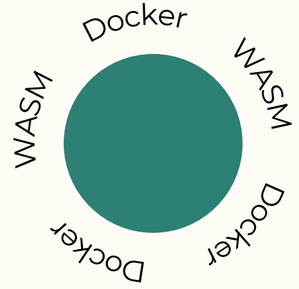
KubeCon China 2025

WASM vs Docker: Partners, Not Rivals

Presentation by
Pradumna Saraf



你好

$ whoami

Pradumna Saraf

- Open Source Developer/DevRel
- Docker Captain
- Create Content

Solomon Hykes

@solomonstre

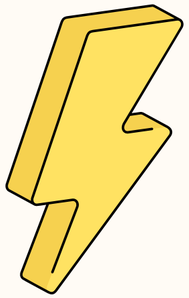
If WASM+WASI existed in 2008, we wouldn't have needed to created Docker. That's how important it is. Webassembly on the server is the future of computing. A standardized system interface was the missing link. Let's hope WASI is up to the task!

WebAssembly (WASM) is a binary instruction format that helps run high-performance code written in languages like C, C++, or Rust directly in the browser.

This makes it possible to use complex apps like Figma and Photoshop or play advanced games online, all without installing anything.

Vs

Package Code	Near Native Performance
High Portability	High Security
Runtime Isolation	Quick Startup Time



Package Code

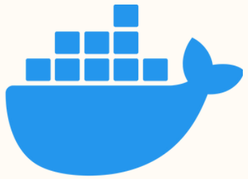
High Portability

Runtime Isolation

Near Native Performance

High Security

Quick Startup Time



Package Code

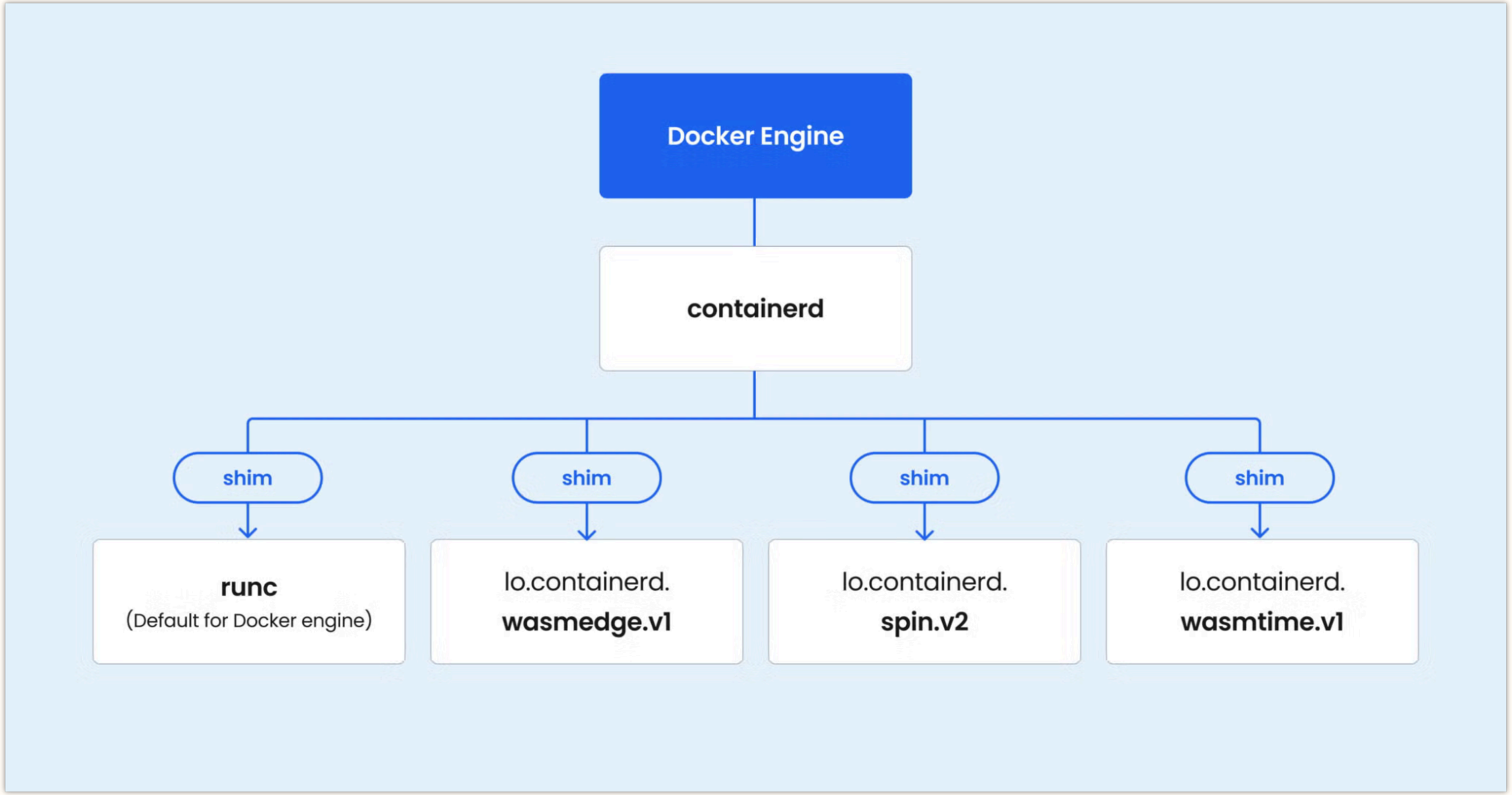
High Portability

High Security

Runtime Isolation

Near Native Performance

Quick Startup Time


Docker Engine
containerd
shim
shim
shim
shim
runc
(Default for Docker engine)
lo.containerd.
wasmedge.v1
lo.containerd.
spin.v2
lo.containerd.
wasmtime.v1

- Running Windows, Linux, and **WebAssembly containers** side by side.
- Supporting multiple **WASM runtimes**.
- **Workflow**: WASM containers are being built, tagged, and pushed to Docker Hub using standard Docker workflows. Docker Desktop integration, the same docker build, docker tag, and docker push commands. Plus, they are OCI-compliant

```
1  FROM scratch
2  COPY /helloworld.wasm /helloworld.wasm
3  ENTRYPOINT ["helloworld.wasm"]
```

Connect With Me

links.pradumnasaraf.dev