

研究主持者：



Google Cloud

# 加速發展： 2023 年 開發運作 現狀報告



Premier 賛助者：



Deloitte.

Qarik

# 目錄

## 序章

內容提要 .....	03
概念與評估指標 .....	06

## 第 1 章

相較之下,您的團隊表現如何? .....	10
----------------------	----

## 第 2 章

以使用者為主 .....	17
--------------	----

## 第 3 章

技術能力對成效的助益 .....	20
------------------	----

## 第 4 章

說明文件奠定成功基礎 .....	27
------------------	----

## 第 5 章

可靠性帶來更高成效 .....	31
-----------------	----

## 第 6 章

彈性基礎架構是成功關鍵 .....	38
-------------------	----

## 第 7 章

營造健康的文化才能推動革新 .....	45
---------------------	----

## 第 8 章

身分與身心健康的關聯性: 原因、影響方式和時機 .....	51
----------------------------------	----

## 後記

結論 .....	57
----------	----

特別銘謝 .....	58
------------	----

作者 .....	59
----------	----

研究方法 .....	62
------------	----

作答者和企業統計結構 .....	72
------------------	----

模型 .....	81
----------	----

延伸閱讀 .....	91
------------	----

附錄 .....	92
----------	----

所有引用資料均檢索於 2023 年 9 月 27 日

## 序章

# 內容摘要

近十年來,DORA 研究計畫致力調查高績效、技術導向的機構具備哪些能力及採取什麼措施。我們從各種規模和類型的產業收到超過 36,000 位專業人士的回覆,感謝各位與我們分享寶貴見解!

DORA 試圖釐清工作方式(亦即能力)與成果之間的關係,也就是實質成就與整個機構和人員之間的關聯性。這項研究採用嚴謹且各平台通用的統計評估法(參見「[研究方法](#)」一節)。

我們期望這些洞察結果能為主管和從業人員指引方向,成功發揮影響力。

今年的研究探討了三個關鍵成果,以及實現這些成果的能力:

- **機構績效**:機構不能只創造收益,還需要為消費者和廣大社群創造價值。
- **團隊績效**:應用程式/服務團隊是否能創造價值、創新及合作。
- **員工身心健康**:機構/團隊應採取對員工有利的策略,以減少工作倦怠、提高工作滿意度及協助員工締造高價值成果(即提升工作效率)。

此外,本研究也探討我們經常談論的做法和成效評估指標:

- **軟體推送成效**:團隊能以安全、快速且有效率的方式,變更自家技術系統。
- **運作成效**:服務提供可靠的使用體驗。



# 重要研究結果

## 營造健康的文化

文化是培養技術能力、提升技術成效、實現機構績效目標，以及協助員工邁向成功的重要基礎。擁有創造型文化的團隊，機構績效高出 30%。

## 經營事業時須考量使用者

秉持以使用者為核心的機構，能提升本研究所探討的各項技術、程序和文化能力，幫助團隊隨時視需要快速部署項目。如果不將使用者納入考量，恐怕就無法做到這點。重視使用者的團隊在機構績效上高出 40%。

## 加快程式碼審查速度可提升軟體推送成效

加速審查程式碼，是最能提升軟體推送成效的方法之一。程式碼審查速度更快的團隊，在軟體推送成效上高出 50%。

## 優質的說明文件可強化技術能力

優質的說明文件可提升技術能力，進而強化機構績效。以主幹式開發為例，與劣質說明文件相比，優質說明文件對機構績效的正面影響高出 12.8 倍。

## 以雲端技術提升基礎架構彈性

雲端運算的優勢在於改善基礎架構彈性。舉例來說，相較於不使用雲端，使用公有雲可增加 22% 基礎架構彈性。與無彈性基礎架構相比，這些更具彈性的基礎架構將提升 30% 機構績效。要充分發揮雲端的價值，關鍵就是善用「基礎架構彈性」這項雲端獨有的特性和能力。

## 在加快推送速度、提升運作成效和重視使用者之間取得平衡

機構若想實現最佳績效，需要同時具備強大的軟體推送和運作能力。在加快推送速度、提升運作成效及重視使用者之間取得平衡，能締造最佳機構績效，同時兼顧員工身心健康。

## 平分工作

女性、弱勢族群或選擇自述性別的員工較容易出現嚴重工作倦怠，這種現象的成因可能涉及多種系統性和環境因素。一如預期，我們從作答者的回答中發現，從事的重複性工作越多，越有可能出現嚴重工作倦怠，而這類工作通常由弱勢員工負責。與非弱勢族群的作答者相比，屬於弱勢族群的作答者工作倦怠程度高出 24%，而且負責的重複性工作多出 29%。與男性作答者相比，女性或自述性別的作答者負責的重複性工作多出 40%。

# 根據業務情境運用 DORA 洞察結果

團隊若秉持精益求精的思維與做法，將更有機會創造最多優勢<sup>10</sup>。DORA 可支援您的改善計畫，

但您仍需考慮自家團隊和使用者的實際情況，才能充分發揮這項研究的效用。舉例來說，前文提到團隊若加快程式碼審查速度，可提升 50% 軟體推送成效。然而，如果程式碼審查速度已夠快，但系統其他部分的速度受限，則軟體推送成效不太可能提升。若想根據實際情況運用這項研究提升成效，從業人員可以互相交流目前的工作模式。這些對談有助於促進合作、瞭解每位參與者的動機，並對彼此所處的狀況更具同理心。

改善永遠沒有盡頭，您必須找出並解決系統瓶頸，然後重複這個過程。重點是追蹤應用程式的長期成效變化，而不是與其他應用程式、機構或產業比較。

## 指標與評估數據

團隊可以透過指標和資訊主頁數據控管進度及修正方向。

不過，在致力提升機構成效、團隊績效和身心健康的過程中，從業人員和主管不能只看重評估數據，就像業務目標不應只是推送軟體。

一味追求成效指標，可能導致從業人員和主管採取無實質效果的措施。更有效的做法是培養能力並持續學習。最積極學習的團隊，才能出現飛躍性的成長。

## 改善成效無法僅憑一己之力

我們可以從彼此的經驗中學習：歡迎瀏覽 DORA 社群網站 (<https://dora.community>)，在這個卓越的論壇中分享及探索各種改善計畫。

<sup>10</sup>《加速發展：2022 年開發運作現狀報告》。

<https://dora.dev/research/2022/dora-report/2022-dora-accelerate-state-of-devops-report.pdf#page=7>



# 概念與評估指標

本章將說明 DORA 嘗試評估的各項概念<sup>1</sup>，也就是我們建立模型和編寫這份報告的基礎。以下各節將介紹這份報告的構成要素，身為作者，我們有必要清楚瞭解這些概念，並維持用語的一致性。接下來幾頁的表格將闡述這些概念和用語，為讀者和作者奠定共同的理解基礎。

這份報告中有不少概念涉及多個面向，因此我們通常會透過多個指標加以記錄和呈現。我們評估是否成功呈現這些概念的方法之一，是探索性因素分析和驗證性因素分析。如要進一步瞭解相關流程，請參閱「[研究方法](#)」一節。我們會衡量評估方法，然後給予 0 到 10 分的評分，0 分代表完全無法呈現概念，10 分代表充分呈現概念。這有助於我們以更一致的方式描述這些概念的作用，並更容易比較各年度的資料。

探討各項概念時，我們也會一併提供以下資訊：

- 圖示：幫助傳達涵義，以及讓使用者在參考本章時，能快速找到所需概念。
- 平均值：特定概念在樣本中獲得的平均分數。
- 四分位距的界線值，也就是第 25 和第 75 百分位數。由於 50% 的資料會落在這兩個界線值之間，因此我們能看出回應的分散情形。
- 中位數：一組資料的中間值。如果中位數與平均值差異過大，表示資料分布有偏移情形。
- 概念說明和評估方式。

<sup>1</sup> 分析所使用的問卷調查問題已發布於 <https://dora.dev>

# 關鍵成果

關鍵成果是我們認為人員、團隊或機構極力追求的目標(例如機構績效),或希望避免的情形(例如工作倦怠)。因此,我們認為評估指標是人員衡量自身、所屬團隊和機構的重要方式。

 <h3>機構績效</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.3</td><td>5-8</td><td>6.3</td></tr></tbody></table> <p>高績效機構擁有較多顧客和利潤,且主要產品或服務的相對市占率較高。</p>	平均值	四分位距	中位數	6.3	5-8	6.3	 <h3>團隊績效</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.6</td><td>6.6-9</td><td>8</td></tr></tbody></table> <p>高績效團隊可靈活因應變化、信任彼此、工作效率高、勇於創新且合作程度高。</p>	平均值	四分位距	中位數	7.6	6.6-9	8	 <h3>軟體推送成效</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.3</td><td>5.1-7.8</td><td>6.4</td></tr></tbody></table> <p>我們透過以下四個指標評估軟體推送速度和穩定性:</p> <ul style="list-style-type: none"><li>• 部署頻率</li><li>• 變更的前置時間</li><li>• 變更失敗率</li><li>• 失敗部署作業復原時間</li></ul>	平均值	四分位距	中位數	6.3	5.1-7.8	6.4
平均值	四分位距	中位數																		
6.3	5-8	6.3																		
平均值	四分位距	中位數																		
7.6	6.6-9	8																		
平均值	四分位距	中位數																		
6.3	5.1-7.8	6.4																		
 <h3>運作成效</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.2</td><td>5-7.5</td><td>6.3</td></tr></tbody></table> <p>服務能滿足使用者期望的程度,評估指標包括可用性和效能。</p>	平均值	四分位距	中位數	6.2	5-7.5	6.3	 <h3>可靠性目標</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7</td><td>5-7.5</td><td>7.5</td></tr></tbody></table> <p>服務達到聲稱目標的程度,評估指標包括可用性、效能和正確性。</p>	平均值	四分位距	中位數	7	5-7.5	7.5							
平均值	四分位距	中位數																		
6.2	5-7.5	6.3																		
平均值	四分位距	中位數																		
7	5-7.5	7.5																		

身心健康包含工作倦怠、工作效率和工作滿意度

 <h3>工作倦怠</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>4.1</td><td>2-6</td><td>4</td></tr></tbody></table> <p>除了指工作造成的生理和心理影響外,也包括人員如何看待工作的價值和意義。工作倦怠會引發負面思考<sup>2</sup>。</p>	平均值	四分位距	中位數	4.1	2-6	4	 <h3>工作效率</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.5</td><td>6.7-8.8</td><td>7.9</td></tr></tbody></table> <p>工作效率高的人員能夠以高效率執行符合自身技能的工作,並創造價值。</p>	平均值	四分位距	中位數	7.5	6.7-8.8	7.9	 <h3>工作滿意度</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.08</td><td>5.7-7.1</td><td>7.1</td></tr></tbody></table> <p>一個單項問題,要求作答者考慮所有因素並對整體工作感受評分<sup>3</sup>。</p>	平均值	四分位距	中位數	6.08	5.7-7.1	7.1
平均值	四分位距	中位數																		
4.1	2-6	4																		
平均值	四分位距	中位數																		
7.5	6.7-8.8	7.9																		
平均值	四分位距	中位數																		
6.08	5.7-7.1	7.1																		

<sup>2</sup>Maslach C 和 Leiter MP,《Understanding the burnout experience: recent research and its implications for psychiatry》(瞭解工作倦怠:近期研究及工作倦怠對精神病學的影響),World Psychiatry,2016年6月,15(2):103-11 doi:10.1002/wps.20311 PMID:27265691;PMCID:PMC4911781。

<sup>3</sup>Warr, P., Cook, J. 和 Wall, T.《Scales for the measurement of some work attitudes and aspects of psychological well-being》(評估部分工作態度與心理健康各面向的量表), Journal of Occupational Psychology, 52(2), 1979年,129-148 <https://doi.org/10.1111/j.2044-8325.1979.tb00448.x>

# 程序與技術能力

程序與技術能力代表團隊或機構採取的行動或做法，或是所處的狀態，即團隊處理的事務或作風。

 <h3>人工智能技術使用度</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>3.3</td><td>0.3-6.3</td><td>2.4</td></tr></tbody></table> <p>執行各種技術工作時，仰賴人工智能技術的程度。</p>	平均值	四分位距	中位數	3.3	0.3-6.3	2.4	 <h3>鬆耦合架構</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.4</td><td>4.7-8.3</td><td>6.7</td></tr></tbody></table> <p>軟體可獨立編寫、測試及部署。</p>	平均值	四分位距	中位數	6.4	4.7-8.3	6.7	 <h3>持續整合</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.9</td><td>5-8.9</td><td>7.8</td></tr></tbody></table> <p>自動建構並測試軟體變更的做法。</p>	平均值	四分位距	中位數	6.9	5-8.9	7.8
平均值	四分位距	中位數																		
3.3	0.3-6.3	2.4																		
平均值	四分位距	中位數																		
6.4	4.7-8.3	6.7																		
平均值	四分位距	中位數																		
6.9	5-8.9	7.8																		
 <h3>持續推送軟體更新</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.0</td><td>5.7-8.7</td><td>7.3</td></tr></tbody></table> <p>快速、安全且持續將所有變更類型 (包括新功能、設定變更、錯誤修正項目和實驗功能) 推送到正式環境或使用者裝置的能力<sup>40</sup>。</p>	平均值	四分位距	中位數	7.0	5.7-8.7	7.3	 <h3>程式碼審查速度</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.5</td><td>6-8</td><td>6</td></tr></tbody></table> <p>一個單項問題，請作答者評估從提取要求到程式碼變更審查所需要的時間。</p>	平均值	四分位距	中位數	6.5	6-8	6	 <h3>說明文件</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>5.8</td><td>3.8-7.9</td><td>6.25</td></tr></tbody></table> <p>機構人員編寫和日常使用的業務文件品質。</p>	平均值	四分位距	中位數	5.8	3.8-7.9	6.25
平均值	四分位距	中位數																		
7.0	5.7-8.7	7.3																		
平均值	四分位距	中位數																		
6.5	6-8	6																		
平均值	四分位距	中位數																		
5.8	3.8-7.9	6.25																		
 <h3>可靠性做法</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>5.9</td><td>3.9-8.3</td><td>6.1</td></tr></tbody></table> <p>團隊為了提升服務運作成效所採取的行動和做法。</p>	平均值	四分位距	中位數	5.9	3.9-8.3	6.1	 <h3>主幹式開發</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>5.6</td><td>3.9-7.8</td><td>5.6</td></tr></tbody></table> <p>這項開發做法意指頻繁進行小型變更，並定期併入版本管控系統的主要程式碼分支版本。</p>	平均值	四分位距	中位數	5.6	3.9-7.8	5.6	 <h3>彈性基礎架構</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.6</td><td>5-8.3</td><td>7.3</td></tr></tbody></table> <p>彈性、可擴充、方便存取且可評估的基礎架構<sup>50</sup>。</p>	平均值	四分位距	中位數	6.6	5-8.3	7.3
平均值	四分位距	中位數																		
5.9	3.9-8.3	6.1																		
平均值	四分位距	中位數																		
5.6	3.9-7.8	5.6																		
平均值	四分位距	中位數																		
6.6	5-8.3	7.3																		

<sup>40</sup>《What is Continuous Delivery》(什麼是持續推送軟體更新) <https://continuousdelivery.com/>

<sup>50</sup> 2018 年美國國家標準暨技術研究院 (NIST),《The NIST Definition of Cloud Computing》(NIST 雲端運算定義)。詳情請參閱 <https://csrc.nist.gov/pubs/sp/800/145/final>

# 文化層面

定義文化並不容易，但或許我們能稱之為職場的常規(如彈性)、主流趨勢(例如以使用者為核心)，以及氛圍(如機構穩定性)。

 <h3>工作分配</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>5.8</td><td>3.8-7.9</td><td>5.8</td></tr></tbody></table> <p>公平分派工作給團隊成員的正式程序。</p>	平均值	四分位距	中位數	5.8	3.8-7.9	5.8	 <h3>工作彈性</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.7</td><td>6.6-8.9</td><td>8.3</td></tr></tbody></table> <p>員工處理工作的方式、地點和時間<sup>6</sup>。</p>	平均值	四分位距	中位數	7.7	6.6-8.9	8.3	 <h3>就業保障</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>5.9</td><td>3.3-8.3</td><td>6.7</td></tr></tbody></table> <p>一個單項評估問題，要求作答者分享他們多常擔心工作穩定性。分數越高，擔憂程度越低。</p>	平均值	四分位距	中位數	5.9	3.3-8.3	6.7
平均值	四分位距	中位數																		
5.8	3.8-7.9	5.8																		
平均值	四分位距	中位數																		
7.7	6.6-8.9	8.3																		
平均值	四分位距	中位數																		
5.9	3.3-8.3	6.7																		
 <h3>機構穩定性</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.2</td><td>6.7-8.3</td><td>8.3</td></tr></tbody></table> <p>一個單項評估問題，要求作答者分享工作環境對員工而言是否穩定。</p>	平均值	四分位距	中位數	7.2	6.7-8.3	8.3	 <h3>知識共享</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>6.4</td><td>5.0-8.3</td><td>6.7</td></tr></tbody></table> <p>想法和資訊在整個機構內的流通情形。在知識共享程度高的機構中，團隊成員只需回答問題一次，大家就能取用資訊，不必費時等待答案<sup>7</sup>。</p>	平均值	四分位距	中位數	6.4	5.0-8.3	6.7	 <h3>以使用者為主</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.8</td><td>5.6-8.3</td><td>7.8</td></tr></tbody></table> <p>瞭解使用者的需求和目標並反映在產品與服務中，使其更臻完善<sup>8</sup>。</p>	平均值	四分位距	中位數	7.8	5.6-8.3	7.8
平均值	四分位距	中位數																		
7.2	6.7-8.3	8.3																		
平均值	四分位距	中位數																		
6.4	5.0-8.3	6.7																		
平均值	四分位距	中位數																		
7.8	5.6-8.3	7.8																		
 <h3>Westrum 機構文化</h3> <table><thead><tr><th>平均值</th><th>四分位距</th><th>中位數</th></tr></thead><tbody><tr><td>7.3</td><td>6.1-8.6</td><td>7.8</td></tr></tbody></table> <p>機構因應問題和機會的方式，包含三種文化類型：創造型、官僚型和病態文化<sup>9</sup>。</p>	平均值	四分位距	中位數	7.3	6.1-8.6	7.8	<p><sup>6</sup>Shifrin、Nicole V. 和 Jesse S. Michel, «Flexible work arrangements and employee health: A meta-analytic review»(彈性工作模式與員工健康：綜合分析評論)，Work &amp; Stress 36, no. 1, 2022 年, 60-85</p> <p><sup>7</sup>《2022 Developer Survey》(2022 年開發人員問卷調查) <a href="https://survey.stackoverflow.co/2022#overview">https://survey.stackoverflow.co/2022#overview</a></p> <p><sup>8</sup>Kersten, Mik, «Project to Product: How to survive and thrive in the age of digital disruption with the flow framework»(從專案到產品：如何在數位變革時代下運用流程架構克服挑戰並蓬勃發展)，IT Revolution 出版，2018 年, 54。<a href="https://itrevolution.com/product/project-to-product/">https://itrevolution.com/product/project-to-product/</a></p> <p><sup>9</sup>Westrum R, «A typology of organizational cultures»(機構文化分類學)，BMJ Quality &amp; Safety, 2004 年。<a href="https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1765804/">https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1765804/</a></p>													
平均值	四分位距	中位數																		
7.3	6.1-8.6	7.8																		

# 相較之下，您的團隊表現如何？

古德哈特定律指出，一旦指標成為目標，就不再是好的指標<sup>1</sup>。

## 重點摘要

改善成效的第一步是設定基準，以利評估應用程式目前的軟體推送和運作成效，以及使用者至上的程度。這些指標能幫助團隊評估當前的表現，並有效監控長期變化。

不過，團隊不應將這些指標視為改善方法。設定基準的目的，是評估團隊在員工多樣性、程序和技術能力方面的相對強弱，以找出可能阻礙進步的因素<sup>2</sup>。接著，團隊需要時間和空間來調整、實驗及重新評估。團隊可藉由重複這段程序，培養精益求精的心態和行動力。

進行比較時，請避免以下陷阱和其他錯誤：

- **不對等比較：**僅根據這些集群來比較應用程式，無法得出準確結果。這麼做會忽略每個應用程式的開發運作背景，而有時這些背景資訊是擬定改善目標的關鍵。
- **將指標設為目標：**若忽視古德哈特定律，喊出「每個應用程式都必須在年底達到卓越成效」等空泛的目標，團隊可能會試圖以投機取巧的方式達成指標。
- **僅憑單一指標判定整體成效：**評估複雜系統不能抱持「一個指標就夠」的心態，結合多項指標才能更深入瞭解開發運作情形<sup>3</sup>。
- **指標評估範圍狹窄：**團隊/機構傾向選擇最容易評估的指標，而不是最有助益的指標。
- **將產業做為拒絕改進的擋箭牌：**舉例來說，如果團隊屬於受嚴格管制的產業，可能會以法規為由，拒絕改變現狀。

<sup>1</sup>Strathern, Marilyn (1997 年)，《Improving ratings': audit in the British University system》(改進評等：英國大學系統中的稽核)，European Review, John Wiley & Sons, 5 (3):305–321, doi:10.1002/(SICI)1234-981X(199707)5:3<305::AID-EURO184>3.0.CO;2-4; S2CID 145644958。

<sup>2</sup>如要瞭解更多資訊，請參閱本報告和「Capability catalog」(能力目錄) 中所列的資源。網址為：<https://dora.dev/devops-capabilities/>。

<sup>3</sup>Forsgren, N.、Storey, M-A. 等人，《The SPACE of Developer Productivity: There's more to it than you think》(運用 SPACE 架構提升開發人員工作效率：考量層面比想像中更多)，2021 年。<https://queue.acm.org/detail.cfm?id=3454124>

如要透過我們的研究結果和建議，進一步瞭解如何採取持續改善的思維和做法，請參閱「How to transform」(轉型方法) 一文。網址為：[dora.dev/devops-capabilities/cultural/\\_devops-culture-transform/](https://dora.dev/devops-capabilities/cultural/_devops-culture-transform/)

# 前言

我們每年會執行一或數次集群分析，以找出應用程式共同的開發運作趨勢。您可以透過這些分析數據瞭解自身的相對表現，但不必執著在這些比較結果。想瞭解改善成果，最佳做法是追蹤應用程式的長期成效變化，而不是與其他應用程式比較，因為每個應用程式的開發運作情形不盡相同。

建構軟體的目的是服務使用者，他們才是服務可靠性和實用性的最終評斷者。重視使用者需求的團隊擅於打造符合期望的軟體，若同時注重軟體推送和運作成效，還能發展出合適的開發運作做法。

重視使用者需求的團隊擅於  
打造符合期望的軟體，還能  
發展出合適的開發運作做  
法。

# 研究結果

## 軟體推送成效

我們使用以下指標來評估軟體推送成效：

- **變更前置時間**：從建立修訂版本到部署變更所需時間。
- **部署頻率**：將變更推送至正式環境的頻率。
- **變更失敗率**：部署作業失敗且需立即介入處理的頻率。
- **失敗部署作業復原時間**：部署作業失敗時所需的復原時間。

提升以上四項指標成效的常見做法，是減少應用程式的變更批量<sup>4</sup>。小規模變更較容易觀測、檢驗及推送，一旦出現失敗情形，也較容易復原。團隊應盡可能縮小變更規模，讓推送流程保持快速穩定，這有助於提高變更的速度和穩定性。

<sup>4</sup>一項功能通常會拆成多項變更項目，並單獨推送。我們的軟體推送成效評估指標，即是衡量團隊對應用程式或服務所做的變更。

今年，我們改良了軟體推送成效的評估方式，相關異動請參見「[附錄](#)」的「改良軟體推送成效評估方式」一節。

以下是今年問卷調查作答者在軟體推送成效方面的表現：

成效等級	部署頻率	變更前置時間	變更失敗率	失敗部署作業復原時間	作答者百分比
卓越	隨需部署	不到一天	5%	不到一小時	18%
高	介於每天一次到每週一次	介於一天到一週之間	10%	不到一天	31%
中	介於每週一次到每月一次	介於一週到一個月之間	15%	介於一天到一週之間	33%
低	介於每週一次到每月一次	介於一週到一個月之間	64%	介於一到六個月之間	17%



## 運作成效

我們詢問作答者自家服務遭遇下列情形的頻率，藉此評估運作成效：

- 使用者對系統可靠性表示不滿意。
- 服務無法使用、運作速度不如預期或未正確運作。

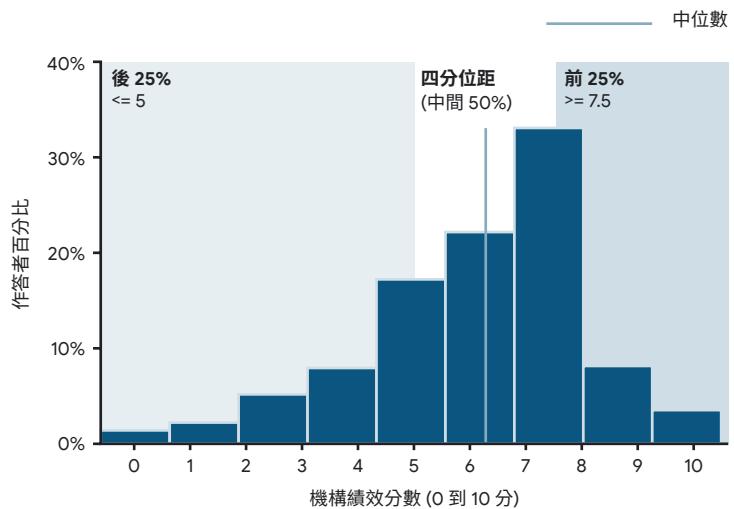
如要進一步瞭解運作成效如何影響機構績效，請參見「[第 5 章：可靠性帶來更高成效](#)」。

## 重視使用者

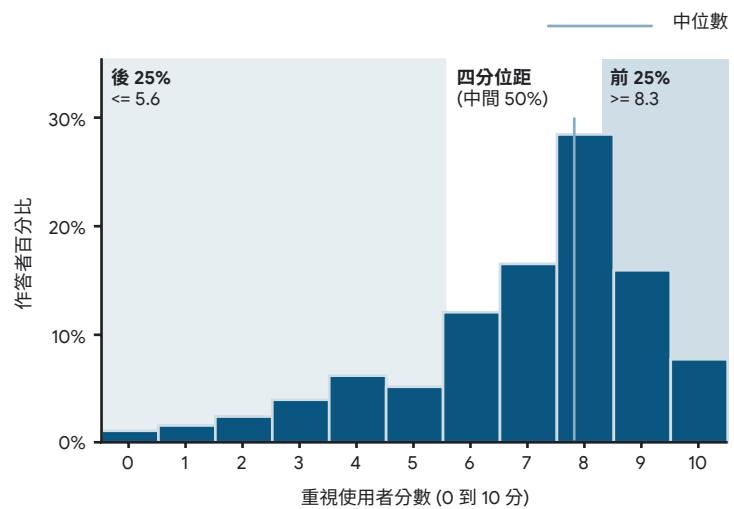
重視使用者的應用程式或服務，在建構時就會考量使用者需求。要打造這類產品，團隊必須充分瞭解使用者需求，並將這些需求融入產品發展藍圖。我們詢問作答者對下列敘述的認同度，藉此評估他們是否以使用者為主：

- 團隊清楚瞭解使用者想實現什麼目標。
- 團隊的成功與否，取決於他們為應用程式使用者和所屬機構創造的價值。
- 根據使用者信號，不斷重新審視產品規格（例如需求規劃）及調整優先度。

下圖為問卷調查作答者今年在運作成效方面的表現：



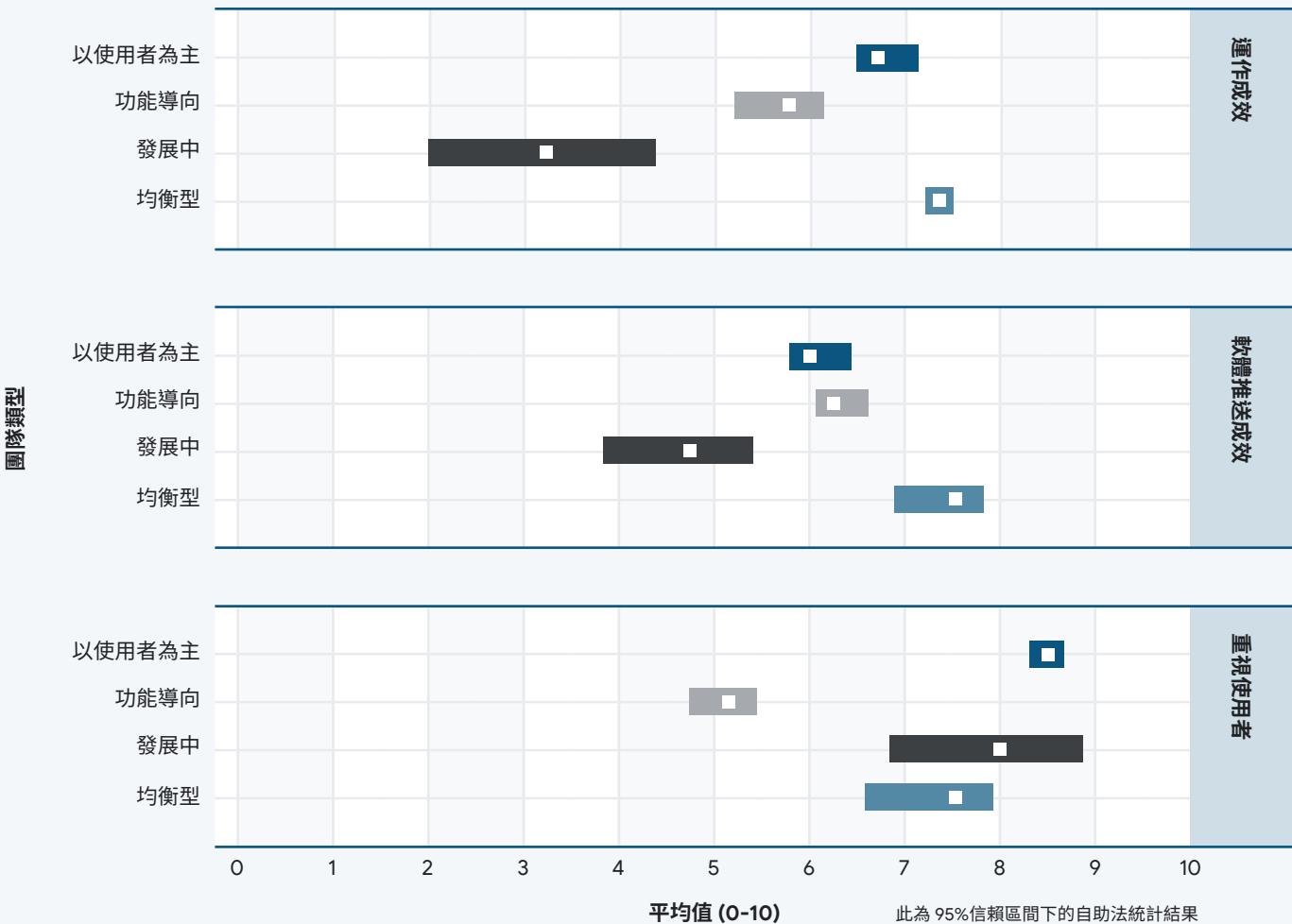
下圖為問卷調查作答者今年在重視使用者方面的表現：



## 團隊分群類型

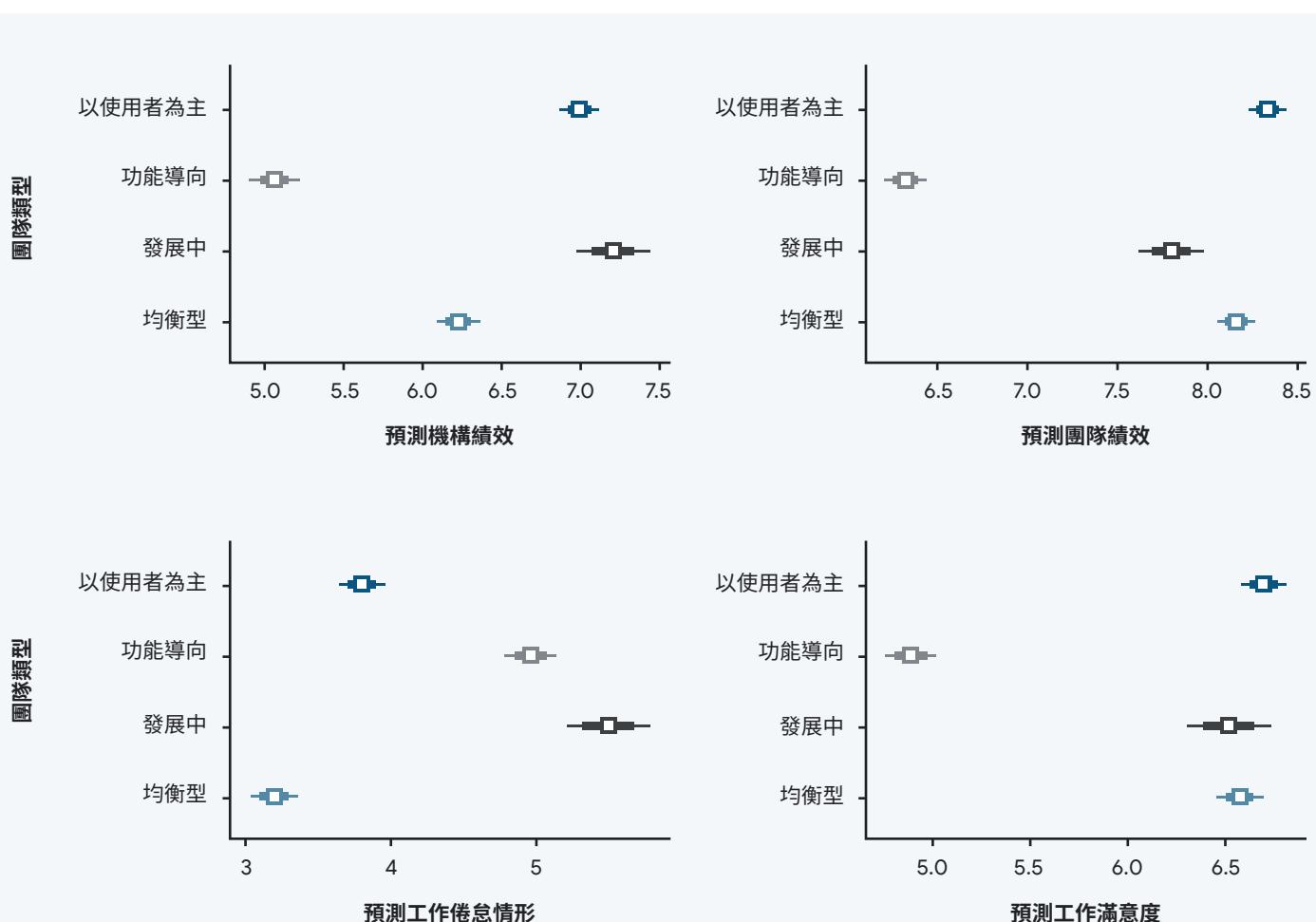
以軟體推送成效、運作成效和重視使用者為單位進行比較後，可將團隊分成四個類型。如同所有評估指標，這些類型反映的是應用程式或服務層級的表現。

我們將這些團隊類型稱為「以使用者為主」、「功能導向」、「發展中」和「均衡型」。



機構或團隊可以根據此處探討的成效指標來調整作業模式，以提升機構、團隊績效及團隊成員的身心健康。

以下圖表呈現各團隊類型的預測結果。



\*每個點代表團隊類型平均值的點估計。粗線代表 66% 模擬資料所在的區間，細線代表 89% 模擬資料所在的區間。

每個團隊類型的特徵各有不同，在所有作答者中都占有一定的比例，而且預測成果不盡相同。您的團隊不一定完全符合某個類型，我們也不建議長期維持特定類型。



# 這些分析結果的意義為何？

## » 以使用者為主的團隊

這類團隊最重視使用者需求，如果同時具備強大的軟體推送和運作能力，將獲得最高機構績效。然而，這類團隊的工作倦怠程度略高於均衡型團隊。對這類團隊而言，提升軟體推送和/或運作成效可能是減少工作倦怠的最佳做法。

## » 功能導向團隊

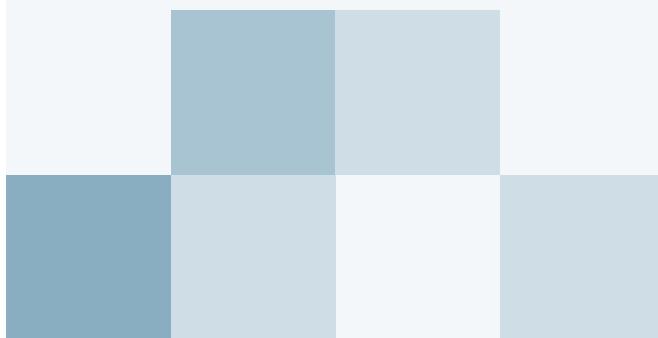
這類團隊極度強調推送功能，因而較無法顧及使用者需求，如圖表所示，其使用者重視程度和運作成效的數據偏低。這類團隊的工作倦怠程度最高，工作滿意度、團隊和機構績效最低。員工注重的是提供價值，而不只是推送功能。對功能導向團隊而言，推送功能時如果能多考量使用者需求，將有助於創造更多價值。

## » 發展中團隊

這類團隊相當注重應用程式使用者的需求，因此可實現良好機構績效。不過，他們仍在摸索產品的市場契合度，或發展技術能力。發展中團隊常見於小規模機構，其軟體推送和運作成效都偏低。此外，與均衡型和以使用者為主的團隊相比，發展中團隊在開發應用程式時，員工工作倦怠程度更高。背後的原因可能是作業程序過於繁瑣，或工作相當費時費力，也就是說只要將這些作業自動化，即可提升軟體推送和運作成效。

## » 均衡型團隊

這類團隊以均衡且永續的做法，運用技術達成良好的機構成效和團隊績效、工作滿意度佳，且工作倦怠程度最低。這些團隊已妥善調整自身能力，因此在以上三項評估指標都展現亮眼成績。若能持續提升使用者重視程度，機構成效可望再創佳績。



# 「以使用者為主」 對機構績效的影響

## 重點摘要

機構嘗試提升開發人員工作效率和組織成效時，可能會陷入「追隨最新技術和管理趨勢」的迷思。我們的研究顯示，採取重視使用者的做法來建構應用程式和服務，是最有望提高整體機構績效的因素之一。想提升績效，您需要深入瞭解使用者需求、採納他們的意見並反覆調整。

## 前言

業界推行「開發運作」的初衷，是鼓勵開發團隊和運作團隊相互合作，為使用者提供更多價值。隨著這兩個團隊彼此協調並取得初步成功，其他部門也紛紛導入開發運作的理念和能力。如今高績效的技術導向機構，都認同所有團隊應先達成共識，才有可能實現機構目標。

我們從以下三大特徵研究團隊重視使用者的程度：

- 團隊對使用者需求的瞭解程度。
- 就滿足使用者需求而言，團隊達成何種程度的共識。
- 如何根據使用者意見回饋來安排業務優先順序。

高度重視使用者的團  
隊，可提升  
**40%**  
機構績效

# 研究結果

我們發現，以使用者為核心的軟體開發做法，能確實提升績效。以使用者為先的機構能獲得莫大效益。使用者意見有助於團隊安排專案優先順序，並打造符合使用者需求的產品和服務，進而創造更好的使用者體驗、提高使用者滿意度並帶動收益成長。

## 以使用者為主帶來的影響

機構績效		大幅提升
團隊績效		大幅提升
軟體推送成效		小幅提升
運作成效		大幅提升
主幹式開發		大幅提升
可靠性做法		大幅提升
持續整合		大幅提升
持續推送軟體更新		大幅提升
鬆耦合架構		大幅提升
工作倦怠		小幅降低*
工作滿意度		大幅提升
工作效率		大幅提升

\*工作倦怠減少是好事！

# 這些分析結果的意義為何？

重視使用者需求是強化機構整體績效的有利因素。您需要採取適當的獎勵措施、協調機制和工作方式，才能確實做到以使用者為主。秉持以使用者為核心的機構，能提升本研究所探討的各項技術、程序和文化能力。

現在，讓我們深入探索這些分析結果對機構內各種團隊的影響：

### 產品開發和推送團隊

秉持重視使用者的原則，可確保產品開發和推送團隊打造出適合使用者的產品，甚至能在開發過程中融入永續做法。均衡型團隊正是如此：他們在推送、運作和組織成效上都十分優異，也非常注重使用者需求。在這類團隊中，每位成員都清楚瞭解使用者的需求，而且能根據使用者意見調整計畫。

相較之下，功能導向團隊無法達成最佳機構績效。這類團隊過度看重推送成效，反而有損機構的績效和員工身心健康。

### 運作團隊

注重運作成效的團隊，可能會盡量提高 CPU 用量等系統指標的表現。但如果不懂得使用者對服務的期望，他們可能仍會發現使用者經常回報服務效能低落。網站可靠性工程 (SRE) 做法可幫助運作團隊培養以使用者為主的思維。這套做法包括找出使用者關注的服務水準指標，以及設定服務等級目標，藉此滿足一般使用者需求。

## » 平台工程團隊

建構平台時，有的平台工程團隊可能只抱持「只要架設完畢，使用者自然就會來用」的心態。但更有效的方式，是將開發人員視為平台使用者。這項轉變能否成功，有賴平台工程團隊深入瞭解開發人員目前的工作方式，這樣才能找出及排除工作上的阻礙。團隊可以透過軟體推送和運作成效評估指標，監控平台是否幫助開發人員取得更好的成果。

## » 主管

如要培養重視使用者的文化，主管可以建立獎勵制度來表揚團隊為使用者創造價值。如果沒有這些制度，團隊可能會覺得主管只關心功能推送數量，或服務中斷減少次數。DORA 研究了轉型領導力所扮演的角色<sup>1</sup>，並提供幾項建議給正在提升這項能力的主管參考。詳請請參閱「DevOps Capabilities: Transformational Leadership」(開發運作能力：轉型領導力)一文。網址為：<https://dora.dev/devops-capabilities/cultural/transformational-leadership>

秉持重視使用者的原則，可提升

20%

工作滿意度

## 入門資源

發展更重視使用者的能力，是通往成功的關鍵。2018 年的 Google 研究結果指出，精實產品管理能力有利於提升軟體推送成效和機構績效，而 2023 年的研究結果再度印證這個觀點<sup>2</sup>。

培養以使用者為主的能力可提升團隊績效，這些能力包括採納顧客意見回饋<sup>3</sup>、掌握價值流程中的工作<sup>4</sup>、將工作劃分成較小的批次來進行<sup>5</sup>，以及進行團隊實驗<sup>6</sup>。

<sup>1</sup>《2017 年開發運作現狀報告》，<https://dora.dev/publications/pdf/state-of-devops-2017.pdf>, 12-19

<sup>2</sup>《加速發展：2018 年開發運作現狀報告：新經濟策略》，<https://dora.dev/publications/pdf/state-of-devops-2018.pdf>, 49-51

<sup>3</sup>「Customer feedback」(顧客意見回饋)。<https://dora.dev/devops-capabilities/process/customer-feedback/>

<sup>4</sup>「Visibility of work in the value stream」(掌握價值流程中的工作)。<https://dora.dev/devops-capabilities/process/work-visibility-in-value-stream/>

<sup>5</sup>「Working in small batches」(將工作劃分成較小的批次來進行)。<https://dora.dev/devops-capabilities/process/working-in-small-batches/>

<sup>6</sup>「Team experimentation」(進行團隊實驗)。<https://dora.dev/devops-capabilities/process/team-experimentation/>

# 技術能力對成效的助益

## 重點摘要

投入資源和心力來推動持續整合、導入鬆耦合架構及加快程式碼審查速度，均可帶來許多效益，例如提升機構和團隊績效，以及軟體推送和運作成效。這些做法不僅沒有負面影響，而且通常對經營應用程式或服務的員工身心健康有助益。

## 前言

在「內容提要」中，我們提到這份報告探討的技術能力，以及這些能力對各種成效和身心健康指標的影響。DORA 的核心目標，一向是探究及量化各種程序和技術能力對成效的影響程度。

今年，我們調查了下列技術能力對成效的影響：

- 人工智慧
- 主幹式開發
- 鬆耦合架構
- 持續整合
- 快速程式碼審查

我們也分析這些能力對以下成效評估指標的影響：

- 團隊績效
- 機構績效
- 軟體推送成效
- 運作成效

此外，我們也檢驗這些能力與下列因素之間的關聯性，以確認這些能力對員工的影響：

- 工作倦怠
- 工作效率
- 工作滿意度

# 研究結果

我們研究的技術能力和程序各有不同，但整體而言，這些能力和程序都對主要成效指標帶來正面影響。

技術能力和程序	對團隊績效的影響	對機構績效的影響	對軟體推送成效的影響	對運作成效的影響
AI	 無影響	 小幅提升	 小幅降低	 大幅降低
持續整合	 小幅提升	 小幅提升	 小幅提升	 無影響
程式碼審查速度	 小幅提升	 無影響	 大幅提升	 大幅提升
鬆耦合架構	 大幅提升	 大幅提升	 小幅提升	 大幅提升
主幹式開發	 小幅提升	 小幅提升	 小幅提升	 小幅降低

獨立運作的團隊或採用鬆耦合架構的團隊，不需仰賴其他團隊便可自行大幅變更系統設計，因此作業速度更快。此外，各領域專家更能即時協助團隊，加快程式碼審查速度，因為他們更瞭解變更帶來的影響。鬆耦合設計可讓團隊自行測試、建構及部署軟體，避免各團隊彼此牽制的情況發生。

不過，即使在鬆耦合架構中進行變更的副作用較小，仍需要確保開發人員的工作不會互相衝突。將工作劃分成小批次進行可降低衝突機率，確保每個修訂版本都已建構軟體並觸發自動化測試，以利開發人員快速獲取意見回饋。

程式碼審查時間較短的團隊，軟體推送成效高出 50%。高效率的程式碼審查程序，有助於改善程式碼、交流知識、建立程式碼共同擁有權和團隊擁有權，以及實現資訊透明化。

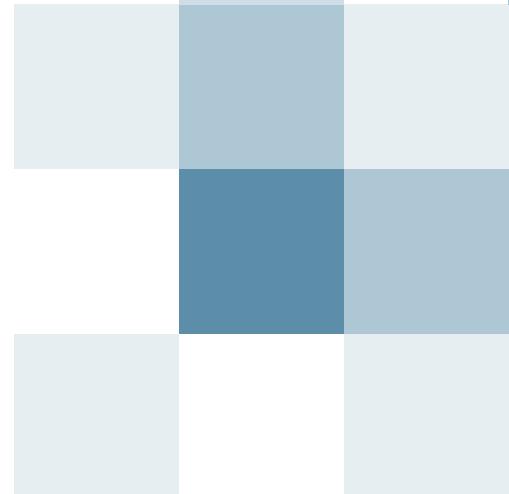
程式碼審查是您難以突破的瓶頸嗎？您可以評估程式碼審查程序，和此程序對變更前置時間的影響，以深入瞭解可改善之處。請思考以下幾個問題：

- 程式碼審查程序是否包含同儕審查？
- 程式碼補全與審查之間間隔多久？
- 每批審查的程式碼數量平均是多少？
- 多少團隊參與審查？
- 參與審查的團隊來自幾個不同的地理位置？
- 團隊系統是否會根據程式碼審查建議，改善程式碼品質自動審核機制？

程式碼補全與審查之間的間隔越長，會對開發人員效率和推送的軟體品質造成負面影響。如果參與審查的團隊眾多且橫跨多個地理位置，將拉長審查程序的時間、降低參與度並增加成本<sup>1</sup>。

在 2022 年的研究中，我們發現技術能力之間相輔相成：加速審查程式碼可提升多項技術能力，包括程式碼維護能力，以及培養學習文化（知識交流）和創造型文化<sup>2</sup>。

鬆耦合團隊的優點之一包括能加快審查程式碼，進而大幅提升軟體推送和運作成效。您可以透過幾種方法提升



程式碼審查效率。如果待審程式碼的影響範圍只涉及團隊使用的架構，審查者可更深入瞭解該程式碼對系統的影響。程式碼審查規模越小，審查者越容易瞭解變更帶來的影響。將工作分成數個小批次執行，可提高意見回饋循環成效、審查效率和團隊專注力<sup>3</sup>。您也可以採取結對程式設計，這個開發方法可以縮短程式碼審查時間，且適用於各種架構和整合做法<sup>4</sup>。

這些能力和程序不但不會對員工身心健康造成負面影響，其中大部分甚至有利於員工的身心健康。

<sup>1</sup>「Investigating the effectiveness of peer code review in distributed software development based on objective and subjective data」（根據客觀與主觀資料，研究分散式軟體開發作業中的同儕程式碼審查效率）。<https://jserd.springeropen.com/articles/10.1186/s40411-018-0058-0>

<sup>2</sup>「Expectations, Outcomes, and Challenges of Modern Code Review」（現代程式碼審查的預期、成果和挑戰）。<https://dl.acm.org/doi/10.5555/2486788.2486882>

<sup>3</sup>「Working in small batches」（將工作分成數個小批次來執行）。<https://dora.dev/devops-capabilities/process/working-in-small-batches/>

<sup>4</sup>「On Pair Programming」（認識結對程式設計）。<https://martinfowler.com/articles/on-pair-programming.html>

技術能力和程序	對工作倦怠的影響*	對工作滿意度的影響	對工作效率的影響			
AI		小幅降低		小幅提升		小幅提升
持續整合		無影響		小幅提升		無影響
程式碼審查速度		大幅降低		小幅提升		小幅提升
鬆耦合架構		大幅降低		大幅提升		大幅提升
主幹式開發		大幅提升		無影響		無影響

\*您可能會發現「工作倦怠」部分的色彩配置不同，因為工作倦怠減少是好事！

我們發現，鬆耦合架構、持續整合和高效率程式碼審查這三項做法有助於改善機構績效，同時還能維持或促進團隊身心健康。

如果團隊享有更多自主權，能維護及改善系統的可靠性，為使用者提供價值，員工的工作滿意度、團隊績效和軟體推送成效都會更高。團隊能否考量使用者的需求並提升軟體推送成效，關鍵在於架構。無論團隊採用主幹式開發、鬆耦合架構、持續整合、持續推送軟體更新或網站可靠性工程做法，只要從小規模作業著手，並秉持重視使用者的原則，都能大幅改善成效。想提升技術能力，請給予團隊進行實驗和持續改善的機會<sup>5</sup>。



<sup>5</sup>「Team experimentation」(進行團隊實驗)：<https://dora.dev/devops-capabilities/process/team-experimentation>

# 持續推送軟體更新的好處

作者：Dave Farley

持續推送軟體更新 (CD) 的基本原則，是確保軟體隨時處於可發布狀態。要做到這點，團隊必須提高工作品質，這樣才能有效偵測及修正問題，快速輕鬆地將軟體復原到可發布狀態。

為了確保軟體隨時處於可發布的最佳狀態，團隊需要建立快速的意見回饋機制，才能在失敗時迅速復原。

相信正在讀今年報告的您，已經對以下概念很熟悉：**穩定性** (變更失敗率和失敗部署作業復原時間) 指標與品質相關，**處理量** (變更前置時間和部署頻率) 指標則與意見回饋和問題偵測簡便度有關。

如果您採行持續推送軟體更新做法，會在**穩定性與處理量**方面獲得高分；換句話說，如果您的機構在**穩定性與處理量**方面獲得高分，基本上可以斷定您實行的做法包含持續推送軟體更新。

在今年的分析中，我們探討各項技術能力的中介因素，以瞭解這些能力如何提高成效。「持續推送軟體更新」是許多技術能力的重要中介因素，它代表機構有能力依需求快速安全地大量發布各種變更。換句話說，這些技術能力之所以有用，是因為創造出有利於實行持續推送軟體更新做法的環境，正是因為機構實行持續推送軟體更新機制，才能成功運用這些技術能力提升軟體推送成效。

軟體開發領域普遍重視可發布性，這也是為什麼持續推送軟體更新做法特別注重這項標準。可發布性雖然是主觀的衡量標準，卻能明確描述軟體在各種情境中應具備的品質，因此至關重要。舉例來說，開發攸關安全的系統，和編寫蛋糕店使用的軟體時，兩者講求的可發布性嚴謹度可能不同。不過在這兩個例子中，「可發布」都代表開發人員已妥善完成所有必要作業，程式碼的品質和安全性已達到一定標準，隨時能發布給使用者。

因此，致力確保變更項目的可發布性，等同於盡力維持系統在具體情況下可接受的最低品質。

團隊若講求快速取得高品質意見回饋，並據以採取行動，將能提升軟體推送成效。

出乎我意料的是，持續整合 (CI) 及主幹式開發並未對軟體推送成效帶來明顯影響。我個人認為持續整合是相當基本的做法，因此這個結果與我的觀念有點衝突。不過，軟體推送程序中的各個面向都會相互影響，很難分別看待。舉例來說，如果不整合程式碼，如何在處理量獲得高分？如果不檢查，如何確定穩定性高？對我來說，持續整合是解答這些問題的關鍵，可說是影響軟體推送成效的重要中介因素。究竟是我們的解讀結果有問題，還是其中蘊含更深、更重要的涵義？確實值得思考！

技術能力和程序	對軟體推送成效的影響	持續推送軟體更新是否有中介效果*
AI		小幅降低 <input checked="" type="checkbox"/> 否
持續整合		小幅提升 <input checked="" type="checkbox"/> 是，具有完全影響力
程式碼審查速度		大幅提升 <input checked="" type="checkbox"/> 是，僅有部分影響力
鬆耦合架構		小幅提升 <input checked="" type="checkbox"/> 是，僅有部分影響力
主幹式開發		小幅提升 <input checked="" type="checkbox"/> 是，具有完全影響力

\*我們透過測試來評估某些潛在機制或途徑是否具有中介效果。例如我們可能會這樣描述：「在採用持續部署做法（中介因素）的情況下，主幹式開發才有提高軟體推送成效的效果，且有資料支持這個假設」。當中介因素能完全影響結果，我們稱為完全中介；當中介因素只能影響部分結果，則稱為部分中介。

## 機構程序和能力最佳化

我們知道良好的文化帶來成功，但什麼能造就良好的文化？這個問題很有趣，且最常聽到的答案就是：因機構而異！

就從業人員的角度來看，改善日常工作方式有助於營造正向文化，包括共同承擔風險、促進合作及建立心理安全感。例如，定期將變更併入版本管控系統的主分支版本，可以增進知識共享及合作。資安團隊和開發人員以「政策即程式碼」為基礎來合作，可加深團隊之間的信任，並對部署的變更項目更具信心。

從主管的角度來看，良好的文化始於體認及推廣這類文化的重要性。培養轉型領導力<sup>6</sup>有助於營造不互相責難的職場、鼓勵員工積極學習與實驗、加深從業人員的信任感，並讓他們勇於表達己見。工程師的職責是解決複雜問題，不只是回應工作要求。為了做到這點，他們需要清楚掌握業務，並擁有採取行動的自主權。最終，正向文化就能從主管階層向下推行至所有員工。

最理想的結果，是機構上下團結，共同推動文化。

<sup>6</sup>「Transformational leadership」(轉型領導力)：<https://dora.dev/devops-capabilities/cultural/transformational-leadership>

# 人工智慧 (AI)

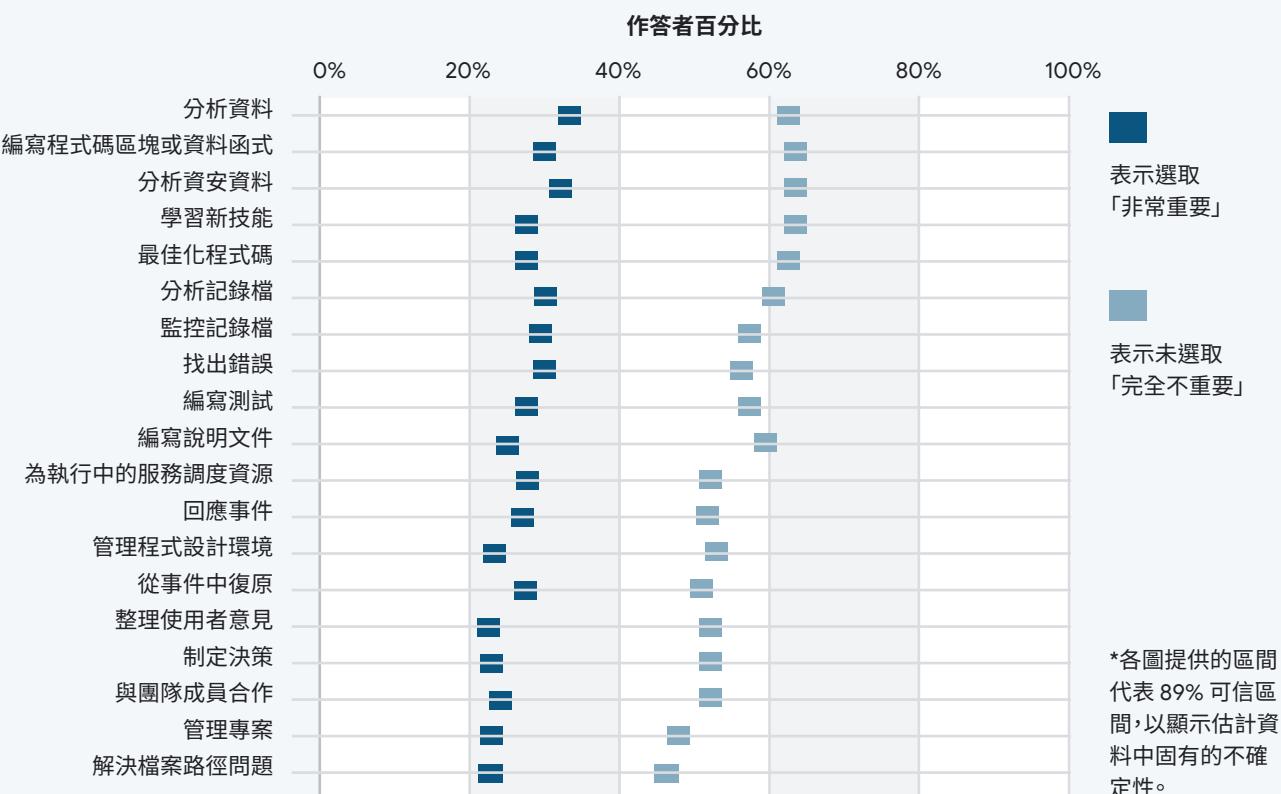
部分技術專家和分析師認為 AI 能提升軟體團隊工作效率，且不會對人員身心健康造成負面影響，但我們的問卷調查證據目前不支持這個觀點。我們的研究數據顯示，AI 僅微幅改善工作倦怠和工作滿意度等人員身心健康指標，但對團隊績效和軟體推送成效等團隊成果沒有明顯作用，甚至造成負面影響。

至於為何結果有好有壞，我們推測原因在於企業仍處於 AI 工具的早期採用階段。此外，部分大型企業可能仍在試用各種 AI 技術輔助工具，尚未決定是否全面採用。不過，大多數作答者至少會在題目列舉的工作中用到 AI，可見很多人還是對 AI 開發工具的潛力充滿期待(如下方圖表所示)。然而，我們認為業界需要一段時間，才能將 AI 技術輔助工具廣泛融入各項作業。

## AI 的重要性

### 技術工作中的 AI 採用度

以您目前開發的主要應用程式或服務而言，人工智慧 (AI) 在以下各項工作中的重要程度為何？



AI 採用率如何隨時間成長？這股成長趨勢如何影響機構注重的成效指標和成果？

這兩個問題將是我們未來的關注重點。

# 說明文件奠定成 功基礎



## 重點摘要

優質說明文件是開發運作的重要基礎，能幫助團隊成功導入技術功能，並強化這些功能對機構績效的影響。此外，說明文件也有助於提升團隊績效、工作效率和工作滿意度。不過，提高說明文件品質不一定能促進每位員工的身心健康。部分作答者表示，工作倦怠程度反而因說明文件品質提高而增加。

## 前言

內部說明文件是一種書面記錄，記載機構員工在日常工作中使用的知識。今年，我們進一步調查說明文件對技術能力和關鍵成果的影響。

我們從可靠性、尋獲度、更新程度和關聯性等指標評估說明文件品質，並以一個分數來代表整體說明文件體驗。我們不會逐頁審視文件，而是衡量整體品質。

# 研究結果

## 說明文件是推動及強化技術能力的重要基礎

如同我們在 2021 年<sup>1</sup>和 2022 年<sup>2</sup>的調查結果，說明文件品質越好，機構越容易成功導入我們研究的各項技術能力。

此外，我們在 2022 年<sup>3</sup>的調查結果也顯示，說明文件品質能強化各項技術能力對機構績效的影響，如下表所示。

技術能力	對機構績效的影響力提升幅度*
持續整合	2.4 倍*
持續推送軟體更新	2.7 倍*
主幹式開發	12.8 倍*
鬆耦合架構	1.2 倍*
可靠性做法	1.4 倍*
人工智慧技術使用度	1.5 倍*

\*計算方式：

優質說明文件對技術能力的影響力

劣質說明文件對技術能力的影響力

## 說明文件可增進個人和機構成員的工作效率和身心健康

除了提升技術能力之外，我們還發現優質說明文件對個人身心健康有正面助益，包括降低工作倦怠、提升工作滿意度和工作效率，部分原因在於優質說明文件能促進知識共享。

這個結果並不令人意外，因為當員工身處知識共享的職場且能學習到必要知識，就更容易完成工作及減少挫折感。

身心健康層面	優質說明文件的影響
工作倦怠	大幅降低
工作滿意度	大幅提升
工作效率	大幅提升

此外，優質說明文件也對團隊和機構績效、運作成效等關鍵成果有所幫助。

關鍵成果	優質說明文件的影響
團隊績效	大幅提升
機構績效	大幅提升
軟體推送成效	無影響*
運作成效	大幅提升

<sup>1</sup>《加速發展：2021 年開發運作現狀報告》。<https://dora.dev/publications/pdf/state-of-devops-2021.pdf>

<sup>2</sup>《加速發展：2022 年開發運作現狀報告》。<https://dora.dev/research/2022/dora-report/2022-dora-accelerate-state-of-devops-report.pdf>

<sup>3</sup>「2022 State of DevOps Report data deep dive: Documentation is like sunshine」(2022 年開發運作現狀報告資料深度剖析：說明文件像陽光一樣不可或缺)。<https://cloud.google.com/blog/products/devops-sre深深剖析2022年DevOps报告中的说明文件>

\* 這個結果依舊令人感到驚訝，因為這是我們連續第二年發現，優質說明文件對軟體推送成效沒有影響。

優質說明文件為何能對這三項關鍵成果帶來正面影響？以讀者的角度而言，清楚的說明文件對工作大有助益。此外，編寫流程也可能是重要因素。要製作優質說明文件，首先需要確定流程，因此機構內的所有團隊都必須針對目標行動和執行方式，明確進行討論並達成共識。

優質說明文件好比團隊知識庫，即使人員流動，經驗也能繼續傳承下去。長期下來，機構將能累積豐富的知識。

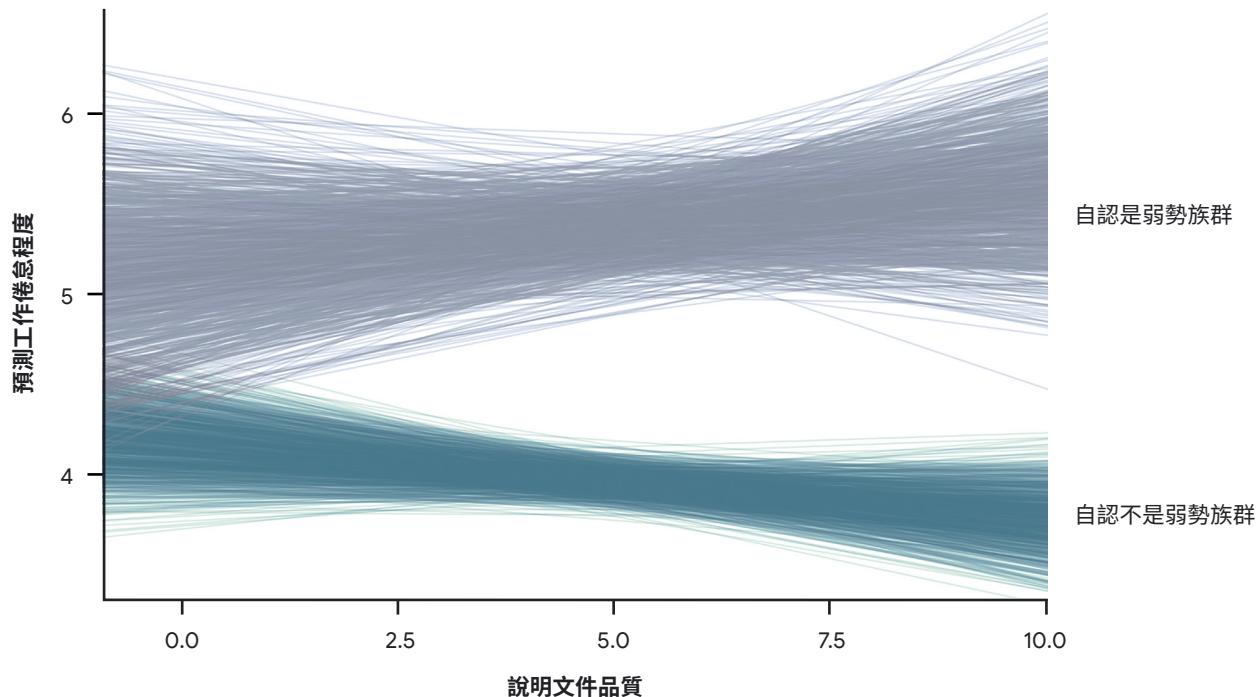
#### 說明文件是否會降低某些人員的身心健康？

我們調查具弱勢身分的作答者時，發現了意料之外的趨勢。對弱勢族群而言，優質說明文件反而會增加工作倦怠。

令人驚訝的是，這個現象並不存在性別差異，無論是男性、女性和自述性別的作答者，都能受惠於優質說明文件，大幅降低工作倦怠。然而，即便有優質說明文件，弱勢族群作答者的工作倦怠率都偏高，無論性別認同為何皆是如此。

下圖是根據資料繪製的模擬預測情形。在下層集合中，隨著說明文件品質增加，大多數作答者的工作倦怠逐漸降低。然而在上層集合中，弱勢作答者的工作倦怠情形大幅增加。

這張圖表以趨勢線模擬各組作答者的工作倦怠情形（每組1,000條線），根據我們的資料，線條越密集，斜率可能越大。



我們研究說明文件品質、創造型文化和團隊穩定性時也發現類似結果：這類特徵越明顯的團隊，弱勢族群的工作倦怠程度越高。為什麼說明文件會有這種現象？

製作及維護優質說明文件需要投入大量心力。這是一項對技術能力、團隊工作效率和機構績效帶來顯著影響的技術工作，但機構不一定看得見其重要性和影響力。弱勢族群的作答者是否承擔過多說明文件相關工作？如果是，這類工作真的就是造成工作倦怠的原因嗎？

仰賴說明文件是否會衍生其他問題？對部分作答者而言，文件品質提高是否無法促進知識共享？如果對這群作答者而言，提高文件品質確實能促進知識共享，是否仍不足以抵銷工作倦怠的各種成因？

或許有意料之外的因素造就說明文件品質提高，但同時使弱勢作答者產生或持續處於工作倦怠。我們需要進行更多研究。

從以上分析來看，「團隊中的身分」對身心健康有顯著影響。優質說明文件等職場因素能為團隊和整個機構帶來莫大效益，但可能也會對部分員工帶來負面影響。我們將在「[第8章：身分與身心健康的關聯性：原因、影響方式和時機](#)」進一步探討這點。

## 入門資源

歡迎閱讀 2021 年報告，瞭解如何提升說明文件品質<sup>4</sup>。今年，我們還發現工作分配（包括說明文件工作的正式分配程序）有助於大幅提升說明文件品質。

**目前有關技術文件寫作的資源和訓練課程相當豐富，歡迎透過以下資源瞭解詳情：**

- Society for Technical Communication（美國技術傳播協會）：[stc.org](http://stc.org)
- Technical Writing Courses for Engineers（給工程師的技術文件寫作課程）：[developers.google.com/tech-writing](http://developers.google.com/tech-writing)
- Write the docs（「學寫說明文件」網站）：[writethedocs.org](http://writethedocs.org)

<sup>4</sup>《加速發展：2021 年開發運作現狀報告》，22。<https://dora.dev/publications/pdf/state-of-devops-2021.pdf#page=22>

# 可靠性帶來更高成效

## 重點摘要

強大的可靠性做法能提高團隊、機構和運作成效。資料顯示，改善這些做法與成果之間屬於非線性關係。也就是說，機構培養更強大的能力後，成效不一定跟著提升。不過，只要持續致力改進這些做法，長期下來仍有希望取得良好成果。

## 前言

可靠性是 IT 營運領域中廣泛使用的概念，我們將其定義為服務達到聲稱目標的程度，評估指標包括可用性、效能和正確性。網站可靠性工程 (SRE) 是實現可靠性的常見做法，最初由 Google 提出 (<https://sre.google>)，現在獲眾多機構採用。SRE 優先考量經驗學習、跨部門協作、廣泛仰賴自動化技術，以及使用服務等級目標等評估技巧。

可靠性做法獲許多機構採用，但不一定都稱為 SRE，其他替代名稱包括產品工程、平台團隊、基礎架構團隊、技術營運等等。為了盡量客觀評估這些做法的採用程度，我們特別留意問卷調查的用字遣詞，力求使用中立的敘述性文字。

此外，我們也收集可靠性工程成果的資料，以瞭解團隊達成可靠性目標的程度。可靠性做法、成果（我們稱為**運作成效**）和其他技術能力，都反映在我們的預測模型中。

### 可靠性做法

我們請作答者從三大層面，思考自家運作情形的可靠程度。第一：是否制定計來減少依附關係？第二：是否定期測試災難復原計畫，包括模擬中斷情形、實際容錯移轉或進行桌上演練？第三：如未達到可靠性目標，會選擇直接改善，還是重新安排工作順序並調整工作內容？

我們認為，以上指標最能反映團隊是否遵循既定的 SRE 原則，例如「勇於面對風險」及「衡量使用者滿意度」。這類團隊會根據使用者滿意度設定合理的目標，然後執行測試來確保他們能達到目標。如果無法達成，便會調整計畫。我們會從這個角度判斷團隊是否成功實踐 SRE 概念，而非評估團隊是否確實導入特定 SRE 做法。

## 研究結果

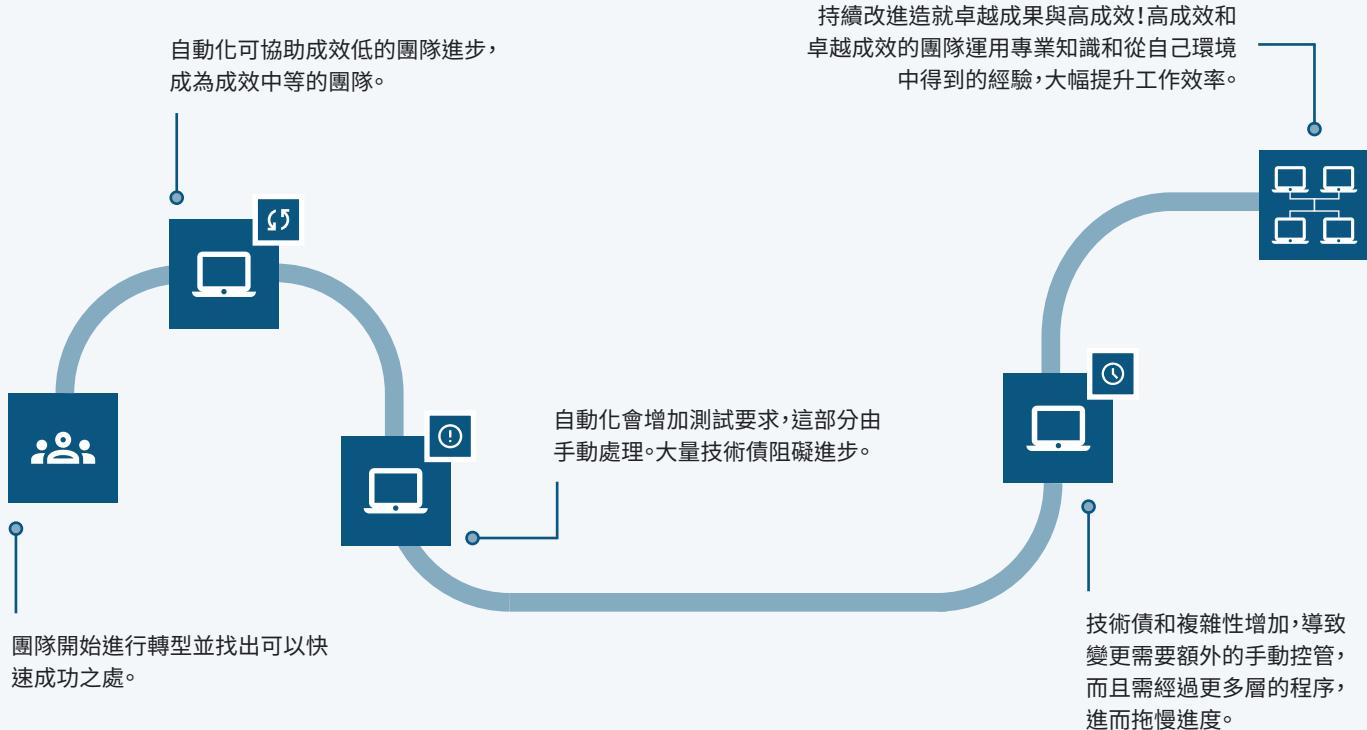
### 確認 J 曲線是否能描述可靠性做法

自 2018 年以來，DORA 推測運作成效與自動化等做法之

間具有非線性關係（圖 1）。深入研究可靠性做法後，我們從問卷調查資料中找到這個模式的證據。

在 2022 年，我們直接對團隊進行調查，並發現可靠性做法和成果之間的關係確實遵循這種非線性曲線（圖 2）。這代表團隊採用多種可靠性做法後，才能大幅提升可靠性<sup>1</sup>，但我們認為，從這個角度解讀資料無法掌握全貌。2022 年的曲線顯示，除非由專家使用，否則 SRE 並無投資價值。這與許多 SRE 團隊的經驗相悖，也意味著我們需要收集更多資料。

圖 1：2018 年的假設性 J 曲線



<sup>1</sup>《加速發展：2022 年開發運作現狀報告》，27–28。<https://dora.dev/research/2022/dora-report/2022-dora-accelerate-state-of-devops-report.pdf#page=27>

圖 2:2022 年曲線

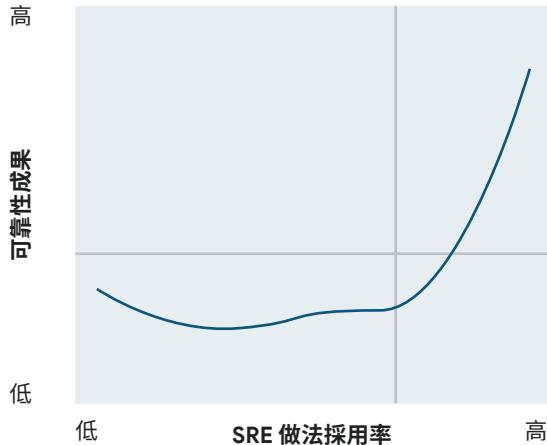
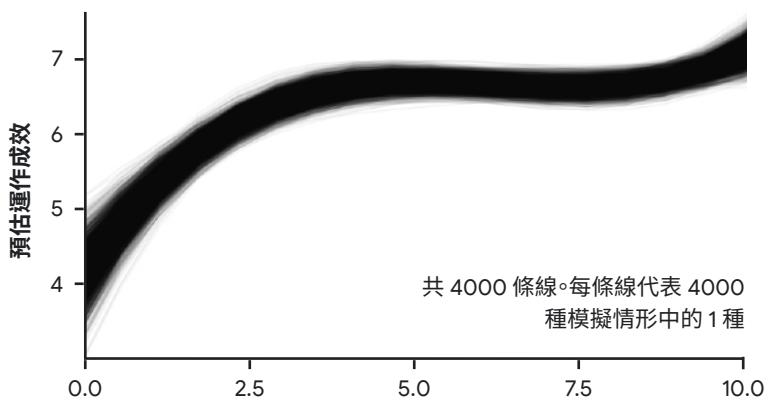


圖 3:2023 年曲線



在 2023 年的問卷調查中，我們加入更多問題，藉此描繪出更符合實際經驗的曲線。這條新曲線更接近 2018 年報告所述的假設性轉型 J 曲線（詳細分析方式請參見「[研究方法](#)」一節）。由此可知，在採用可靠性做法的早期階段，運作成效確實有所增加，隨後複雜性帶來的新難題將拖慢成長步伐，最後成效會再次上升。這項結果印證了許多團隊所經歷的狀況。

這條曲線有以下幾個重要性：

- 公司可在一開始安心籌措資金並導入 SRE，即使目標不是將可靠性提升到最高，或投注大量資金與心力在 SRE 也無妨。就算可靠性做法採用程度較低，也能改善運作成效，進而提升團隊和機構績效。

- 如果公司「有意」大量投資來提升可靠性，可根據 J 曲線做好準備，度過停滯期。機構往往認為 SRE 投資時間越長，獲得的正面成果越可觀，但實際資料顯示並非如此。如果團隊事先瞭解 J 曲線的非線性關係，即可決定是否進行這項投資並提早規劃，以免在 SRE 做法完全發揮效益前貿然放棄。

- 公司可能需推動文化轉型<sup>2</sup>，才有利於進行這類變革。我們發現，公司若能上下一心同時調整，將更容易取得成功。主管可以即時獎勵採用可靠性做法的團隊，鼓勵其他團隊跟進，進而提高採用率。相關獎勵和具體計畫可根據 J 曲線設計。

<sup>2</sup>「How to transform」(轉型方法)。<https://dora.dev/devops-capabilities/cultural/devops-culture-transform/>

## 可靠性做法與身心健康

傳統營運做法通常過於被動，且注重技術系統的健康狀態，而非系統使用者的身心健康。在這種做法下，值班團隊可能會收到與使用者體驗無關的事件快訊、必須手動處理重複性工作，甚至害怕出錯等等，導致每個團隊成員工作倦怠，身心健康不佳。

我們發現，採用可靠性做法的團隊不僅工作效率和工作滿意度更高，而且工作倦怠程度更低，與不使用這類做法的團隊完全相反。我們認為，某些已發表的 SRE 做法就是促進身心健康的主因：

- 減少手動作業<sup>3</sup>
- 不究責的檢討報告<sup>4</sup>
- 團隊自主權<sup>5</sup>
- 營運團隊的擴編速度，不一定要等於系統數量的成長速度<sup>6</sup>

## 運作成效

另外，我們也請作答者描述自家服務的運作成效：第一，是使用者主動表示不滿意服務可靠性的頻率。第二，是服務無法使用、速度緩慢或未正確運作的頻率。

### 可靠性做法能強化運作效率，進而提升團隊和機構績效

採用可靠性做法有助於團隊提升運作成效。我們發現，如果正式環境中的機群能以高效率運作，將連帶強化機構其他方面的成效。如果這些面向的成效原本就很高，可靠性做法可進一步推升成效。不過，如果這些面向的成效原本就偏低，可靠性做法無法帶來任何助益，成效也會維持低水準。

因此，您仍需具備適當的軟體能力，才能透過可靠系統有效率地向顧客推送服務。這很合理，因為 SRE 無法憑空運作。達到可靠性目標，是 SRE 團隊取得成功的關鍵指標，這個成果也會反映在運作成效。採用可靠性做法或許能帶來其他效益，不過資料顯示，最大的效益就是提升運作成效。提升運作成效的好處不只是改善服務健康狀態，我們的證據顯示，採用可靠性做法能確實促進從業人員的身心健康。

### 運作成效影響身心健康

業界普遍認為，服務可靠性高，代表營運人員必須隨時待命，或在非工作時間進行緊急維護作業，因此會對他們的身心健康造成負面影響。然而我們發現，高運作成效其實可降低工作倦怠，以及提升工作效率和工作滿意度。這點符合 SRE 的「減少手動作業」原則<sup>7</sup>，也就是將手動作業自動化可促進人員身心健康，同時減輕團隊的日常工作負擔。

<sup>3</sup>Beyer、Betsy 等人，《Site Reliability Engineering: How Google Runs Production Systems》(網站可靠性工程：Google 的系統管理之道)，O'Reilly 出版，2016 年，49–54。<https://sre.google/sre-book/eliminating-toil/>

<sup>4</sup>出處同上。<https://sre.google/sre-book/postmortem-culture/>

<sup>5</sup>Beyer、Betsy 等人，《The Site Reliability Workbook》(網站可靠性工程工作手冊)，O'Reilly 出版，2018 年。<https://sre.google/workbook/team-lifecycles/>

<sup>6</sup>Brookbank, James 和 McGhee, Steve，《Enterprise Roadmap to SRE》(企業的 SRE 導入藍圖)，O'Reilly 出版，2022 年，11。<https://sre.google/resources/practices-and-processes/enterprise-roadmap-to-sre/>



## 機構和團隊績效可強化運作成效

運作成效對團隊和機構績效具有顯著的正面影響。對奉行開發運作做法的公司而言，這點並不令人意外。能高效營運機器的團隊可取得更多成果，帶動機構蓬勃發展。



## 運作成效可強化軟體推送成效

雖然軟體推送和運作成效都可幫助提升團隊和機構績效，不過後者帶來的助益更大。此外，若團隊的軟體推送成效很高，但未能實現同等的運作成效，則依然無法取得優異的團隊和機構績效，換句話說，兩者缺一不可。事實上，如果團隊提高軟體推送成效，卻不具相應的運作成效，反而會拉低機構績效。因此，即使您可快速編寫優質軟體，但軟體在正式環境運作的表現無法達到目標使用者的期望，還是無法獲得市場青睞。

## 不足之處與後續行動

我們認為，其他衡量指標可以幫助我們瞭解這些交互關係。例如在今年的問卷調查中，我們經常詢問作答者成本管理對改進這些能力和成果的作用。部分機構較看重成本，這也間接透露他們如何制定計畫和決策。我們也推測，即使機構並未明確推行 SRE 或制定相關計畫，仍可能經由高度合作的文化自然發展出可靠性做法<sup>8</sup>。我們希望更深入瞭解團隊如何改進現有的 IT 營運做法，以及這對系統可靠性、團隊績效和身心健康的影響。

最重要的是，我們希望能獲得您的寶貴意見。歡迎前往 [DORA.community](#)，與我們和其他從業人員相互交流<sup>9</sup>。

SRE 仍是相當新穎的領域，某些機構在 SRE 問世後才開始採行，某些則已行之有年，因此 SRE 對每個機構的影響力不盡相同。成效變化並非短時間能觀察到的現象，我們希望制定一致的評估標準來衡量長期進展。若能集結社群的力量，相信我們能在這條路上互助合作、持續成長。

<sup>7</sup> Beyer、Betsy 等人，《Site Reliability Engineering: How Google Runs Production Systems》(網站可靠性工程：Google 的系統管理之道)，O'Reilly 出版，2016 年，49–54。<https://sre.google/sre-book/eliminating-toil/>

<sup>8</sup> Brookbank, James 和 McGhee, Steve，《Enterprise Roadmap to SRE》(企業的 SRE 導入藍圖)，O'Reilly 出版，2022 年，5。<https://sre.google/resources/practices-and-processes/enterprise-roadmap-to-sre/>

<sup>9</sup> DORA 社群，<https://dora.community/>

# Google 如何實行 SRE

身為持續成長且資金充裕的企業，Google 已發展 SRE 二十年。我們導入 SRE 的目的，是在不耗費龐大成本的情況下高速發展 Google 搜尋和 Google Ads。這些產品講求即時性，意味著我們在開發之初必須做到高可靠性。就算是在搜尋內容或放送廣告期間發生暫時性錯誤，產品也會立刻失去消費者和客戶的信任，而且沒有機會重來。動態廣告一向以毫秒為單位計算，搜尋速度緩慢將使 Google 品牌的信用大打折扣。

與此同時，我們將自己視為一間全新公司並發展 SRE 做法，以工程導向的精神從頭建構技術，而非購買現成技術。Google 聘用了一群網站可靠性工程師，他們在分散式系統和編譯器設計等電腦科學領域擁有豐富的專業知識。這個核心文化奠定了深厚基礎，孕育出強大的 SRE 做法。SRE 團隊不僅肩負將系統投入正式環境的重任，還要設法以新方式擴充系統。

那麼，SRE 團隊如何隨著時間拓展規模？一言以蔽之，我們遵守「非線性」的概念。也就是說，每當 Google 業務規模擴展一倍，我們聘用的網站可靠性工程師人數並不會跟著增加一倍。早期 Google 需要快速擴大產品服務範圍才能滿足全球需求，同時還要推出 Gmail、Google 地圖、Android、YouTube 和 Google Cloud 等新產品，加上 SRE 是全新領域，相關人才稀缺，因此 SRE 團隊的拓展速度不可能跟上客群成長速度。

這正是以下說法的由來：

- SRE 不可隨著使用者人數線性成長。
- SRE 不可隨著伺服器數量線性成長。
- SRE 不可隨著叢集數量線性成長。
- SRE 不可隨著服務數量線性成長。

為了控制團隊規模成長速度，Google 制定了一套管理結構。在這套結構下，網站可靠性工程師會互相協調並達成共識，同時與產品開發團隊密切合作。如有需要，網站可靠性工程師會透過專門管理鏈，向上呈報至主管 Ben Treynor-Sloss<sup>10</sup>。SRE 團隊與產品開發團隊隸屬於相同產品領域，且會保持同步。SRE 團隊會與開發團隊合作決定如何運用網站可靠性工程師，另一方面，開發團隊也可以充任網站可靠性工程師。並非所有開發團隊都負責打造面向使用者的產品，其中包括許多共用基礎架構的團隊，例如 Bigtable（結構化資料儲存庫）<sup>11</sup>、Borg（運算排程）<sup>12</sup>和 Colossus（分散式儲存空間）<sup>13</sup>。由於這些服務共用基礎架構，因此面向消費者的團隊可以在沒有專屬 SRE 團隊的情況下調度資源。

<sup>10</sup> 工程部門副總裁 Benjamin Treynor Sloss。《Site Reliability Engineering: How Google Runs Production Systems》（網站可靠性工程：Google 的系統管理之道），O'Reilly 出版，2016 年。<https://sre.google/sre-book/part-i-introduction/>, <https://sre.google/sre-book/introduction/#id-2opuzSjFr>

當各個開發部門擁有自己的 SRE 團隊，招募和晉升流程將更能維持一致。SRE 團隊規模通常不到開發團隊的十分之一，因此確保 SRE 團隊擁有自主權且不違背 SRE 原則至關重要。

網站可靠性工程師會與內部產品經理合作，開發內部 SRE 產品，而產品的使用者，就是期望提升正式環境運作成效的其他團隊。SRE 團隊開發的產品，都與事件回應、監控與觀測能力、版本管理、處理能力規劃和疑難排解相關。

SRE 團隊的擴編過程，始終以成員身心健康和永續發展為優先考量。舉例來說，新成立的待命 SRE 團隊最少包含 12 名成員，兩個網站各配置 6 人。這能確保團隊人數足以提供跨時區支援、避免工作倦怠，讓每個人都確實享有工作與生活的平衡。

時至今日，SRE 的內容仍不斷調整。並非所有團隊都完全遵循 SRE 模式運作，部分會採取替代做法。如同業界其他大型公司，Google 內部的大型 SRE 組織也會受市況影響，因此保持靈活非常重要。不過，SRE 團隊仍有一些不可動搖的原則，包括勇於面對風險、衡量服務水準、消除手動作業、採取自動化作業及追求簡便設計。



<sup>11</sup> Chang, Fay 等人，《Bigtable: A Distributed Storage System for Structured Data》(Bigtable：結構化資料的分散式儲存系統)，第 7 屆 USENIX 作業系統設計與實作研討會 (OSDI)；{USENIX} (2006 年)，pp. 205–218，<https://research.google/pubs/pub27898/>

<sup>12</sup> Verma, Abhishek 等人，《Large-scale cluster management at Google with Borg》(在 Google 使用 Borg 管理大量叢集)，歐洲電腦系統會議論文集 (EuroSys)，電腦協會，波爾多，法國 (2015 年)，<https://research.google/pubs/pub43438/>

<sup>13</sup> Hildebrand, Dean 等人，「Colossus under the hood: a peek into Google's scalable storage system」(隱身幕後的 Colossus：一窺 Google 可擴充的儲存空間系統)，2021 年 4 月 19 日，<https://cloud.google.com/blog/products/storage-data-transfer/a-peek-behind-colossus-googles-file-system>

# 彈性基礎架構是成功關鍵

## 重點摘要

彈性基礎架構有利於提升團隊、機構、運作及軟體推送成效。雲端運算是彈性基礎架構的重要推手，但機構也不能抱持坐享其成的心態，因為我們的資料顯示，「雲端使用方式」才是關鍵。

## 前言

在許多 DORA 研究中，我們曾詢問從業人員的自家基礎架構，是否具備美國國家標準暨技術研究院 (NIST) 定義的雲端運算必要特徵：<sup>1</sup>

- 隨選自助式服務
- 多元的網路存取方式
- 資源集區
- 機動彈性
- 計量服務

我們一向認為，以上五個特性有助於提升機構績效和軟體推送成效。今年，我們想瞭解雲端運算是否能提高基礎架構彈性。

與缺乏彈性的基礎架構相比，  
彈性基礎架構可提升  
**30%**  
機構績效

<sup>1</sup>NIST 特別出版品 800-145：《The NIST Definition of Cloud Computing》(NIST 雲端運算定義)。

百分比	
多雲端	19.6%
公有雲	51.5%
混合雲	33.6%
地端部署	19.8%
桌下工作站	3.2%
其他	2.5%

\* 作答者可選取多個答案。

## 研究結果

今年，我們再度印證往年的研究結果：相較於「是否使用雲端」，對團隊績效的影響更大的是「雲端使用方式」。雖然「是否使用雲端」是相當重要的驅動因素，但不會自動帶來各種效益。事實上，研究結果明顯指出，「除非」團隊同時使用彈性基礎架構，否則公有雲會「拉低」軟體成效和運作成效。這個研究結果進一步證明，若機構只是「隨即轉移」到雲端（也就是將資料中心工作負載轉移到雲端），將無法享有任何效益，甚至會招來負面影響。

使用雲端運算有助於大幅降低工作倦怠，並顯著提升工作效率和滿意度。

### 運算環境

上表為作答者回覆的主要應用程式或服務運作環境。



雲端類型	機構績效	團隊績效	軟體推送成效	運作成效
私有雲		無影響		無影響
公有雲				雲端運算大幅提升這項指標
混合雲		雲端運算極大幅提升這項指標		雲端運算大幅提升這項指標
多雲端				無影響

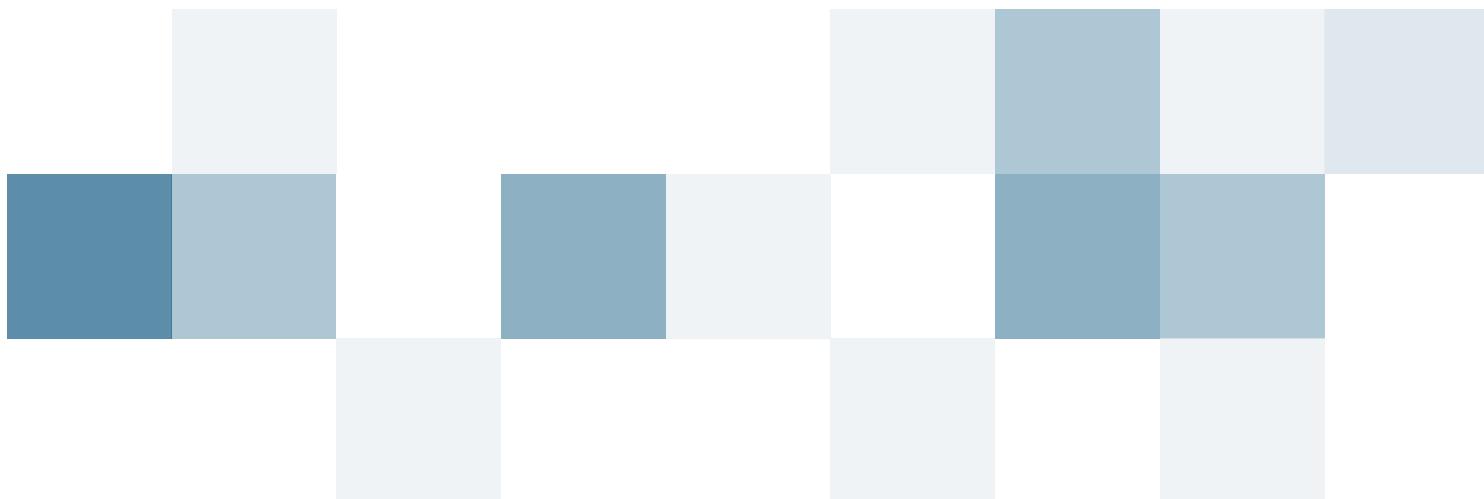
### 純粹「使用雲端」的效果有好有壞

如上方研究結果表所示，純粹「使用雲端」對軟體推送和運作成效沒有明顯效益，甚至造成負面影響，原因可能是從業人員剛開始雲端技術，仍在適應新的環境和作業工具，以及嘗試不同的工作方式。企業通常會將自家資料中心的使用方式沿用至雲端，但新環境會使複雜性和認知負擔升高。如果無法適應新環境，軟體推送或運作成效不僅難以成長，還

可能會衰退。

這項研究結果有一個例外情況：使用私有雲能大幅提升運作成效。

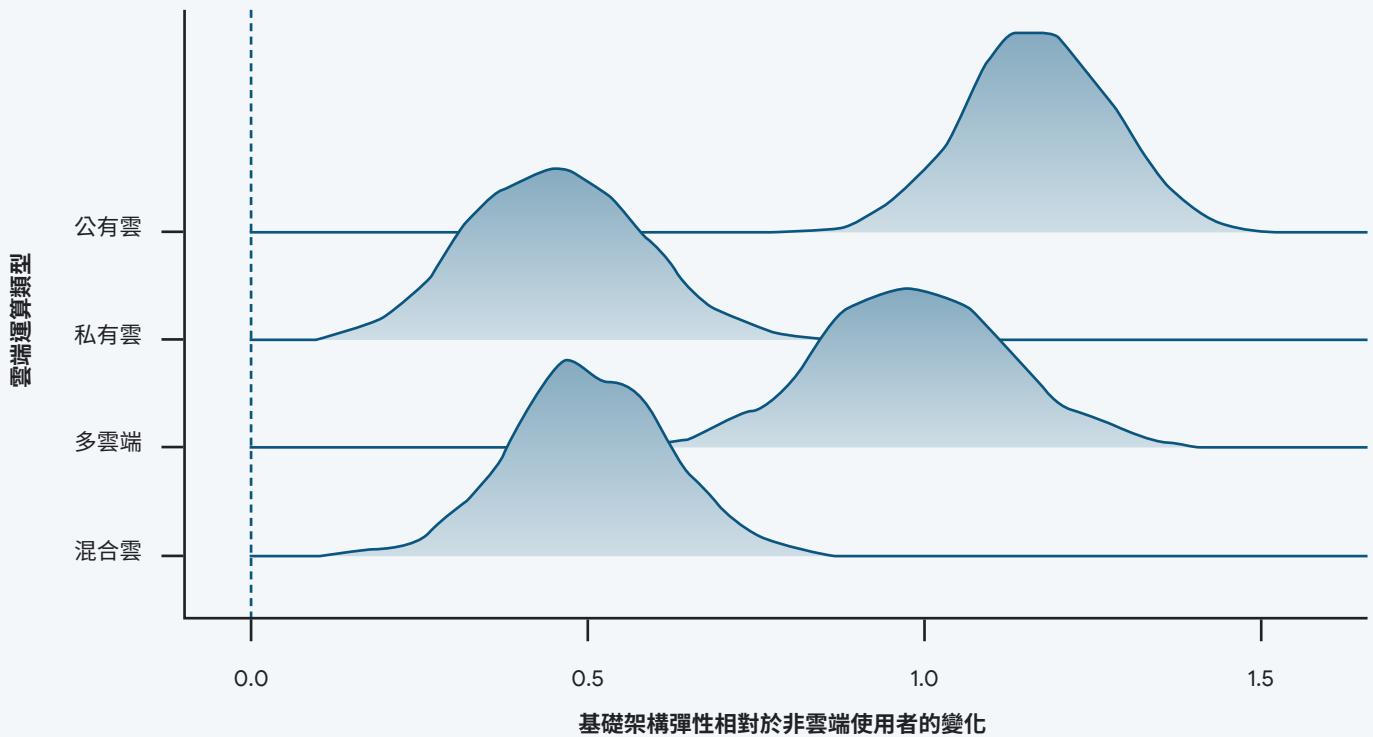
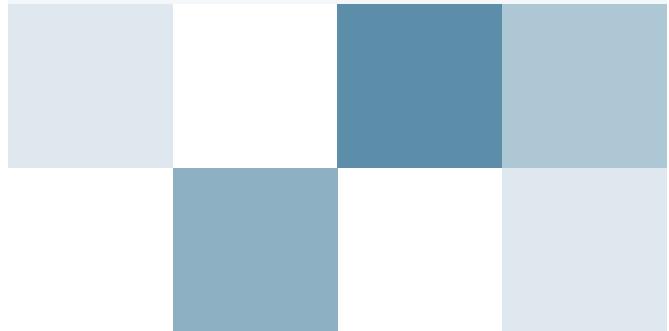
**彈性基礎架構**才是「確實」提升軟體推送和運作成效的關鍵，我們稍後會討論這點。



## 雲端基礎架構可提升彈性

相較於不使用雲端，使用公有雲可增加 22% 基礎架構彈性。使用多雲端也能增加彈性，但效果不如單一公有雲。究竟為什麼會這樣？資料顯示，相比只採用單一雲端平台，使用彈性基礎架構（通常由雲端運算驅動）的效益更大。對大多數公司而言，雲端意味著新的工作方式，需要一段時間才能運用自如。每個雲端平台的特性各有不同，使用的數量越多，就需要花越多心力熟悉操作方式。

相較於不使用雲端，使用公有雲可增加基礎架構彈性



## 彈性基礎架構有助於改善關鍵成果

技術能力	機構績效	團隊績效	軟體推送成效	運作成效
彈性基礎架構	 更彈性的基礎架構可大幅提升成效	 更彈性的基礎架構可大幅提升成效	 更彈性的基礎架構可大幅提升成效	 更彈性的基礎架構可提升成效

就機構、團隊、軟體推送和運作成效而言，彈性基礎架構的助益確實不容忽視。許多機構選擇將基礎架構隨即轉移到雲端，這是很好的開始，但只是整個旅程中的第一步。如果您決定隨即轉移部分工作負載，下一步請透過重構來翻新工作負載，以充分運用彈性基礎架構。

## 雲端運算造就彈性基礎架構，進而改善多項關鍵成果

採用彈性基礎架構的雲端類型	機構績效	團隊績效	軟體推送成效	運作成效
私有雲	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果
公有雲	 彈性基礎架構具完全中介效果	 彈性基礎架構具完全中介效果	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果
混合雲	 彈性基礎架構具部分中介效果	 彈性基礎架構具完全中介效果	 彈性基礎架構具部分中介效果	 彈性基礎架構具完全中介效果
多雲端	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果	 彈性基礎架構具部分中介效果

在充分善用彈性基礎架構的情況下，雲端運算平台可對軟體推送和運作成效帶來正面影響。這種影響力差異證明了大多數從業人員和主管都瞭解的一件事：純粹將資料中心工作負載轉移到雲端，並不會帶來任何效果，關鍵是善用雲端環境賦予的彈性基礎架構。

基礎架構類型	成果
搭配彈性基礎架構的雲端	
未搭配彈性基礎架構的雲端	

為了充分發揮雲端的效用，您必須重新思考如何建構、測試、部署及監控應用程式，並特別著重善用雲端運算的五大特性：隨選自助式服務、多元的網路存取方式、資源集區、機動彈性，以及計量服務。



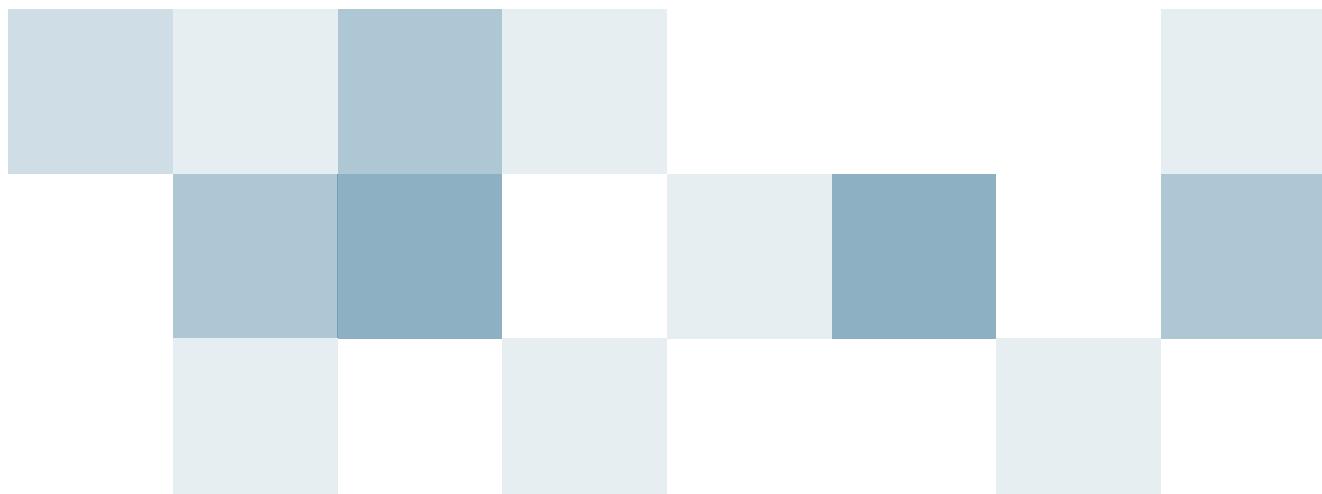
## 雲端運算可促進身心健康

雲端類型	工作倦怠*	工作滿意度	工作效率
私有雲	 無影響		
公有雲	 雲端運算極大幅降低這項指標	 雲端運算大幅提升這項指標	 雲端運算大幅提升這項指標
混合雲	 無影響		
多雲端	 無影響		

\*您可能會發現「工作倦怠」部分的色彩配置不同，因為工作倦怠減少是好事！

資料顯示，雲端運算對員工身心健康大有助益，也能顯著提升工作效率和滿意度，在減緩工作倦怠方面也有正面影響（不過有時無明顯效果）。換句話說，採用雲端技術後，雖然員工需要花額外心力學習新的工具和工作方式，但身心健康不會受到嚴重影響。

身為從業人員，我們認為這個現象的可能原因如下：工程師樂於學習及解決問題，且喜歡在彈性運算環境中工作。學習新技術不僅有趣，還有助於職涯發展，工程師也樂見公司取得成功。



# 營造健康的文化才能推動革新

## 重點摘要

文化是增進員工身心健康和機構績效的重要驅動力。健康的文化能降低工作倦怠，提高工作效率和滿意度，也能有效提升機構和團隊績效，以及軟體推送和運作成效。健康的機構文化，能讓團隊更順利導入技術能力，進而取得更優異的營運成效。



# 前言

定義文化並不容易，因此我們著重觀察能呈現工作體驗的指標。我們選用的是 Westrum 機構文化分類學<sup>1</sup>，因為這套原則長久以來都能準確評估機構績效。今年的研究結果與往年相同：我們發現機構若具備 Westrum 定義的創造型文化，將擁有良好績效。

下表是我們認為有利於培養團隊和機構文化的關鍵。

層面	定義
Westrum 機構文化	機構因應問題和機會的方式，包含三種文化類型：創造型、官僚型和病態。
機構穩定性	工作環境對員工而言是否穩定。
就業保障	員工對於工作穩定性的擔憂頻率。
工作彈性	員工處理工作的方式、地點和時間。
知識共享	想法和資訊在整個機構內的流通情形。在知識共享程度高的機構中，團隊成員只需回答問題一次，大家就能取用資訊，不必費時等待答案。
以使用者為主	以使用者為重心來開發軟體，且相當瞭解他們的需求和目標，也會根據使用者信號持續完善產品和服務。
工作分配	團隊將繁重工作平均分配給成員的正式程序。

文化層面、程序能力和技術能力之間的界線往往不明確，而 Google 相信文化塑造做法，做法孕育文化。稍後討論研究成果時，我們將一併說明這點。

<sup>1</sup><http://bmj.co/1BRGh5q>

# 研究結果及其意義

## 健康的文化有助於改善關鍵成果

整體而言，健康文化對所有關鍵成果都有正面影響。今年的研究結果與往年相同：創造型文化可提升運作、軟體推送和機構成效，甚至連今年新加入的團隊績效指標也有所提升。

我們發現，以使用者為核心的軟體開發做法，能確實提升績效。這是值得關注的重點。以使用者為先的機構能獲得莫大效益。使用者意見有助於團隊安排專案優先順序，並打造符合使用者需求的產品和服務，進而創造更好的使用者體驗、提高使用者滿意度並帶動收益成長。

我們也評估了所有團隊的工作分配情形，以衡量機構文化的健康狀態。結果顯示，平均分配工作有助於提升團隊和機構績效，但會「降低」軟體推送成效。也許正式的工作分配程序，反而會導致軟體推送管道中的繁重工作更慢完成，或者打亂了團隊成員應負責的特定工作。

另外我們也注意到一項令人意外的結果，機構穩定性會顯著降低軟體推送成效，不過幅度並不很大。原因可能在於，成熟且規模較大的機構已適應快速的工作步調；相較之下，剛起步且規模較小的機構難以快速作業，因而備感

與不具創造型文化的團隊相比，擁有這類文化的團隊可提升

30%  
機構績效

壓力。成熟的機構可能已有成熟的產品，因此軟體推送做法更靈活，速度更快。

只要資訊容易流通，就能更順利完成工作。我們發現，知識共享程度越高，軟體推送和運作成效越好。如果知識流通，資訊隨取即得，員工就能將時間投入重要的工作，而不是四處查詢處理工作所需的資訊。

最後，彈性工作模式代表員工可以自行決定工作的時間、地點和方式。所有成效指標都能因此提升，特別是軟體推送成效。這表示即使機構收緊遠距工作政策，讓員工維持一定工作彈性可能會是利大於弊。

文化層面	對團隊績效的影響	對機構績效的影響	對軟體推送成效的影響	對運作成效的影響
Westrum 機構文化	↑↑ 大幅提升	↑↑ 大幅提升	↑↑ 大幅提升	↑↑ 大幅提升
機構穩定性	↑ 小幅提升	↑↑ 大幅提升	↓ 小幅降低	(-) 無影響
就業保障	↑ 小幅提升	(-) 無影響	↑ 小幅提升	↑ 小幅提升
工作彈性	↑ 小幅提升	↑ 小幅提升	↑↑ 大幅提升	↑ 小幅提升
知識共享	↑ 小幅提升	↓ 小幅降低	↑↑ 大幅提升	↑↑ 大幅提升
以使用者為主	↑↑ 大幅提升	↑↑ 大幅提升	↑ 小幅提升	↑↑ 大幅提升
工作分配	↑↑ 大幅提升	↑↑ 大幅提升	↓↓ 大幅降低	(-) 無影響



## 健康的文化有助於改善技術能力

今年的研究結果指出，良好的文化對於導入技術能力有幫助。我們認為，文化與技術能力之間會互相影響：文化塑造做法，做法孕育文化。

文化的概念較廣泛且難以定義，但技術能力通常在範圍和定義上較明確，而這項能力會影響所有機構成員推動變革的方式。

舉例來說，主管可以制定獎勵制度來推廣創造型文化，主管和個人貢獻型員工都能著重採用以使用者為主的軟體

開發做法，個人貢獻型員工能協助導入可改善成效的技術能力，包括主幹式開發、持續整合、可靠性做法和鬆耦合架構。導入這些技術能力並不容易，需要眾人同心協力、保持開放心態、相互幫助及學習才能成功，這些都是構成健康文化的要素。身處這種環境下的團隊，更有能力運用手邊資源推動革新，成為機構內其他團隊的榜樣。

如要實現長期有效的機構文化革新，機構需要展現上下一心的團結精神。

文化層面	對主幹式開發成效的影響	對可靠性做法成效的影響	對持續整合成效的影響	對持續推送軟體更新成效的影響	對鬆耦合架構成效的影響
Westrum 機構文化	 大幅提升	 大幅提升	 大幅提升	 大幅提升	 大幅提升
機構穩定性	 小幅提升	 大幅提升	 無影響	 無影響	 無影響
就業保障	 小幅降低	 小幅降低	 無影響	 無影響	 無影響
工作彈性	 無影響	 小幅降低	 大幅提升	 小幅提升	 大幅提升
知識共享	 無影響	 無影響	 無影響	 小幅提升	 小幅提升
以使用者為主	 大幅提升	 大幅提升	 大幅提升	 大幅提升	 大幅提升
工作分配	 大幅提升	 大幅提升	 大幅提升	 大幅提升	 大幅提升

## 健康的文化有益於員工身心健康

健康的文化可降低工作倦怠、提升工作效率和滿意度，以及改善員工身心健康狀況。員工身心健康並不是可有可無，而是奠定機構整體營運狀態和成功的基礎。

工作效率下滑，身心健康都受到嚴重影響<sup>2,3</sup>。工作倦怠是長期累積的結果，僅靠休假一段時間無法完全復原。員工也可能會選擇離職，尋找更健康的工作環境，導致機構的人員流動率提高<sup>4</sup>。因此，改革機構文化是減緩工作倦怠的根本之道。

如果機構「不培養」更健康的文化會發生什麼事？員工的工作倦怠機率增加、工作滿意度下降、容易陷入負面思考、工

文化層面	對工作倦怠的影響*	對工作滿意度的影響	對工作效率的影響	
Westrum 機構文化		大幅降低		大幅提升
機構穩定性		大幅降低		大幅提升
就業保障		大幅降低		小幅提升
工作彈性		小幅降低		小幅提升
知識共享		大幅降低		小幅提升
以使用者為主		小幅降低		大幅提升
工作分配		無影響		小幅提升

\*您可能會發現「工作倦怠」部分的色彩配置不同，因為工作倦怠減少是好事！

<sup>2</sup> Adam Bayes、Gabriela Tavella 和 Gordon Parker (2021年),《The biology of burnout: Causes and consequences》(工作倦怠生物學：成因與結果), The World Journal of Biological Psychiatry, 22:9, 686–698 DOI:10.1080/15622975.2021.1907713. <https://doi.org/10.1080/15622975.2021.1907713>

<sup>3</sup> Maslach C 和 Leiter MP,《Understanding the burnout experience: recent research and its implications for psychiatry》(瞭解工作倦怠：近期研究及工作倦怠對精神病學的影響), World Psychiatry, 2016 年 6 月, 15(2), 103–11 DOI:10.1002/wps.20311, <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4911781/> PMID:27265691; PMCID:PMC4911781

<sup>4</sup> L.A.Kelly 等人,《Impact of nurse burnout on organizational and position turnover》(護理師工作倦怠對機構和職位流動率的影響), Nursing Outlook, 2021 年 1 月, 96–102, 2021 年 1 至 2 月, 96–102 DOI:[doi.org/10.1016/j.outlook.2020.06.008](https://doi.org/10.1016/j.outlook.2020.06.008)

# 身分與身心健康的關聯性： 原因、影響方式和時機

## 重點摘要

身分對身心健康的影響顯著。我們今年發現，某些作答者在身心健康方面的狀態他人較差，例如工作倦怠程度更高，或工作效率較低。我們也歸納出幾項可減輕這些負面狀態的具體做法。

## 前言

2022 年的研究結果反覆提到一個現象：工作安排方式可能只對部分員工的身心健康有利，而非所有員工。

在 2022 年的研究中，我們發現具弱勢身分的作答者有更嚴重的工作倦怠<sup>1</sup>。今年的研究依舊呈現相同結果，因此這一章我們將探討弱勢族群更容易出現工作倦怠的原因，以及哪些做法有助於避免這種情況。

此外，經濟環境動盪也導致許多產業在培養新人上面臨問題。機構擔憂新進員工需要較長時間才能提升工作效率，因此正在設法協助他們更快上手。本章也會深入探討這個現象。

## 研究結果及其意義

### 部分作答者更容易出現工作倦怠

我們去年發現，與男性和非弱勢族群的作答者相比，女性、自述性別以及在各方面都屬於弱勢的作答者，更容易出現工作倦怠。這些研究結果與往年一致，當時我們發現弱勢族群的工作倦怠程度<sup>2</sup>、工作壓力<sup>3</sup>都比非弱勢族群來得明顯。

因此，我們想瞭解今年的資料是否仍在工作倦怠方面呈現差異，而結果確實如此。與男性作答者相比，女性或自述性別的作答者工作倦怠程度高出 6%；與非弱勢族群的作答者相比，在各方面都屬於弱勢的作答者工作倦怠程度高出 24%。

<sup>1</sup>《加速發展：2022 年開發運作現狀報告》。<https://dora.dev/research/2022/dora-report/>

<sup>2</sup> Sigalit Ronen 和 Ayala Malach Pines, «Gender Differences in Engineers' Burnout»(工程師工作倦怠現象的性別差異), Equal Opportunities International, 2008 年 11 月 7 日, <https://www.emerald.com/insight/content/doi/10.1108/02610150810916749/full/html>

<sup>3</sup> Dalessandro C.、Lovell A. 和 Tanner O.C., «Race, Marginalization, and Perceptions of Stress Among Workers Worldwide Post-2020»(2020 年以來全球勞工的種族、邊緣化和壓力意識), Sociological Inquiry, 2023 年 8 月 3 日, <https://onlinelibrary.wiley.com/doi/10.1111/soin.12505>

## 部分工作類型更容易引發工作倦怠

今年的研究結果顯示,看似無害或有益的職場特徵(例如優質說明文件或穩定的團隊),未必能降低所有員工的工作倦怠情形(參見「[第4章:說明文件奠定成功基礎](#)」)。我們想瞭解這是否代表某些工作有利於機構,卻可能導致部分員工出現工作倦怠<sup>4</sup>。

為瞭解作答者的工作倦怠情形,我們從以下兩個面向評估他們負責的工作:

- 具體工作內容,例如寫程式、開會或支援團隊成員。
- 工作性質,例如臨時業務、工作重要性或手動作業量。

工作性質是評估工作倦怠的重點,因為同一份工作在不同時間處理,或交由不同人處理,帶來的工作體驗大不相同。例如,對您的團隊而言,某些程式碼審查工作可能只是臨時手動作業,其他審查工作才是團隊主要業務,能充分彰顯團隊的領導能力和專業知識<sup>5</sup>。

與非弱勢族群的作答者相比,弱勢身分作答者負責的重複性工作(即手動作業)多出24%。與男性作答者相比,女性或自述性別的作答者負責的重複性工作多出40%。這兩組作答者也負責更多臨時業務,以及重要性不如同事、且用不到自身專業技能的業務。以上研究結果多少透露了這些族群出現工作倦怠的原因。

<sup>4</sup> Linda Babcock、Brenda Peyser、Lise Vesterlund 和 Laurie Weingart,《The No Club》(說「不」俱樂部),紐約Simon and Schuster 出版,2022 年,17。

<sup>5</sup> Murphy-Hill, E. 等人,《Systemic Gender Inequities in Who Reviews Code》(程式碼審查人員的系統性性別不平等),Computer Supported Cooperative Work (2023 年),出版中,<https://research.google/pubs/pub52204>

<sup>6</sup> Babcock 等人,《The No Club》(說「不」俱樂部),17。

## 無助於升遷的業務

《The No Club》(說「不」俱樂部)這本書提出了一種工作類型,稱為「無助於升遷的業務」。作者 Babcock 等人對其描述如下:「無助於升遷的業務對企業至關重要,但對個人職涯發展沒有助益」<sup>6</sup>。

這句話中的「職涯發展」涵蓋很多意思,可以是加薪,或是提升職場競爭力並換工作。

證據指出,無助於升遷的業務通常由女性負責。對於這種工作分配不平等的現象,Babcock 等人點出幾項原因。例如,女性較常獲派處理這類工作,且往往礙於人際關係而難以拒絕。

作者也指出工作分配不平等的後果,例如:

- 薪水或職涯難以成長。
- 須耗費更多時間完成職務範圍內的工作

## 正式的工作分配程序幫助部分作答者降低工作倦怠

我們詢問作答者任職的公司是否透過正式程序平均分配工作。我們稱這個程序為**工作分配**，且認為這個做法應可減緩部分作答者的工作倦怠。

結果顯示，工作分配程序確實減少男性、女性和自述性別作答者的工作倦怠。如果分配程序相當完善，作答者的工作倦怠程度甚至不存在性別差異。

然而，對弱勢身分作答者而言，工作分配程序無法降低工作倦怠。這個調查結果讓我們相當驚訝，也產生更多疑問：正式的工作均分程序是否仍會造成工作分配不平等？實現「工作量平等」時是否會考量干擾或重要性等工作性質？除了職務範圍內的工作之外，還有哪些因素導致這個族群出現工作倦怠？我們該如何減緩這些因素的影響？



# 培養有歸屬感的職場文化

作者：Jeffrey Winer 博士，  
波士頓兒童醫院主治心理師，哈佛醫學院助理教授

## 重要研究結果與相關脈絡

這份報告的重要研究結果之一，是弱勢族群員工比其他同事更容易出現工作倦怠，前文也已經探討過可能原因。在本節中，我們將嘗試從更廣泛角度來解讀這些研究結果，一探歸屬感和相關的機構實踐策略。

當員工認為自己在團體中屬於弱勢族群，可能意味著他們容易受到「歸屬感不確定性」的影響<sup>7</sup>。這是一種廣為人知的心理現象（參見《Understanding and Overcoming Belonging Uncertainty》（瞭解並克服歸屬感不確定性），作者為 Geoffrey Cohen 博士）<sup>8</sup>，會讓員工懷疑「我屬於這裡嗎？」、「像我這樣的人能在這裡獲得成功嗎？」等等。而且隨著員工持續經歷或解讀這類經驗，可能會一再強化或重新看待這種心理狀態。歸屬感不確定性即是由此產生，這也或許解釋了為何具弱勢身分的員工出現更明顯的工作倦怠。

## 機構可以採取什麼行動？

機構必須瞭解，多元、包容、平等與歸屬感是不同的概念，需要採取個別對應但彼此關聯的策略並持之以恆，才能達成這些目標。想成功營造歸屬感，機構必須給予真心且堅定的承諾。

如果員工陷入困境，機構首先不該問：「這名員工發生了什麼事？」，而是該問：「為什麼他會有這種感受？公司哪部分的組織架構造就這種現象，例如什麼要素讓他抱持或加深這種負面感受？」

找出問題後，就要採取雙管齊下的解決措施，從機構層面進行改革，同時為員工提供支持。若機構能秉持堅持不懈的心態，支持員工並改革公司背後的治理系統，就能直接改變體制本身，使影響力遍及機構上下，而不只是單純為幾位員工帶來改變。這種創造性是機構營造歸屬感的動力，「努力」更是成功與否的關鍵。歸屬感的形成需要長時間的感受和行動，而且永遠都有進步空間，可以說是增進職場健康和工作效率的基礎。

機構可以運用許多現有工具提升歸屬感。例如在 2023 年的美國醫務總監報告中，研究人員探討了「孤獨感」這項主題，並指出增進社交關係和歸屬感是孤獨感和工作倦怠的重要解方<sup>9</sup>。

<sup>7</sup> Walton G.M. 和 Cohen G.L.,《A Brief Social-Belonging Intervention Improves Academic and Health Outcomes of Minority Students》(以簡短的社交歸屬感介入治療改善弱勢族群學生的學業和健康) Science (紐約:N.Y.) 2023年9月20日出版：<https://pubmed.ncbi.nlm.nih.gov/21415354/>

<sup>8</sup> <https://behavioralscientist.org/understanding-and-overcoming-belonging-uncertainty/>

<sup>9</sup> 美國衛生部助理國務卿辦公室 (OASH),「New Surgeon General Advisory Raises Alarm about the Devastating Impact of the Epidemic of Loneliness and Isolation in the United States」(新醫務總監警示：孤獨與疏離感已成美國流行病，恐造成破壞性影響) HHS.gov 2023年5月3日, <https://www.hhs.gov/about/news/2023/05/03/new-surgeon-general-advisory-raises-alarm-about-devastating-impact-epidemic-loneliness-isolation-united-states.html>

## 新進員工難以提升工作效率

新進員工(年資不到1年)的工作效率比資深員工(年資超過1年)低8%，這點並不令人意外。即使員工在某個職務上累積了豐富工作經驗，也很難一到職就做出貢獻，因為他們必須先大量學習新團隊特有的知識，才能順利上手。此外，組成團隊的要素並非只有技能和知識，還有歸屬感等涉及人際關係的元素。歸屬感能讓團隊成員覺得有所貢獻，這種心理安全感需要時間培養，也是提升工作效率的關鍵。

## 如何幫助新進員工成長？

我們認為機構可以透過以下三種方式支援新進員工：

- 提供優質說明文件。
- 將人工智慧融入工作流程：部分研究已指出，比起資深員工，這項技術對新員工更實用。
- 和新員工一起進辦公室工作：部分專家提到，在到職訓練階段採取這種做法，對新員工特別有幫助。



今年的研究結果指出，優質說明文件可大幅提升員工的工作效率(參見「[第4章：說明文件奠定成功基礎](#)」)，AI 則可小幅提升工作效率(參見「[第3章：技術能力對成效的助益](#)」)。我們相信這些做法對新進員工也有效。不過實際資料顯示，這些做法雖然對新進員工有益，但效果與其他人差不多。換句話說，這些做法無法特別幫到新進員工。

如果機構想支援新進員工和其他人員，可以從編製優質說明文件著手，因為證據顯示這確實能大幅提升工作效率。值得注意的是，比起提供品質不佳的說明文件(比平均低1個標準差)，如果團隊提供清楚詳盡的說明文件(比平均高出1個標準差)，新進員工的工作效率將提升130%。

接下來，我們將探討「短暫重返辦公室」為何能防止工作效率降低。在「[第7章：營造健康的文化才能推動革新](#)」中，我們討論了彈性工作模式的重要性。雖然今年的研究資料並非來自實驗，且我們已盡可能控制會使結果出現偏誤的因素，但工作模式的優點涉及眾多社會層面，且本質相當複雜，因此難以得出具體結論(閱讀相關研究或思維文章時，亦需注意這點)。我們可以從資料中清楚看出，彈性工作模式有助於提升工作效率，但不清楚工作地點是否也有同樣效果。

對新進員工而言也是如此，我們尚無證據可證明，與新進員工一起在公司工作能帶給他們特別的助益。如果您想提升工作效率，更妥當的做法應該是讓新進員工在工作方式、地點和時間上擁有彈性，而不是強制他們進公司上班。機構不會、也不該單純為了工作效率而改善體制。我們也認為，工作效率的真諦應該是創造價值，而不只是高效產出程式碼、減少倦怠或消除手動作業。



## 後記

# 結論

感謝各位參與今年的研究及閱讀這份報告。我們始終都在尋求更完善的研究方法，以探究團隊工作方式與目標成果之間的關聯性。

多年來的 DORA 研究帶給我們一項重要啟示：只要秉持精益求精的思維，採取行動持續改善，團隊將能達到最佳成果。

機構或團隊可將我們探討的各項能力，做為改善業務成果的著手點。其中部分目標可憑員工的力量達成，其他則需要整個團隊共同努力實現。第一步是確定機構需要調整的能力，接著投入資源和心力推動變革。

改善永遠沒有盡頭，但可以協助員工、團隊和機構取得長期成功。推動改善，是主管和從業人員共同肩負的責任。

## 如何將研究結果化為行動？

您可以深入瞭解如何運用這些研究結果來改善機構、團隊運作情形，以及提供更優質的服務給顧客。

歡迎加入 DORA 社群 (<https://dora.community>)，與其他成員分享您的經驗並互相學習，從他們的持續改善計畫中汲取靈感。

# 特別銘謝

有賴全球眾多熱情貢獻者的支持，Google 才能年年發布這項研究報告。多虧同仁在各個製作階段的鼎力協助，我們才能完成問卷調查題目設計、本地化、數據分析、撰文、編輯和排版。全體作者由衷感謝所有參與者對報告的付出、指導和熱心協助。

## 貢獻者

核心團隊	編輯		
James Brookbank	Mandy Grover	Yuki Iwanari	Rob Edwards
Kim Castillo	Jay Hauser	Vincent Jobard	Dave Farley
Derek DeBellis	Stan McKenzie	Gustavo Lapa	Steve Fenton
Nathen Harvey	Anna Eames Mikkawi	Mauricio Meléndez	Nicole Forsgren 博士
Michelle Irvine	Mike Pope	Jeremie Patonnier	Aaron Gillies
Amanda Lewis	Tabitha Smith	Miguel Reyes	Denali Lumma
Eric Maxwell	Olinda Turner	Pedro Sousa	Emerson Murphy-Hill
Steve McGhee	問卷調查本地化	Laurent Tardif	Harini Sampath
Dave Stanke		Kimmy Wu	Robin Savinar
Kevin Storer	Daniel Amadei	Vinicius Xavier	Dustin Smith
Daniella Villalba	Kuma Arakawa	Yoshi Yamaguchi	Jess Tsimeris
Brenna Washington	William Bartlett	顧問和領域專家	Laurie Weingart 博士
	Antonio Guzmán		Betsalel (Saul)
	Shogo Hamada	Jared Bhatti	Williamson
		Lisa Crispin	Jeffrey Winer 博士

## 贊助者



# 作者



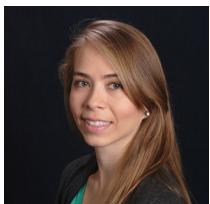
**Derek DeBellis**

Derek 是 Google 使用者體驗量化研究員和 DORA 首席研究員，主要負責問卷調查研究、記錄檔分析，以及找出概念評估方法來檢驗產品或功能是否可創造資本價值。Derek 在過去發表的研究中探討了真人與 AI 之間的互動、COVID-19 確診對戒菸的影響、自然語言處理錯誤相關設計、使用者體驗在隱私權討論中的角色、團隊文化，以及 AI 與員工身心健康和工作效率之間的關係。目前他在進行一項業餘研究，探討如何模擬信仰和權力的傳播過程。



**Amanda Lewis**

Amanda Lewis 是 DORA.community 開發主管，以及 Google Cloud DORA 推廣團隊的開發人員關係工程師。她的職責是協助開發人員、運作人員、產品經理、專案管理團隊和主管建立關係。Amanda 曾任職於電子商務平台、內容管理系統、觀測工具的開發團隊，以及開發人員支援團隊，致力透過交流互動和對談來滿足顧客的需求，創造更好的業務成果。她在工作中充分運用經驗及發揮同理心，協助團隊認識及導入軟體推送及可靠性做法。



**Daniella Villalba**

Daniella Villalba 是 Google 使用者體驗研究員，專門透過問卷調查研究，探索哪些因素能提升開發人員工作效率和滿意度。加入 Google 前，Daniella 曾研究冥想訓練的優點，以及影響大學生體驗的社會心理因素。她在佛羅里達國際大學取得實驗心理學博士學位。



### Dave Farley

Dave Farley 是 Continuous Delivery Ltd. 的創辦人暨總經理，也是《Modern Software Engineering》(現代軟體工程) 的作者和暢銷書《持續推送軟體更新》的共同作者。此外，他也是《Reactive Manifesto》(響應式宣言) 的作者之一，並以 LMAX Disruptor 開放原始碼專案榮獲 Duke Award。Dave 不僅是持續推送軟體更新 (CD) 的先驅，也是 CD、開發運作、測試驅動開發 (TDD) 和軟體設計的思維領袖和專家級從業人員，在建立高績效團隊、塑造成功機構及打造出色軟體方面都擁有豐富經驗。Dave 目前積極與全球軟體開發人員分享個人經驗和技能，協助對方改善軟體的設計、品質和可靠性。他開設了諮詢服務<sup>1</sup>、YouTube 頻道<sup>2</sup>，以及多項訓練課程來分享自身專業知識。



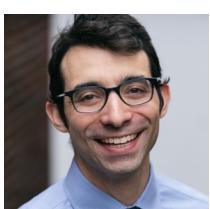
### Eric Maxwell

Eric Maxwell 負責領導 Google 的開發運作轉型業務，提供建議讓這家全球最優秀的公司持續進步，更快為使用者創造價值。Eric 以第一線工程師身分度過了前半段職涯，當時他建立了自動化程序來處理所有作業，且設身處地為其他從業人員著想。Eric 共同制定了 Google Cloud 應用程式翻新計畫 (CAMP)，同時也是 DORA 團隊成員。在加入 Google 前，Eric 任職於 Chef Software 並與同事有一段愉快的合作時光。



### James Brookbank

James Brookbank 是 Google 的雲端解決方案架構師，專門協助 Google Cloud 客戶解決複雜的技術問題，並提供專家級架構指引。加入 Google 前，James 任職於多家大型企業，負責 IT 基礎架構和金融服務相關業務。



### Jeffrey P. Winer 博士

Jeffrey P. Winer 博士是波士頓兒童醫院創傷與社區復原力中心 (TCRC) 的主治心理師、行為健康系統顧問和社會心理療法開發人員，以及哈佛醫學院助理教授。他主要與 TCRC 的同仁共同建立、測試、推廣及實施社會心理介入療法，以考量患者文化背景的方式，為具難民和移民身分的青少年和家庭治療創傷。他是《Mental Health Practice with Immigrant and Refugee Youth: A Socioecological Framework》(青少年移民和難民適用的心理健康做法：社會心理架構) 一書的共同作者<sup>3</sup>，也曾擔任美國和加拿大的心理健康計畫顧問。目前，他協助開發或採用的社會心理預防和介入療法，在世界各地獲得廣泛使用。詳情請見：<http://www.drjeffwiner.com>。

<sup>1</sup> <https://continuous-delivery.co.uk/engineering-for-software>

<sup>2</sup> <https://www.youtube.com/@ContinuousDelivery>

<sup>3</sup> <https://www.apa.org/pubs/books/4317536>



### **Kevin Storer**

Kevin M. Storer 是 Google 使用者體驗研究員，他領導的研究專門探討軟體開發團隊如何與開發運作工具互動，以及透過這些工具進行互動。加入 Google 前，Kevin 在加州大學爾灣分校取得資訊學博士學位。在公部門和民間企業任職期間，他撰寫了許多深具影響力的出版品，主題涵蓋以人為本的程式設計、開發人員體驗、資訊行為、資訊易取得性和普及運算。



### **Kim Castillo**

Kim Castillo 是 Google 使用者體驗專案經理，負責領導 DORA 計畫背後的跨部門作業，從監督研究運作情形，到出版這份報告都涵蓋在內。Kim 也研究 Google Cloud Duet AI 的使用者體驗。加入 Google 前，她曾從事軟體推送、技術計畫管理和敏捷開發輔導工作。Kim 主要透過社會心理研究，探討其家鄉菲律賓的法外處決、都市貧民窟發展，以及社區創傷和復原力。她將自己對軟體推送做法、心理安全感和青色組織的研究熱情都投入 DORA。



### **Michelle Irvine**

Michelle Irvine 是 Google 技術文件撰稿者，專門領導技術文件製作和影響的相關研究。加入 Google 前，她曾在教育出版業任職，以及擔任物理模擬軟體的技術文件撰稿者。Michelle 擁有滑鐵盧大學的物理學士學位，以及修辭與傳播設計碩士學位。



### **Nathen Harvey**

Nathen Harvey 是 Google Cloud 開發人員關係工程經理，負責領導 DORA 推廣團隊。Nathen 曾有幸與一些頂尖團隊和開放原始碼社群共事，協助他們運用開發運作和 SRE 原則和做法。他是過去三年的《加速發展：開發運作現狀》報告共同作者，也曾參與編輯《97 Things Every Cloud Engineer Should Know》(所有雲端工程師都應知道的 97 件事)。



### **Steve McGhee**

Steve McGhee 是可靠性做法倡議者，主要教導團隊以最適當的方式建構及執行可靠的世界級服務。負責這項職務前，他已擔任 Google 網站可靠性工程師超過 10 年，熟知如何擴充 Google 搜尋、YouTube、Android 和 Google Cloud 等全球系統的規模。他帶過加州、日本和英國的多個工程團隊，也曾與一間加州企業合作，協助他們遷移至雲端。

# 研究方法

本章說明我們如何將一系列初步構想轉化成較完整的概念，並撰寫成您眼前的報告，希望能解答您對報告編製方式的諸多疑問，甚至是幫助您規劃及著手進行自己的研究。

## 步驟 1：訂出一組我們認為高績效技術導向機構會重視的成果

定義成果是本研究的關鍵，因為 DORA 計畫的宗旨就是引導機構、團隊和員工取得他們重視的成果。如果不瞭解員工和機構的目標，研究可能從一開始就會走錯方向。該如何確定他們重視的成果？我們同時採用多種方法，包括進行質性研究（也就是詢問作答者、所屬團隊和機構想達成的目標）、發送問卷調查、與廣大社群互動，以及舉辦多場研討會。最終，我們從收到的回答歸納出以下主要成果：

- **機構績效**：機構不能只創造收益，還需要為消費者和廣大社群創造價值。
- **團隊績效**：應用程式/服務團隊是否能創造價值、創新及合作。
- **員工身心健康**：機構/團隊應採取對員工有利的策略，以減少工作倦怠、提高工作滿意度及協助員工締造高價值成果（即提升工作效率）。

此外，我們發現作答者也相當重視以下目標：

- **軟體推送成效**：團隊能快速有效率地部署軟體。
- **運作成效**：推送的軟體創造可靠的使用者體驗。



## 步驟 2：針對達到成果的方式、時間和原因提出假設

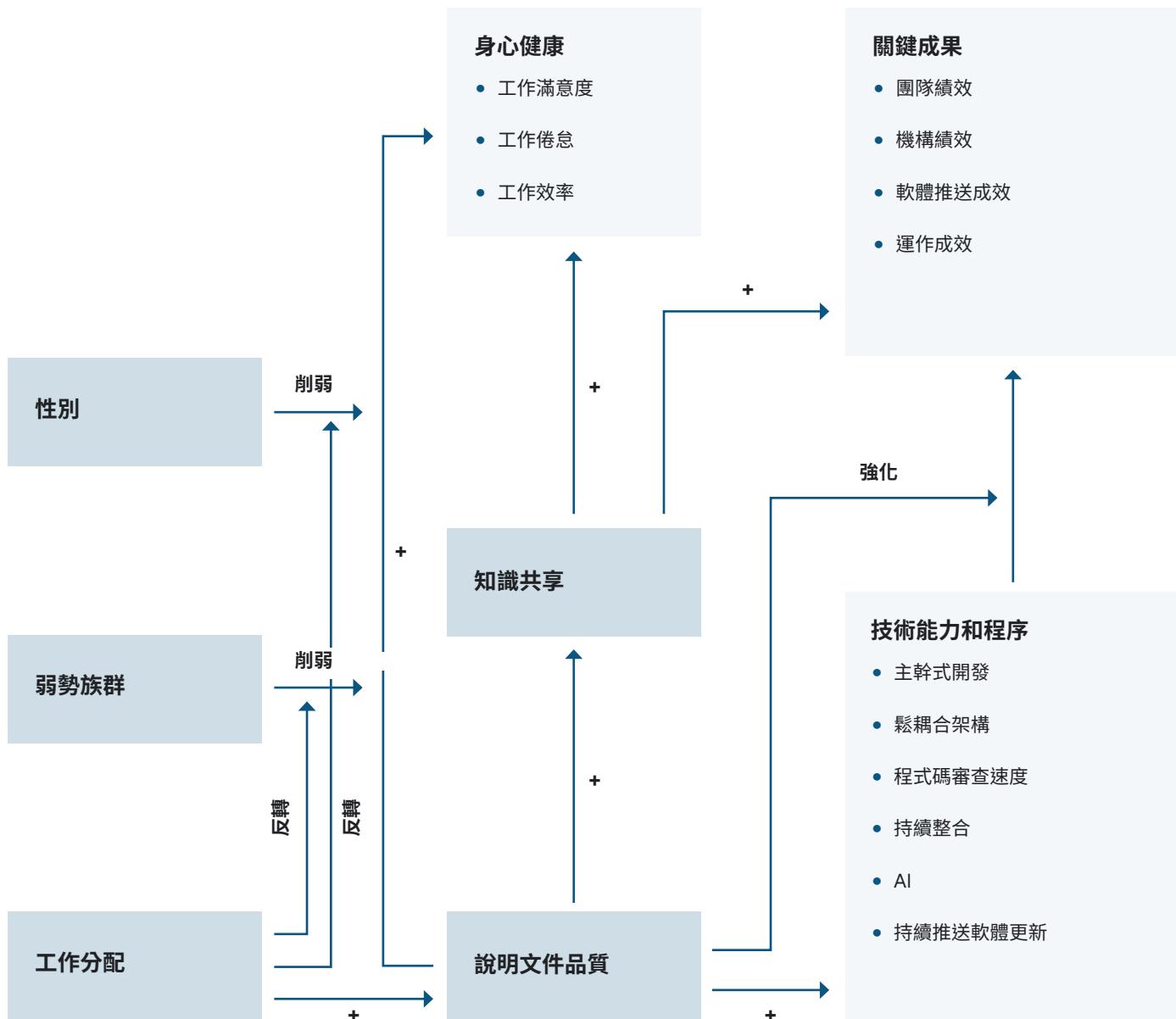
透過步驟 1 確定團隊重視的成果後，我們接著針對實踐方式提出假設，包括尋找會對成果造成實質影響的因素。我們的判斷原則是「在所有條件相等的情況下， $x$  會對  $y$  造成影響」。從業人員可以根據這個原則，透過資料決定要嘗試哪些類型的變革。

除了瞭解哪些方法具影響力，我們也研究哪些「條件」可能會強化或削弱影響力，也就是研究這些方法適用的「時機」和「對象」。例如，我們已證明說明文件品質平均而言可大幅降低工作倦怠，但這項結論不適用於具弱勢身分的作答者，說明文件品質上升反而會「加重」他們的工作倦怠。並非所有團隊或個人都落在平均值，因此瞭解這些方法在不同條件下的影響力非常重要。

此外，我們也針對原因或方式背後的「形成機制」，提出一套假設。我們以去年的研究結果及相關主題的大量文獻為基礎，並在今年的調查中假設弱勢員工有更明顯的工作倦怠。大家首先會問：「為什麼發生這種情況？」為回答這個問題，我們假設了幾種可能的機制，並逐一測試。例如，具弱勢身分的員工有更明顯的工作倦怠，是「因為」他們負責（或遭分配到）較費力的工作。



隨後，我們根據這些假設歸納出一套模型，以利設計問卷調查和執行分析。以下示例呈現說明文件該節的假設模型，其中完整涵蓋影響、條件和機制之間的關係：

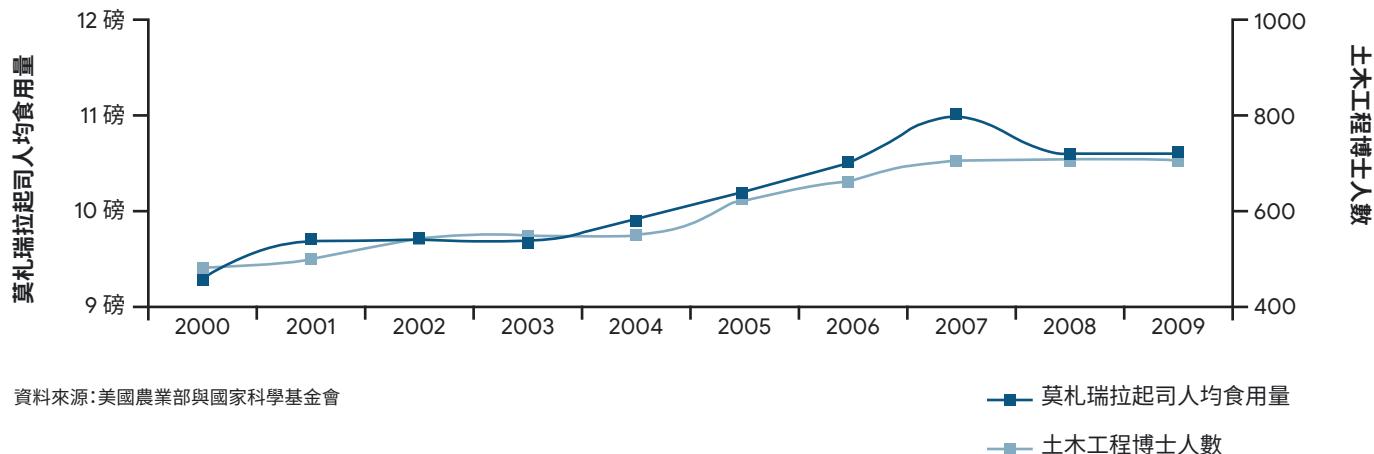


## 步驟 3：假設潛在干擾因素

若您曾分析過資料，可能對「偽相關」這個現象不陌生，或曾在某些網站中看過許多偽相關<sup>1</sup>的例子，如下所示：

莫札瑞拉起司的人均食用量，與取得土木工程博士學位的人數具相關性

相關性：95.86% ( $r=0.958648$ )

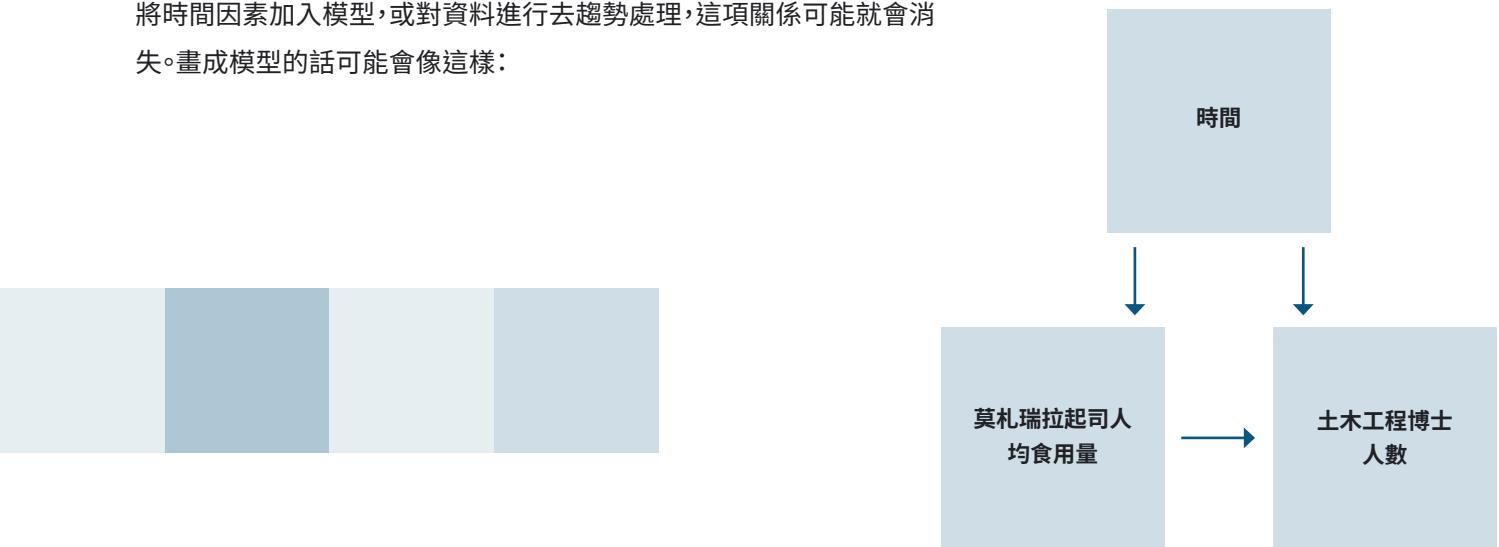


資料來源：美國農業部與國家科學基金會

■ 莫札瑞拉起司人均食用量  
■ 土木工程博士人數

土木工程博士人數與莫札瑞拉起司人均食用量之間不可能有任何因果關係。然而，這段關係背後潛藏一個干擾因素：時間。如果莫札瑞拉起司人均食用量和土木工程博士人數在同一段時間內都呈現正成長，這兩者之間就可能屬於正相關。

將時間因素加入模型，或對資料進行去趨勢處理，這項關係可能就會消失。畫成模型的話可能會像這樣：



<sup>1</sup> <https://www.tylervigen.com/spurious-correlations>

如果沒有留意時間(第三個變數)的干擾，解讀資料時便可能會得出「莫札瑞拉起司人均食用量與土木工程博士人數之間有關係」的結論，但這屬於偽相關。

研究人員可以運用工具判斷偽相關，例如 Dagitty (<https://dagitty.net/dags.html>) 就能幫助我們詳細定義因果模型。我們可以運用 Dagitty 瞭解模型的意義、需要及「不該」考量的因素，藉此正確評估 X 對 Y 的影響。Dagitty 這類工具帶給我們一項啟示：相關性不代表因果關係，但極有可能反映人們「思考」因果關係的方式。

研究人員並非全知全能，無法找出會造成估計偏誤的因素，這恐怕需要一份作答時間長達 50 小時的問卷調查才能做到。不過，我們還是盡力留意偏誤的成因，以準確估計各種行動、技術和架構如何影響您重視的成果。歸根究柢，許多從業人員想知道的就是哪些因素會影響這些關鍵成果。如果模型未將偏誤納入考量，便無法為從業人員提供所需指引。我們不希望最終告訴大家「莫札瑞拉起司能提升軟體推送成效」，但這種錯誤其實比想像中還容易發生。



## 步驟 4：編製問卷調查

編製問卷調查時應留意三個面向：定義評估方式、體驗和本地化。

### 定義評估方式

我們的目標是以可靠的方式，確實評估我們感興趣的概念。如何將抽象概念轉化為具體的評估指標，取決於「定義評估方式」的技巧。評估指標是所有後續分析的基礎，如果這些指標無法提供明確信號，我們該如何相信後續分析結果？又如何評估工作效率、工作倦怠或運作成效等等難以具體表述的概念？

首先，我們查閱文獻，看看是否已有經實證的評估指標。如果能在問卷調查中，納入過去通過驗證的評估指標，即可將問卷調查連結到與該問題相關的所有文獻。例如，我們一再使用的 Westrum《Typology of Organizational Cultures》(機構文化分類學)，即是過去經實證的評估指標。

然而在我們的研究領域中，許多概念仍未經過驗證。因此，我們以質性研究來釐清大眾如何理解這些概念，並查閱與概念細節相關的學術文獻。

### 問卷調查體驗

我們對問卷調查的要求是淺顯易懂、不應超過必要作答時間，且方便大多數人作答。這是相當困難的目標，因為作答者需要具備一定技術知識，且需要在有限時間內回答我們想知道的所有問題，而且業界對某些做法的稱呼可能各有不同。問卷調查採遠端執行且無人監管，確保資料收集成效超過一定門檻，我們也為此反覆多次調整。

### 本地化

我們每年都收到來自全球各地的問卷調查回覆。為了將調查範圍擴及更多目標對象，今年我們的問卷調查提供英文、西班牙文、法文、葡萄牙文和日文等本地化版本。這些本地化版本由優秀的 DORA 社群成員，帶領全球 Google 員工以及相關領域的合作夥伴共同翻譯而成，是集結群眾力量的結晶。我們由衷感謝 Zenika (<https://www.zenika.com>) 負責法文版翻譯。期望未來更多人加入這個行列，讓這份問卷調查成為真正的跨文化研究。

## 步驟 5：收集問卷調查回答

我們透過多種管道收集問卷調查回答，共可分為兩類：「自然募集」和「固定樣本」。

「自然募集」是指在所有可利用的社群媒體管道中，公開邀請大眾填寫問卷調查。其中包括發布網誌文章、進行電子郵件廣告活動，以及在社群媒體上發布貼文，而且我們也要求社群成員採取同樣做法（也就是「雪球式抽樣」）。

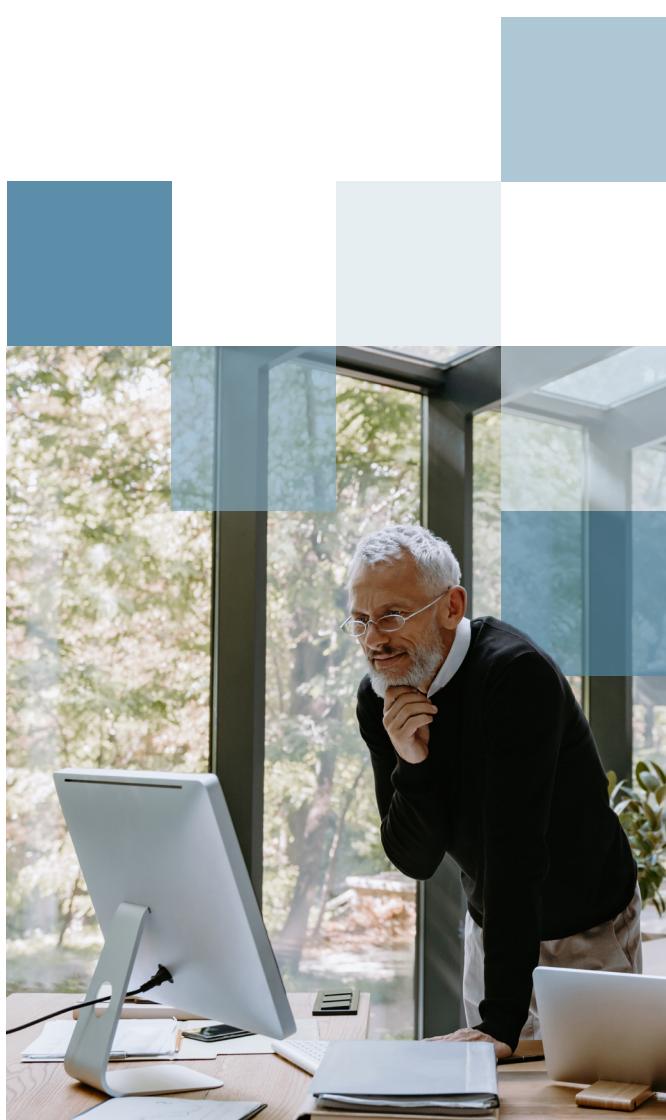
同時，我們運用「固定樣本」來彌補自然募集管道的不足之處。我們從廣大技術社群中，招募一般屬於弱勢族群的作答者，也嘗試向特定類型的產業和機構收集回答。簡單來說，由於自然募集管道無法限制對象，所以我們透過固定樣本補足所需的研究對象。此外，固定樣本管道能確保收集到足夠的作答者，因為我們不確定自然募集的回覆內容是否適用於本研究的分析類型。

## 步驟 6：分析資料

分析的三個核心步驟包括清理資料、驗證測量指標和評估模型。

### 清理資料

清理資料的目的是去除無效問卷及留下可用回覆。基於各種原因，部分問卷可能包含不乾淨的資料。從某些回答模式可以看出作答者其實未專心填寫、匆忙完成問卷調查或並未如實回答。這種情況鮮少出現在我們的資料中，但確實會發生。如果回答無法如實反映實際經驗，我們會將其視為雜訊並從分析範圍排除。消除雜訊的困難之處，在於確保我們不會因偏見或為了驗證假設，而排除特定信號。舉例來說，如果我們預設某項回答不可能超過或低於特定值，便可能會剔除該份問卷，因而使資料更印證我們的看法，提高假設成立的機率。



## 驗證測量指標

在這份報告的開頭，我們談到要評估的概念。描述這些概念的方式有很多種，其中一種做法是將概念評估指標稱為變數<sup>2</sup>，這些變數是我們模型和研究的構成要素。大致而言，我們可以從內部和外部來分析這些評估指標的效力。

為了瞭解評估指標的內部效力，我們會研究可代表某個概念的具體行為。例如，「優質說明文件」可能指的是用說明文件解決問題。

本研究大多數的變數是由多個指標構成，因為我們想探究的概念通常具有多種面向。為了瞭解變數的多面向性質，我們會測試可代表該構想的具體行為，以評估它們之間的共通程度。如果共通程度高（也就是共同變異性高），我們會假設這些行為背後有共同的概念，例如利益。

以幸福感這個多面向的概念為例。當一個人處於幸福的狀態時，我們預期他會有特定的情緒、行為和思考模式，也就是說，我們假設幸福是產生特定情緒、想法和行為的基礎。因此，我們預期幸福感會與特定幾種情緒、想法和行為同時出現。隨後，我們會詢問有關這些情緒、想法和行為的問題，並使用驗證性因素分析，來測試這些元素是否確實同時出現。

今年，我們使用 lavaan R 套件進行這項分析<sup>3</sup>。Lavaan 會傳回各種適合度統計資料，幫助我們瞭解作答者的回答模式是否確實反映某些概念。如果某個概念的指標不具共通性，表示我們尚未找到可靠的評估方法，因此可能需要修改或捨棄這個概念。

至於概念的外部效力，我們則是評估該概念在真實世界中的合理度。我們可能會預期某個概念與其他概念之間有特定關係，有時甚至呈現負相關，例如快樂與悲傷。如果我們測出來發現快樂與悲傷呈現正相關，便可能會懷疑評估方法或理論有誤。同樣地，我們可能預期兩個構想之間呈現正相關，但並非高度正相關。例如，工作效率和工作滿意度之間可能呈現正相關，但我們不認為這是兩種相等的概念。如果相關性過高，反而代表我們評估的可能是同一個概念，換句話說，我們的評估方法不足以準確區分這兩個概念之間的差異，或者現實中並沒有我們假設的差異。

<sup>2</sup> Moore, Will H. 和 David A. Siegel,《A Mathematics Course for Political and Social Research》(為政治與社會研究設計的數學課),Princeton University Press 出版,2013 年。

<sup>3</sup> Rosseel, Y.,《lavaan: An R Package for Structural Equation Modeling》(結構方程模型專用的 R 套件),Journal of Statistical Software,48(2),2012 年,1–36。  
<https://doi.org/10.18637/jss.v048.i02>

## 評估模型

在步驟 2 和步驟 3 中，我們建立了假設模型。完成步驟 6 後，我們得到了乾淨的資料，並瞭解模型與資料的適合度。今年，我們採用貝葉斯做法，來瞭解各種假設在指定資料下合理性，而不是在虛無假設（也就是無效果）下，指定資料的可能性。我們主要使用 R 套件工具 blavaan<sup>4</sup> 和 rstanarm<sup>5</sup> 來進行測試，以瞭解影響力顯著/可觀（而非輕微）的機率。我們以簡約法評估模型，也就是從非常簡單的模型開始，然後不斷提高複雜性，直到模型不再合理為止。例如，我們預測軟體推送和運作成效之間的交互作用，將影響機構機效。

最單純的模型不含交互作用：

機構績效 ~ 軟體推送成效 + 運作成效

第二個模型則加入交互作用：

機構績效 ~ 軟體推送成效 + 運作成效 + 軟體推送成效  $\times$  運作成效

我們根據《Regression and other stories》（迴歸分析與其他實例）<sup>6</sup>，以及《Statistical Rethinking》（統計反思）<sup>7</sup> 的建議，使用留一交叉驗證（LOOCV）和渡邊–赤池通用資訊量準則<sup>8</sup>，判斷是否要進一步提高複雜性。

<sup>4</sup> Merkle, Edgar C. 和 Yves Rosseel, «blavaan: Bayesian structural equation models via parameter expansion»(blavaan：透過擴展參數建立貝葉斯結構方程式模型), arXiv 預行刊印, 2015 年。

<https://doi.org/10.48550/arXiv.1511.05604>

<sup>5</sup> Goodrich, Ben、Jonah Gabry、Imad Ali 和 Sam Brilleman, «rstanarm: Bayesian applied regression modeling via Stan»(rstanarm：透過 Stan 將貝葉斯做法套用至迴歸模型), R 套件版本 2, no. 1 (2020 年)。

<sup>6</sup> Gelman, Andrew、Jennifer Hill 和 Aki Vehtari, «Regression and Other Stories»(迴歸分析與其他實例), Cambridge University Press 出版, 2020 年。

<sup>7</sup> McElreath, Richard, «Statistical rethinking: A Bayesian course with examples in R»(統計反思：R 與 Stan 中的貝葉斯做法教學與範例), Chapman and Hall/CRC 出版, 2018 年。

<sup>8</sup> Vehtari, Aki、Andrew Gelman 和 Jonah Gabry, «Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC»(使用留一交叉驗證和 WAIC 評估實用貝葉斯模型), Statistics and Computing, 27, 2017 年, 1413–1432。<https://doi.org/10.1007/s11222-016-9696-4>

## 步驟 7：將研究結果製成報告

接著，我們組成團隊審查這些研究結果。今年，我們在科羅拉多州的博爾德，花了幾天時間統整資料與主題專家的經驗，逐一檢驗本報告各章中的研究假設。解讀資料時，往往容易發生扭曲、臆測、過度關注個別事件，以及忽略特定要素等問題。為減少這類問題，我們邀集各領域的專家相聚討論，也鼓勵互相提問、拋出不同見解和獨特觀點及保持好奇心<sup>9</sup>。

取得結果後，作者群便各自返回母國開始撰寫報告，過程中也會諮詢編輯團隊和主題專家，他們的觀點有利於我們清楚傳達想法，因此不可或缺。資料分析人員則負責確認我們撰寫的內容並未偏離研究資料蘊含的意義。

最後，我們優秀的設計合作夥伴 BrightCarbon<sup>10</sup> 將這些章節集結成冊，並套用風格一致的美編設計。

## 步驟 8：整合研究結果和社群的看法

我們邀請社群成員思考如何解讀及運用這些研究結果。這份報告盡可能提供具體建議，但我們也瞭解業界團隊其實能透過無數種方式運用這些研究結果。舉例來說，我們的評估結果指出鬆耦合架構可帶來助益，但建立這個架構的方法不只一種。與社群共同思考及分享這些做法，才是持續改善的唯一途徑。本研究以概念化的方式解釋開發運作領域的現況，提供您、您的團隊和機構營運的方向。

歡迎前往 DORA 社群網站 (<https://dora.community>) 與全球社群一同交流改善做法。

<sup>9</sup> Stasser, G. 和 Titus, W. (1985 年),《Pooling of unshared information in group decision making: Biased information sampling during discussion》(匯集集體決策中未共享的資訊：討論過程中的偏誤資訊抽樣)，Journal of Personality and Social Psychology, 48(6), 1985 年, 1467–1478。<https://doi.org/10.1037/0022-3514.48.6.1467>

<sup>10</sup> <https://www.brightcarbon.com/>

# 作答者和企業統計結構

## 問卷調查對象

近十年來，DORA 研究計畫致力調查高績效、技術導向的機構具備哪些能力，以及採取什麼做法和措施。我們從各種規模和類型的產業收到超過 36,000 位專業人士的回覆，感謝各位與我們分享寶貴見解！今年，來自全球各行各業近 3,000 名專業人士分享了他們的經驗，協助我們進一步瞭解高績效的技術導向機構是如何誕生的。

今年的作答者和企業統計結構問題採用 Stack Overflow 進行的研究。在 2022 年，超過 70,000 名作答者參與了 Stack Overflow 開發人員問卷調查<sup>1</sup>。出於許多原因，該份調查並未涵蓋各行各業的技術從業人員，但已相當接近全球開發人員普查。得力於該調查的規模，我們能找出我們資料中的回覆偏誤，並瞭解本研究的解釋力可能有多廣。這些資料和 Stack Overflow 開發人員問卷調查中的作答

者和企業統計結構問題都經過精心設計，值得參考。相較於 Stack Overflow 開發人員問卷調查，我們的樣本集涵蓋更多女性和身心障礙作答者，以及任職於大型機構的作答者。在種族和族群方面，我們的樣本集與 Stack Overflow 類似。

今年自然募集的問卷調查作答人數為 2022 年的 3.6 倍。

今年自然募集的問卷調查  
作答人數為 2022 年的  
**3.6 倍**

<sup>1</sup> <https://survey.stackoverflow.co/2022#overview>

# 作答者統計結構

## 性別

與 2022 年 (18%) 相比，今年樣本中的女性作答者比例較低 (12%)。

性別	作答者百分比
不方便透露	3%
自述性別	2%
女性	12%
男性	81%

## 弱勢族群

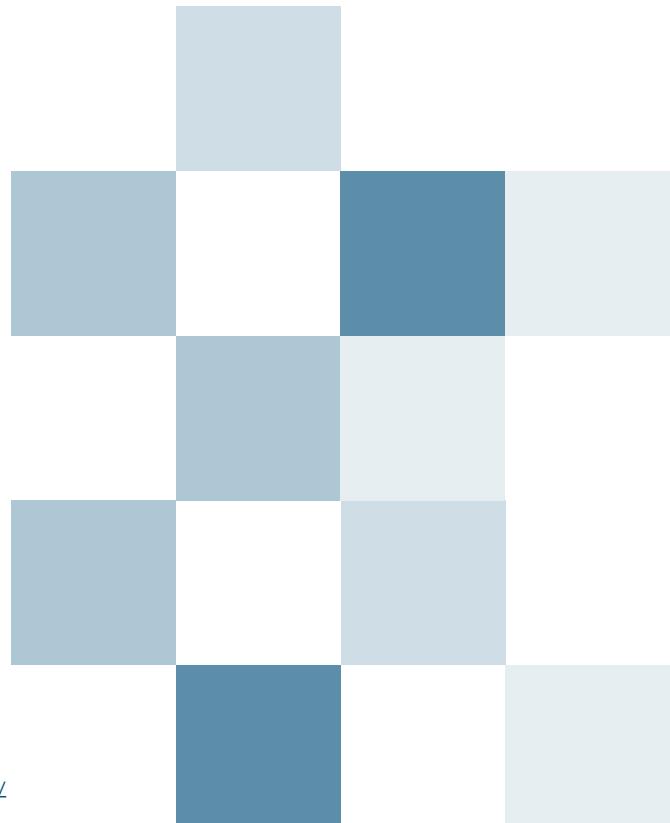
具有弱勢身分指的可能是種族、性別或其他屬性。我們已連續六年詢問有關弱勢族群的問題，具弱勢身分的作答者比例從 2022 年的 19% 小幅下降至今年的 15%。

弱勢族群	作答者百分比
否	77%
是	15%
不方便透露	7%

## 身心障礙

我們遵循《Washington Group Short Set》(華盛頓身心障礙小組簡短問題集)，整理出六個方面的身心障礙情形<sup>2</sup>。我們已連續五年詢問有關身心障礙的問題，今年的身心障礙作答者比例為 6%，低於 2022 年的 11%。

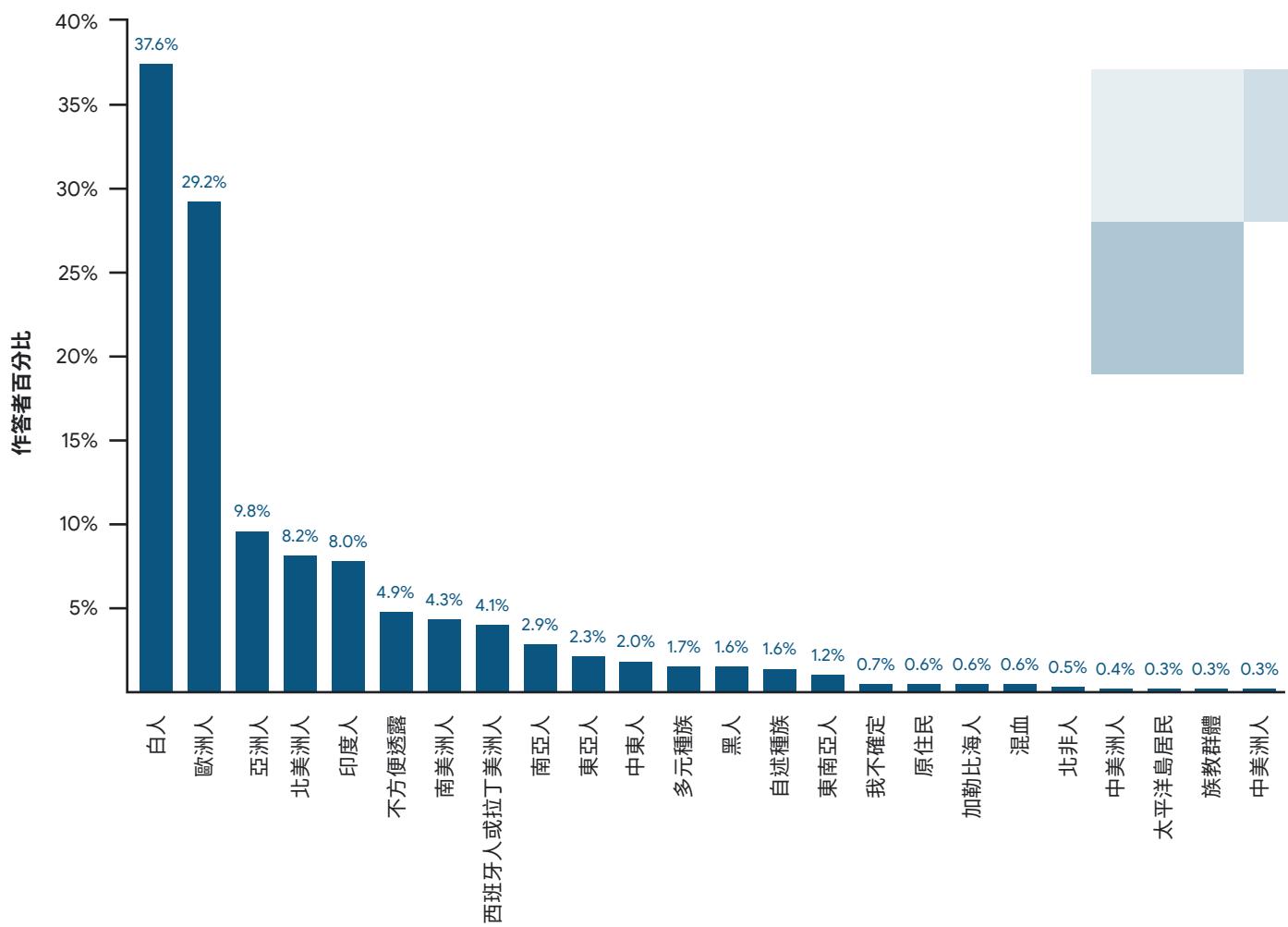
身心障礙	作答者百分比
非身心障礙者	87%
是	6%
不方便透露/未回答	7%



<sup>2</sup> <https://www.washingtongroup-disability.com/question-sets/wg-short-set-on-functioning-wg-ss/>

## 種族和族群

我們沿用了 2022 年 Stack Overflow 開發人員問卷調查的問題<sup>3</sup>。如先前所述，本研究的樣本集與 Stack Overflow 類似，但有一個明顯差異：我們資料中的歐洲作答者比例較低。

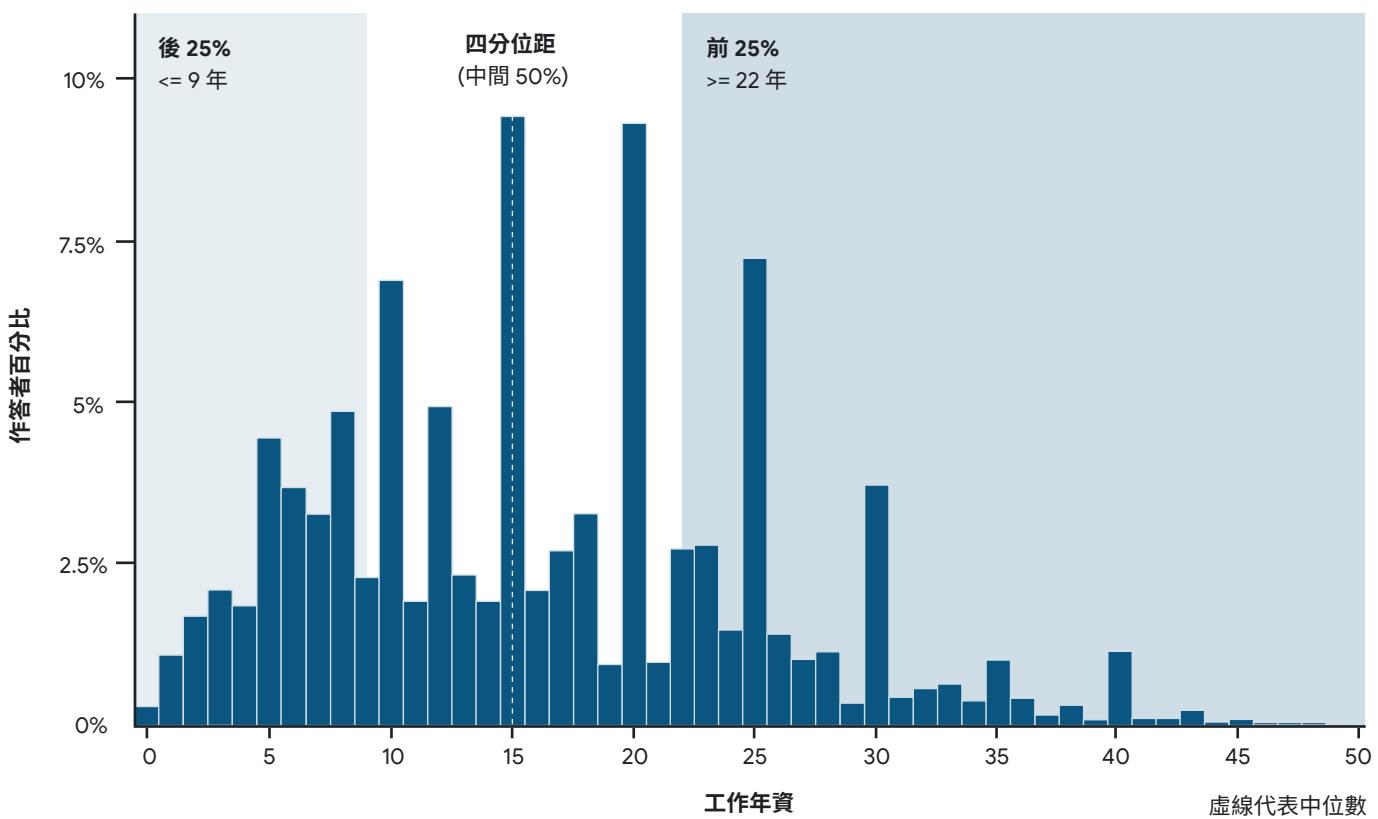


<sup>3</sup> <https://survey.stackoverflow.co/2022#overview>

# 年資經驗

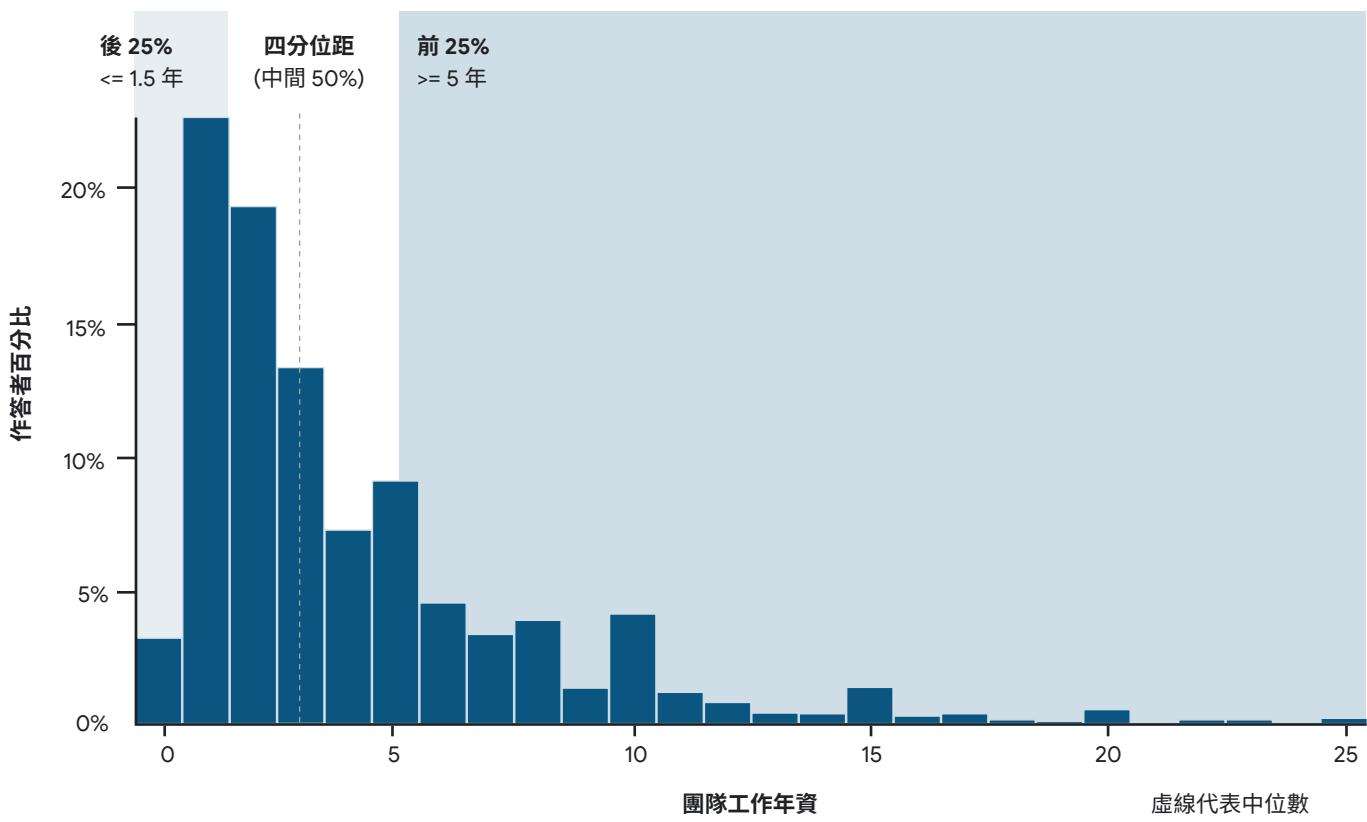
## 工作經驗

為瞭解作答者的年資，我們在問卷調查中詢問「您有幾年工作經驗？」總結來說，今年訪問到的幾乎都是年資很高的從業人員。其中，50% 作答者擁有超過 15 年工作經驗，25% 作答者擁有超過 22 年工作經驗，25% 作答者的工  
作經驗不到 9 年。不過事後來看，我們難以判斷作答者對「工作」的定義。



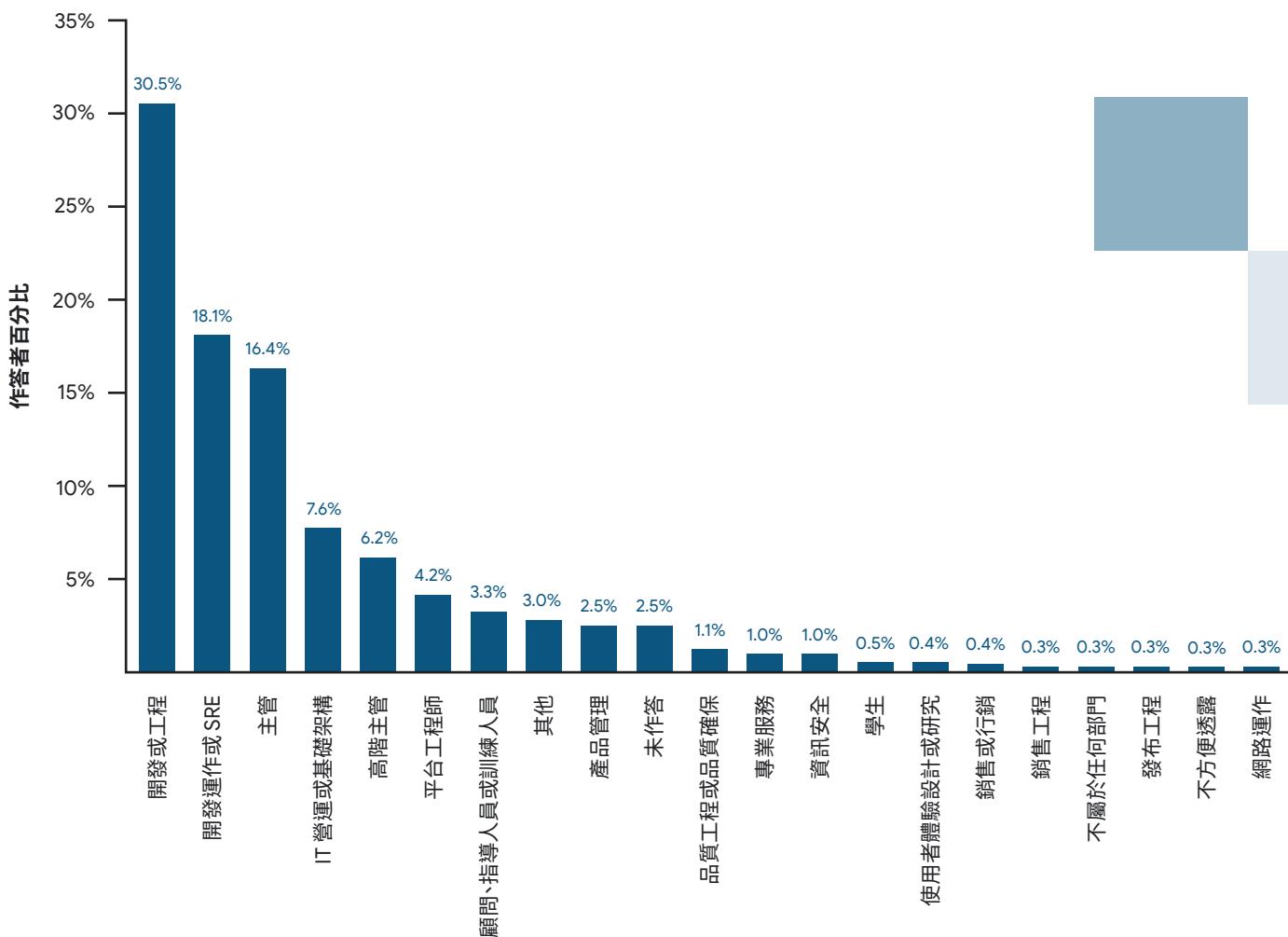
## 團隊工作經驗

有時，進到新團隊工作就像回到起點，重新開始累積經驗。為瞭解這項資訊，我們詢問作答者「您在目前所屬團隊已工作幾年？」許多作答者雖然工作年資很長，但在目前團隊中仍算是新人。其中，50% 作答者在目前團隊的工作年資不到 3 年。25% 作答者在目前團隊的工作年資低於或等於 1.5 年。僅 25% 作答者已在目前團隊工作 5 年以上。這些數據代表作答者擁有不斷成長的心態？還是代表經濟情勢動盪？



## 職務

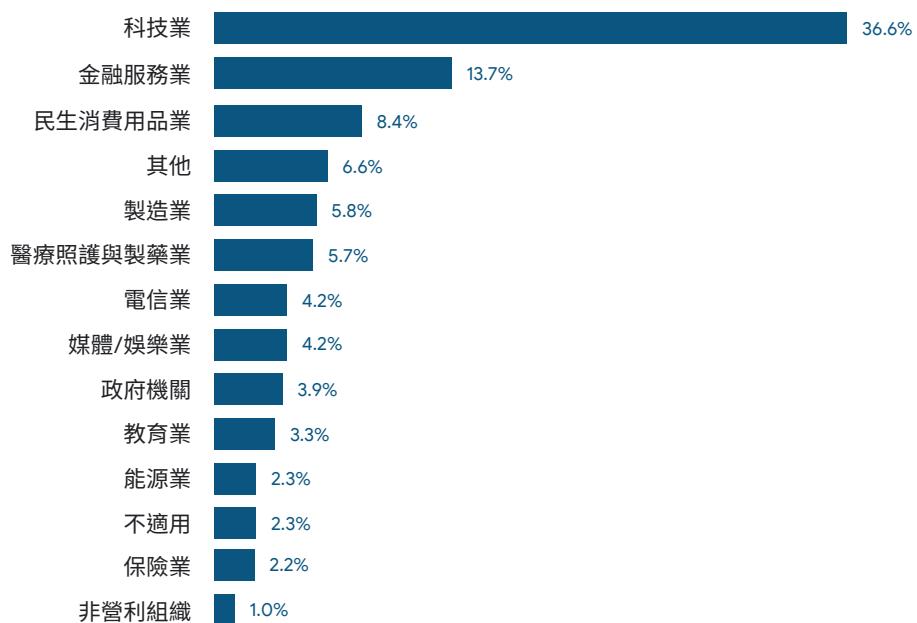
72% 的作答者分別任職於開發或工程團隊 (30%)、開發運作或 SRE 團隊 (18%)、IT 運作或基礎架構團隊 (8%)，或是擔任主管 (16%)。相較之下，在 2022 年，負責以上四個職務的人占全體作答者的 85%。由於今年較多作答者擔任其他職位，負責以上四個職務的作答者比例下降。另外，任職於 IT 運作或基礎架構團隊的作答者比例在 2022 年升至 19%，隨後在今年降至 8%，回到接近 2021 年的水準 (9%)。



# 企業統計結構

## 產業別

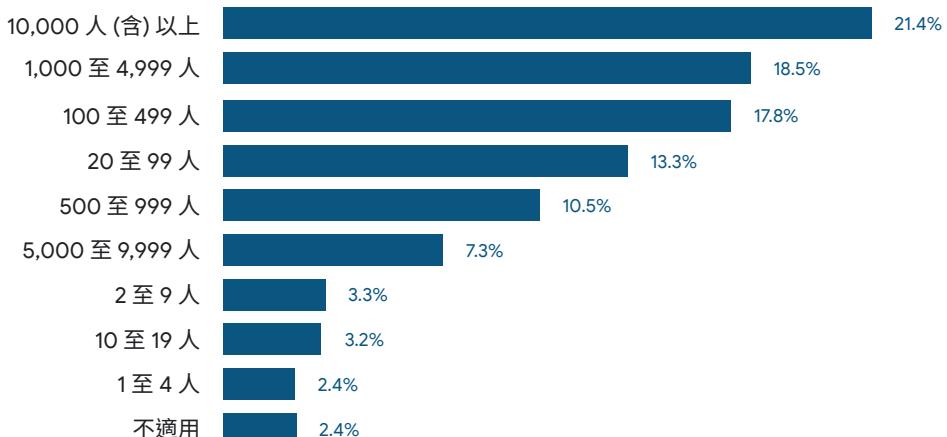
### 作答者百分比



## 員工人數

### 作答者百分比

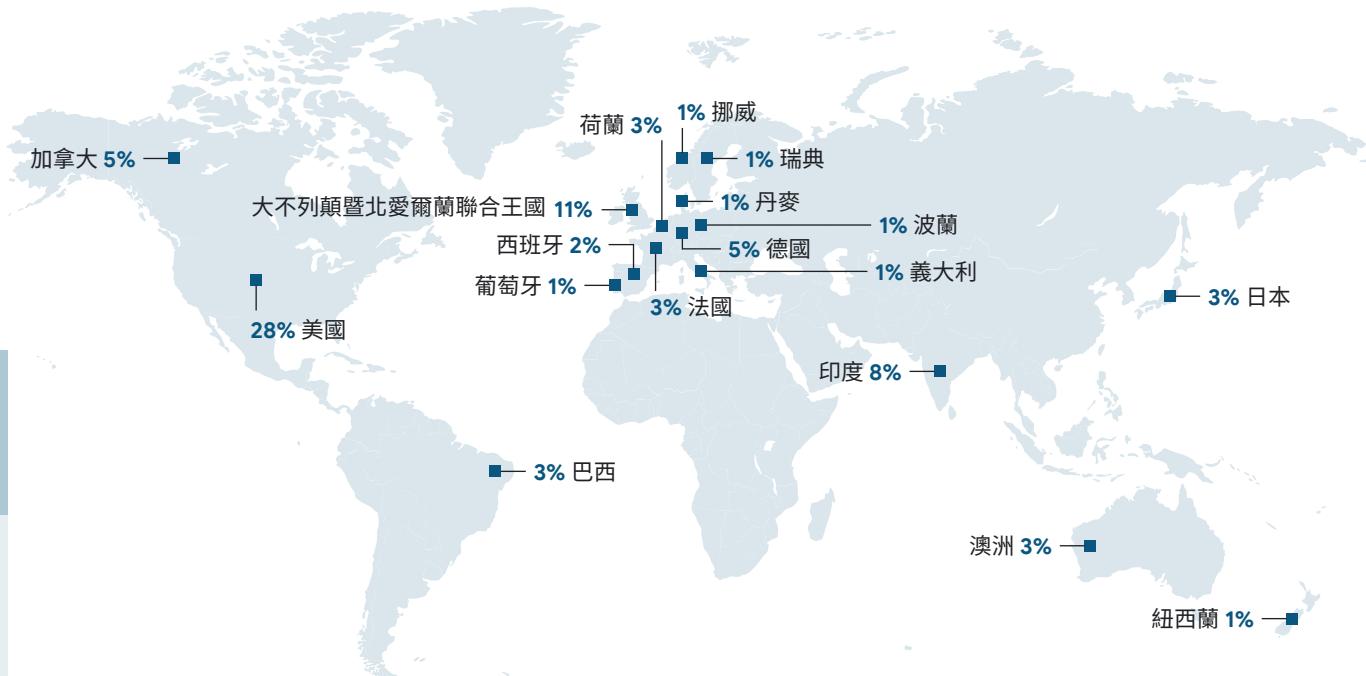
#### 貴機構有多少員工？



## 國家/地區

我們一向很開心收到全球從業人員的問卷調查回覆，

感謝各位鼎力相助！



## 國家/地區

美國	丹麥	立陶宛	突尼西亞	孟加拉	瓜地馬拉
英國	瑞士	泰國	烏拉圭	多明尼加共和國	宏都拉斯
印度	奧地利	匈牙利	阿富汗	迦納	拉脫維亞
加拿大	肯亞	以色列	阿爾及利亞	香港特別行政區	黎巴嫩
德國	南非	越南	埃及	哈薩克	盧森堡
澳洲	阿根廷	阿拉伯聯合大公國	愛沙尼亞	緬甸	馬爾地夫
巴西	捷克共和國	保加利亞	冰島	沙烏地阿拉伯	馬爾他
不適用	比利時	克羅埃西亞	伊朗	索馬利亞	模里西斯
荷蘭	哥倫比亞	厄瓜多	奈及利亞	蘇丹	蒙古
日本	芬蘭	印尼	秘魯	烏干達	摩洛哥
法國	愛爾蘭	菲律賓	斯洛伐克	阿爾巴尼亞	尼泊爾
西班牙	中國	亞美尼亞	斯洛維尼亞	巴哈馬	卡達
瑞典	羅馬尼亞	喬治亞	韓國	白俄羅斯	前南斯拉夫馬其頓共和國
義大利	新加坡	希臘	斯里蘭卡	玻利維亞	
紐西蘭	墨西哥	馬來西亞	安道爾	柬埔寨	千里達與托巴哥
波蘭	土耳其	巴基斯坦	安哥拉	哥斯大黎加	坦尚尼亞聯合共和國
挪威	烏克蘭	俄羅斯聯邦	安地卡及巴布達	吉布地	
葡萄牙	智利	塞爾維亞	巴林	薩爾瓦多	辛巴威

# 工作模式

## 聘僱身分

88% 的作答者為全職員工，10% 的作答者為約聘人員。部分約聘人員回答的工作體驗與全職員工截然不同。

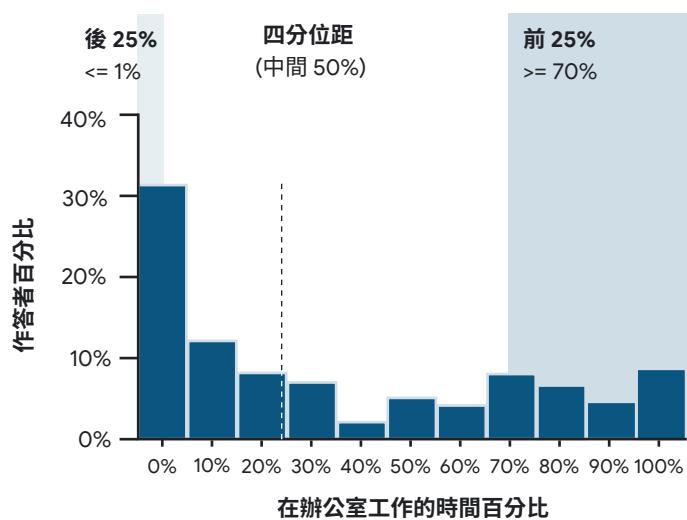
約聘人員	作答者百分比
全職員工	88%
全職約聘人員	8%
兼職員工	2%
兼職約聘人員	2%

\*以您負責的主要應用程式或服務而言，在擁有該應用程式/服務的機構中，您的聘僱身分屬於下列何者？

這項差異的原因可能是兩者的團隊融入程度不同。部分約聘人員表示自己完全融入工作團隊，他們每天與團隊成員密切合作，並認為自己幾乎等同於全職員工。70% 約聘員工對於「自己完全融入團隊」的說法表示同意或非常同意。

## 工作地點

今年的回答模式顯示，雖然企業逐漸要求員工重返辦公室，實際情況是許多員工仍繼續在家工作。將近 33% 的作答者幾乎完全在家工作（在辦公室的時間不到 5%），63% 作答者在家工作的時間多於在辦公室工作的時間，其餘作答者通常採取混合辦公，而 75% 作答者在辦公室工作的時間不到 70%。偏好在辦公室工作的作答者不多，只有 9% 作答者在辦公室工作的時間超過 95%。



# 模型

## 前言

傳統上，我們會建立一個巨大的模型，但基於以下原因，今年我們決定拆分成多個模型：

- 巨型模型很快會變得難以掌握。每加入一個變數，都會改變模型函數，因而導致預估失準且難以分析變化的原因除。
- 今年報告各章分別含有不同假設，因此各自建立相應模型也很合理。
- 在「預估 X 對 Y 的影響」方面，巨型模型的效益不明確。為瞭解 X 對 Y 的影響，我們選擇使用有向非循環圖來找出應該和不該納入模型的共變數。
- 我們今年提出了許多假設，如果以單一巨型模型進行研究，相當於將以下幾頁的圖表全都併成一張，導致讀者難以理解。



# 如何閱讀這些圖表？

瞭解圖表的解讀方式後，您將能有效率地掌握我們想傳達豐富資訊。

## 變數

變數是指我們嘗試評估的概念，例如說明文件品質。

## 變數類別

- 變數
- 變數
- 變數
- 變數

變數類別僅代表我們認為某些變數屬於同一類，但與分析本身無關。我們已透過統計方式，評估變數類別是否為概括性的概念。

+

正面影響，僅代表效果提升，但不一定是好事。

-

負面影響，僅代表效果下降，但不一定是壞事。



假想效果，但資料未能佐證。



中介途徑的一部分，已經過我們明確分析。

## 警告：以下模型僅呈現一般加總結果！

為了方便閱讀，我們將部分變數歸入同一類，以便用箭頭描述變數類別之間的單向或雙向關係，但這些關係不一定適用於同類別中的各個變數。舉例來說，知識共享對大多數關鍵成果有正面影響，因此我們畫了一個箭頭從「知識共享」指向「關鍵成果」變數類別的箭頭，並標上加號 (+)。不過，知識共享對軟體推送成效沒有正面影響。如要瞭解更多資訊，請參閱相關章節。

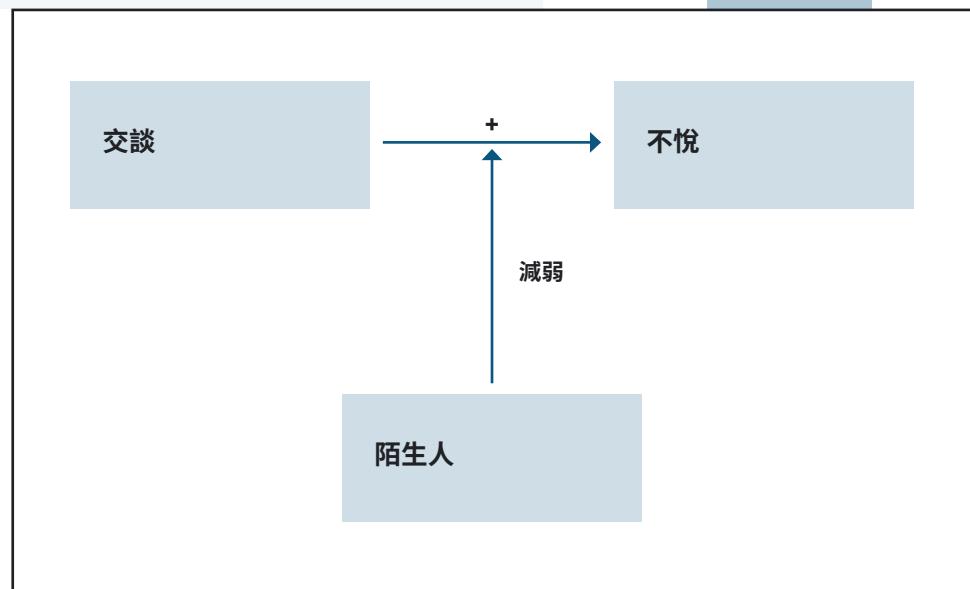
## 調節效果範例

在統計學中，「調節」是個較難掌握的概念，不過這其實類似於一般生活中常說的「看情況」。以下用一個簡短的例子，說明調節概念在這份報告中的意義。

在美國電視情境喜劇《人生如戲》(Curb Your Enthusiasm)第3季中，主角賴瑞大衛 (Larry David) 說：「我不喜歡跟我認識的人說話，跟陌生人說話倒是沒問題。」我們可以用下方簡短圖表呈現這句話：

# 「我不喜歡跟我認識的人說話，跟陌生人說話倒是沒問題。」

賴瑞大衛



從上方圖表可以看出，對賴瑞而言，「交談」會對「不悅」帶來正面影響。這裡的正面影響僅表示具提升效果，不一定是好事。我們畫了從「交談」指向「不悅」的黑色實線和箭頭，以表示這個正面影響。讀者可以從這個箭頭看出「交談造成不悅」的因果關係，並瞭解交談會導致賴瑞不悅。

第二個重點，是陌生人（以布林值「陌生人：是/否」代表）並未指向任何變數，而是指向影響，也就是箭頭。我們認為陌

生人調整的不是變數，而是影響力，因此箭頭是從陌生人指向其他箭頭，而非其他變數。這表示賴瑞是否與陌生人說話，會影響「交談造成不悅」的程度。換句話說，交談對不悅的影響，取決於賴瑞的交談對象是否為陌生人。如果對方為陌生人，賴瑞比較能接受交談所產生的影響，我們可以說陌生人減輕了賴瑞在交談時感受到的不悅。

以下是幾種不同的調節方式：

- **強化**:增強正面和負面影響。
- **削弱**:降低影響力。
- **減輕**:減緩正面和負面影響。
- **反轉**:將正面影響變成負面影響,將負面影響變成正面影響。
- **調整**:影響力有所變化,但變化模式無法簡短總結。這種情況通常是類別中的變數所導致。舉例來說,某些產業在不同條件(如經濟波動)下可能有不同行為。

### 中介效果範例

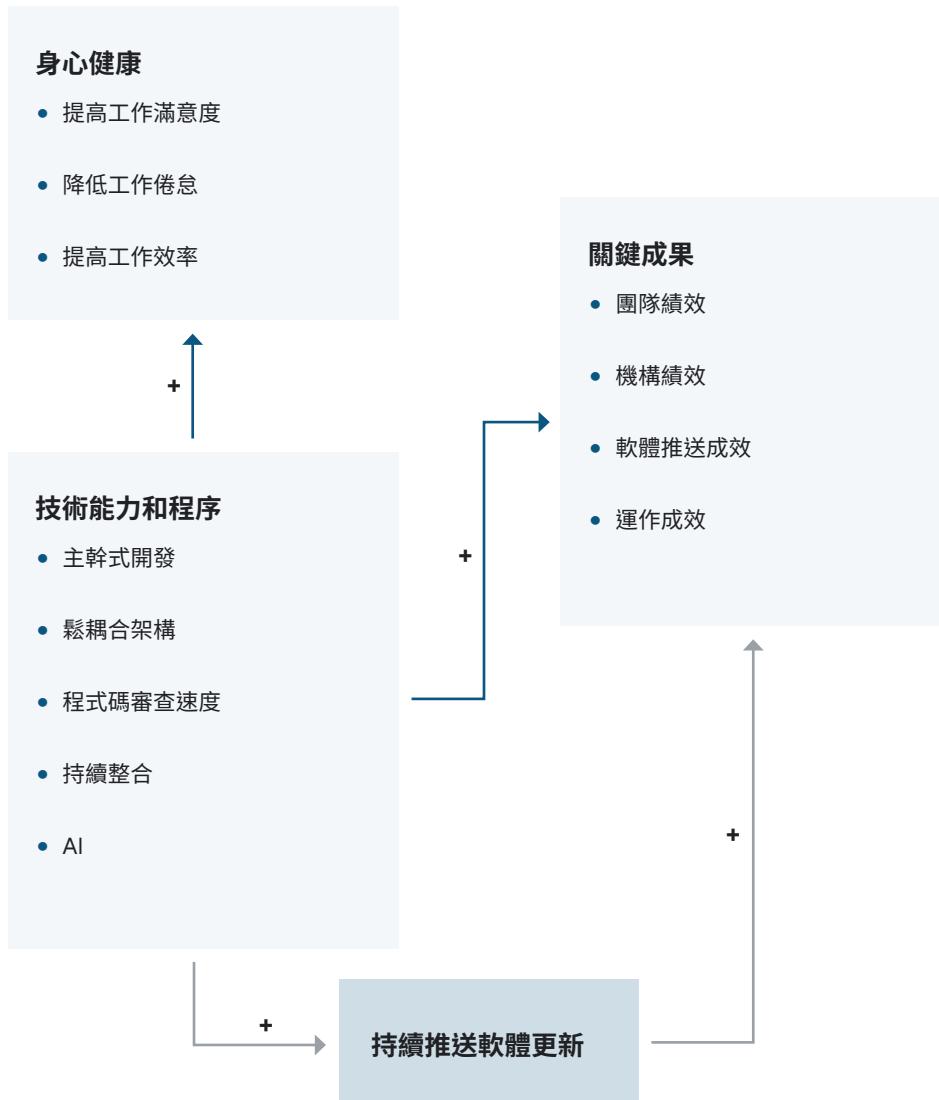
與調節效果相同,我們也深入研究了中介效果,也就是探究「原因」和「方式」,例如:為何照射太陽能使植物長高?為何草莓嚥起來是甜的?踩油門為何能使汽車移動?止痛藥如何減緩不適?統計學能幫助我們測試中介效果和做出結論<sup>1</sup>,像是「X 會對 Y 造成影響,部分或完全是因為 M 的關係」。例如,植物照射陽光後會長高,是因為行光合作用。



<sup>1</sup><https://lavaan.ugent.be/tutorial/mediation.html>

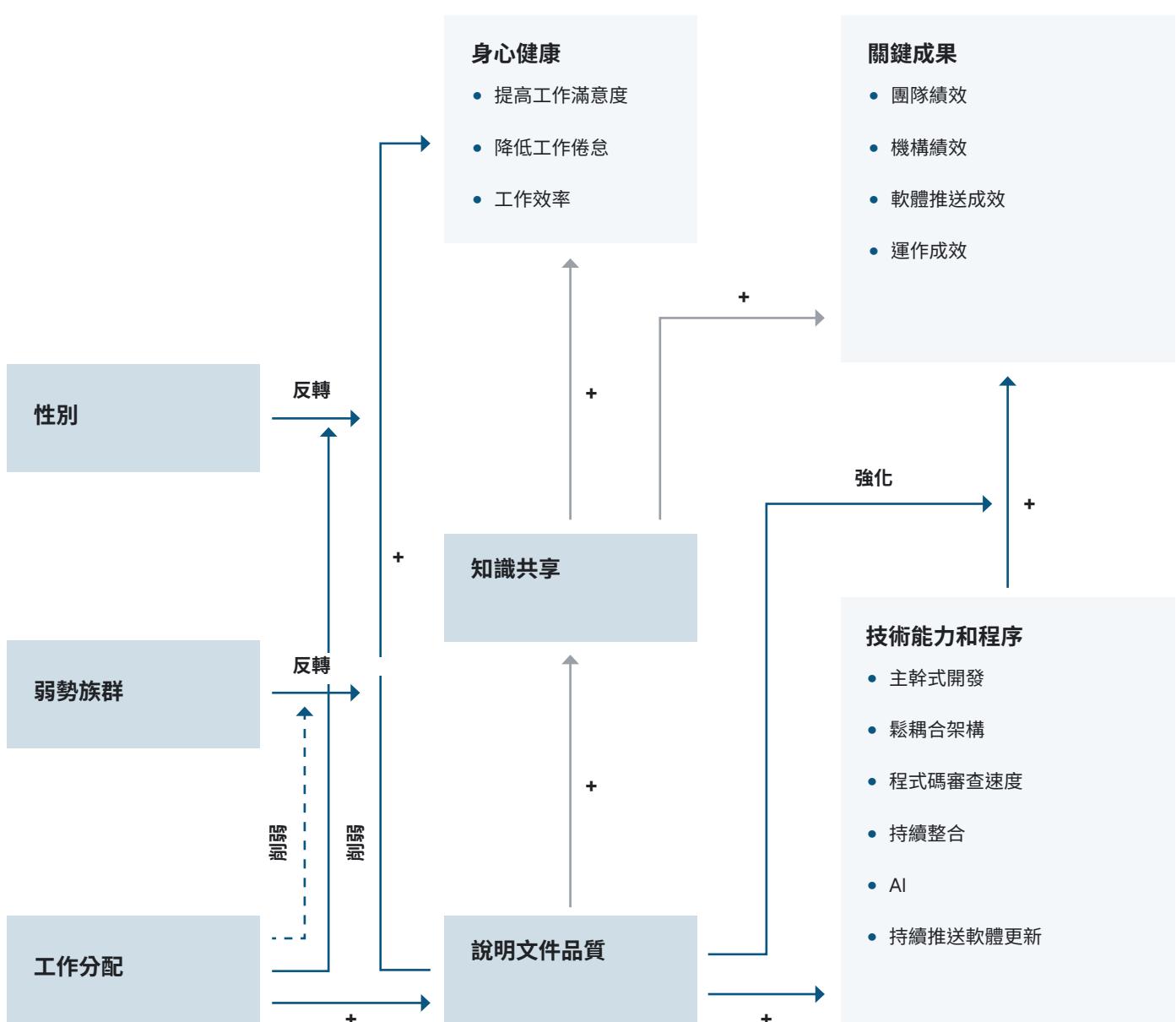
## 第3章模型：技術能力對成效的助益

技術能力和程序對身心健康和關鍵成果帶來正面影響，「持續推送軟體更新」在這段關係中發揮部分中介效果。換句話說，這些技術能力造就可持續推送軟體更新的作業環境，進而影響關鍵成果。



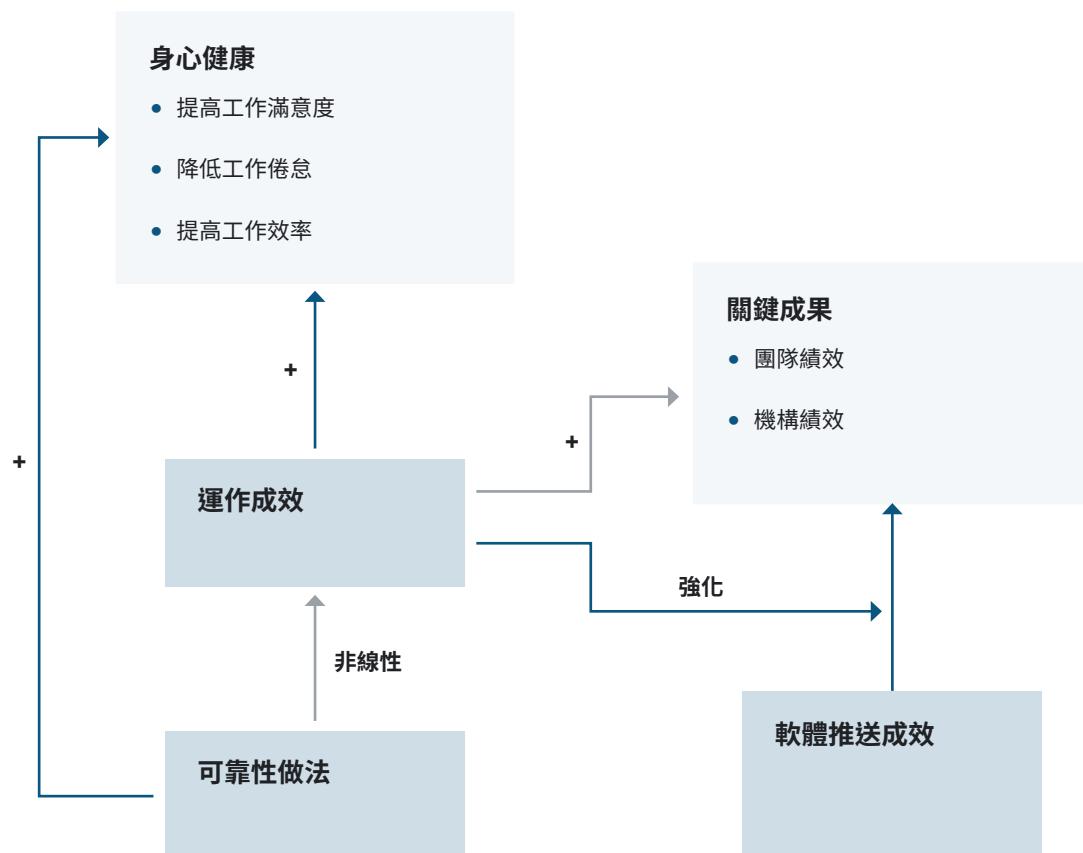
## 第4章模型：說明文件奠定成功基礎

簡單來說，優質說明文件對身心健康和關鍵成果帶來正面影響。這句話部分正確，因為優質說明文件能創造更容易共享知識的工作環境。此外，優質說明文件也能幫助團隊奠定技術能力和程序，進而強化技術能力和程序對關鍵成果的影響。最後，說明文件品質無法促進每位員工的身心健康。如要進一步瞭解這項複雜的研究結果，請回頭參閱相關章節。



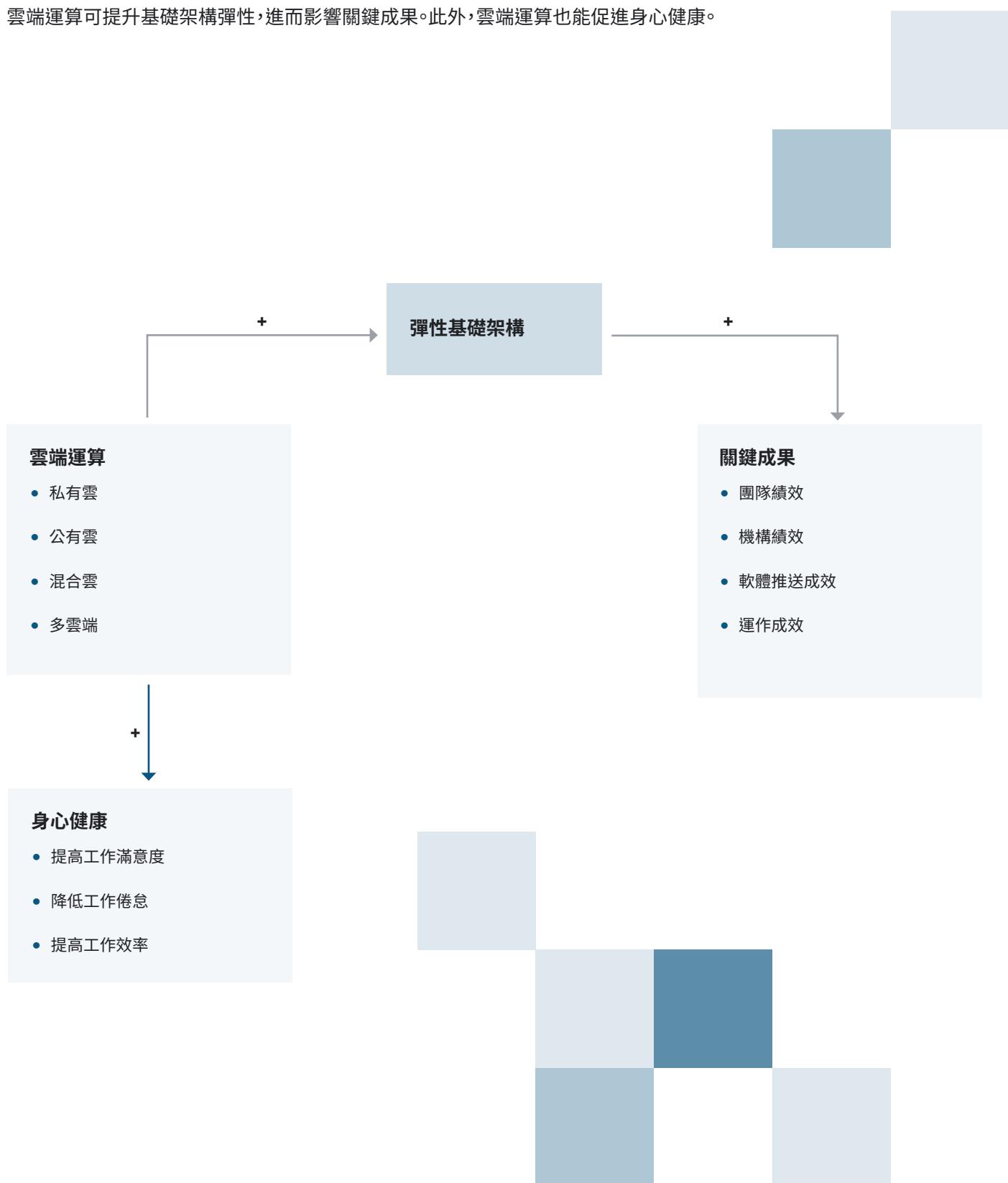
## 第5章模型：可靠性帶來更高成效

這一章探討運作成效的核心角色。運作成效不僅能改善身心健康和關鍵成果，也能強化軟體推送成效的影響力。可靠性做法與運作成效之間屬於非線性關係，如要瞭解細節，建議回頭參閱相關章節。



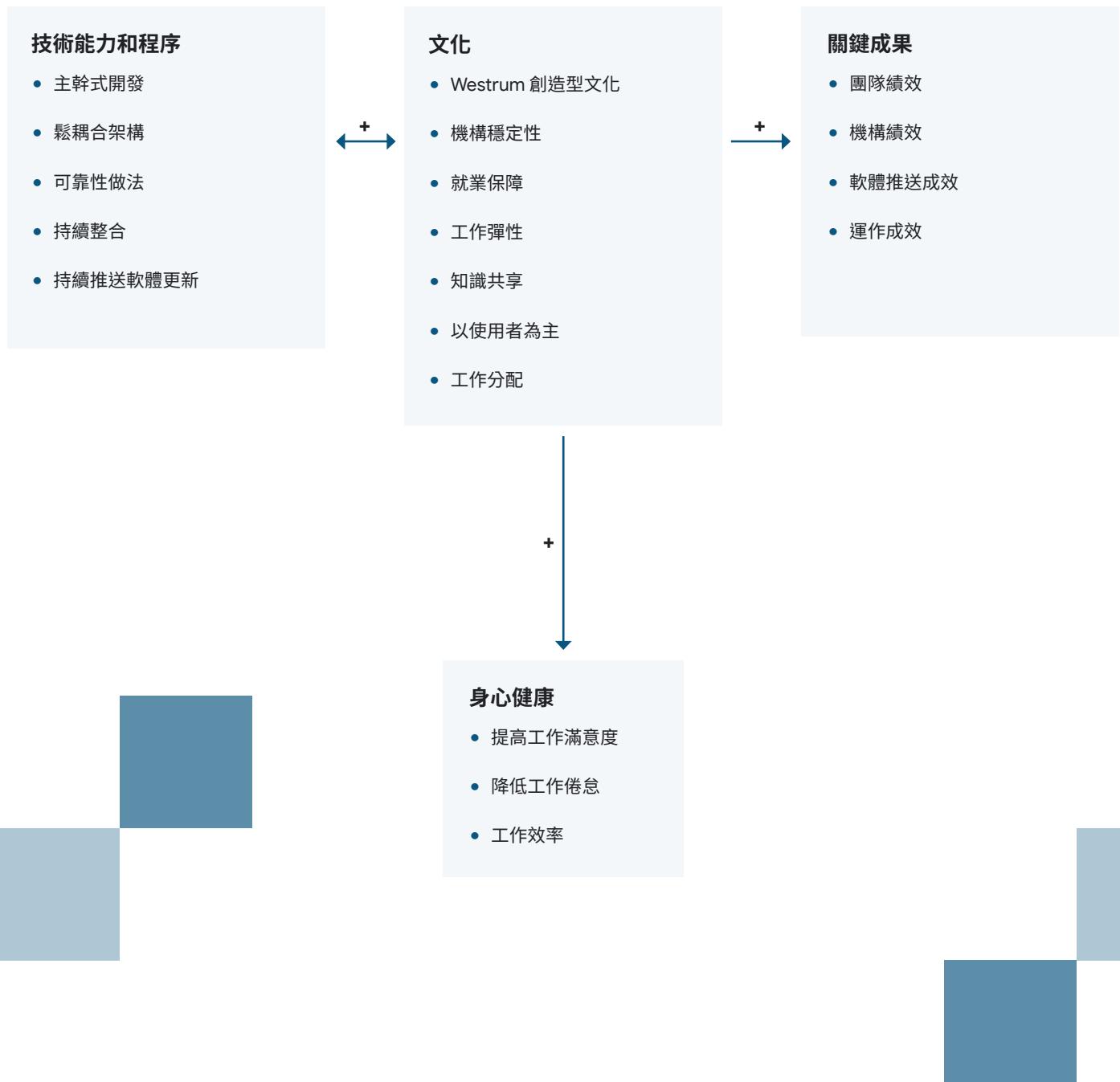
## 第6章模型：彈性基礎架構是成功關鍵

雲端運算可提升基礎架構彈性，進而影響關鍵成果。此外，雲端運算也能促進身心健康。



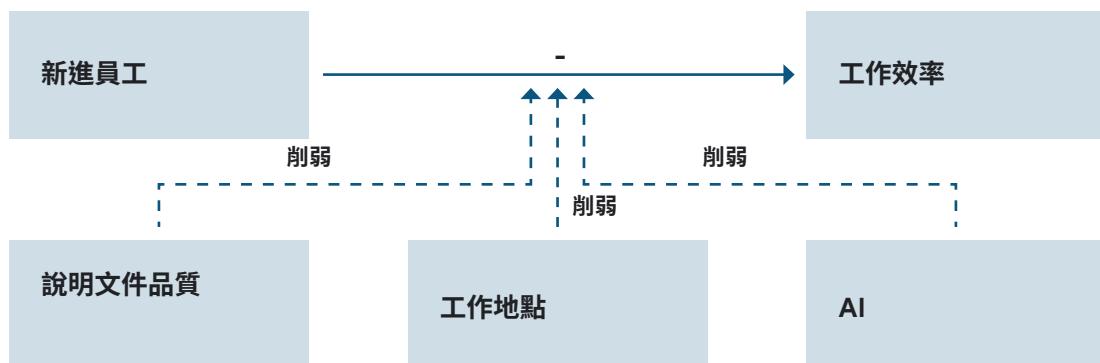
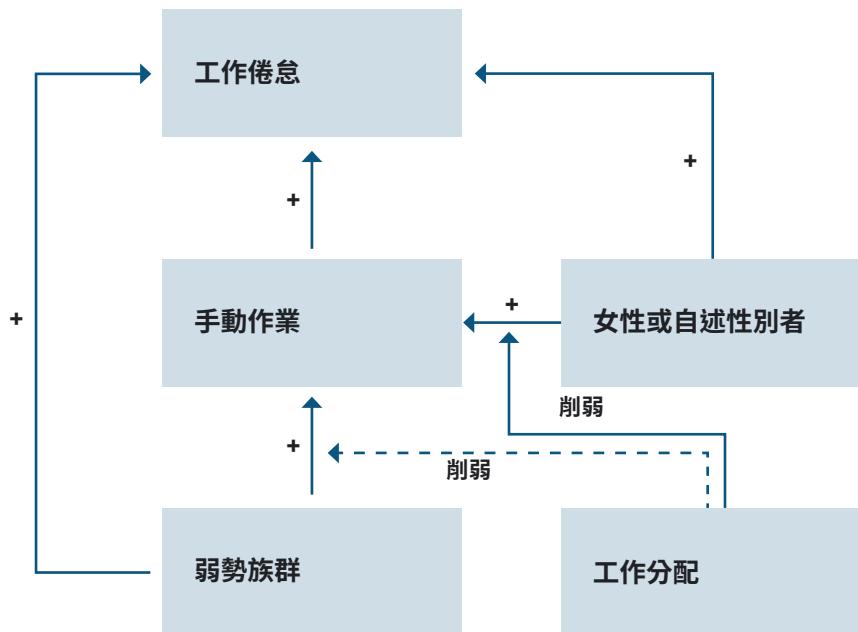
## 第 7 章模型： 營造健康的文化才能推動革新

這張圖表顯示，文化是所有變數類別的核心，具體而言，文化與技術能力、關鍵成果和身心健康都有正向關係。



## 第8章模型：身分與身心健康的關聯性： 原因、影響方式和時機

這一章包含兩個模型，其中一個探究弱勢員工和非男性員工「為何」和「何時」發生高度工作倦怠。另一個探究說明文件品質、工作地點或 AI 是否有助於新進員工提高工作效率。



# 延伸閱讀

加入 DORA 社群來討論、學習和合作，一起提升軟體推送和運作成效 [DORA.community](https://dora.community)

進行 DORA 開發運作做法快速檢驗：<https://dora.dev/quickcheck>

認識有助於提升軟體推送成效和機構績效的技術、程序和文化能力：<https://dora.dev/devops-capabilities/>

尋找 SRE 資源：

<https://sre.google>

<https://goo.gle/enterprise-roadmap-sre>

閱讀書籍：《Accelerate: The science behind devops: Building and scaling high performing technology organizations》(ACCELERATE：精益軟體與 DevOps 背後的科學)，IT Revolution 出版：<https://itrevolution.com/product/accelerate/>

使用 SPACE 架構找出團隊適用的一組指標：《The SPACE of Developer Productivity: There's more to it than you think》(運用 SPACE 架構提升開發人員工作效率：考量層面比想像中更多)：<https://queue.acm.org/detail.cfm?id=3454124>

如要深入瞭解現代程式碼審查做法，歡迎參閱以下幾篇研究報告：

- 「Expectations, Outcomes, and Challenges of Modern Code Review」(現代程式碼審查的預期、成果和挑戰) <https://dl.acm.org/doi/10.5555/2486788.2486882>

- 「Code Reviews - From bottlenecks to Superpowers」(程式碼審查：從突破瓶頸到發揮超能力)

<https://learning.acm.org/techtalks/codereviews>

- 「Modern Code Review- A Case Study at Google」(現代程式碼審查：Google 個案研究)

<https://dl.acm.org/doi/10.1145/3183519.3183525>

- 「The Choice of Code Review Process: A Survey on the State of the Practice」(選擇程式碼審查程序：做法現況問卷調查) [https://link.springer.com/chapter/10.1007/978-3-319-69926-4\\_9](https://link.springer.com/chapter/10.1007/978-3-319-69926-4_9)

- 「Investigating the effectiveness of peer code review in distributed software development based on objective and subjective data」(根據客觀與主觀資料，研究分散式軟體開發作業中的同儕程式碼審查效率) <https://jserd.springeropen.com/articles/10.1186/s40411-018-0058-0>

閱讀書籍：《The No Club: Putting a Stop to Women's Dead-End Work》(說「不」俱樂部：停止讓女性陷入絕境的工作)。Simon & Schuster 出版：

<https://www.simonandschuster.com/books/The-No-Club/Linda-Babcock/9781982152338>

DORA 研究計畫出版品，包括往年的《加速發展：開發運作現狀報告》：<https://dora.dev/publications/>

研究和報告相關常見問題：<http://dora.dev/faq>

查看有關這份報告的勘誤表，以及提交變更、修正和澄清事項：<https://dora.dev/publications/errata>

# 附錄

## 改良軟體推送成效評估方式

今年，我們調整了「變更失敗」和「從失敗狀態復原」的評估法，

作答者能以新方式回答**變更失敗率**。往年我們提供六個選項 (0-15%、16-0% 等) 供作答者選擇，今年則改採拉動滑桿的方式，方便作答者從 0% 和 100% 之間選取符合自身狀況的值。

調整的原因有二：

- 變更失敗率始終與其他三個軟體推送成效指標的評估結果略有不同<sup>1</sup>，我們推測部分原因可能在於值區大小。若能收集到更精確的回答，將有助於產生更準確的統計數據，而且我們確實能驗證這個假設。
- 我們假設，當今的團隊比十年前（約是我們開始進行研究的時候）更瞭解自己的變更失敗率。不過，實際上我們無法驗證這個假設。

今年我們以兩種不同的方式，瞭解失敗狀態復原情形：

- 往年我們會這樣問：以您開發的主要應用程式或服務而言，一旦發生影響使用者的服務事故或瑕疵（如服務意外中斷或服務異常），通常多久才能還原？
- 今年，我們在問題中限定情況（以下使用粗體凸顯差異之處，但在問卷調查中並未這麼做）：「以您開發的主要應用程式或服務而言，**由於實際工作環境或對使用者發布的變更，導致服務品質下降（例如造成服務異常或服務中斷），因而必須採取補救措施（例如提供 Hotfix、復原版本、向前修正、修補程式）**，通常多久才能復原？」

<sup>1</sup> Forsgren, N., Humble, J. 和 Kim, G.,《Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations》(ACCELERATE: 精益軟體與 DevOps 背後的科學)，IT Revolution Press 出版，2018 年，37–38。

過去詢問復原時間時，我們沒有特別區分「由軟體變更」或「由其他原因」引起的失敗情形，例如地震導致資料中心服務中斷。我們推測，使用更精確的說法有助於我們比較類似的失敗類型，也能讓失敗相關指標在統計上與其他三項軟體推送成效評估指標更一致。

現在我們採用「失敗部署作業復原時間」這項指標，來代替過去較籠統的「還原時間」（有時縮寫為「MTTR」）。MTTR 中的「M」曾引起社群成員的困惑，因為這個字母可能代表平均值 (mean) 或中位數 (median)。此外，韌性工程等領域的從業人員在深入分析失敗原因時，逐漸不再將 MTTR 視為指引學習及改善方向的可靠指標<sup>2</sup>。

我們相信，新加入的問題和「失敗部署作業復原時間」這項新指標，更符合軟體推送成效的評估精神。

<sup>2</sup> 「Moving Past Simple Incident Metrics: Courtney Nash on the VOID」（捨棄籠統的事件指標：Courtney Nash 談 VOID），<https://www.infoq.com/articles/incident-metrics-void/>



# 數據比較背後的數學原理

在這份報告中，您可能會經常看到以下的統計數據呈現方式：特定變數的評估結果越高，將導致其他變數的分數提高 N 倍。這是什麼意思？又與什麼相關？以下是我們建立這些公式的方法：

- 首先，我們使用已考量潛在偏差路徑的迴歸方法評估模型，並根據該模型建立公式：

$$\text{快樂} \sim 5.64 + 0.19 * \text{日照量} + 0.14 * \text{溫度}$$

- 針對目標變數（例如日照量），找出要比較的高值和低值：

$$\text{平均值} = 6.3, \text{標準差} = 1.4$$

$$\text{日照量高} = \text{高於平均值 1 個標準差} = \text{平均值} + \text{標準差} = 6.3 + 1.4 = 7.7$$

$$\text{日照量少} = \text{低於平均值 1 個標準差} = \text{平均值} - \text{標準差} = 6.3 - 1.4 = 4.9$$

- 計算共變數的平均值，在這個例子中就是溫度，等於 0.24（標準溫度）。

- 代入步驟一的公式，算出日照量高和日照量低的值。您會發現公式中只有一個數字不同，我們正是透過此方式維持其他要素均等，並只單獨區隔目標變數。

$$\text{日照量高的預期快樂程度} = 5.64 + 0.19 * 7.7 + 0.14 * 0.24 = 7.1$$

$$\text{日照量低的預期快樂程度} = 5.64 + 0.19 * 4.9 + 0.14 * 0.24 = 6.6$$

- 計算比率：

$$\frac{\text{日照量高的預期快樂程度}}{\text{日照量低的預期快樂程度}} = \frac{7.1}{6.6} = 1.1 \text{ 倍}$$

- 這個比率顯示，相對於日照量少，日照量多帶來的快樂高 10%。

# 什麼是「模擬」？

「模擬」並不是編造資料。我們使用貝葉斯統計法算出**事後機率**，來代表「不同參數值預期出現的機率」<sup>3</sup>。「模擬」指的是計算這個事後機率超過 1,000 次，以找出對資料而言最可信的參數值 (平均值、Beta 權重值、Sigma 值、截距值等等)。「將事後機率想成一個桶子，其中裝滿 0.1、0.7、0.5、1 等等的參數值。在桶子內，每個值的出現機率與其事後機率成正比，因此接近高峰的值會比接近尾部的值更常見」<sup>4</sup>。

換句話說，我們能經由模擬找出資料的可能解讀方式，以及瞭解不確定性高低。您可以將每次模擬視為一個小型

AI，這個 AI 只知道我們的資料和幾項規則，然後要試圖根據資料猜出參數值。只要模擬 4,000 次，即可針對指定參數得出 4,000 個小型 AI 的猜測值。這些猜測值非常實用，例如我們可以算出平均猜測值、89% 猜測值介於哪兩個數值之間、多少猜測值在特定門檻以上、猜測值之間的變異度多大等等。您甚至可以做一些有趣的操作，例如合併多個模型的猜測值 (即模擬)。

當您 在圖表中看到許多線條或潛在值分布情形，代表我們盡可能呈現資料最合理的解讀方式，以及不確定性高低。

<sup>3</sup> McElreath, Richard, 《Statistical rethinking: A Bayesian course with examples in》(統計反思：R 與 Stan 中的貝葉斯做法教學與範例)，Chapman and Hall/CRC 出版，2018 年，第 50 頁

<sup>4</sup> McElreath, Richard, 《Statistical rethinking: A Bayesian course with examples in》(統計反思：R 與 Stan 中的貝葉斯做法教學與範例)，Chapman and Hall/CRC 出版，2018 年，第 52 頁

