

# Jason LEE

## Golang Engineer

in [linkedin.com/in/meng-ze-li-0330](https://www.linkedin.com/in/meng-ze-li-0330) blog [blog.jjmengze.website](https://blog.jjmengze.website) @ [jason.meng.lee@gmail.com](mailto:jason.meng.lee@gmail.com)  
+886 930 198 651 NewTaipei, Taiwan



目前任職於思華科技財務系統部門擔任 Golang Engineer，協助部門導入 Cloud Native 相關技術，透過 skaffold 與 telepresence 加速整體開發流程，最終能快速的整合到 Kubernetes 環境。由於系統中 redis cache 使用了大量的慢查詢，拖慢了整體系統的回應時間，以更合理的資料結構重構 redis 的使用。

在此期間導入多項技術與文化例如 DevOps 敏捷式開發、OpenTelemetry 追蹤與除錯、Kubernetes 容器管理以及相關的 Cloud Native 雲原生相關技術。

此外將舊有的服務與應用建立完整的 CI/CD Pipeline 能夠自動化的測試與部署相關服務，以及協助設置效能指標作為監控的對象，在問題發生時能透過告警系統將系統狀況即時的反應給個負責單位。

工作之餘嘗試各式開源專案並且透過實作反覆驗證與了解該專案底層實作技巧與法方精進自己，並且透過 blog 將實作過程與反思記錄下來，除了可以分享給社群還能達到日後複習的成效。

## SKILLS

Programming	Golang, Java, Kotlin
Frameworks	Kubernetes Operator, Grpc, Gin, Android, ONOS
Cloud	Kubernetes, Azure, GCP, OpenShift, OpenStack
DevOps	CI/CD, Gitlab CI, Argo CD, Terraform, Ansible
Tools	Argo Workflow, Knative
Network	Linkerd, SDN, Open Vswitch
Message Queue	Kafka, Redis, NATS
Strace/Monitor/Log	Jaeger, Prometheus, Grafana, Fluentd

## EXPERIENCE

Now Jan 2021	<b>思華科技   Golang Engineer, 現職, Xinyi</b> <ul style="list-style-type: none"><li>整合 OpenTelemetry 服務追蹤服務物行為，加快除錯流程。<ol style="list-style-type: none"><li>由於系統微服務過多在除錯與追蹤上非常困難，因此我積極導入 OpenTelemetry 技術用以追蹤服務的調用鍊與相關的 SQL 操作。</li><li>由於 OpenTelemetry 當前沒有完整整合 Gorm V1 框架，另外撰寫一個 puglins 進行整合用以追蹤 Gorm hook 事件。</li></ol></li><li>調教 legacy 系統 Redis 慢查詢事件與 cache Avalanche 等異常問題。<ol style="list-style-type: none"><li>處理 cache Avalanche 時整合了 promethues 與 alert manager 提供即時的告警讓負責的相關人員能及時收到告警通知，並且自動觸發服務熔斷與限縮請求湧入 DB 造成後續問題。</li></ol></li></ul> <div>Golang OpenTelemetry Redis</div>
-----------------	--

Now  
Jan 2021

思華科技 | Golang Engineer, 現職, Xinyi

- 導入 Skaffold 與 Kind, 加快整體開發與測試速度。
  1. 要開發新的功能到 legacy 系統中是一個十分繁瑣的事情, 需要模擬產品線上的環境又有可能弄髒自己的開發環境, 導入 docker 也需要等待 image build time, 讓整體開發速度下降。
  2. 因此我在導入 Kind 與 Skaffold 到每個開發者的開發流程中, 修改程式碼的時候就會觸發 image building, 最後部署到 kind kubernetes 環境中, 整體動作一氣呵成以便後續功能測試。
- Skaffold 仍有不足靠 Telepresence 來幫忙, 接入 remote kubernetes 環境。
  1. 開發者透過 Skaffold 建置與發佈 image 到 container registry, 並且自動化部署到 kubernetes 環境, 但是服務一但牽扯到的服務一多在開發端要部署的服務也跟著多了起來。
  2. 因應上述問題, 我置另外一個 Kubernetes 環境部署較不會變更的服務, 並且導入了 Telepresence 來幫忙導引遠端流量到本地, 以利開發者將遠端服務流量導流到本地進行後續功能測試與開發。
- <POC> 基於 AWS step function 與 Skaffold 與 Kind, 加快整體開發與測試速度。
  1. 要開發新的功能到 legacy 系統中是一個十分繁瑣的事情, 需要模擬產品線上的環境又有可能弄髒自己的開發環境, 導入 docker 也需要等待 image build time, 讓整體開發速度下降。
  2. 因此我在導入 Kind 與 Skaffold 到每個開發者的開發流程中, 修改程式碼的時候就會觸發 image building, 最後部署到 kind kubernetes 環境中, 整體動作一氣呵成以便後續功能測試。

Kubernetes Skaffold Telepresence Golang Argo Workflow Knative

Now  
Jan 2020

遠傳電信 | Software Engineer, 11-MONTHS, Neihu

- 透過 Golang 開發 Kubernetes Operator 部署及管理內部有狀態的服務 (POC)。
  1. 設計 admission controller 驗證使用者操作行為。
  2. 整合 Custom Resources Definition 用以操作企業內部有狀態服務管理的自定義資源。
- 導入 Infra as Code 之工具 Terraform, 以取代傳統在 Azure Web console 操作並部署及維護 OpenShift。
- 整合 Gitlab CI 取代原 IT 手動部署與測試行為, 以加快產品的迭代速度及上線之速度。
- 協助產品開發團隊使用與導入 Kafka、GRPC 技術
  1. 分析 Consumer 吞吐量問題並提供相應的解決方案。
  2. 搭建 Kafka 監控畫面以利開發團隊追蹤 Broker 相關性能。
  3. 解決 Kubernetes 導入 GRPC 負載平衡問題。
- 搭建混合雲 (Azure+On-premise) 部署策略與工具開發。
- 整合 Prometheus 與 Grafana 同時制定告警規則, 將服務異常行為與狀態推播至相關負責人之通訊軟體。
- 分析現行 OpenShift 環境導入 Service Mesh (Istio, Linkerd) 相關問題效益以及影響範圍。

Golang Kubernetes Gitlab CI OpenShift Terraform Kafka Grpc

November 2019	天坦遊戲   Backend Developer, 兵役前, Nangang
August 2019	<ul style="list-style-type: none"> <li>› 透過 <b>Golang</b> 撰寫後台禮券發放邏輯以及串接第三方金流。</li> <li>› 透過 <b>Golang</b> 撰寫自動化測試腳本用以測試後台 <b>WebSocket</b> 相關邏輯。</li> <li>› 將傳統 IT 人員把執行檔手動部署的方式到 Server 的方式透過 <b>Container</b> 虛擬化技術封裝，用以加速版本迭代與更新。</li> </ul> <div>Golang Websocket Container Auto Test</div>
August 2019	NCTUxNUTC   Cooperation plan, 跨校合作計畫, Taichung
December 2017	<ul style="list-style-type: none"> <li>› ETSI MANO 設計與開發 <ol style="list-style-type: none"> <li>1. 分析 Open Network Foundation 開源專案 CORD 如何從 OpenStack 環境移植到 Kubernetes 平台。</li> <li>2. 拆解 CORD <b>Makefile</b> 與 <b>Ansible</b> 安裝過程，並將元件 Container 化，撰寫 Kubernetes Helm 腳本安裝 CORD 基礎元件。</li> <li>3. 透過 <b>Golang</b> 撰寫 Kubernetes Operator 讓使用者自定義 CORD 相關元件以及管理服務的生命週期。</li> <li>4. 研究在 5G Core Network 場景中，電信業者自建機房如何透過 CORD 降低營運成本，與整合 Kubernetes 不足之特性。</li> <li>5. 使用 <b>Golang</b> 開發能夠介接 OpenVswitch 同時管理 IP 的 Kubernetes CNI Plugins。</li> </ol> </li> </ul> <div>Golang Kubernetes OpenStack</div>
August 2019	NCTUxNUTC   Cooperation plan, 跨校合作計畫, Taichung
December 2018	<ul style="list-style-type: none"> <li>› 3GPP Network Management 設計與開發 <ol style="list-style-type: none"> <li>1. 分析國際電信 3GPP 核心網路管理平台所需文件</li> <li>2. 統整符合國際標準虛擬網路管理 API 相關文件</li> <li>3. 規劃系統平台所需元件以及定義相關開發流程</li> </ol> </li> </ul> <div>ETSI 3GPP 5G-Core-Network</div>
October 2017	inwinSTACK 迎棧科技   Intern, 暑期實習, Banqiao
June 2017	<ul style="list-style-type: none"> <li>› Kubernetes Integration OpenStack with kuryr project 整合 <ol style="list-style-type: none"> <li>1. 分析 OpenStack 虛擬網路與 Kubernetes 虛擬網路異同處。</li> <li>2. 透過 Open Vswitch 整合 OpenStack 與 Kubernetes 網路將兩個叢集透過 vlan 與 vxlan 接通。</li> <li>3. 透過 Open Flow 定義規則並使用 Open Vswitch 達到 Kubernetes 與 OpenStack 網路路由與轉發。</li> </ol> </li> <li>› 分析 Kubernetes OpenContrail Container Network Interface <ol style="list-style-type: none"> <li>1. 分析 OpenContrail 插件運行方式。</li> <li>2. 搭建 OpenContrail CNI 透過商用網路設備介接 Kubernetes 網路</li> </ol> </li> </ul> <div>Kubernetes OpenStack Open Vswitch SDN</div>

August 2019  
Sep 2017

國立台中科技大學 | Computer Science & Information Engineering, MASTER, Taichung

› The Study of A Containerized NFV Orchestrator Based on Kubernetes

1. 透過 **Golang** 重新設計一個基於 Kubernetes 且符合國際標準的 ETSI MANO NFV 調度平台框架。
2. 透過 **Golang** 撰寫 **Kubernetes Operator** 使 DPDK SRIOV 等高速網路設備能夠自由掛載 Container Network Interface。
3. 重新撰寫 Kubernetes CNI 介接 OVS 使得 NFV 能夠以軟體定義網路方式控制其網路流量與生命週期。

› ACENS Designing Custom Resource Definition Mechanism using Kubernetes for Virtual Network Management on the Containerized OpenStack

1. 透過 **Golang** 設計一個 Kubernetes Custom Controller 監控自定義的 OpenStack 相關資源。
2. 設計 admission controller 驗證使用者操作 OpenStack 行為是否合法。
3. 透過 Custom Resources Definition 用以定義 OpenStack 相關資源如 virtual network, virtual storage 以及 virtual compute 等。

**Golang** **Kubernetes** **OpenVswitch** **ONOS** **NFVO**

June 2017  
Sep 2013

國立台中科技大學 | Computer Science & Information Engineering, BACHELOR, Taichung

› 普德家電——智慧生活飲水雲端監控系統 (POC)

1. 擔任 **Backend Developer**, Backend 主要由 **Golang** 開發主要將 IOT 裝置送來的資料進行彙整、重組與儲存, 提供 REST API 給手機與前端網頁查詢, 並且當資料有異常時自動發送推播給手機。
2. 當天的熱資料儲存於 **Redis Cluster** 中, 加快使用者對當天的熱資料的查詢。
3. 過往歷史紀錄則存放於 **Elasticsearch Cluster** 進行聚合彙整並且以日期作為 index。
4. 擔任 **Android Developer** 透過 **JAVA** 撰寫 Android Framework 定義畫面擺放位置與相關 UI 呈現效果, 透過 REST API 向後端取得當前水質狀況。
5. 當後台發出水質警告時 (推播), 處理推播資訊告知使用者與飲水機廠商需要更換濾芯或是其他機台設備。

**Golang** **Java** **Redis** **Elasticsearch** **Android**

## 草本屋生技——GoBuy

2015 - 2017

Project Leader/Developer

擔任 **Project Leader**，負責與廠商接洽排定開發時程並且導入**敏捷式開發**如 Kanban、Scrum 透過燃燭圖預測專案執行速度。

也參與專案的開發其中也擔任 **Backend Developer** 負責開發後台廠商新增刪除商品列表的功能以及設置限時優惠推播功能。

此外也擔任 Android Developer 主要帶學弟開發手機畫面以及元件特效呈現。

Golang MySQL Docker JAVA Android

## 雲主易——雲主易食譜這樣做 (POC)

2016 - 2017

Developer

透過 **Golang** 開發後台，負責兩個模組的開發分別是，店家上傳食譜與照片區塊，會員評論食譜與留言區塊，在期間導入 **Docker Container** 技術將程式碼打包成 Container Image 後加快廠商部署時間與能夠排除環境造成的相關問題。

Golang MySQL Docker

## 衛生福利部——Cwis

2016 - 2016

Project Leader/Developer

擔任 **Project Leader** 與廠商協議 RESI-API 格式與內容，並且討論 APP 畫面如何呈現效果，開發主要介接衛生福利部開發的 REST-API，透過 Google MAP API 插入部分圖資，方便家長透過手機找尋離家最近托嬰中心並且能夠留下相關評論與照片，此外導入 Android 自動化測試工具:monkey test。

JAVA Android

## 常茂科技——QBEE 智慧電視

2015 - 2016

Developer

透過 **JAVA** 在 Android 機上盒開發音樂播放器，主要開發播放器界面與串接第三方音樂播放 Jin Library，使用者可以透過滑鼠或是遙控器操作音樂播放介面。

JAVA Android

## 迎棧科技——CEPH MONITOR

2015 - 2015

Developer

使用 **JAVA** 串接 **Ceph Monitor** REST API 即時監控 Cloud Storage 狀態，並開發聊天界面，機房管理員能透過手機通訊了解目前儲存叢集的狀態，並能夠緊急聊天室將相關人員自動加入。

JAVA Android

## 客家文化莊——客家數位探索樂園

2014 - 2014

Developer

主要透過 **JAVA** 開發 Android APP 介面與串接 unity 實境，其中跨廠商溝通與 Unity 環境設置為專案開發最大成本。

JAVA Android

## HONORS AND AWARDS

Speaker	Cloud Native Taiwan User Group(CNTUG)
Speaker	COSCUP x GNOME.Asia x openSUSE.Asia 2018
Speaker	台中自由軟體愛好者社群 (TFC)
Volunteer	Cloud Native Taiwan User Group(CNTUG)
Contributor	Kubernetes official Document
Contributor	Kubernetes cluster lifecycle SIG