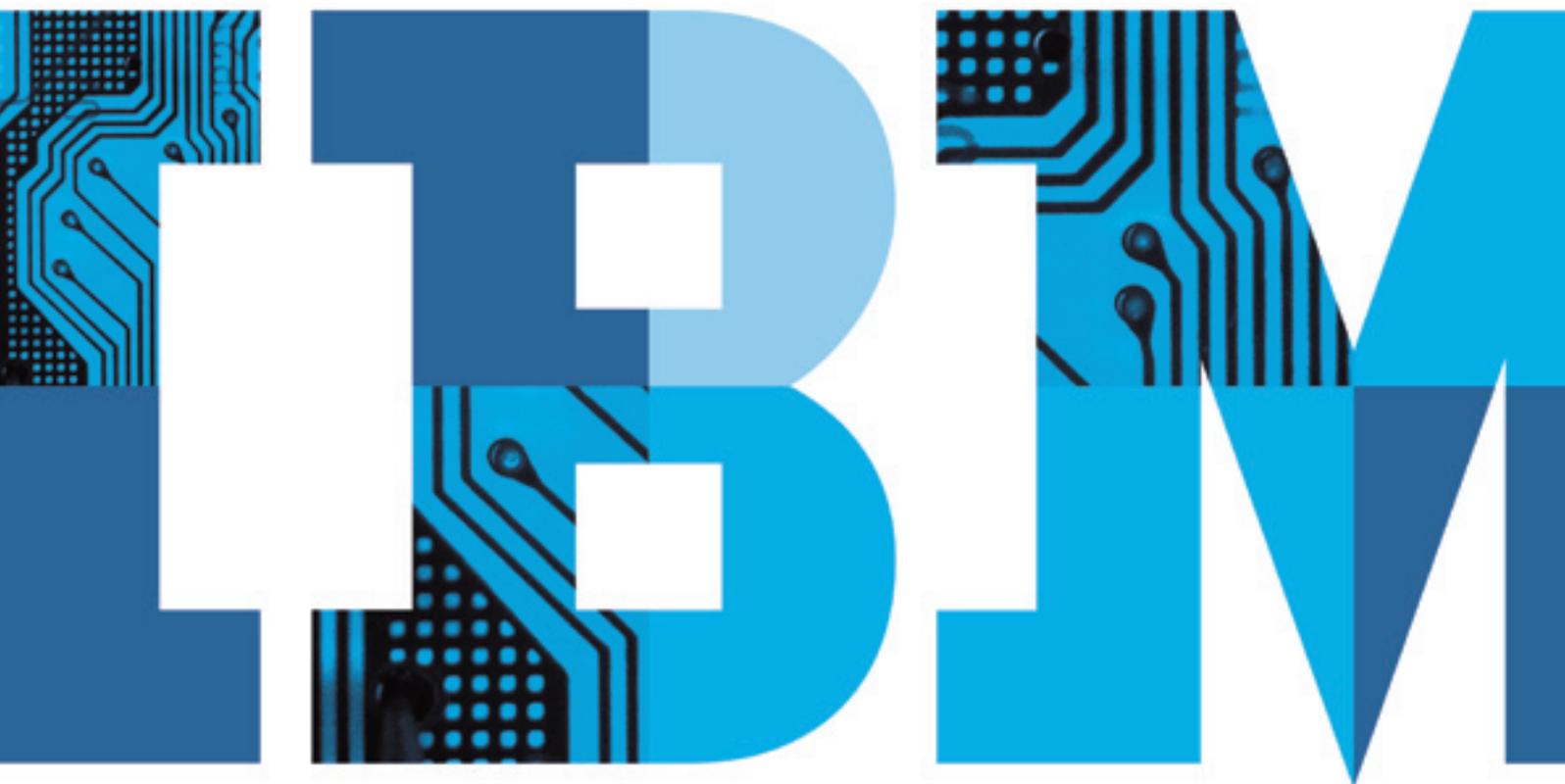


適合新世代電子產品的尖端多核心開發技術



2 適合新世代電子產品的尖端多核心開發技術

前言

微型化趨勢持續帶動電子產業的走向。裝置 (device) 的尺寸及其基本元件不斷縮小，而其功能及效能卻日益增加。

裝置也因為加入通訊連結，而成為高度互連生態系統的元件。電子公司採用多核心處理器及一套由兩個以上獨立核心所組成的處理系統¹，使其新世代產品中達成這些進展。

因為多核心處理器在電子產品—包括個人電腦、行動電話、遊戲系統、網路設備、工業控制系統甚至於醫療裝置—變得更為無孔不入，針對這些系統的軟體開發需求也已變成電子業界的主要關切重點。多核心架構具備多項優點，包括更高的效能、較低的耗電量、低成本及更大的彈性，但只有在開發出足以發揮這些優點的相對應軟體時，才能夠實現這些優點。現今電子業界中的許多軟體開發人員，對於最適合用於多核心的軟體，皆缺乏撰寫技巧。此外，多核心架構也隨著核心數目從二到四核心，增加到三十二個不同核心而呈現指數性複雜化；也就是說，傳統的開發方法已不敷使用。處理複雜性的唯一方式，就是透過自動化及使用可針對不同核心自動化軟體部署的更高階語言。因為上市時間在電子產業中是極為重要的區分要素，因此若想符合市場的交期，就必須充分運用企業現有的軟體程式碼。但它們多半是並未針對多核心技術最佳化的單核心原始碼。

本 IBM® Rational® 白皮書討論可讓您的開發團隊充分利用新多核心技術優點的尖端軟體開發方法，以及在多核心系統重複使用單核心軟體的自動化方式，以便讓您透過重複使用現有系統，同時在業界中達到超水準的表現及大幅節省時間和金錢。

多核心的未來

設計潮流的演變，已驅使建構現今先進電子產品的電腦運算平台必須增加效能，以及減少耗電量和成本。以往處理器總是被迫增加晶粒上的電晶體數目及增加時脈 (clock speed)，以不斷挑戰摩爾定律 (Moore's law) 的方式來提供更多效能。² 雖然這種情況仍將繼續，但通往真正效能及電源管理躍進的途徑，應該是朝向多核心技術的創新。

以多核心為基礎的系統可以降低耗電量，而且通常可以較低的時脈運作，並能大幅改善電池壽命及減少所產生的熱量。較低的操作溫度，代表企業可以採用較為安靜及更具能源效率的無風扇冷卻系統。能開發出具有較小板型的產品、降低不同處理器的數目並有利於多核心處理器數目的減少。此外，多核心作業環境比以單核心為基礎的環境更容易達成真正的多工作業，以及更佳的應用程式效能，因為單核心環境會較快達到 CPU 運用臨界值。現在在處理不斷增加的電腦運算密集應用程式的負載時，已能達成真正的平行處理。另外，以多核心進行系統性能的負載平衡及分離，可改善系統牢靠性及安全性。

從個人電腦及工作站，到高階智慧手機及可攜式裝置，不斷改善效能、電池壽命及進階功能的消費者需求，已使裝置製造商必須建立及提供使用多核心技術的產品。新一代的個人電子裝置及智慧型產品將必須仰賴多核心環境，為日益重視技術層次的客戶群提供更大的連線能力、反應能力、實用性和產能應用程式。軟體將成為實現多核心技術的關鍵因素，而成功或失敗則取決於嵌入式軟體團隊。

多核心的挑戰

實現多核心技術需要大量的成本，並會對現今產品開發社群帶來風險。多核心環境對產品的設計及交付，以及用來驅動其軟體密集應用程式，皆帶來更大的複雜性。必須做出適當的資源及功能分配選擇，此選擇也會影響產品的架構。對系統設計師來說，決定處理器實際數目、所需要的核心數、作業系統（對稱或非對稱）特性，以及所需中介軟體 (middleware) 會變得更為困難。設計師必須在軟體前端，將應用程式磁碟分割、核心之間及工作之間通訊，以及以多核心為基礎的系統的設計延展性等新問題納入考量。作業系統及所使用的潛在中介軟體的功能，會使整體軟體設計更複雜。

改用多核心的企業將須改善其軟體設計團隊的設計技巧。必須普遍採用平行處理，而建構應用程式的方法也必須改變，才能充分利用多核心作業環境中所提供的平行架構的優點。因為多工使用增加的考量，工作間的通訊機制及工作對核心的配置，都成為是否能成功開發應用程式的關鍵因素。對於軟體密集系統的工作和元件來說，重複使用的重要性日益增加。

必須重新建構現有應用程式，使其在新的多核心架構上執行，以便利用平行架構基本效能改善的優點，而更為重要的是確保能夠正常運作。即使能在單核心系統（甚至是多作業系統）上正確、安全地執行，在多核心系統所提供的平行環境中，也可能會出現錯誤甚至不安全的運作行為。

在將軟體部署至多核心系統時，除錯及測試變得更為重要。傳統的程式碼層級除錯和後期測試技巧，通常無法擴展至多核心環境。需要新技巧及測試環境，以確保軟體能正確運作，並能提供所需要的性能水準。

IBM Rational 的多核心開發方法

建議採取三種方法，以進行有效的多核心開發：

- 執行折衷研究以評估替代方案
- 使用現有的軟體加速開發
- 自動化軟體生成以改善品質

執行折衷研究以評估替代方案

直接將現有系統移至多核心處理器，實際上可能會造成它以相同速度或更慢的速度執行。³ 這是因為實際核心在個別運作時，通常會比原始處理器來得慢。在共用資料拖慢通訊時尤其明顯。整體應用程式甚至可能會以單核心的速度執行。執行折衷研究 (trade-off studies) 對於評估替代方案，以及確保實現所計畫的效能、速度及資源運用改善是非常重要的。

4 適合新世代電子產品的尖端多核心開發技術

為您產品的軟體架構建模，並為其建立不同的選項，以便在多核心上執行，對於成功實作是非常重要的。它可以將現有工作對應各核心，並視需要加入額外工作。此外，建模 (modeling) 可讓您瞭解如何針對與應用程式相關的工作和程序，最佳化其間的通訊。

另一種關鍵方法是模擬您要建構的模型，以驗證一切是否能依期望運作。這樣的模擬是真正能確保系統正常運作的有效方法。

在進行折衷研究時，模型能讓您記錄特定決策的理由。另外一項重要的層面，是可將您的設計對應當初協議的原始需求。瞭解需求的即時性層面，能幫助您做出應將哪些工作對應至哪個核心的決策。例如，您可能不希望產品的軟體使用者介面與即時或關鍵控制元件都在相同核心上。

折衷研究對於瞭解您軟體的哪些部份的選項應搭配哪個核心，是相當重要的。若沒有這樣的研究，您就無法評估最佳的替代方案，也無法確定您的應用程式是否能以最佳化方式在核心上運作。

使用現有的軟體加速開發

您現有的產品或許已經能夠配合關鍵應用程式軟體成功運作。它在現有的單核心作業環境中執行情況良好，甚至可在新的多核心環境中正常運作。因為想要保有它的價值，您會希望在工作時儘可能重複使用這些產品，確保能更順利的執行及擴展，以符合新環境的需求。

以模型驅動方法為多核心進行開發工作，可為您現有軟體帶來大量的重複使用機會。從軟體和其元件部件之結構及關係的簡單視覺化，到重新建構程式碼以最佳化對多核心環境的部署，到整合及擴展至全新設計和應用程式架構，正確的模式驅動工具方法可以大幅加速您重複使用您所信賴軟體的能力。

視覺化您現有的應用程式軟體，可增加您對軟體如何架構，以及如何關聯個別元件的瞭解，為您帶來立即的收穫。它可以讓您釐清您設計的最佳及次佳層面，並產生目前設計的記錄文件，以協助分析如何在新設計中善用程式碼，甚至不需要加以變更。此視覺化的程式碼可鎖定新多核心環境，讓您可以執行折衷研究，以判斷究竟是使用應用程式或加以重新建構 (refactoring)，才能達到最佳效果。

若您判定現有軟體無法在新環境中順利執行，或是未能針對多核心環境妥善架構，重新建構會是相當實用的選項。藉由以模型元件方式匯入程式碼，即可重新建構及重新產生程式碼，以支援為多核心所量身打造、更佳的應用程式架構。透過建模及自動產生程式碼，即可使重新建構現有的程式碼變成替代重新撰寫應用程式的划算方案。

最後，可在為鎖定多核心環境的產品所建立的新設計及應用程式中，使用現有的軟體元件。而強大模型驅動開發工具中的視覺化、重新建構及自動化產生功能，也能大幅加強您做出這些重複使用決策的能力。可為產品撰寫及提供新應用程式，並從頭開始限制設計需求。此外，重複使用的元件具有一份在早期產品成功的追蹤記錄，因此能瞭解及測試其行為層面，並將新測試及除錯的範圍限制在應用程式的效能部份。

在移至新硬體時，重複使用舊有程式碼一直是非常重要的，因為從頭重新撰寫所有程式碼是極為耗時的。另外，擁有已經過測試的程式碼，對於及時完成專案也極為重要。以圖形化方式檢視可重複使用的程式碼，能幫助您瞭解它在您切換至新的多核心硬體平台時，與您所加入功能的配合程度。

自動化軟體生成以改善品質

因為多核心本身即可提供高度平行處理的環境，因此應用程式內的工作之間，以及應用程式之間的通訊即成為是否能成功部署這些應用程式的關鍵因素。在大部份的情況下，工作在從相同核心切換至不同核心時，必須與不同的機制進行交流。這種對於有效通訊的高度依賴，也進一步增加了工作間具備良好介面，以及使用建模環境達到自動化產生支援機制的重要性。

想完全瞭解、指定及發掘能切換以鎖定不同硬體及通訊協定的替代方案，就必須使用模型及具體的自動化程式碼生成。這可以將設計視覺化，並在所產生的程式碼中系統性地複製該設計。當將工作從某個核心移至另一核心時，即可重新產生已建模的變更，並視需要反映在新核心專屬的程式碼當中，即使它使用不同的作業系統或必須使用不同的通訊協定來與原始核心上的工作交流亦然。這通常需要手動撰寫程式碼以進行變更，但藉由經過建模的應用程式，即可以在簡單指定選項後，由模型驅動開發工具的自動產生功能來接手其餘的轉換作業。在開發過程的每個階段，建模和程式碼生成對於協助開發人員測試及部署其多核心應用程式，都是非常重要的。在分秒必爭的市場，像是電子產業中尤其是如此。

結論

為了追求能在更短的時限內在市場上推出創新及與眾不同的產品，電子產業勢必都得逐漸採用多核心處理方式，才能享受更高效能、更低耗電量、更長電池壽命、更低成本及增加彈性的好處。但唯有先建構出多核心的相對應軟體，才可能實現這些優點。電子業可使用數種尖端方法，以簡化及加速多核心軟體開發。藉由使用模型驅動開發方式，軟體設計團隊可以執行多核心設計折衷研究、視覺化及重新建構可達成更大重複使用率的程式碼，並依據硬體組態及所使用的通訊協定自動產生程式碼。

關於進一步的資訊

若要瞭解更多關於多核心開發技術的資訊，請聯絡IBM業務代表或事業夥伴，或請造訪：

ibm.com/software/rational/info/multicore/



台灣國際商業機器股份有限公司

台北市松仁路7號3樓

市場行銷處：0800-016-888按1

技術諮詢熱線：0800-000-700

© 版權所有 IBM Corporation 2010

於台灣列印

2010年3月

版權所有

IBM、IBM 標誌、ibm.com 以及 Rational 均為 IBM 股份有限公司在美國、其他國家或兩者的商標或註冊商標。如果這些和其他 IBM 商標名稱於本文首次出現時標有商標符號 (® 或 ™)，則這些符號代表本文付梓時 IBM 在美國的註冊商標或普通法商標。這類商標也可能是在其他國家的註冊商標或普通法商標。最新的IBM商標清單請見 ibm.com/legal/copytrade.shtml 網頁的「著作權與商標資訊」。

其他公司、產品或服務名稱可能是其代表公司的商標或服務標誌。

在本刊物中對 IBM 產品或服務之參照，並不代表 IBM 計劃在 IBM 所有有服務據點的國家中提供該產品或服務。

本文件中所包含的資訊，僅針對諮詢用途提供。儘管本公司已盡力確認本說明文件中所包含的資訊的完整性及精確性，惟本說明文件僅以「現況」提供，不含任何明示或默示的保固。此外，本資訊是根據 IBM 目前的產品計劃及策略為基礎，IBM 得隨時在不通知情況下變更這些計劃及策略。IBM 不為任何因使用本說明文件或任何其他文件所產生，或與其相關的損害負責。本說明文件中所包含任何資訊，皆非用於亦不具備建立任何與 IBM (或其供應商或授權者)相關之保固或表示之效果，或用於針對掌管 IBM 軟體使用之適用授權合約變更其條款及條件。

¹ http://en.wikipedia.org/wiki/Multicore_processor

² http://en.wikipedia.org/wiki/Moore's_law

³ <http://www.forbes.com/2009/11/23/google-microsoft-programming-technology-cio-network-multicore-hardware.html>

請回收使用