

IBM Rational Rhapsody 及 Rational Rose

白皮書

2009 年 6 月



Rational. software

改用 IBM Rhapsody 軟體

提高生產力與品質的手冊

Scott Niemann · IBM

內容	
2	重點摘要
2	MDD 與 CASE 的比較
3	IBM Rational Rhapsody 及 IBM Rational Rose
7	特定領域建模
14	總結

IBM Rational Rose 促使 UML 成爲系統設計市場的一員。IBM Rational Rhapsody 則可提供隨附其他功能的模型導向開發 (MDD) 解決方案，可因應新興技術和開發標準。

重點摘要

過去十年來，使用統一建模語言 (UML) 的建模方式已發展爲描述系統架構和設計目的的有效方法。而憑著 IBM® Rational® Rose® 軟體，IBM 也成爲系統市場中的主要 UML 廠商。Rational Rose 軟體不僅可透過 UML 解決系統的設計問題，還隨附有助於整合其他生命週期開發工具的架構規範，以便建立從概念到目標系統的完整解決方案。雖然此工具仍然有用，但技術及開發標準的不斷更新已促成新解決方案的誕生，如 IBM Rational Rhapsody® 軟體，可提新的功能。

Rational Rose 應用程式透過工程師可輕鬆採用及理解的 UML 作爲共用語言，降低設計系統與軟體架構的複雜性，並提高協同作業效率。儘管此方法已取得卓越的成果，但爲了因應日新月異的技術發展及越來越複雜的系統，必須改用新的解決方案才能處理電腦輔助系統設計 (CASE) 的技術落差。模型導向開發 (MDD) 是處理測試及實作的技術，同時可讓您輕鬆利用 Rational Rose 模型的現有智慧財產，甚至是原始碼。

MDD 與 CASE 的比較

模型導向開發是使用 UML 2 模型，來指定設計、需求及實作。因此，可執行設計流程的不同組成元件規格，即系統開發過程中可進行驗證。Rational Rose 解決方案的基礎技術 CASE 主要用來記錄系統架構，並指定如何進行實作。然而，文件是靜態的，所以 CASE 技術無法提供一個理想的方法，來確認規格是否正確或系統如何運作。因此，如果系統設計初期就有錯誤，可能要等到完成實作之後才會發現。

產品特色

圖 1 說明若生命週期的初期階段忽略錯誤所需付出的代價。許多錯誤發生在進階 (HL) 及軟體 (SW) 設計的初期階段，主要是因為誤解需求。若能在實作之前找出錯誤，便可節省金錢和時間。隨著生命週期的發展，以及架構和實作功能的加入，尋找並更正設計錯誤的代價可能更加龐大。

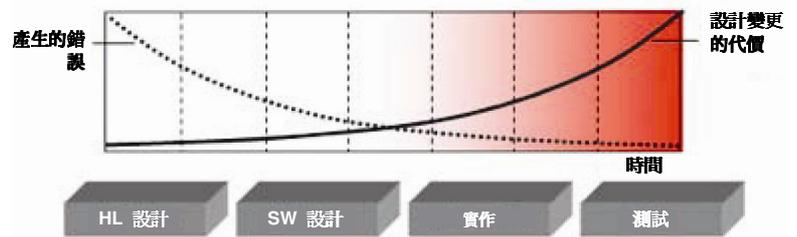


圖 1. 設計階段發生錯誤的代價

Rhapsody 可協助您在實作之前，先找出並修正錯誤，進而節省時間及金錢。

IBM Rational Rhapsody 的 MDD 方法旨在找出 HL 設計初期的錯誤，因為在此階段進行更正的代價最小。在設計階段修正錯誤不僅能降低成本，還可節省開發時間。

IBM Rational Rhapsody 及 IBM Rational Rose

IBM Rational Rose 使用者可輕鬆將資料移轉到 Rhapsody 軟體，以利用其進階的 MDD 功能。

Rhapsody 軟體不但能支援最新的 UML 標準，還可支援標準的特定領域設定檔，如系統建模語言 (SysML)，因此 Rational Rose 基本上可說是 IBM Rational Rhapsody 的子集。開發人員可利用此彈性，使用最適合說明其問題領域的設計語言。屬於 MDD 平台的 IBM Rational Rhapsody 主要用於簡式建模無法應付的環境面向。本白皮書稍後將討論這些面向。其移轉程序可讓 Rational Rose 軟體使用者在移轉資料之後，利用 IBM Rational Rhapsody 的進階功能。

產品特色

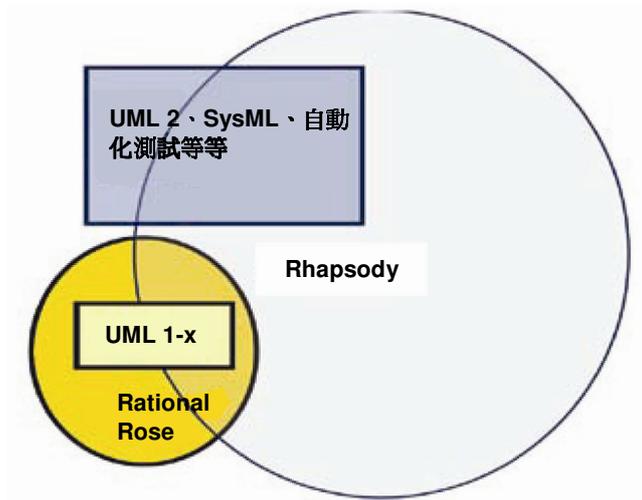


圖 2. IBM Rational Rose 及 Rational Rhapsody

IBM Rational Rhapsody 透過強大的模型導向測試與除錯誤功能，從眾多競爭對手中脫穎而出。

模型導向測試

Rational Rhapsody 軟體的最大差異化優勢是其測試功能，尤其是模型導向測試。有了這項功能，工程師即可在建立架構時進行驗證，以便在未撰寫程式碼的開發初期減少錯誤。如此一來，即可降低設計後期修正錯誤所花的時間，以較低成本加速開發流程。此外，還可自動化設計測試，以便每晚執行迴歸套件。

模型導向測試的兩個主要部分是模型層次除錯及自動化的需求導向測試。

模型層次除錯

您可使用模型層次除錯功能，在建置系統時加以測試，並在引進系統時減少錯誤。此除錯功能可分為幾個不同的精度層次，即實作的系統層次及來源層次。

產品特色

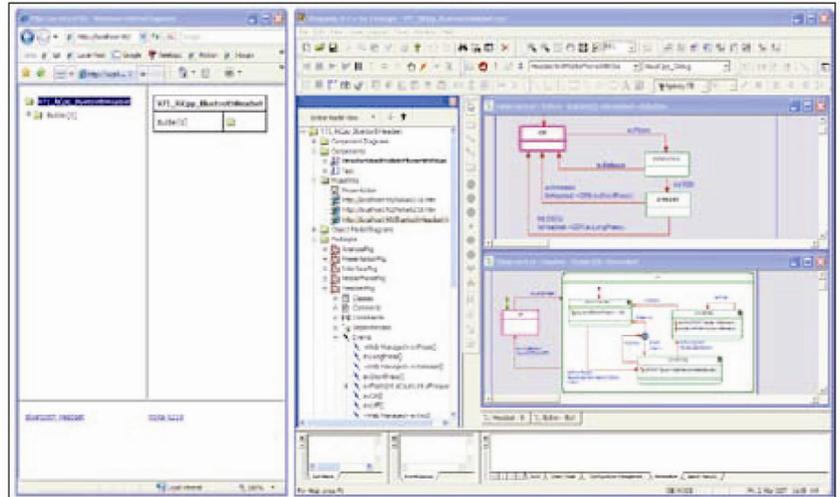


圖 3. 模型層次除錯

視覺化的模型導向除錯功能可讓您在實作時、甚至是建置元件之前，查看整合型系統的運作方式。

在圖 3 中，右邊的設計會在系統執行時強調顯示。如此一來，便可看到您在特定系統元件中的確切位置，即子系統或實際類別。左邊的 HTML 畫面可將新刺激加入設計中。Rational Rhapsody 應用程式會自動建立這個畫面，以便您點選畫面上的事件，輕鬆測試系統。如果該系統是開發中的裝置，如手機，則可延伸畫面，以較真實的方式呈現系統。此方法可有效地向客戶、銷售人員及工程師表達設計概念。

開發人員往往需要在有硬體之前，便執行系統的功能測試。模型導向除錯可在生命週中的這些階段修正許多錯誤，有助於降低整合及系統測試所花的時間。具備硬體之後，如果可從目標連到主機平台，即可建置適用於目標平台的應用程式，並在實際目標上執行模型導向除錯。

產品特色

將測試案例連到需求資料庫可讓您輕鬆追蹤並顯示各測試如何驗證客戶需求，以及如何測試各需求以確保符合規定。

需求導向測試

需求導向測試可讓您確認系統是否符合客戶需求，這些需求可能是使用 IBM Rational DOORS®、IBM Rational RequisitePro® 或 Microsoft® Word 或 Excel 文件進行維護。在 MDD 環境中，這些需求可描述成模型元素，並連到說明系統如何滿足客戶需求的實務案例。這些實務案例就會變成可建置並自動執行的測試向量，以進行回歸測試。實務案例只是根據外部刺激，說明系統的預期執行作業。而實務案例與客戶需求之間的連結則可同步處理至需求資料庫，並提供可追蹤的資訊。圖 4 即說明此流程，先將需求資料庫匯入 Rhapsody 軟體，然後連到設計層次的測試向量。



圖 4. 需求—設計—測試連結。

產品特色

整合功能強大的 UML 及特定領域設定檔之後，組織便可建立設計目的明確的特定產業模型。

您可使用 MDD 環境的執行結果，判斷模型化通訊系統是否已符合需求。換言之，圖 5 右邊的測試向量可當作基準性能測試指標，以判斷是通過或失敗。執行模型可顯示系統實際執行的作業，並自動與基準性能測試指標做比較。追蹤功能可讓您輕鬆追蹤結果執行系統所滿足的客戶需求，以及哪些部分仍需加強。您可使用此方法輕鬆推出符合客戶期待的系統。

特定領域建模

UML 是功能強大的建模語言，但只針對系統提供一般的資訊範本。建立類似相關領域的模型可讓同事及客戶更清楚設計目的。IBM Rational Rhapsody 支援最新版的 SysML，因此系統工程師可使用功能分解方法，來建置進階系統元件和建立後續子系統的模式。

除了軟體及系統工程等相關領域，Rhapsody 軟體還支援汽車、航太、國防及通訊業的特定領域設定檔。您也可以針對貴組織建立設定檔，以便有效指定確實的設計目的。圖 5 是以 Rhapsody 解決方案呈現的美國國防部架構規範 (DoDAF) 範例，此規範用於航空及國防市場。

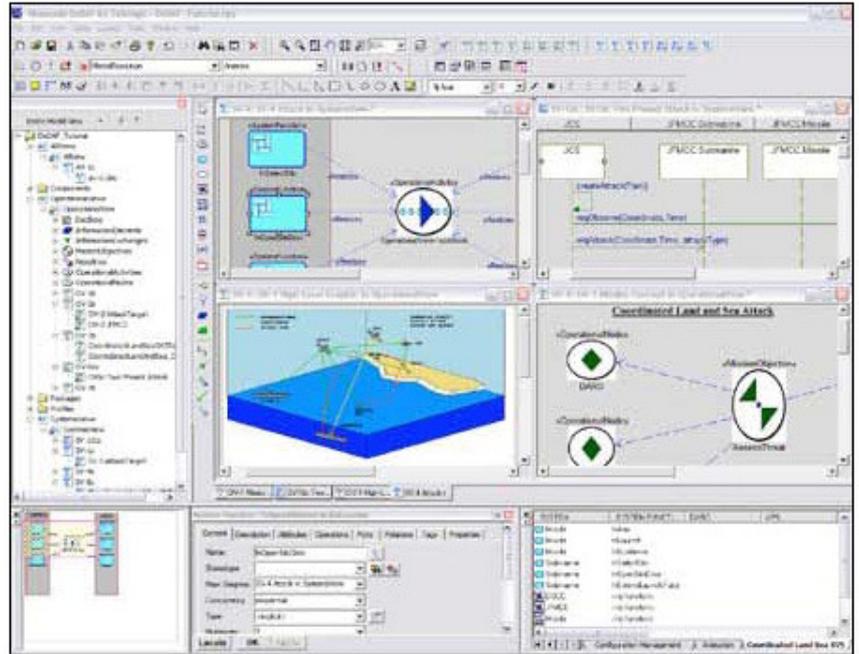


圖 5. 特定產業的設定檔範例—DoDAF。

您也可以針對特定的組織目的建立設定檔。設定檔是 UML 2 的延伸套件，因此相容於 XML Metadata Interchange (XMI)，可進行模型交換。系統設計師可在 Rhapsody 軟體中建立設定檔模型，當作建模範本，然後發給全公司的系統開發人員。此方法可讓貴組織更有效地遵守設計最佳實務及準則。

產品特色

Rhapsody 軟體可提供整合多種配置管理工具的彈性編碼環境，並促進協同開發。

協同合作

另一個能夠彰顯 Rational Rhapsody 好處的領域，就是著重開發強大軟體的開發團隊。Rational Rhapsody 軟體可提供整合模型及程式碼的環境。您可以只在 Rational Rhapsody 中建立作業，然後以手動方式撰寫 C 語言的程式碼。資訊會儲存在模型中。您將擁有一個移入或移出配置管理 (CM) 儲存庫的模型元素，可大幅降低及簡化平行開發，不需要分別維護模型和程式碼。

圖形差異分析與合併

Rhapsody 的圖形差異分析與合併功能可整合模型及程式碼。如圖 6 所示，Rhapsody 軟體不僅隨附本身的自動合併功能，也整合了 CM 工具，如 IBM Rational ClearCase® 軟體。例如，您可直接在 Rational ClearCase 軟體中執行自動合併功能，而 Rhapsody 解決方案的差異分析與合併工具則負責自動合併設計（模型及程式碼）。若有任何衝突，系統開發人員即可檢視 IBM Rational Rhapsody 中的圖形差異來解決相關問題。模型及程式碼合併後，有助於簡化並大幅降低整合時間。

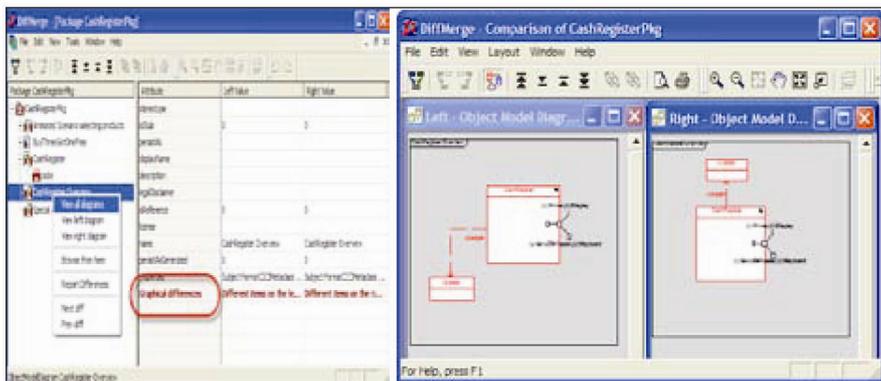


圖 6. Rhapsody 軟體的差異分析與合併功能

產品特色

圖 6 的左視窗顯示模型資料，並說明是否有需要合併的差異。如果部分模型資料是程式碼，開發人員即可透過整合共用內容差異分析工具的 Rational Rhapsody 解決方案，選用適當的差異分析工具，包括 Rational ClearCase 軟體所提供的工具及其他 CM 工具。但只有在無法自動合併變更時才需要這麼做。

圖 6 右邊的圖表則說明如何顯示無法解決的變更。您可逐步執行這些變更，以建立合併項目。

此功能整合了其類型管理程式及 Rational ClearCase 應用程式，因此自動合併功能的呼叫及 Rational Rhapsody 軟體的差異分析/合併其實是透過 Rational ClearCase 軟體指令行或版本樹狀結構來完成。

程式碼導向工作流程

Rational Rhapsody 的同步化功能可確保模型已反映程式碼變更，而程式碼中亦反映了模型變更。

對多數開發人員而言，有效編碼十分重要。Rational Rhapsody 解決方案可提供一系列功能強大的工具，不僅更方便編碼，甚至可讓您設計模型，以自動輸出程式碼，不必手動撰寫。無論選擇何種方法，Rational Rhapsody 軟體可確保程式碼及模型始終同步，而設計說明文件則是實作的確切呈現。程式碼導向工作流程可細分為同步化，以及透過視覺化重複使用元件。

產品特色

IBM Rhapsody 可協助 Rational Rose 使用者重複使用其智慧財產，同時善用 Rhapsody 的加強型建模及測試功能。

程式碼同步化

Rational Rhapsody 的程式碼同步化功能可讓您以最熟悉且習慣的方式產生程式碼，同時確保設計和模型仍同步。您將程式碼輸入編輯環境時，系統也會自動更新模型。「程式碼遵守」功能可確保建模環境遵守原始檔中的資訊放置。換言之，如果將資料和作業加入檔案的特定位置，整個後續設計作業中，該資訊都會保持在相同位置。

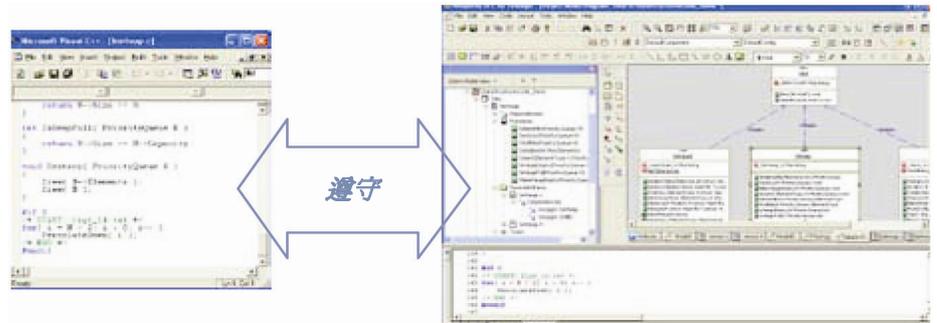


圖 7. 程式碼同步化：無論更新程式碼或模型，Rhapsody 應用程式環境都會將其同步處理，並遵守程式碼放置。

建模可協助開發人員視覺化其程式碼，以瞭解整個系統架構。

視覺化

軟體重複使用對 Rational Rose 使用者而言尤其重要，因為多數軟體是根據 Rational Rose 應用程式建立的設計規格所撰寫。雖然有時必須重新建立新的系統元件，很多時候 Rhapsody 軟體都可以重複使用可靠的智慧財產，提供比競爭對手更強大的優勢。

產品特色

Rational Rhapsody 軟體的元件導向設計功能有助於建立可重複使用的元件，以加速設計並促進協同作業。

這些智慧財產通常是程式碼。在嵌入式系統中，程式碼往往是以 C++、C 或 Java™ 語言撰寫的。您可使用模型視覺化程式碼，加強對架構的瞭解。必要時，您甚至可將程式碼還原為模型，以便擷取智慧財產並調整為進行中的設計部分。此外，模型也可提供自動設計說明文件，以便您將時間投入在設計和實作上，不需要花太多時間撰寫文件。

元件導向設計

元件導向設計會使用建模方法建立隨插即用的開發環境，將軟體建立為可重複使用的元件。Rational Rhapsody 透過下列功能提供這種開發方法：

- *UML 2 相容性*
- *儲存類別、子系統或其他軟體單元等個別元件的功能*
- *匯出或參照元件至其他專案的功能*

這些技術組合可加強協同作業和重複使用功能，讓團隊加速設計流程。

元件導向設計的另一項好處是可針對絕大部分即時目標，建立完整的應用程式，而非程式碼架構。至於納入第三方或其他群組元件的建置程序，Rhapsody 軟體可提供功能豐富的指令行介面，以便輕鬆納入建置指令碼。此工具可自動產生 make 檔（有時是個難題），而您也可以指定如何產生建置檔的精確規範。



圖 8. Rhapsody 應用程式架構

圖 8 說明簡化的 Rhapsody 應用程式。Rhapsody 應用程式代表用於建立系統模型的 UML 2 或其他特定領域設定檔。有了 Rhapsody 軟體，您即可建立完整的程式碼行為模型，包括狀態機及活動圖行為。您也可以使用圖形元件圖表等視覺化工具，輕鬆整合舊版程式碼。然後從主機環境快速重新調整至實際的目標系統，如 Linux[®]、Wind River VxWorks、16 位元作業系統 (OS) 或完全沒有 OS。

產品特色

Rational Rhapsody 可協助企業使用 MDD 方法，更快速、有效地開發優質系統。

Rational Rhapsody 應用程式不限於特定目標基礎作業系統。雖然 Rhapsody 應用程式一律使用基礎作業系統的服務，但僅此而已。Rational Rhapsody 是透過輕巧的擷取層，針對特定作業系統建立應用程式，來達到此目的。使用 Rational Rhapsody 環境的功能表，即可輕鬆選取具體的作業系統。此機制可延伸到絕大多數的作業系統。

總結

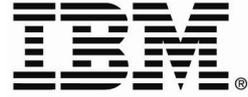
若改用 Rational Rhapsody 並將 MDD 的最新功能納入開發環境，將有助於提高生產力、系統效能及上市時間，同時降低成本。MDD 方法的主要好處包括：

- 使用模型導向測試減少開發初期的錯誤。
- 透過以 UML 2 為基礎的特定領域建模加強組織內部的溝通。
- 簡化並自動化文件建立。
- 改善複雜的系統整合及團隊協同作業。
- 透過程式碼導向工作流程，重複使用目前軟體開發流程中的現有 IP。
- 透過適用於幾乎所有平台及現有工具基礎架構的隨插即用方法，加速應用程式開發。

更多資訊

如需進一步瞭解 IBM Rational Rhapsody，請聯絡 IBM 業務專員或 IBM 事業夥伴，或造訪：

ibm.com/software/rational



© Copyright IBM Corporation 2009

台灣國際商業機器股份有限公司

110 台北市松仁路 7 號 3 樓
市場行銷處：0800-016-888 按！

台灣印製
2009 年 12 月
版權所有。

IBM、IBM 標誌、ibm.com、Rational、Rhapsody、Rose、ClearCase、RequisitePro 及 DOORS 是國際商業機器股份有限公司 (IBM) 在美國和/或其他國家或地區的商標或註冊商標。若此等及其他 IBM 註冊名稱在本文件首次出現時標有商標符號 (® 或 ™)，則代表其在本文件出版當時，係 IBM 所擁有的美國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一般法律商標。最新的 IBM 商標清單，請見「版權與商標資訊」網站：ibm.com/legal/copytrade.shtml

Microsoft 是 Microsoft Corporation 在美國及(或)其他國家或地區的商標。

Java 及所有使用 Java 的商標及標誌是 Sun Microsystems, Inc. 在美國及(或)其他國家或地區的商標。

Linux 是 Linus Torvalds 在美國和/或其他國家的註冊商標。

其他公司、產品或服務名稱可能是其所屬公司的商標或服務標誌。

本文所提及的 IBM 產品、程式或服務，並未暗示 IBM 會在設有 IBM 據點的所有國家提供此類產品、程式或服務。

本文所含資訊目的僅在提供資訊，雖然本文已力求所含資訊的完整與正確，但這些資訊是依「現狀」提供，不代表任何明示或暗示之保證。此外，本資訊乃以 IBM 現行產品計畫和策略，而 IBM 可能改變這些計畫而無須事先通知。若因使用本文資訊或其他文件，而蒙受損失，或這些損失與文中資訊無關，IBM 概不負責。本文不表示 IBM (或其供應商或授權商) 有意提出任何保證或陳述，或具有這些行為之效果。本文亦不表示 IBM (或其供應商或授權商) 有意更改規範使用 IBM 軟體之適用授權協定之條款，或具有這些行為之效果。