



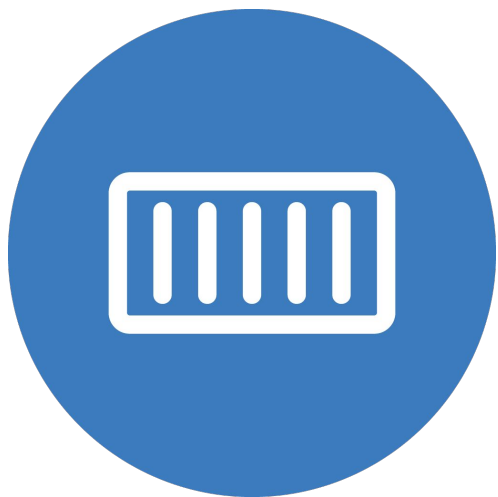
**PUZZLE ITC**  
changing IT for the better



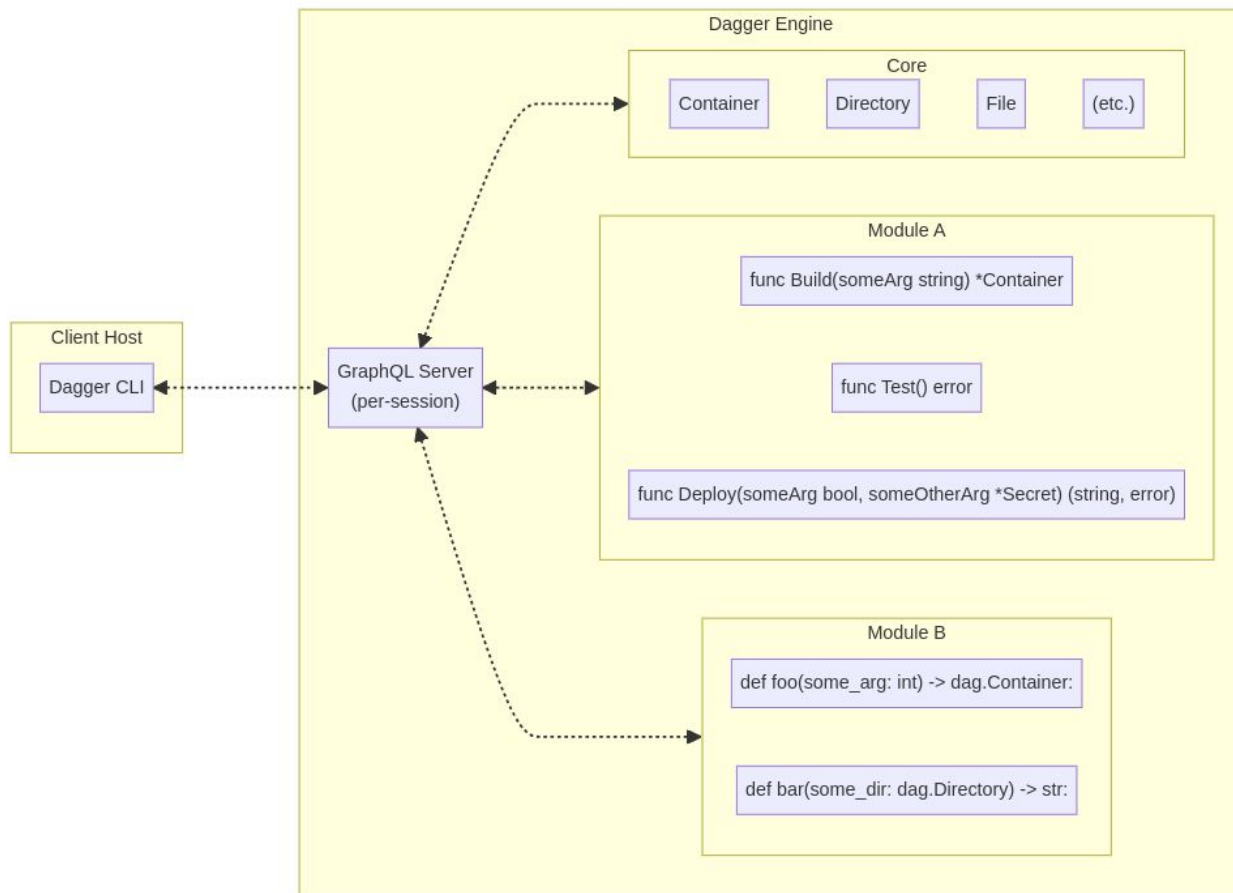
**Dagger**

# Dagger Concepts

# Solve current CI/CD problems



# Architecture



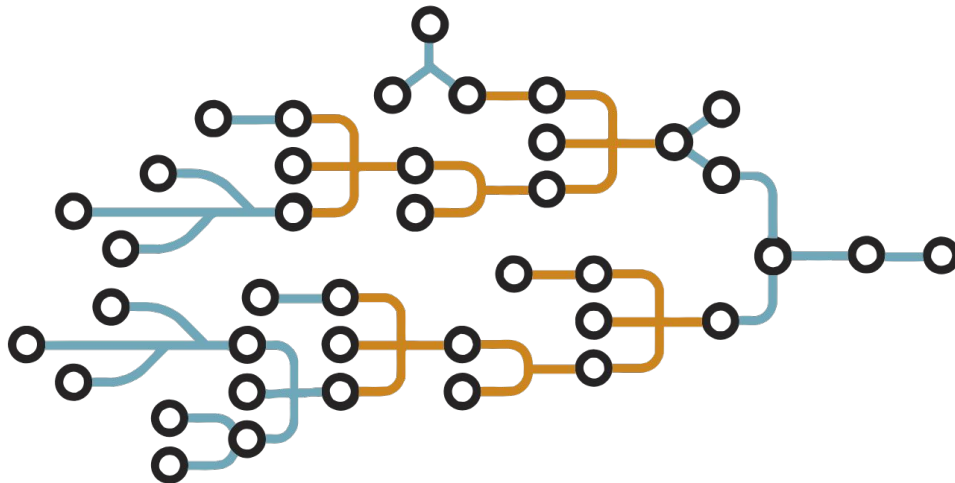
# How does Dagger really work?

Directed Acyclic Graph (**DAG**)  
--> DAGger 😊

## BuildKit

Low-Level Build (LLB) definition

Docker layer caching



# Execution Environment

## Containerization:

- Reproducibility
- Caching
- Security

## Host resources:

- Files and directories
- Environment variables
- Local network services

## Code execution:

- Language functions
- Language Libraries
- Dagger API
- Dagger Module functions

# Caching

Local / Single node:  
High degree of caching

Distributed / ephemeral runners:  
Caching possible

Dagger caches two types of data:

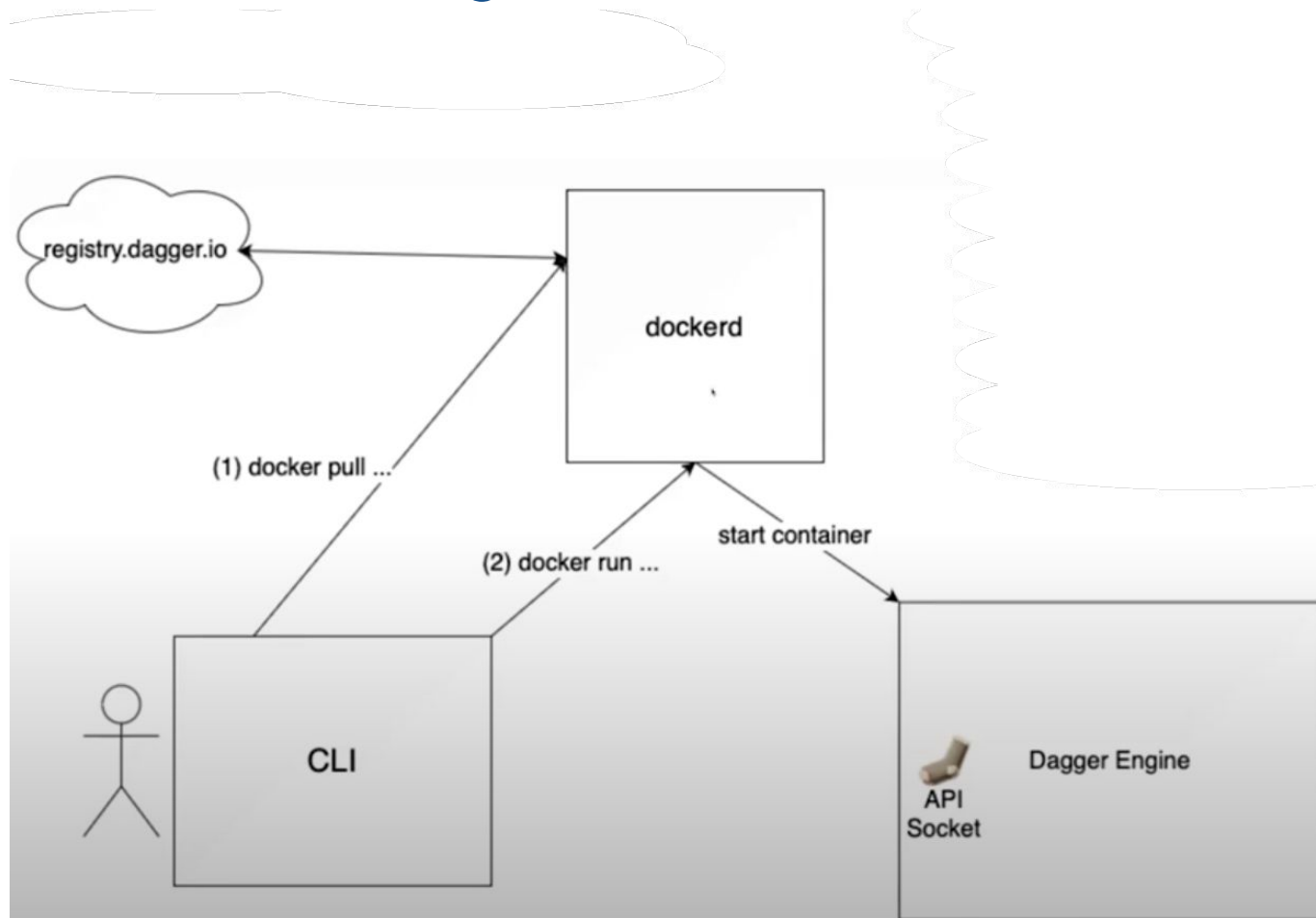
- Layers
- Volumes:

see [Cache Volumes](#)

Under the hood

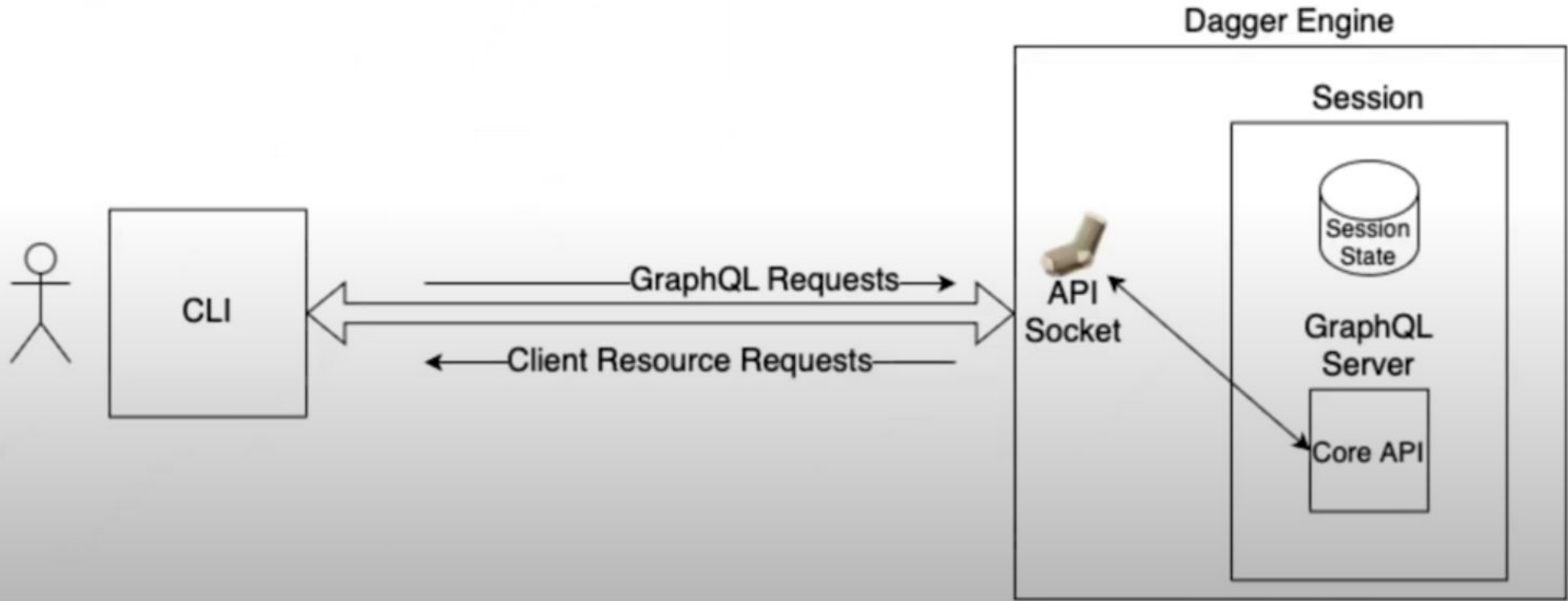


# Engine Provisioning

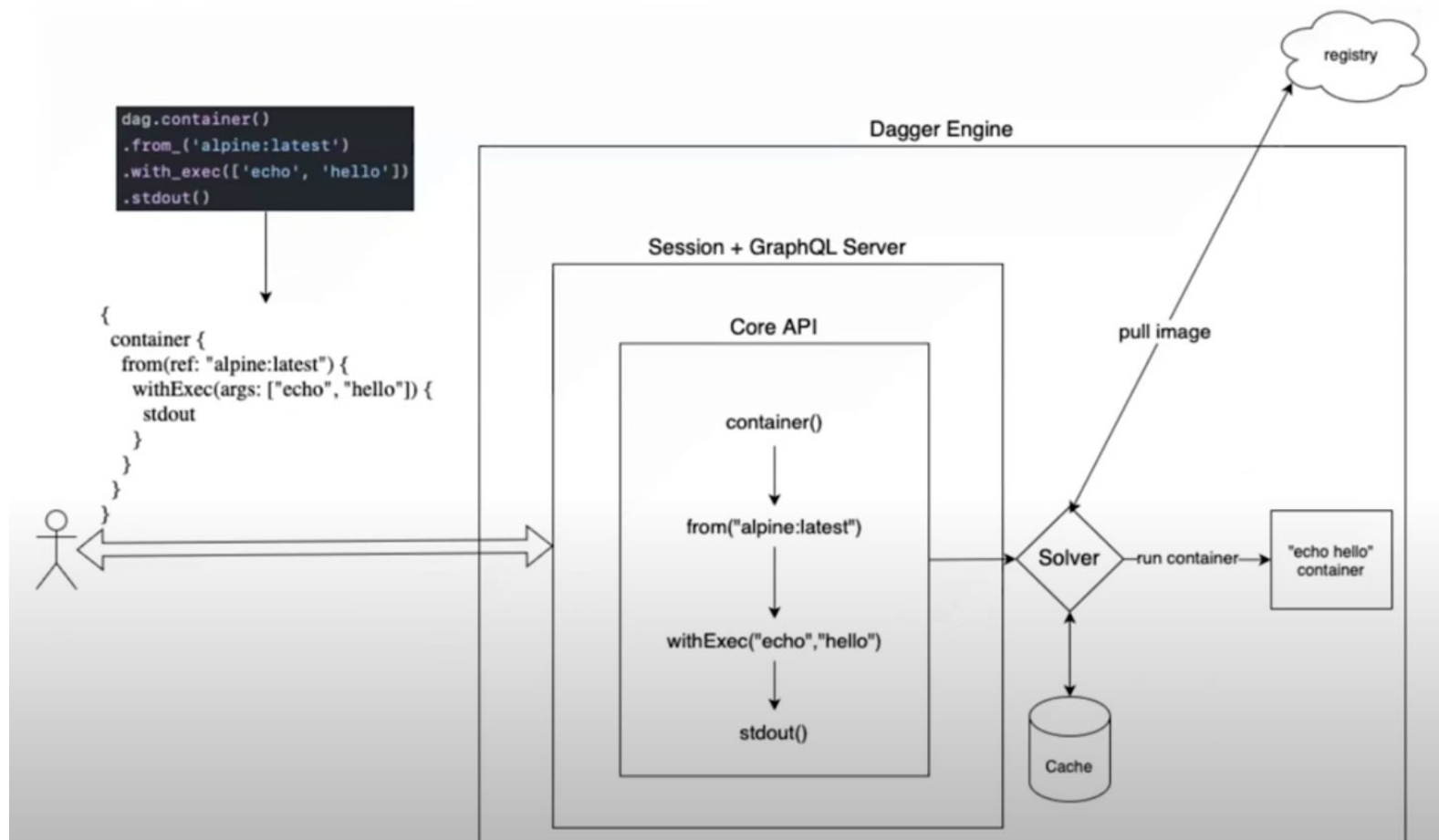




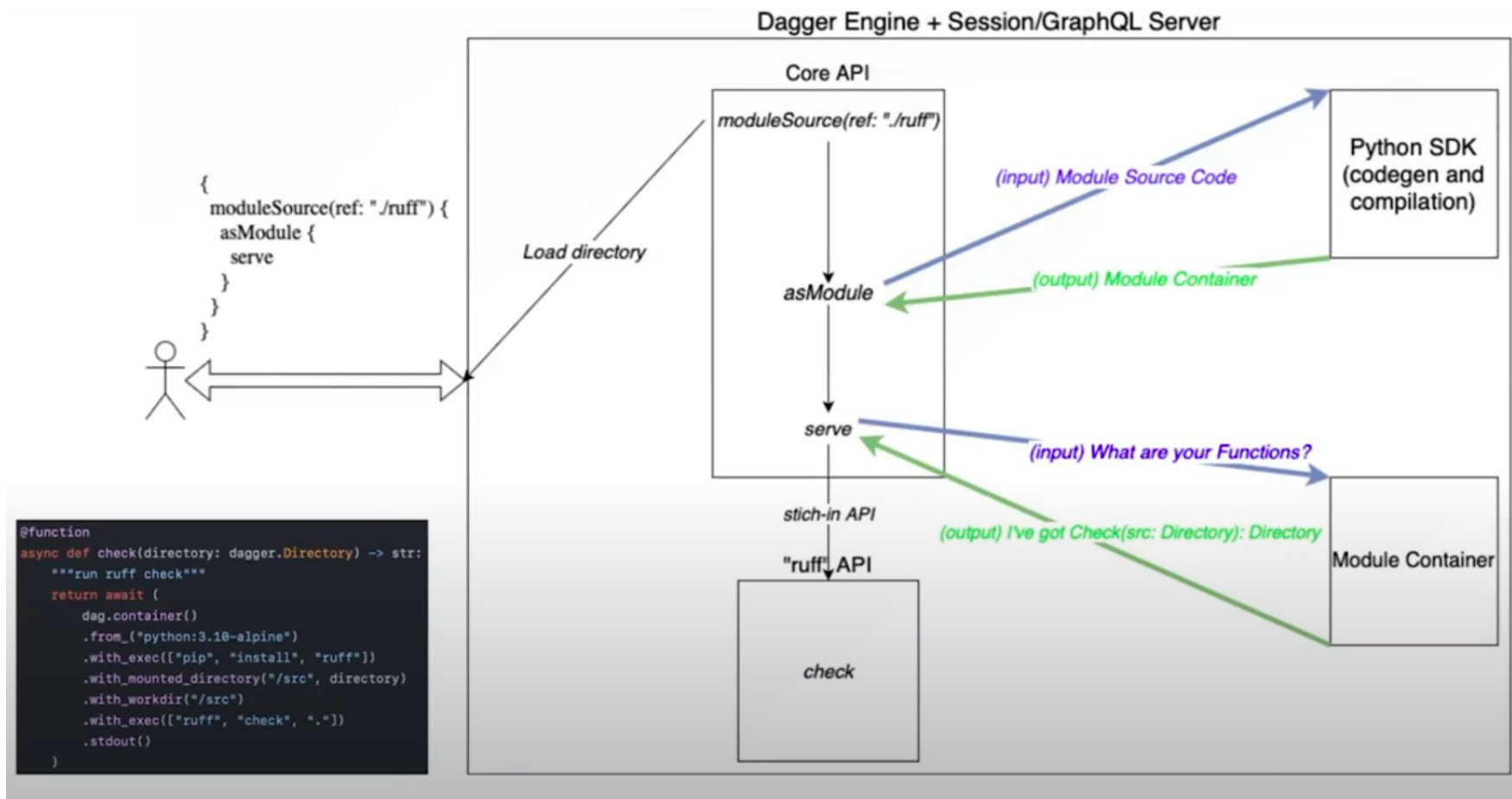
# Session + GraphQL Server



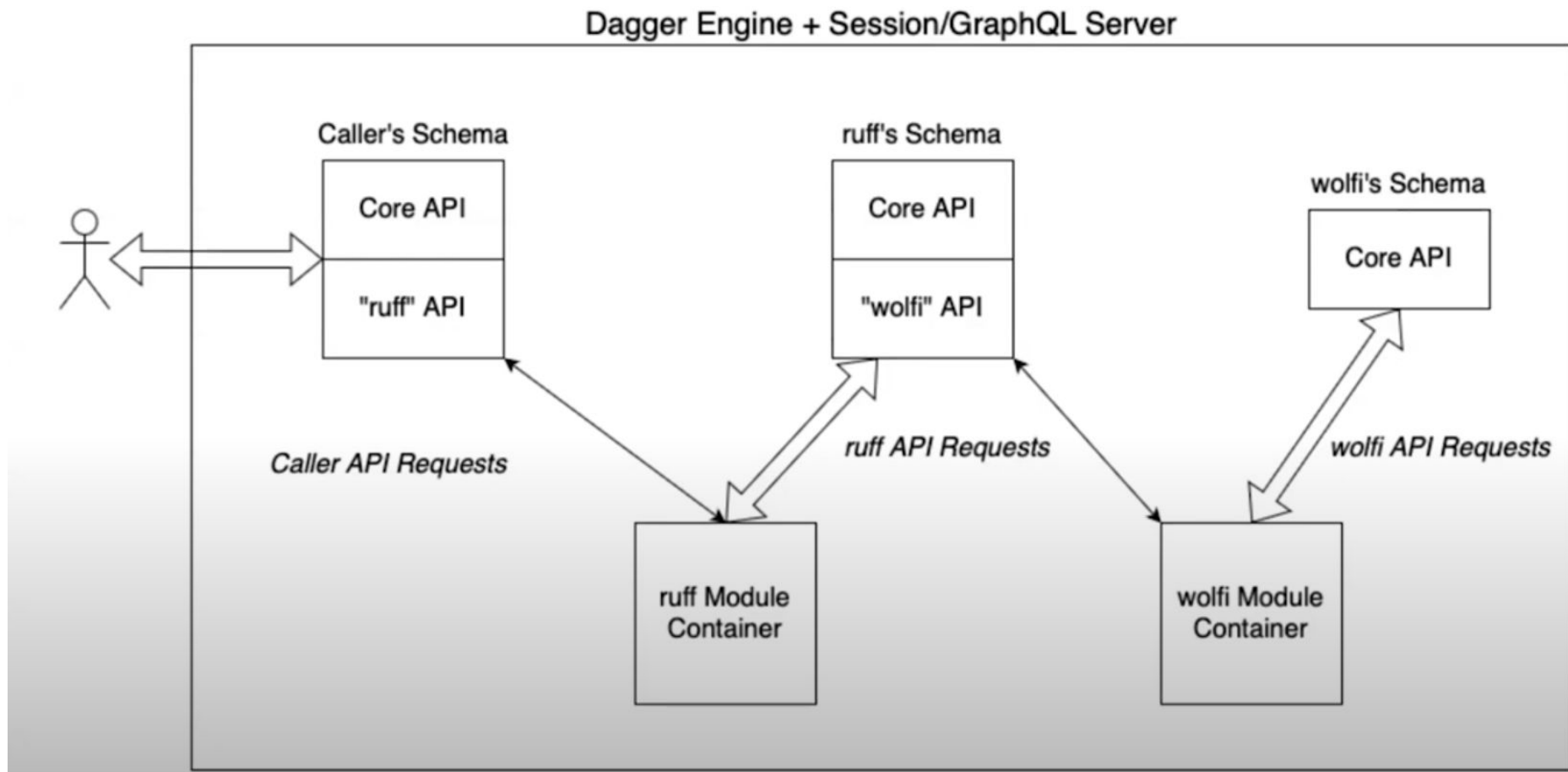
# Core API Calls



# Module API Calls - Loading



# Module API Calls - Loading (recursively)



# An actual (example) call

