



# Jumpbox®

Tech Passion | Sharing | Society



**Saritrat Jirakulphondchai (JoJo)**

**Jumpbox**

**Jumpbox®**

ไปแวนปะ



วัยรุ่น

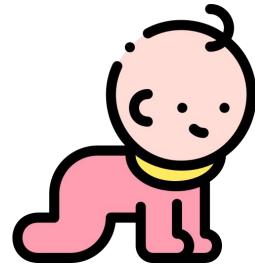


อึก



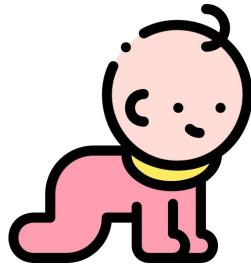
# Reasoning? (การให้เหตุผล)

# วัยรุ่น Reasoning



Baby

# วัยรุ่น Reasoning



Baby



Racing Story

# วัยรุ่น Reasoning

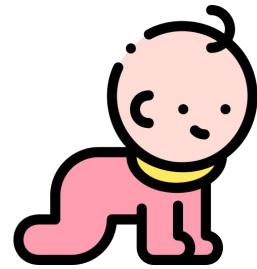


Baby



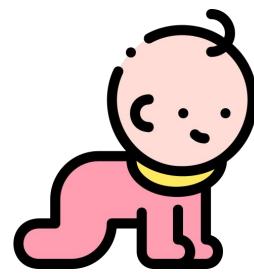
Racing Story

# วัยรุ่น Reasoning



Baby

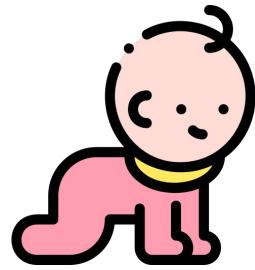
# วัยรุ่น Reasoning



Baby

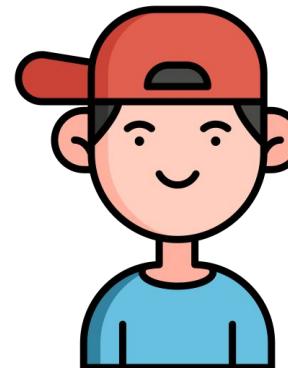


# วัยรุ่น Reasoning



Baby

Transform



วัยรุ่น

# Application?

Web

Application?

Jumpbox®

Web

Mobile

Android

iOS

# Application?

Web

Computer

Mobile

Android

iOS

# Application?

Mac

...etc

Debian

Linux

Jumpbox®

# Application

# Application (app)

# Application (app)

computer software package

# Application (app)

computer software package that performs a specific function directly for an end user

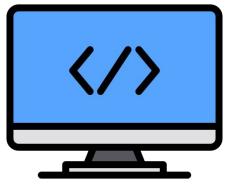
# Application (app)

computer software package that performs a specific function directly for an end user or, in some cases, for another application.

# Application Process

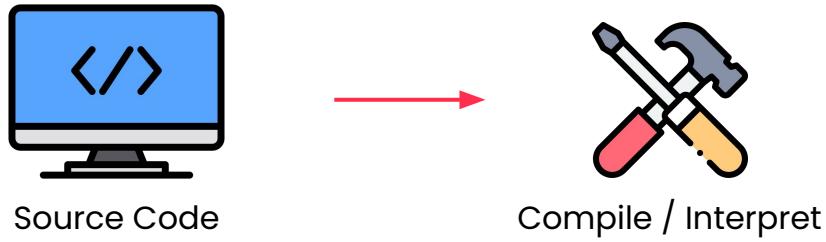
# Application Process

# Application Process

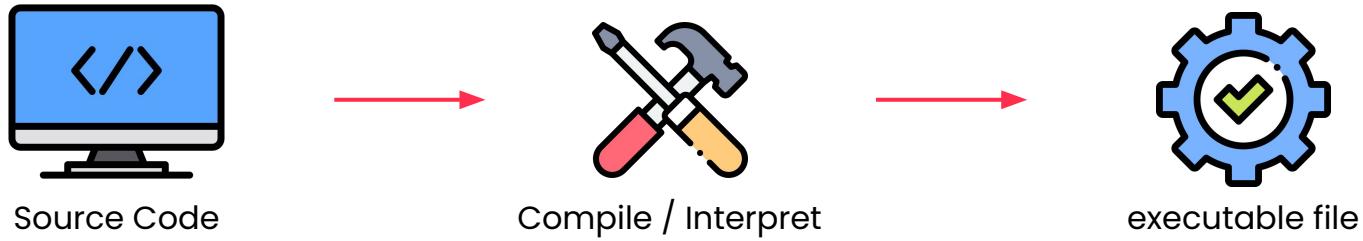


Source Code

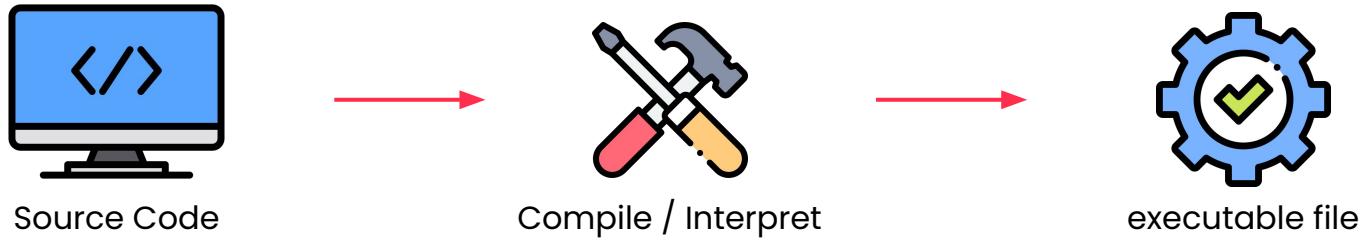
# Application Process



# Application Process



# Application Process



- OS Executable
- Need Application Runtime

**Executable** file: System Programming Language (**OS Executable**)

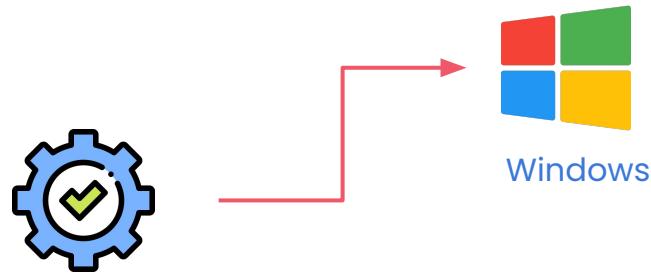
**Executable** file: System Programming Language (**OS Executable**)



executable file

(OS Executable)

## Executable file: System Programming Language ([OS Executable](#))



executable file

(OS Executable)

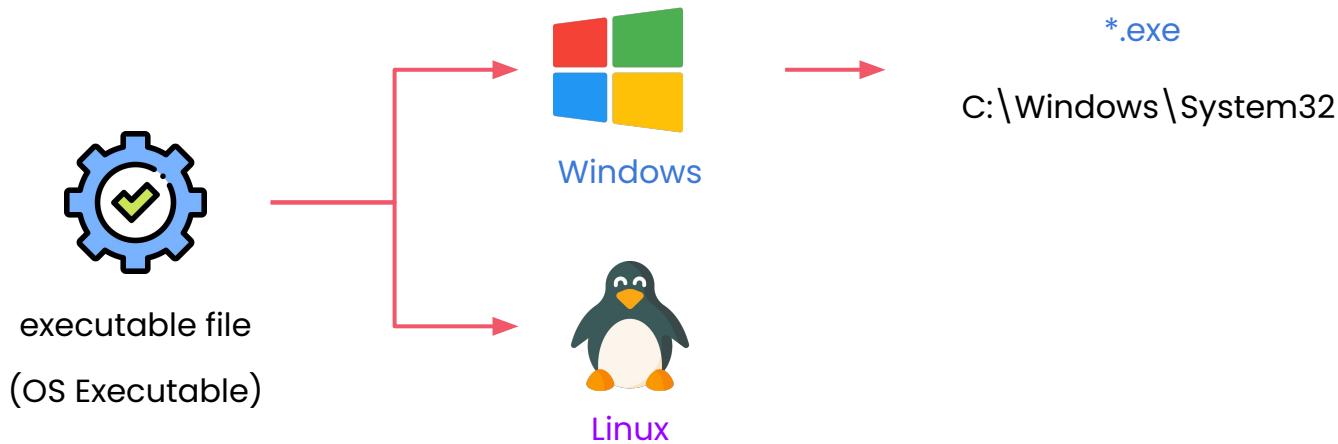
## Executable file: System Programming Language ([OS Executable](#))



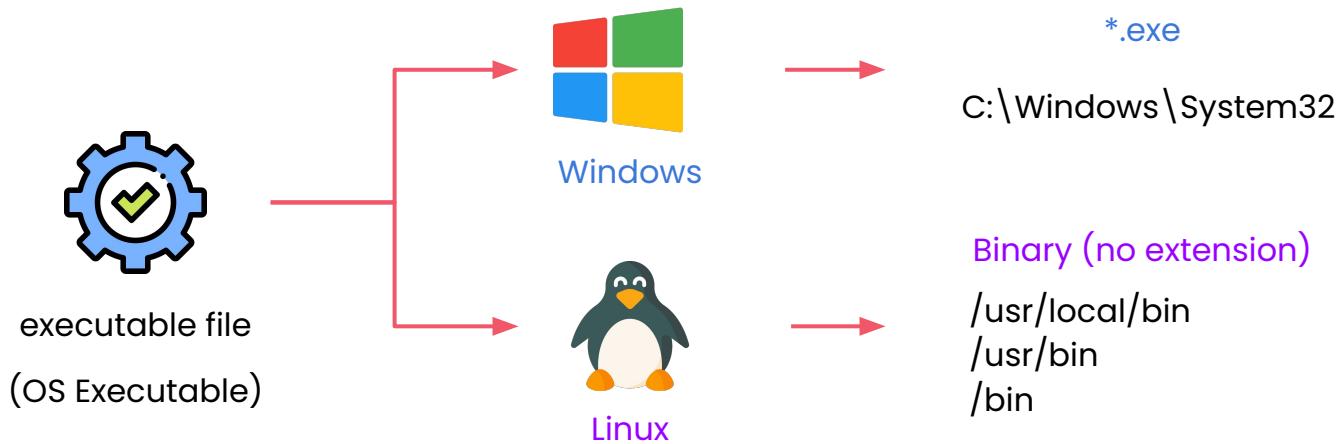
executable file  
(OS Executable)

\*.exe  
C:\Windows\System32

## Executable file: System Programming Language (OS Executable)

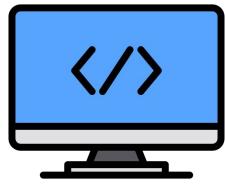


## Executable file: System Programming Language (OS Executable)



**Executable** file: Application Programming Language (**Need Runtime**)

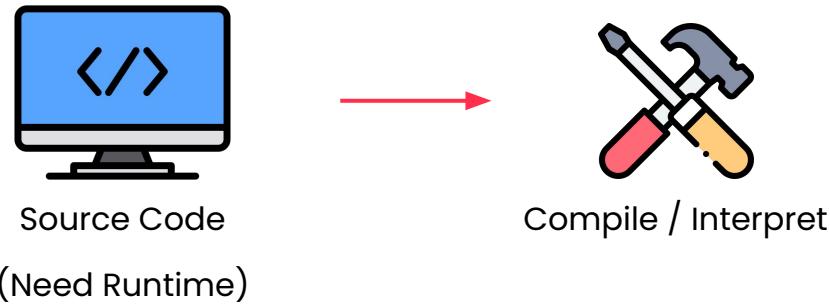
Executable file: Application Programming Language (**Need Runtime**)



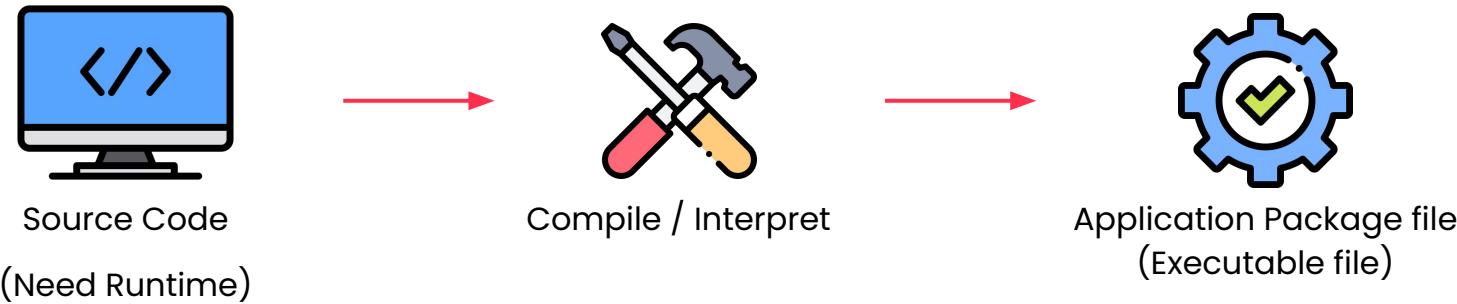
Source Code

(Need Runtime)

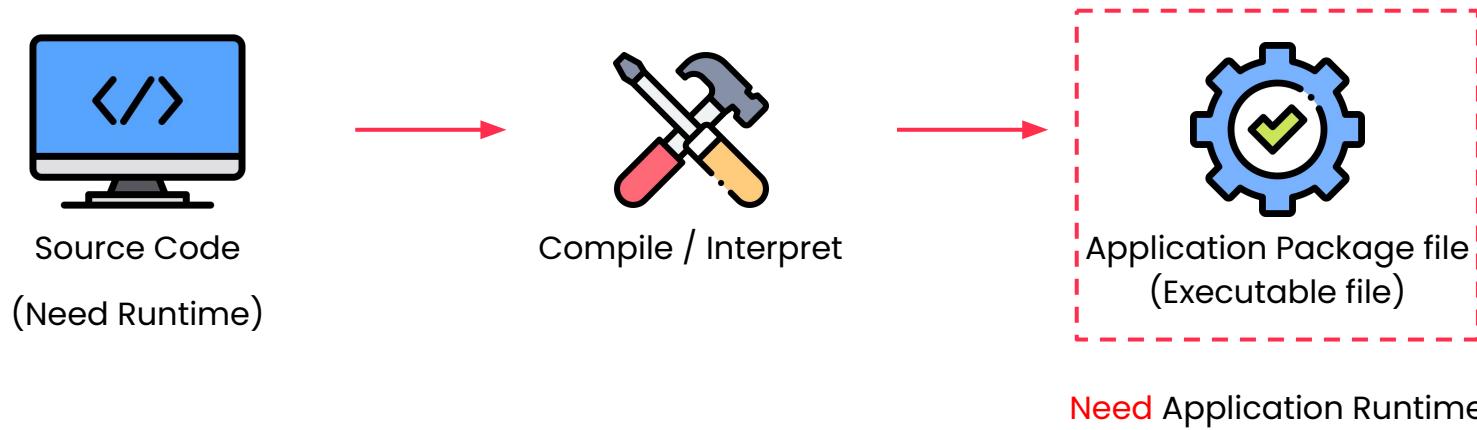
**Executable** file: Application Programming Language (**Need Runtime**)



**Executable** file: Application Programming Language (**Need Runtime**)



## Executable file: Application Programming Language (**Need Runtime**)

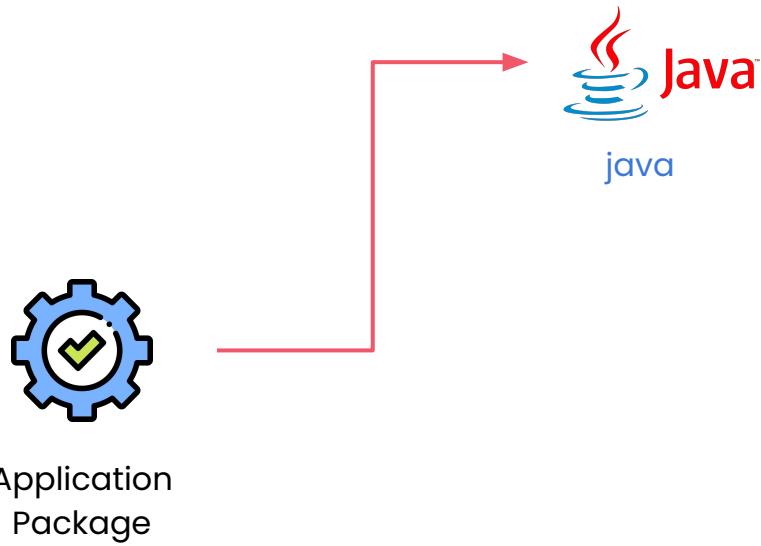


# Application Package

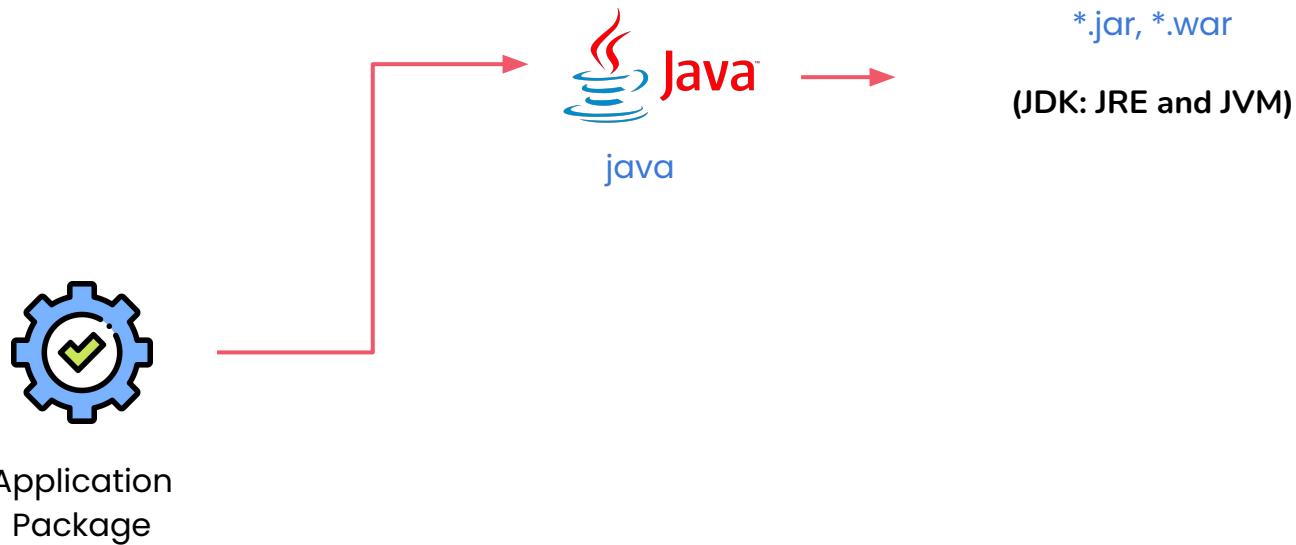


Application  
Package

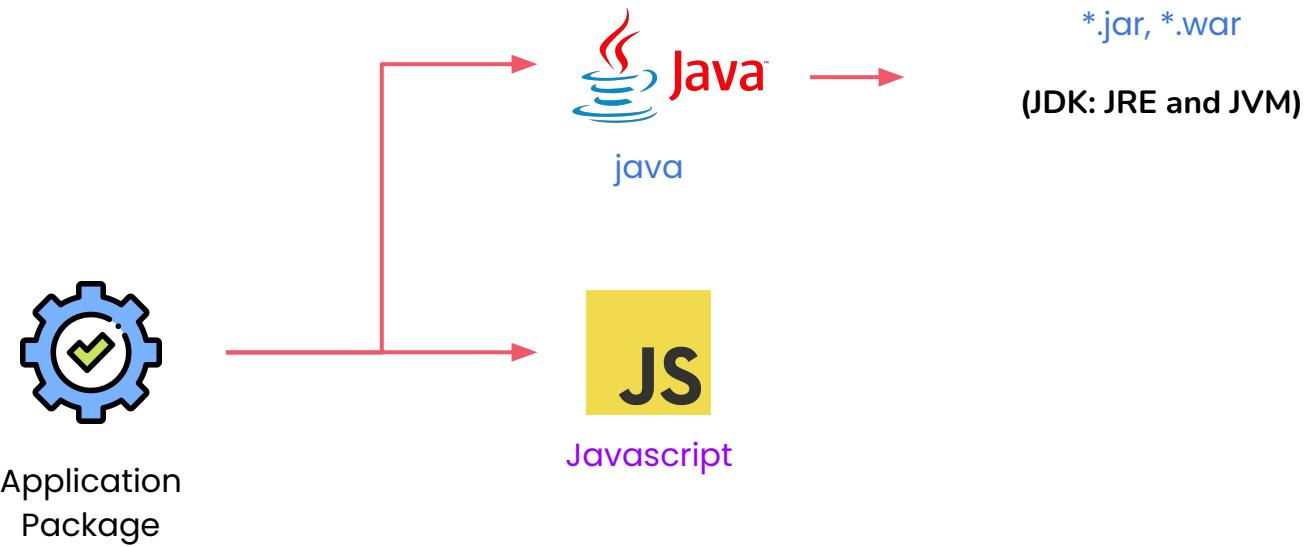
## Application Package



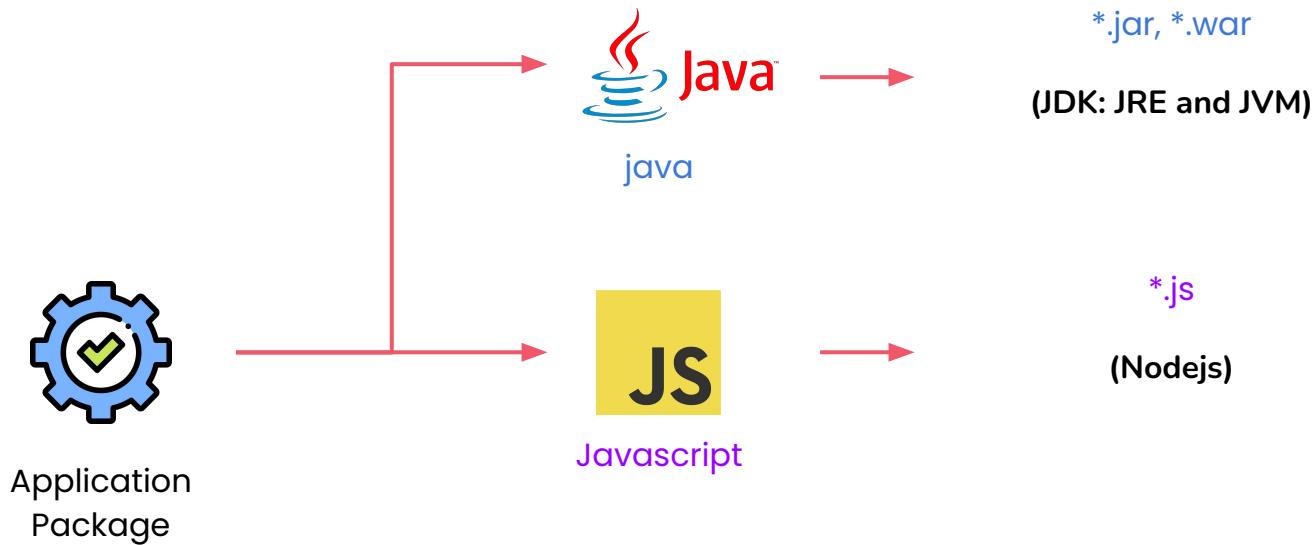
## Application Package



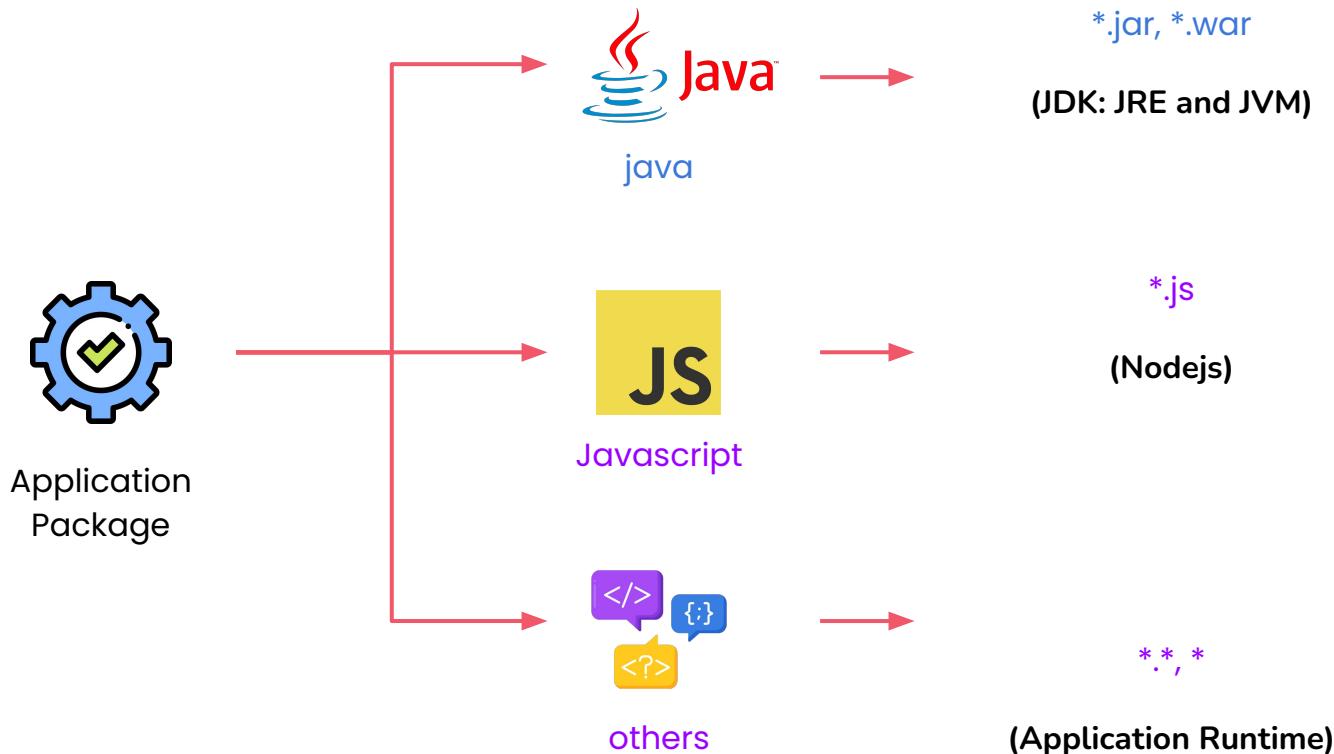
# Application Package



# Application Package



# Application Package



## Summary Executable file

## Summary Executable file

### **System Programming Language:** OS Executable

- Window: \*.exe
- Linux, MacOS: binary

## Summary Executable file

### **System Programming Language:** OS Executable

- Window: \*.exe
- Linux, MacOS: binary

### **Application Programming Language:** Need Application Runtime

- JDK: Java
- Nodejs: Javascript
- Dotnet: .Net
- Others: others

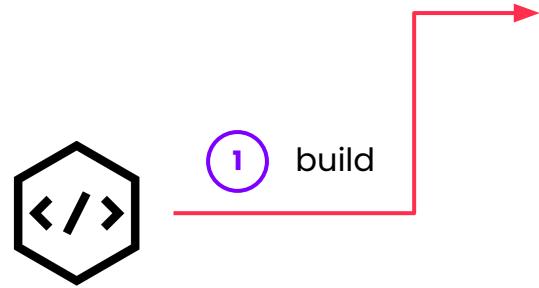
# Application Deployment

# Application Deployment Process

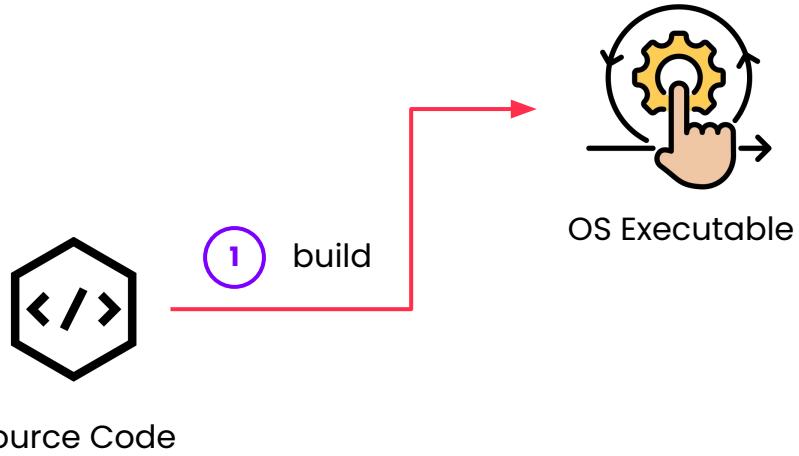


Source Code

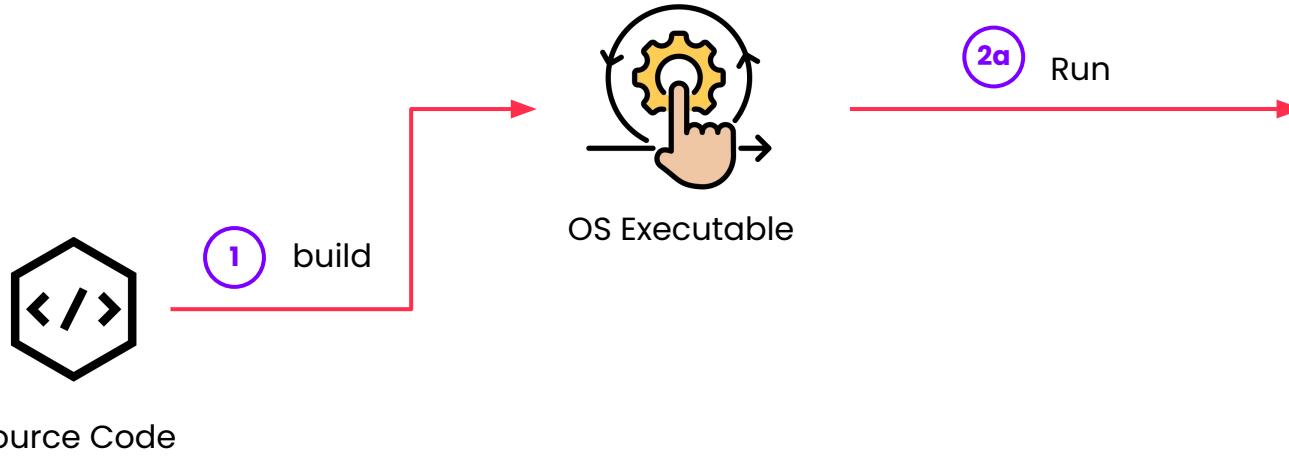
# Application Deployment Process



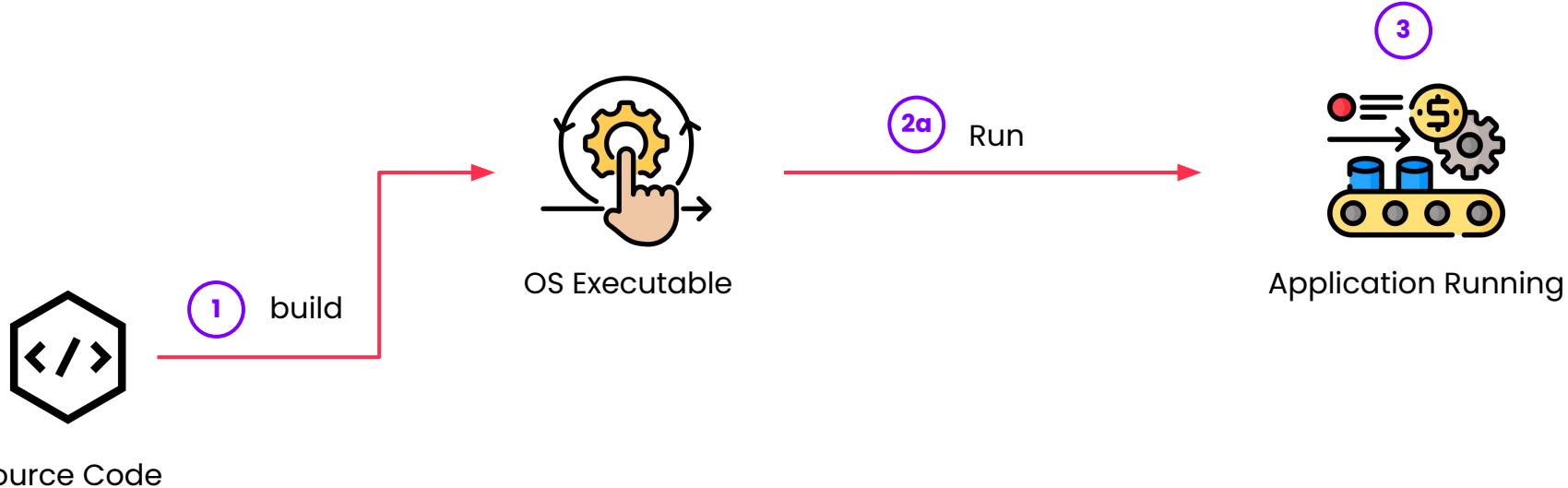
# Application Deployment Process



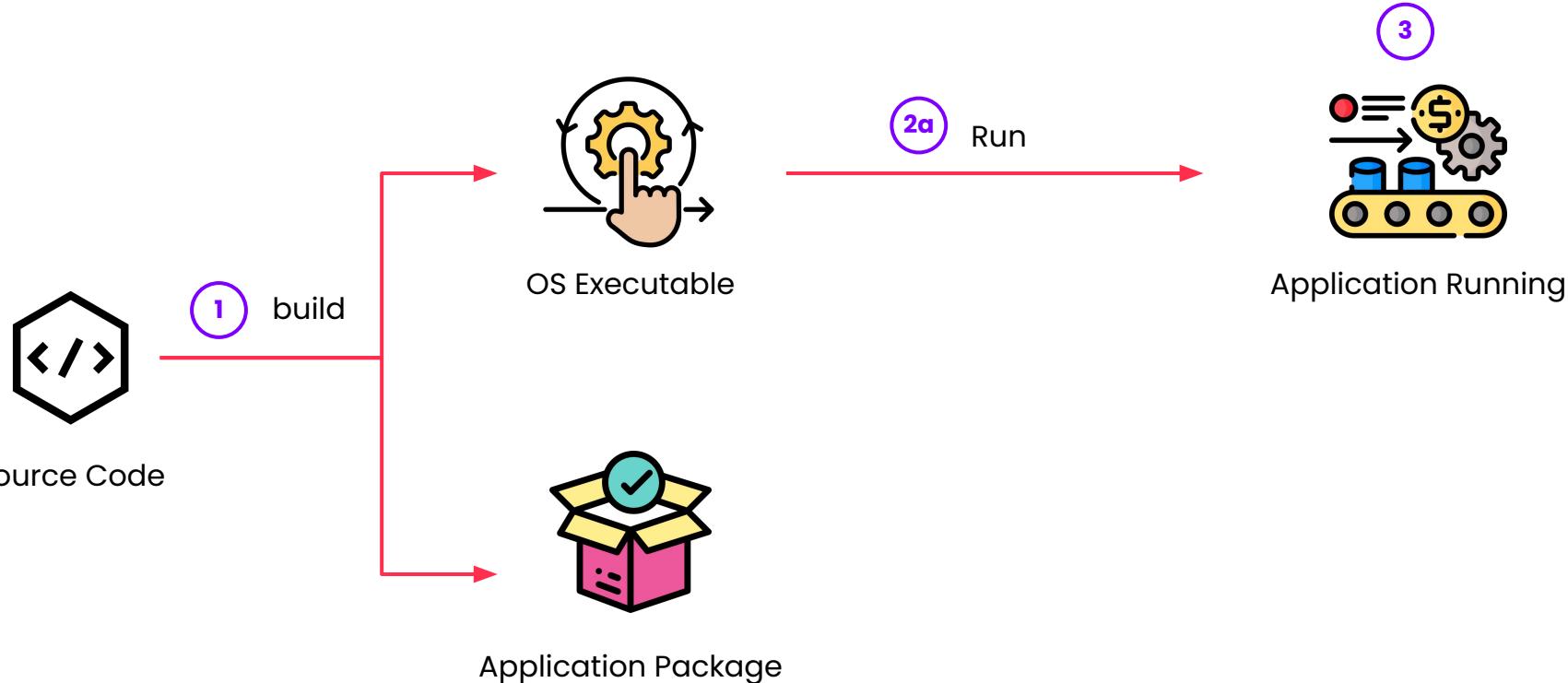
# Application Deployment Process



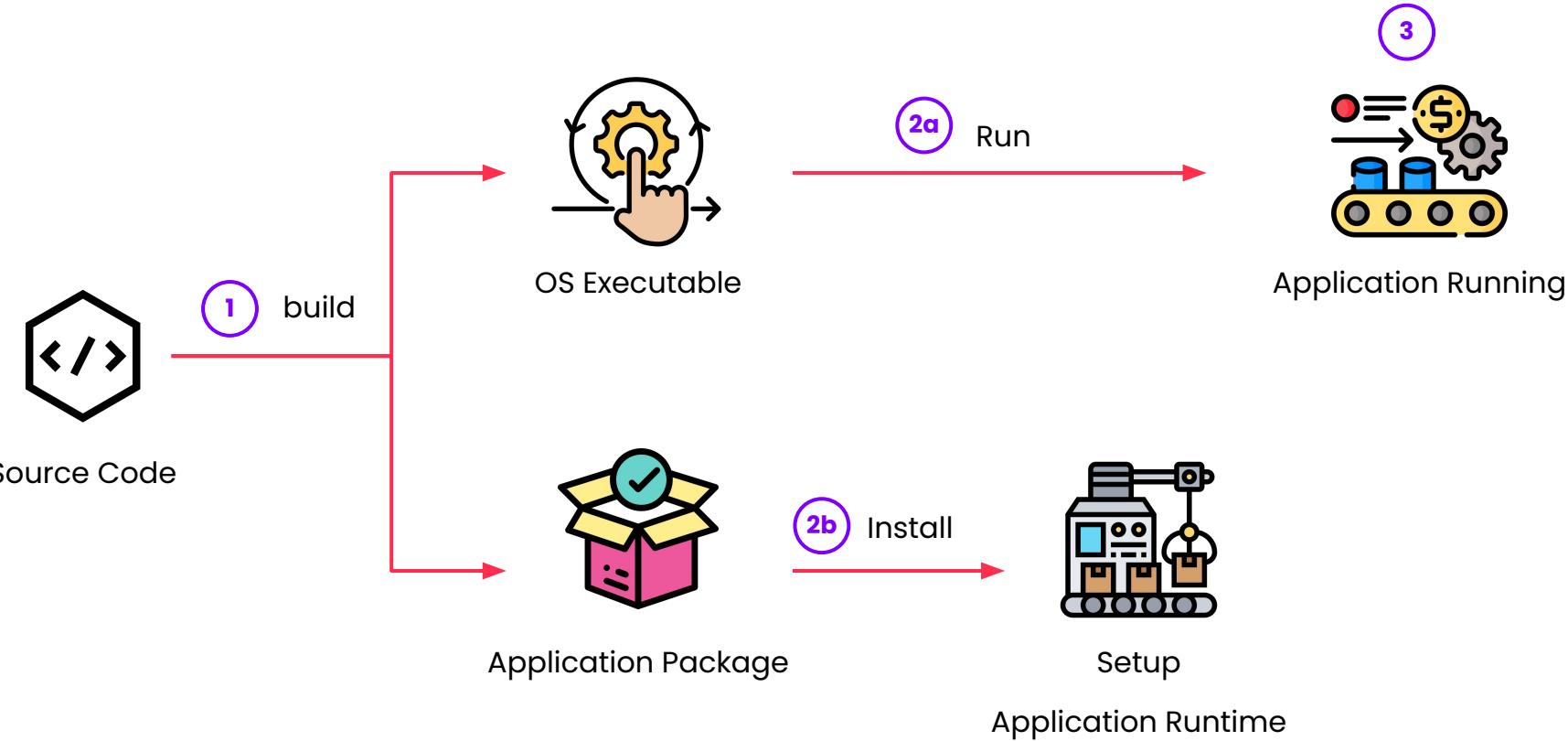
# Application Deployment Process



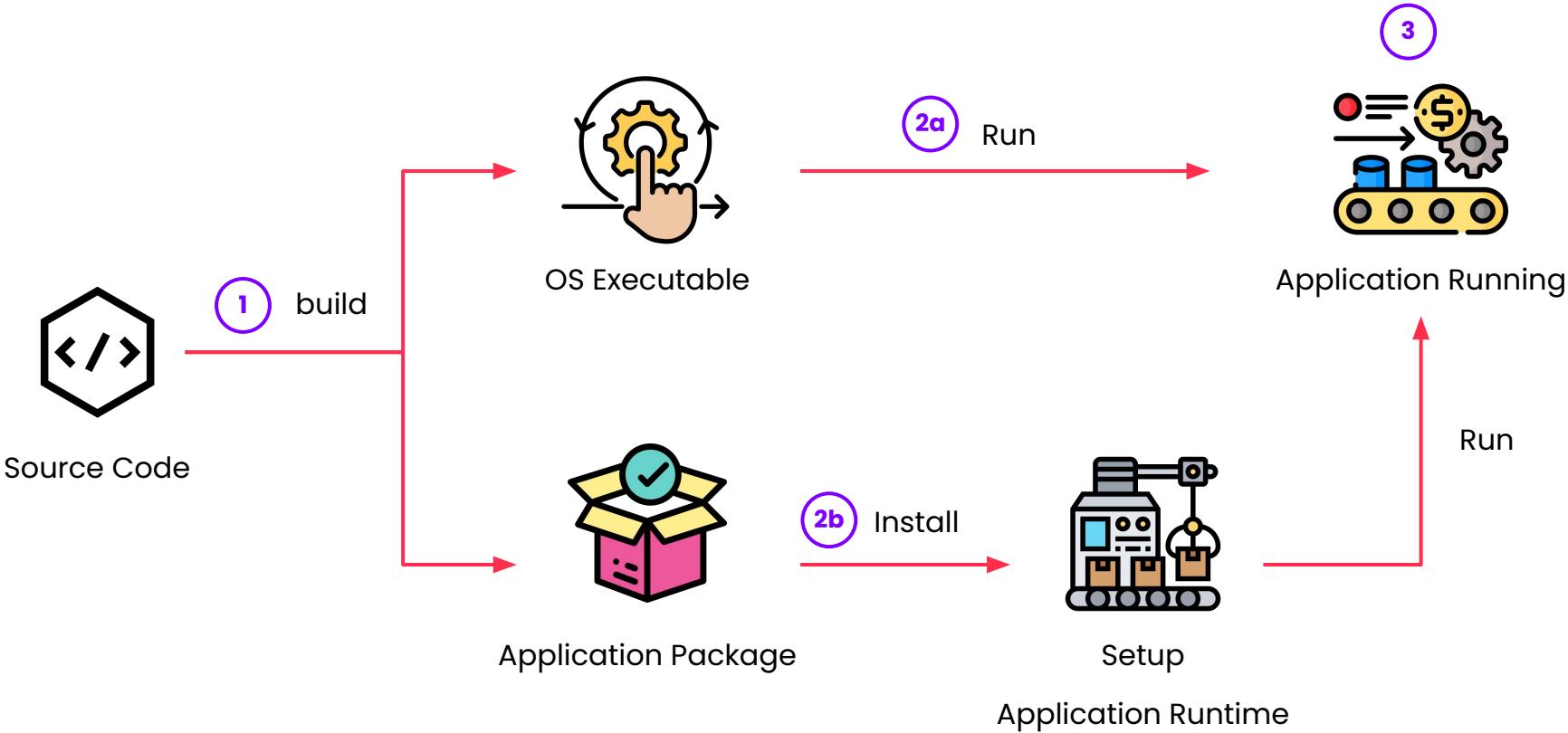
# Application Deployment Process



# Application Deployment Process



# Application Deployment Process



# Trend Application Programming with OS Executable

# Trend Application Programming with OS Executable

Node.js v20.2.0 | ► Table of contents | ► Index | ► Other versions | ► Options

▼ Table of contents

- Single executable applications
  - Generating single executable preparation blobs
  - Notes
    - `require(id)` in the injected module is not file based
    - `__filename` and `module.filename` in the injected module
    - `__dirname` in the injected module
    - Single executable application creation process
    - Platform support

**Single executable applications** #

Stability: 1 - Experimental: This feature is being designed and will change.

Source Code: [lib/internal/main/single\\_executable\\_application.js](#)

Nodejs Single executable

# Trend Application Programming with OS Executable

Node.js v20.2.0 | ► Table of contents | ► Index | ► Other versions | ► Options

▼ Table of contents

- Single executable applications
  - Generating single executable preparation blobs
  - Notes
    - `require(id)` in the injected module is not file based
    - `__filename` and `module.filename` in the injected module
    - `__dirname` in the injected module
    - Single executable application creation process
    - Platform support

## Single executable applications

Stability: 1 - Experimental: This feature is being designed and will change.

Source Code: [lib/internal/main/single\\_executable\\_application.js](#)

Nodejs Single executable

GraalVM

Docs Community Videos Blog Download

Home > Latest > Reference Manual > Native Image > Guides >

## Build a Native Executable from a JAR File

You can build a native executable from a class file, from a JAR file, or from a module. This guide demonstrates how to build a native executable from a JAR file.

To build a native executable from a JAR file in the current working directory, use the following command:

```
native-image [options] -jar jarfile [executable name]
```

Copy

Java with GraalVM

Jumpbox®

# Trend Application Programming with OS Executable

# Trend Application Programming with OS Executable

## Pros

- No Need Application Runtime
- Lightweight
- High Performance

# Trend Application Programming with OS Executable

## Pros

- No Need Application Runtime
- Lightweight
- High Performance

## Cons

- Need export to Cross Multiple Architecture and Platform

# **Summary Application Deployment Process**

## **1. OS Executable**

# **Summary Application Deployment Process**

## **1. OS Executable**

- a. Transfer Source Code
- b. Build & Install Dependencies
- c. Execute and Running

# **Summary Application Deployment Process**

## **1. OS Executable**

- a. Transfer Source Code
- b. Build & Install Dependencies
- c. Execute and Running

## **2. Need App Runtime**

- a. Transfer Source Code
- b. Install Application Runtime
- c. Build & Install Dependencies
- d. Execute and Running

# Summary Application Deployment Process



**Source Code**

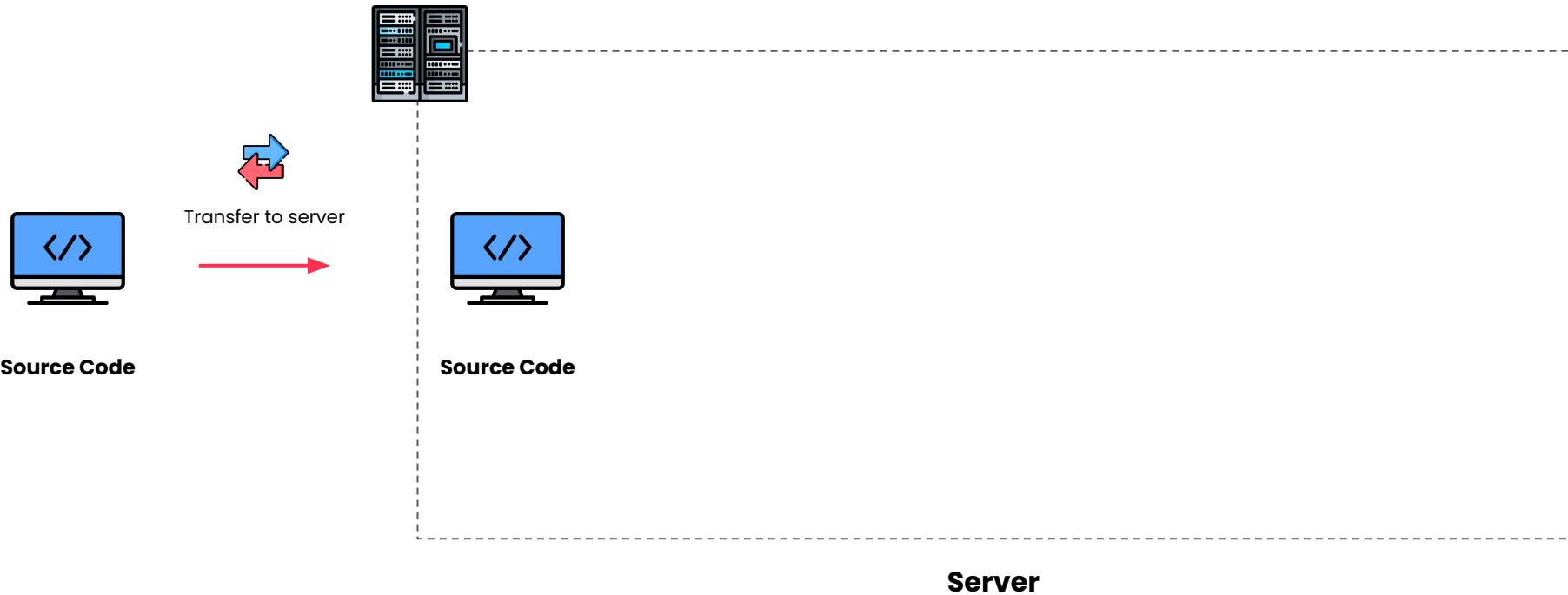
# Summary Application Deployment Process



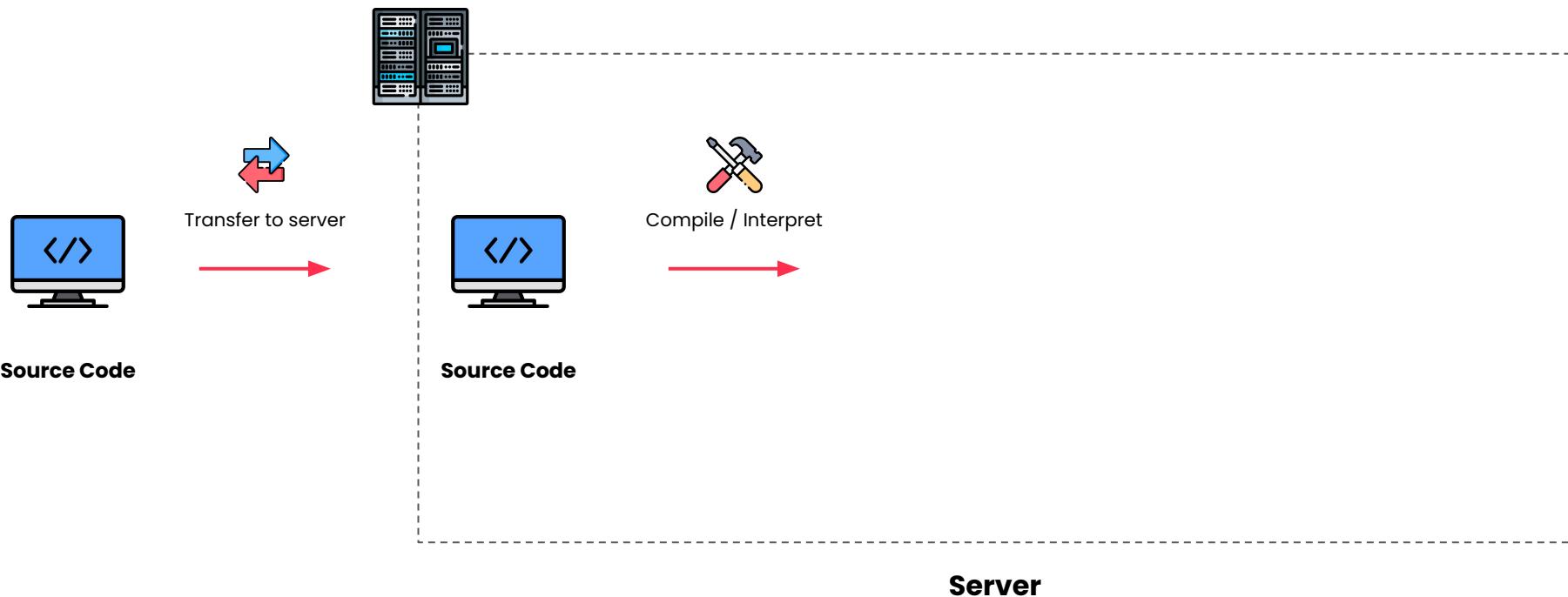
# Summary Application Deployment Process



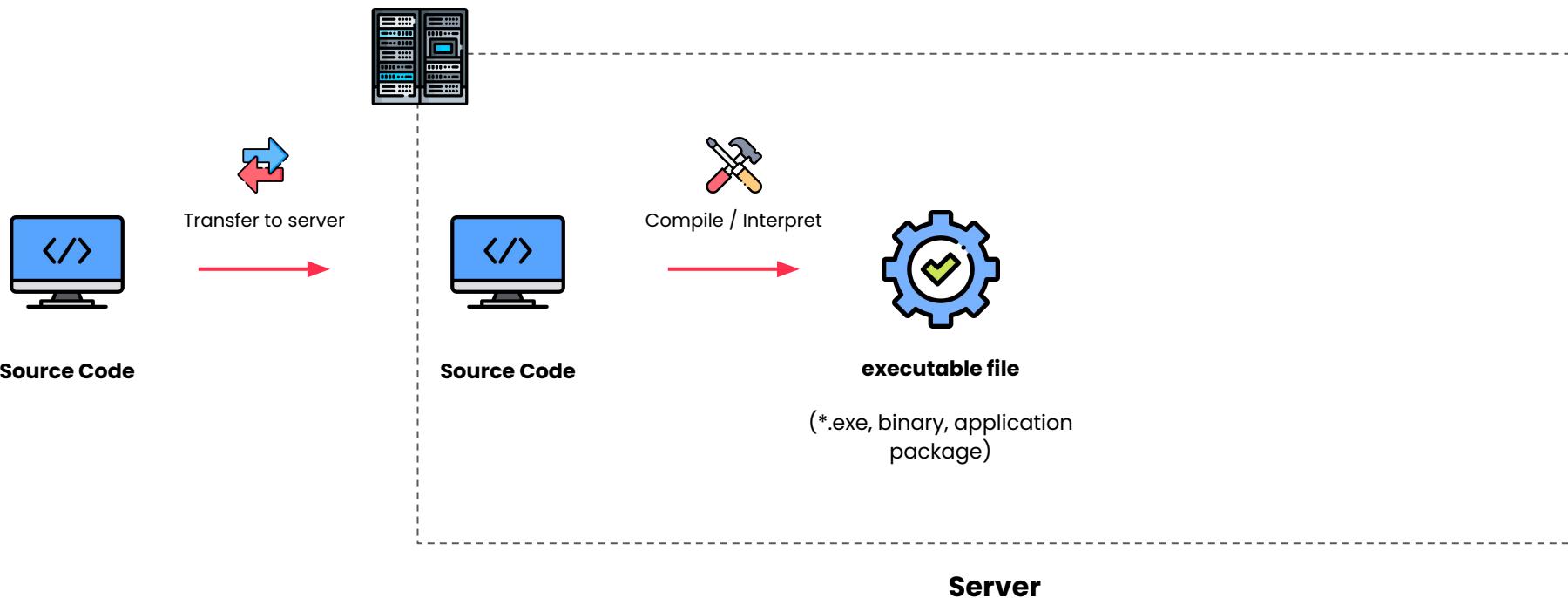
# Summary Application Deployment Process



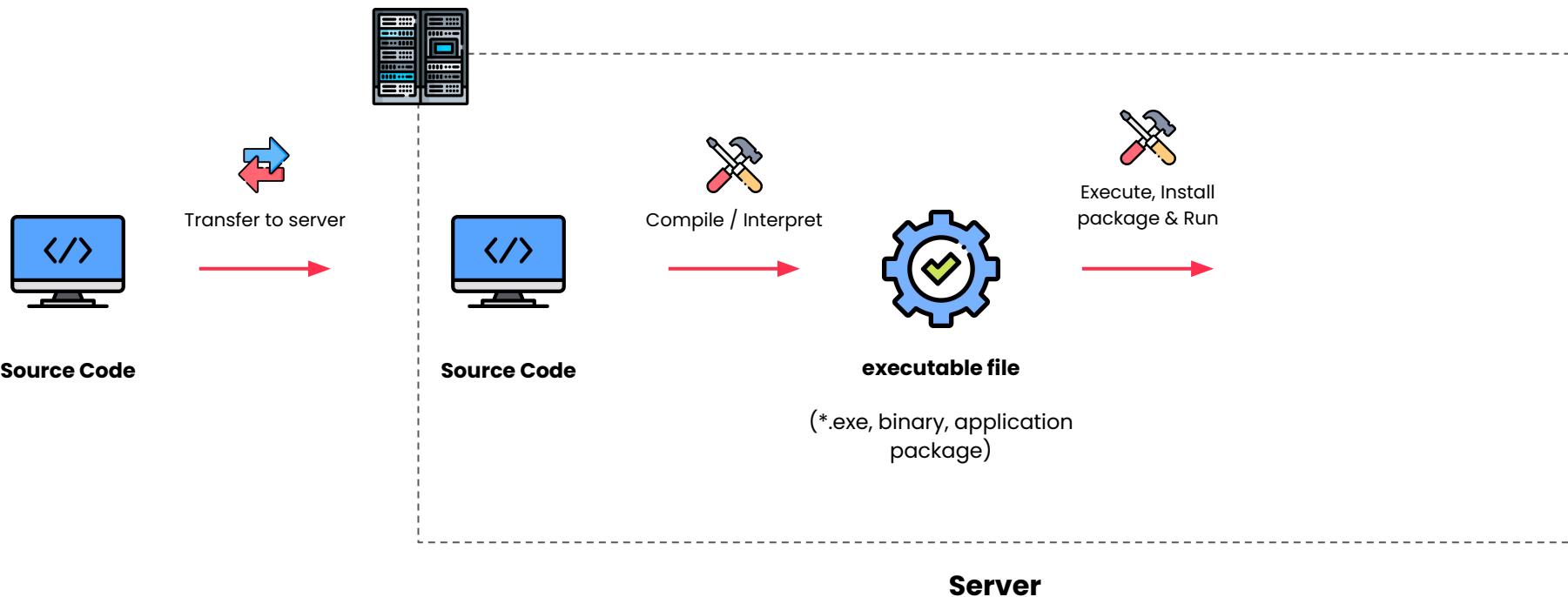
# Summary Application Deployment Process



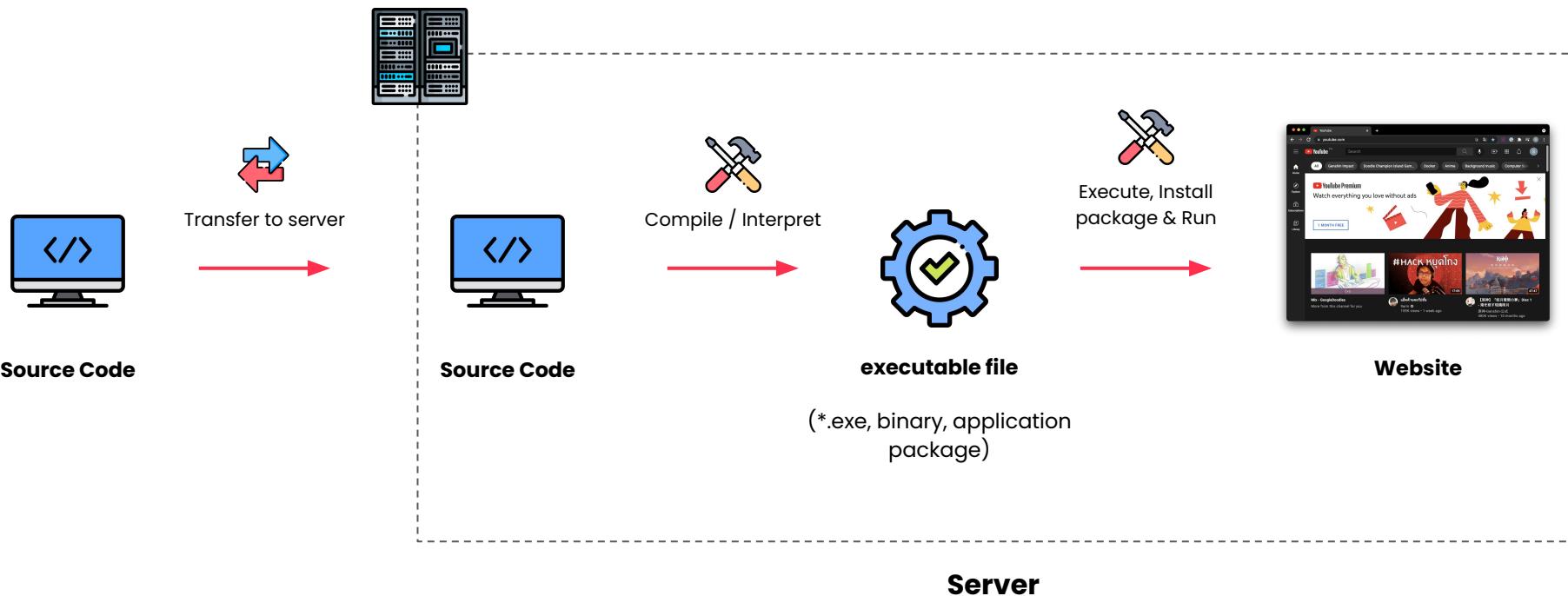
# Summary Application Deployment Process



# Summary Application Deployment Process



# Summary Application Deployment Process



# Cloud Native

Application

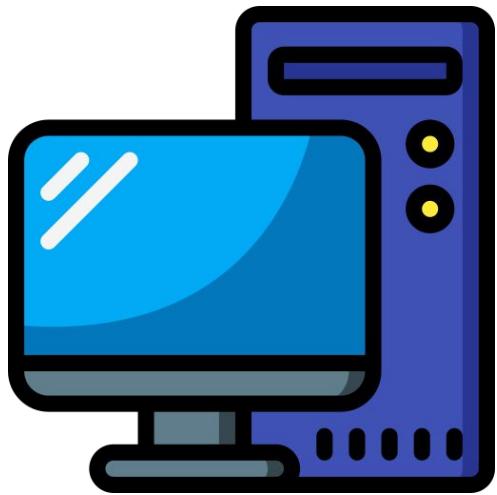
# Before we go

# Before we go

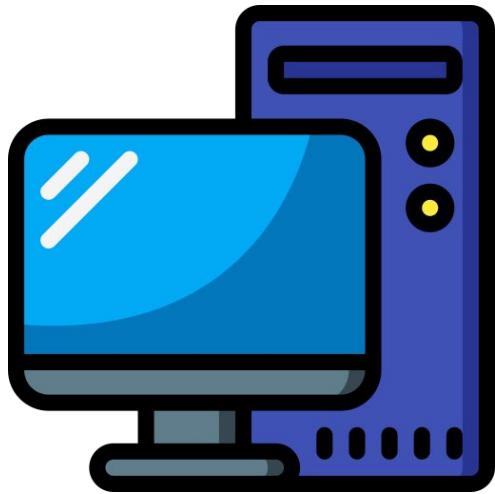
Cloud Native Application

# Cloud?

Why?

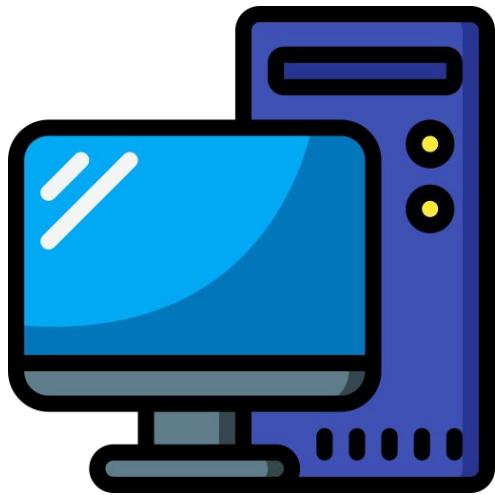


Computer



**CPU: 4 Core**

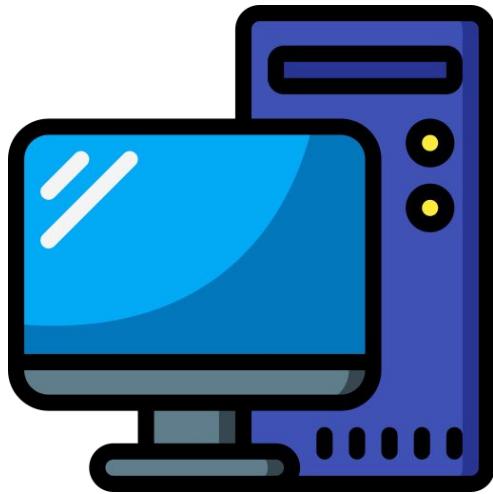
Computer



Computer

**CPU: 4 Core**

**RAM: 8 GB**

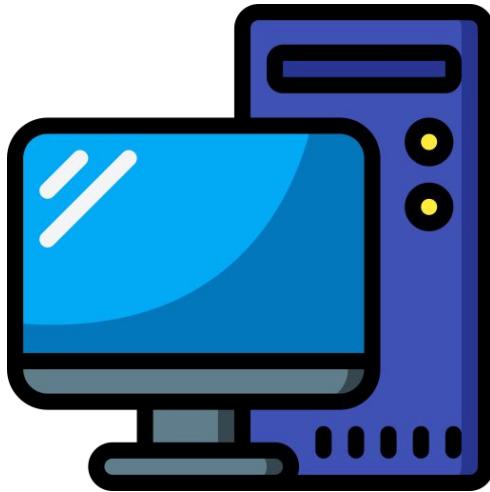


Computer

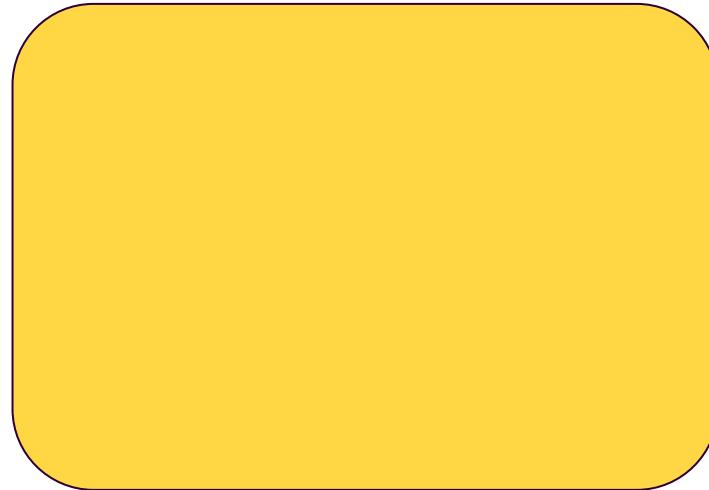
**CPU: 4 Core**

**RAM: 8 GB**

**GPU: 16 GB**



Computer



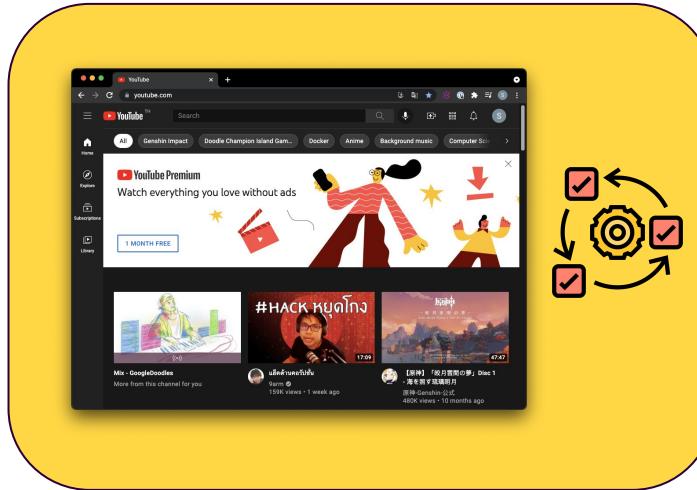
เริ่มต้นใช้งาน



**CPU 2 Ram 4**

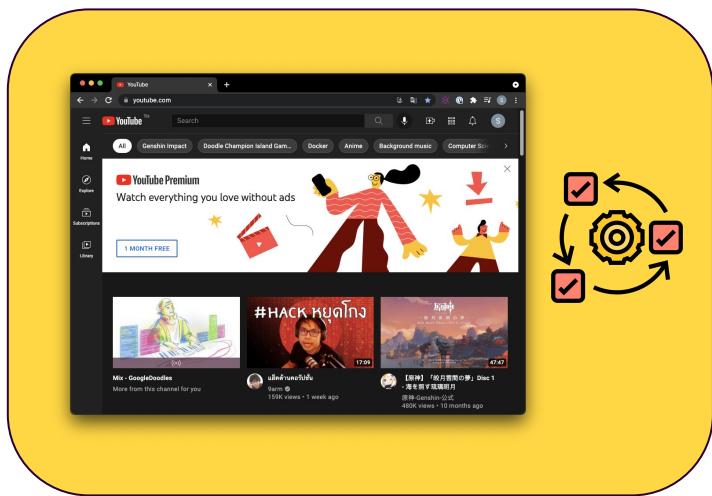


Computer



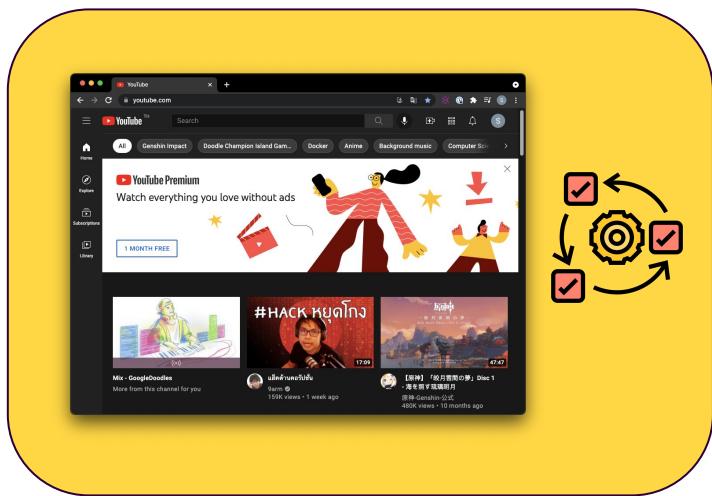
**Processing**

## CPU 2 Ram 4



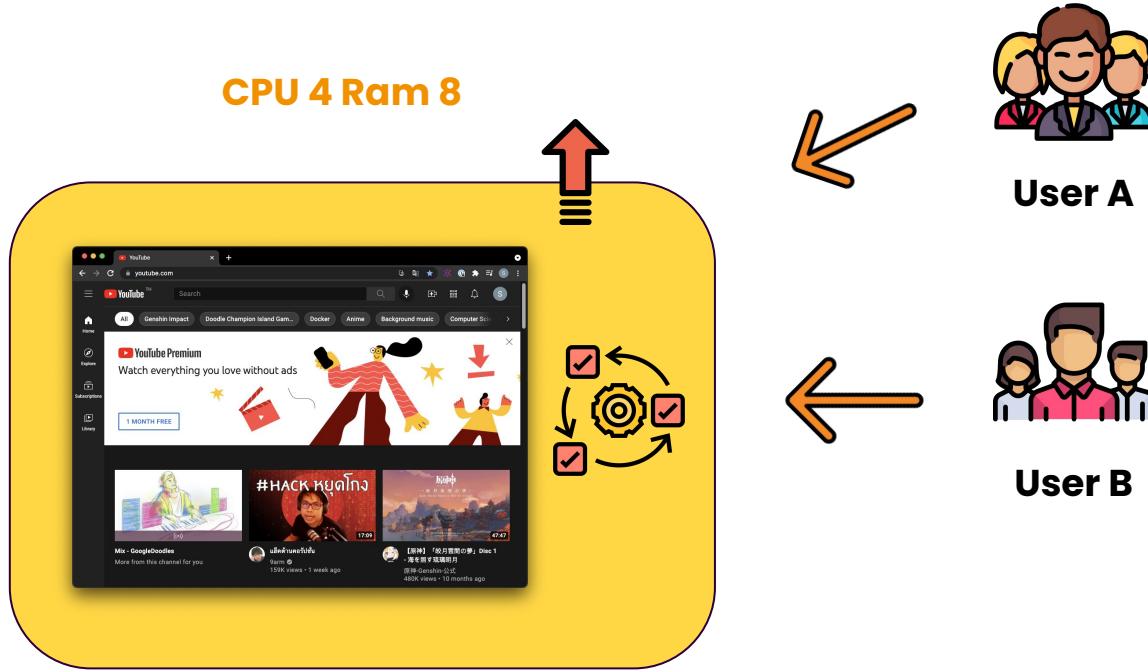
Processing

**CPU 2 Ram 4**

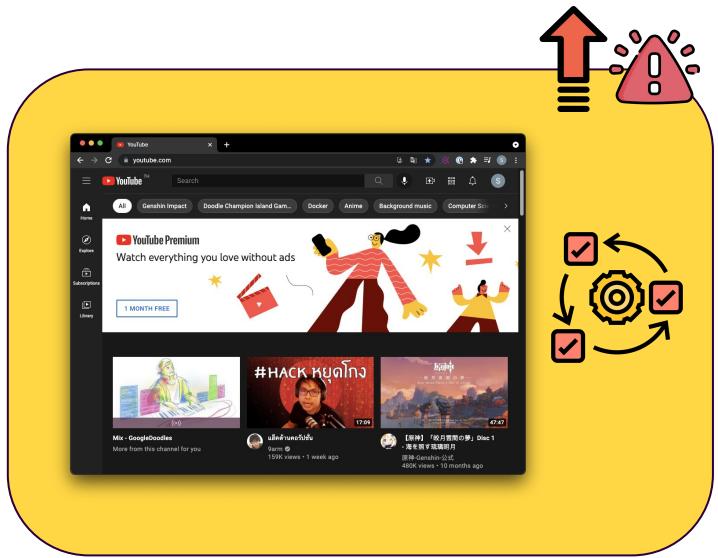


**User A**

**Processing**



**CPU 4 Ram 8**



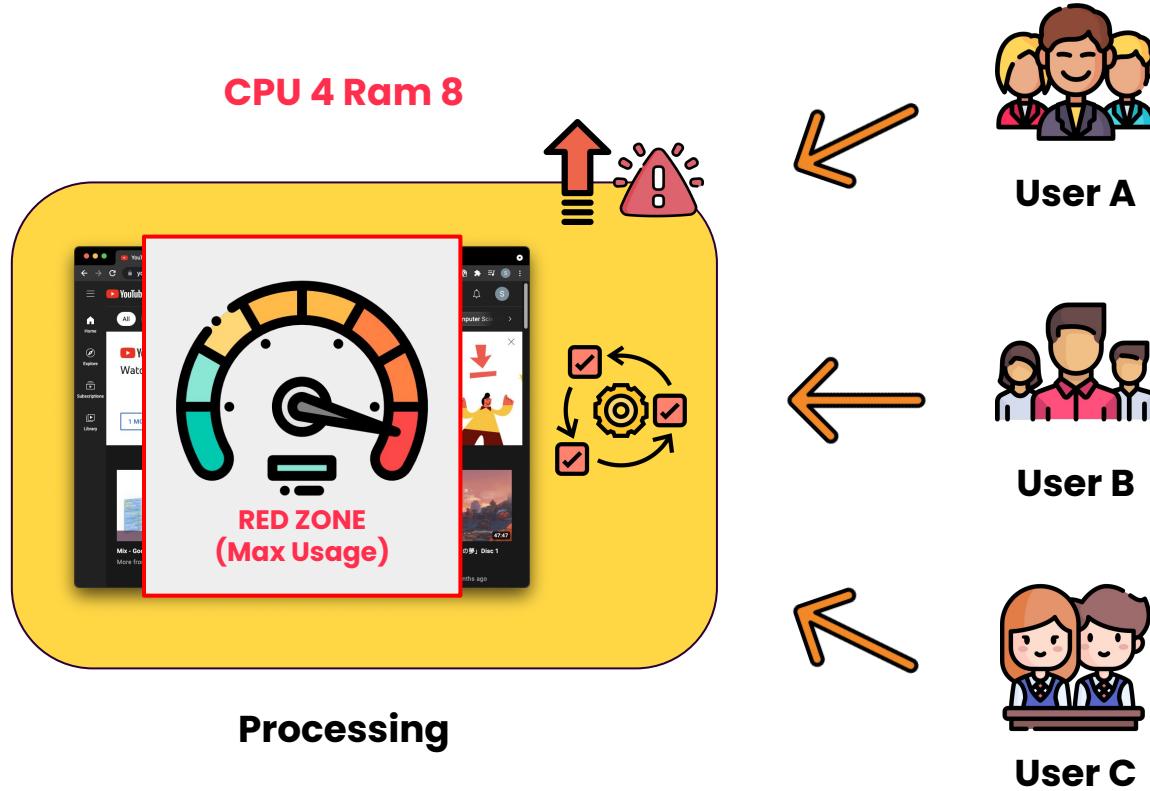
**User A**



**User B**



**User C**





Human



Human

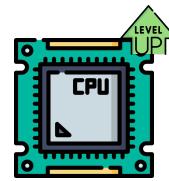
Upgrade



Human



Upgrade



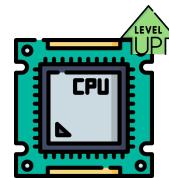
**CPU: 16**



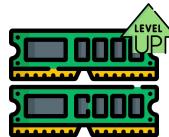
Human



Upgrade



CPU: 16



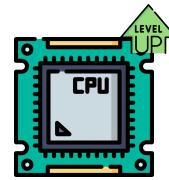
Ram: 32



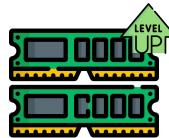
Human



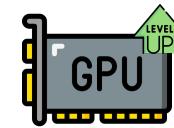
Upgrade



CPU: 16

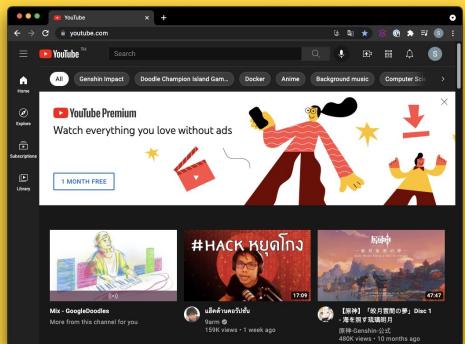


Ram: 32



GPU: 32

**CPU 4 Ram 8**



**Processing**



**User A**



**User B**



**User C**

# **Upgrade Process**



Human



Human

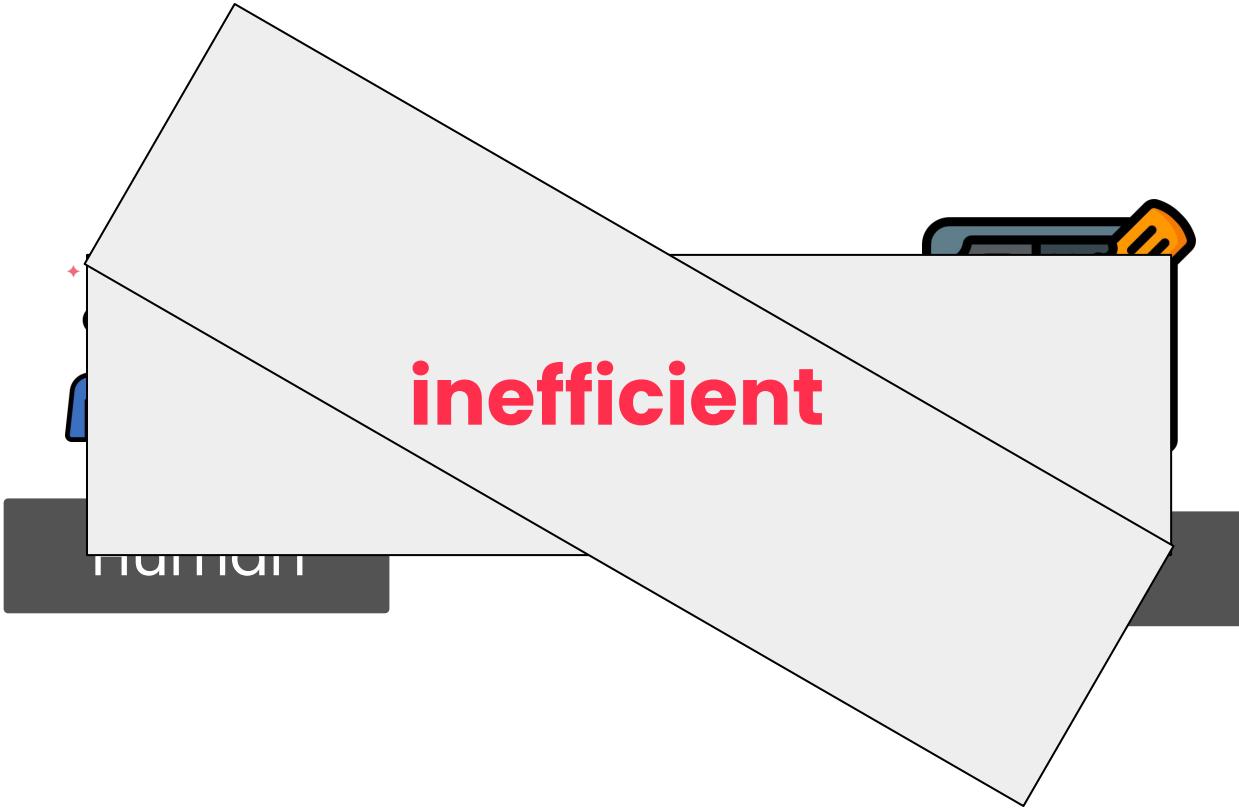
Upgrade



**slow**

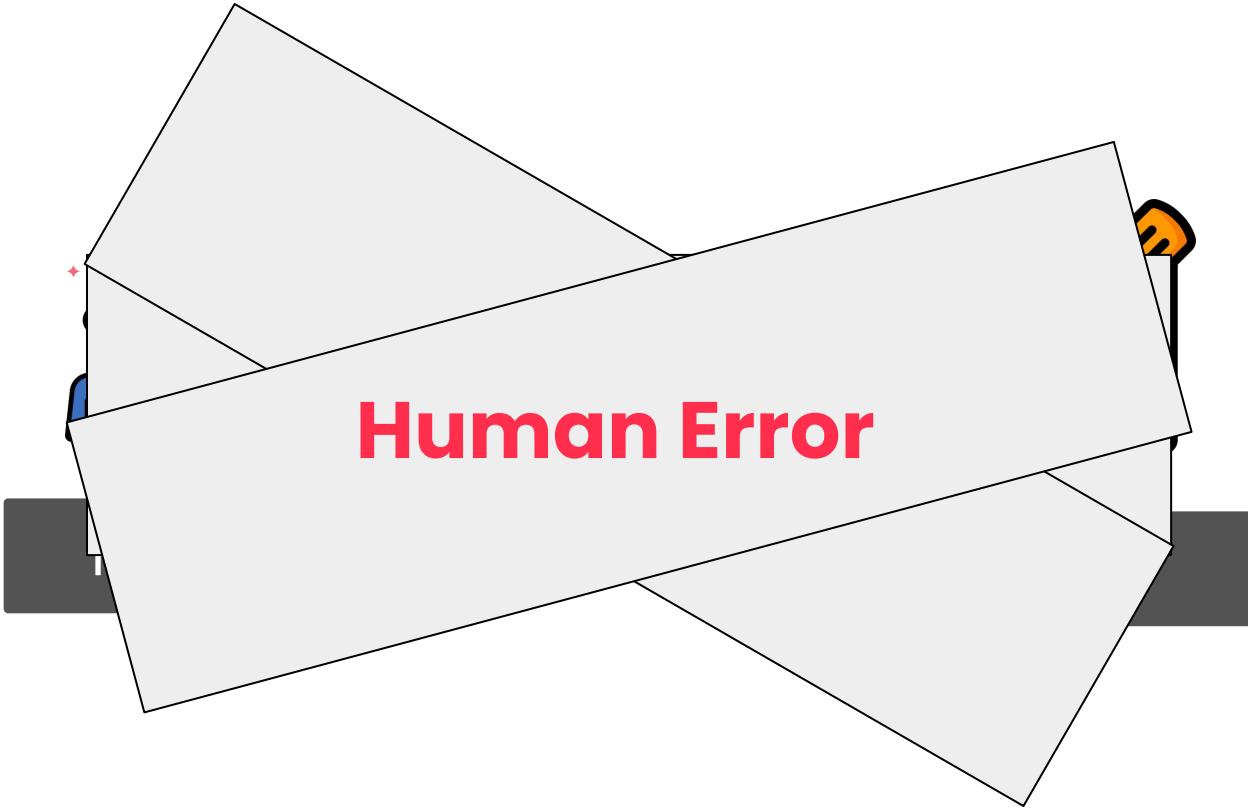
Human

Upgrade



**inefficient**

human



**Human Error**

We Evolve

Jumpbox®

We Evolve

Digital Transformation  
(Cloud Transformation)

Jumpbox®



Human



**transform**



Human



Human

**transform**



Click



Human

transform



via Internet



Click

# The beginning of Cloud ERA

**Cloud**

**Cloud as known as Cloud Computing**

## **Cloud as known as Cloud Computing**



**Reduce Cost**

# Cloud as known as Cloud Computing



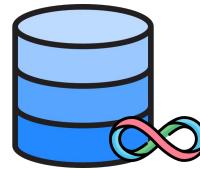
**Reduce Cost**

**Unlimited Storage**

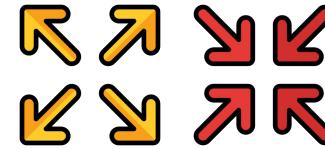
# Cloud as known as Cloud Computing



**Reduce Cost**



**Unlimited Storage**

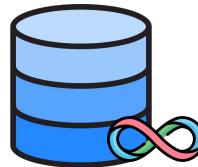


**Scalable**

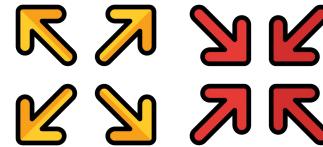
# Cloud as known as Cloud Computing



**Reduce Cost**



**Unlimited Storage**



**Scalable**

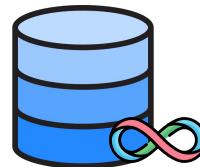


**Security**

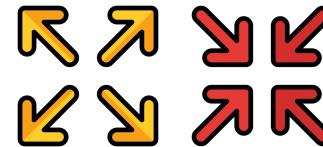
# Cloud as known as Cloud Computing



**Reduce Cost**



**Unlimited Storage**



**Scalable**



**Security**

Achieved

Frequently these are delivered by a

Frequently these are delivered by a  
**Cloud Service Provider**

# Cloud Service Provider



Jumpbox®

## Summary of Cloud

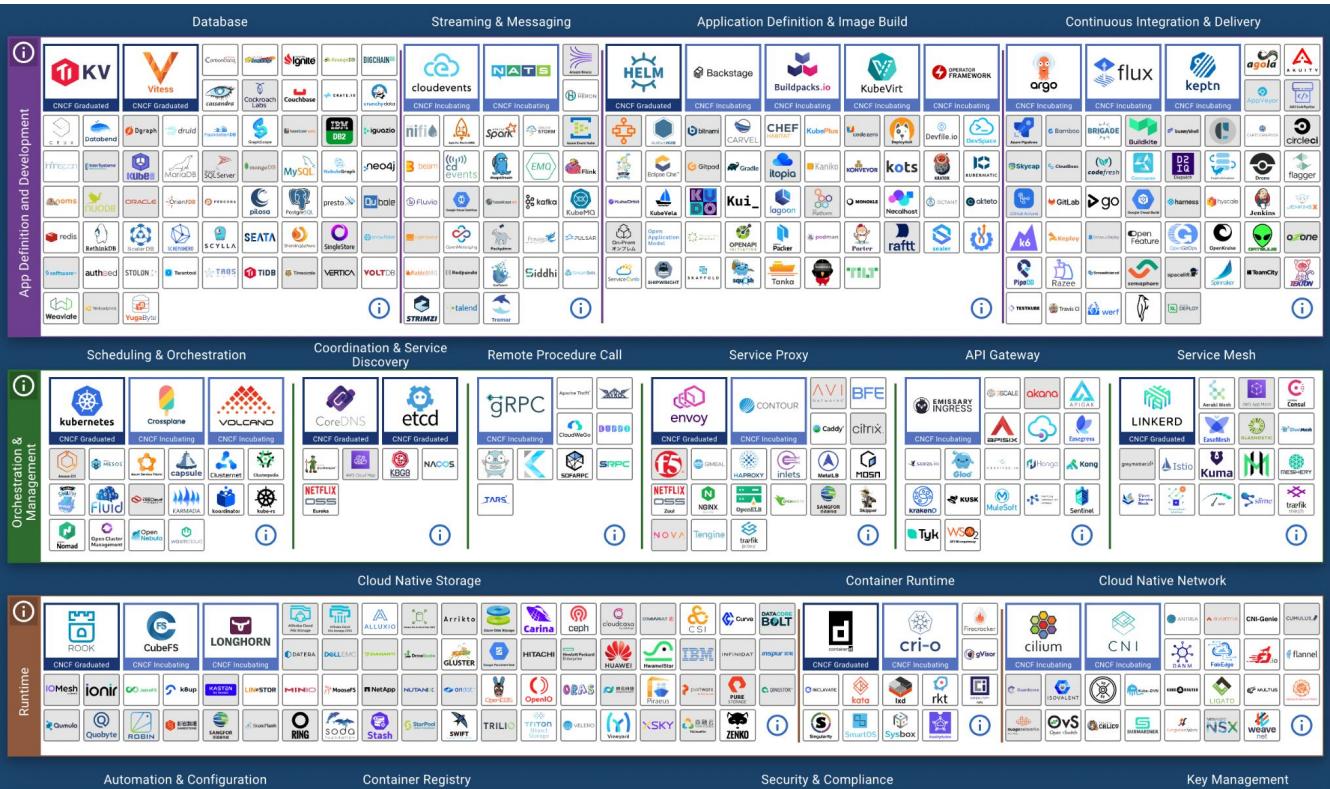
Cloud is the **on-demand** delivery of infrastructure (hardware/servers), storage, databases, and all kinds of application services **via the internet.**

# Cloud Native?

# Cloud Native?

**Is not the Cloud**

# Tools Set?



Cloud native is  
**more than** a tool set

Cloud Native?

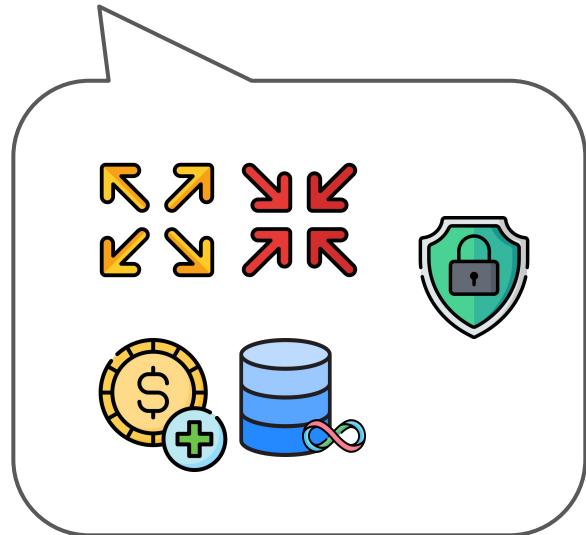
**Native = Local**

# Native Speaker

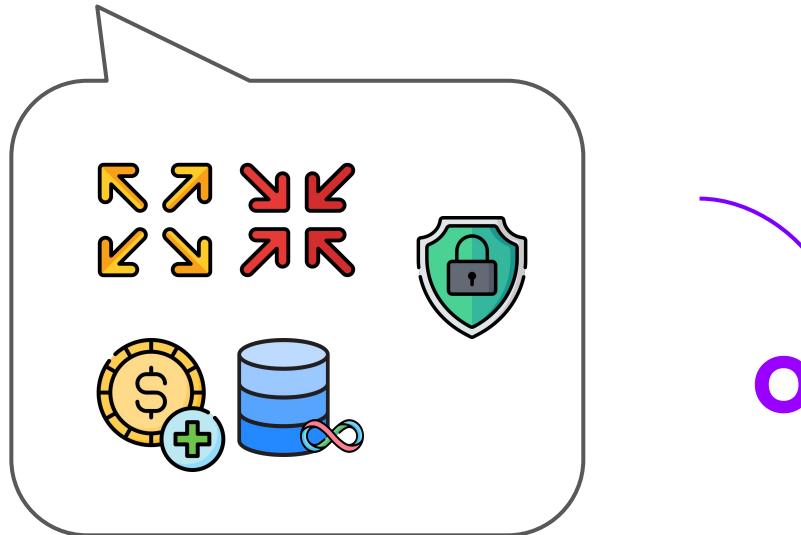
# Native Speaker

(พูดภาษาถิ่น)

# Cloud



# Cloud Native



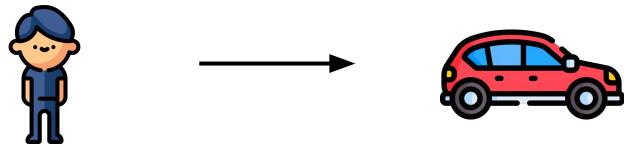
**Optimized way**

## Example

Implement CAR + Cloud Native

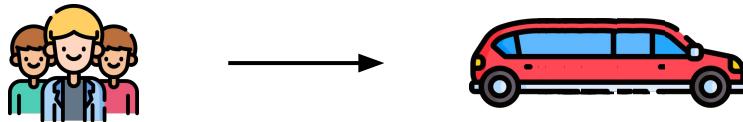
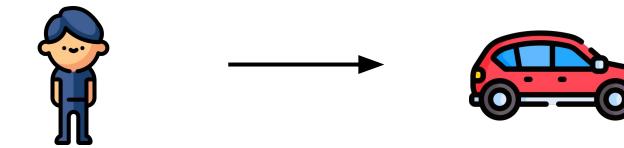
## Example

Implement CAR + Cloud Native



## Example

Implement CAR + Cloud Native

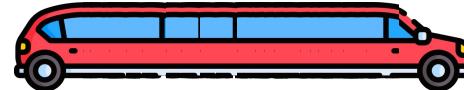
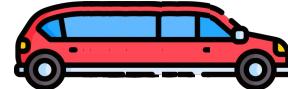


## Example

Implement **CAR**

+

**Cloud Native**

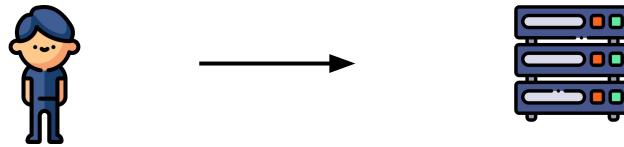


## Example

Implement **Computer** + **Cloud Native**

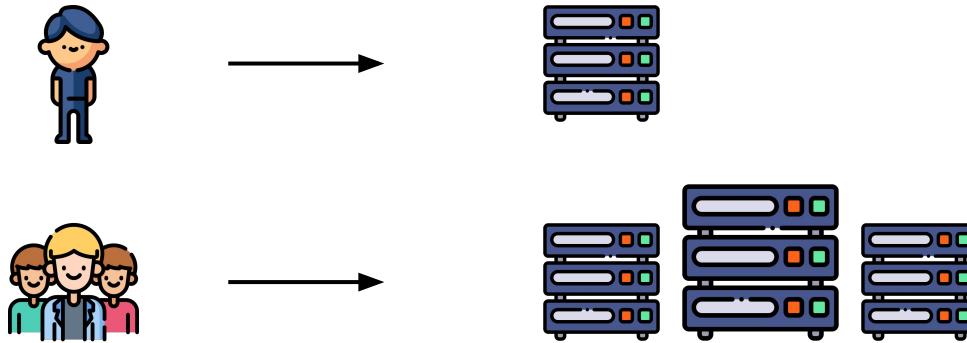
## Example

Implement **Computer** + **Cloud Native**



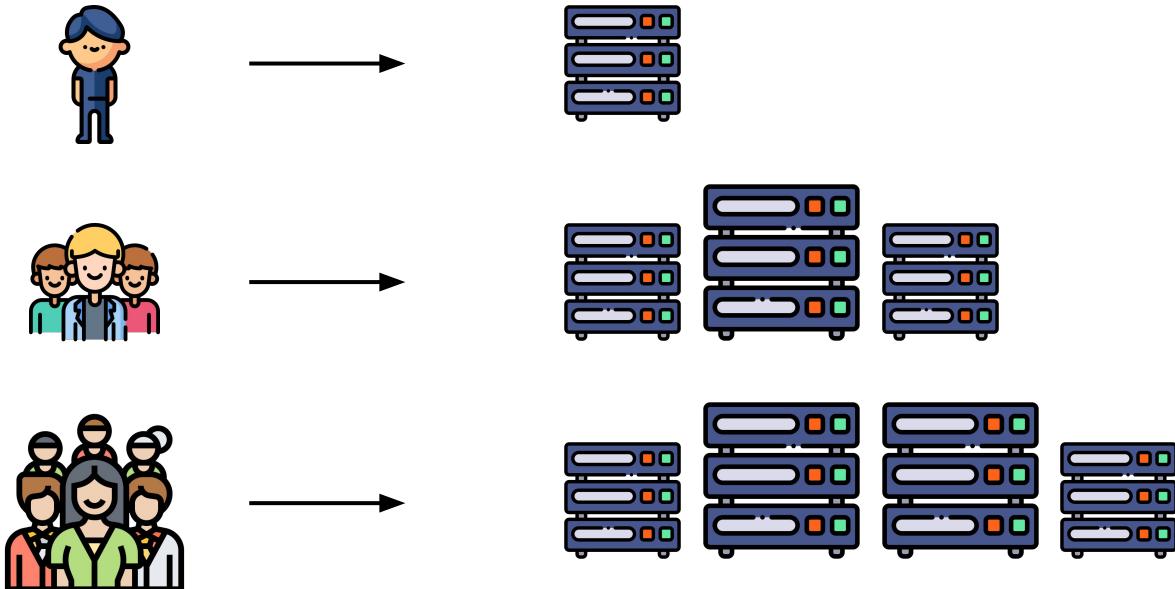
## Example

Implement **Computer** + **Cloud Native**

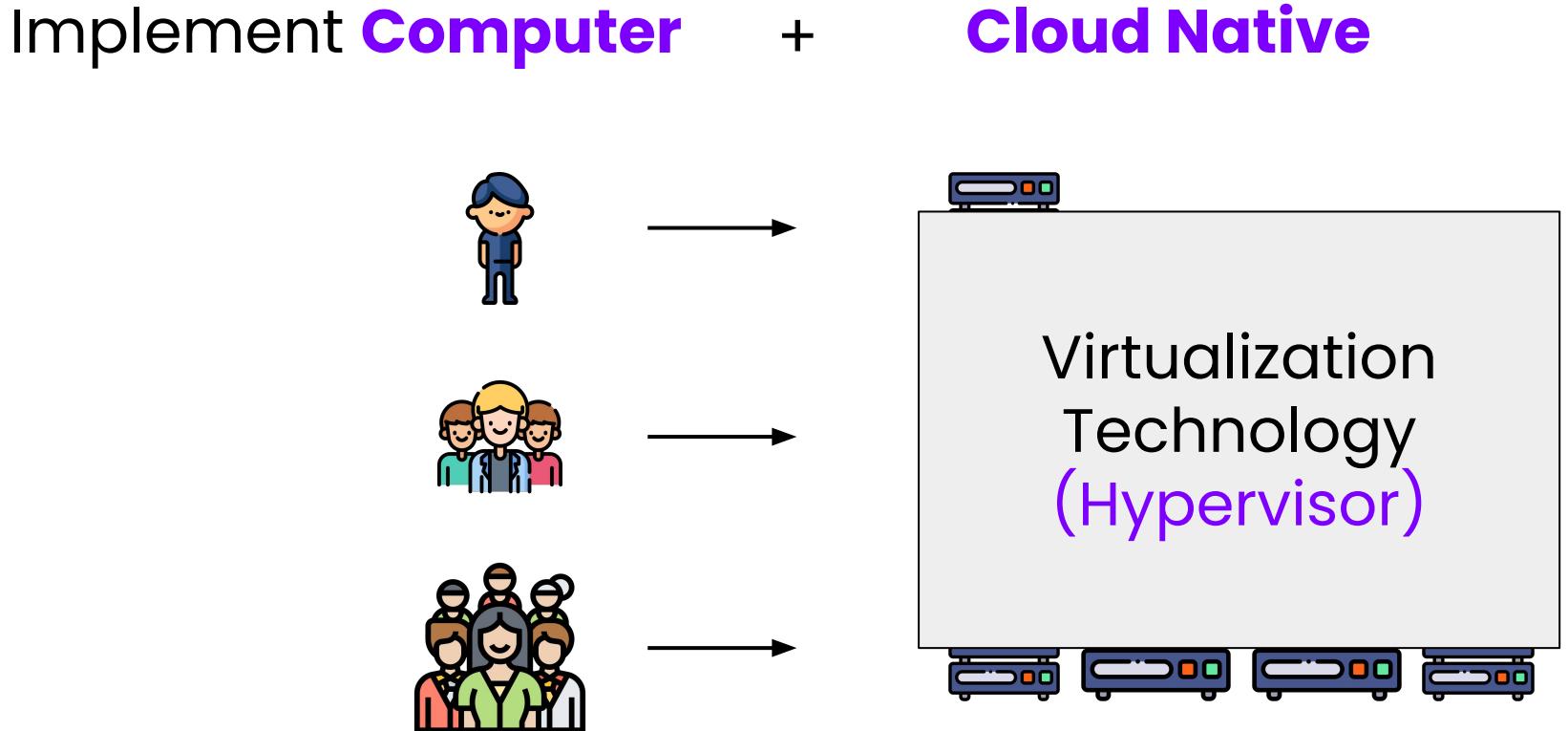


## Example

Implement **Computer** + **Cloud Native**



## Example



Do you know?

Do you know?

Thailand has Cloud Native Product  
for a long time

# The Jar

## Example

Implement Jar + Cloud Native



# The Jar

Thailand's local tool that can store every people

# The Jar

Thailand's local tool that can store every people  
who run away from ผีปอบ

# Adapt **Thai Local Tools** with Cloud Native

Implement **Jar**

+

**Cloud Native**

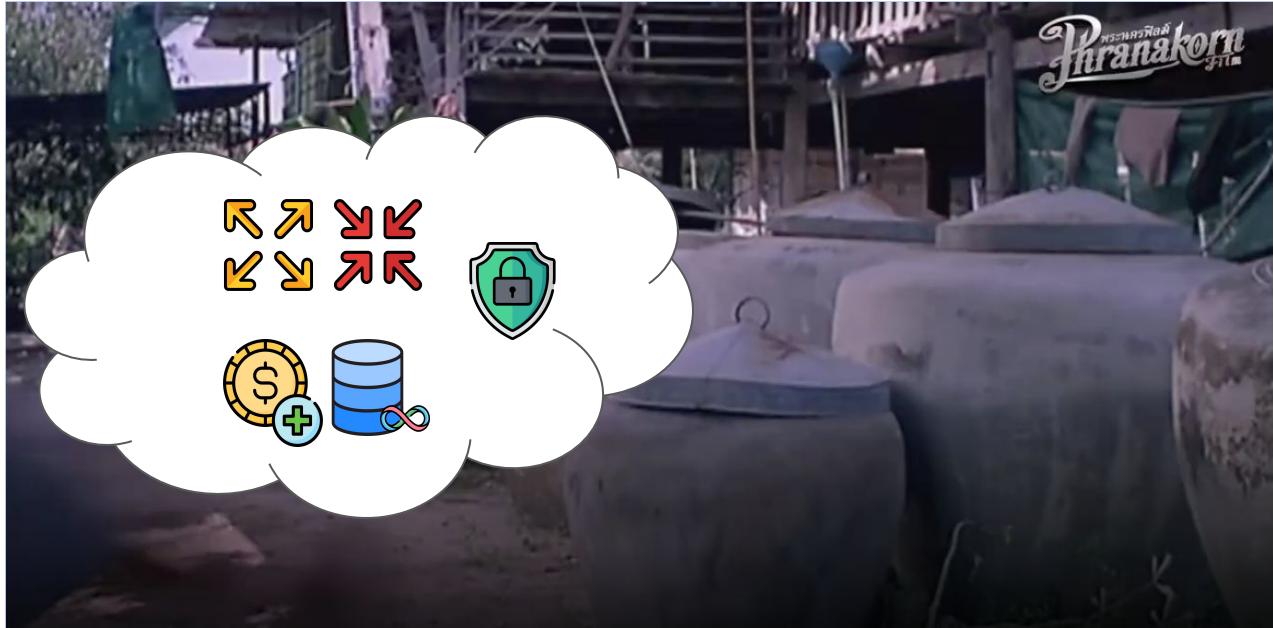


# Adapt Thai Local Tools with Cloud Native

Implement Jar

+

Cloud Native

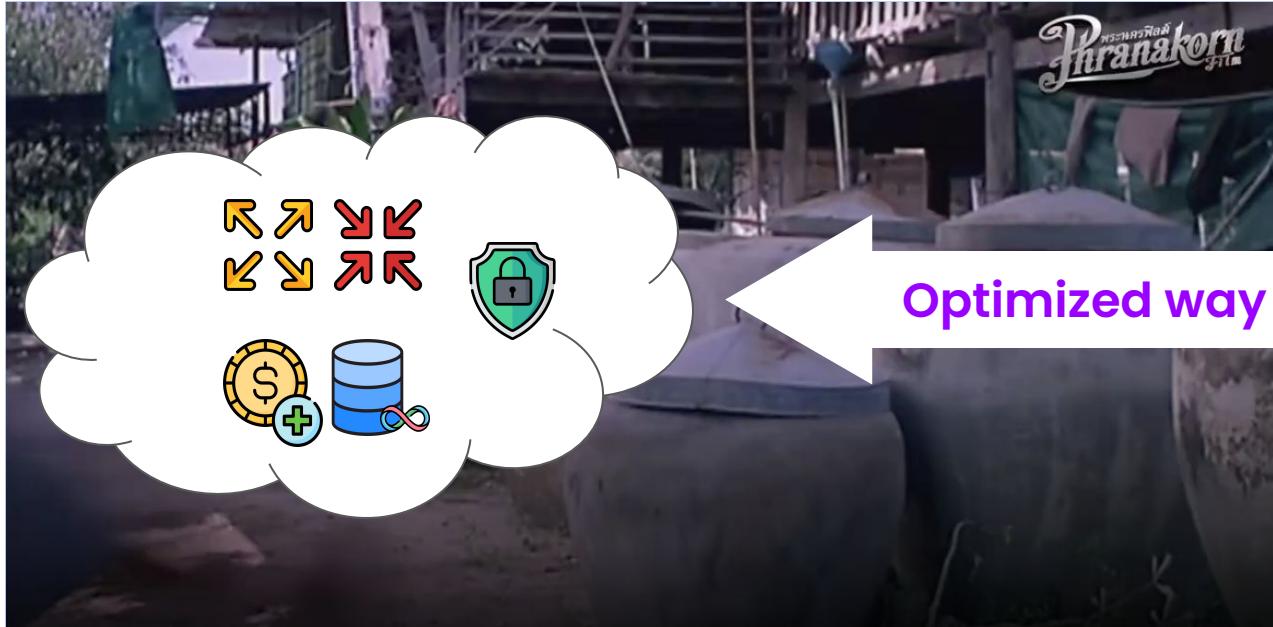


# Adapt Thai Local Tools with Cloud Native

Implement Jar

+

Cloud Native



# Cloud Native

# Cloud Native

for Application

# Summary of Cloud Native - Application way

# Summary of Cloud Native - Application way

It is a complete architecture

## Summary of Cloud Native - Application way

It is a complete architecture, a philosophical approach for

## Summary of Cloud Native - Application way

It is a complete architecture, a philosophical approach for **building applications** that **take full advantage of cloud computing.**

## Example

Implement App

+

Cloud Native



Web App



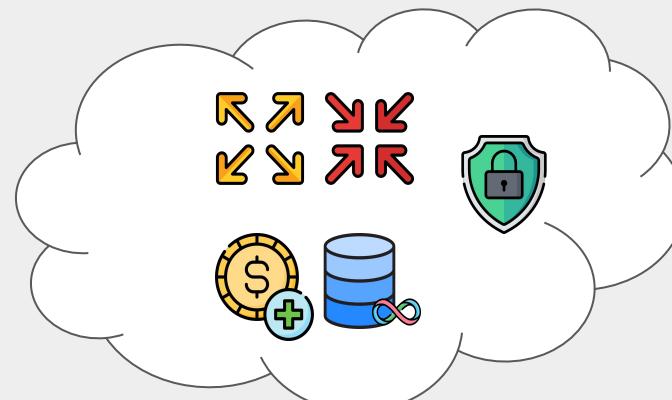
iOS App



Android App



Linux App



Optimized way

# How?

# Application Characteristic

# Output

# Output (ผลลัพธ์)

# Tree Output

# Tree Output



**Tree**

# Tree Output



**Tree**

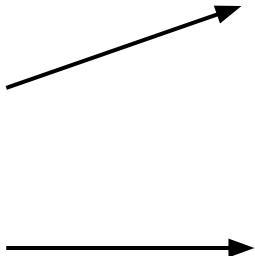


**Gas**

# Tree Output



Tree



Gas

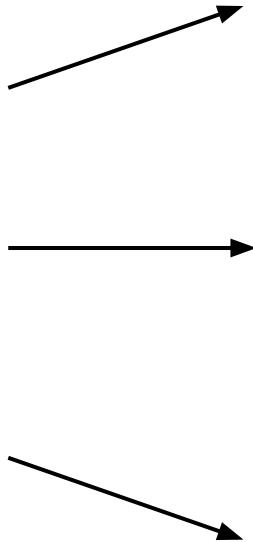


Leaf

# Tree Output



Tree



Gas



Leaf



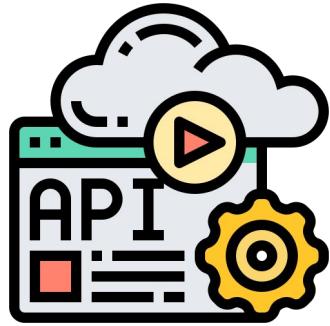
Fruit

# Application Output

Maintain **Things** that your app **continues to produce**

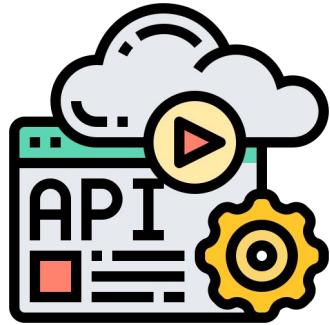
# Application Output

# Application Output



App

# Application Output

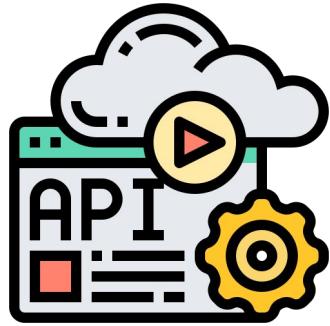


App

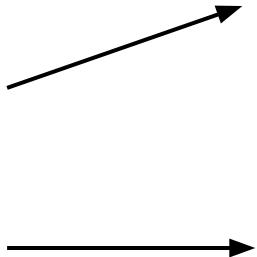


Logs

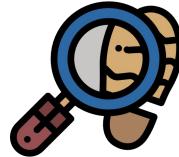
# Application Output



App



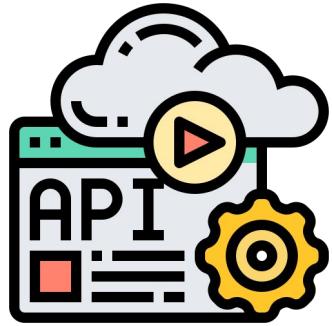
Logs



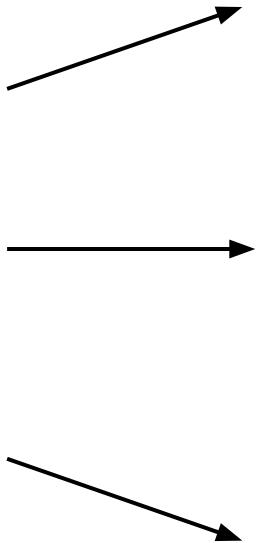
Traces

Jumpbox®

# Application Output



App



Logs



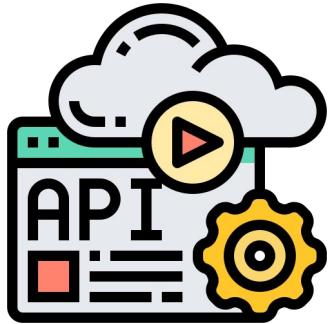
Traces



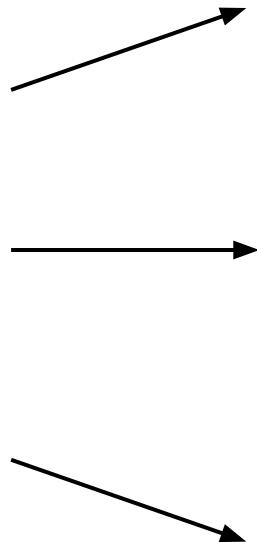
Metrics

Jumpbox®

# Application Output



App



Logs



Traces



Metrics



Grafana loki



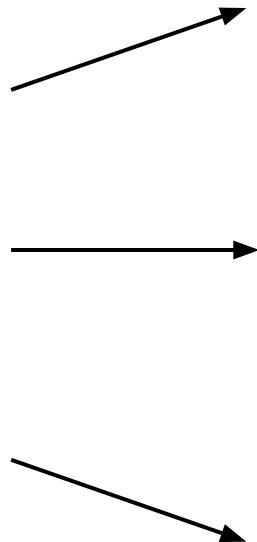
elasticsearch

Jumpbox®

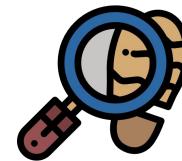
# Application Output



App



Logs



Traces



Metrics



Grafana loki



elasticsearch



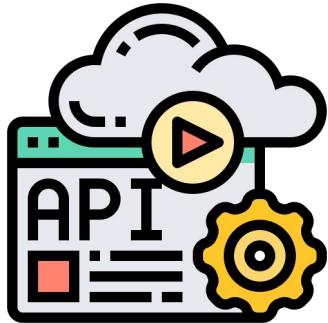
Grafana Tempo



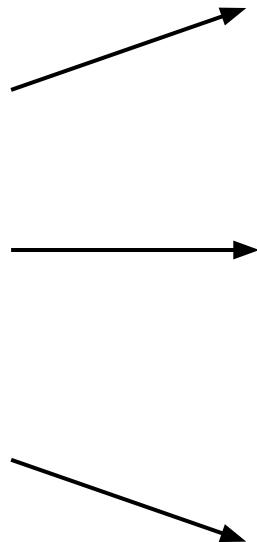
elastic apm

Jumpbox®

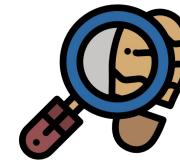
# Application Output



App



Logs



Traces



Metrics



Jumpbox®

# Observability

# Observability

Observe + ability

# Observe

**Observe** = ลังเกต

# Ability

**Ability** = ความสามารถ

# Observability

ความสามารถในการสังเกต

Telemetry



# Observability

ความสามารถในการสังเกต

# For what?



รู้ว่าผู้ใช้งานเข้ามาในระบบ  
ว่าอยู่ส่วนไหน (Trace)

# For what?



รู้ว่าผู้ใช้งานเข้ามาในระบบ  
ว่าอยู่ส่วนไหน (Trace)



ทำอะไรบ้าง (Log)

# For what?



รู้ว่าผู้ใช้งานเข้ามาในระบบ  
ว่าอยู่ส่วนไหน (Trace)



ทำอะไรบ้าง (Log)

# For what?



ทำแล้วกี่ครั้ง / ใช้งานกี่คน (Metric)

# Telemetry

# Telemetry

Tele (ไกล) + Metron (การวัด)

# Telemetry

Tele (ไกล) + Metron (การวัด)

นั่นหมายถึง กระบวนการส่งข้อมูลการวัดจากที่ห่างไกล เพื่อนำมาวิเคราะห์  
ก่อนเราจะมีความสามารถในการสังเกตได้ หรือที่เรียกว่า Observability

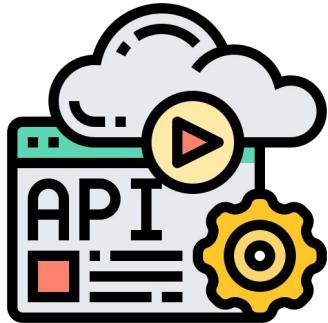
# How to Get Started

with code

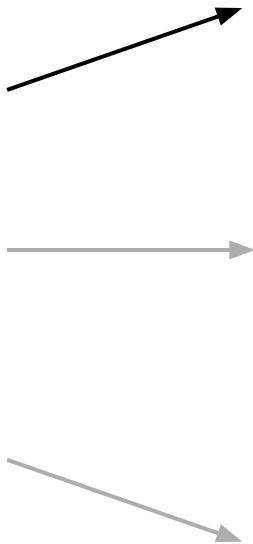


# Logs

## Application Output



App



Logs



Grafana loki



elasticsearch



Traces



Grafana Tempo



elastic apm



Metrics



Grafana  
Mimir



Prometheus

Jumpbox®

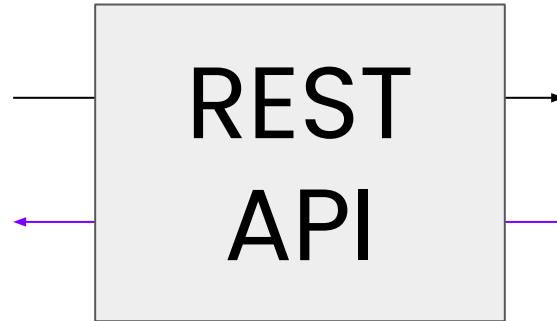
# Application Scenarios



# Application Scenarios



# Application Scenarios



# Application Scenarios



```
{  
  "timestamp": "2023-06-15T10:00:00+07:00",  
  "message": "Unable to connect to payment provider!",  
  "level": "INFO",  
  "trace_id": "4e30f7340b3fb631,4e30f7340b3fb631",  
  "service": "web-api",  
  "meta_data": {  
    "error_code": "ECUS001",  
    "error_msg": "Invalid API Token",  
    "user_name": "JoJo",  
    "user_id": "C168"  
  },  
  "data": {  
    "key_1": "value",  
    "key_2": "value-2",  
    "key_3": "value-3"  
  }  
}
```



Standard Logging

Jumpbox®

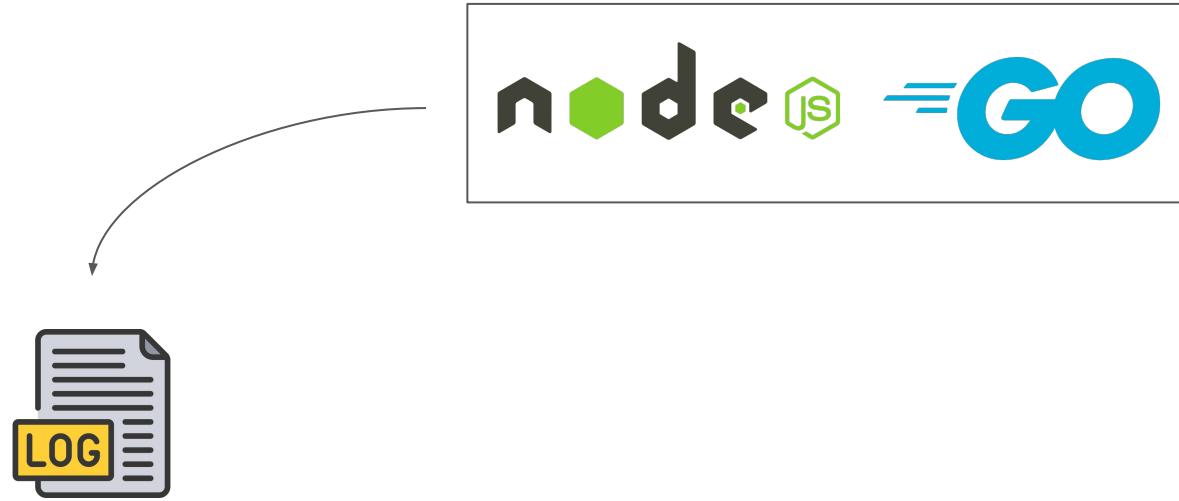
# Centralized Logging

Tooling Stack

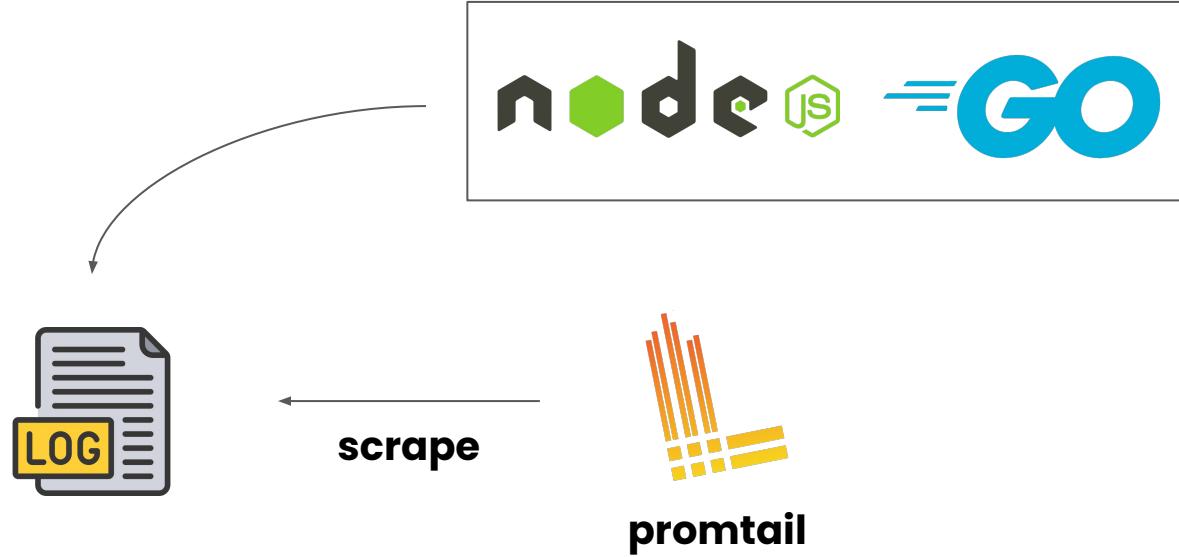
# Logging Scenarios



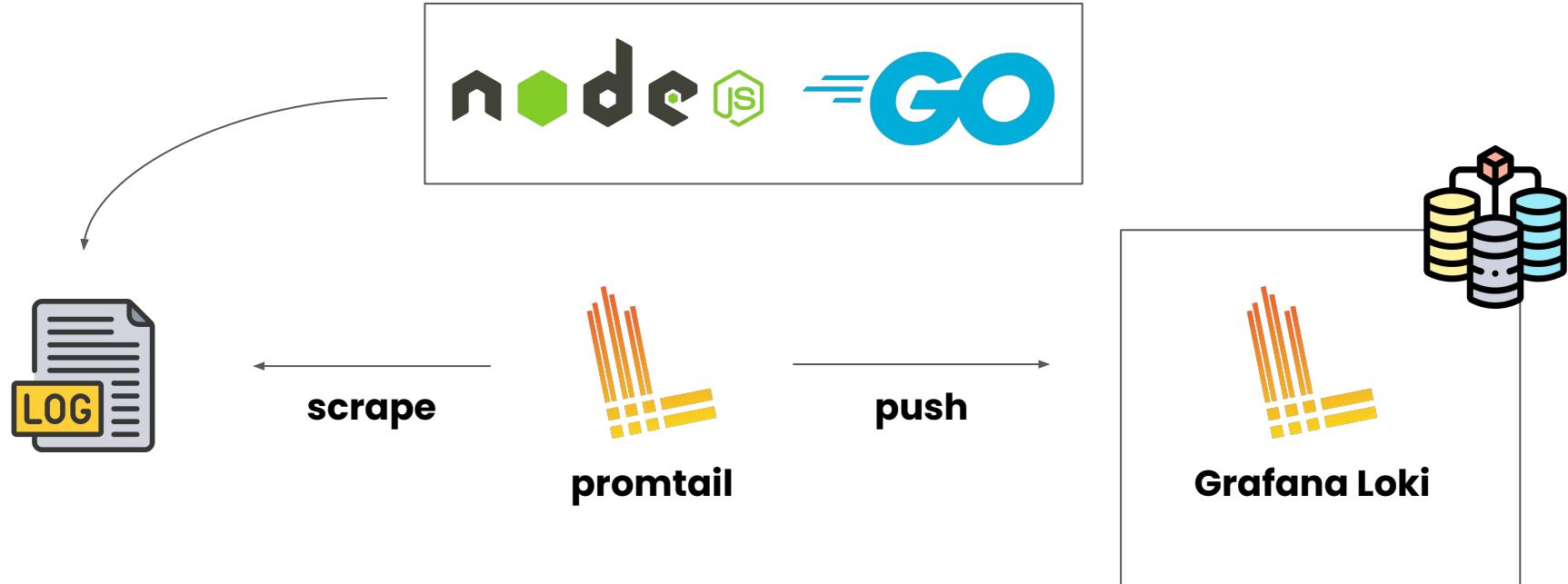
# Logging Scenarios



# Logging Scenarios



# Logging Scenarios



Centralized Logging

# Logg

Explore - Loki - Grafana x +

jumpbox.academy

Search or jump to... cmd+k

Home > Explore

Logs

Time  Unique labels  Wrap lines  Prettify JSON  Deduplication  None Exact Numbers Signature

Display results Newest first Oldest first

Common labels: value value-2 value-3 app-log web-api C168 JoJo Line limit: 1000 (8 returned) Total bytes processed: 6.66 kB

Download

```
> 2023-06-15 20:16:57.517 {
    "timestamp": "2023-06-15T14:48:31+07:00",
    "message": "Secret is: Jumpbox",
    "level": "INFO",
    "service": "web-api",
    "meta_data": {
        "user_id": "C168",
        "user_name": "JoJo"
    },
    "data": {
        "key_1": "value",
        "key_2": "value"
    }
}
```

Start of range 20:16:57



Jumpbox®

# Let's do the pipeline



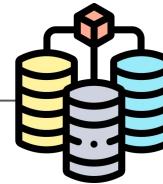
**promtail**

Jumpbox®

# Let's do the pipeline



**promtail**

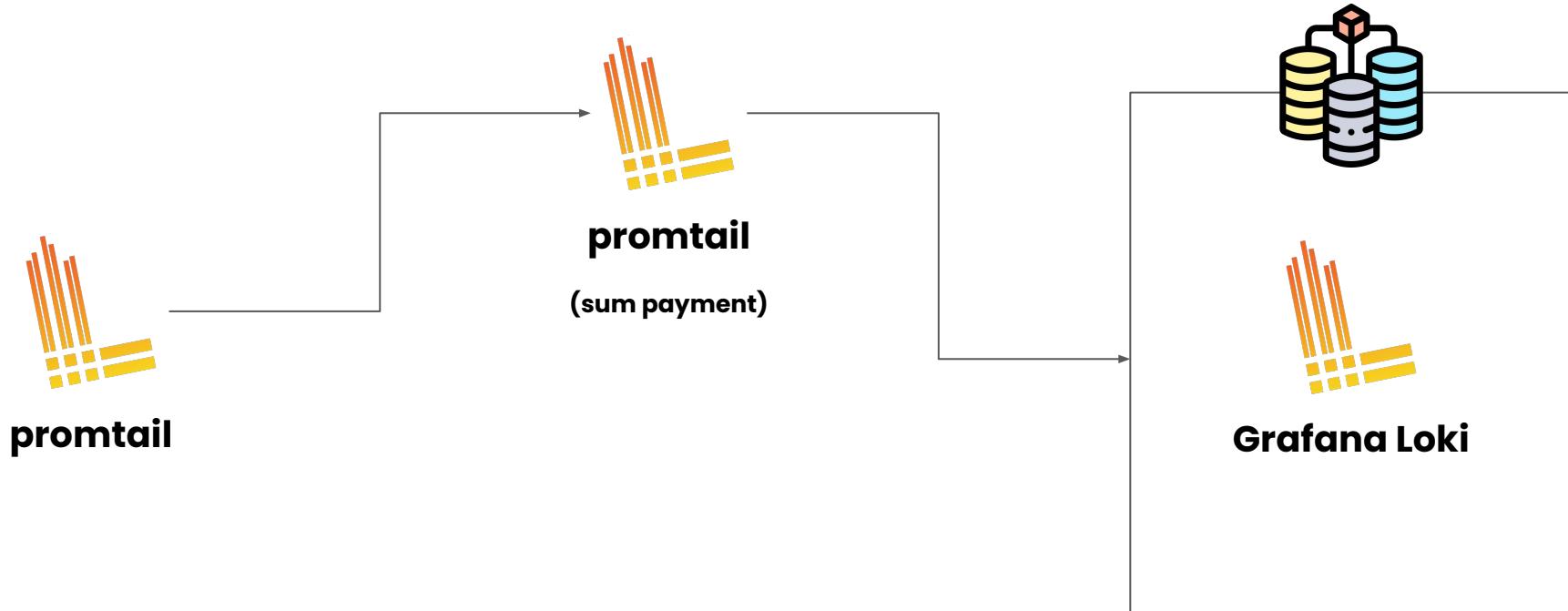


**Grafana Loki**

Centralized Logging

**Jumpbox®**

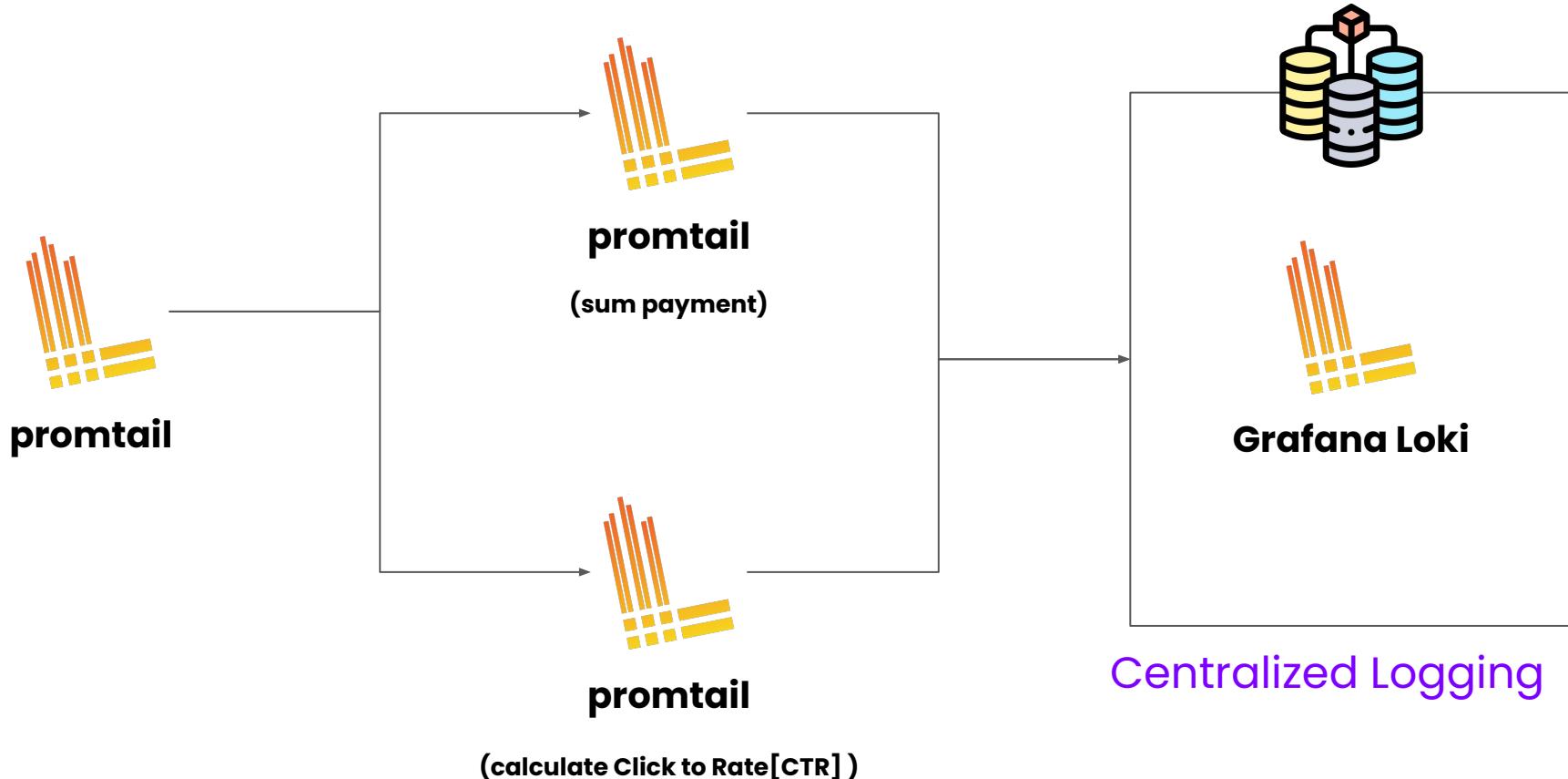
Let's do the pipeline



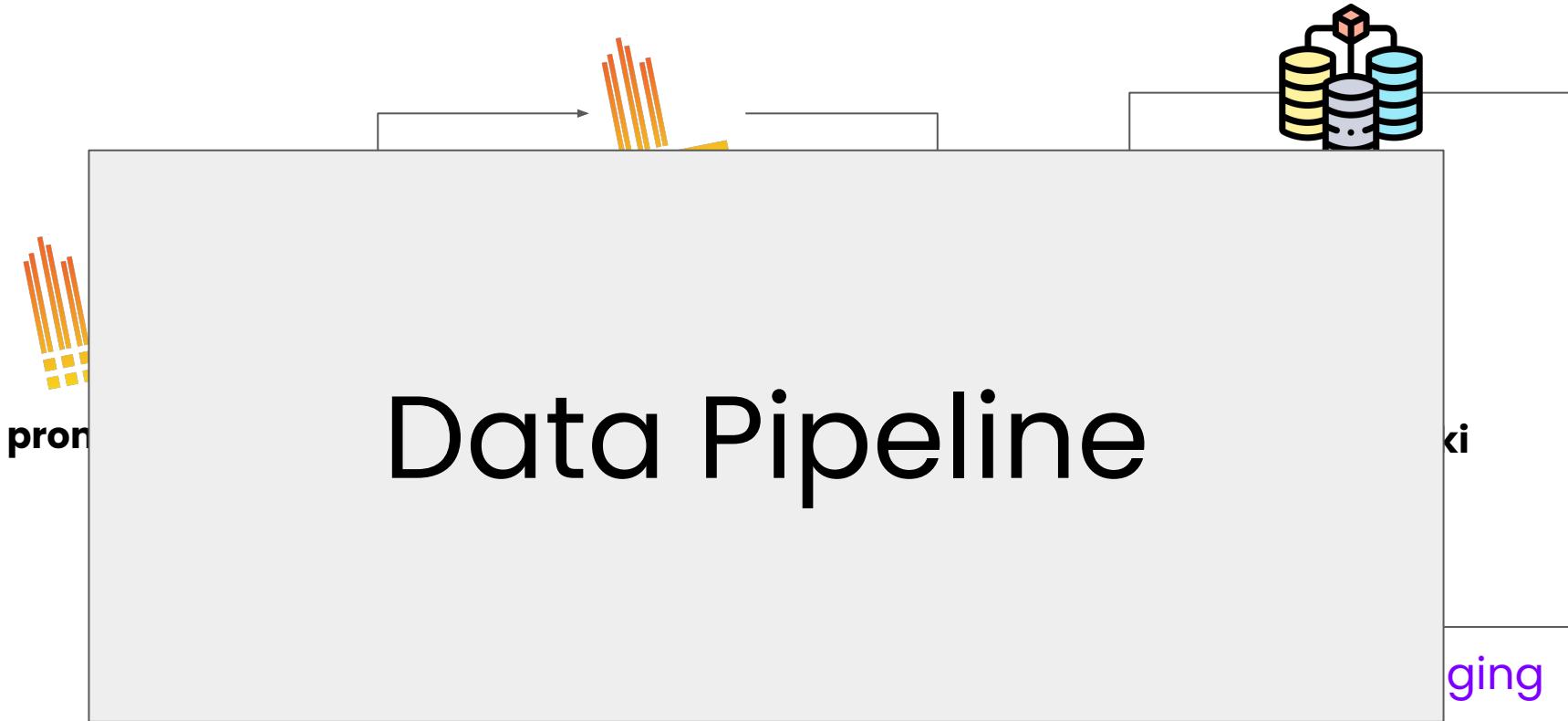
Centralized Logging

Jumpbox®

# Let's do the pipeline



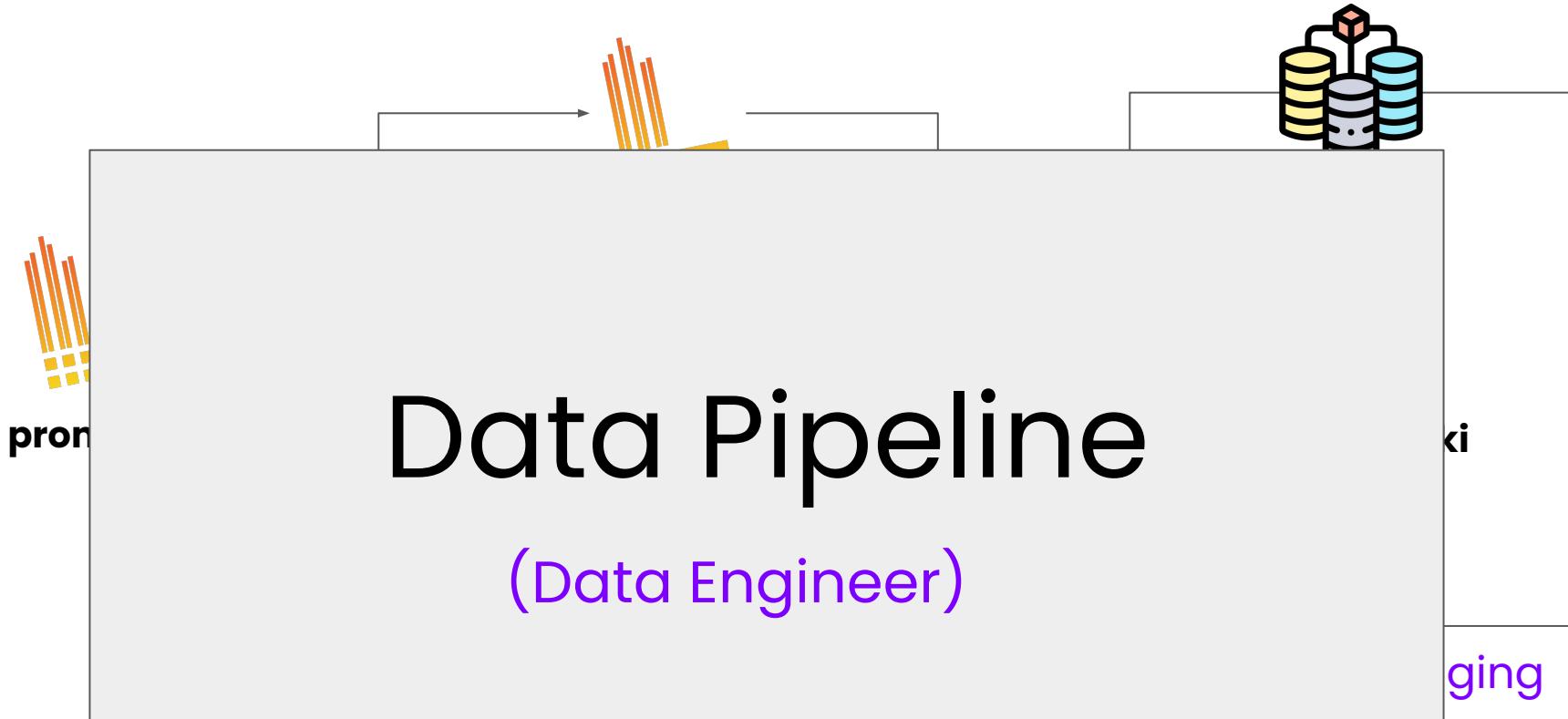
Let's do the pipeline



(calculate Click to Rate[CTR] )

Jumpbox®

Let's do the pipeline



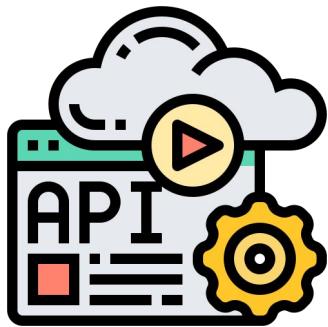
**(calculate Click to Rate[CTR] )**

Jumpbox®

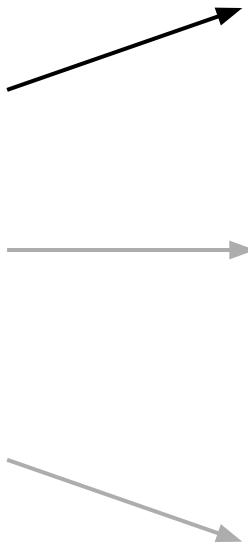


# Traces

# Application Output



App



Logs



Traces



Metrics



Grafana loki



elasticsearch



Grafana Tempo



elastic apm



Grafana

Mimir



Prometheus

Jumpbox®

# Application Tracing



Users

# Application Tracing

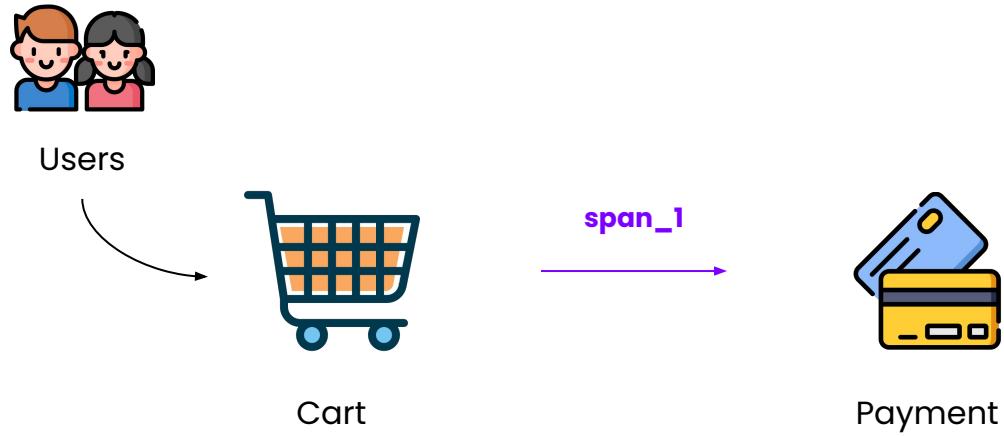


Users

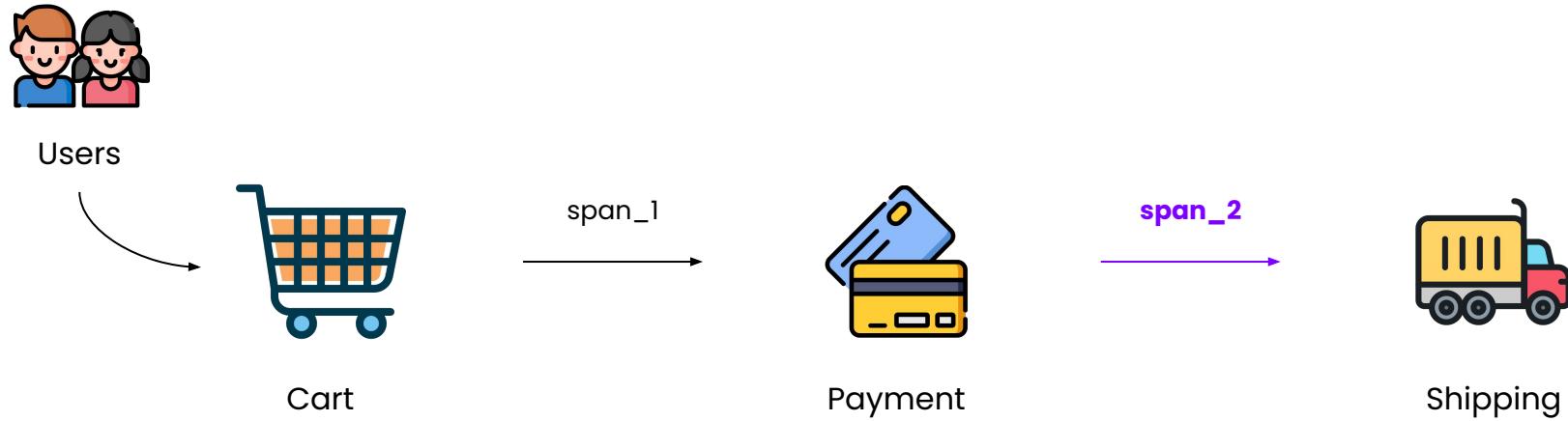


Cart

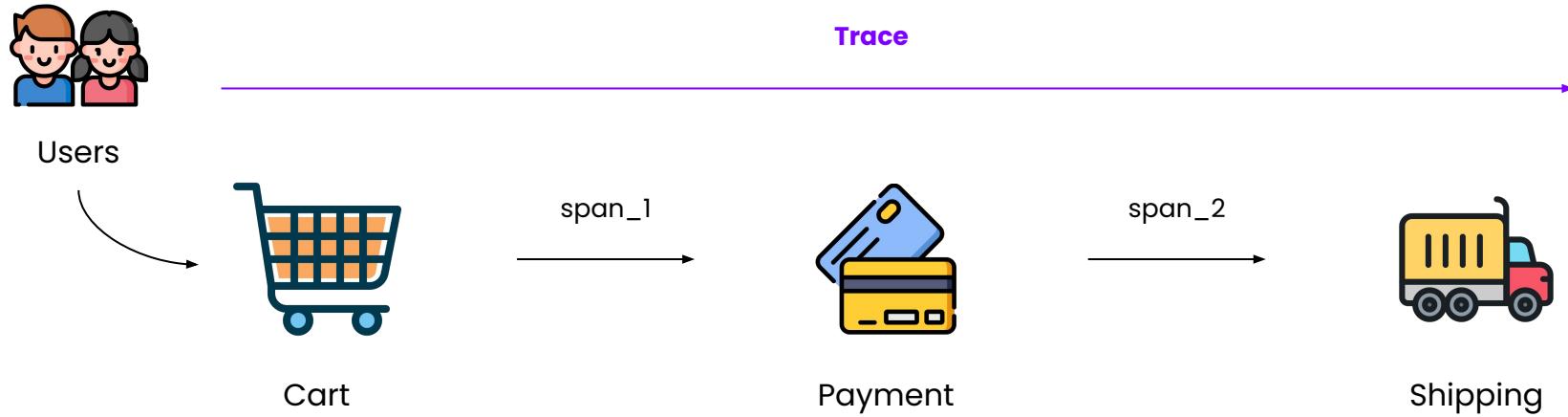
# Application Tracing



# Application Tracing



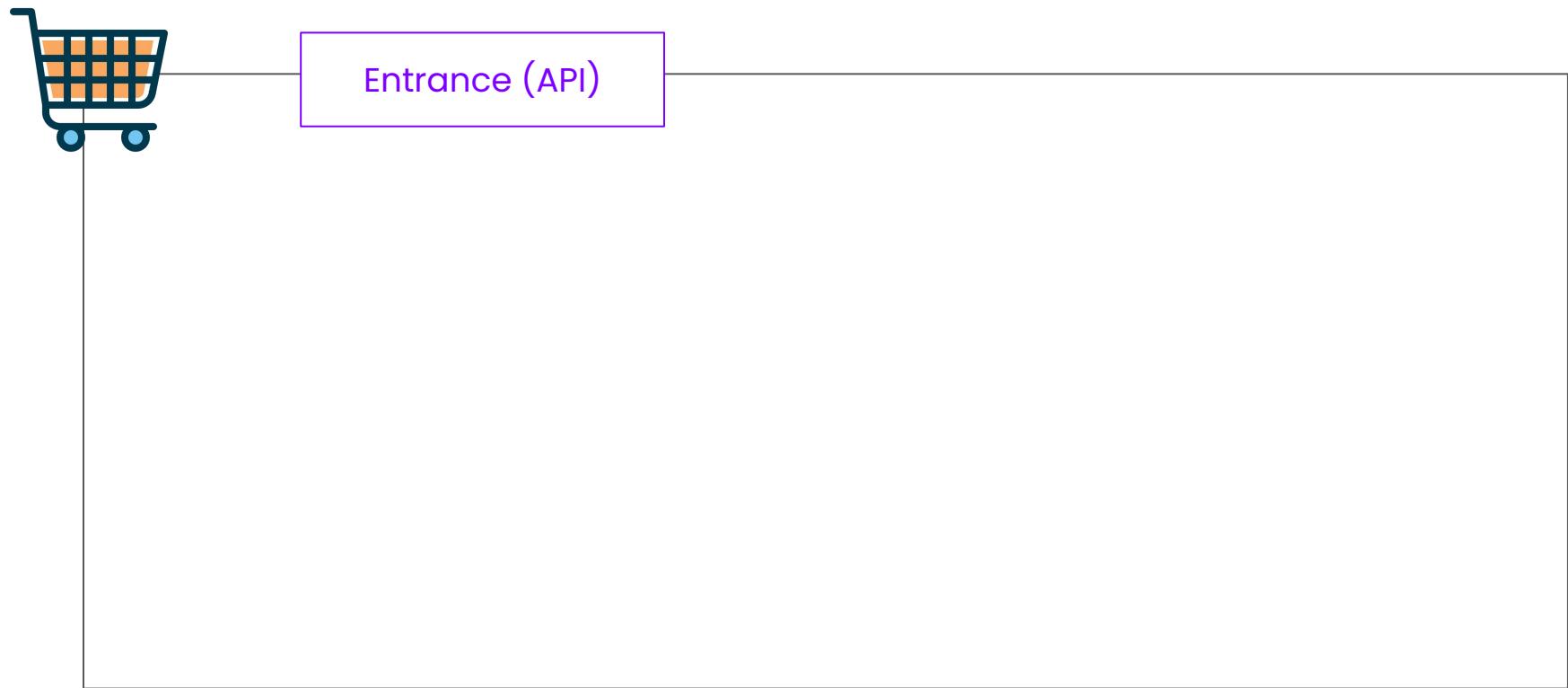
# Application Tracing



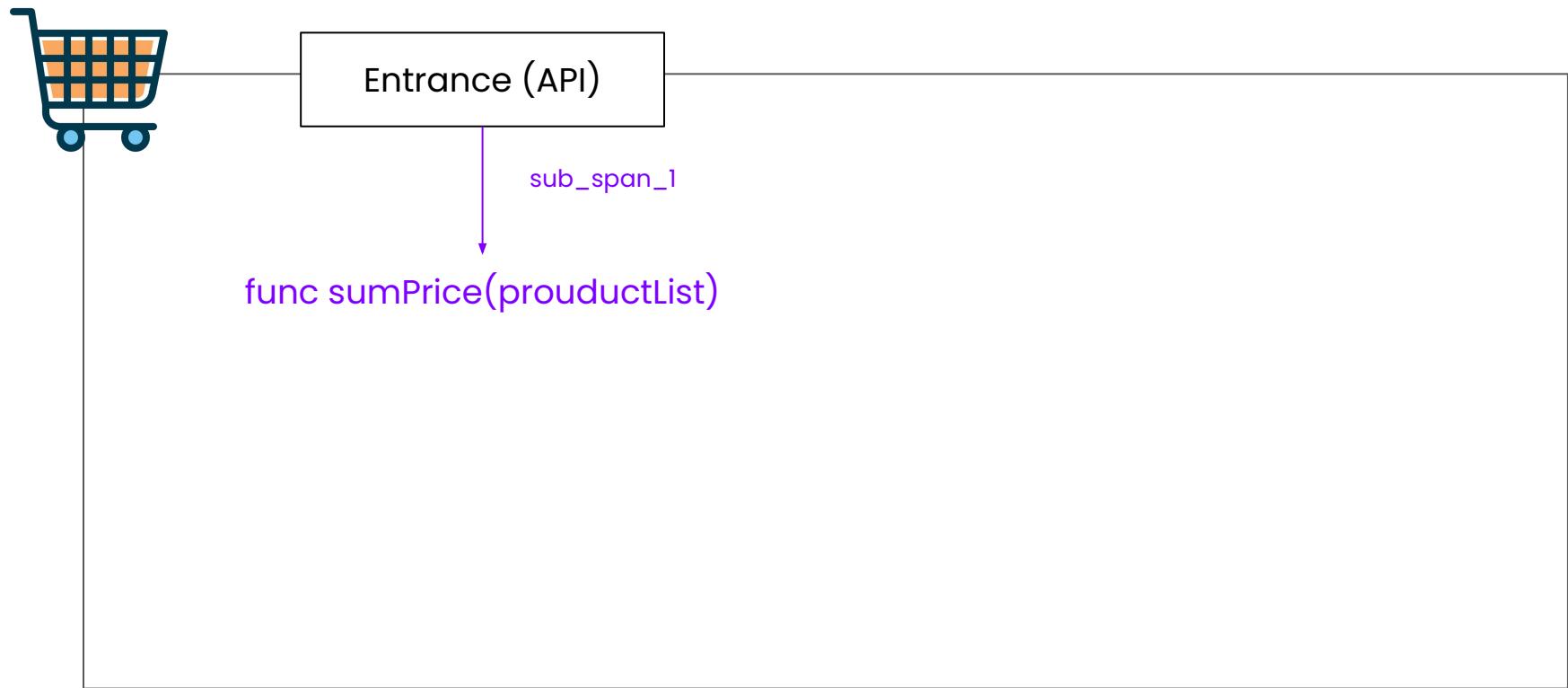
## Application Tracing - In Cart Process



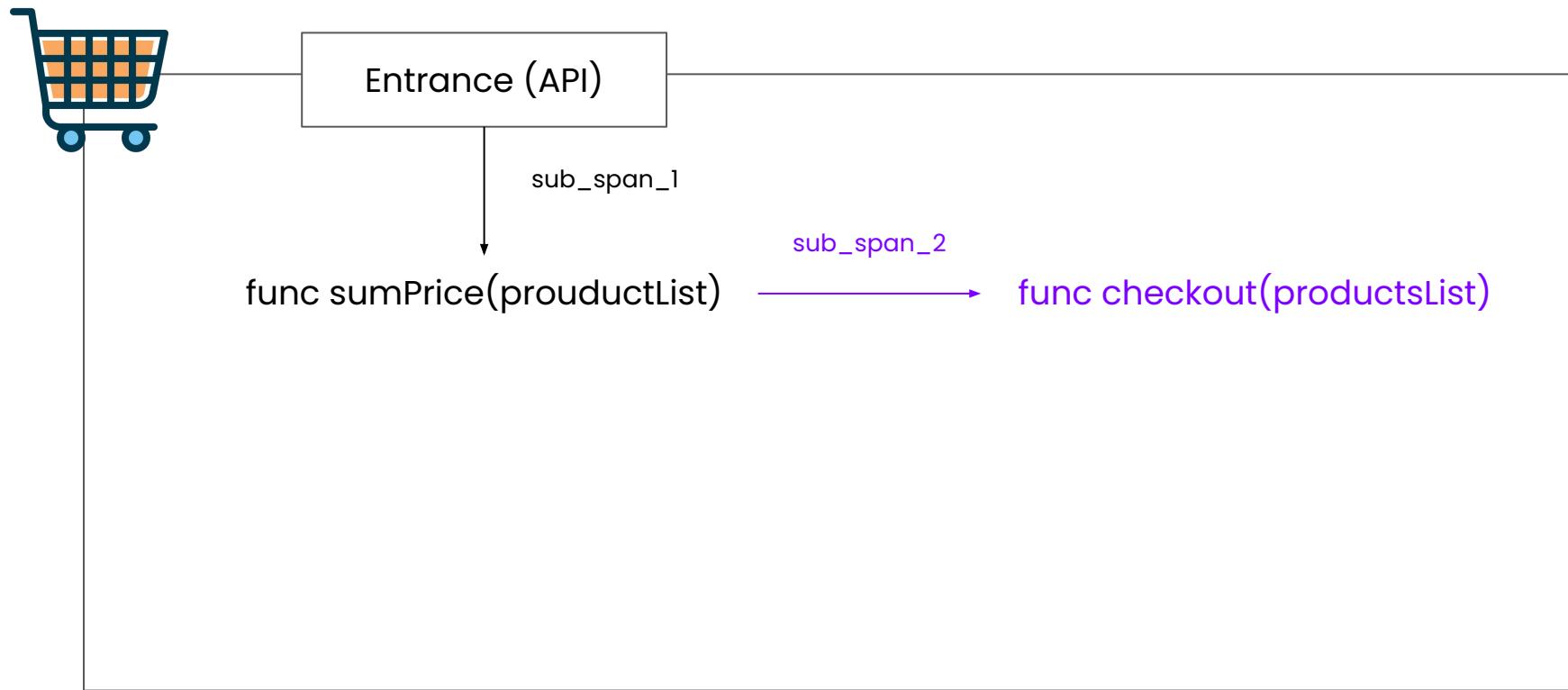
## Application Tracing - In Cart Process



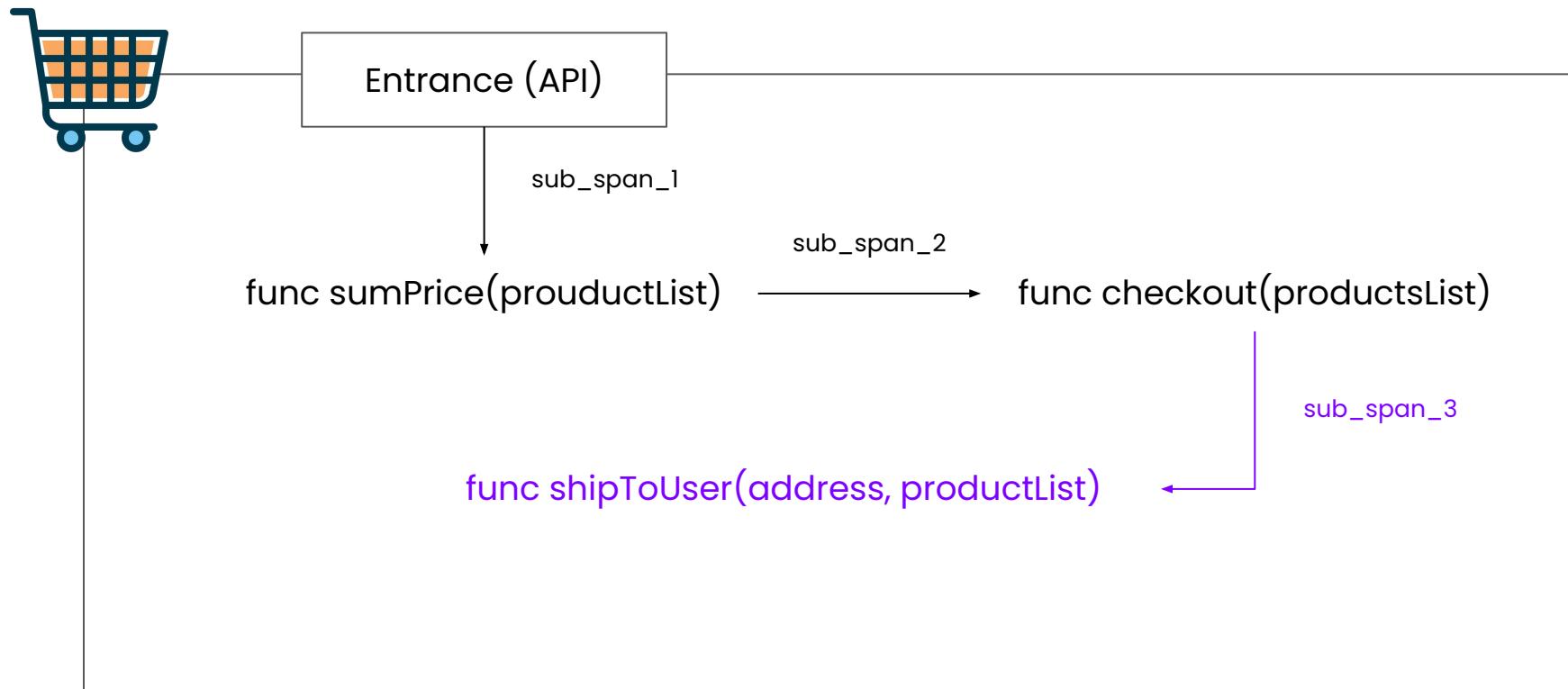
## Application Tracing - In Cart Process



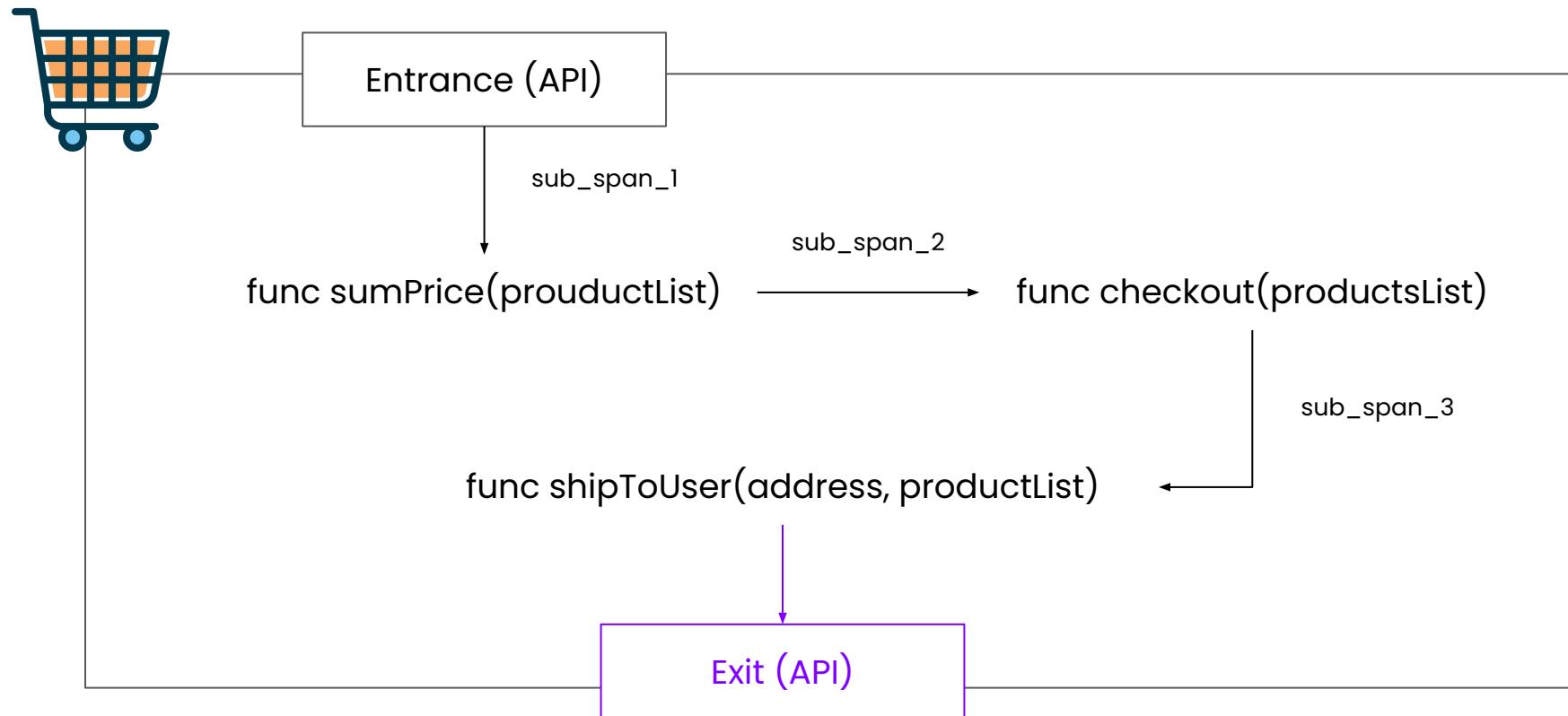
## Application Tracing - In Cart Process



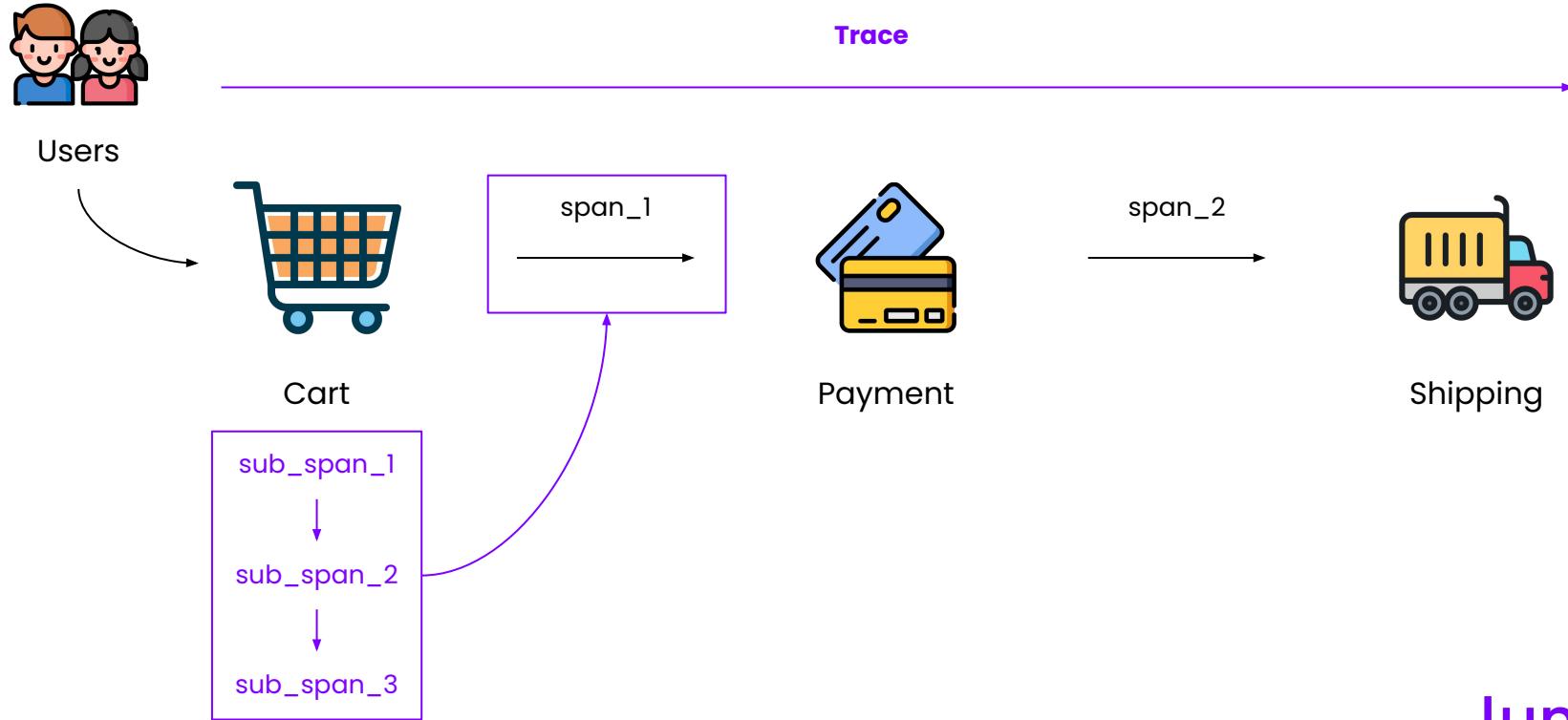
## Application Tracing - In Cart Process



## Application Tracing - In Cart Process



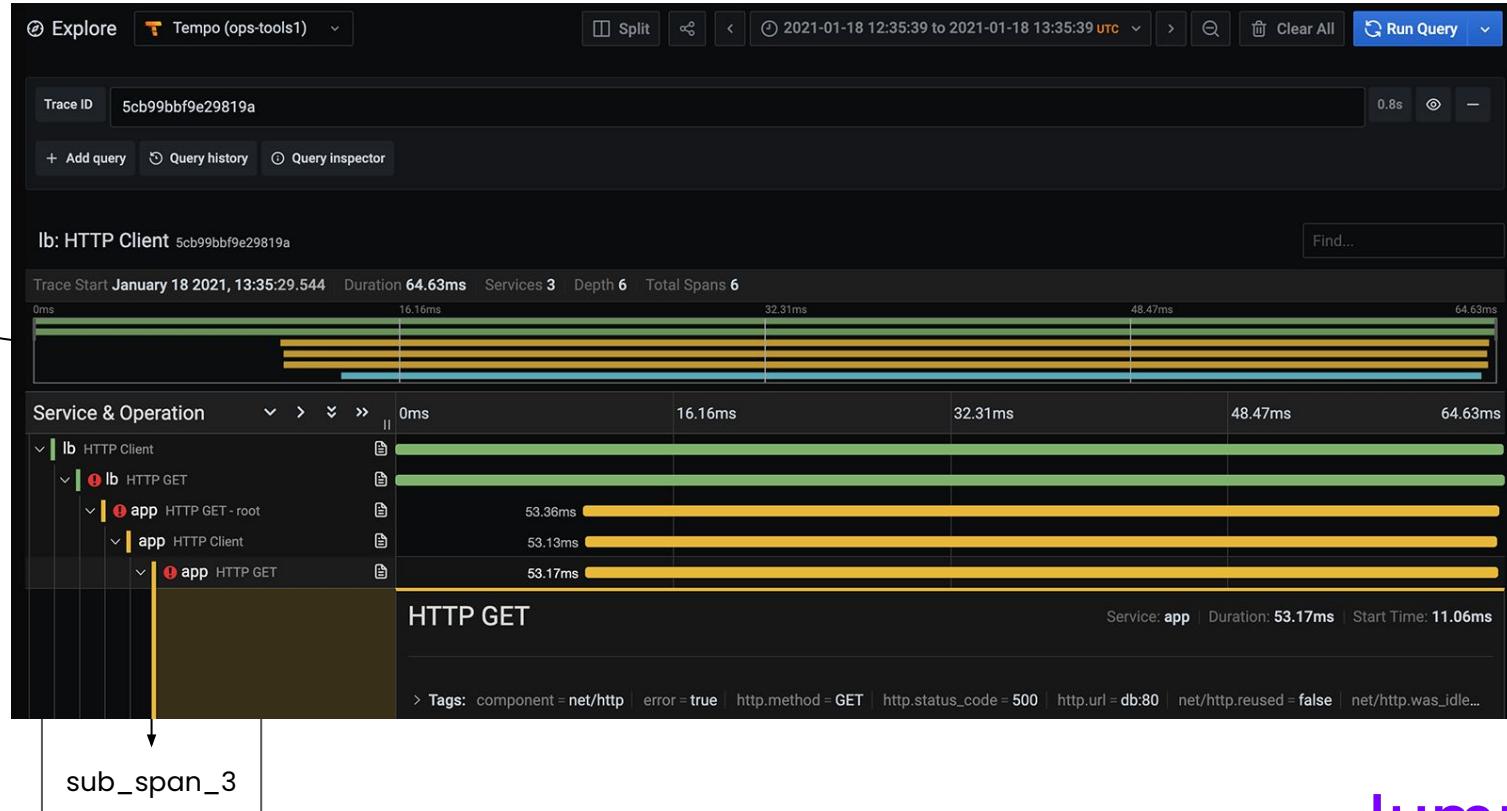
# Application Tracing - with sub process



# Application Tracing



Users

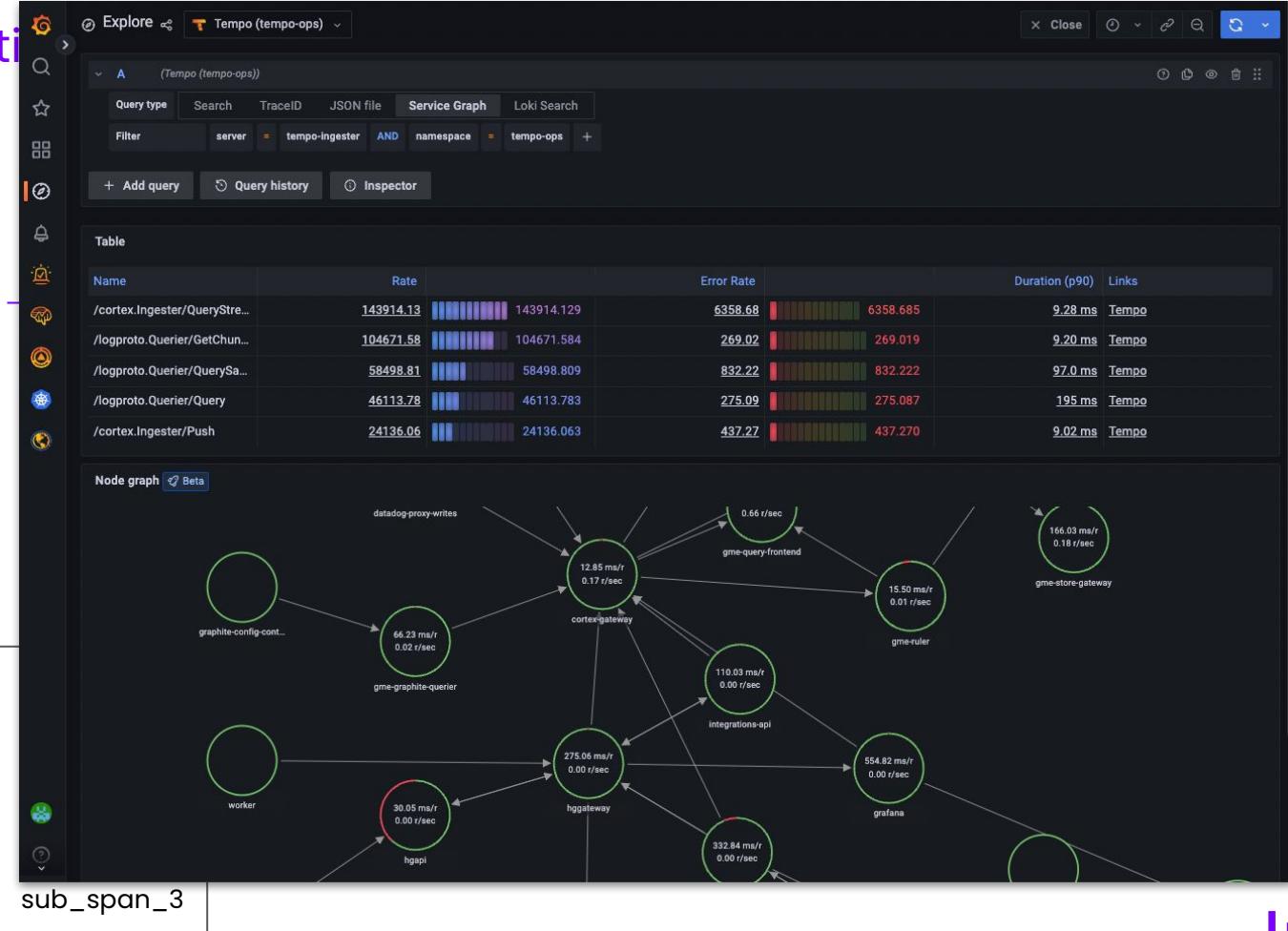


Jumpbox®

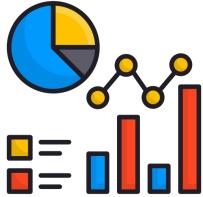
# Application



Users

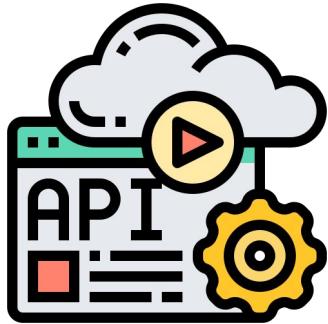


Jumpbox®

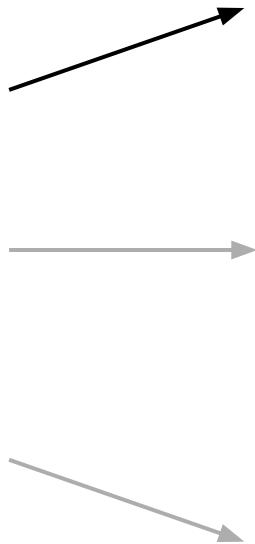


# Metrics

# Application Output



App



Logs



Traces



Metrics



Grafana loki



elasticsearch



Grafana Tempo



elastic apm



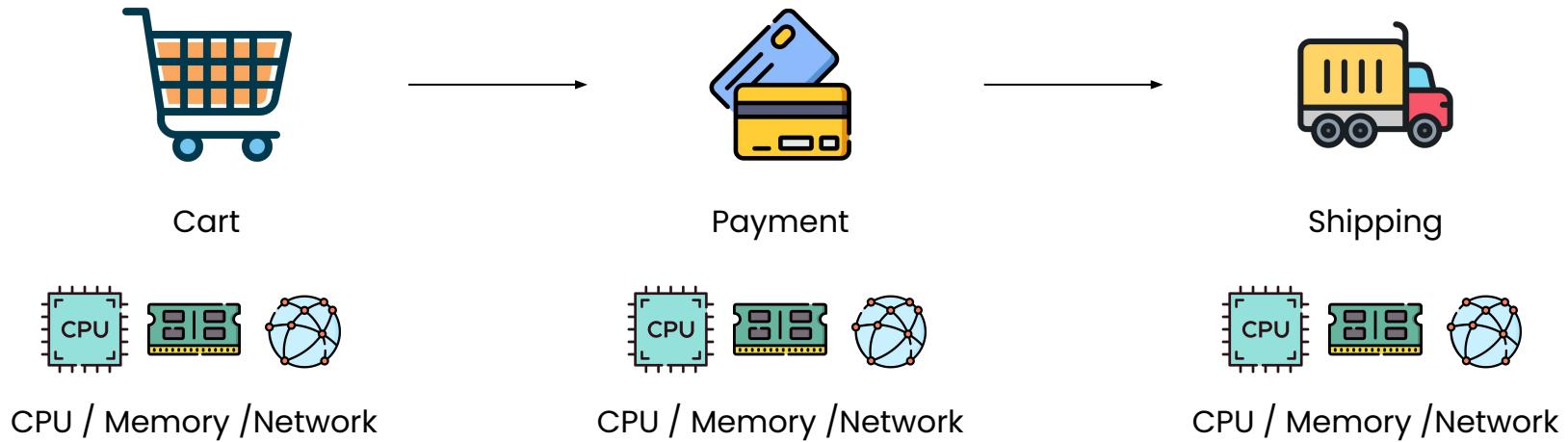
Prometheus

Jumpbox®

# Application Metric



## Application Metric (Cont.)



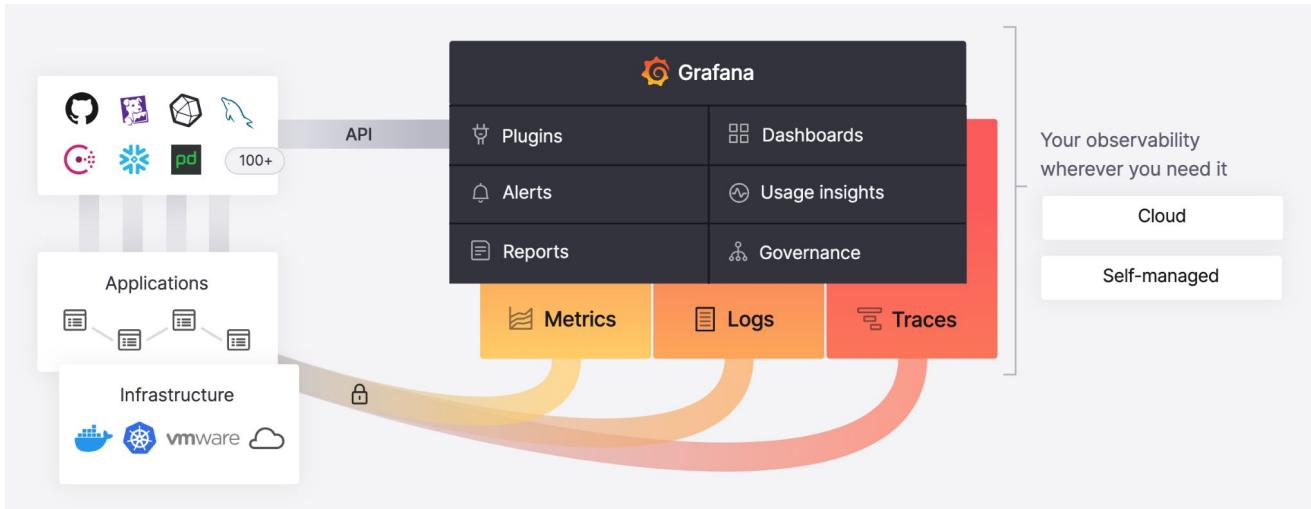
# Application Metric (Cont.)



# Observability

Platform

Jumpbox®



**Jumpbox®**

"**เราเชื่อว่าการเรียนรู้จะทำให้ชีวิตคุณจะดีขึ้นกว่าเดิม**"

-Jumpbox Team-

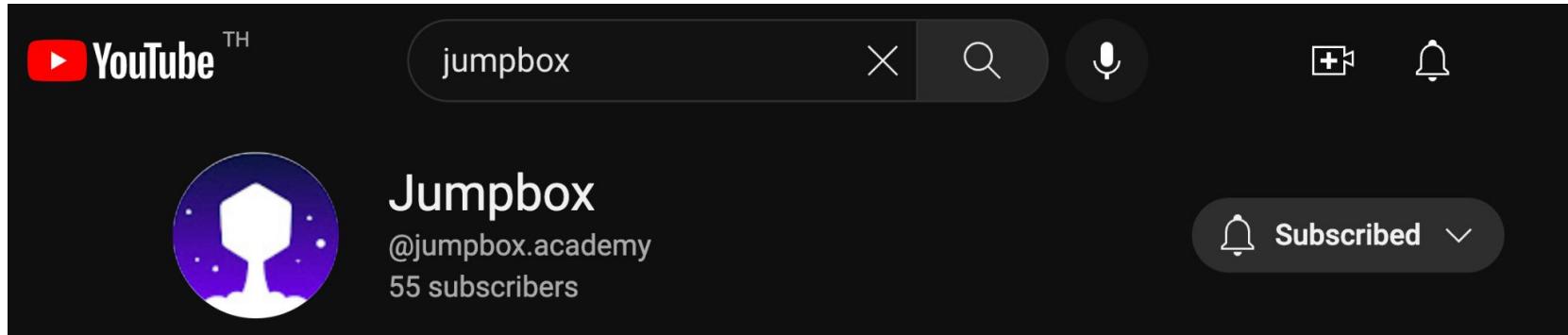
**Jumpbox®**

# facebook



<https://www.facebook.com/jumpbox.academy>

Jumpbox®



The image shows a screenshot of a YouTube channel page. At the top, there's a search bar with the text "jumpbox". To the right of the search bar are icons for a microphone, a camera, and a bell. Below the search bar, the channel's profile picture is displayed, which is a purple circle containing a white silhouette of a person's head and shoulders. Next to the profile picture, the channel name "Jumpbox" is written in white, along with the handle "@jumpbox.academy" and the subscriber count "55 subscribers". To the right of the channel information, there's a "Subscribed" button with a bell icon and a downward arrow. The background of the page is dark.

<https://www.youtube.com/@jumpbox.academy>

Jumpbox®

# Contact Us



**Jumpbox**



**@jumpbox**



**admin@jumpbox.co**



**063-245-2168 (JoJo)**

**062-796-1559 (Beau)**



**Jumpbox®**



**Jumpbox®**