

行政院國家科學委員會補助專題研究計畫成果報告

行動計算系統中資源管理及容錯軟體元件 之研究與實作

計畫類別： 個別型計畫 整合型計畫

計畫編號：NSC89-2213-E-002-114

執行期間：89年08月01日至90年07月31日

計畫主持人：郭斯彥

共同主持人：

本成果報告包括以下應繳交之附件：

赴國外出差或研習心得報告一份

赴大陸地區出差或研習心得報告一份

出席國際學術會議心得報告及發表之論文各一份

國際合作研究計畫國外研究報告書一份

執行單位：台灣大學電機工程學系

中 華 民 國 90 年 10 月 31 日

行政院國家科學委員會專題研究計畫成果報告

行動計算系統中資源管理及容錯軟體元件之研究與實作

Research and Implementation of Resource Management and Fault Tolerant Components
in Mobile Computing Systems

計畫編號： NSC89-2213-E-002-114

執行期限： 民國89年08月01日到90年07月31日

主持人：郭斯彥 台灣大學電機工程學系教授

一、中文摘要

隨著通訊技術的發展，透過固定網路以及無線網路來傳輸資訊，已發展出形形色色的應用。為使得分散式計算環境得以提供使用者各種資訊的高度可及性，適當的容錯機制是保證應用伺服器穩定運作的必要條件。此外，為提供高效能的系統服務，系統資源管理的議題更是不可忽視。為了有效率地建構一個行動計算系統，我們將使用軟體元件技術作為系統建設的基礎。軟體元件技術的特性是可經由適當地組合已經存在的軟體元件，來建構我們所需要的系統功能，因此可以大幅降低系統建設的複雜度，並節省系統開發時間。目前較為普遍使用的元件技術有COM/DCOM、CORBA、JavaBean。我們將比較這三種技術的優劣點，尤其對於行動計算環境之支援度作徹底之評估。在本子計畫中，我們首重的議題是對於由元件技術所建構的行動計算系統，其中系統資源管理及容錯服務之研究與實作。我們將研究分析行動計算系統與傳統分散式計算系統當中的差異，並提出新的架構使其能符合行動計算環境的需求。

關鍵詞：分散式計算，行動計算，容錯，資源管理，軟體元件技術。

Abstract

As mobile communication services become popular, information becomes ubiquitously available wherever we are located. To achieve high availability for users in distributed computing systems, fault tolerant mechanisms are vital for robust

application servers. Also, to provide high system performance, the issue of resource management should not be overlooked.

In order to build the mobile computing system in an efficient manner, we will utilize one of the popular component technologies which alleviate developer efforts to build a complex system from scratch. Among the component technologies, we plan to compare pros and cons of COM/DCOM, CORBA, and JavaBean, and then the issue of possible support for mobile computing environment is surveyed. In the sub-project, we will concentrate on resource management and fault tolerance issues for mobile computing systems built by service components. The primary propose is to analyze the special features of distributed component object models for a mobile computing environment to enhance its performance and reliability.

Keywords: distributed computing, mobile computing, fault tolerance, resource management, component technology.

二、計畫緣由與目的

行動計算乃分散式計算中一個重要的領域。近年來，由於可攜式電腦設備的技術日趨成熟，筆記型電腦、行動電話、個人數位助理 (PDA) 等具有高度移動性的行動設備已廣為大眾所使用；加上通訊技術日新月異的發展，透過無線網路介面連接上網，使得資訊的取得不再受到時間與地點的侷限。在行動計算環境中，行動設備將透過無線通訊頻道去存取資料庫，而行動設備因為受限於本身電力供應以及

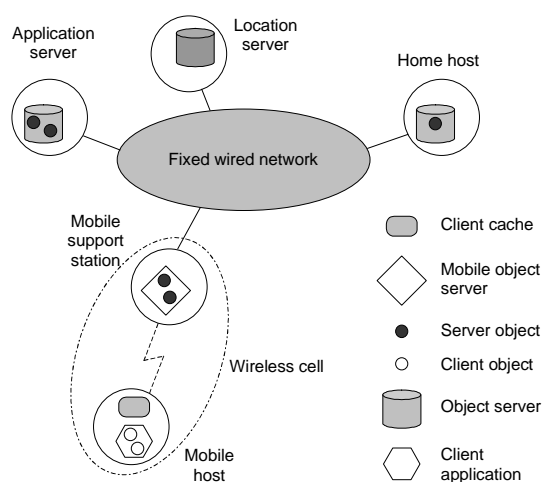
無線網路頻寬的限制，大部分時間將與無線網路維持離線狀態。加上行動設備具有高度移動性，它常在不同的無線網域 (wireless cell) 中登入與登出，並且連結到許多不同的資料庫伺服器。因此，在行動計算環境中，傳統分散式計算所依據的一些基本性質不再適用；行動設備的可移動性導入了許多新的議題，需要我們加以深入探討。

為了提供行動計算一個充分發揮其效用的環境，除了上述的無線網路及行動設備等硬體設施外，適合行動計算的資訊軟體基礎建設也是不可或缺的。然而，在軟體方面，尤其是提供相關資訊服務的軟體發展，卻遠遠落後硬體發展。這是因為傳統的資訊服務技術和系統，都是為有線網路而設計。對於具有移動性、高度變異性、以及個人化特色的行動計算環境並不適用。另外，由於系統開發牽涉到的層面相當廣泛與複雜，如果系統建構的每個部分都從頭做起，勢必耗日費時，十分費力。如果我們能採用促使硬體快速發展相同的原理，利用已開發的軟體元件來建構系統，則可以省時省力。因此，本整合型計畫將採用商用軟體元件 (Commercial Off-The-Shelf components/COTS components) 來建構系統，發展行動資訊服務軟體元件與組合技術，達到遍在計算的理想。簡言之，本整合型計畫的目標在於研發行動計算環境中，提供資訊服務軟體元件和組合之關鍵技術，並建立以軟體元件為基礎之行動資訊系統應用程式發展架構和執行環境。

三、研究方法與成果

近年來，在電腦及電信產業當中最受矚目的趨勢之一，就是行動通訊與計算的整合。這個分散式網路環境中的行動計算系統，與傳統有線電腦網路有著許多的相異之處。因為無線連網的技術使得具有計算能力的電腦設備不再受限於有線的網路介面，行動計算的理想得以實現。行動計算系統的主要組成，可分為固定網路及無

線網路兩部分。圖一為一個行動計算系統的架構示意圖。在固定網路上有許多固定主機作為伺服器，或是具有無線網路存取介面的是所謂行動支援站 (mobile support station)，兩種網路環境之間的通訊服務就由行動支援站所提供。在無線網路上則分為許多的 wireless cell，只要行動主機在 wireless cell 的涵蓋範圍之內，就可以藉由無線介面與行動支援站進行通訊。而行動主機上也擁有參與行動計算的應用程式。



圖一、行動計算系統架構示意圖

在這樣的架構下，我們希望應用程式的各個子行程可以選擇是存在同一部伺服器上或者是分散於跨越網路連結的不同主機，甚至能將部份的工作交由參與運算的行動主機來執行。不管是哪一種方式，我們都需要一種特別的方式或協定才能夠使用在不同行程之間的資料。作業系統通常會提供某種行程間的溝通機制，而近幾年來由於分散式應用元件技術的發展，使得在不同主機上的行程亦能方便的進行訊息與資料的交換，例如 Microsoft 提出的 Distributed Component Object Model (DCOM) 或 OMG 提出的 Common Object Request Broker Architecture (CORBA) 規格就提供了讓不同行程間的資料溝通的機制。這將對於我們在具有高度可靠度與執行效能的行動運算系統設計上提供了有效率與方便使用的建構工具。

為提供行動計算系統中各種資訊的高度可及性 (availability)，我們必需專研適當的容錯機制以保證應用伺服器穩定運

作。此外，為提供高效能的系統服務，系統資源的管理更是不可忽視。再者，由於行動計算本身具有高度可移動性的特徵，服務元件的名稱登記與導向也十分重要。為此，本子計畫的重要發展項目則可大略分為下列三個項目：

(一) 容錯服務

為提供使用者資訊取得的高可及性，我們必須考慮在系統元件加入適當的容錯機制。對於以商業軟體元件為基礎而發展的行動計算系統，在不同的系統層級上，可能遇到的錯誤情況有：軟體元件的錯誤動作、軟體元件間的不相容情形、伺服器應用程式的失序、伺服器端與客戶端應用程式版本衝突、伺服器端與客戶端之連線無預警中斷等等。這些問題與軟體元件整合架構以及分散式物件平台執行環境皆有關聯，因此本子計畫將輔助解決其他子計畫所可能面臨之問題，研究可行之容錯機制，實作成為提供容錯功能之服務元件，與其他系統元件進行整合，從中截斷可能造成錯誤之事件，或使系統在錯誤發生之後儘可能地迅速恢復正常運作。

(二) 資源管理服務

隨著使用者數目的增加，伺服器端的負荷也隨之上升。為了有效達到系統資源之充分與合理的利用，我們需要提供系統資源管理服務元件。在以軟體元件為基礎的行動計算系統中，我們可能面臨的資源管理方面之議題有：服務元件如何取得與回收系統資源、資源分配的優先順序、資源共享的程度、系統負荷的評定與負載平衡的機制等等。資源管理服務元件將配合應用軟體服務元件及網路與基礎建設元件之發展，甚而影響服務仲介 (service broker) 之設計與運作，確保系統資源之有效利用。

(三) 名稱管理服務

在行動計算系統中，無論是伺服器端或客戶端都擁有可移動性，為提供使用者有效的服務資訊，我們需要提供名稱管理服務元件。使用者可透過名稱管理服務元件去探詢可供使用的元件。因此客戶端不需要由自身去追蹤伺服器元件到底位於何處；伺服器元件可移動性的問題，就交由名

稱管理服務加以處理。目前與此項議題相關的應用有 OMG 的 CORBA Naming Service 和 Sun 的 JINI 等技術，我們可利用此機制增加本研究計畫的實用性。

為了要兼顧經濟與效率，開發一個大又複雜的軟體系統，許多機構已開始採用商用軟體元件 (COTS) 作為軟體系統的基本建構方式。很明顯的，造成這股趨勢的動力，就在於經濟效益。

由商用軟體元件為基礎 (COTS-based) 的軟體系統，指的就是一個軟體系統主要是由一組的商用軟體元件組合在一起所建構而成。負責系統建構者 (integrator) 所需做的事，不若傳統程式設計師親自撰寫程式去實現系統功能，而是去挑選適當的商業軟體元件，再以一些所謂結合格式 (glue code) 將這些元件彼此組合在一起。因此，這些軟體元件可能來自不同的商業來源，彼此之間也不一定能完美地相互配合。COTS software component 是購買自軟體市場，而且是在不更動其原始程式碼的前提下所使用。值得注意的是，或許有些 COTS component 允許使用者對其原始程式碼進行一些小部分的更動，但一旦使用者去更動了 COTS component，所謂 Off-The-Shelf 的性質便即失去。但有些 COTS component 允許使用者不需修改原始程式碼，改而採用 plug-ins、scripting、inheritance 的方式來達成使用者自訂的功能。

採用 COTS-based system 的優點，主要在於因為 COTS component 若久經各機構所使用，則其技術也愈成熟，比起自行設計所得元件要更可靠。而 COTS component 的維護與訓練費用相當於由許多的客戶共同分擔，因此對於單一客戶的在元件上所需負擔的成本相對不高。另外，對於專門開發商業軟體元件的公司對於新技術與新標準的採用常較為積極與迅速。

相對於這些優點，採用 COTS component 的主要風險在於使用 COTS component 來開發軟體的人通常無法直接對於元件的原始程式碼進行修改。也就是

說，要對於這些元件進行分析、使用、測試，都必須要在黑箱 (black box) 狀態下進行，我們無法得知元件當中到底是如何動作的。再者，商用軟體元件的使用者對於元件將來之發展沒有控制權。使用者不一定能夠取得商用軟體元件之完整且正確的行為規格。最後，許多的 COTS component 不一定能夠相互匹配且元件功能上可能與需求不符，以致系統功能必須遷就於功能不足的元件。

由以上所列這些可能的風險，使用商用軟體元件作為系統建構基礎的方式，仍然面對了許多有待解決的問題。因此，本計畫藉由實驗系統的實作，將可深入這些問題的核心，提供許多可能的解決方案。

本計畫第一年度首先擬與其他子計畫配合，共同對於各種分散式軟體元件架構作深入的探討。研究範圍包含有 COM/DCOM、CORBA、JavaBean 等分散式元件物件模型。我們進行了規格、執行效率分析與容錯能力的比較，同時配合元件實作以驗證其運作方式，以期能更深入了解其元件架構之交互運作模式。另外我們將對於各種分散式軟體元件技術在行動計算的支援程度上進行探討，以便將來能選定最適合的技術，作為發展以商用軟體元件為基礎之行動計算系統的核心技術。

同時，我們也將對軟體市場中可取得之商用軟體元件進行了解，依其提供之功能先用來實作 standalone 之小型應用程式。接下來，我們將進行服務狀況模擬，尤其針對系統資源的使用及容錯能力進行測試，以作為選取適當元件來建構系統之參考指標。

四、結論與討論

為了徹底發揮行動計算帶來的好處，我們必需了解行動主機 (mobile host) 與基地台 (base station) 之間的通訊連結受到許多因素的影響，例如行動主機的移動或電波受到建築物的干擾，因此具有相當之不確定性。而且因為行動主機的位置時常移動與連線的中斷或恢復，使得相對

應之網路拓樸 (network topology) 一直變動。另外，隨著行動主機所在位置的不同，其可獲得的頻寬為有限且不固定。因此無線頻道的重複使用的概念變得十分重要，如此才得以提供充足的系統服務。最後，因為行動主機的電力供應是有限的，所以必須要減少傳送與接收的動作以節省能源。對於上述的問題，我們將探討其特性，並配合能有效增加系統可靠度與系統管理效能的分散式應用元件架構，逐步建構在行動計算系統中具體的解決方案。

五、參考文獻

- [1] Larry T. Chen and Tatsuya Suda, "Designing Mobile Computing Systems Using Distributed Objects", IEEE Communications Magazine, Feb. 1997.
- [2] Mark Vigder, "An Architecture for COTS Based Software Systems", NRC Report No. 41603, National Research Council of Canada, Nov. 1998.
- [3] Dale Rogerson, "Inside COM", Microsoft Press, 1996.
- [4] Imrich Chlamtac and Yi-Bing Lin, "Mobile Computing: When Mobility Meets Computation", IEEE Trans. on Computers, Vol. 46, No. 3, Mar. 1997.
- [5] R. L. Carter, D. St. Louis, Ed P. Andert Jr., "Resource Allocation in a Distributed Computing Environment", 17th Digital Avionics Systems Conference, Vol. 1, 1998.
- [6] P. D. Le, B. Srinivasan, V. Malhotra, and N. Mani, "Resource and Load Sharing in Mobile Computing Environments", IEEE Region 10 Int'l Conference on Global Connectivity in Energy, Computer, Communication and Control, Vol.1, 1998.
- [7] M. G. Rubinstein and O. C. Duarte, "Service Location for Mobile Agent Systems", IEEE Int'l Telecomm. Symposium, Vol. 2, 1998.
- [8] J. Siegel, "COBRA Fundamentals and Programming", John Wiley & Sons, 1996.