From event-driven to automotive

P J Łaszkowicz (2023) / omnifi.foundation

Introduction

From event-driven to automotive

- Where WebAssembly works, and where it often doesn’t
- Common constraints of full-stack WebAssembly
- Shifting architectural patterns with WebAssembly
- Tooling landscape
- What’s next

Going privacy-first

Going privacy-first

Key requirements

Build a modern web app that:

- Performs machine learning inference in-browser
- Uses web services as a progressive enhancement
- Re-uses models built for native mobile runtimes
- Trains the model with usage data

Going privacy-first

Key components

A modern privacy-first machine learning stack for the Web:

- PyTorch for model development
- TensorFlow.js for inference
- PySyft and PyGrid for federated learning
- Workbox for service workers
- HTML, CSS, and JavaScript

Building bridges

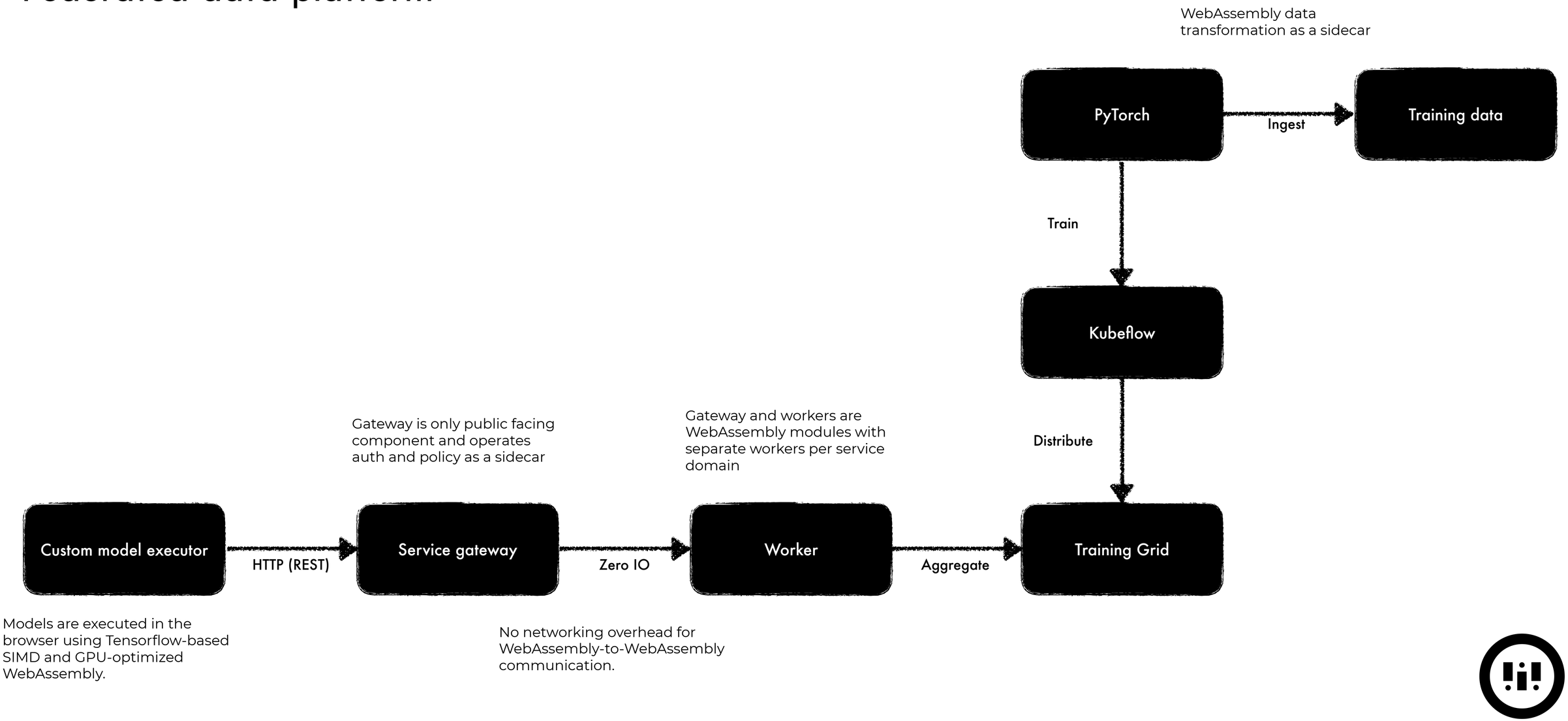
On the (Cloudflare) edge

When extending from the browser to the server, we:

- Used modular packages and repositories
- Determined shared objects using domain-driven design
- Started with web services using Cloudflare Workers
- Extended Kubernetes and the training pipeline

Going privacy-first

Federated data platform



Going privacy-first

Retrospective

What we discovered whilst building a privacy-first web app:

- Browsers are great at rendering
- JavaScript is fast (enough)
- Apps don’t need to send data anywhere*

Isomorphic analytics

Isomorphic analytics

Key requirements

Replacing legacy products and services, that:

- Use JavaScript extensions on the server
- Use the JVM, Python, and R for analytics and statistical modelling
- Run on medical equipment and in-browser
- Have unknown connectivity capabilities

Taking WebAssembly for a Spin

Scaling services rapidly

Building from previous experiences, we:

- Used Spin from Fermyon on Nomad for some new services
- Built isomorphic analytical functions
- Extended data processing to in-process Kafka event-queues
- Extended Kubernetes clusters for critical infrastructure

In-process extensions

Moving data processing closer to the ingress

Using RedPanda as a Kafka drop-in replacement enabled:

- Inline processing for a serverless delivery pattern, anywhere
- Reduced network overhead due to sidecar architecture
- High-performance data transformations at huge scale
- Re-useable domain code in data ingest and core services
- Applied regulatory and data governance on-queue

WebAssembly as critical infrastructure

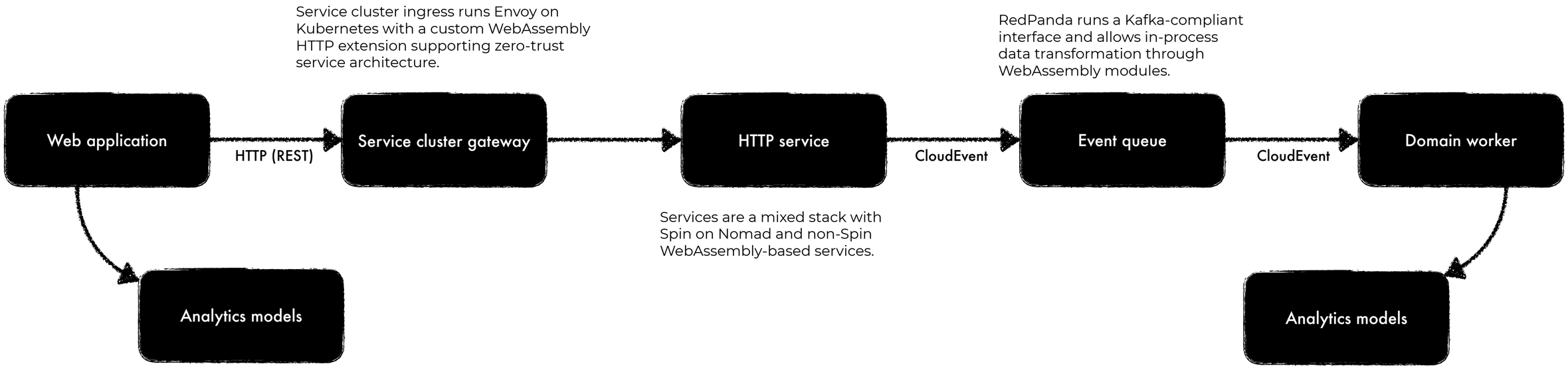
Universal approach to policy control

Using Envoy as the core part of our proxy and service mesh we:

- Can apply a singular policy control across all clusters easily
- Can optimise traffic for our specific network stack
- Support closer integrations with hybrid environments
- Reuse API routing universally across various target environments
- Have more comprehensive control over critical infrastructure

Isomorphic analytics

Target on-premise architecture



Automotive and decentralized

Automotive and decentralized

Key requirements

An open ecosystem for the automotive industry, where:

- Data is private, and actionable by all
- Focus is on ecosystems, rather than products or platforms
- Highly computational workloads due to cryptography
- Logic and structure re-use is critical

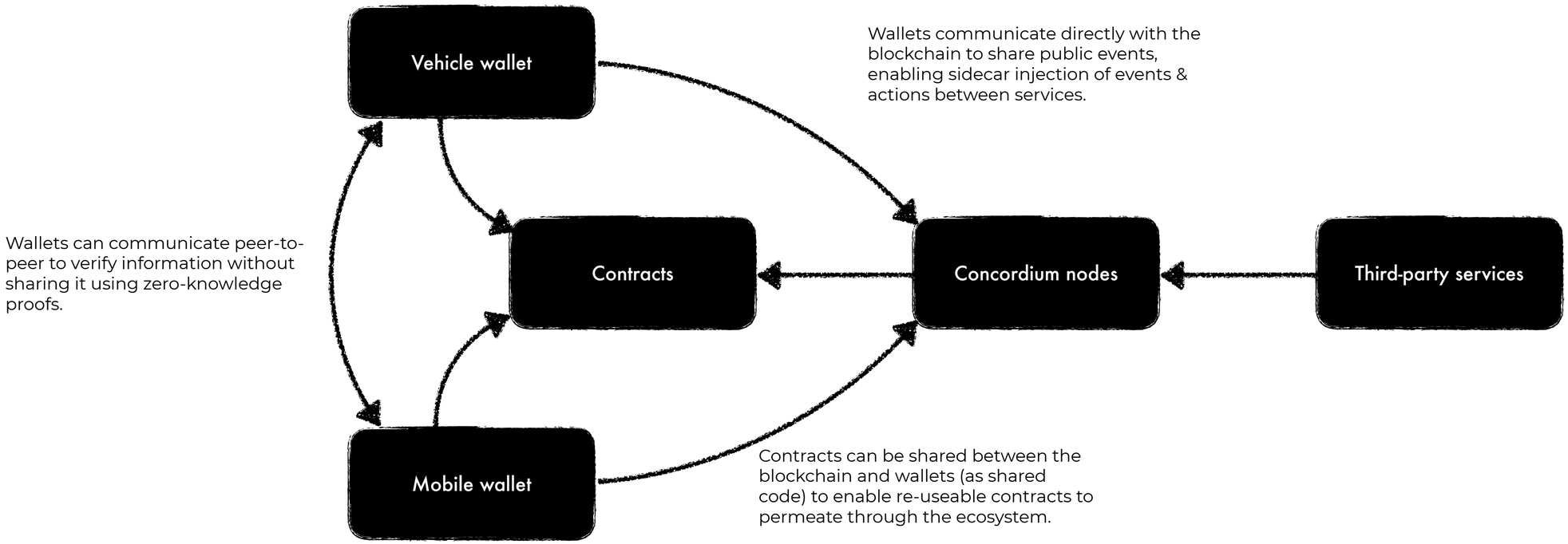
Smart contracts as extensions

Plugging into public networks

- Every contract is public and verifiable
- Contracts can never be updated or deleted
- Authorization and RBAC requires delegation
- Testing can be (more) complicated
- Accidents are (far more) damaging

Automotive and decentralized

Target architecture



Going Fastly

(Another) edge compute

Running multiple projects on Fastly:

- Easy(-ish) to build on
- Performant with zero maintenance
- No (real) database or storage
- HTTP interface needed to be customised (in Rust)

Data on the edge

This was surprisingly difficult

- Most edge environments lack any real database solution
- Edge databases are ambiguous regarding regulations
- Securing edge databases can be very difficult
- Supporting databases is complex
- TerminusDB is a promising edge-compliant knowledge graph
- SurrealDB is simple and (mostly) painless

Naamio

A Solid foundation

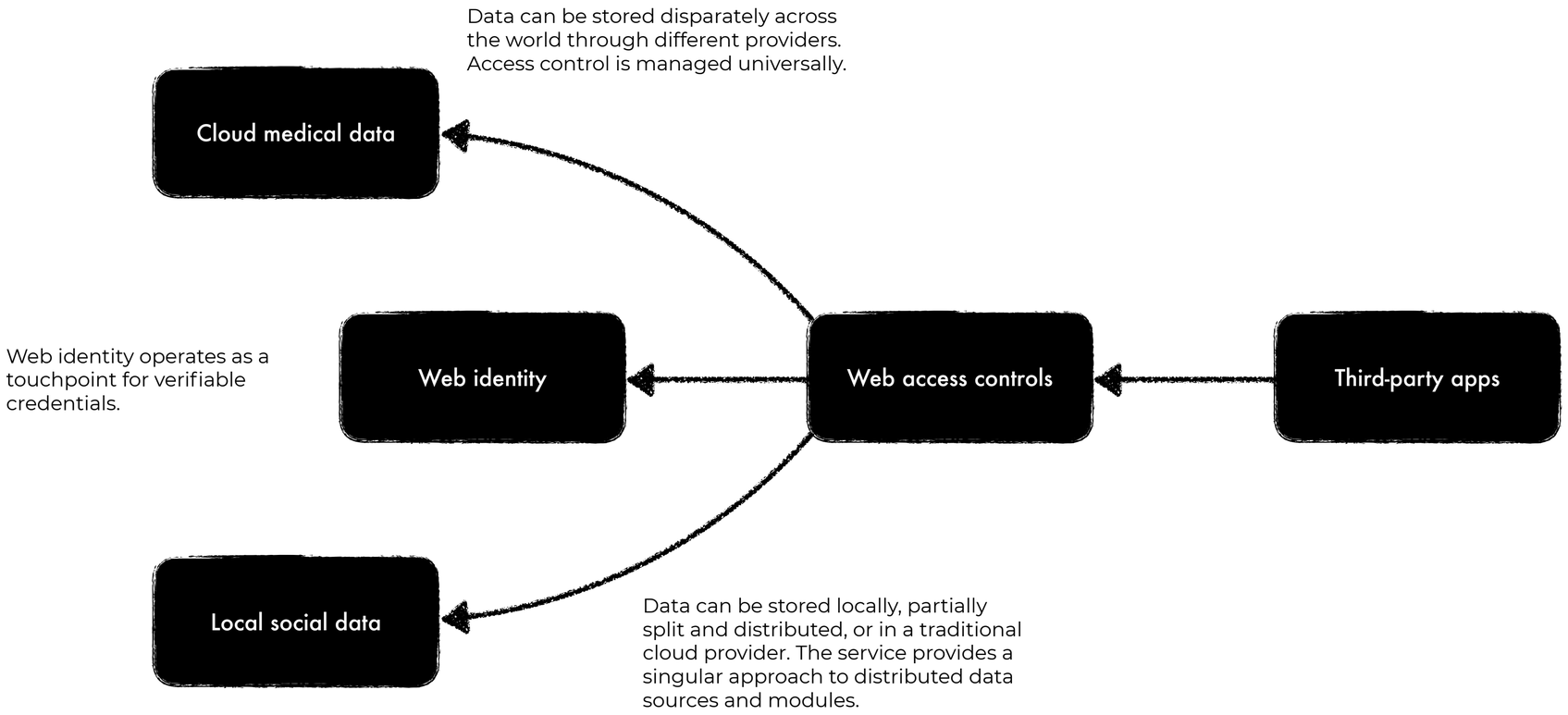
Self-sovereign data spaces

Naamio is a distributed cloud project designed for humans:

- Everything open sourced under an ethical license
- Built on Solid to support powerful, personal data spaces
- Support for ActivityPub and the Fediverse
- Designed to be convenient and simple
- Supports privacy-preserving AI for zero compromise
- Fully extensible with WebAssembly modules
- Can run on commodity hardware at home or on the cloud

Naamio

Decentralized knowledge graphs



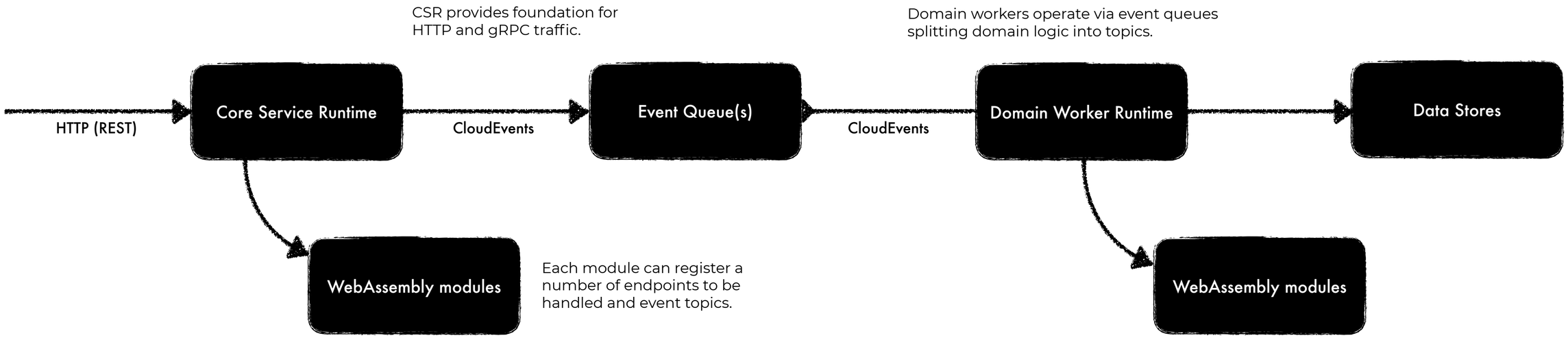
A hot mesh

Self-sovereign data meshes

- Data lakes and meshes are expensive and difficult
- A lot of mediocre re-invention of the same blueprints
- Regulation and ethics mean data control is complex
- Data has meaning and should be useful to those creating it
- Anyone can build a web site, data spaces should be just as easy

Modern Data Mesh Revisited

Domain-Driven Service Development

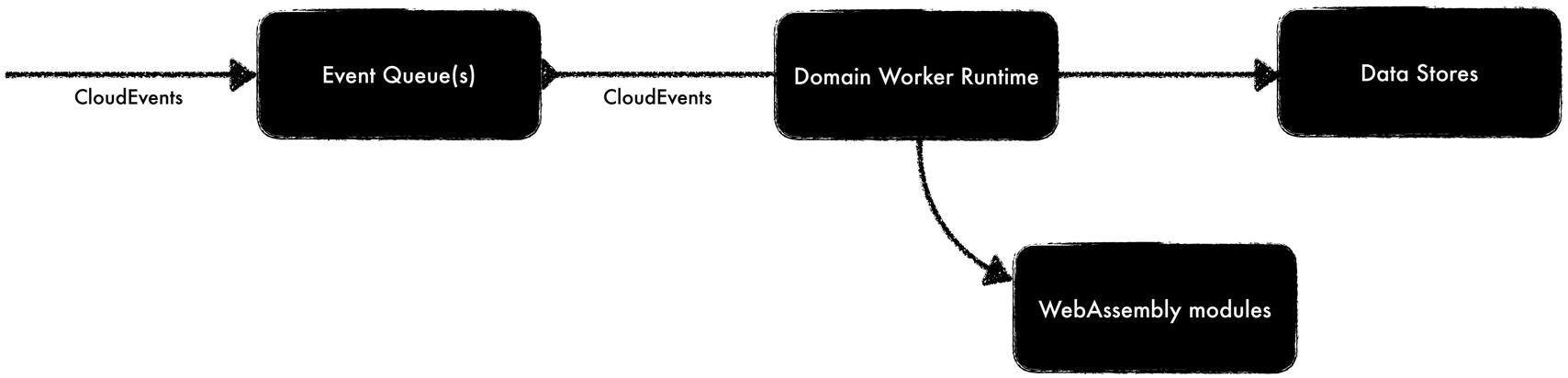


Modern Data Mesh Revisited

Domain-Driven Service Development

CSR provides foundation for HTTP and gRPC traffic.

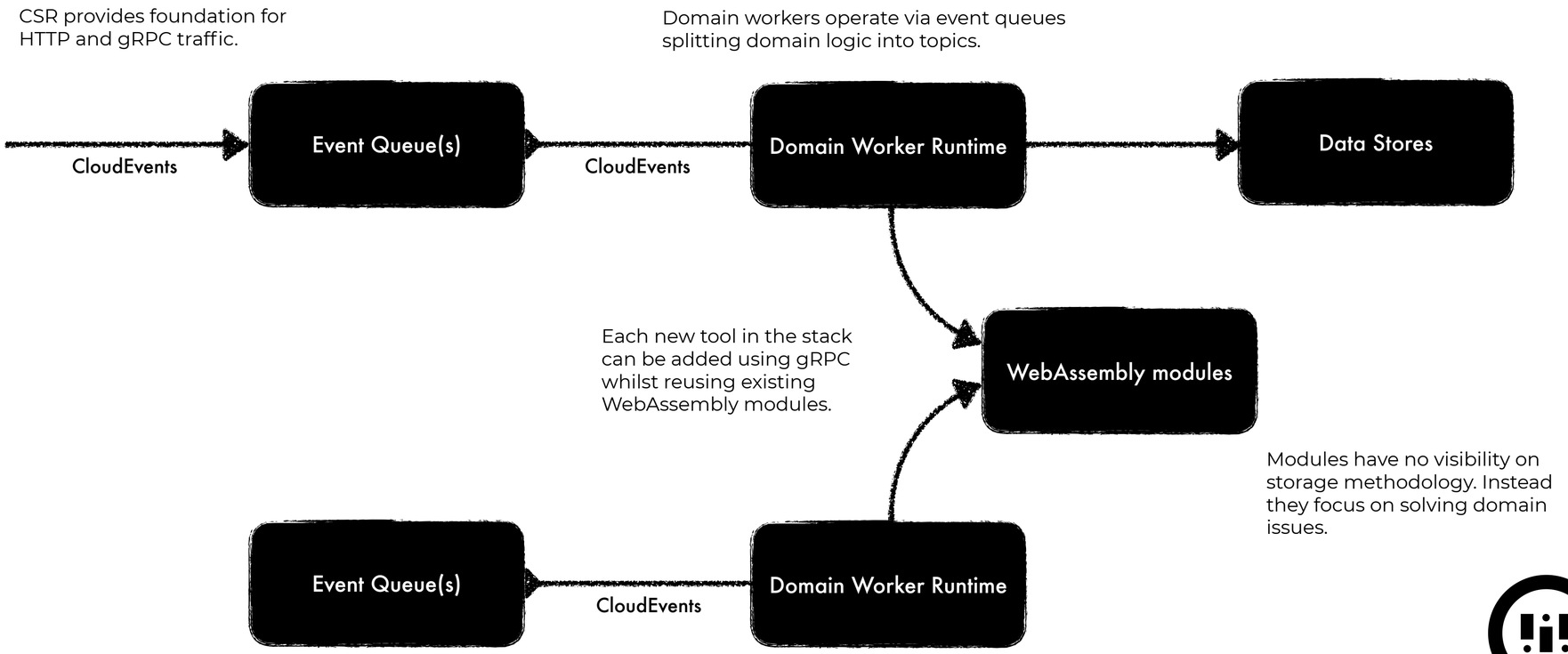
Domain workers operate via event queues splitting domain logic into topics.



Modules have no visibility on storage methodology. Instead they focus on solving domain issues.

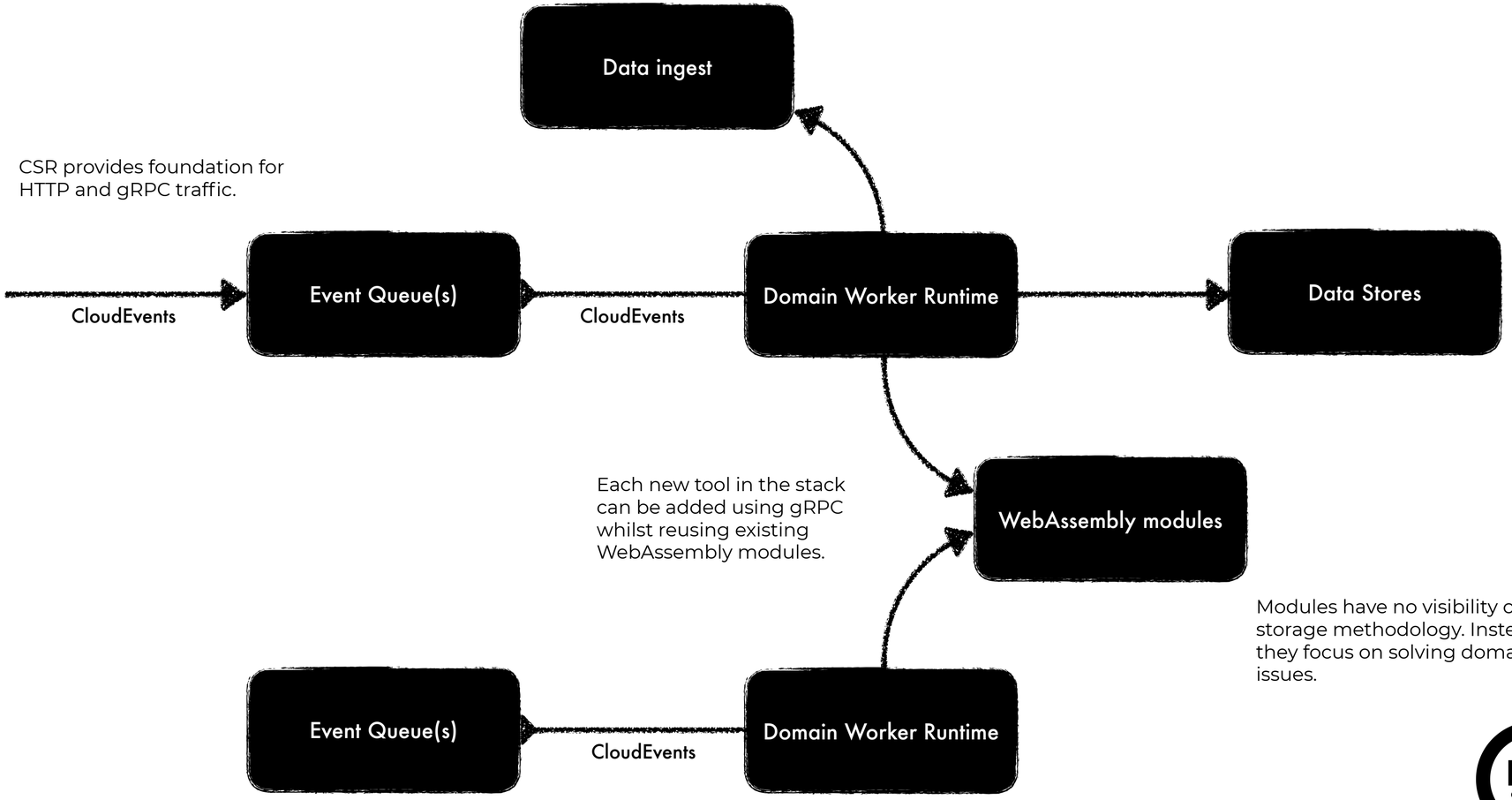
Modern Data Mesh Revisited

Domain-Driven Service Development



Modern Data Mesh Revisited

Domain-Driven Service Development



Habanero 🌶️

Introducing a modular WebAssembly auth project

- Simple and small Wasmtime-based framework for portable auth services
- Distributed open source runtime over gRPC for separation and scale
- OAuth2, OIDC, WebID, and IndieAuth as WebAssembly extensions
- SMTP and SMS-based magic links as WebAssembly extensions

Habanero 🌶️

Introducing a modular WebAssembly auth project

- Simple and small Wasmtime-based framework for portable auth services
- Distributed open source runtime over gRPC for separation and scale
- OAuth2, OIDC, WebID, and IndieAuth as WebAssembly extensions
- SMTP and SMS-based magic links as WebAssembly extensions
- Runs as a module on Fastly’s Edge Compute service

Thanks!

futurice