

# 雲與地的彼端 - 全面啟動

## PAIA 3D之K8s混合雲部署

李緒成 & 張頌宇 & Ivan Chiou

# Agenda

PAIA專案架構介紹

雲端系統說明

地端系統說明

PAIA 3D遊戲介紹

Demo

結論

# Members



李緒成

成功大學資工系二年級

<https://www.linkedin.com/in/peterxcli>

<https://github.com/peterxcli>

<https://www.facebook.com/peterxcli>

<mailto:peterxcli@gmail.com>



張頌宇

成大 AI Robotics 碩士生  
PAIA工程師



Ivan Chiou

wyattkidd@gmail.com

Mentor and inspire the team, providing young engineers with guidance and hope for their future careers.



# PAIA介紹

## 產品與課程規劃圖





1. start.xml x

Python 尚未登入 語言 選項

編輯  
迴圈  
數學式  
文字  
清單  
字典  
多維陣列  
模型  
變數  
函式  
繪圖  
檔案  
MLGame

MLPlay

初始化：

- 變數 已經發球 設為 假
- 變數 前一個球的x座標 設為 0

更新：

- 如果 取得 遊戲狀態 = 遊戲 通關 或 取得 遊戲狀態 = 遊戲 失敗
  - 執行 回傳行動 重置
- 如果 已經發球 不成立
  - 執行 如果 英文字母鍵 A 被按下
    - 變數 已經發球 設為 真
    - 回傳行動 向左發球
  - 否則，如果 英文字母鍵 D 被按下
    - 變數 已經發球 設為 真
    - 回傳行動 向右發球
- 否則
  - 變數 球的x方向 設為 取得 球的 x 座標 - 前一個球的x座標
  - 變數 前一個球的x座標 設為 取得 球的 x 座標
  - 如果 球的x方向 > 0 或 取得 球的 x 座標 > 取得 平台的 x 座標
    - 執行 回傳行動 向右移動
  - 否則，如果 球的x方向 < 0 或 取得 球的 x 座標 < 取得 平台的 x 座標
    - 執行 回傳行動 向左移動

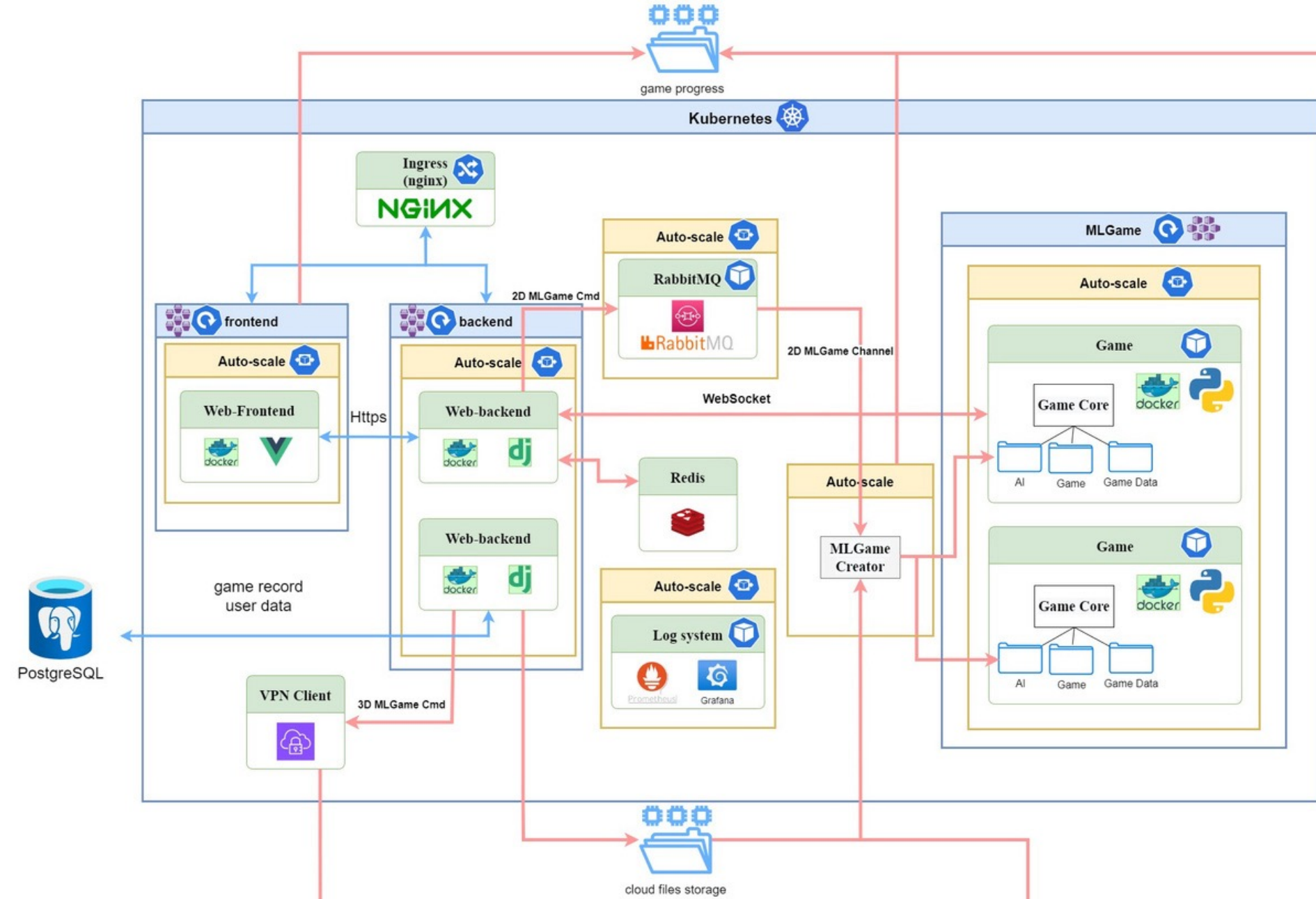
PAIA 遊戲引擎的遊戲畫面截圖，展示了多種遊戲場景和介面。

- 第一排左圖：球在迷宮中移動。
- 第一排中圖：俯視圖顯示賽道和障礙物。
- 第一排右圖：遊戲結束畫面，顯示 'Game Over'。
- 第二排左圖：俯視圖顯示賽道和檢查點。
- 第二排右圖：第一視角賽車遊戲，顯示計時器 0:53。
- 第三排左圖：俯視圖顯示賽車在賽道上。
- 第三排右圖：第一視角賽車遊戲，顯示速度表 0 km/h。

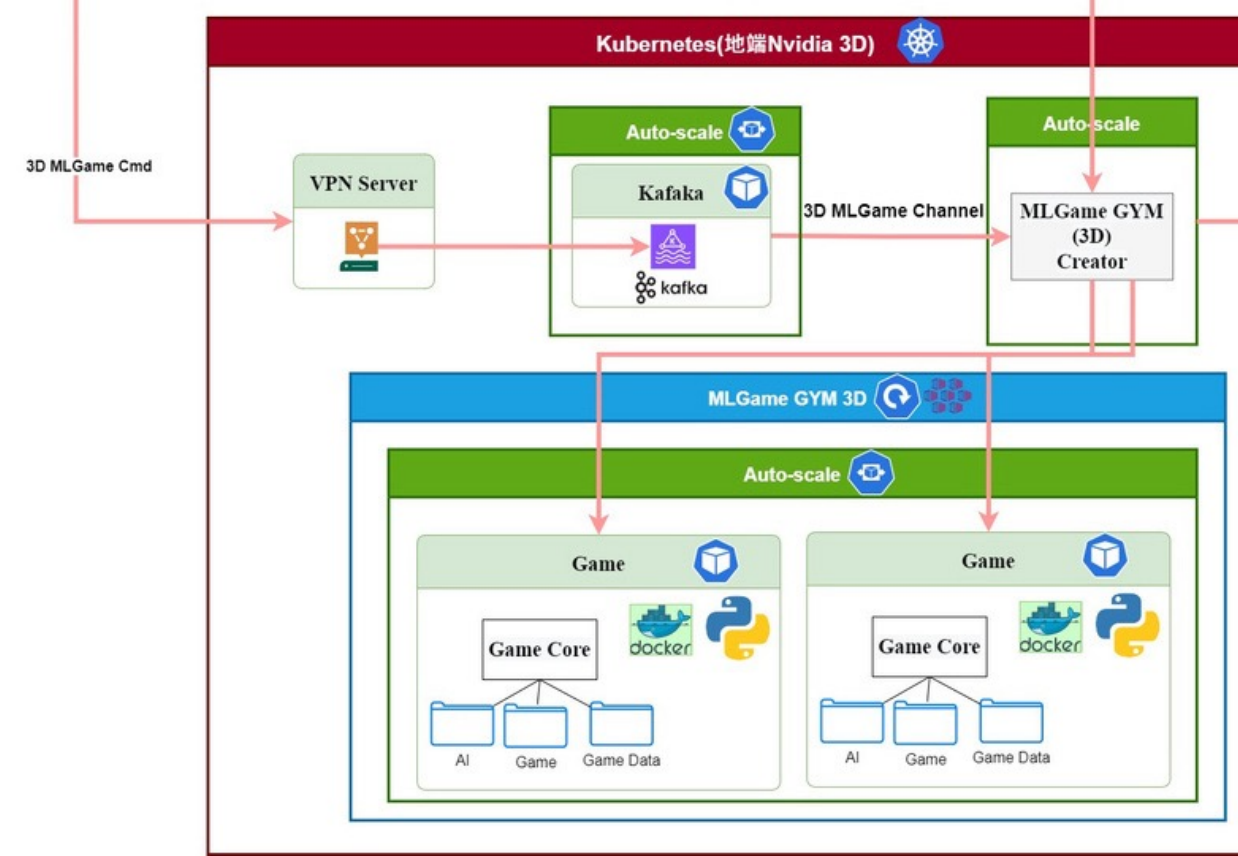
# Why 混和雲

- 原先PAIA機器學習AI 線上遊戲平臺部署在雲端，雲端運算費用昂貴。
- 決定建置地端的3D運算機器，承擔部份雲端費用。

# 現有雲端架構



# 佈建地端架構





# 前端觸發遊戲介面

PAIA 遊戲學習 AI 競

載桌面版 中文

跑跑卡丁車3D

遊戲介紹 我的遊戲AI 遊戲

編號 AI 名字

831	test
-----	------

執行 AI

是否使用輔助道具

開啟

輔助道具選擇

道具1

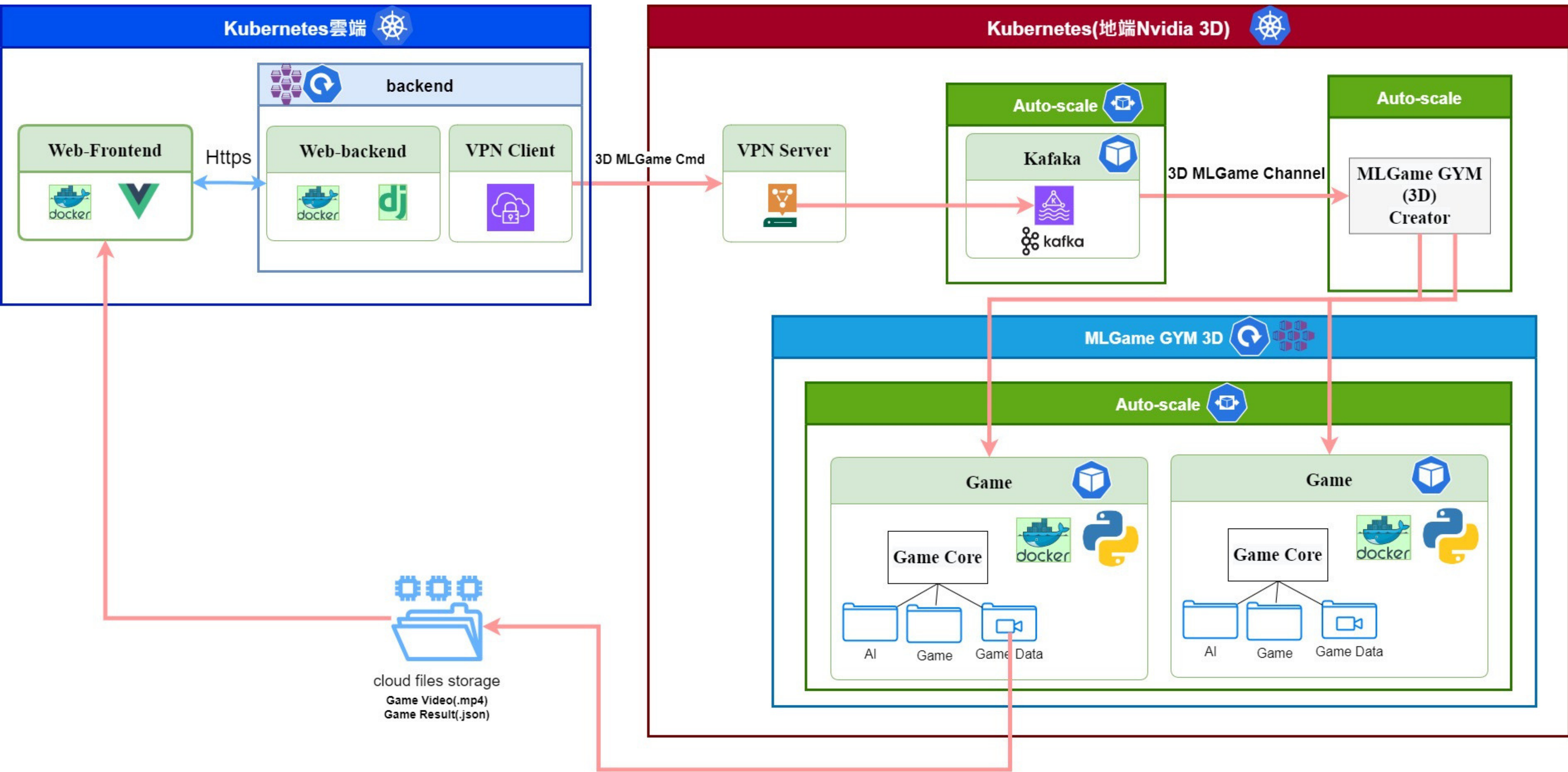
遊戲人數

1

上傳新增

Cancel OK

# PAIA 3D遊戲執行流程



TAB

Pause/Options

1:00

# GO

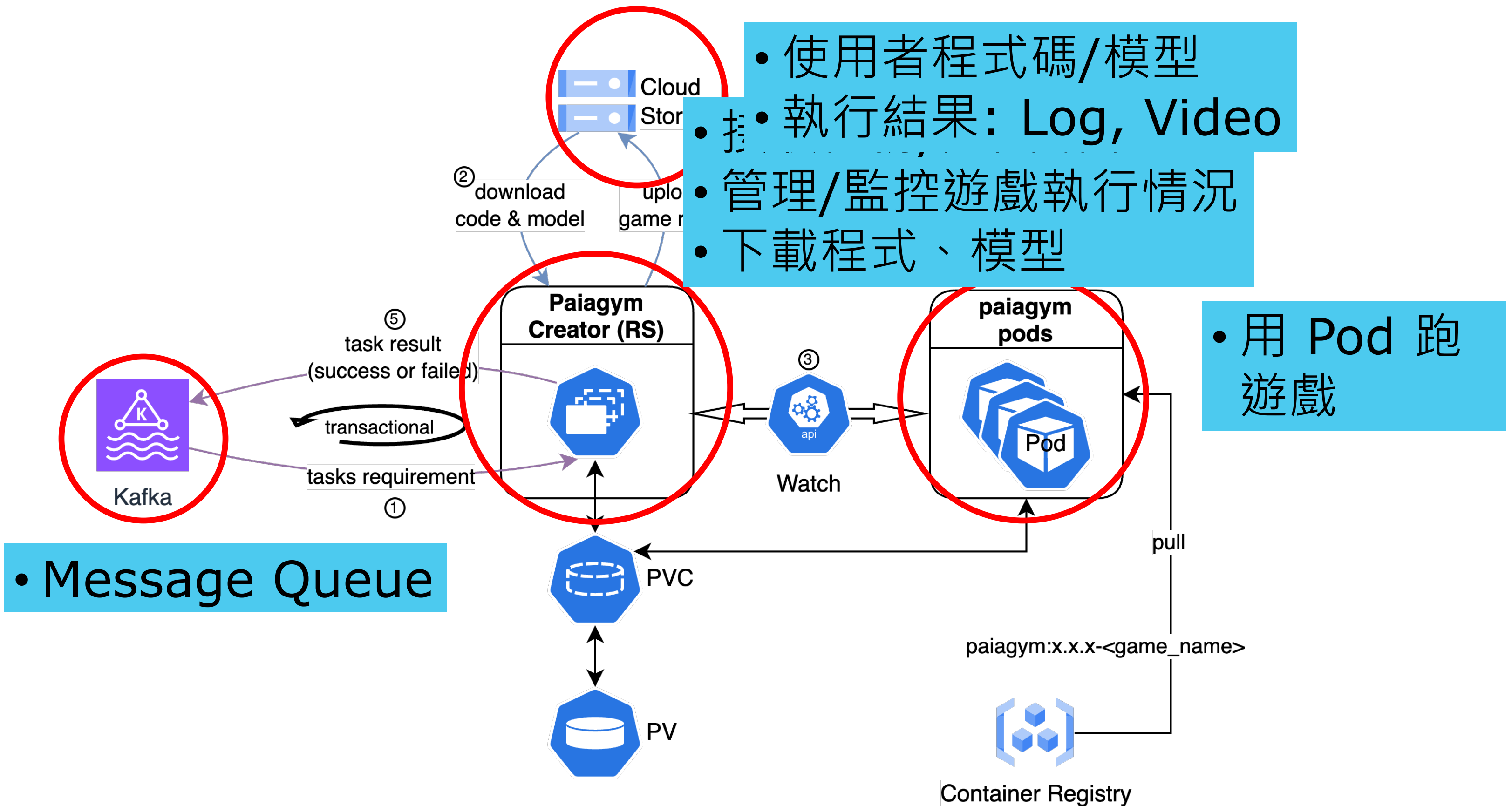
## KART

Pass all checkpoints before time runs out





# Local PAIA 3D 流程圖(組件)



- 使用者程式碼/模型
- 執行結果: Log, Video
- 管理/監控遊戲執行情況
- 下載程式、模型

- 用 Pod 跑遊戲

- Message Queue

# Read-Process-Write

```
Cloud
2023-12-08 03:36:45.797 | INFO | __main__:<module>:179 - Received message
{"text": "2023-12-08 03:36:45.797 | INFO | __main__:<module>:179 - Received message\n", "record": {"elapsed": {"repr": "23:31:41.221641", "seconds": 84701.221641},
855\", \"code_name\": \"kart3d_ppo_test\"}], \"ws_uri\": \"ws://webservice-websocket-service.default.svc.cluster.local/ws/game_server/room/8f8ef481-911b-432a-aacd-670b
<module>\", \"level\": {\"icon\": \"\", \"name\": \"INFO\", \"no\": 20}, \"line\": 179, \"message\": \"Received message\", \"module\": \"transactional_task\", \"name\": \"__main__\", \"process\"
2023-12-08 03:36:45.799 | INFO | __main__:process_and_produce:97 - Processing records
{"text": "2023-12-
08 03:36:45.799 | INFO | __main__:process_and_produce:97 - Processing records\n", "record": {"elapsed": {"repr": "23:31:41.223347", "seconds": 84701.223347}, "exce
08 03:36:45.799226+00:00", "timestamp": 1702006605.799226}}
{'url': 'https://paiastage.blob.core.windows.net/user-102/kart_3d/code-855', 'code_name': 'kart3d_ppo_test'}
Download from azure storage
2023-12-08 03:37:07.828 | INFO | pod_creator:run_3d_game_pod_process:45 - Download program successfully
{"text": "2023-12-08 03:37:07.828 | INFO | pod_creator:run_3d_game_pod_process:45 - Download program successfully\n", "record": {"elapsed": {"repr": "23:32:03.2524
3_0"}, "time": {"repr": "2023-12-08 03:37:07.828351+00:00", "timestamp": 1702006627.828351}}
2023-12-08 03:37:07.864 | INFO | pod_creator:run_3d_game_pod_process:64 - Pod created
| pod_creator:run_3d_game_pod_process:64 - Pod created\n", "record": {"elapsed": {"repr": "23:32:03.288854", "seconds": 84
864733+00:00", "timestamp": 1702006627.864733}}
tor:run_3d_game_pod_process:72 - Pod status changed
| pod_creator:run_3d_game_pod_process:72 - Pod status changed\n", "record": {"elapsed": {"repr": "23:32:03.303222", "secon
: "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name":
tor:run_3d_game_pod_process:72 - Pod status changed
| pod_creator:run_3d_game_pod_process:72 - Pod status changed\n", "record": {"elapsed": {"repr": "23:32:03.305382", "secon
: "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name":
tor:run_3d_game_pod_process:72 - Pod status changed
| pod_creator:run_3d_game_pod_process:72 - Pod status changed\n", "record": {"elapsed": {"repr": "23:32:03.327594", "secon
: "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name":
tor:run_3d_game_pod_process:72 - Pod status changed
| pod_creator:run_3d_game_pod_process:72 - Pod status changed\n", "record": {"elapsed": {"repr": "23:32:06.570918", "secon
ile": {"name": "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name":
| pod_creator:run_3d_game_pod_process:72 - Pod status changed
7 | INFO | pod_creator:run_3d_game_pod_process:72 - Pod status changed\n", "record": {"elapsed": {"repr": "23:33:41.751391", "secon
ile": {"name": "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name":
| pod_creator:run_3d_game_pod_process:72 - Pod status changed
3 | INFO | pod_creator:run_3d_game_pod_process:72 - Pod status changed\n", "record": {"elapsed": {"repr": "23:33:42.952358", "secon
'file': {"name": "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name
| pod_creator:run_3d_game_pod_process:76 - Pod finished
} | INFO | pod_creator:run_3d_game_pod_process:76 - Pod finished\n", "record": {"elapsed": {"repr": "23:33:42.953830", "seconds": 8
'file': {"name": "pod_creator.py", "path": "/app/pod_creator.py"}, "function": "run_3d_game_pod_process", "level": {"icon": "", "name

total: 4991311
re.windows.net/records/8f8ef481-911b-432a-aacd-670b63b8f288/video.mp4?se=2023-12-10T05%3A38%3A48Z&sp=rwl&sv=2020-10-02&sr=c&sig=8yMgeWq
al: 103
re.windows.net/records/8f8ef481-911b-432a-aacd-670b63b8f288/result.json?se=2023-12-10T05%3A38%3A48Z&sp=rwl&sv=2020-10-02&sr=c&sig=8yMge
| pod_creator:run_3d_game_pod_process:121 - Pod deleted
3 | INFO | pod_creator:run_3d_game_pod_process:121 - Pod deleted\n", "record": {"elapsed": {"repr": "23:33:44.902553", "seconds": 8
-08 03:38:49.478432+00:00", "timestamp": 1702006729.478432}}
{"text": "2023-12-08 03:38:49.479 | CRITICAL | __main__:process_and_produce:110 - {"status": "Pod pod-8 finished", "azure_storage_url": ["https://paiastage.blob.core.windows.net
02&sr=c&sig=8yMgeWqTokB5hbUtrt%2BH/y8lxs2vLDW42YUVPKu%2Bp6M%3D"]}\n", "record": {"elapsed": {"repr": "23:33:44.903253", "seconds": 84824.903253}, "exception": null, "
02&sr=c&sig=8yMgeWqTokB5hbUtrt%2BH/y8lxs2vLDW42YUVPKu%2Bp6M%3D\", \"https://paiastage.blob.core.windows.net/records/8f8ef481-911b-432a-aacd-670b63b8f288/result.json?se
[{"status": "Pod pod-8 finished", "azure_storage_url": ["https://paiastage.blob.core.windows.net/records/8f8ef481-911b-432a-aacd-670b63b8f288/video.mp4?se=2023-12-10T
2023-12-08 03:38:49.479 | INFO | __main__:process_and_produce:125 - Produced result
Container Registry
```

```
{
  "status": "Pod pod-8 finished",
  "azure_storage_url": [
    "https://paiastage.blob.core.windows.net/records/8f8ef481-911b-432a-aacd-670b63b8f288/video
    .mp4?se=2023-12-10T05%3A38%3A48Z&sp=rwl&sv=2020-10-02&sr=c&sig=8yMgeWqTokB5hbUtrt%2BH
  ]
},
{
  "type": "game_cmd",
  "data": {
    "category": "3D",
    "game_id": 7,
    "game_name": "kart_3d",
    "room_id": "8f8ef481-911b-432a-aacd-670b63b8f288",
    "game_param": {
      "is_pickup": "true",
      "pickup_map": "1",
      "user_num": 1
    },
    "plug_in_data": [],
    "programs": [
      {
        "url": "https://paiastage.blob.core.windows.net/user-102/kart_3d/code-855",
        "code_name": "kart3d_ppo_test"
      }
    ]
  },
  "ws_uri": "ws://webservice-websocket-service.default.svc.cluster.local/ws/game_server/room
/8f8ef481-911b-432a-aacd-670b63b8f288?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
eyJ0b2t1b190eXB1IjojYWNjZXNzIiwiaXNzIjoiZXhwIjoxNzAzMjY0MDY0OTUyLCJqdGkiOiI5SW9WRkMjA4N2U5ZDA0MDYzOTE4Mj
hjZjMSNDJiOTgzMSIsInVzZXIjOiJlWmMn0.1P6xrtWX44J8X1n5EE5Pe3DzJIq4R7g705wY-4zkD8"
}
}
```

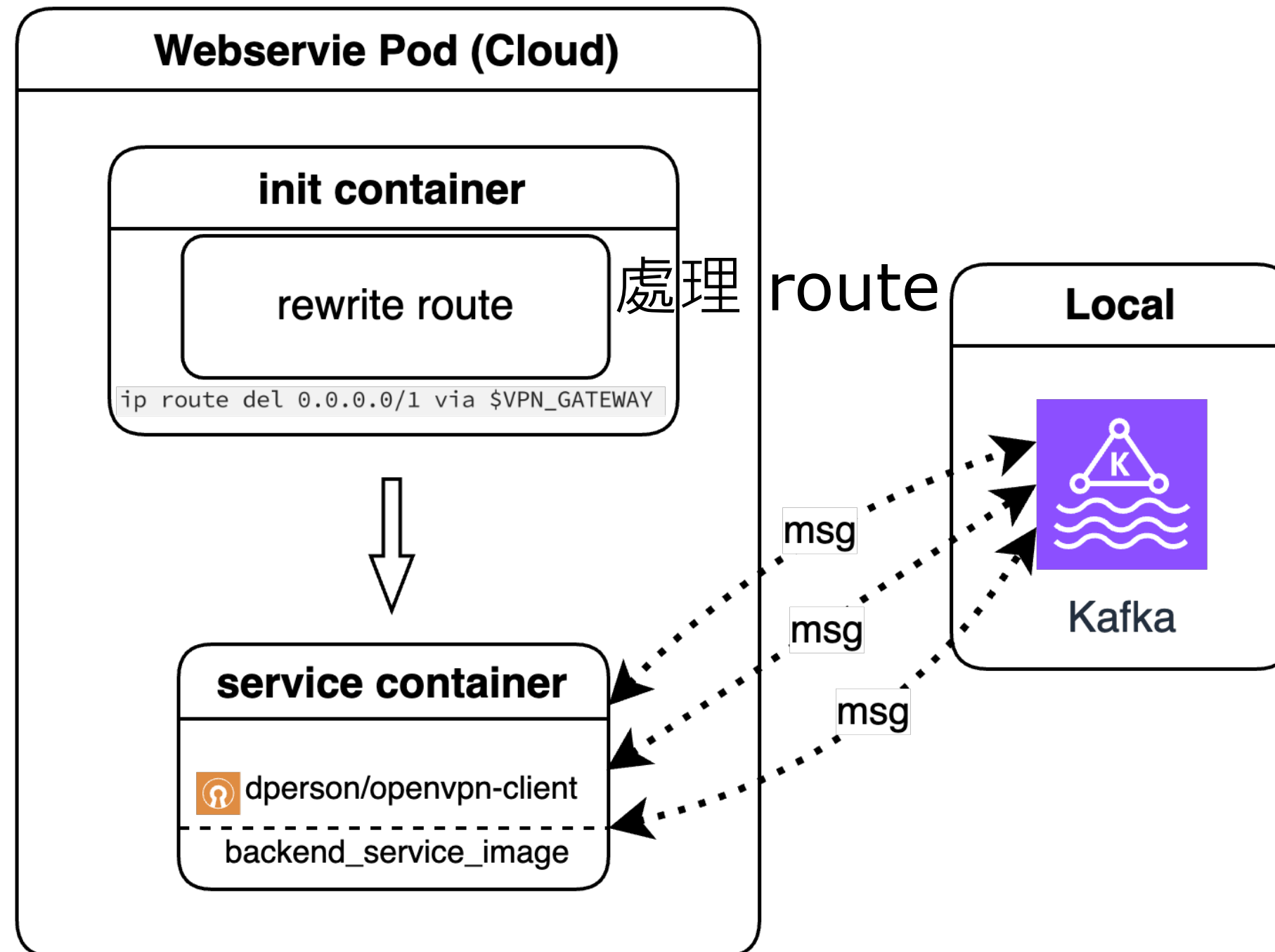
# Local PAIA 3D 詳細流程

1. 從 Kafka 收任務
2. 從後端服務(Webservice)處理好的 Cloud Storage 位置下載使用者上傳的程式
  - 遊戲 Pod 跟 Creator 掛同個 PVC
3. Creator 透過 Watch 去監控 paiagym 遊戲 Pod 的狀態
4. Creator 抓遊戲結果傳給 Cloud Storage
  - 遊戲 Pod (paiagym) 跟 Creator 掛同個 PVC
5. 遊戲執行結果用 Producer 回報給後端 Webservice

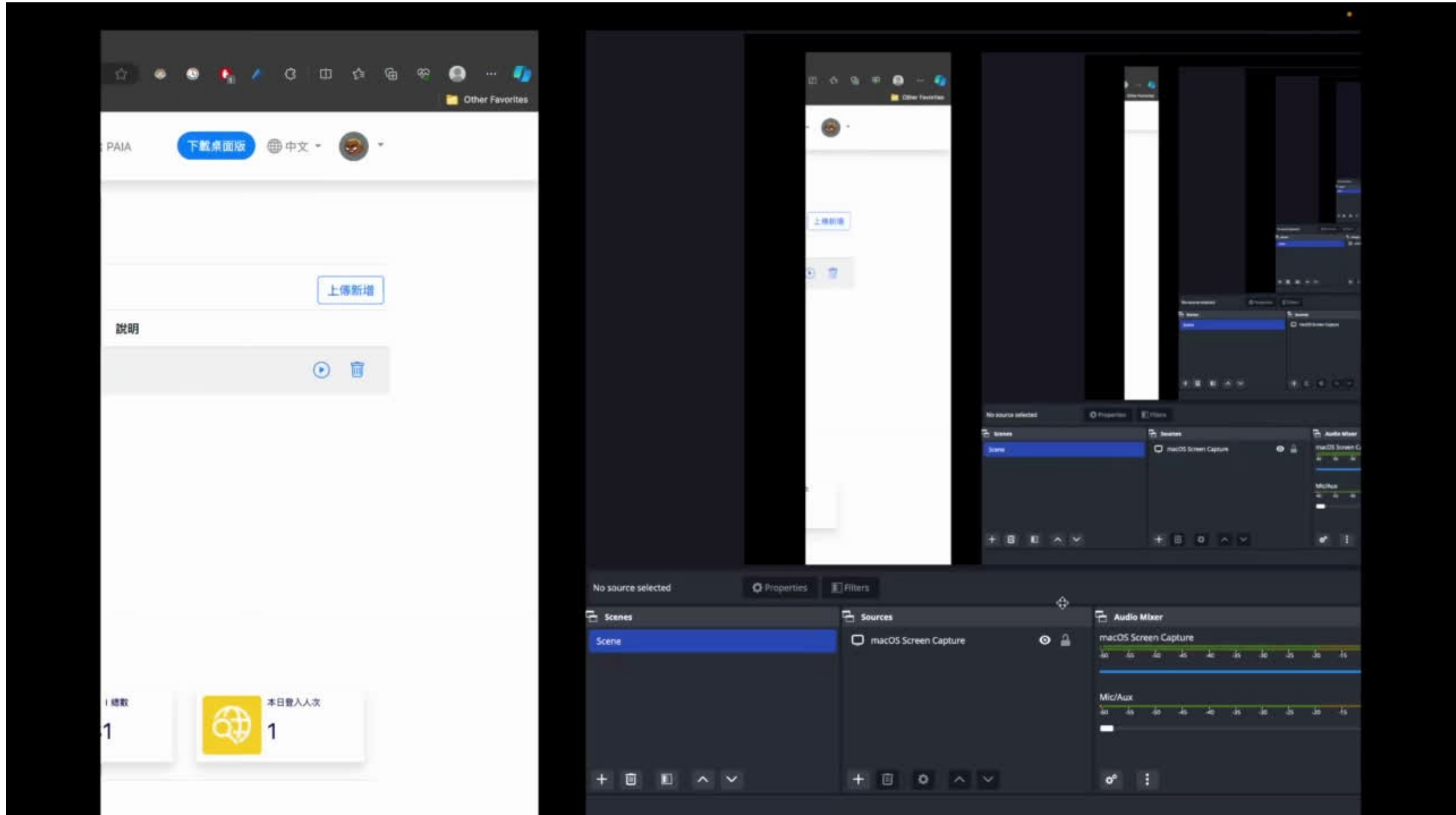


# OpenVPN Client

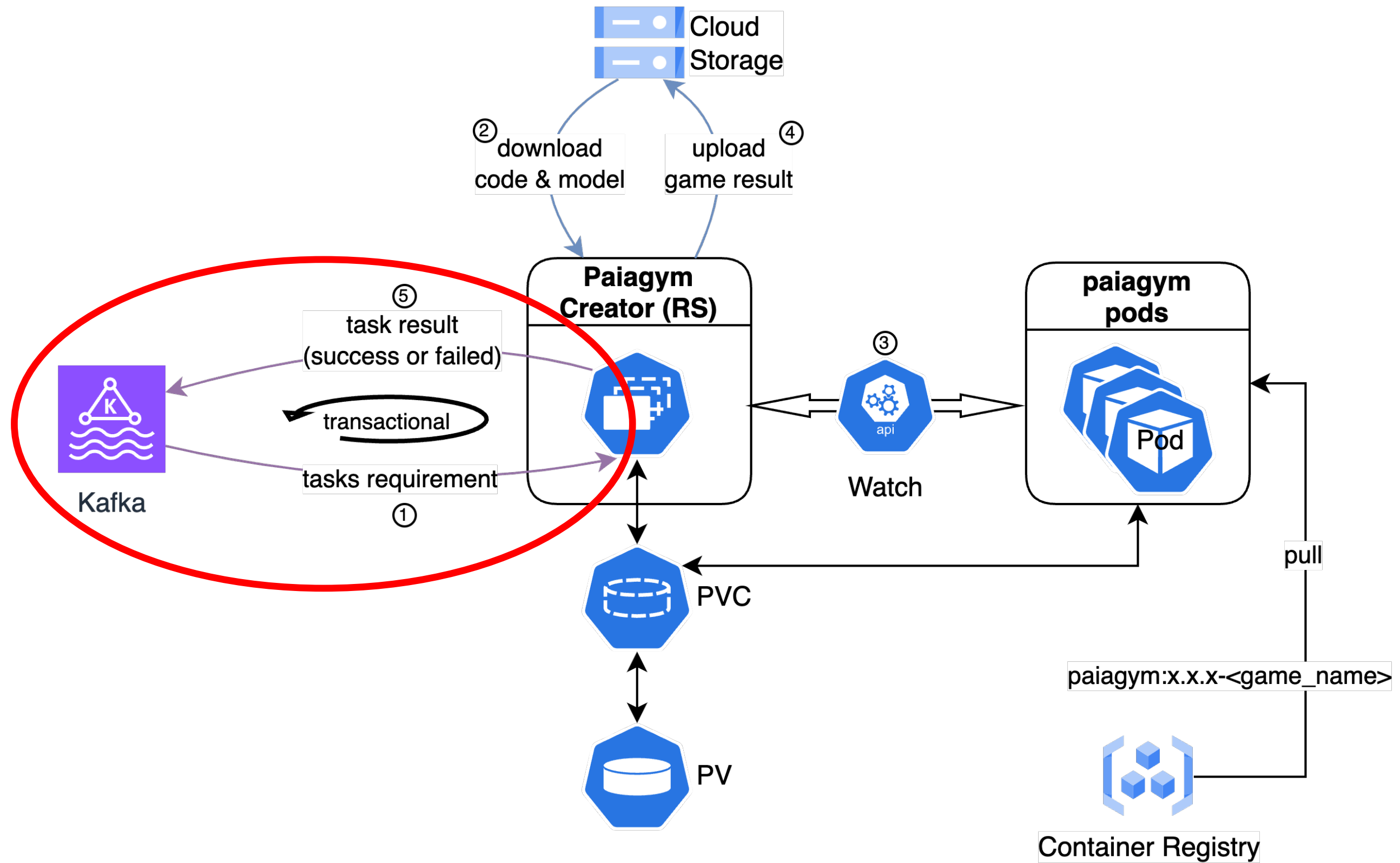
Cloud -> Local Cloud



# Kafka 收到訊息後的處理

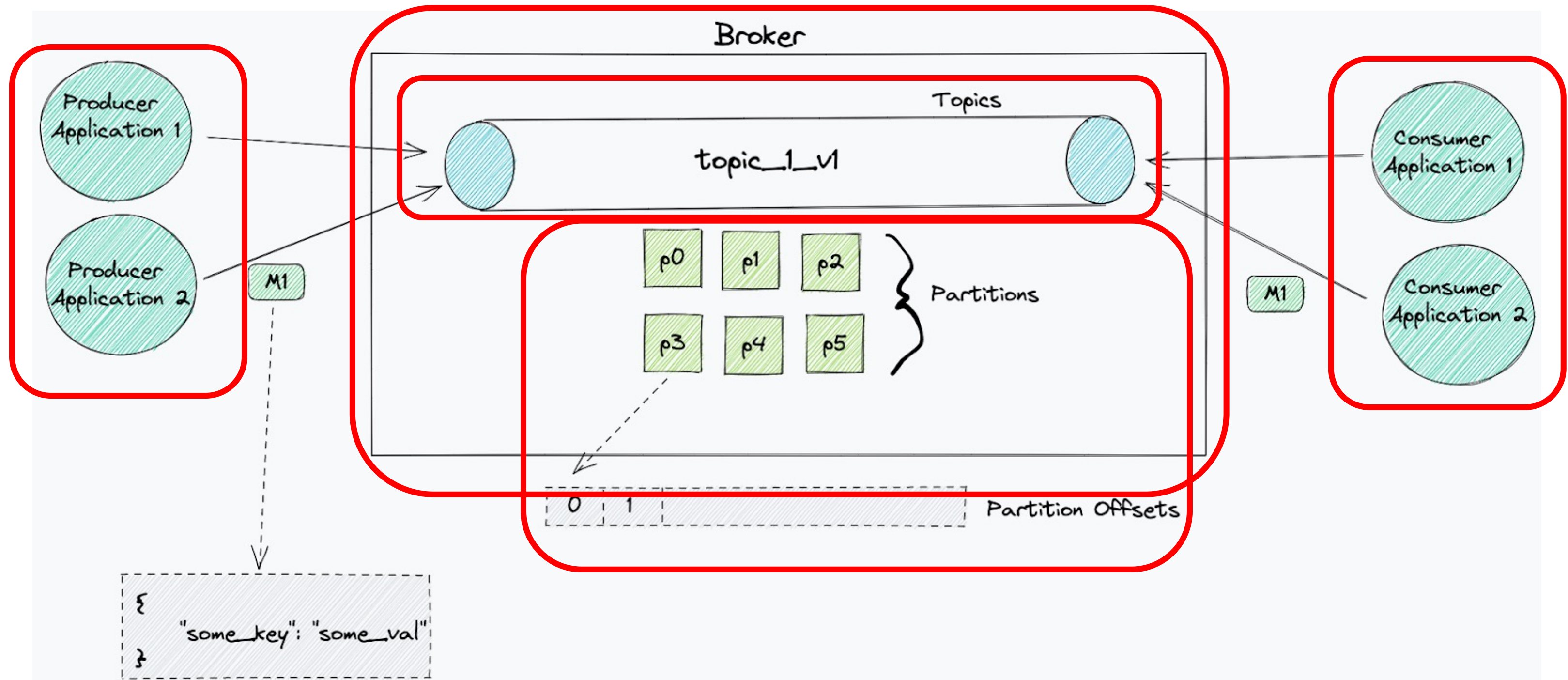


# Kafka

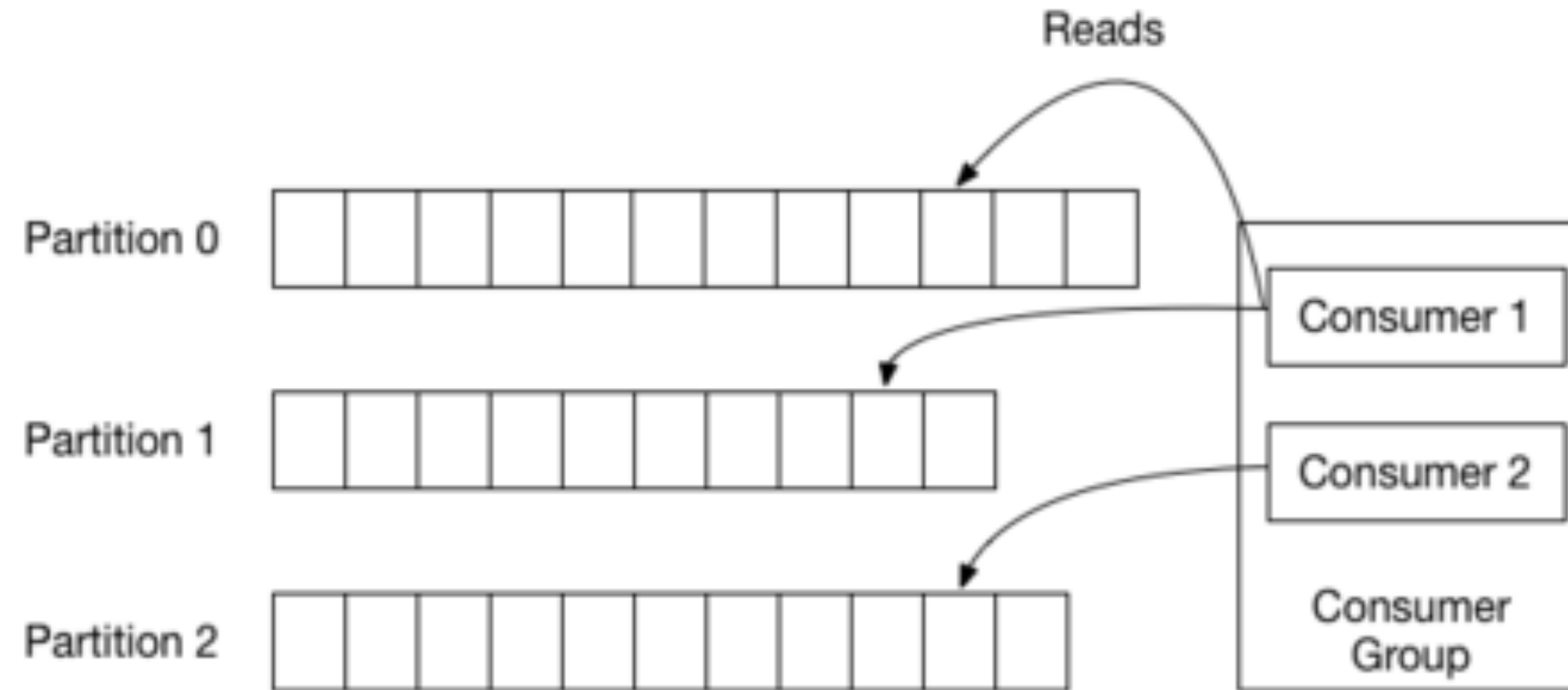




# Kafka



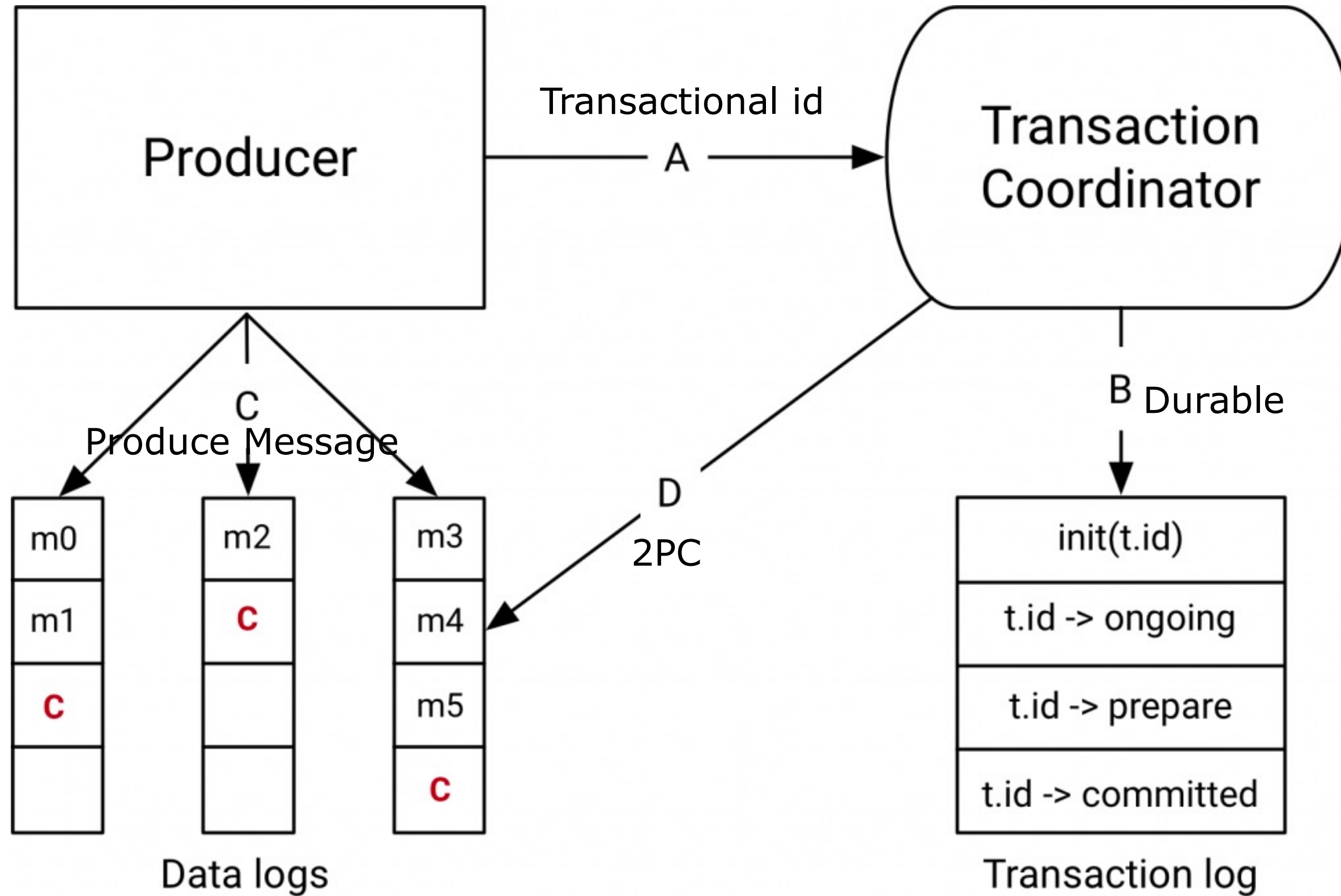
# Kafka Consumer Group



<https://medium.com/@ronnansouza/setting-up-a-kafka-broker-using-docker-creating-a-producer-and-consumer-group-with-multiple-384b724cd324>

- offset management
- distinct partition for each consumer

# Kafka Transaction





# Kafka Transaction

- Producer
  - Establish transaction
  - Produce messages
- Consumer
  - Level: read committed
- Coordinator
  - Unique transactional ID

# Kafka Transaction

```
● ● ●
produce.init_transaction()
producer.begin_transaction()
while true:
    try:
        start_time = time.now()
        messages = []

        while not len(messages) >= BATCH_SIZE and not time.now() - start_time >= TIMEOUT:
            message = consumer.poll_message()
            if not message.error():
                messages.append(message)
                start_time = time.now()

            if len(messages) > 0:
                results = process_records(messages)

                for result in results:
                    producer.send(result)

                topic_partition_list = make_TP_from_msg(message)
                producer.send_offsets_to_transaction(topic_partition_list)
                producer.commit_transaction()
                message = [] # clear messages
                producer.begin_transaction()
    except Exception as error:
        handle_errors(error, producer, consumer)
```

# Kafka Transaction

## Error handling

- 錯誤種類
  - 可重試
  - 終止交易
  - 致命錯誤

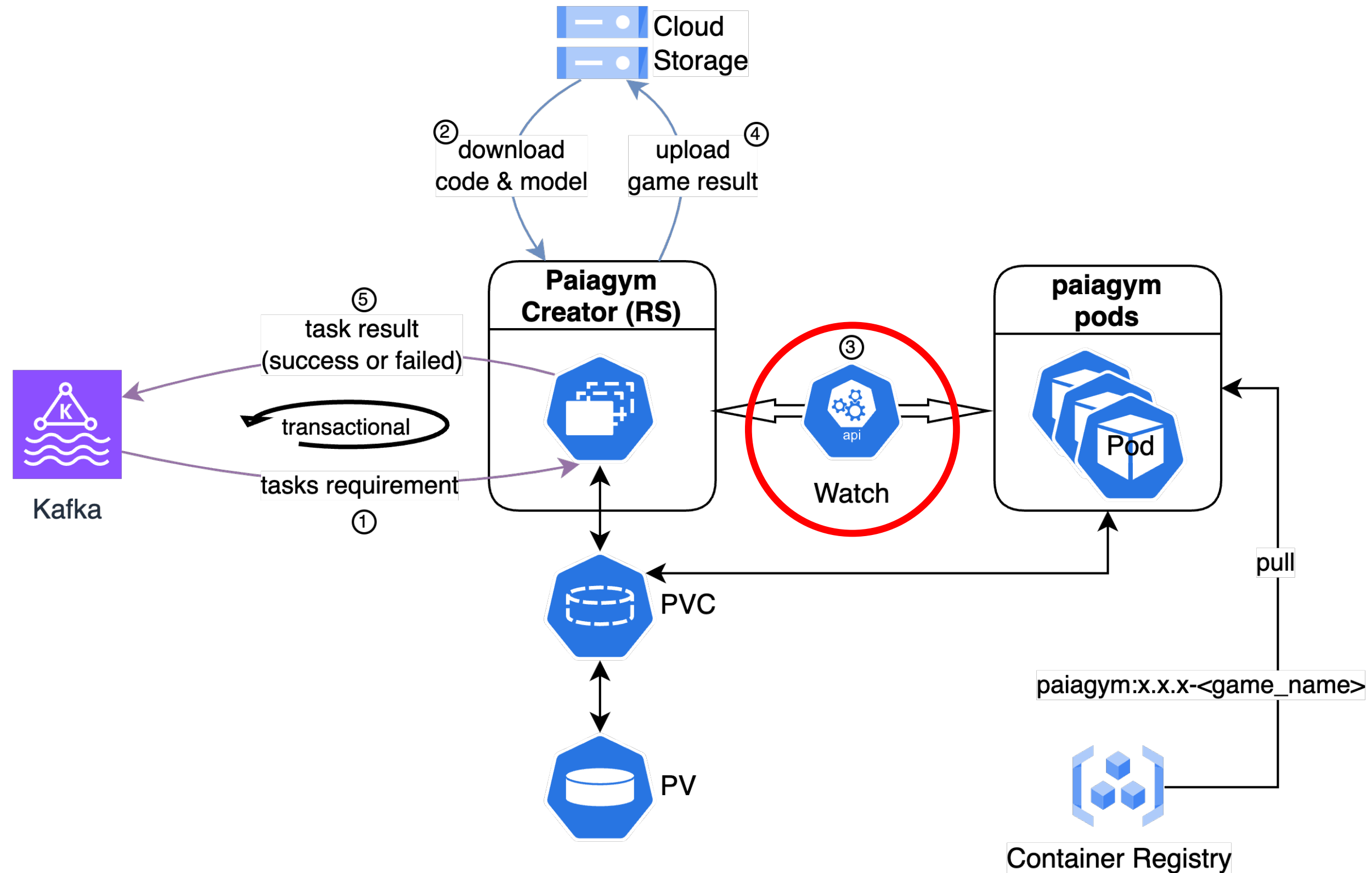


# Kafka Transaction

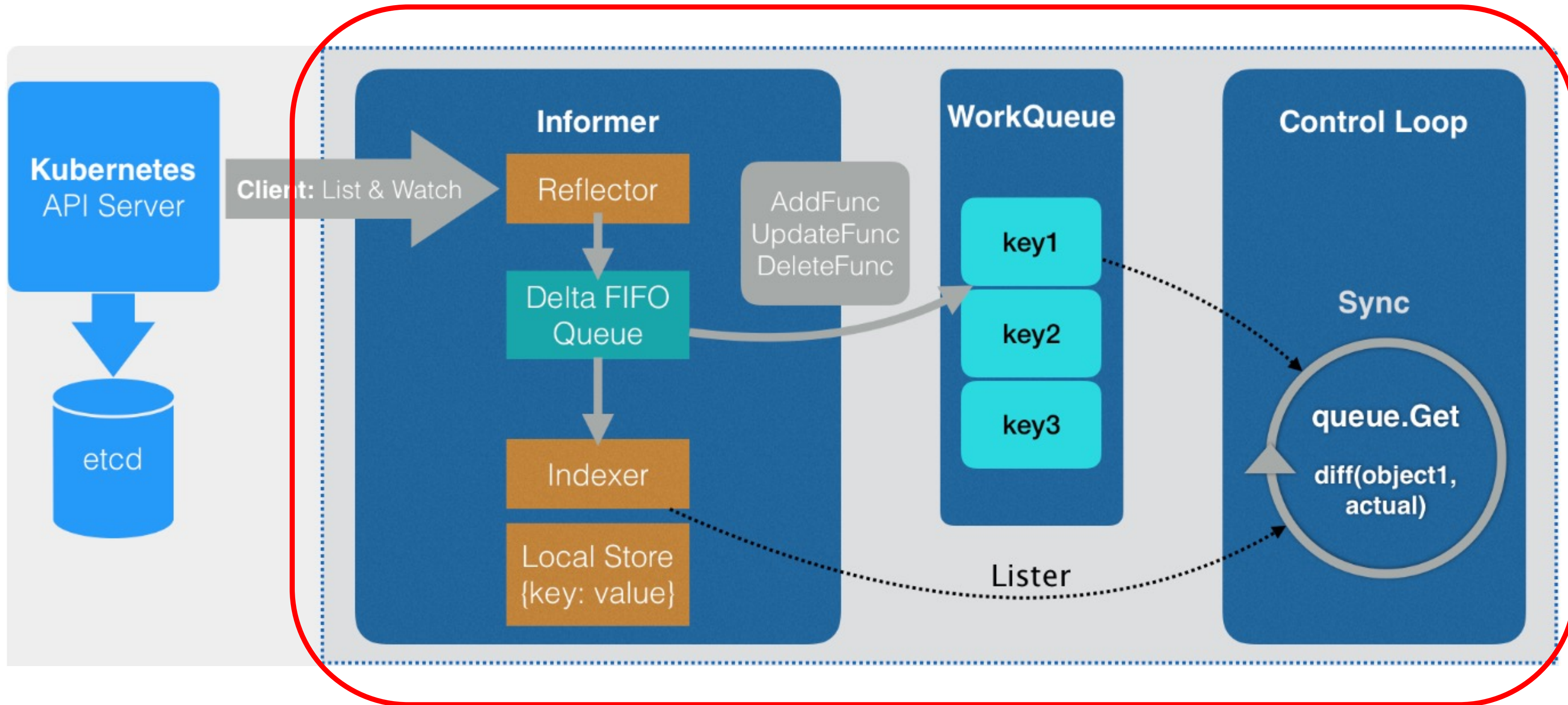
## Error handling

```
● ● ●  
  
while True:  
    try:  
        producer.commit_transaction(10.0)  
        break  
    except KafkaException as e:  
        if e.args[0].retriable():  
            # retriable error, try again  
            continue  
        elif e.args[0].txn_requires_abort():  
            # abort current transaction, begin a new transaction,  
            # and rewind the consumer to start over.  
            producer.abort_transaction()  
            producer.begin_transaction()  
            rewind_consumer_offsets...()  
        else:  
            # treat all other errors as fatal  
            raise
```

# Kubernetes Watch



# Kubernetes Informer





# Kubernetes Informer

Python Informer...

Implement an Informer in python-client #868

Open

ellieayla opened this issue on Jul 9, 2019 · 15 comments

# Kubernetes Watch API

Python generator



```
for event in w.stream(api_instance.list_namespaced_pod, namespace=namespace, timeout_seconds=1000):  
    pod = event['object']
```



```
def stream(self, func, *args, **kwargs):  
    watch_arg = self.get_watch_argument_name(func)  
    if 'resource_version' in kwargs:  
        self.resource_version = kwargs['resource_ver']  
  
    while True:  
        resp = func(*args, **kwargs)  
        try:  
            for line in iter_resp_lines(resp):  
                if watch_arg == "watch":  
                    yield event  
                else:  
                    yield line  
            if self._stop:  
                break
```

# Service-Account bind Role

For monitoring paiagym pod



rules:

- apiGroups: [""]  
resources: ["pods"]  
verbs: ["create", "get", "list", "watch", "delete"]
- apiGroups: [""]  
resources: ["pods/log"]  
verbs: ["get", "list", "watch"]






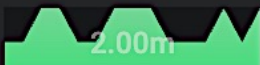






# Nvidia Device Plugin

Rendering...

- NVIDIA drivers
- nvidia-docker or nvidia-container-toolkit
- nvidia-container-runtime
  - container runtime setting (CRI-O, docker, ...)
- K8s Cluster  $\geq$  v1.10
- Device plugin helm chart
  - nvdp/nvidia-device-plugin



# Nvidia Device Plugin

 <a href="#">nvdn-nvidia-device-plugin-fmksc</a>	<a href="#">nvcr.io/nvidia/k8s-device-plugin:v0.14.3</a> <a href="#">nvcr.io/nvidia/k8s-device-plugin:v0.14.3</a>	<a href="#">app.kubernetes.io/instance: nvdn</a> <a href="#">app.kubernetes.io/name: nvidia-device-plugin</a> paia3	Running	0	 2.00m	 38.54Mi	<a href="#">17 days ago</a>	
 <a href="#">nvdn-nvidia-device-plugin-lqr4b</a>	<a href="#">nvcr.io/nvidia/k8s-device-plugin:v0.14.3</a> <a href="#">nvcr.io/nvidia/k8s-device-plugin:v0.14.3</a>	<a href="#">app.kubernetes.io/instance: nvdn</a> <a href="#">app.kubernetes.io/name: nvidia-device-plugin</a> paia2	Running	0	 2.00m	 24.36Mi	<a href="#">17 days ago</a>	
 <a href="#">nvdn-nvidia-device-plugin-qknvw</a>	<a href="#">nvcr.io/nvidia/k8s-device-plugin:v0.14.3</a> <a href="#">nvcr.io/nvidia/k8s-device-plugin:v0.14.3</a>	<a href="#">app.kubernetes.io/instance: nvdn</a> <a href="#">app.kubernetes.io/name: nvidia-device-plugin</a> paia5	Running	0	 1.00m	 39.86Mi	<a href="#">17 days ago</a>	

# Nvidia Device Plugin



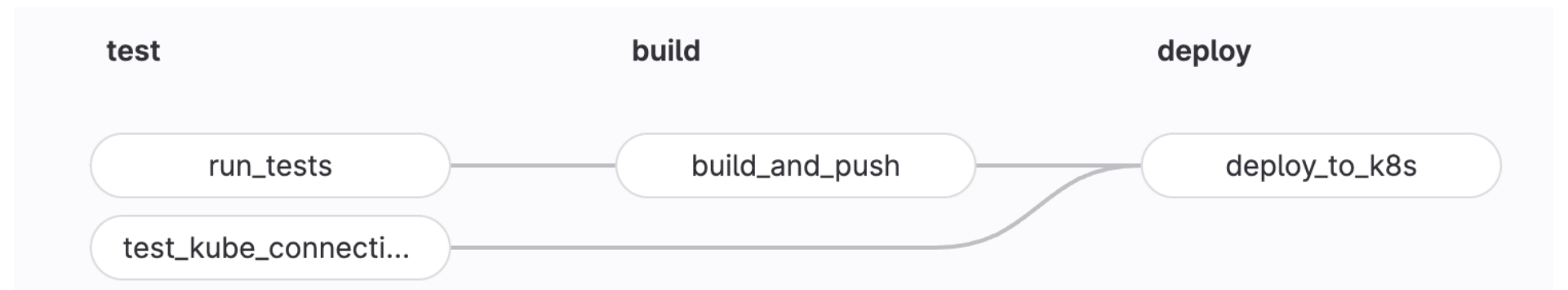
```
resources:
```

```
  limits:
```

```
    nvidia.com/gpu: 1 # requesting one GPU
```

# CI/CD

- Gitlab CI



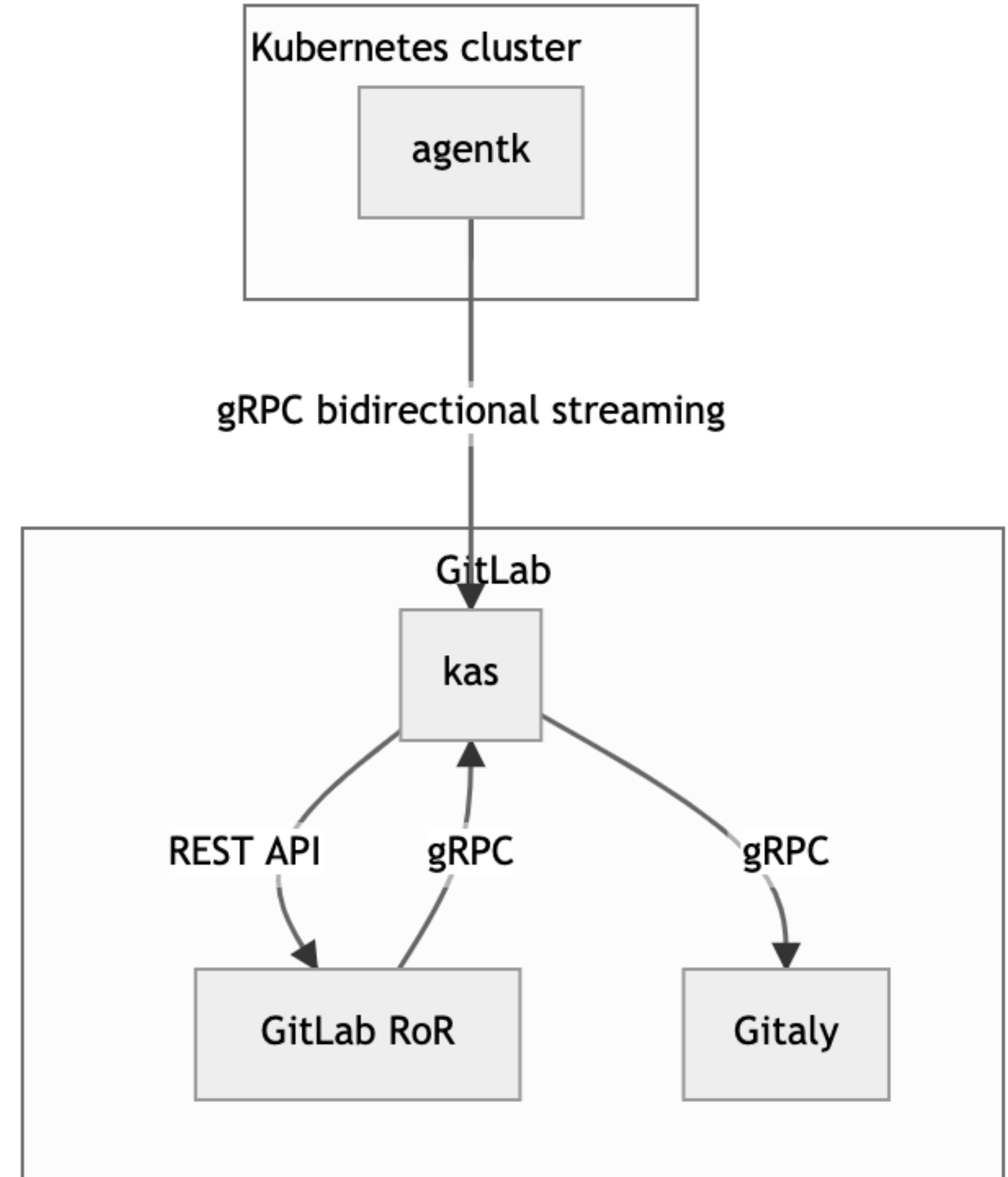
- Gitlab Runner in K8s

- Helm install
- Runner token

<span>●</span> #283 (EjQAtKbUZ)	Remove runner
gitlab-runner-f786454c9-8mm7k	
<span>●</span> #282 (3czinvAg)	Remove runner
gitlab-runner-f786454c9-wx2zt	
<span>●</span> #281 (kFo--qT8W)	Remove runner
gitlab-runner-f786454c9-cp52m	
<span>●</span> #280 (EDH2hHk38)	Remove runner
gitlab-runner-684568bd9-dp8wf	
<span>●</span> #279 (67vTr_ty2)	Remove runner
gitlab-runner-798d547796-v5pld	

# CI/CD

- Gitlab Agent
  - access the K8s cluster





# CI/CD

- Gitlab Agent

Name	Connection status	Last contact	Version	Agent ID	Configuration	
<a href="#">paia-local</a>	 Connected	5 minutes ago	16.0.1	1	Default configuration 	

# CI/CD

- Gitlab Agent & Gitlab CI

```
$ kubectl config get-contexts
CURRENT  NAME                                CLUSTER  AUTHINFO  NAMESPACE
          paia/mlgame-creator:paia-local  gitlab   agent:1
$ kubectl config use-context paia/mlgame-creator:paia-local
Switched to context "paia/mlgame-creator:paia-local".
$ kubectl apply -f paia3d/deploy/deployment.yaml
deployment.apps/paia3d-game-creator-deployment configured
$ kubectl rollout restart deployment paia3d-game-creator-deployment -n ml-game-3d
deployment.apps/paia3d-game-creator-deployment restarted
Cleaning up project directory and file based variables
Job succeeded
```

# Local K8s Placement

- Ubuntu Linux 22.04.2 LTS
- 服務
  - kubelet
  - Container Runtime: docker
  - cgroup drivers: cgroup drivers v2
  - CRI: cri-dockerd
  - CNI: Flannel
- Placement
  - 3 Control Panel + 2 Worker  $\approx$  HA K8s (v1.28)

# PAIA3D : Unity & Python

- Python 端
  - 啟動 Script
  - OpenAI Gym 的 API
- Unity 端
  - 大部分的人會用 ML-Agents
  - 我們則自己開發一個 Gymize 的框架
    - 主要功能：溝通介面、資料對應關係、錄影



# Gymize Framework

2 The Signaling Server will be auto-started if not exist yet

Signaling (Pairing) Server



WebSocket Server

pairing  
(name=env\_name)

4  
one-to-one

pairing  
(name=env\_name)

One can launch Unity application by passing the file path, or press "play" in the Unity Editor.

Python

env (name: env\_name)

AI

Gymnasium

PettingZoo

Bridge

3

launch

launch

Unity 3D (name: env\_name)



GymEnv

Scene Environment

Peer Connection (WebSocket)

5

Videos and configurations



1 Start script



# PAIAGym 介紹

- 用來執行使用者的 AI 腳本
- 執行完後把影片和遊戲結果儲存下來
- 有一個遊戲管理機制，下載安裝指定遊戲
- 方便於 Docker 內部執行



從錯誤中學習經驗

因為有錯誤才知道什麼是正確

-- S.Y. Chang

# Unity 中如何錄影？

- 畫面與音訊分開處理
  - 畫面部分
    - Unity 沒有內建錄影機制
    - 需要自行儲存每一個 Frame、時長
    - 後製銜接畫面，使用 ffmpeg
  - 音訊部分
    - 用 OnAudioFilterRead 擷取，最後銜接成wav音檔
  - ffmpeg 把 Gymize 回傳的畫面與音訊合併成影片



# Unity 中如何截圖？

- 使用 `ScreenCapture.CaptureScreenshot`
  - 圖片為 `RenderTexture` 格式，放在 GPU
  - 需要搬移到 CPU，變成 `Texture2D` 格式
  - 問題：如何消除白邊？
    - `Texture2D` 使用 `TextureFormat.RGB24` 即可

# GPU 資源的問題

- 因為牽扯到 GPU 資源的使用，延伸出一堆問題
  - 伺服器 GPU 不能直接當成桌面版 GPU
  - Docker 中不能直接使用 GPU 資源
  - 用 CPU 虛擬螢幕模擬 GPU，超慢
    - xvfb 或是 xserver-xorg-video-dummy
  - 用 Replay 的套件，降速錄影再後製回來
- 我們迫切需要 GPU 資源，但是預算不夠，怎麼辦？

五餅二魚的故事：

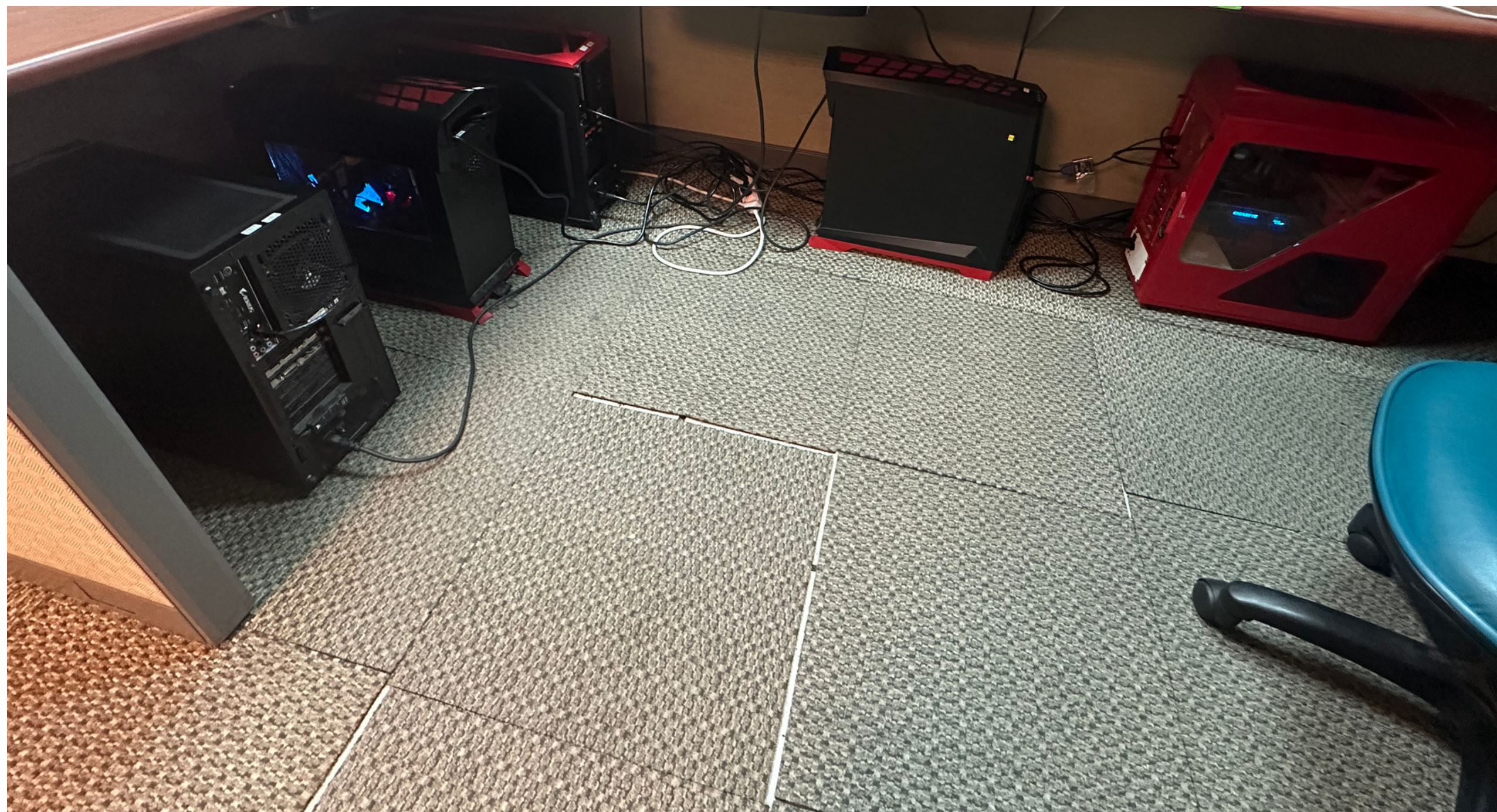
一起合作的老師送我們

五張淘汰下來但還能用的顯卡

我們就架了K8s環境



# 實體機房畫面





# GPU 心酸血淚史

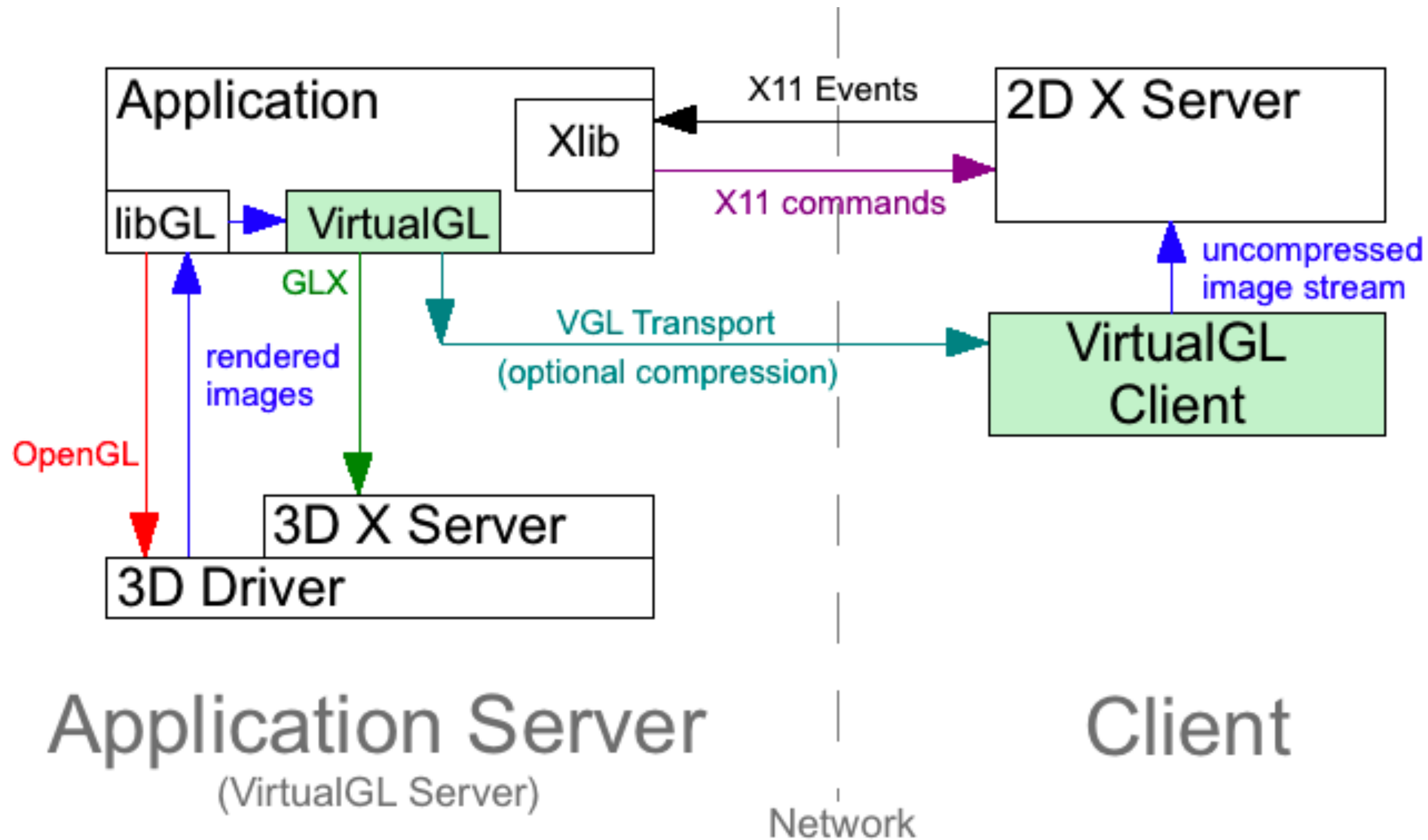
- 以為有 GPU 資源後好日子就來了？並沒有QQ
- 在地端部署 GPU 應用的過程
  - 以下內容在雲端 GPU 主機也適用
  - 尤其是 server 版本也適用
  - 我們部署在地端，是因為有現成的 GPU



# 在 Docker 中使用 GPU

- 用 Docker 才能有彈性的執行遊戲
- Docker 本身不支援 GPU
  - 需要安裝 NVIDIA Container Toolkit
- 殘酷的現實：裝完發現遊戲還是沒有用到 GPU 資源

# 嘗試 VirtualGL



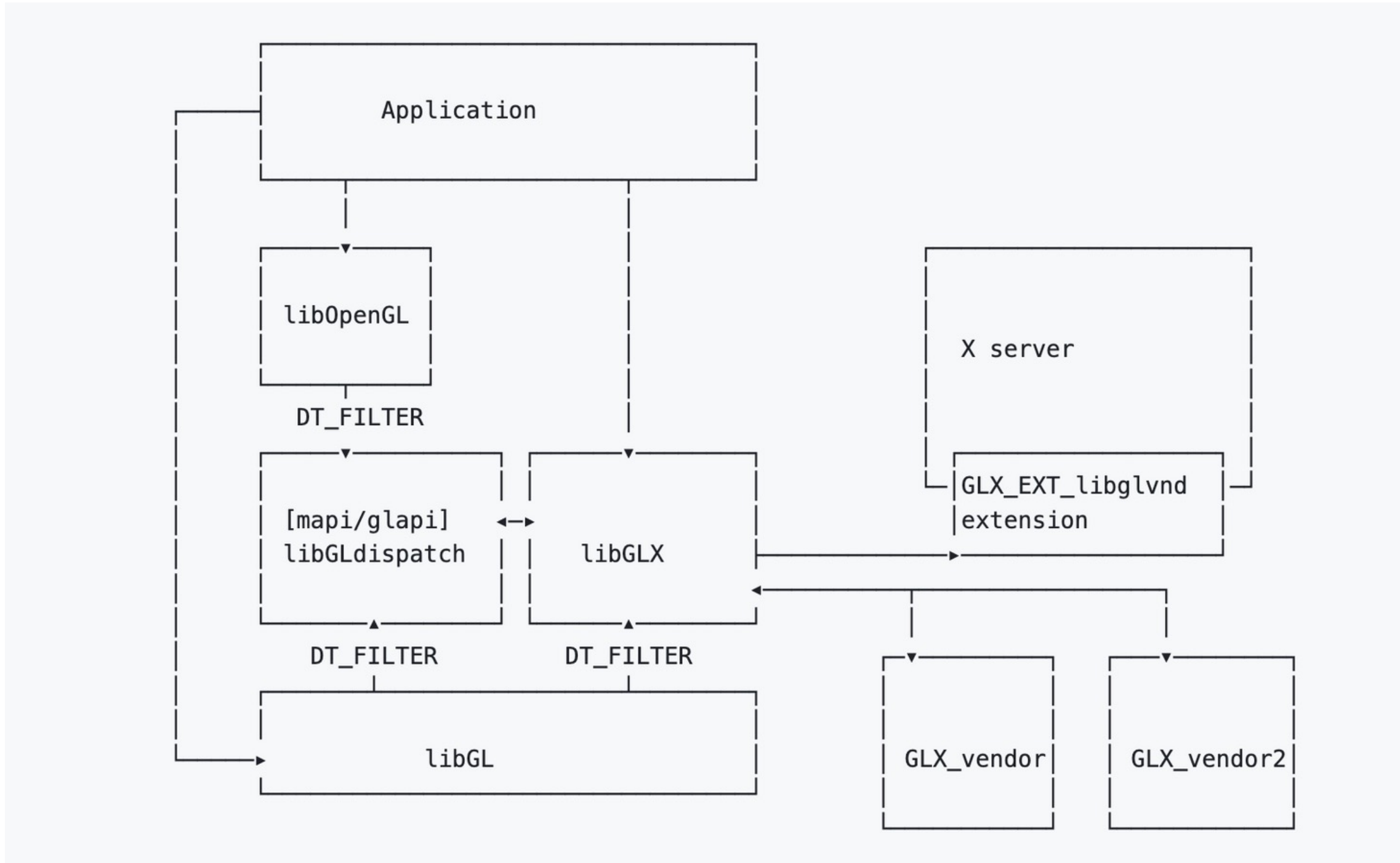
# VirtualGL 的各種雷點

- 查遍了各式各樣的教學，全部都失敗了QQ
- Docker 內很難用 GPU 3D，都預設用 CPU 渲染
  - 所以要透過 vglrun ( 使用 VirtualGL 跑的指令 )
- vglrun 也失敗的原因：會閃退，就算遊戲是正常的
- 總之 VirtualGL 和 Unity 似乎很不對頻

# nvidia/cudagl

- 用這個官方 Image 就可以在 Docker 內使用 GPU 了
  - 不需要用到 VirtualGL
- 裡面預裝 NVIDIA 提供的 OpenGL 套件 ( libglvnd )
- libglvnd: the GL Vendor-Neutral Dispatch library
  - 就算是 Server 版本 GPU 也可以跑桌面應用！
  - 搭配 X server 虛擬螢幕，可以成功啟動 Unity！

# libglvnd





# 關於 X 虛擬螢幕

- X 要裝在 Docker 外面
  - 安裝 xserver-xorg-video-dummy
  - 裝在 Docker 裡面會有錯誤：

parse\_vt\_settings: Cannot open /dev/tty0  
(No such file or directory)

- 容器需要掛載 Volume : /tmp/.X11-unix
  - 讓容器內部可以連接容器外的虛擬螢幕

# 關於 X 虛擬螢幕

- 光有 `xserver-xorg-core` 還不夠
  - `xserver-xorg-core` 包含在 `xserver-xorg-video-dummy` 之中
    - 單純只有 `xserver-xorg-core` 是可以跑 X
    - 但是會沒有螢幕，會出現：  
(EE) no screens found(EE) 錯誤
- 測試過後，跑 Unity OpenGL 不需要開 `xhost` 權限！

# 關於 X 虛擬螢幕

- `nvidia-xconfig` 設定
  - 讓虛擬螢幕可以用到 NVIDIA 資源
  - 開啟 `AllowEmptyInitialConfiguration`
  - 用 `Virtual` 設定虛擬螢幕大小
- 簡化成指令
  - `sudo nvidia-xconfig -a --allow-empty-initial-configuration --virtual=3840x2160`

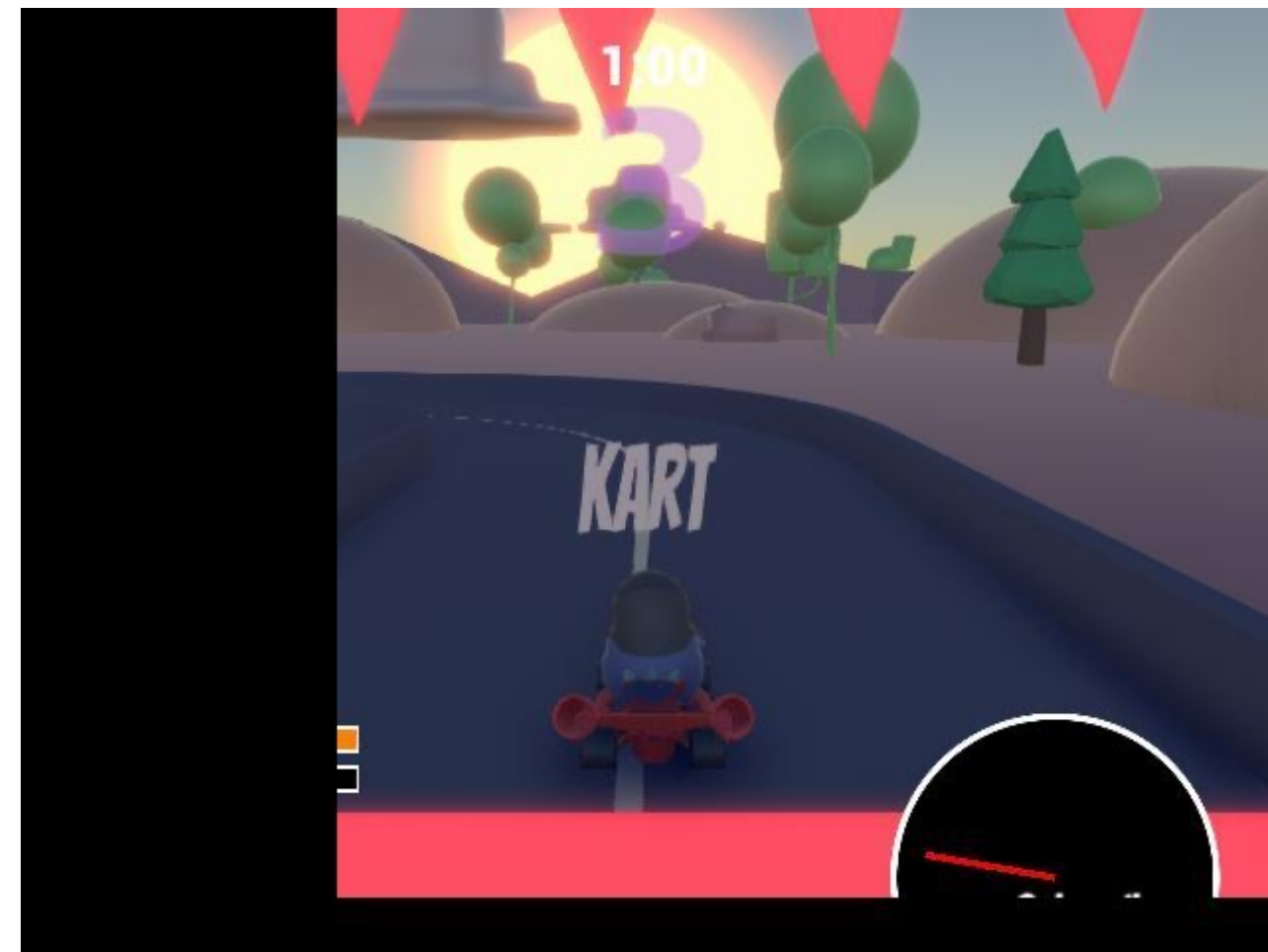
# 關於 X 虛擬螢幕

- nvidia-xconfig 設定

```
Section "Screen"
    Identifier "Screen0"
    Device "Device0"
    Monitor "Monitor0"
    DefaultDepth 24
    Option "AllowEmptyInitialConfiguration" "True"
    SubSection "Display"
        Virtual 1920 1080
        Depth 24
    EndSubSection
EndSection
```

# 關於 X 虛擬螢幕

- 之前的問題：為何遊戲畫面被切到？





# 關於 X 虛擬螢幕

- nvidia-xconfig 設定：用 Modes 會失敗，變預設值

```
Section "Screen"
  Identifier "Screen0"
  Device "Device0"
  Monitor "Monitor0"
  DefaultDepth 24
  Option "AllowEmptyInitialConfiguration" "True"
  SubSection "Display"
    Depth 24
    Modes "1920x1080"
  EndSubSection
EndSection
```

```
[517858.870] (II) NVIDIA(0): Validated MetaModes:
[517858.870] (II) NVIDIA(0): "NULL"
[517858.870] (II) NVIDIA(0): Virtual screen size determined to be 640 x 480
```

# 關於 X 虛擬螢幕

- 大禮包
  - display.sh
  - Ubuntu 開箱即可使用！
    - `sudo sh display.sh`





跑跑卡丁車3D



BLITASTR

# Conclusion

- 地端環境的軟硬體建置
- 透過VPN串聯雲端與地端
- 雲端API發送PAIA 3D任務至地端Kafka執行任務
- paiagym-creator啟動遊戲
- 建置CI/CD持續部署與更新
- paiagym與nvidia docker介紹
- Next - 地端環境擴建與PAIA 2D/3D遊戲整合



# 感謝



李緒成



張頌宇



Ivan Chiou



曾哲瀚

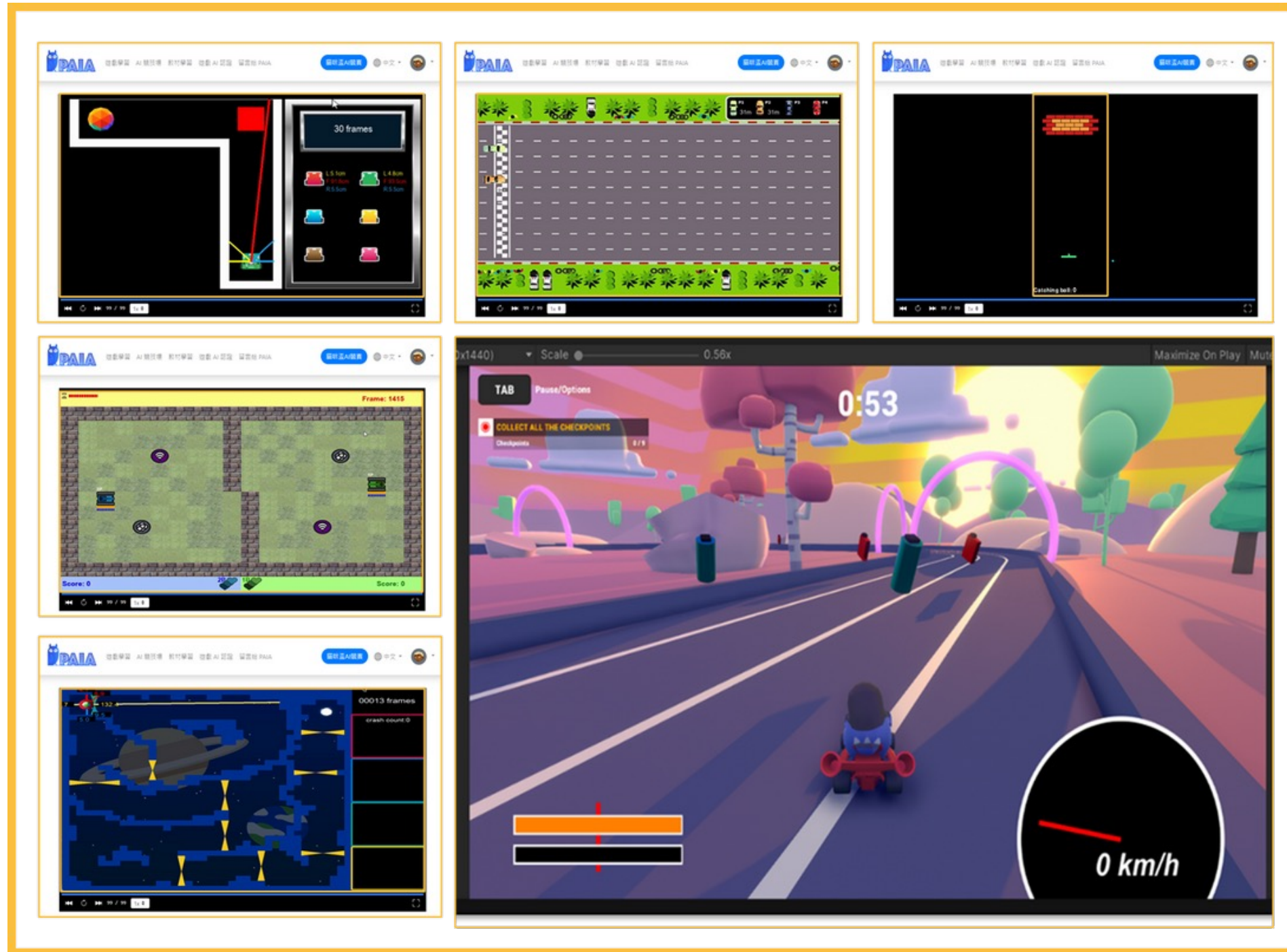


李育衡





# PAIA科技尋找前後雲端工程師



## JOIN US!



[service@paia-tech.com](mailto:service@paia-tech.com)



# Agile Neihu Sprint 40 : 學生志工能敏捷嗎? 之RSG甘苦談

📅 2023/12/18(周一) 18:30(+0800) ~ 21:30(+0800) ( [iCal/Outlook](#), [Google 日曆](#) )

📍 德明財經科技大學 綜合大樓 3 樓(資訊選手培訓基地) / 德明財經科技大學

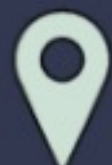
👤 0 / 30



## Agile Neihu Sprint 40 學生志工能敏捷嗎? 之RSG甘苦談



2023/12/18 (Mon.)  
18:30 ~ 21:00



德明財經科技大學  
綜合大樓 3 樓  
(資訊選手培訓基地)





# **+ GRAPYCAL**

Interactivity counts.



Interactivity counts.

目前都陸陸續續把未來期望的功能放到 GitHub Discussion  
歡迎各位的參與! 或是路過給個**星星**給與我們支持都非常感謝喔!

Facebook



GitHub



Discord



如果有任何疑問, 聯絡我們: [contact@grapycal.org](mailto:contact@grapycal.org)