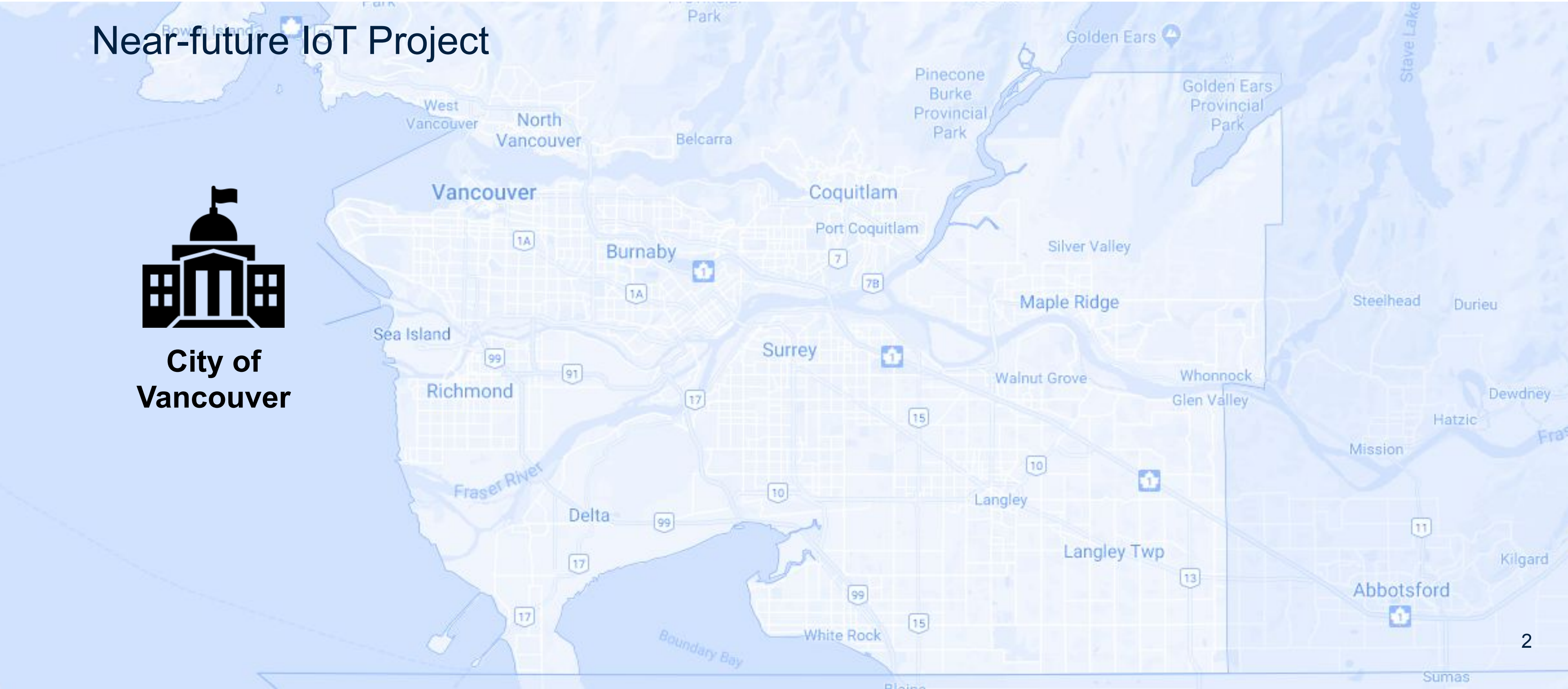
OneOS: IoT Platform based on POSIX and Actors

🎤 Kumseok Jung, UBC
Julien Gascon-Samson, ÉTS Montréal
Karthik Pattabiraman, UBC

Motivation

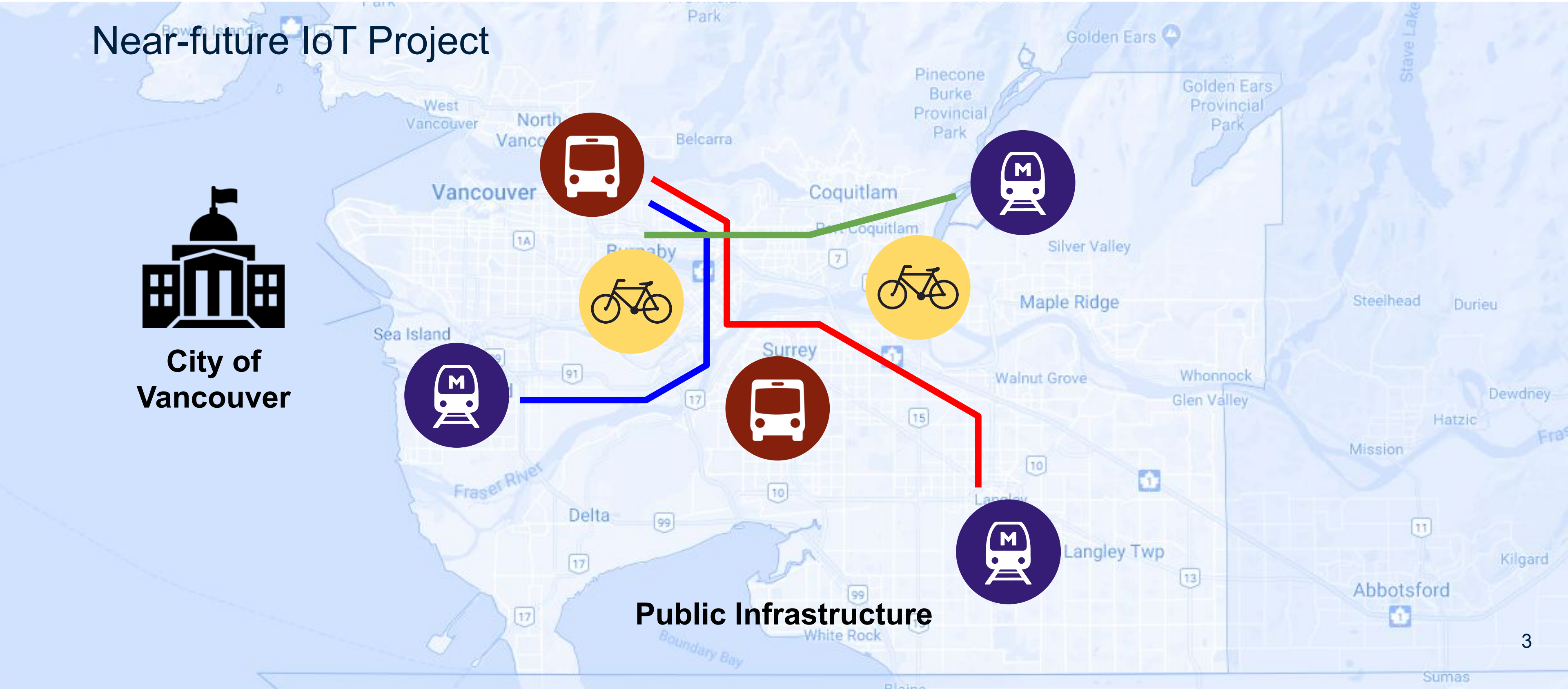
Near-future IoT Project



City of Vancouver

Motivation

Near-future IoT Project

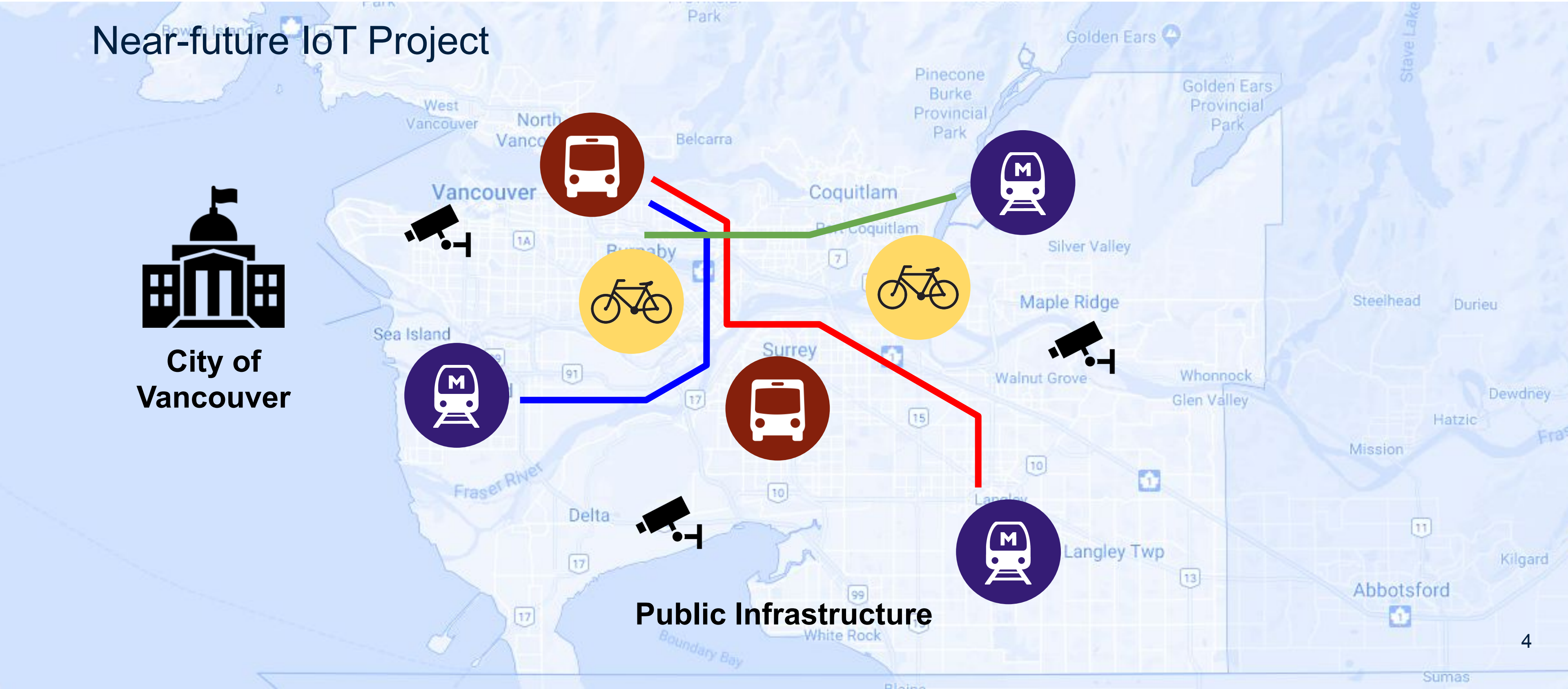


City of Vancouver

Public Infrastructure

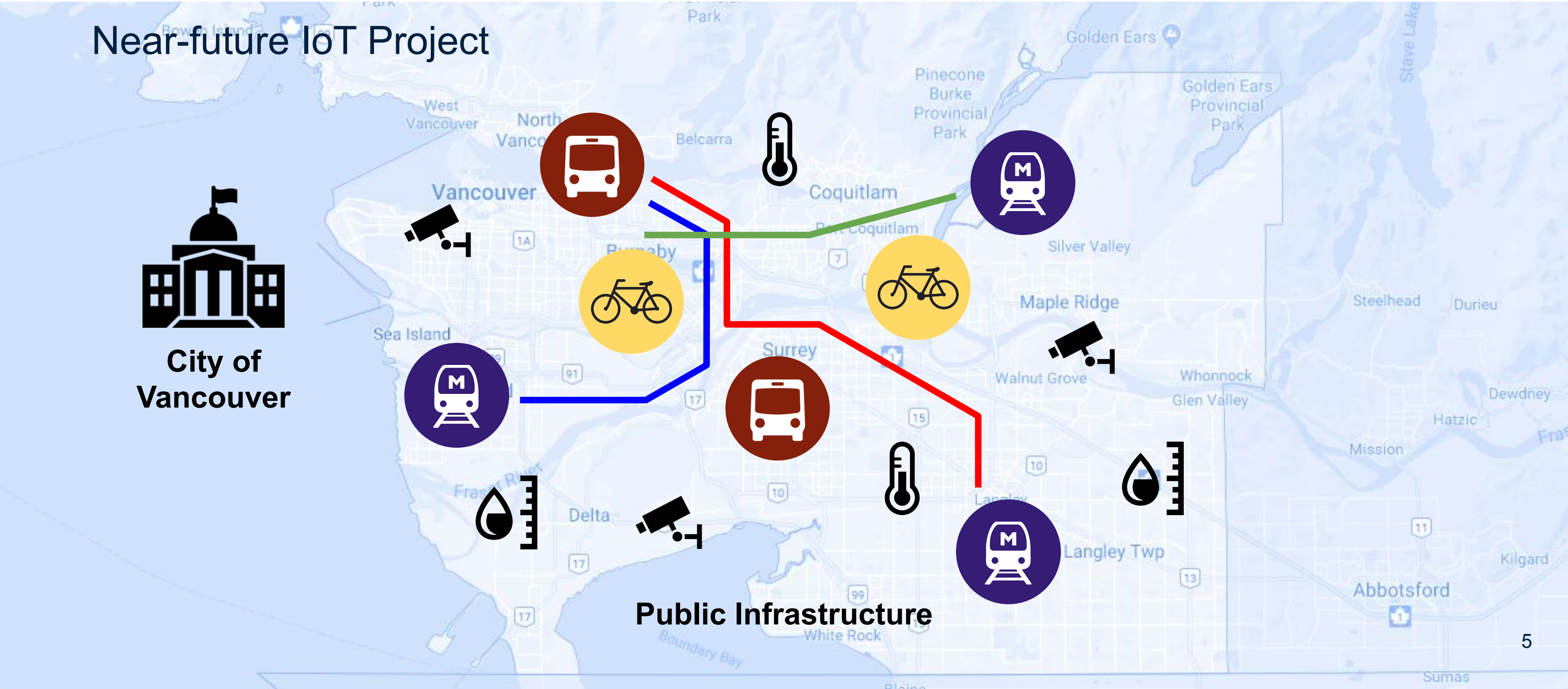
Motivation

Near-future IoT Project



Motivation

Near-future IoT Project

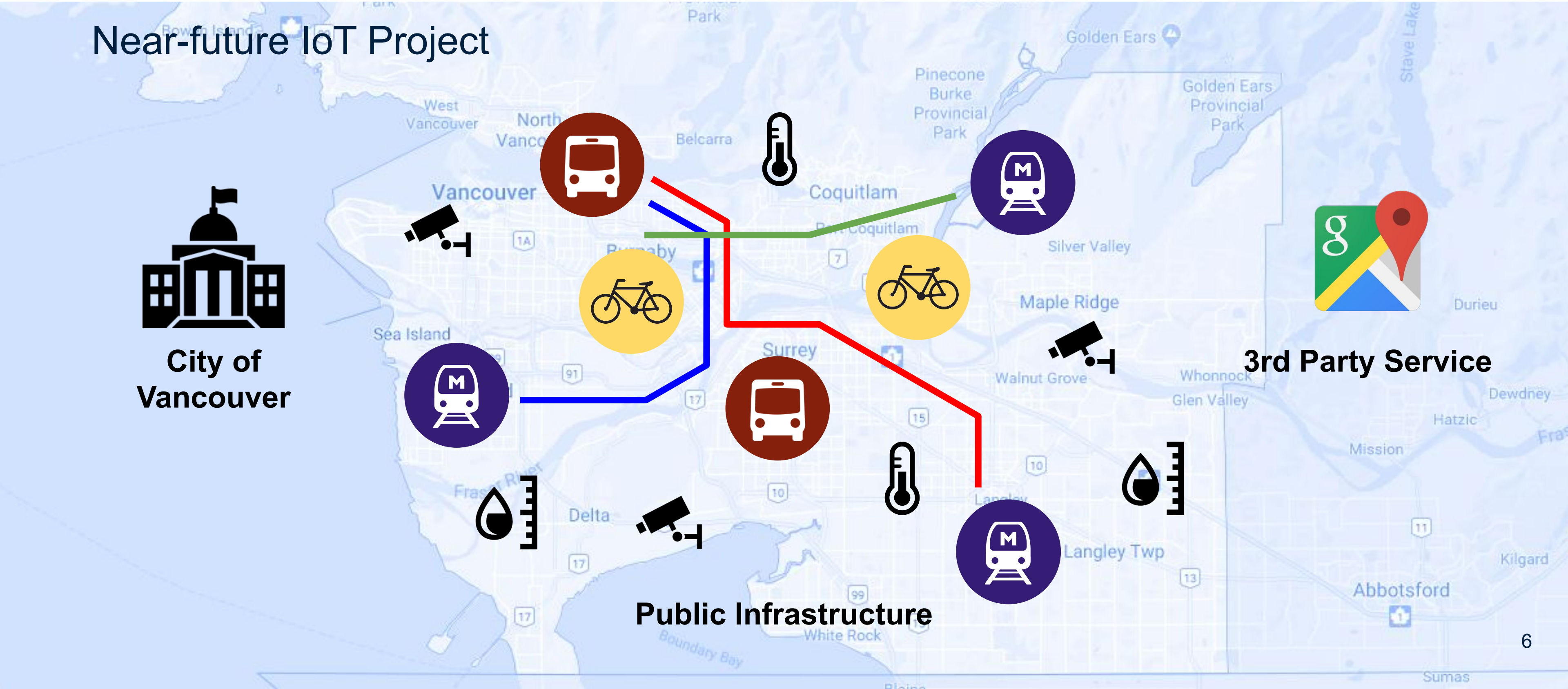


City of Vancouver

Public Infrastructure

Motivation

Near-future IoT Project



City of Vancouver

3rd Party Service

Public Infrastructure

Motivation

City of Vancouver

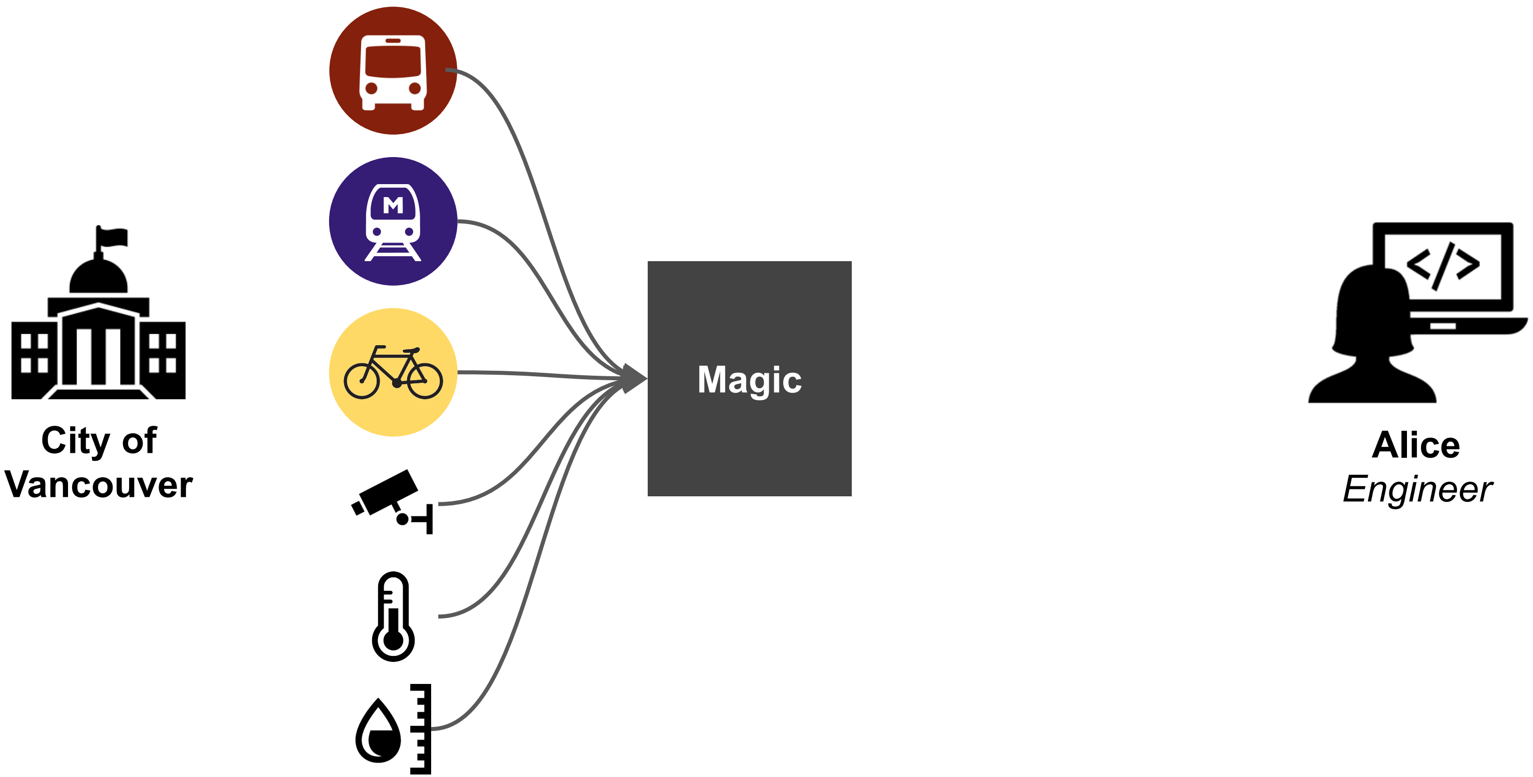
Alice
Engineer

Motivation

City of Vancouver

Alice
Engineer

Motivation

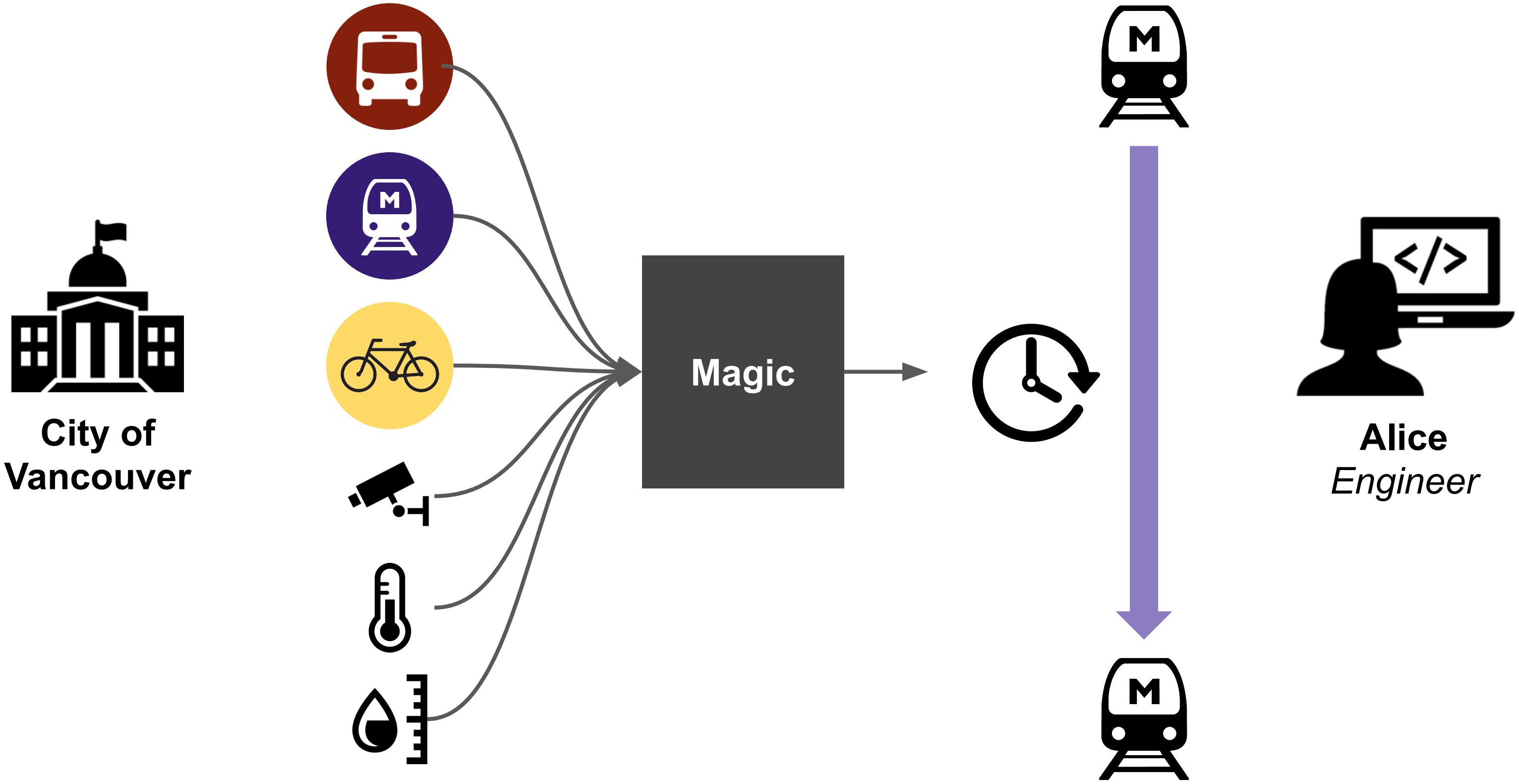


City of Vancouver

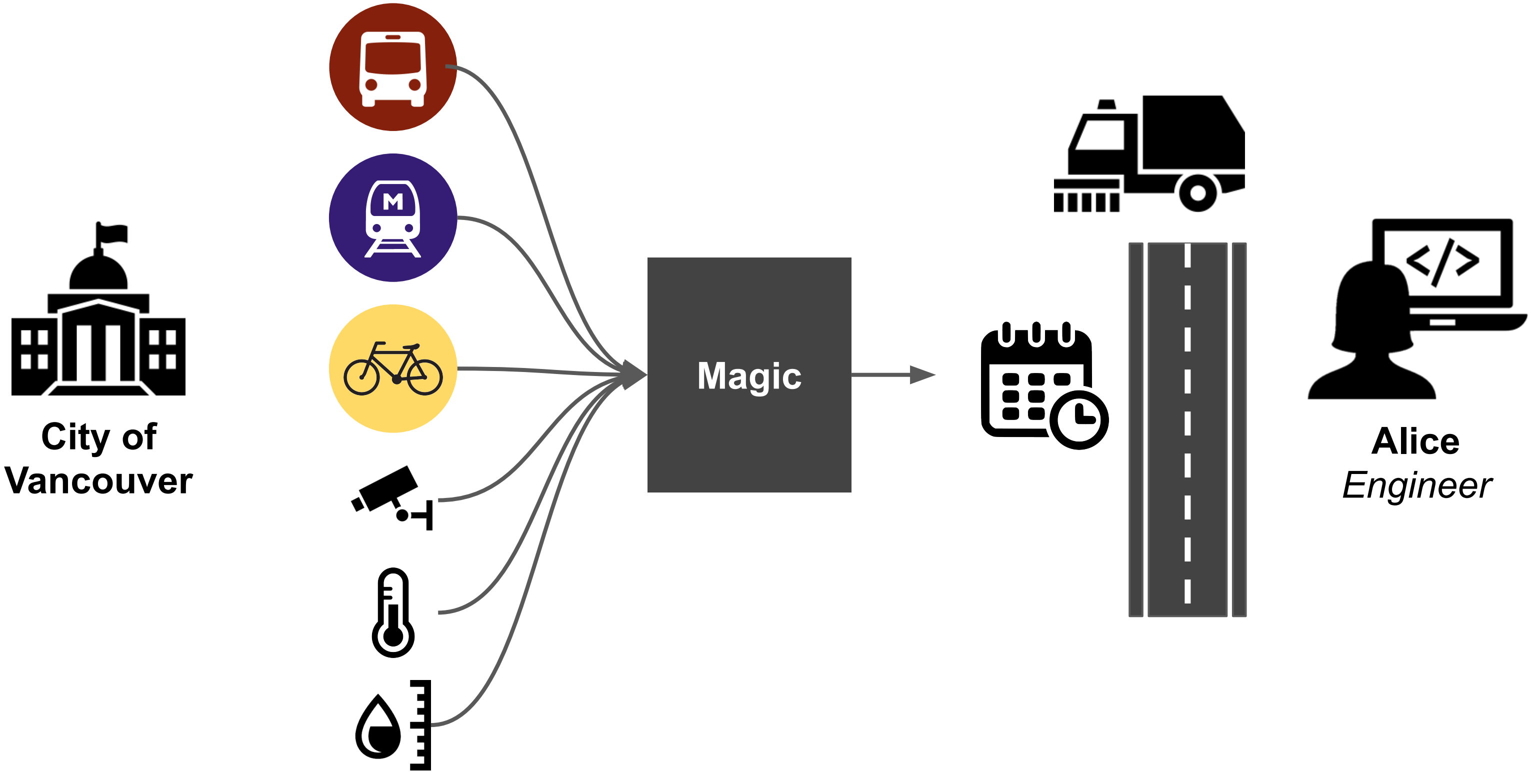
Magic

Alice
Engineer

Motivation



Motivation

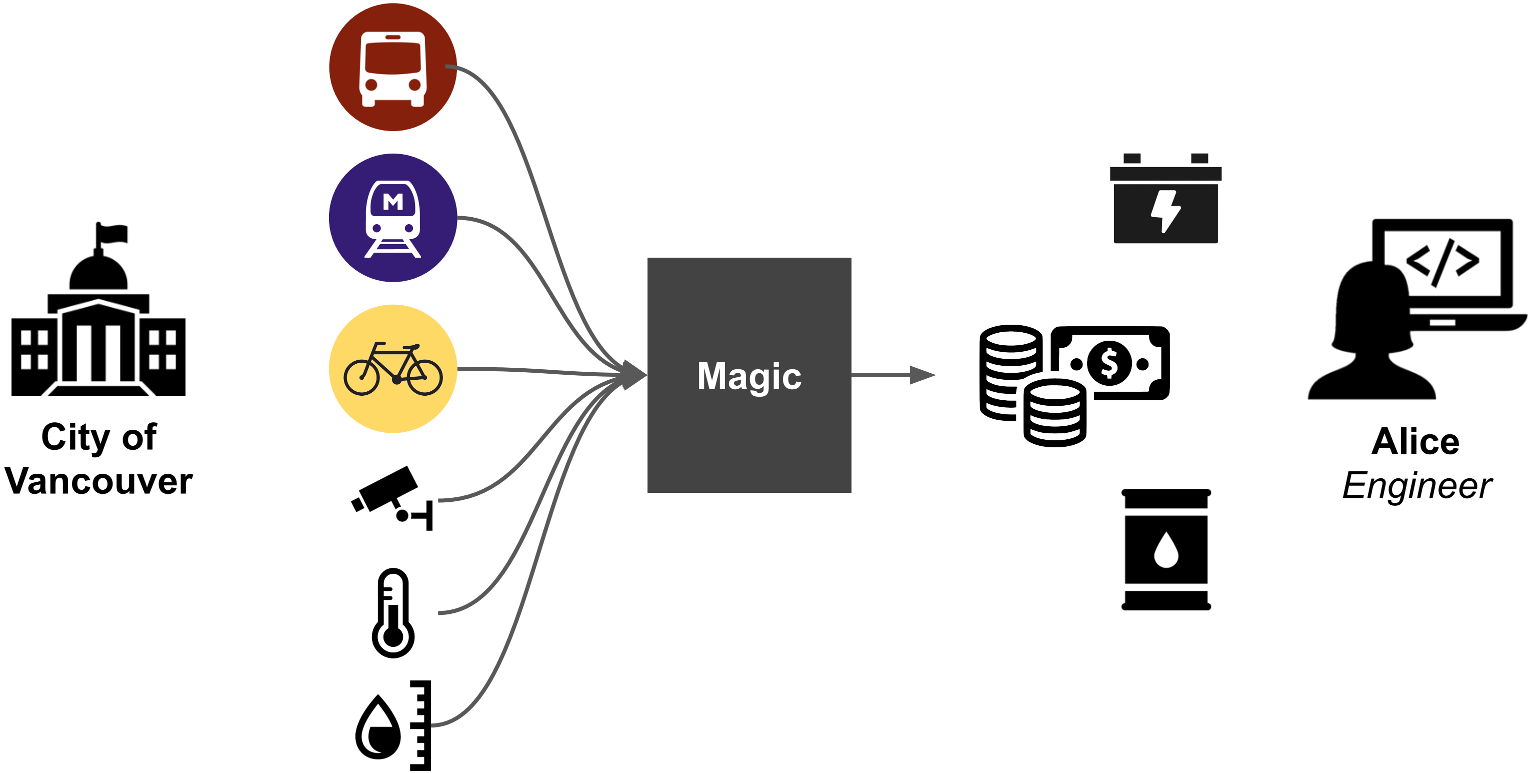


City of Vancouver

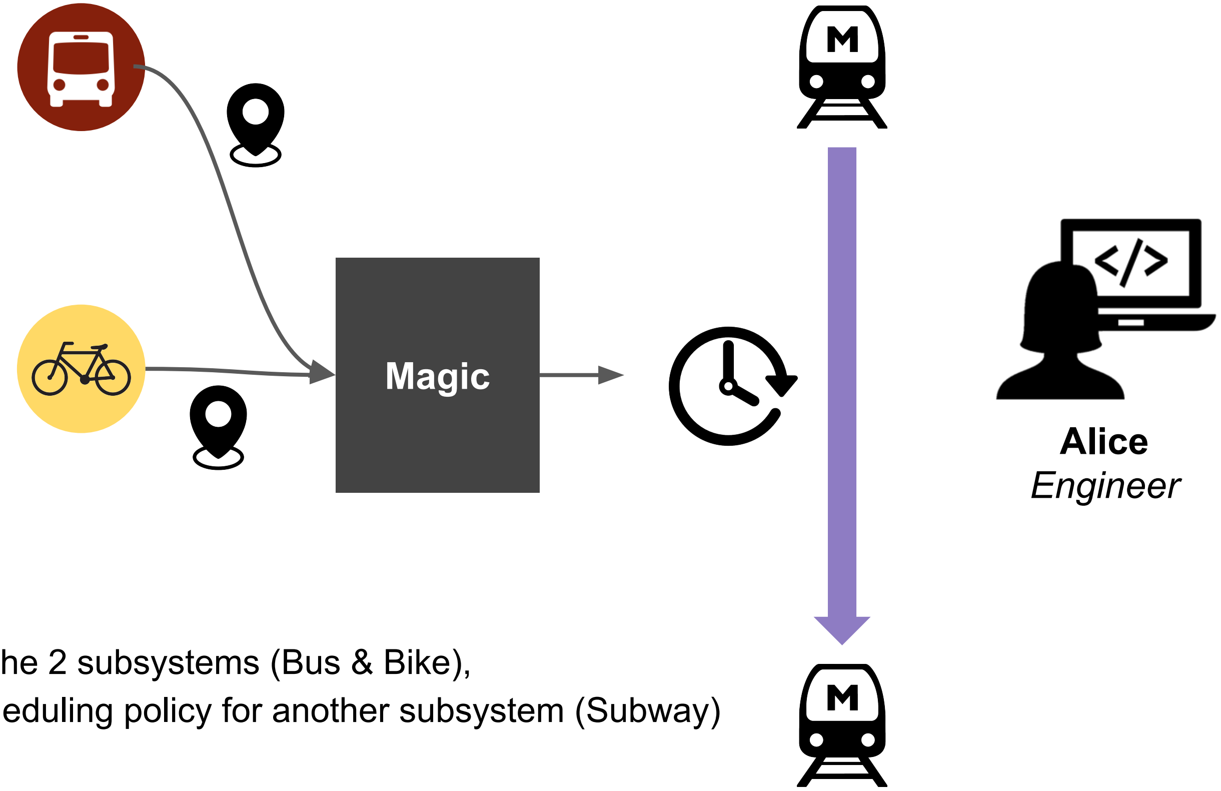
Magic

Alice
Engineer

Motivation



Challenges



Example Problem:

Collect GPS data from the 2 subsystems (Bus & Bike),
compute an optimal scheduling policy for another subsystem (Subway)

Challenges: Heterogeneity

Company A 🚌

Company B 🚲

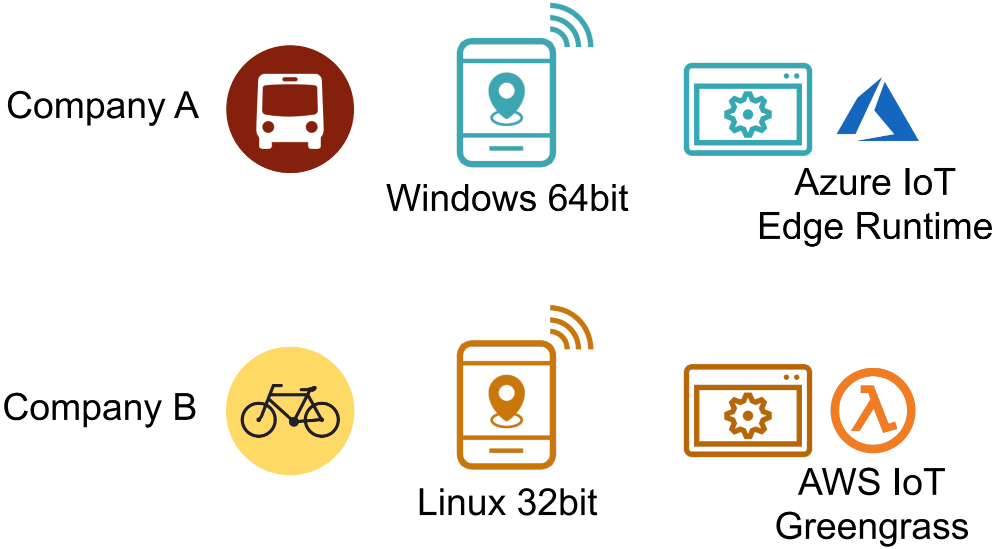
Challenges: Heterogeneity

Company A 🚌 📱 Windows 64bit

Company B 🚲 📱 Linux 32bit

Heterogeneity in Hardware and Operating Systems

Challenges: Heterogeneity



Company A

Windows 64bit

Azure IoT
Edge Runtime

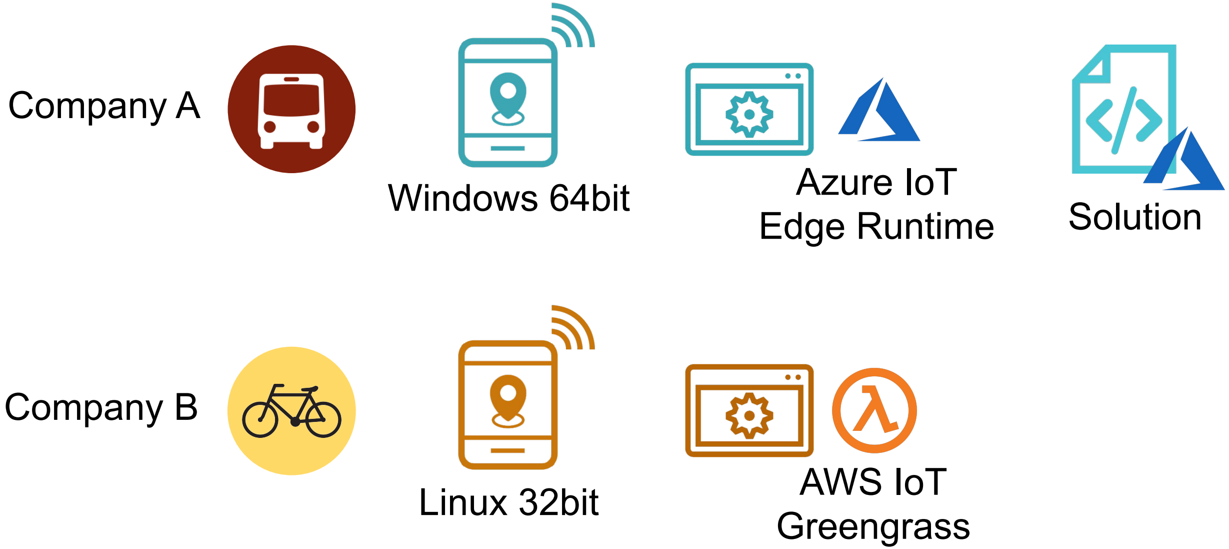
Company B

Linux 32bit

AWS IoT
Greengrass

Heterogeneity in Languages and Frameworks

Challenges: Heterogeneity



Company A

Windows 64bit

Azure IoT Edge Runtime

Solution

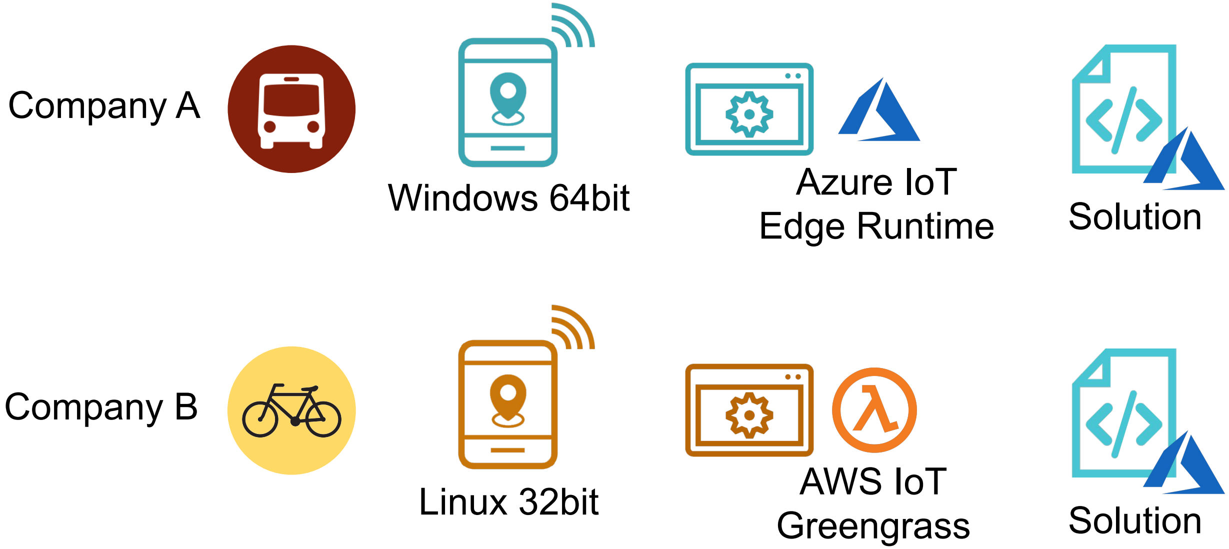
Company B

Linux 32bit

AWS IoT Greengrass

Software within a **specific framework** is **not portable** across different frameworks

Challenges: Heterogeneity



Company A

Windows 64bit

Azure IoT Edge Runtime

Solution

Company B

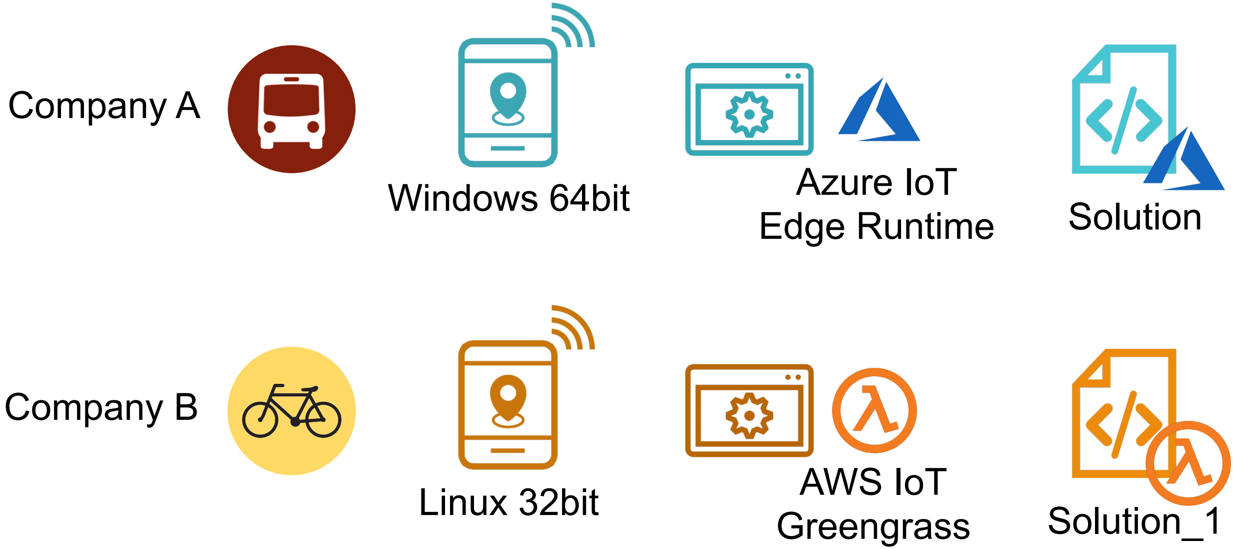
Linux 32bit

AWS IoT Greengrass

Solution

Software within a **specific framework** is **not portable** across different frameworks

Challenges: Heterogeneity



Company A

Windows 64bit

Azure IoT
Edge Runtime

Solution

Company B

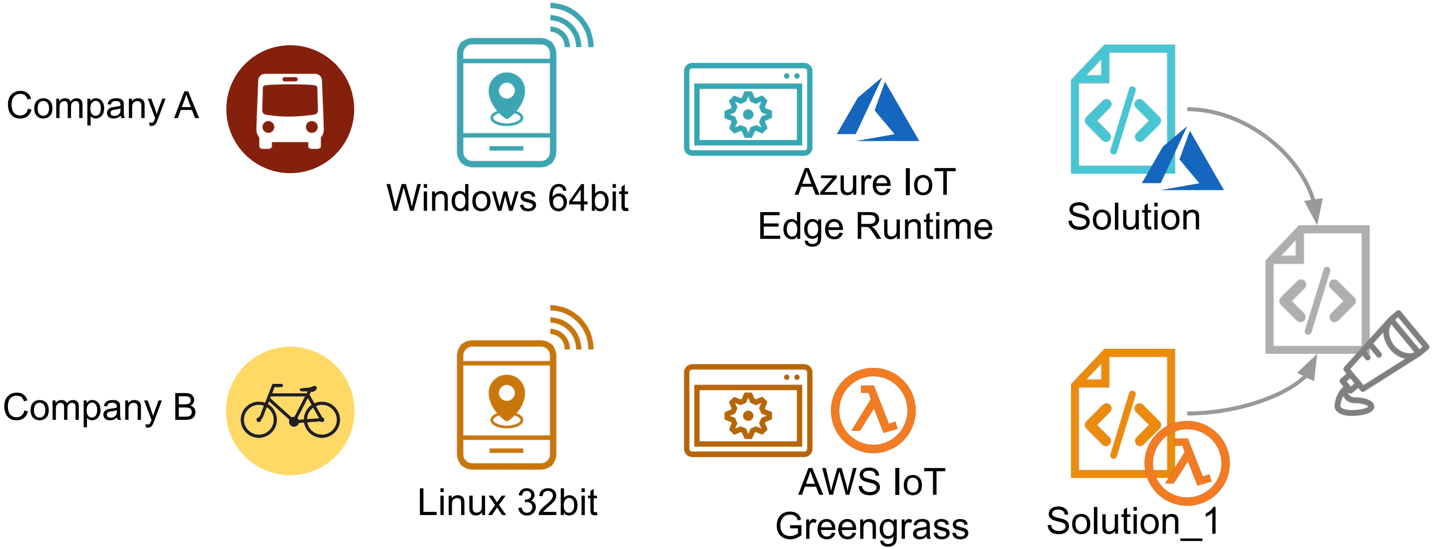
Linux 32bit

AWS IoT
Greengrass

Solution_1

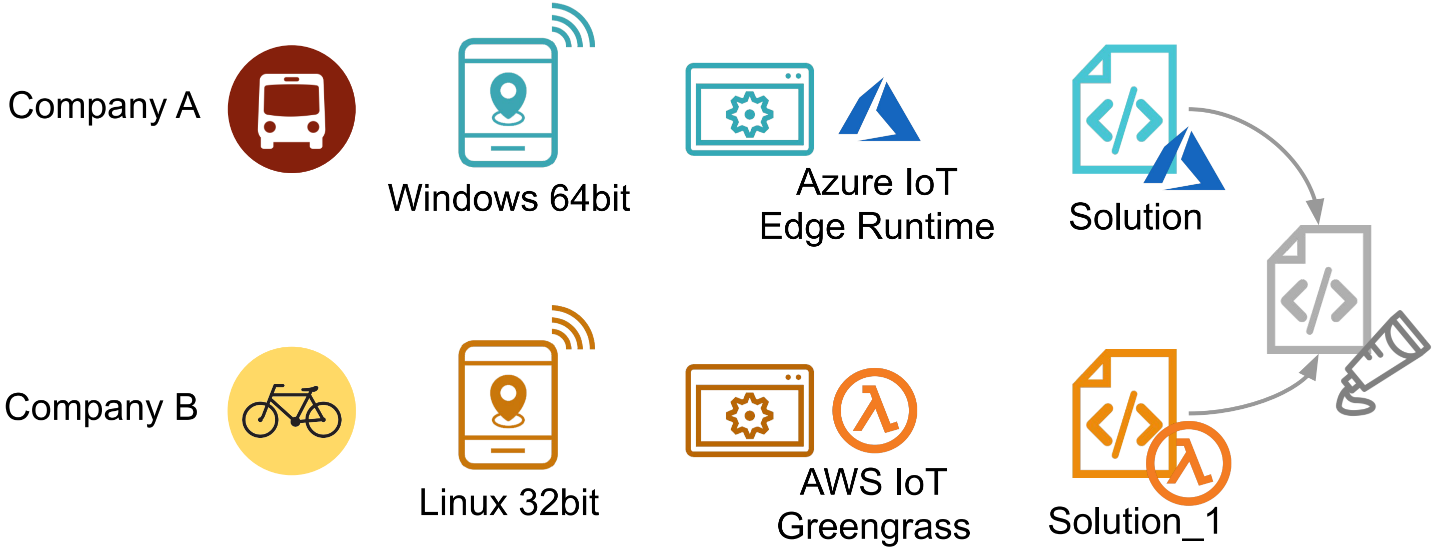
Need to rewrite the **same application logic**
for **different frameworks**

Challenges: Heterogeneity



Difference in application semantics
resolved by **more glue software**

Challenges: Heterogeneity



Company A

Windows 64bit

Azure IoT Edge Runtime

Solution

Company B

Linux 32bit

AWS IoT Greengrass

Solution_1

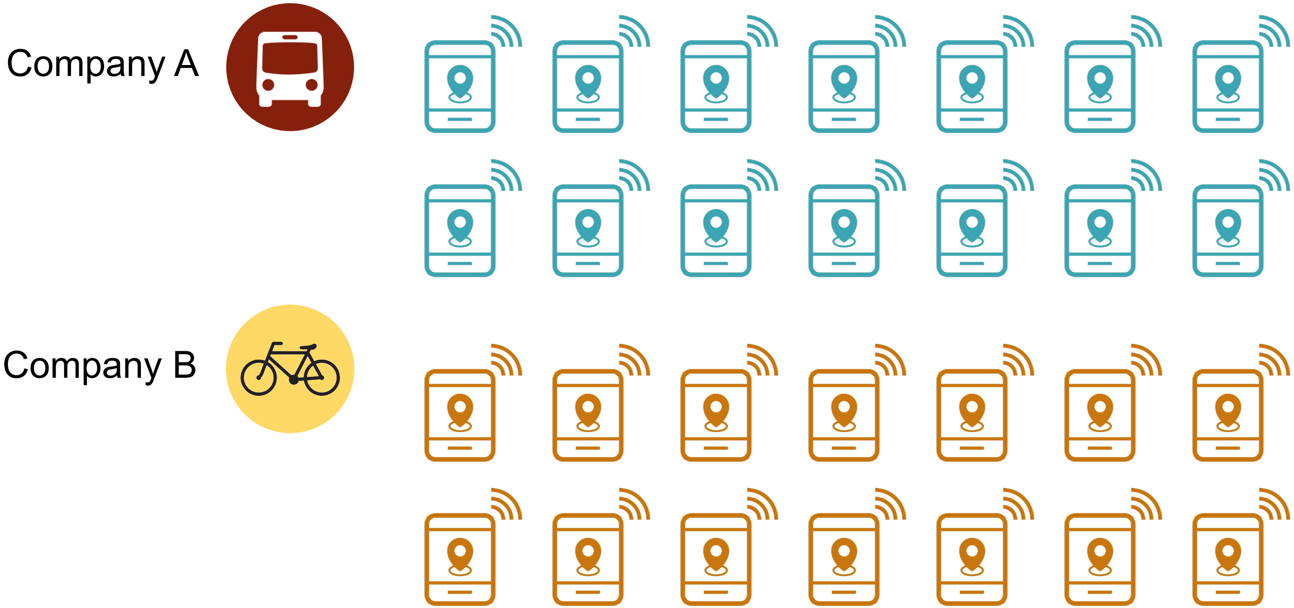
We end up with: **Heterogeneity** in Application Software

Challenges: Scale & Dynamicity

Company A

Company B

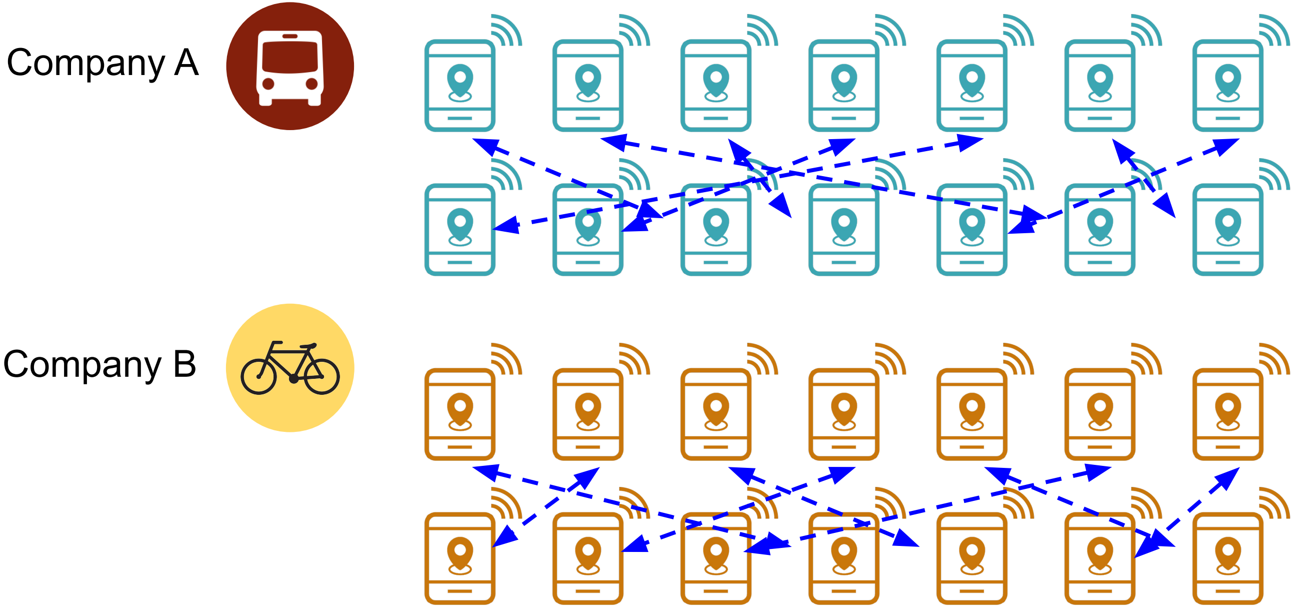
Challenges: Scale & Dynamicity



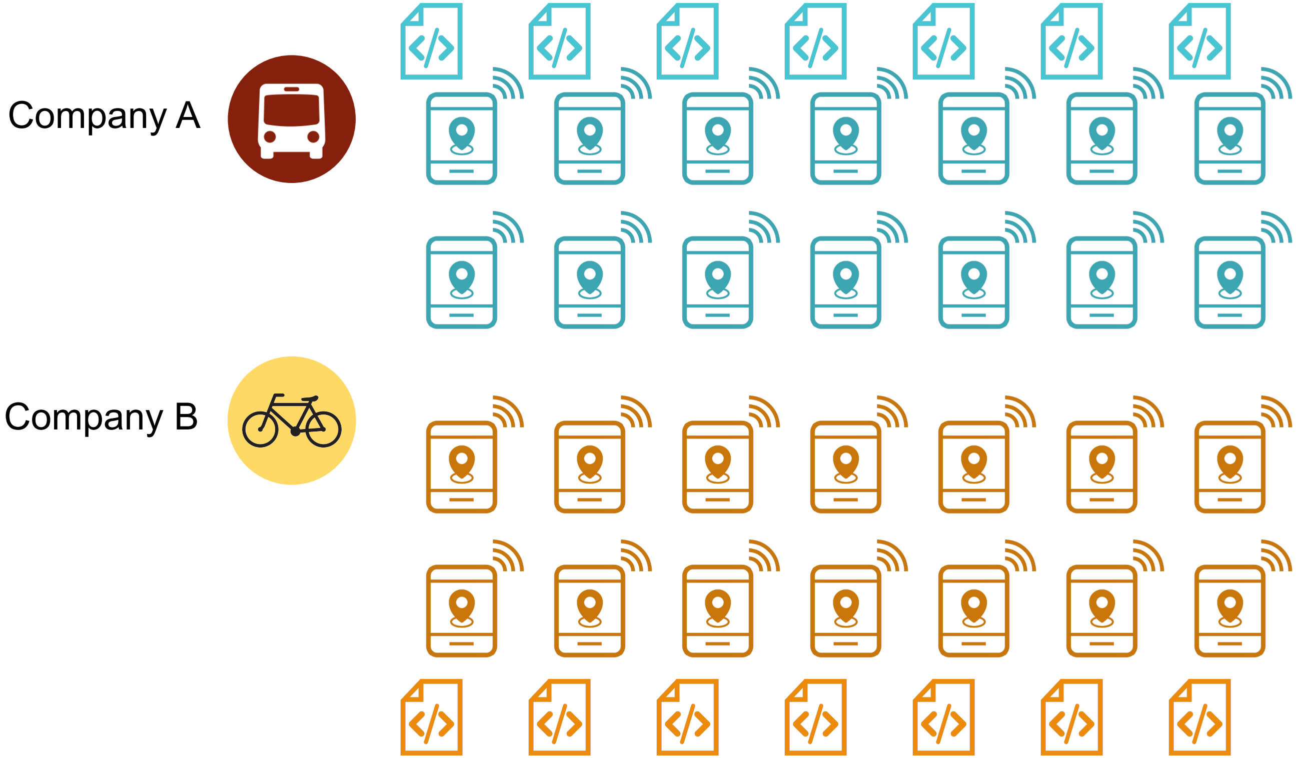
Company A

Company B

Challenges: Scale & Dynamicity



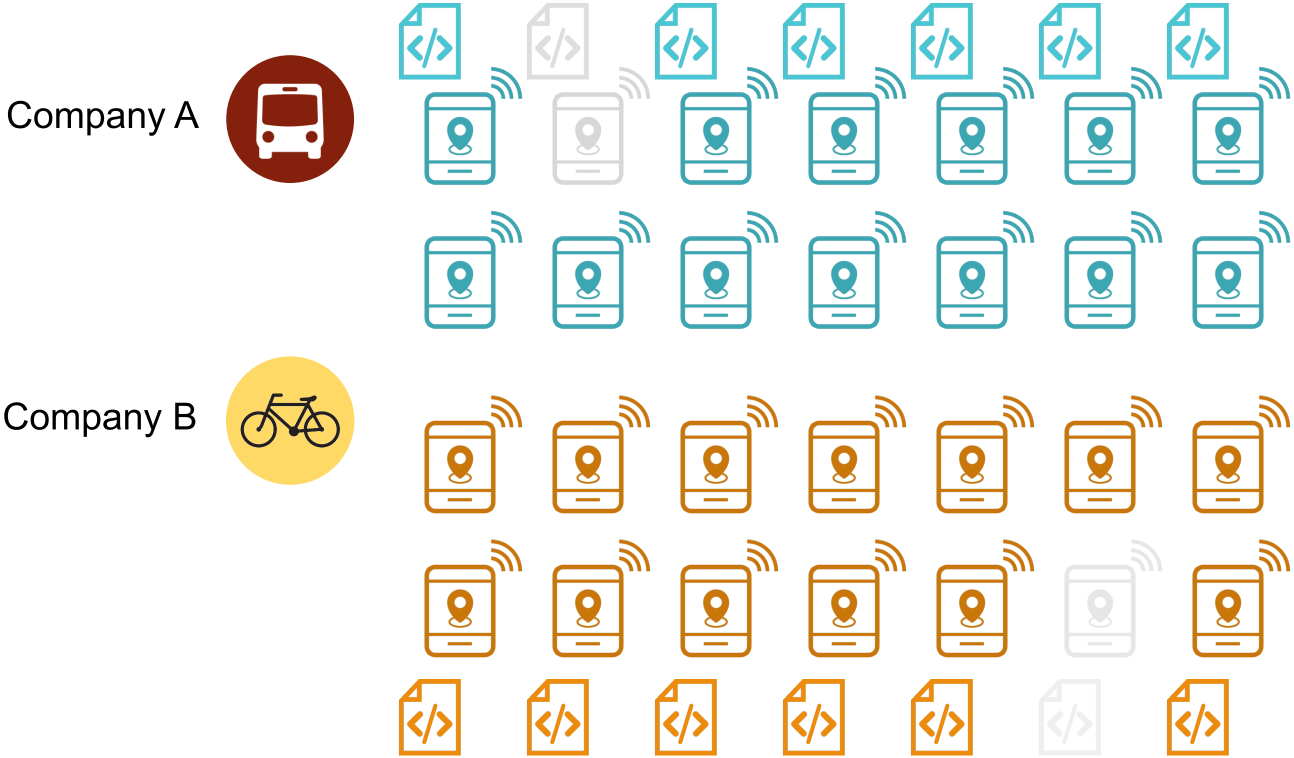
Challenges: Scale & Dynamicity



Company A

Company B

Challenges: Scale & Dynamicity



Company A

Company B

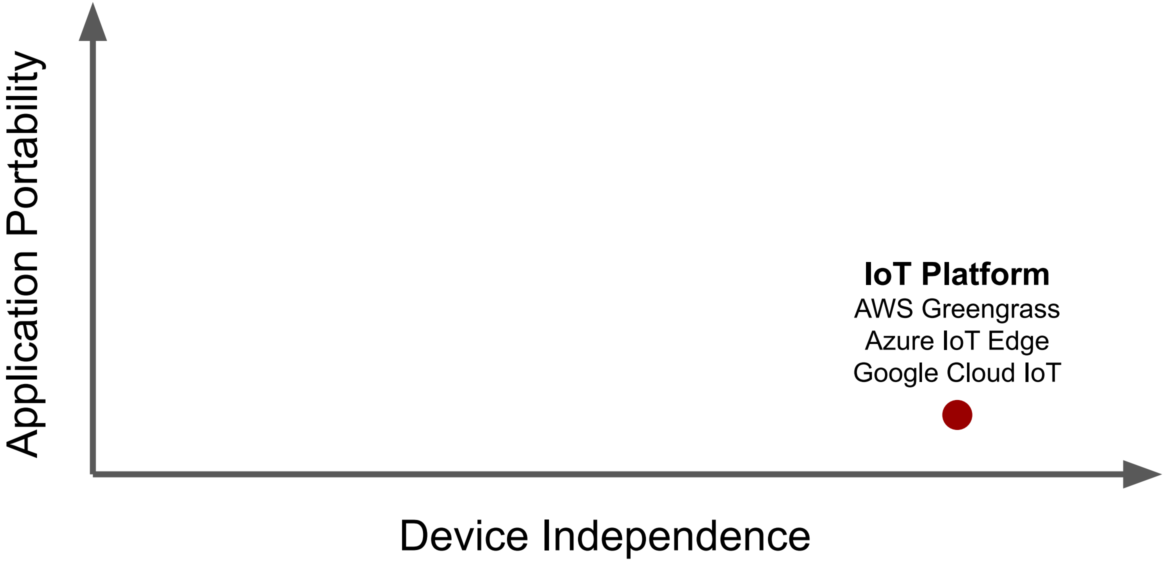
Challenges: Scale & Dynamicity

Company A

Company B

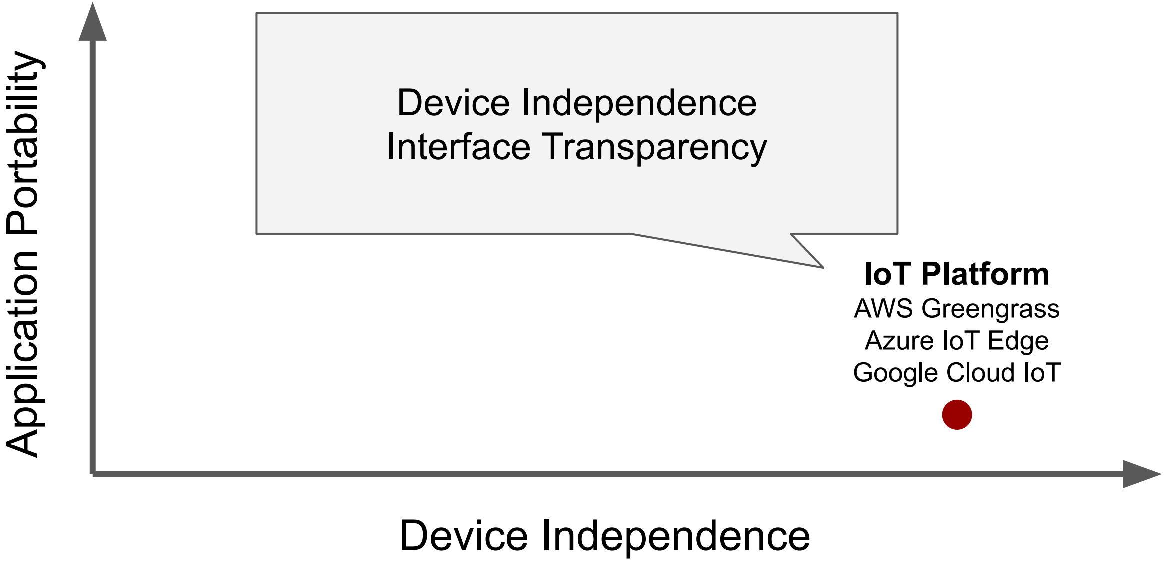
Related Work

Distributed Computing Platforms



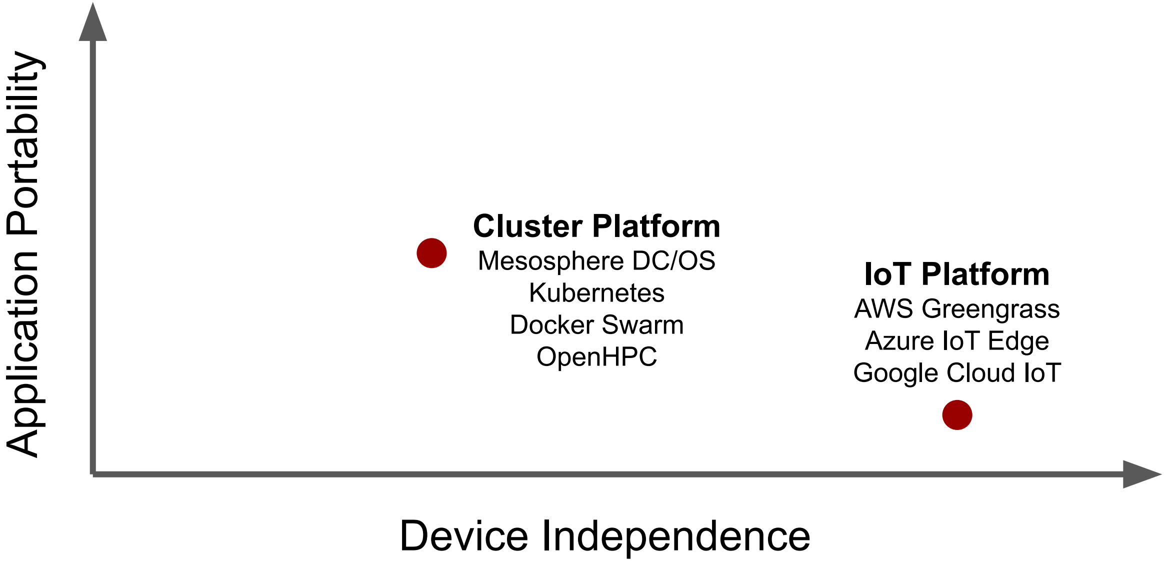
Related Work

Distributed Computing Platforms



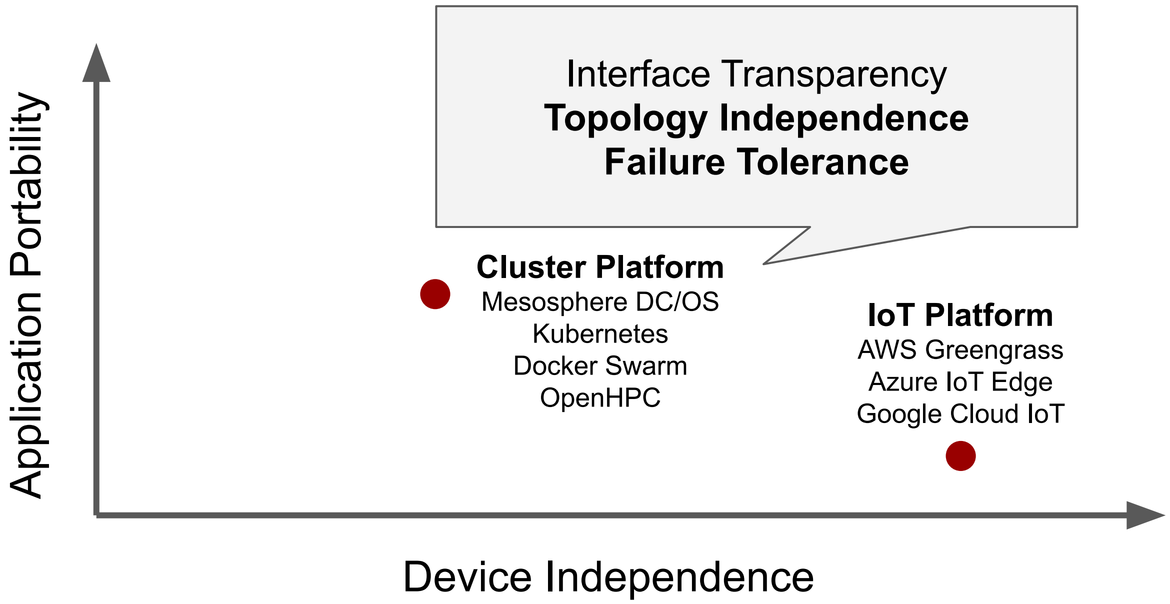
Related Work

Distributed Computing Platforms



Related Work

Distributed Computing Platforms



Interface Transparency
Topology Independence
Failure Tolerance

Cluster Platform
Mesosphere DC/OS
Kubernetes
Docker Swarm
OpenHPC

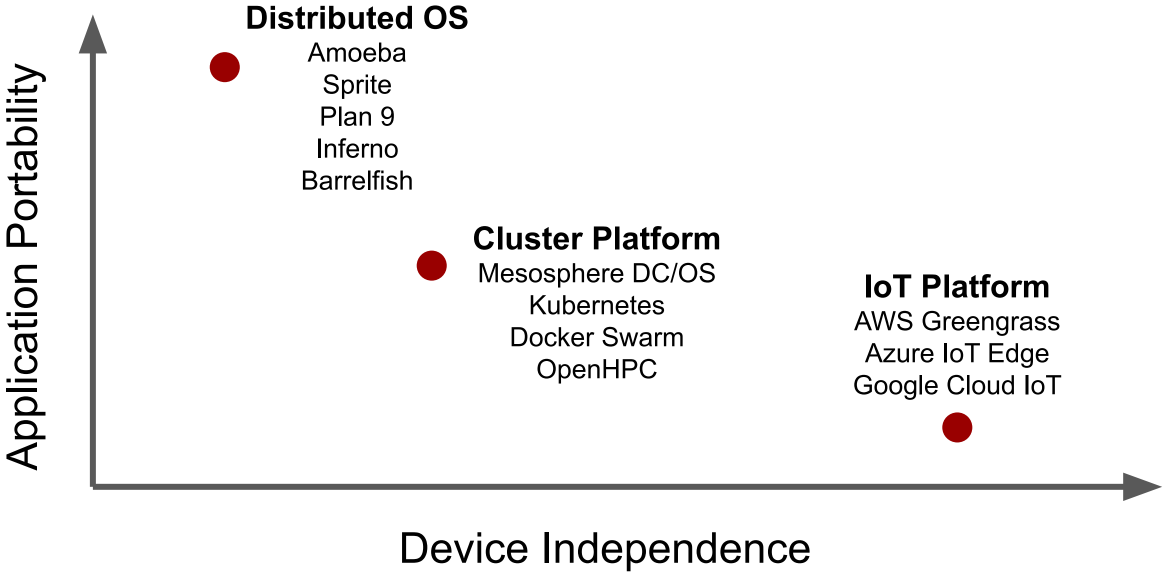
IoT Platform
AWS Greengrass
Azure IoT Edge
Google Cloud IoT

Application Portability

Device Independence

Related Work

Distributed Computing Platforms



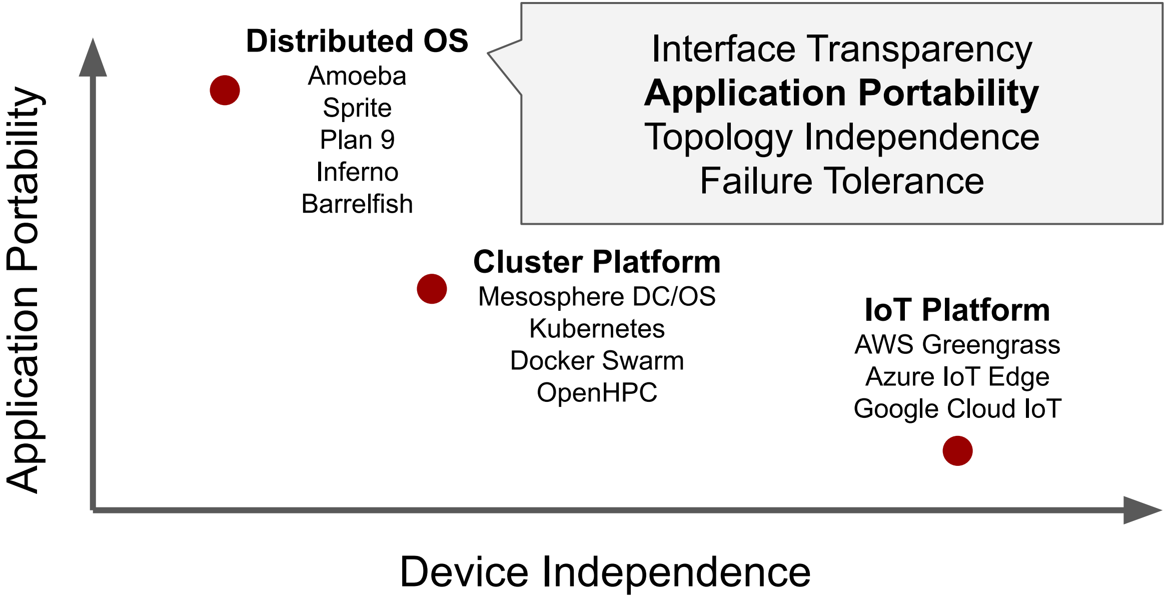
Distributed OS
Amoeba
Sprite
Plan 9
Inferno
Barrelfish

Cluster Platform
Mesosphere DC/OS
Kubernetes
Docker Swarm
OpenHPC

IoT Platform
AWS Greengrass
Azure IoT Edge
Google Cloud IoT

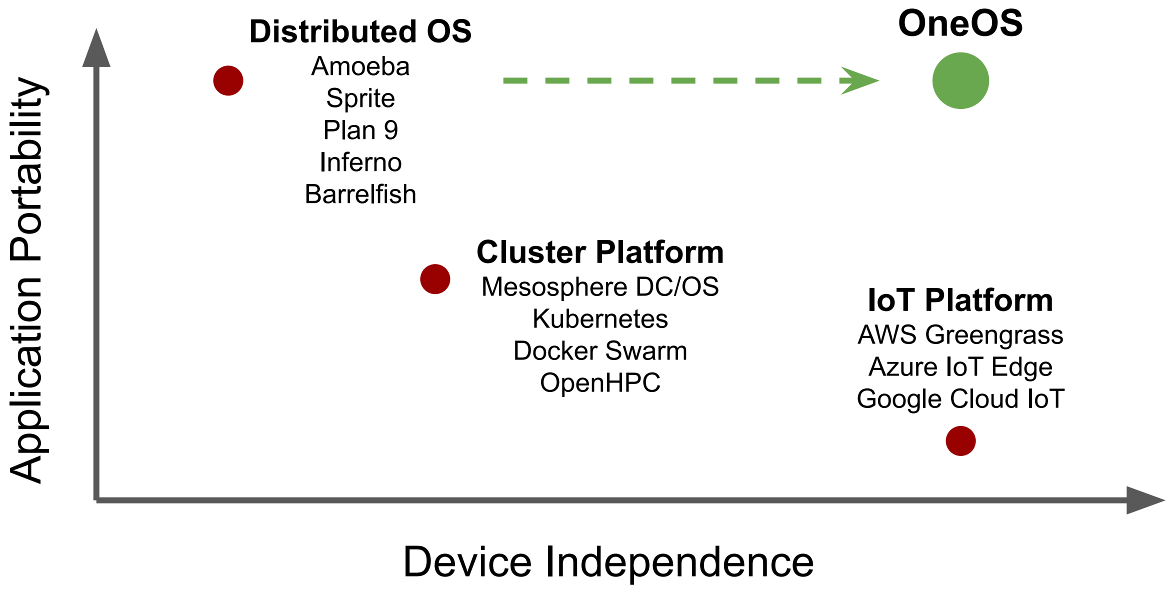
Related Work

Distributed Computing Platforms



Our Goal: OneOS

Distributed Computing Platforms



Distributed OS
Amoeba
Sprite
Plan 9
Inferno
Barrelfish

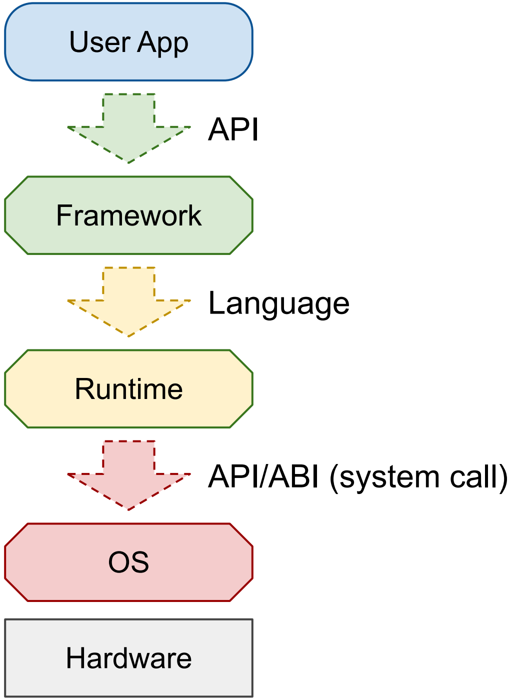
OneOS

Cluster Platform
Mesosphere DC/OS
Kubernetes
Docker Swarm
OpenHPC

IoT Platform
AWS Greengrass
Azure IoT Edge
Google Cloud IoT

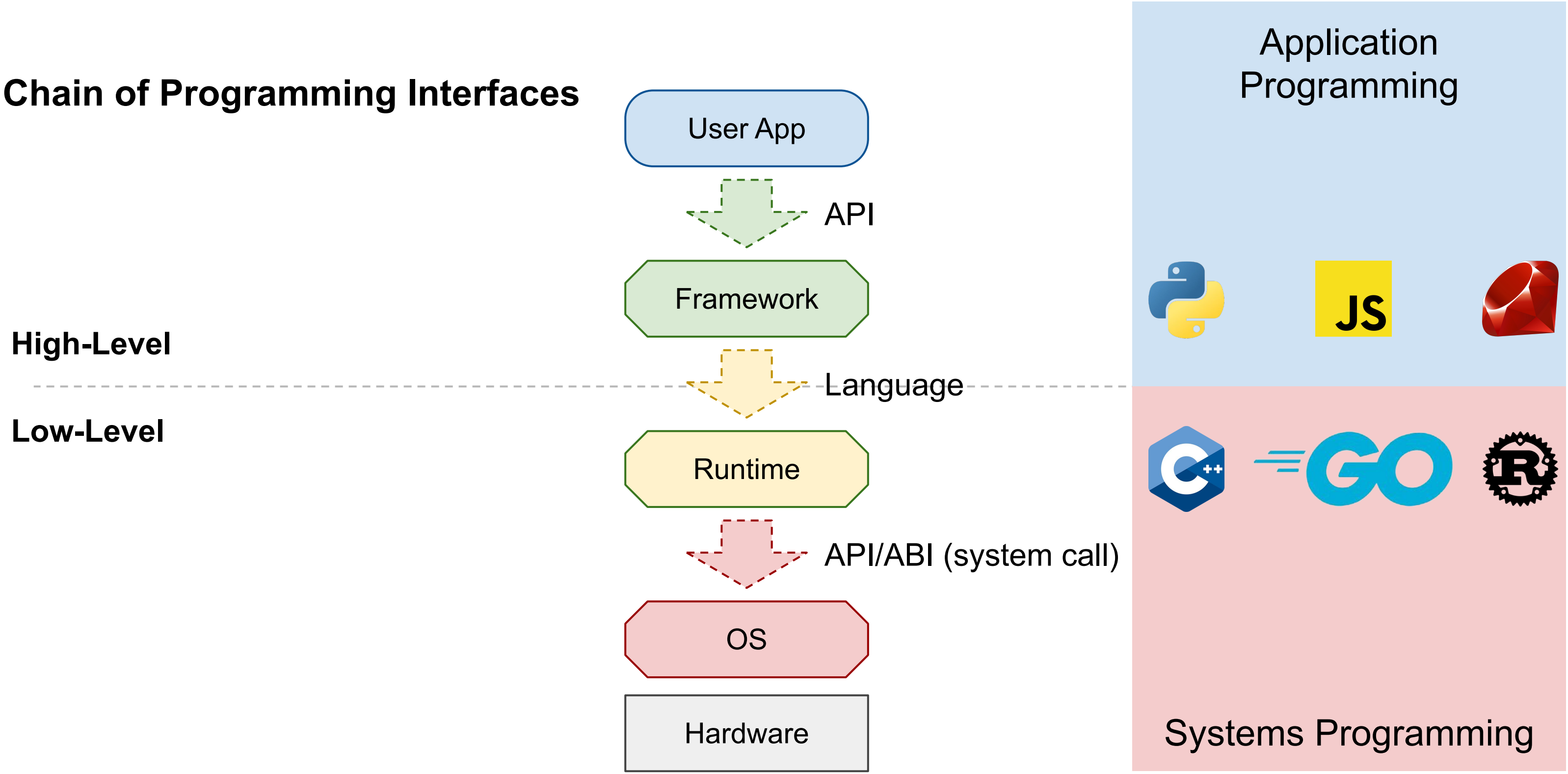
OneOS: Approach

Chain of Programming Interfaces



OneOS: Approach

Chain of Programming Interfaces



High-Level

Low-Level

User App

API

Framework

Language

Runtime

API/ABI (system call)

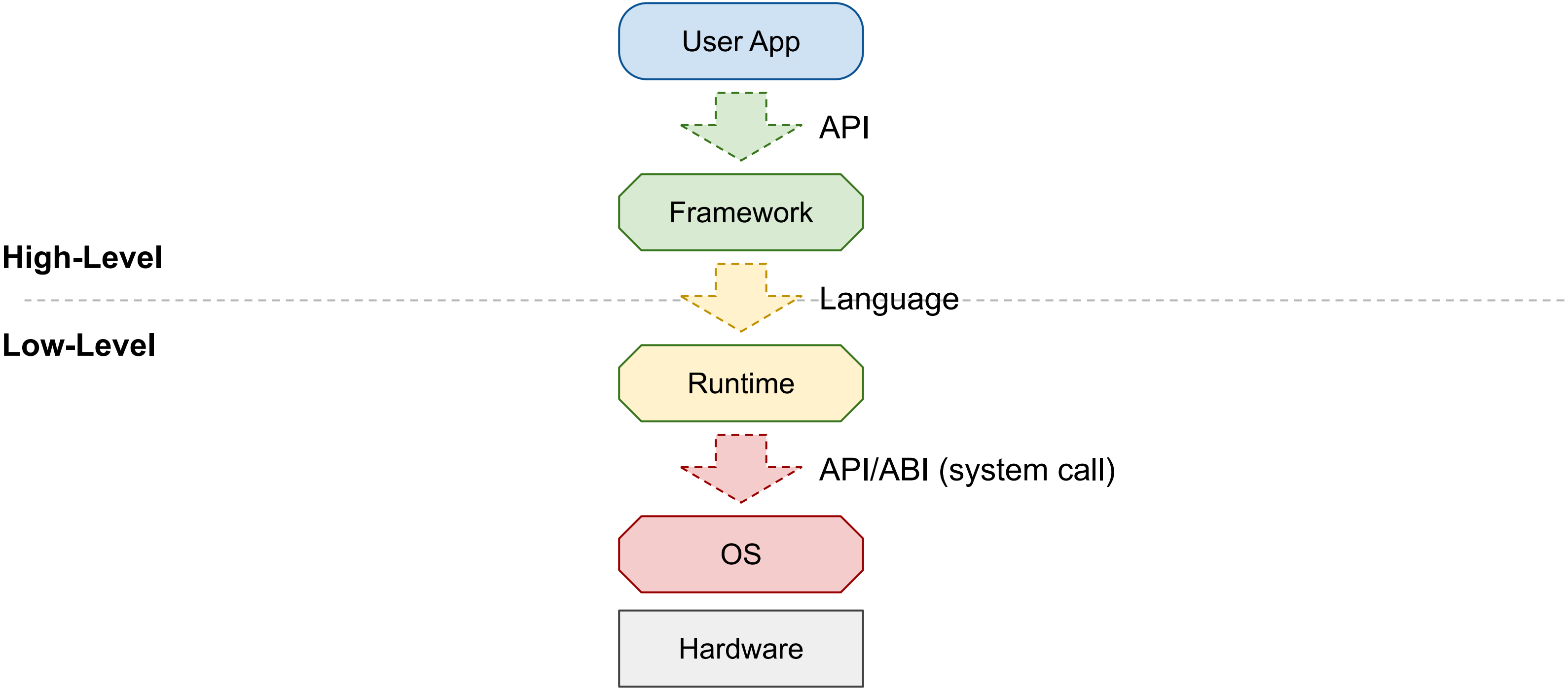
OS

Hardware

Application Programming

Systems Programming

OneOS: Approach



High-Level

Low-Level

User App

API

Framework

Language

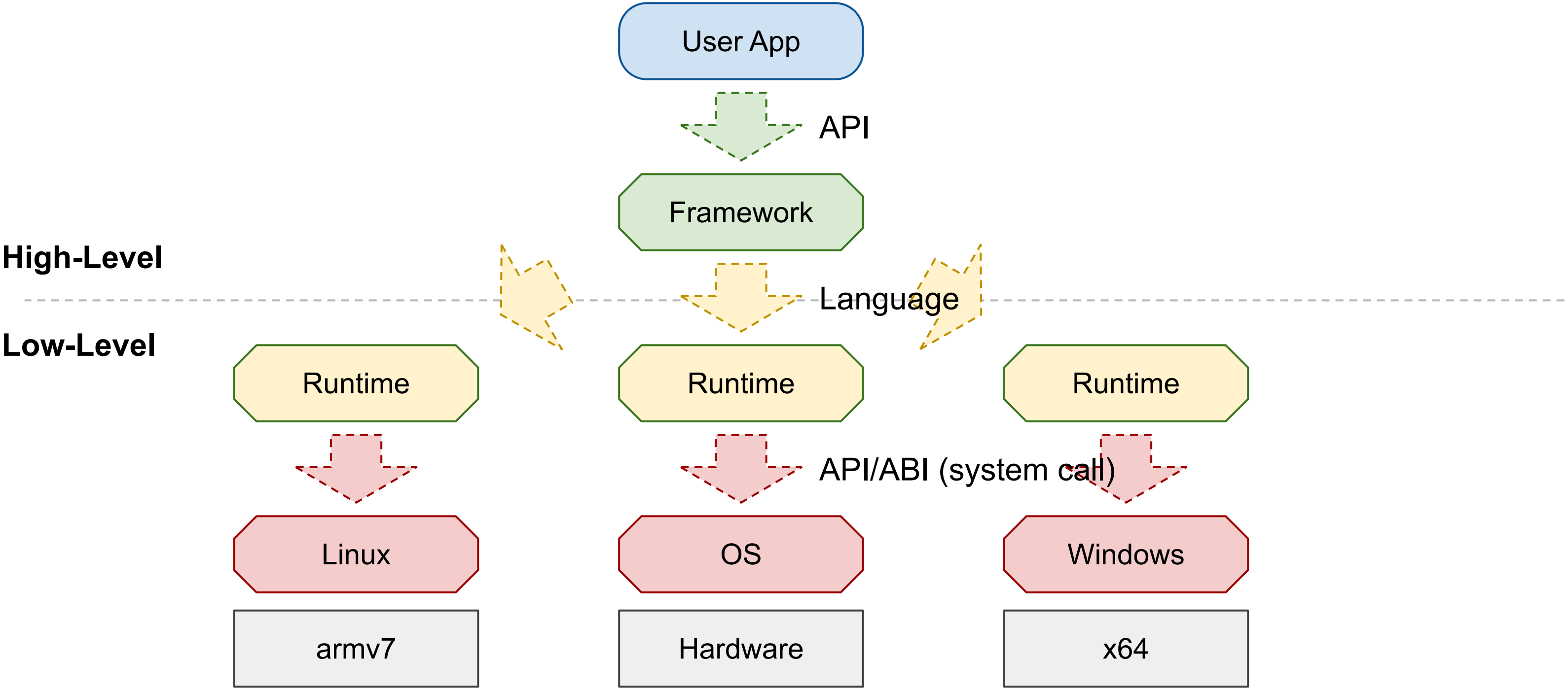
Runtime

API/ABI (system call)

OS

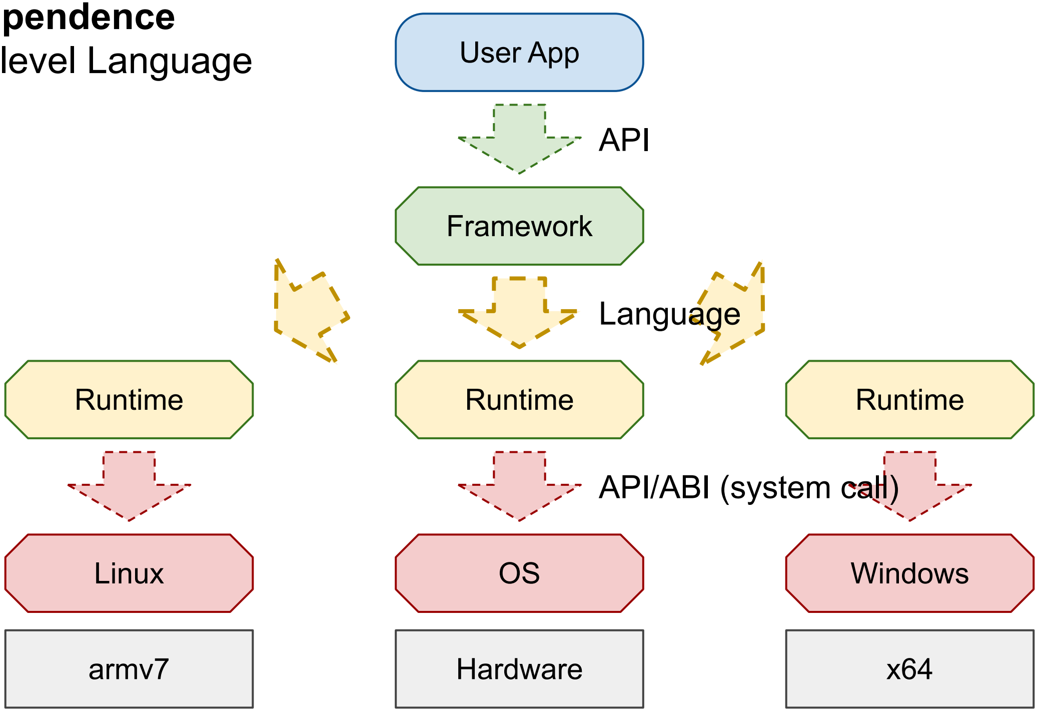
Hardware

OneOS: Approach



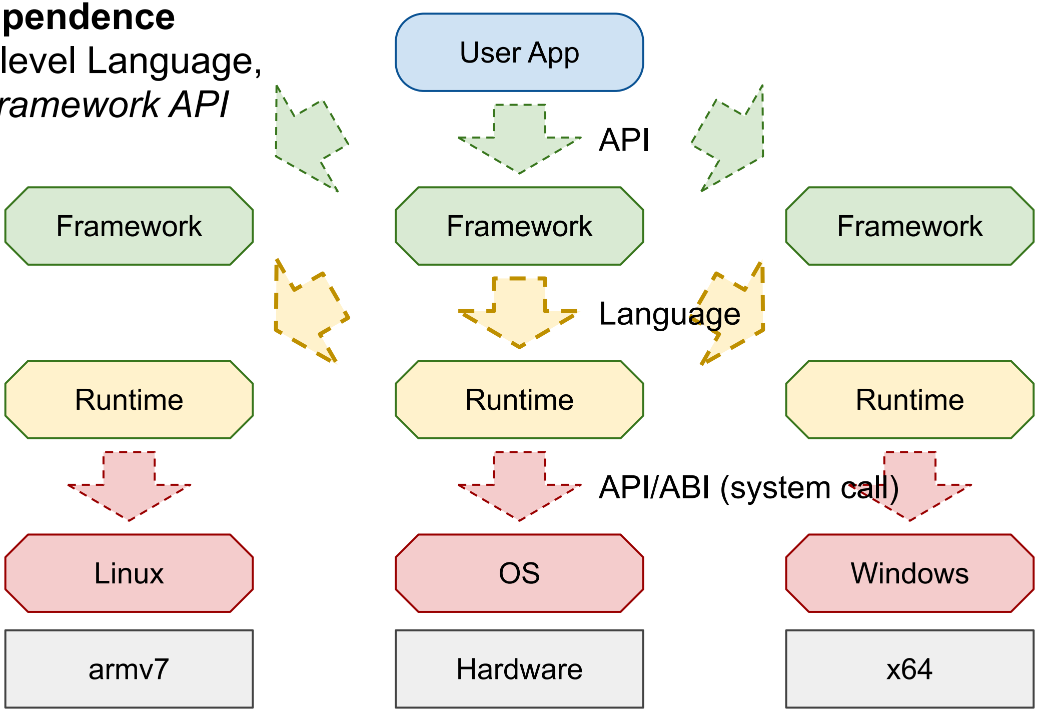
OneOS: Approach

Platform-Independence
by using High-level Language



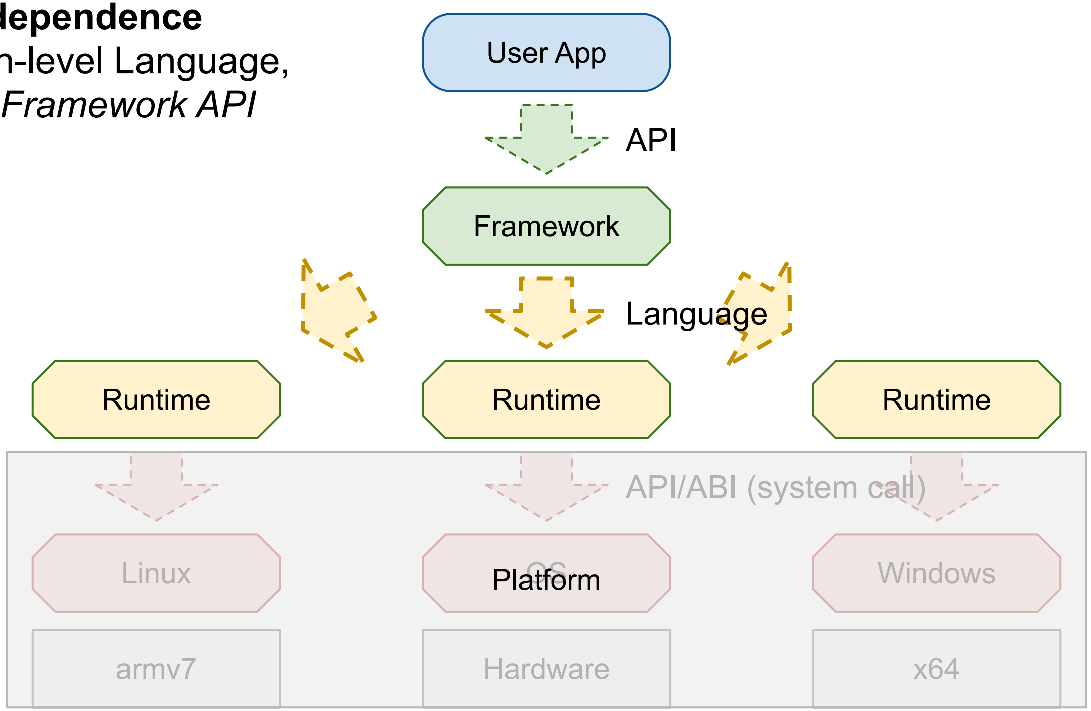
OneOS: Approach

Platform-Independence
by using High-level Language,
not by using Framework API

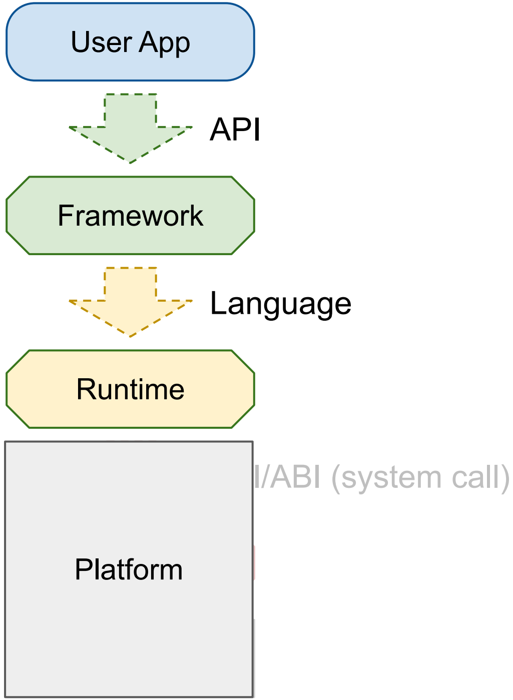


OneOS: Approach

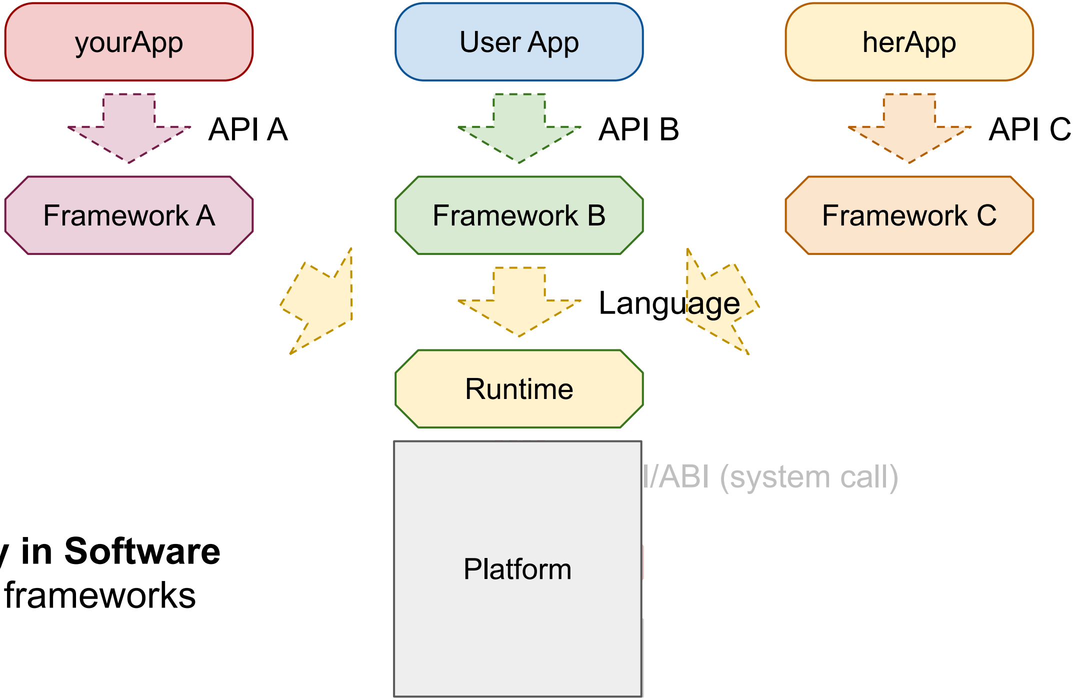
Platform-Independence
by using High-level Language,
not by using Framework API



OneOS: Approach

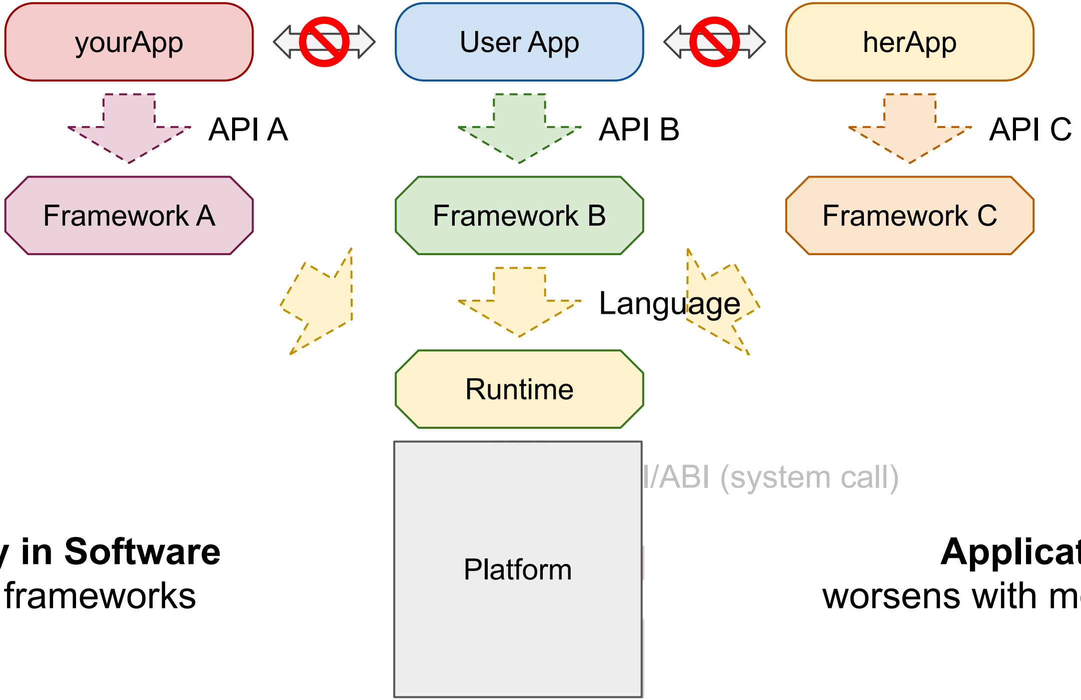


OneOS: Approach



Heterogeneity in Software
unresolved by frameworks

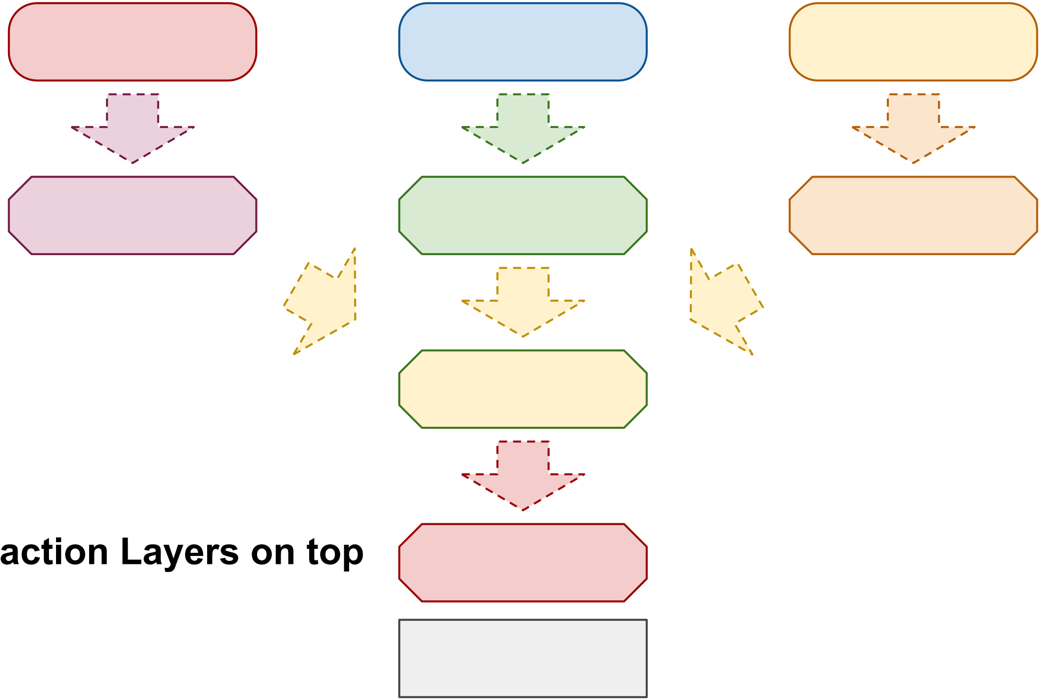
OneOS: Approach



Heterogeneity in Software
unresolved by frameworks

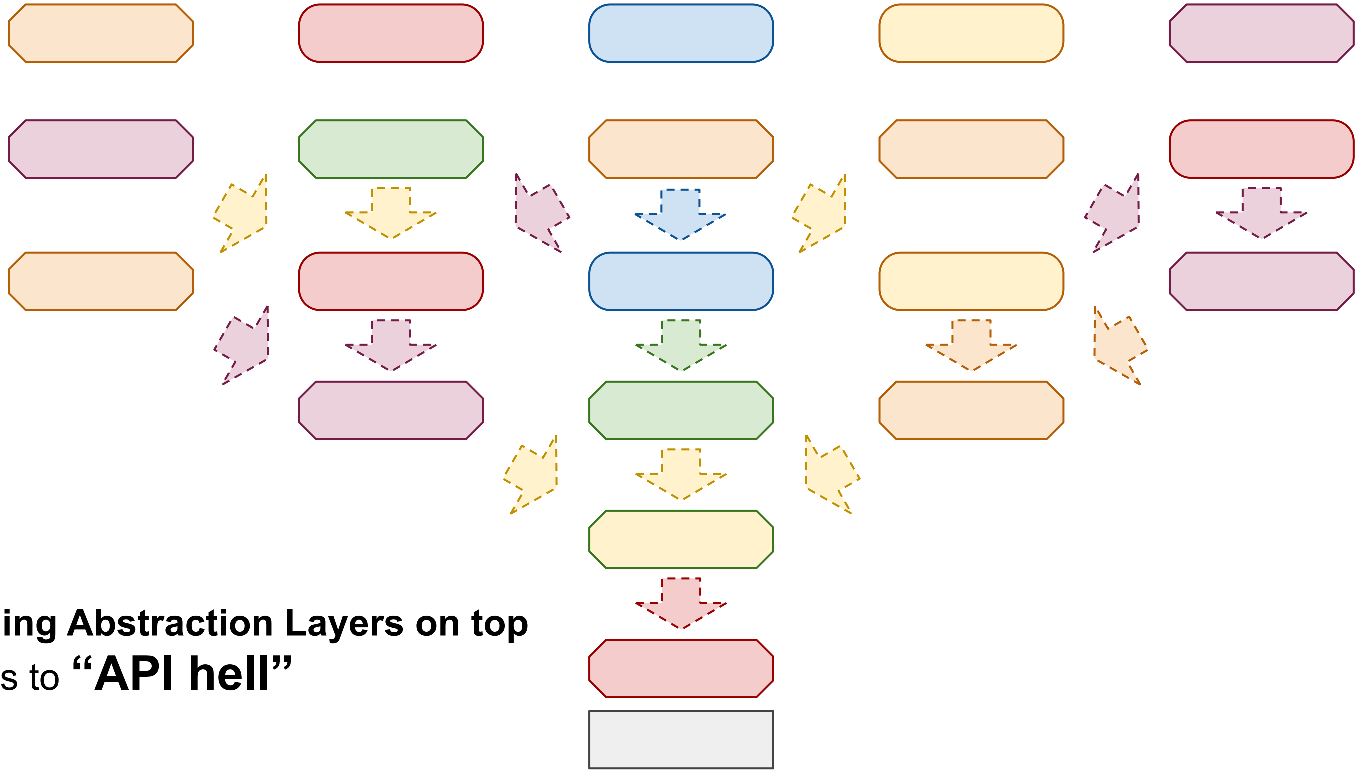
Application Portability
worsens with more frameworks

OneOS: Approach



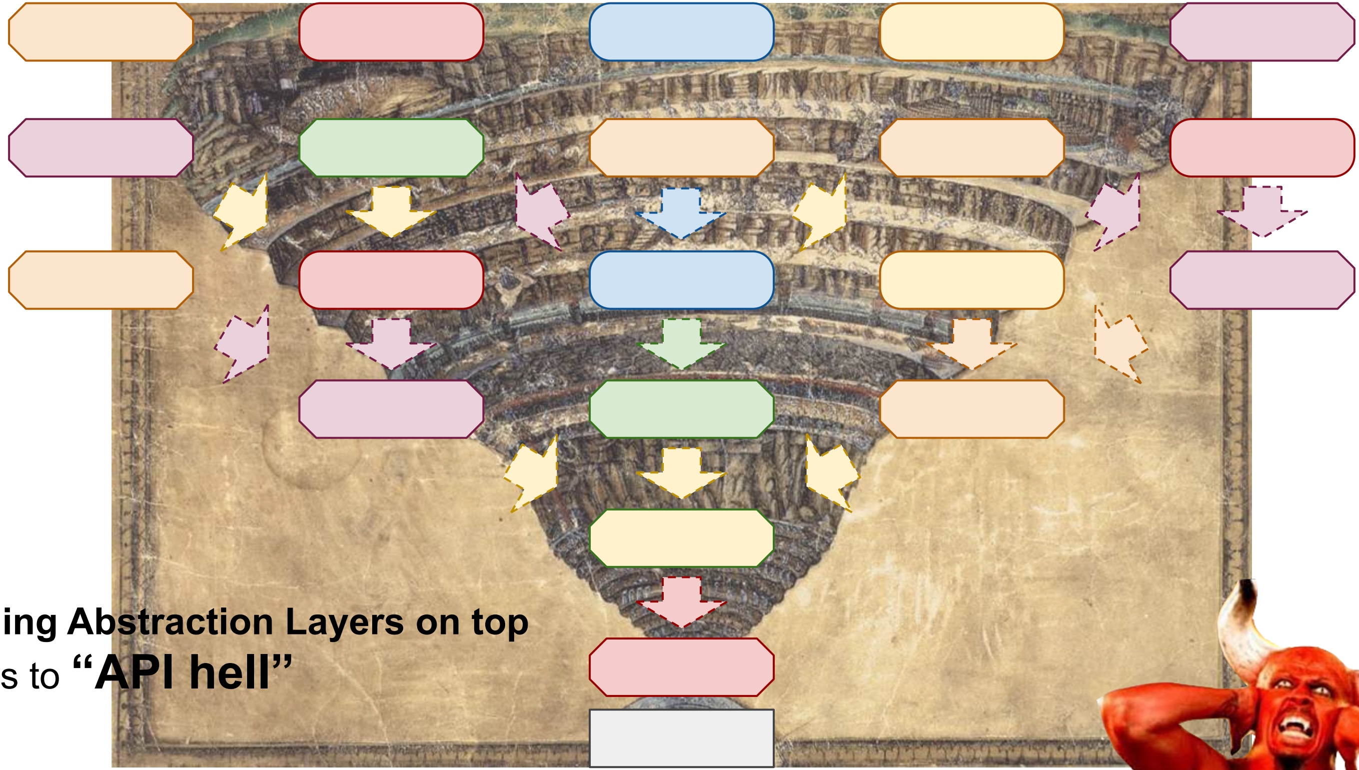
Adding Abstraction Layers on top

OneOS: Approach



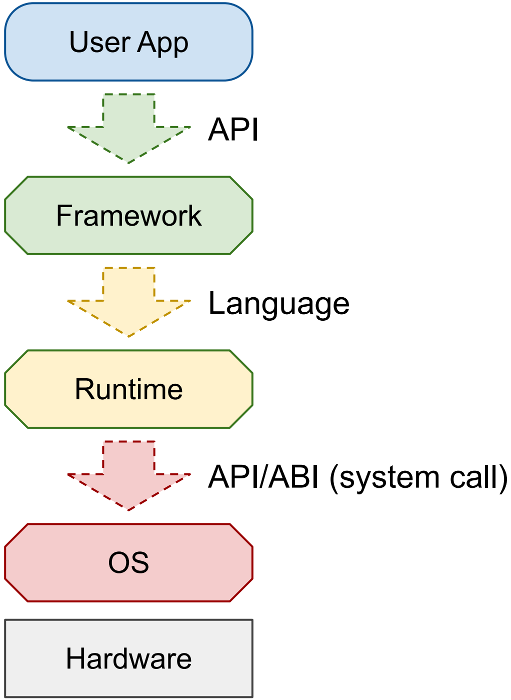
Adding Abstraction Layers on top leads to **“API hell”**

OneOS: Approach



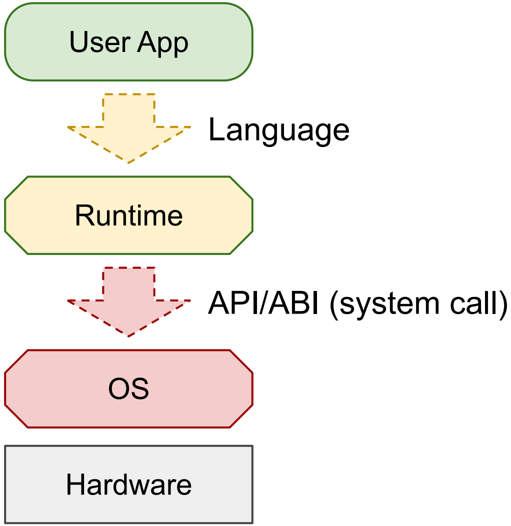
Adding Abstraction Layers on top
leads to **“API hell”**

OneOS: Approach



Our approach:
Not a high-level framework

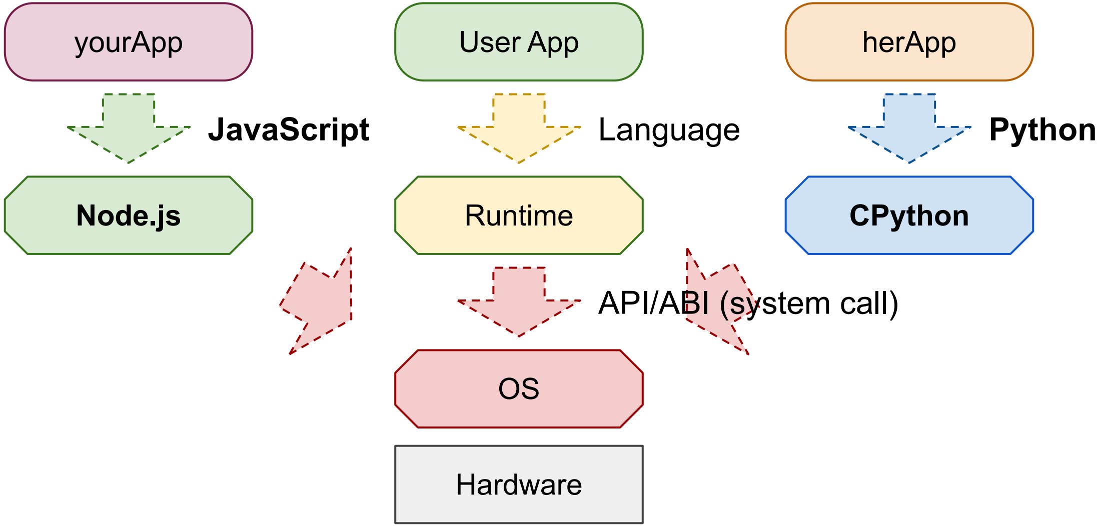
OneOS: Approach



Our approach:
Not a high-level framework

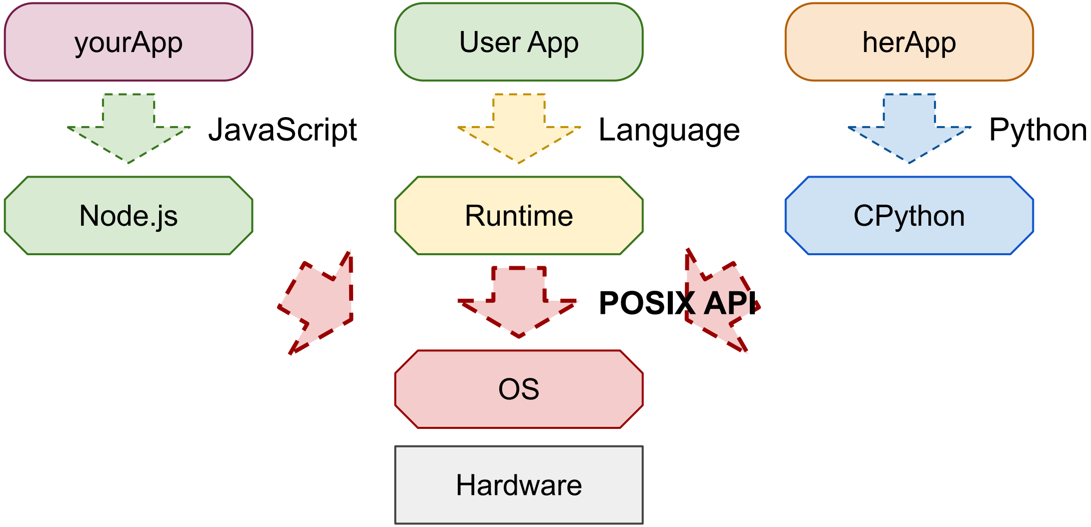
OneOS: Approach

Embrace heterogeneity in software
allow existing technology to work together



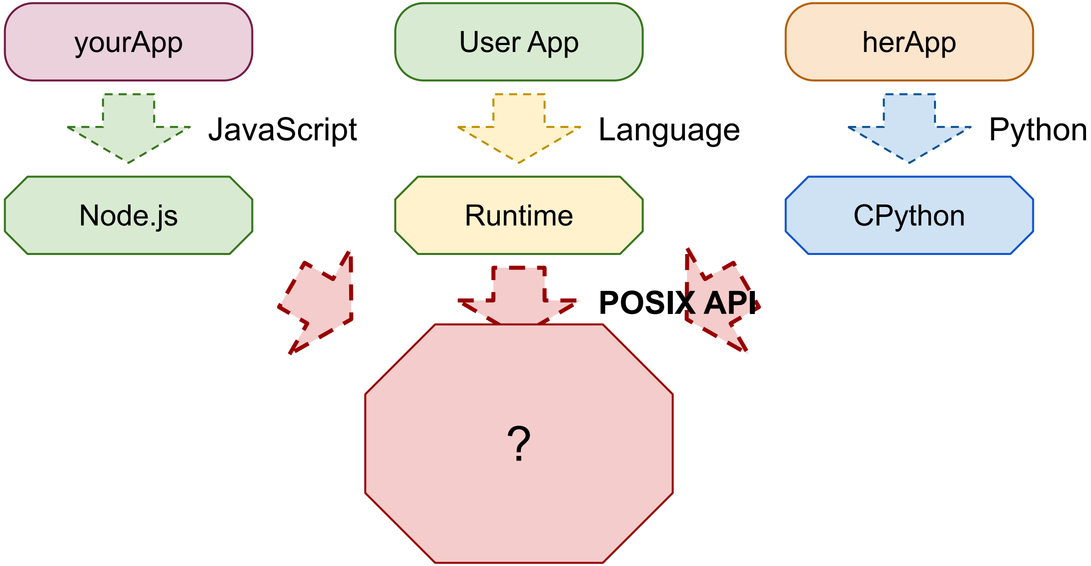
OneOS: Approach

High-level Language VMs share a **common interface**
to the underlying **abstract machine**



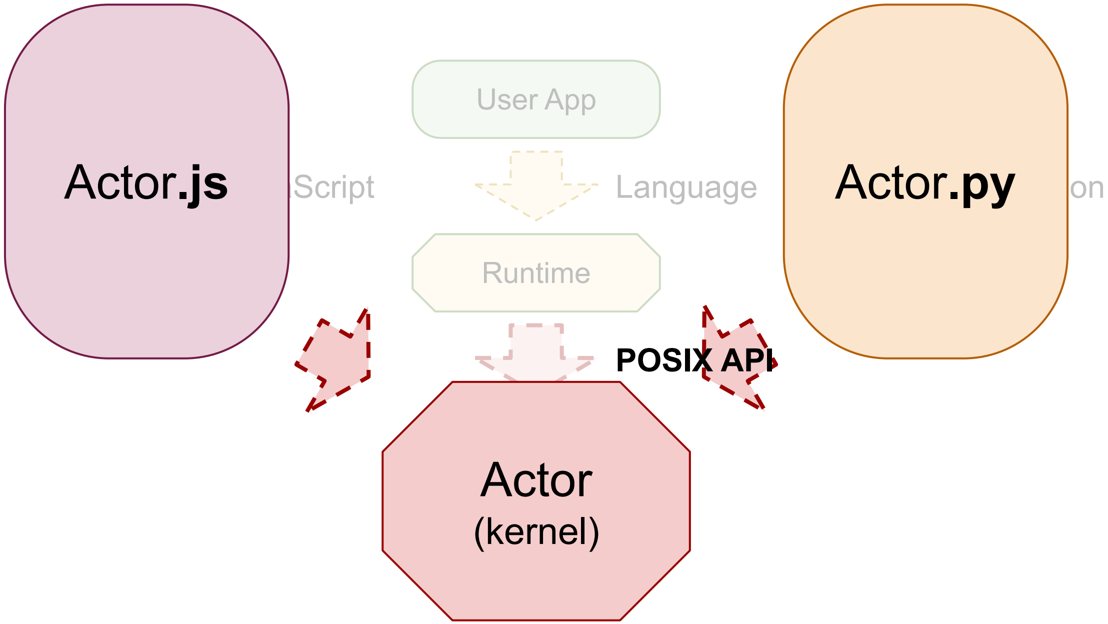
OneOS: Approach

High-level Applications are **agnostic** about the underlying **abstract machine**



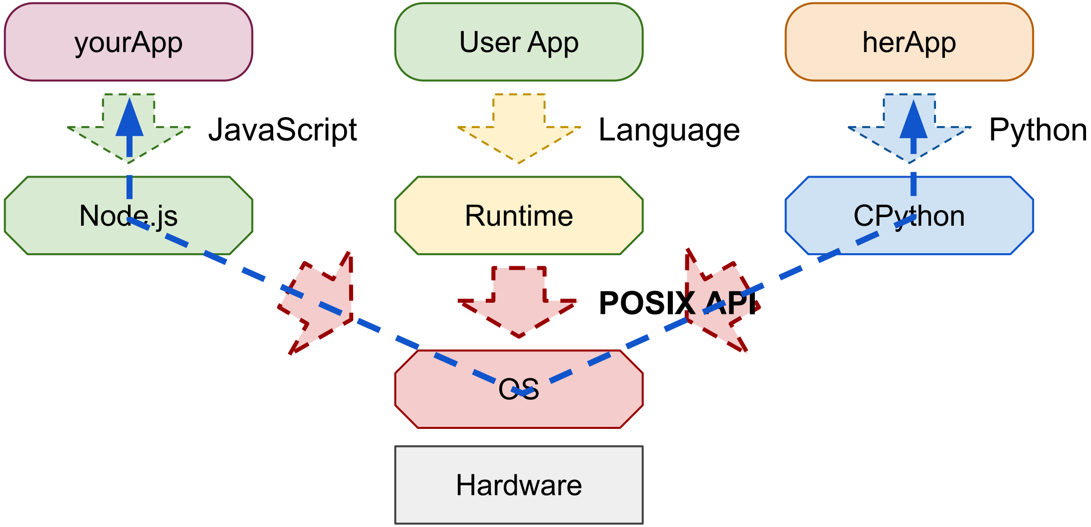
OneOS: Approach

High-level Applications are **agnostic** about the underlying **abstract machine**
System call modeled as **message between Actors**

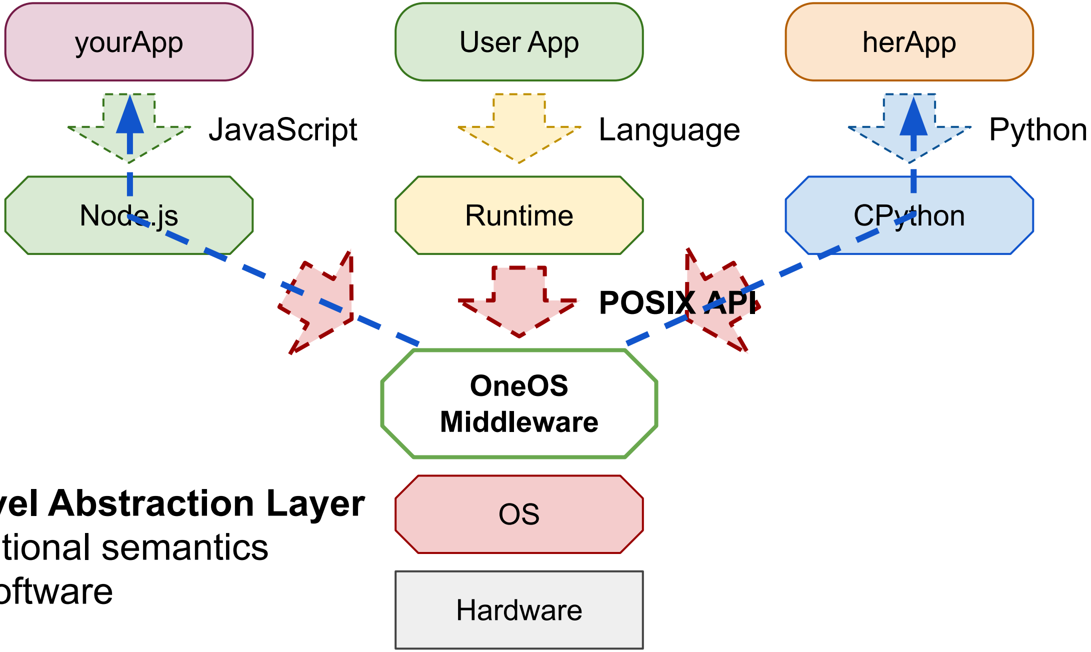


OneOS: Approach

Applications make system calls
to interact with other agents

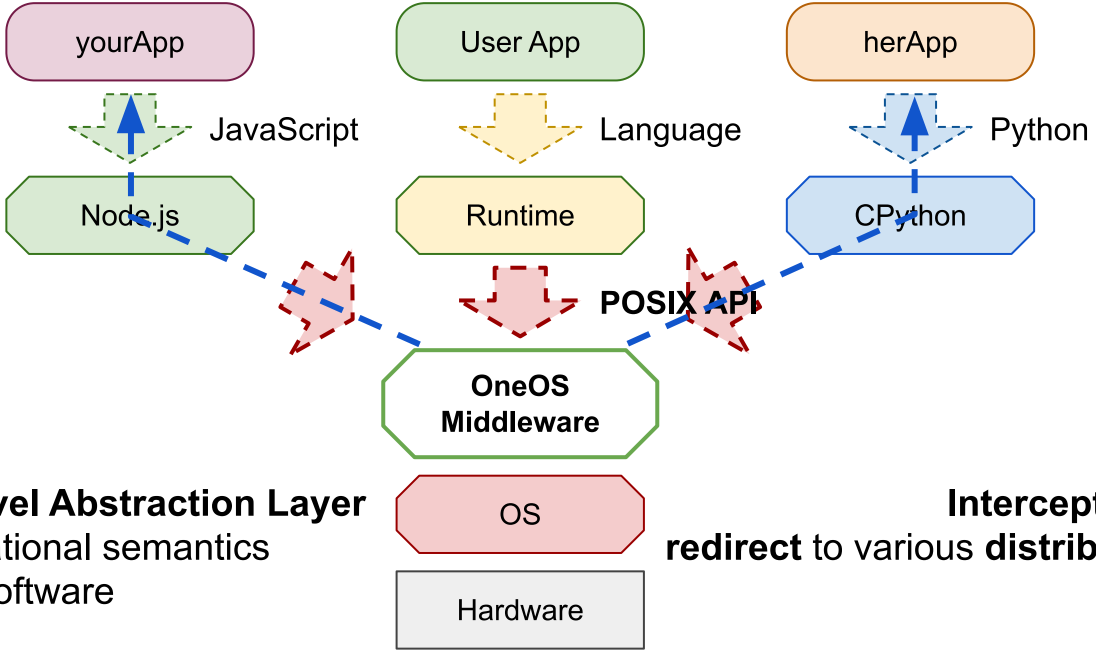


OneOS: Approach



Hijack low-level Abstraction Layer
alter the operational semantics
of high-level software

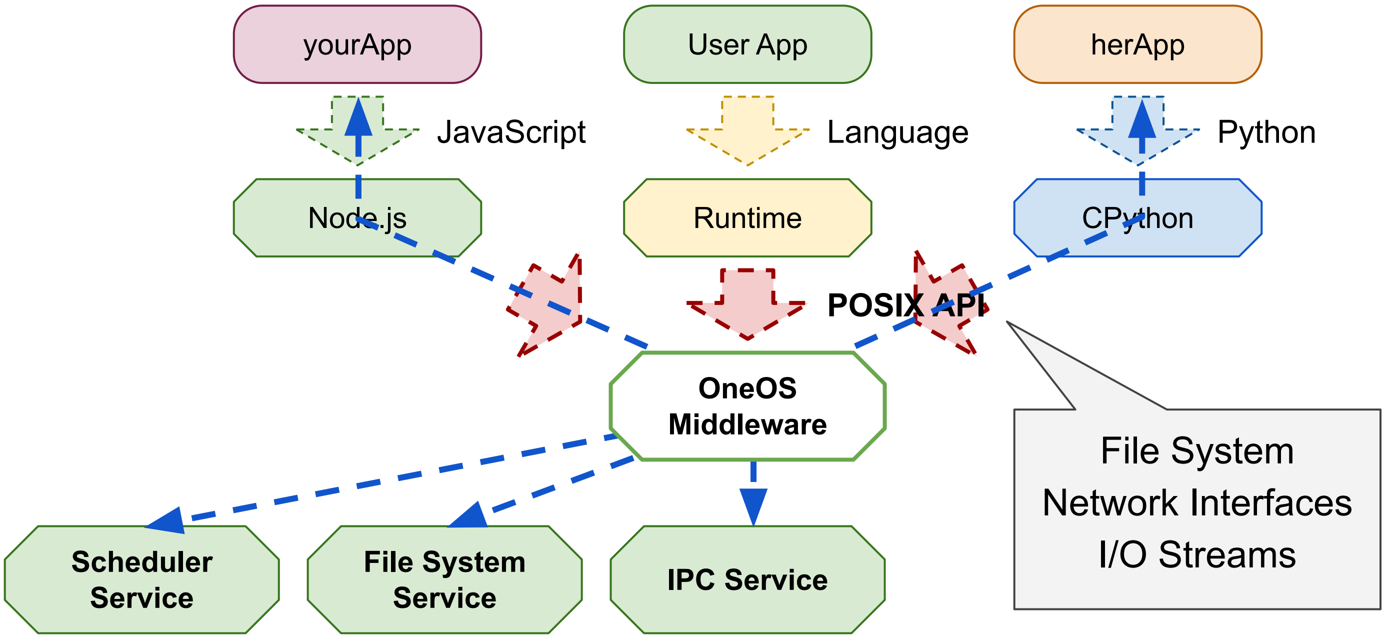
OneOS: Approach



Hijack low-level Abstraction Layer
alter the operational semantics
of high-level software

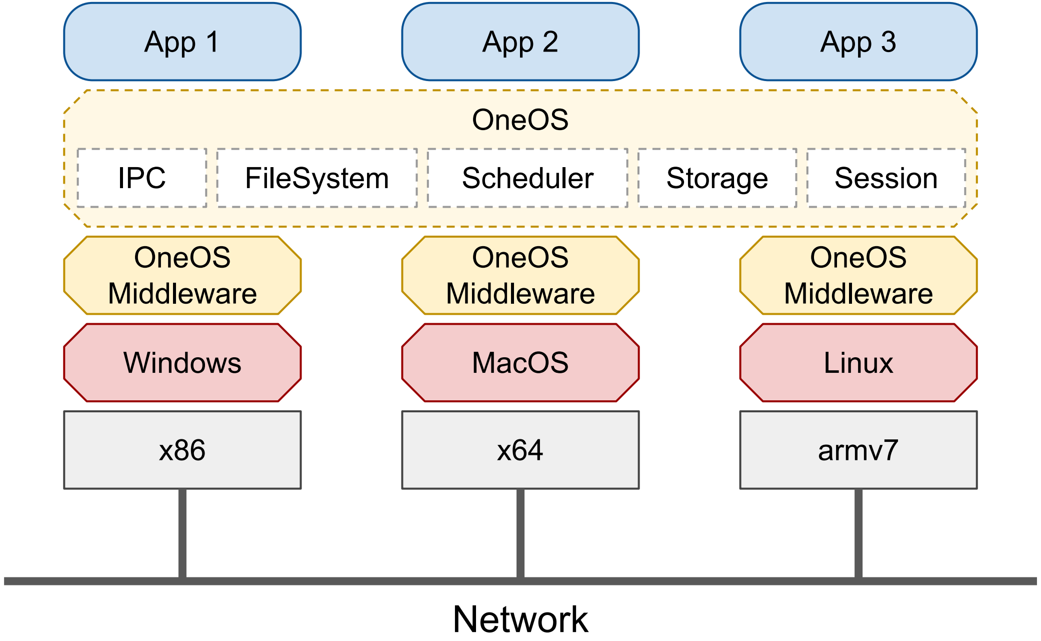
Intercept system calls
redirect to various **distributed services**

OneOS: Approach

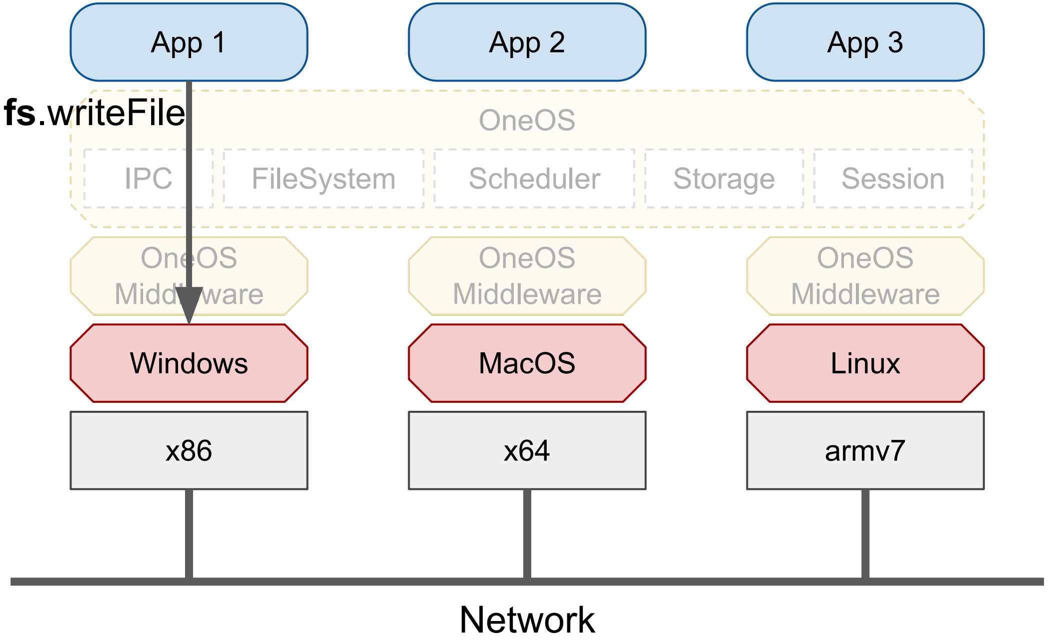


File System
Network Interfaces
I/O Streams

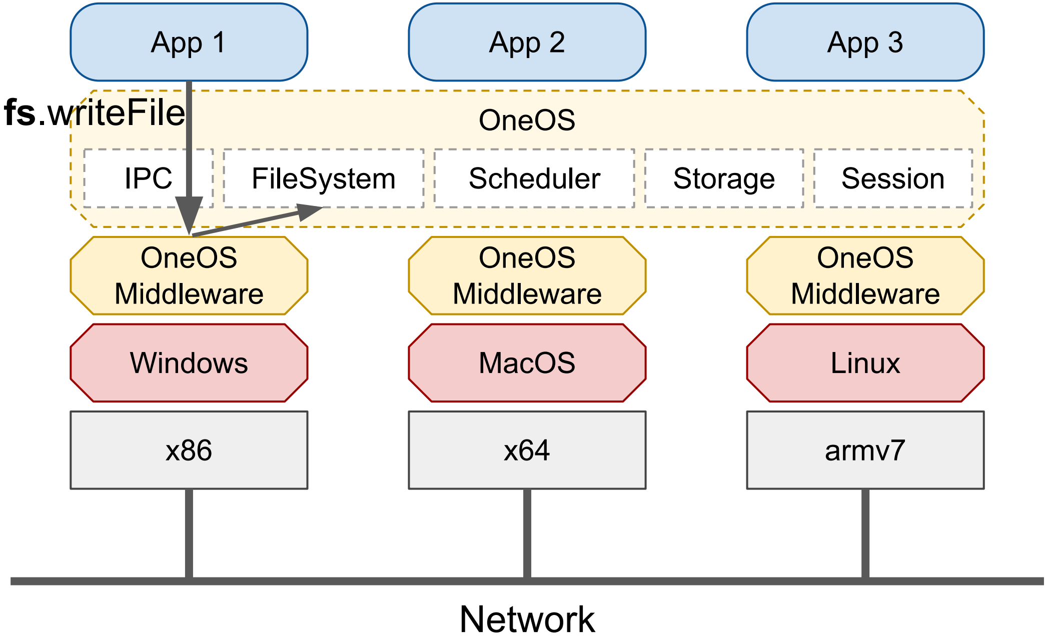
OneOS: Design



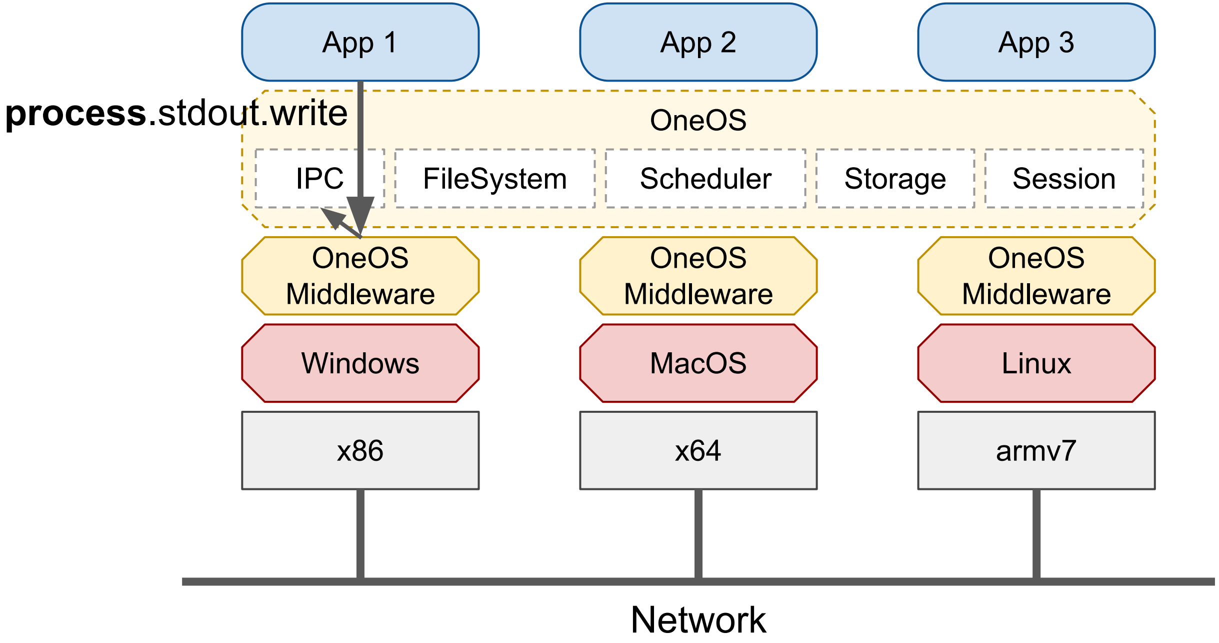
OneOS: Design



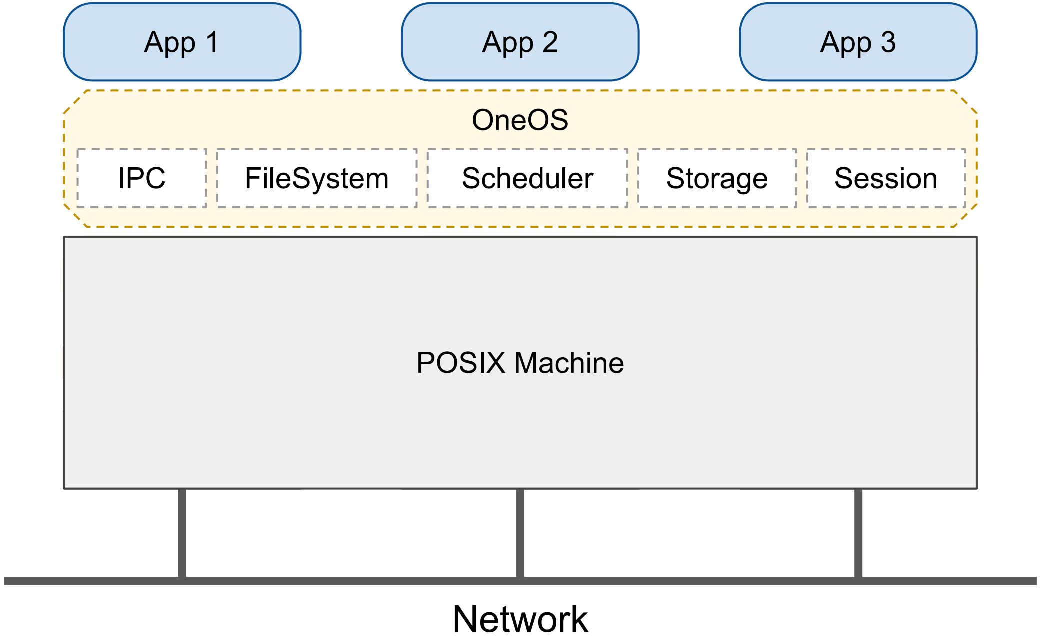
OneOS: Design



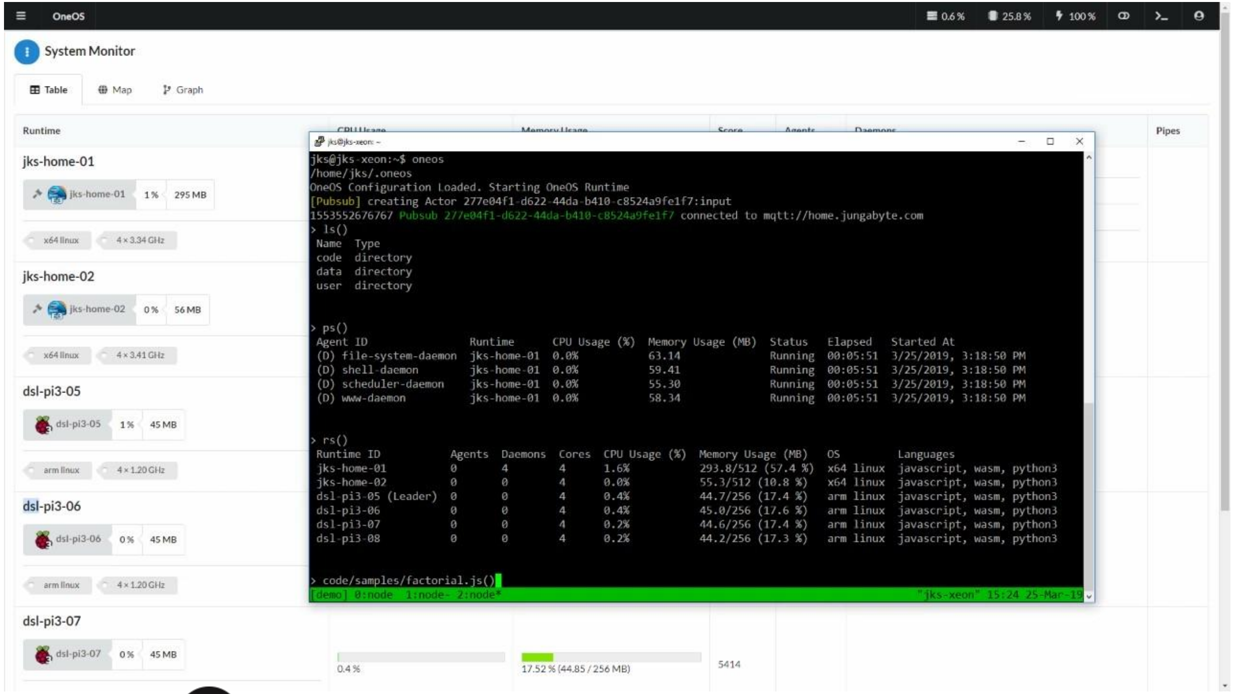
OneOS: Design



OneOS: Design



OneOS: Proof-of-Concept Demo



```
jks@jks-xeon:~$ oneos
/home/jks/.oneos
OneOS Configuration Loaded. Starting OneOS Runtime
[Pubsub] creating Actor 277e04f1-d622-44da-b410-c8524a9fe1f7:input
1553552676767 Pubsub 277e04f1-d622-44da-b410-c8524a9fe1f7 connected to mqtt://home.jungabyte.com
> ls()
 Name  Type
 code  directory
 data  directory
 user  directory


> ps()
 Agent ID                Runtime      CPU Usage (%)  Memory Usage (MB)  Status   Elapsed   Started At
 (D) file-system-daemon  jks-home-01  0.0%           63.14              Running  00:05:51  3/25/2019, 3:18:50 PM
 (D) shell-daemon        jks-home-01  0.0%           59.41              Running  00:05:51  3/25/2019, 3:18:50 PM
 (D) scheduler-daemon    jks-home-01  0.0%           55.30              Running  00:05:51  3/25/2019, 3:18:50 PM
 (D) www-daemon          jks-home-01  0.0%           58.34              Running  00:05:51  3/25/2019, 3:18:50 PM


> rs()
 Runtime ID            Agents  Daemons  Cores  CPU Usage (%)  Memory Usage (MB)    OS         Languages
 jks-home-01           0       4        4      1.6%           293.8/512 (57.4 %)  x64 linux  javascript, wasm, python3
 jks-home-02           0       0        4      0.0%           55.3/512 (10.8 %)   x64 linux  javascript, wasm, python3
 dsl-pi3-05 (Leader)   0       0        4      0.4%           44.7/256 (17.4 %)   arm linux  javascript, wasm, python3
 dsl-pi3-06            0       0        4      0.4%           45.0/256 (17.6 %)   arm linux  javascript, wasm, python3
 dsl-pi3-07            0       0        4      0.2%           44.6/256 (17.4 %)   arm linux  javascript, wasm, python3
 dsl-pi3-08            0       0        4      0.2%           44.2/256 (17.3 %)   arm linux  javascript, wasm, python3


> code/samples/factorial.js()
[demo] 0:node  1:node- 2:node*                                                    "jks-xeon" 15:24 25-Mar-19
```

DependableSystemsLab/OneOS

OneOS: Discussion

Feedback Wanted:

- Evaluation strategies
- Practicality of Actor-based micro-kernel
- Suitability of high-level language for systems programming

Controversial Points:

- *Single system image* appropriate for a *geographically distributed grid*?
- Mapping *POSIX interface* over an *inherently distributed and concurrent* architecture?
- *Limiting application space* to *high-level languages*?

Open Issues & Future Work:

- Security and Privacy model
- Failure handling
- Semantics of cyber-physical resources

Potential Drawbacks:

- Fundamental tension between cyber-physical resources and their abstract representations
- Reasoning about security concerns within high-level programming space
- Inability to make low-level optimizations

✉ kumseok@ece.ubc.ca DependableSystemsLab/OneOS