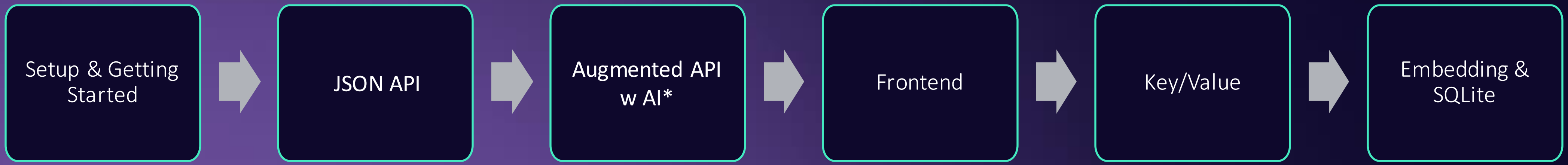
Remembers the responses to questions

A MAGIC **AI-GHT** Ball

Returns a sophisticated response to a question, backed by a LLM

Workshop

https://github.com/fermyon/workshops/blob/main/magic-8-ball/README.md



Setup & Getting Started

JSON API

Augmented API w AI*

Frontend

Key/Value

Embedding & SQLite

*Enhanced with Serverless AI

Workshop Flow

- ✓ 00 - Setup
- ✓ 01 - Getting Started with Spin
- ✓ 02 - Building a JSON API with Spin
- ✓ 03 - Enhancing JSON API with Fermyon Serverless AI
- ✓ 04 - Running your front-end application with Spin
- ✓ 05 - Persisting non-relational data with Spin's key/value store
- ✓ 06 - Deploying to Fermyon Cloud
- ✓ 07 - Bonus Exercise: Using an external database
- ✓ 08 - Embeddings and SQLite

https://github.com/fermyon/workshops

FERMYON

What did you think?

bit.ly/fermyondiscord

Follow along on Twitter

@fermyontech

@spinframework

SPIN 1.0