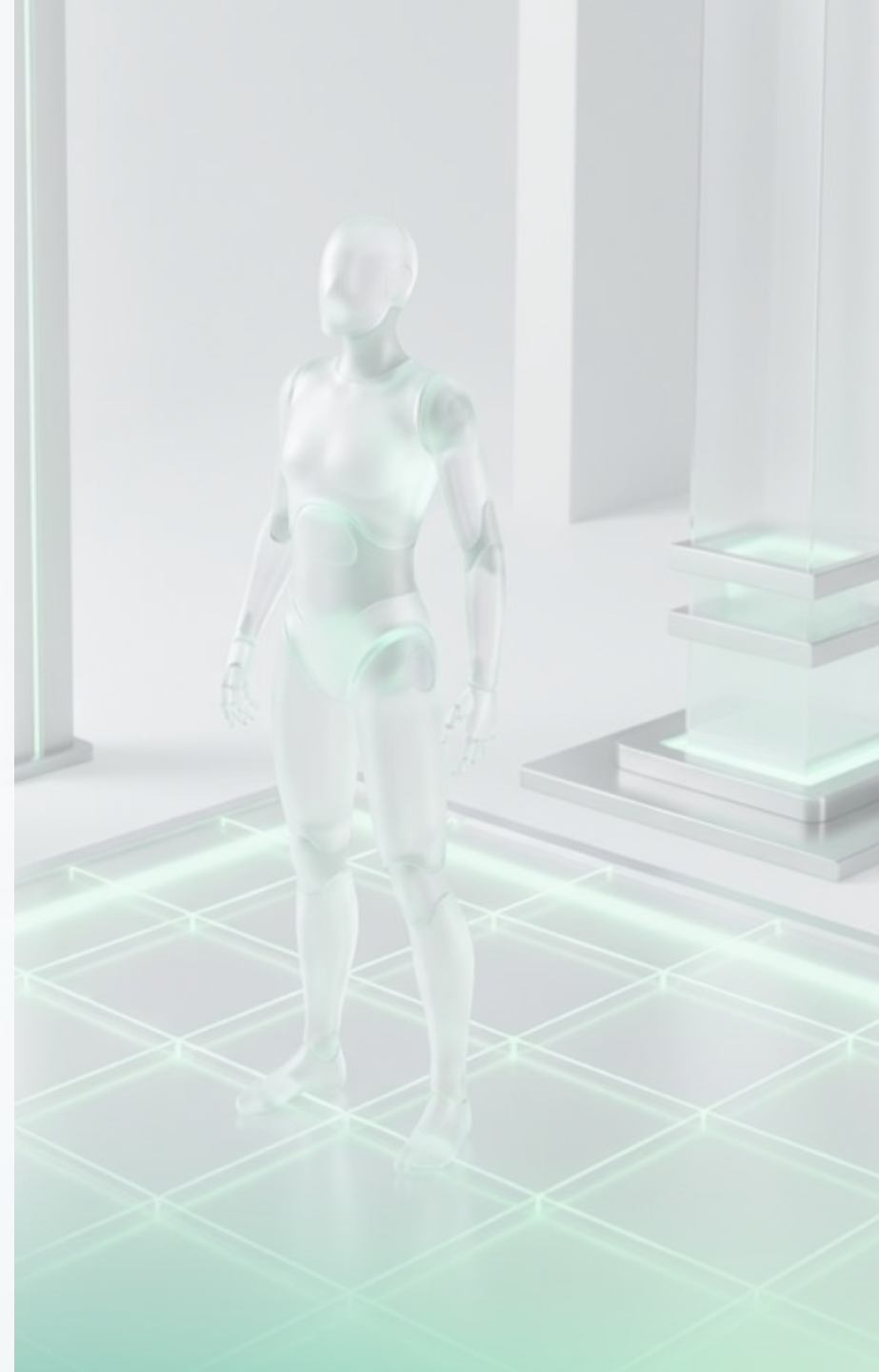


從 Cloud First 到 Agent First

企業 IT 為何要提前為 AI Agent 打底

從個人 Agent、團隊 Agent，到企業級 Agent Platform 的
實戰心得



顏勝豪 Otto



現在

- 國泰金控 數數發中心 雲端策略發展部協理
- 負責擬定集團上雲策略，並執行雲端轉型計劃

過去

- 10 年以上的軟體專案開發經驗
- 歷經紮實的養成訓練：
**程式設計師 → 系統設計師 → 系統分析師
→ 技術經理**
- 帶領專案開發部門承接多種領域之
 - 外商銀行交易系統、商業智慧分析
 - 警政情資整合分析平台、大數據分析平台
架構設計與建置

AI Agent 的想像很美，但企業現場很硬

Agent 最大瓶頸不是模型，而是企業系統能否被安全接入。

- 大家開始很有感地看到 Agent 可以做什麼。
- 真正落地時，資料、權限、API、流程會一起浮上來。
- 今天用三個層次回答：個人、團隊、企業。

01 / 核心提問

企業，真的準備好迎接 Agent 了嗎？

業界狂熱追逐 AI Agent 技術，但多數企業忽略了另一個更致命的問題：我們的環境成熟度，是否足以支撐自主運作的智能體？



核心提問：企業準備好了嗎？

個人、團隊、企業：Agent 能力的三段放大

個人 Agent

從 OpenClaw/Toto 的 40 天使用經驗，看見 Agent 的真實能力邊界。



團隊 Agent

AI Scrum Team 的 6+1 模式，AI 成為流程中的第七人。



企業 Agent

當 Agent 從 1 個變 500 個，核心變成平台工程、治理與營運。

Agent First 不是把 AI 接到企業裡，而是把企業變成 AI Agent 可以安全工作的地方。

OpenClaw 經驗：第一次覺得 Agent 真的在工作

它不是多一個聊天視窗，而是一個住在電腦裡、可以被指派任務的 Agent。

40+

天冒險

30+

One Pages

12

Skills

4

AI Agents

AHA MOMENT

不是回答，而是持續做事

新聞摘要、全文抓取、Podcast、one page，在日常中形成例行工作流。

BOUNDARY

越有用，越需要邊界

營位可以幫忙監控，刷卡付費必須拒絕。
個人場景已經會碰到企業級問題。

Ref: <https://share.ottoyen.dev/otto-toto-journey>

從 Prompt 到 Skill : Agent 的能力不是問答，是可重複 workflow

01

抓取新聞與資料來源

02

整理全文與脈絡

03

產生摘要、頁面或語音

04

沉澱成可重複 Skill

- 每日新聞摘要：抓新聞、上傳 NotebookLM、生成語音摘要。

- The Verge 全文抓取：不只抓標題，而是進入每篇文章抓全文。

- Github 編輯精選 Podcast：抓文章、整理、產生語音。

安全紅線：Agent 越能做事，越需要邊界

個人使用已經會遇到權限、金流、個資與外部網站帳號。企業場景會把這些問題放大十倍、百倍。

- **可以**：監控營位、整理資訊、產出分析頁。
- **必須詢問**：登入、提交表單、對外發布。
- **必須拒絕**：未授權金流、敏感資料外流、高風險操作。

當個人 Agent 開始像助理，下一個問題就是：它能不能進入團隊流程？

這是從個人效率走向組織能力的轉折點。

Openclaw Harness Engineering

用馬車比喻 AI Agent 的編排、控制與執行



1. 駕駛者 = Orchestrator / Human-in-the-Loop

定義目標、審核關鍵決策、處理例外



2. 韁繩 = Prompt / Policy / Control

把任務、限制與節奏傳給 Agent



5. 馬車 = Runtime / Workflow Container

承載任務狀態、上下文與輸出



6. 車輪 = Tools & Integrations

Browser、Code、API、File、Shell



7. 路線 = Plan / Workflow

持續分解目標、規劃下一步、動態校正

什麼是 Harness Engineering?



讓 Openclaw 不只是『有模型』，
而是具備可工作的 Agent 系統



明確角色、任務目標與成功條件



工具接線、權限控制與執行邊界



上下文 / 記憶管理，讓任務可持續推進



觀測、回饋、重試與人工介入機制



護欄設計，兼顧效率、安全與穩定性



3. 馬具 = Harness Engineering

把模型、工具、記憶、上下文與護欄穩定串接



4. 馬匹 = Agents / Executors

推進任務、執行步驟、呼叫工具



重點：AI Agent 的效果，不只取決於模型能力，更取決於 Harness 如何把人、模型、工具與流程有效連接。

6+1 人團隊的 +1 才是重頭戲

AI 不是開在瀏覽器裡，用完就關掉的工具。它是有主機、有帳號、有身份、全程參與的團隊成員。

以 **AI-First** 思維驅動的敏捷開發團隊，**人操作 AI，AI 操作工具**，以AI 為主、人為輔的運作方式

團隊編製

6 成員 + 1 AI 成員

AI 全程參與、不只是工具

併行開發

需求驅動開發

先定好規格再同步開工
開發、設計、測試同時進行

PM 專案經理

定需求、進度管理

1 人

Design 設計師

畫面設計、使用體驗

1 人

RD 工程師

寫程式、做功能

3 人

QA 測試

品質把關測試管理

1 人

AI Member

自動化會議紀錄 /
程式碼監控 / 日報週報
知識整理 / 輔助開發

需求先行，就是先把事情講清楚，再讓大家一起動手。

PM 透過AI來產出需求文件，開發、設計、測試同步開工、互不卡住。

Claude Tag：把 AI 放進團隊頻道，而不是個人對話框

Claude Tag 讓團隊在 Slack 直接 @Claude 指派任務。它不是一次性的聊天工具，而是存在於 channel 裡、共享上下文、可追蹤進度的 AI 隊員。

CHANNEL IDENTITY

共享身份

每個頻道都有自己的 @Claude，像團隊成員一樣被大家看見。

CONTEXT

累積上下文

理解專案背景、團隊術語與討論脈絡，減少重複解釋。

ASYNC WORK

非同步工作

可拆任務、呼叫授權工具，完成後在線程回報結果。

GOVERNANCE

企業控管

管理者控管頻道、工具、資料、預算與操作紀錄。

sprint-planning

PM：@Claude 幫我們整理這週需求異動，標出需要 PO 決策的項目。

@Claude

收到。我會彙整 channel 討論、連結的 issue 與 PR，並在這個 thread 回報 blockers。

Team

這就是第七人：不是問完就消失，而是留在流程裡，幫團隊維持共同上下文。

第七位成員做什麼：從幫忙到參與流程

SPRINT PLANNING

整理需求背景

找相依性、拆任務、補足驗收條件。

DAILY SCRUM

追蹤阻塞

整理昨日與今日進度，提醒尚未落地的承諾。

DEVELOPMENT

協助交付

參與 code review、測試、文件與 PR 摘要。

REVIEW / RETRO

沉澱改善

整理決策紀錄、缺陷模式與下一輪改善建議。

團隊 Agent 的價值不是取代某個角色，而是降低團隊協作的摩擦與上下文流失。

它是在監督團隊，還是在幫團隊？

監控式 Agent

目標是找誰沒做事，團隊會把它視為壓力來源，最後只剩形式化配合。

助攻式 Agent

目標是讓團隊更容易完成 sprint goal，補上下文、提醒阻塞、減少重工。

同樣是 Agent，目標設定不同，團隊感受會完全不同。



Dasher 應用 2026/6/26 晚上8:01

整體雷達 (2026-06-26)

- ida-km：完成度中高，s2s auth 觀測已合併、回饋處理與 auth 收口仍開著；偏離度中，dev seed 受 private-IP 路由卡住，依賴 cloud-infra 修通。
- ida-admin：完成度高，上傳者中文名、權限 grant 與靜態資源修正已合併；偏離度中低，知識庫列表搜尋仍依賴 KM #316 契約。
- ida-engine-poc：完成度中，今天可盤點的紀錄不多；偏離度中，主要風險是 engine→KM OIDC/Cloud Run 第四道驗證未收斂。
- ida-channels：完成度高，熱門提問、含圖引用、讚倒讚多項已合併；偏離度中，倒讚端到端仍受 KM 503 與回饋可修改語意影響。
- ida-agent-portal：完成度高，KM Proxy OIDC、SSO 拓樸註記與 LLM Gateway 文件已合併；偏離度中，風險在跨網域 SSO 與 Gateway/Engine 契約落地。

AI 成員也需要 onboarding

IDENTITY

給它身份

帳號、主機、repo、API
與文件存取權。

RULES

給它規範

哪些可以做、哪些要先問
、哪些永遠不能做。

CONTEXT

給它上下文

產品目標、團隊慣例、
Definition of Done、架構
原則。

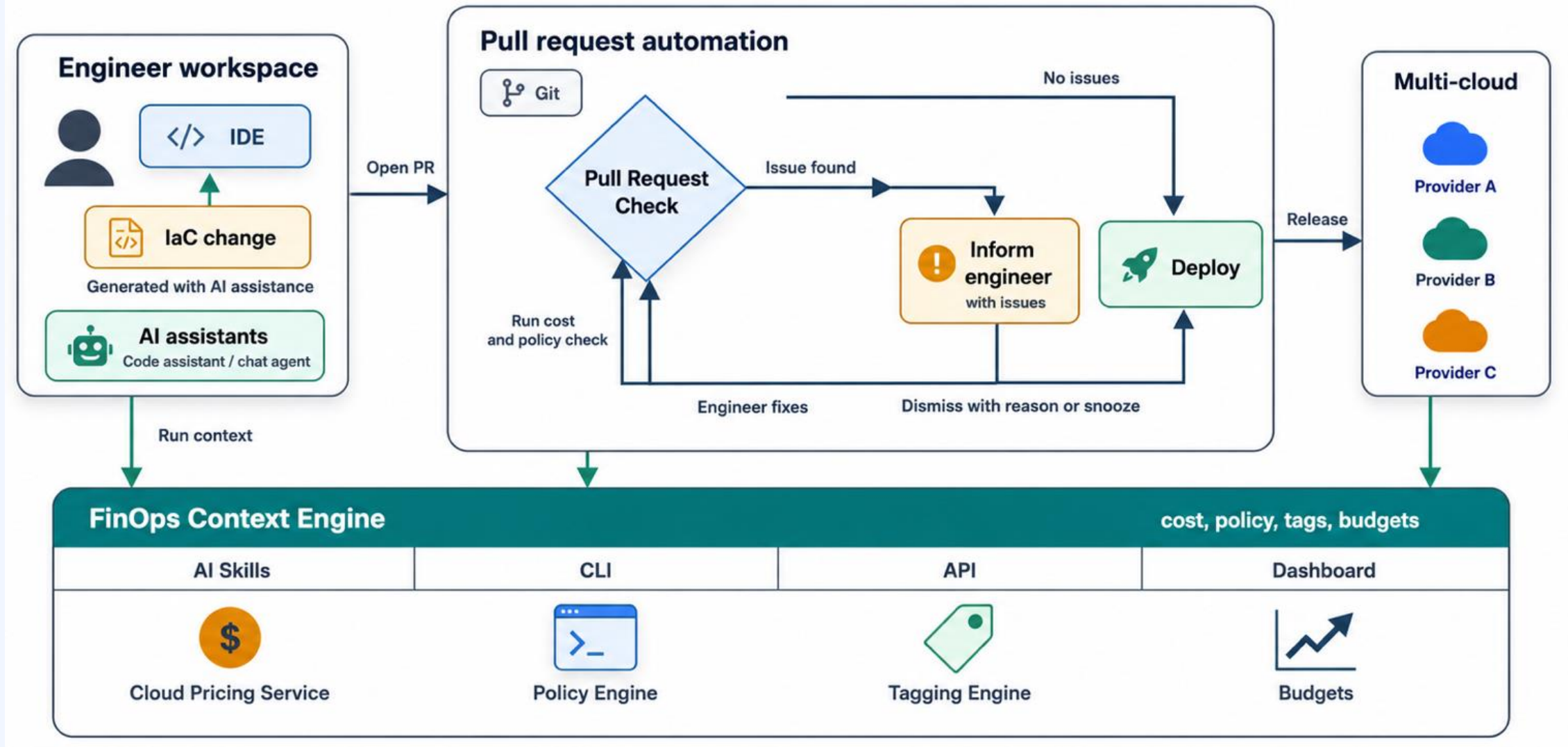
TELEMETRY

給它觀測

它做過什麼、用了哪些工
具、輸出了什麼結果。

當一個團隊可以有第七人，企業很快會問：那可不可以每個團隊都有？每個流程都有？

很適合做為團隊AI成員 — FinOps Agent



大型企業真的準備好迎接 Agent 了嗎？

- 很多 PoC 停住，不是因為模型不夠聰明。
- 卡住的是資料、權限、API、流程、治理與安全。
- Agent 不是一個 App，而是會觸碰整個企業能力的數位勞動力。

01 / 核心提問

企業，真的準備好迎接 Agent 了嗎？

業界狂熱追逐 AI Agent 技術，但多數企業忽略了另一個更致命的問題：我們的環境成熟度，是否足以支撐自主運作的智能體？



從技術狂熱回到環境成熟度

國泰上雲路徑：Cloud Ready、Cloud Adoption、Cloud First

Cathay 6R 與百套系統上雲經驗，提供大型金融集團轉型的共同語言。

2027

2020 Cloud Ready 啟動集團上雲計畫

- 應用系統上雲評估
- 進行雲端架構設計
- 建立雲端遷移計畫
- 建立組織治理框架

基礎架構

管理治理

組織人才

應用系統

2021 Cloud Adoption 5 年 100 套系統上雲

- 提供SaaS上雲轉型平台服務
- 致力雲端應用及技術發展
- 建立雲端治理組織框架
- 打造高安全、高可用雲端環境



國泰金控
Cathay Financial Holdings



國泰世華銀行
Cathay United Bank



國泰人壽
Cathay Life Insurance



國泰產險
Cathay Century Insurance

21

43

28

8

2025 Cloud First 雲端優先加速 IT 現代化

- 解決技術債問題
- 實現永續發展
- 加速業務創新
- 現代化資安與合規

成本節省
Serverless 架構

-39%

時間

員工生產力
容器化技術

+80%

管理效率

營運韌性
托管型資料庫

-13%

時間

商業敏捷性
托管型分析服務

-20%

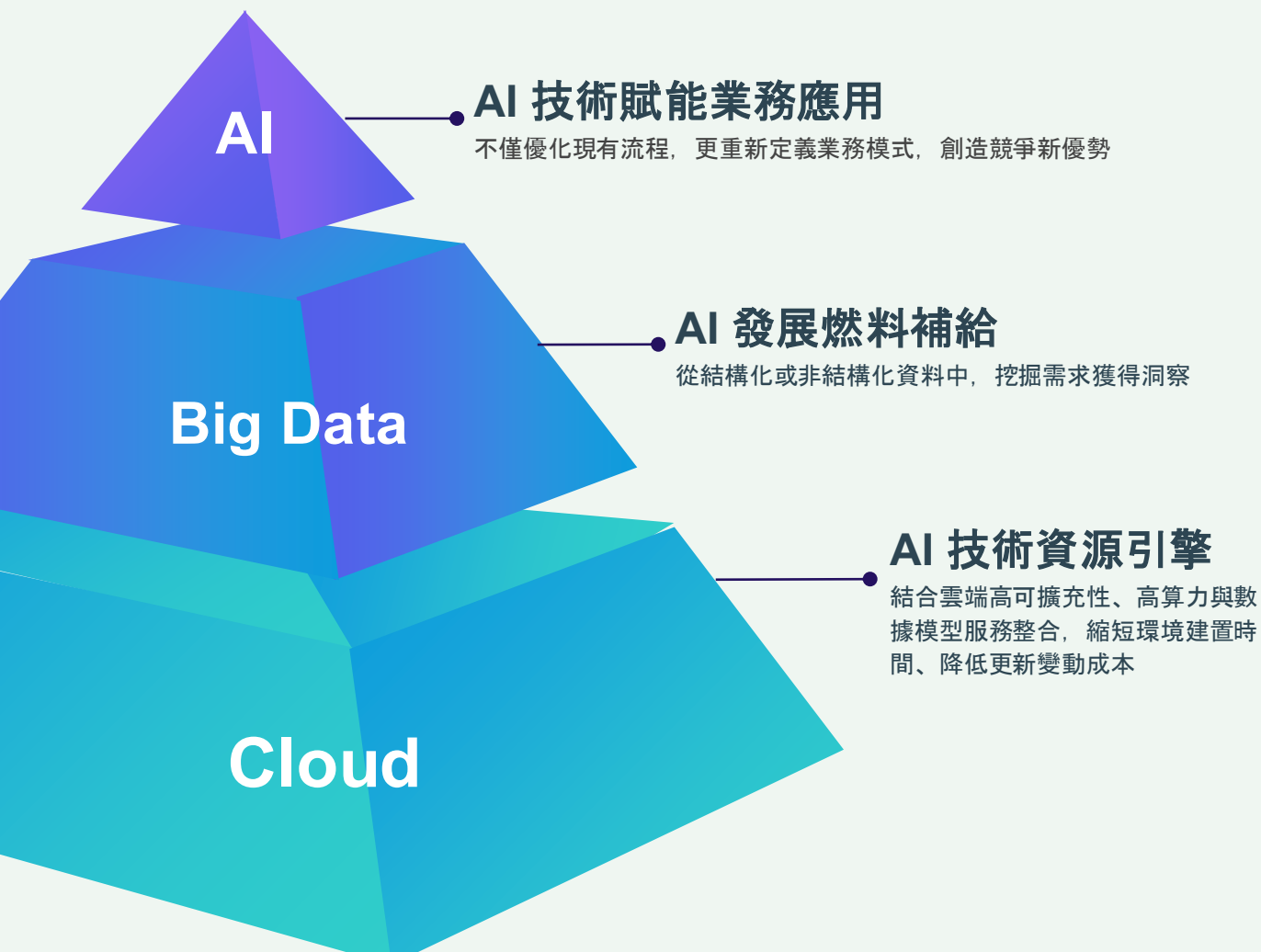
時間

永續性
實現永續發展目標

-60%

能源消耗

Cloud First 其實是在為 Agent First 打底



Landing Zone、IAM、治理、安全、標準化、流程，當時看起來像雲端管理要求。回頭看，這些就是 Agent 可以安全工作的基礎。

我們以為在為 Cloud 做準備，其實也在為 AI 做準備。

大型金融集團的現實條件

監理

合規不能事後補，必須在平台設計階段納入。

安全

身份、權限、資料存取與操作邊界要先設計。

既有系統

主機、資料庫、批次流程與舊 API 不會突然消失。

跨公司治理

子公司需要共同語言與可執行的標準。

組織共識

高管、架構師、開發團隊要能用同一套框架決策。

AI Agent 依賴的企業核心能力

IDENTITY

Agent 是誰？

KNOWLEDGE

可以看哪些知識？

TOOL / API

可以呼叫哪些系統？

WORKFLOW

如何嵌入流程？

Agent Runtime

OBSERVABILITY

如何知道它做了什麼？

SECURITY

如何控管？

AUDIT

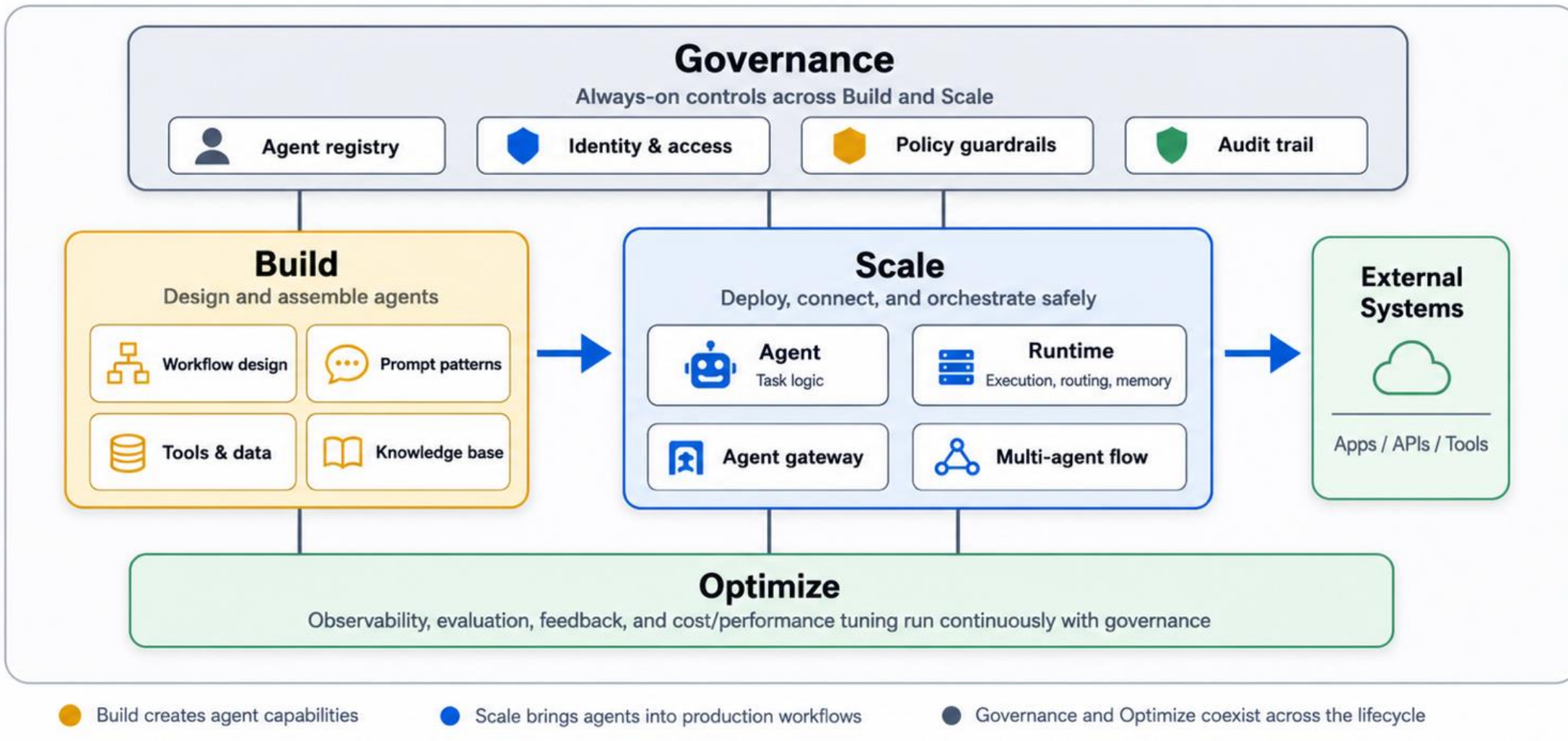
如何追蹤與稽核？

LIFECYCLE

如何上線、退場與更新？

AI Agent Development Lifecycle

Four stages for building, scaling, governing, and continuously optimizing enterprise agents



從治理雲端到治理 Agent：核心不變，對象轉移

- 多年累積的治理機制，可以從管理系統延伸到管理 Agent。
- 管理對象從雲端資源與應用系統，轉為智能代理與數位勞動力。
- AgentOps 可以借鏡 CloudOps、DevOps、MLOps、FinOps 的成熟做法。



打造一個 Agent 很容易，管理五百個才是真正挑戰

1

個人 Agent 靠信任與手動修正。

7

團隊 Agent 需要 onboarding、權限與工作協議。

500

企業 Agent 需要平台化：
身份、授權、工具目錄、知識庫、審計、觀測、成本與生命週期。





從數位基礎建設到數位勞動力

Cloud First

建立 digital infrastructure：彈性、標準、治理、安全與可營運。

Agent First

形成 digital workforce：能安全接入、跨流程協作、被觀測與稽核。

企業今天為雲端、資料、權限、流程與治理打的底，就是明天 AI Agent 能不能真正上工的分水嶺。