

服務流程建置方法之評估：以地政整合作業為例

郝道康

國立中山大學資訊管理學系

ester.yu@gmail.com

李建祥

國立中山大學資訊管理學系

d954020001@student.nsysu.edu.tw

黃三益

國立中山大學資訊管理學系

syhwang@mis.nsysu.edu.tw

摘要

服務導向架構(SOA)是提升企業敏捷性之關鍵資訊技術，藉由再用與組合網路服務(Web service, WS)組成商業流程，是發揮 SOA 效益的重要方法。已有些研究提出再用既存服務組合(service composition)，提升 WS 的再用率和組合 WS 的效能，但仍缺乏實證的研究評比這些方法的優劣。本研究的目標是提出一些具體的指標，以評比前述不同方法之效能。本研究評估了人工組合 WS、參考類似服務組合、和實體化服務範型(service pattern)等三個方法，並以地政整合系統的實際案例，比較上述三方法在組合 WS 耗費的時間與其產出的正確性等指標的差異。結果發現使用服務範型，其平均耗費時間與正確性優於其他方法。

關鍵字：網路服務組合，服務範型，商業流程塑模符號，變異數分析，地政管理

服務流程建置方法之評估：以地政整合作業為例

(一) 緒論

現代企業重視敏捷性以面對持續變化的市場挑戰與機會，以服務導向架構(service-oriented architecture, 簡稱 SOA)封裝商務邏輯為一個個網路服務(Web service, 後文簡稱 WS)，並藉由快速組合既存的 WS，以較低的成本實現新需求，是提升企業敏捷性的關鍵技術(Cummins 2008; Erl 2005)。以目前當紅的雲端運算的軟體即服務(Software as a Service, SaaS)商業模式為例，也是以 WS 的方式來提供服務，於是系統中的商業流程可以由不同的 WS 組合而成。雖然，目前已有一些軟體工具(如：SAP 的 Netweaver)和業界標準(如：商業流程塑模符號, business process model and notation 簡稱 BPMN)可以輔助軟體開發人員建構以工作流程為基礎的網路服務組合(workflow-based Web service composition)，但仍須耗費可觀人力於設計工作流程、尋找適合的 WS、以及測試產出的服務組合等工作。為減輕人力的負擔，有些研究採取再用既存網路服務組合(Web service composition, 後文簡稱 WSC)的策略，透過調整既存 WSC 以產生新的 WSC。這方面的研究，可分成二大類，一類是以既存 WSC 的工作流程為樣版(workflow template)，透過置換其中的組成 WS 或改變子工作流程，建構新的 WSC(Chafle et al. 2006; Geebelen et al. 2008a; Gil et al. 2009);另一類的方法是實體化一個兼容通用與可調適子流程的抽象流程，稱為服務範型(service pattern)，透過設定組態參數值以實體化抽象流程，產生新的 WSC(Fu et al. 2009; Lee et al.)。前述方法提升了 SOA 的再用層次，從傳統單元功能之再用，提升到流程設計的再用。因為既存工作流程被再用，不僅縮短設計時間，也可減少整合測試的範圍，其效益可以反應在數個軟體開發的階段。再者，企業可以將核心商務邏輯表達為流程樣版或服務範型，保留其最佳設計實務，是維持競爭優勢的利器。換言之，WSC 的再用不僅能提升組合 WS 的效能，而且可以作為企業的策略工具，因此，WSC 再用是值得運用於企業的 SOA 發展。

然而，之前的研究並沒有實證再用 WSC 的方法是否具有效益；再者，前述二類再用 WSC 的方法因沒有一個共通的比較基準，無法評斷何種方法在實務上能發揮較佳的效益。因此，本研究的目的是提出一個評估方法，實證再用 WSC 方法的效益，並評估前述二類再用 WSC 方法的優劣，以作為企業導入 SOA 時，選擇 WSC 再用方法的參考。因此，本研究採用實驗法評比三個組合 WS 的方法，分別是人工組合 WS、參考類似 WSC、和實體化服務範型(service pattern)等。第一個方法僅再用 WS，而後二個方法則是再用 WS 和 WSC，透過二個量化指標來比較這三個方法的差異，它們分別是組合 WS 所耗費的時間和產出的 WSC 之正確性等。比較三個方法在前述二個指標的表現，以評斷再用 WSC 的方法是否優於只再用 WS

的方法，以及二種再用 WSC 的方法何者較優。實驗設計方面，招募了十八位有經驗的軟體開發人員進行實驗，並以地政整合系統的實際案例，設計三個不同難易度的問題供受測者作答，量測其作答時間並由領域專家就產出的服務結果評分。最後，以 ANOVA 分析資料並說明結果。

本研究的貢獻有以下二點：一、以量化的指標透過實驗法評比三個不同方法，發現實體化服務範型的方法在平均耗費時間與正確性優於其它二種方法，此一結果可供企業導入 SOA 之參考；二、採用的實驗設計方法可作為未來研究 SOA 相關議題，評比候選方法之參考，如：評估指標的選擇、案例的設計、和受測人員之評選等。

(二) 文獻探討

Service-Oriented Architecture(SOA)服務導向架構是 IT 產業架構演進中，發展出的一種遵循典範。SOA 軟體架構的基本目的就是幫助管理複雜的軟體系統，並且隨著不可避免的時間演進、商業流程修改、組織與科技改變之下跟著改進 (Rosen 2008)。基於 WS 而建立的企業流程架構，其價值在將可重複使用的 WS，可以靈活的、彈性的組合起來成為一個商業流程。以因應未來的擴充性、變化性與易於維護。簡單地應用在不同的商業流程中。

WS 是一個應用程序透過標準的 Web 協定，去訪問、叫用其他的應用程序 (Alonso 2004)。而 WS 本身具有可組合(composable)的特性(Rosen 2008)，可與其他 WS 組合成商業流程、增加 WS 的重複使用率。其特色就是模組化、自主性、重複使用率、可靠度，和降低搜尋成本等。基於模組化，每一個 WS 可以看成一個獨立的組件或元件。故可以將 WS 包裝起來(Yang et al. 2006)，按照商業邏輯及功能需求加以組合成新的工作流程，且使用已經經過驗證的 WS 也是讓整體工作流的可靠度上升。

但每當我們組合 WS 構成需要的商業流程時，總是會困惑於那些 WS 可使用、如何組合選擇的 WS、相似的流程卻常常重複或重頭設計、對於設計出的流程可靠度的重新驗證…等。於是有些研究，從再用既有 WSC 設計的方向著手，這方面的研究包含以流程樣版(workflow template)建構 WSC 與實體化服務範型(service pattern)等二類方法。

在流程樣版方面的研究，流程樣版被表達成一種抽象工作流程(abstract workflow)的形式，其中的元件 WS 可以是對應實體地址的 WS 或是只有介面規格的抽象 WS。藉由選擇能實現抽象 WS 規格的實體 WS，能建構可執行的 WSC 實例，至於選擇的準則可以是 WS 的服務品質(quality of service, 簡稱 QoS)限制，依據使用者給定的 QoS 需求，彙總各元件 WS 的 QoS 屬性值，選擇其中符合需求的 WS 以構成可執行的 WSC(Aggarwal et al. 2004)；或是在工作流程中不同的階段，建立好數個可供選擇的子流程或實體 WS，而在執行期依據環境參數值，決定選擇的對象，而環境參數值可由使用者決定或在執行過程中由呼叫 WS 所傳回的結果(Geebelen et al. 2008b; Gil et al. 2011)等，雖然這方面的研究允

許調整樣版流程以建構多個 WSC 實例來符合不同 QoS 的需求，但受限於元件 WS 規格是固定的而導致 WSC 的功能缺乏彈性。即使有些研究能在執行期動態決定執行的子流程或實體 WS，但這些選項必須預先設計於工作流程中，不僅工作流程複雜不易維護，且影響執行效能。

另一方面，服務範型於工作流程中抓出抽象的程序，捕捉出服務之間典型的交互作用。如此面對不同流程與參數的結構時便可以組合連結起具體的客製化工作流(Fu et al. 2009)。李建祥等人(Lee et al.)提出了服務範型模式(service pattern model)以定義一個高層次的服務範型，以富彈性的工作流程樣版來協助 WS 的組合以迎合不同的需求。服務範型模式制定了一個服務範型的規格表示法，一個服務範型包含：輸出入參數、前提條件(precondition)與事後效果(effect)、組態參數(instantiation parameter)、和可調適的流程樣版(adaptable workflow template)等，其中輸出入參數、前提條件、和事後效果類似於一般 WS 的介面規格，比較獨特的是組態參數與可調適的流程樣版。可調適的流程樣版定義一個可被再用並實體化出數個不同功能 WSC 的抽象流程，換言之，它可提供較高的功能彈性。促成可調適的流程樣版具備較高功能彈性的因素，是服務範型模式擴充了以下三類的流程建構元：

- 1 prospect service: 有別於一般 WS 有固定的介面規格，prospect service 提供可客製化的介面規格，讓開發人員依據不同的功能需求，賦予一個工作流程樣版中某個活動的服務介面有不同的輸出入參數型態或獨特的前提條件與事後效果。因為一個工作流程樣版中某些具變異性的活動之服務介面可以在實體化(instantiation)時才指定具體的介面規格，所以產生的 WSC 具有不同的功能。
- 2 可調適的控制元：這一類的控制元用於功能特色的選擇，它包含 skippable toggle 和 alternative selection 等二種選擇方式。前者用於指定工作流程樣版中某些可選擇性(optional)的服務，供開發人員決定是否納入該服務成為最終工作流程的一個活動或排除該服務；後者則用於多個替代子流程的選擇，對於某段有多個可行方案的子流程，可供開發人員選擇其中一個適合的子流程，完成不同的功能需求。
- 3 訊息連結(message alignment)：用於宣告需要進行資料交換的 WS 間之資訊，目的是為了在實體化服務範型的階段，依據此類的資訊進行資料型態的檢核，以確保最終的工作流程中所包含的客製化 prospect service 間的資料交換能順利進行。

為了使上述建構元能在實體化服務範型時，能透過設定的方式建構可執行的 WSC，服務範型模式也定義了三類的組態參數，分別對應到上述的建構元，讓開發人員依據需求設定參數值，再遵循作者們所提的實體化程序，可以系統性的建構出可執行的 WSC。圖 1 是一個支援銷售資料分析的服務範型例子，而圖 2 則是其可調適流程樣版的範例。另外，為了解決開發人員因為對服務範型或可用 WS 的規格不熟悉，而無法完整定義組態參數值的問題，作者們也開發了一套以

forward-chaining 方式的自動化推論方法，在給定的初始條件下，自動推論出可行的參數值，關於上述的詳細程序與推論方法可參閱(Lee et al.)。

Name: SalesCube pattern
Purpose: To support the commercial requirements which base on sales data with multiple currencies and units considerations.
Context: The applications not only require sales data aggregated by multiple dimensions but also extend other data to derive outcome.
Category: Behavioral pattern, Sales data
IN: TStore:In sl, TSalesClass:In ea, TTimePeriod:In tp, TMoney:In cu, TSalesClass:In ga, ?TypeOfEd In ed
OUT:?TypeOfOut Out
Constraints:
Precondition:
|In sl|≠0^Day(In tp)<CurrentDate()-7
Effect:
|Out|>0
Instantiation parameters:
Workflow adapting parameters:
?SkipConverByCurrency: Boolean;
?SkipConvertByUnit: Boolean;
?SkipDeriveData: Boolean;
?SkipAggregateData: Boolean;
?ChooseFormat: {‘Cube’, ‘Flat’};
?ChooseComputeOrder: {‘AggregateFirst’, ‘DeriveFirst’};
Data type parameters:
?TypeOfEd, ?TypeOfJedOut, ?TypeOfExtrIn, ?TypeOfExtrOut,
?TypeOfDrvIn, ?TypeOfDrvOut, ?TypeOfAggIn, ?TypeOfAggOut,
?TypeOfOut
Restriction rules:
ModelRef(?TypeofEd)=Ont#Table,
ModelRef(?TypeOfExtrOut)=Ont#Table,
ModelRef(?TypeOfJed)=Ont#Table,
ModelRef(?TypeOfJedOut)=Ont#Table,
ModelRef(?TypeOfDrvIn)=Ont#Table,
ModelRef(?TypeOfDrvOut)=Ont#Table,
ModelRef(?TypeOfAggIn)=Ont#Table,
ModelRef(?TypeOfAggOut)=Ont#Table,
ModelRef(?TypeOfOut)=Ont#Table,
KeyAttr(?TypeOfJedIn)=KeyAttr(?TypeOfed)=KeyAttr(?TypeOfJedOut),
Attributes(?TypeOfJedOut)⊆Attributes(TSalesData)∪Attributes(?TypeOfEd)
Predicate parameters:
?fpSelectData(TSalesData:SelIn),
?cpDeriveDataPrecond(?TypeOfDrvIn: DrvIn),
?cpDeriveDataEffect(?TypeOfDrvOut: DrvOut),
?cpAggregateDataPrecond(?TypeOfAggIn: AggIn),
?cpAggregateDataEffect(?TypeOfAggOut: AggOut)

圖 1 一個支援銷售資料分析的服務範型(Lee et al.)

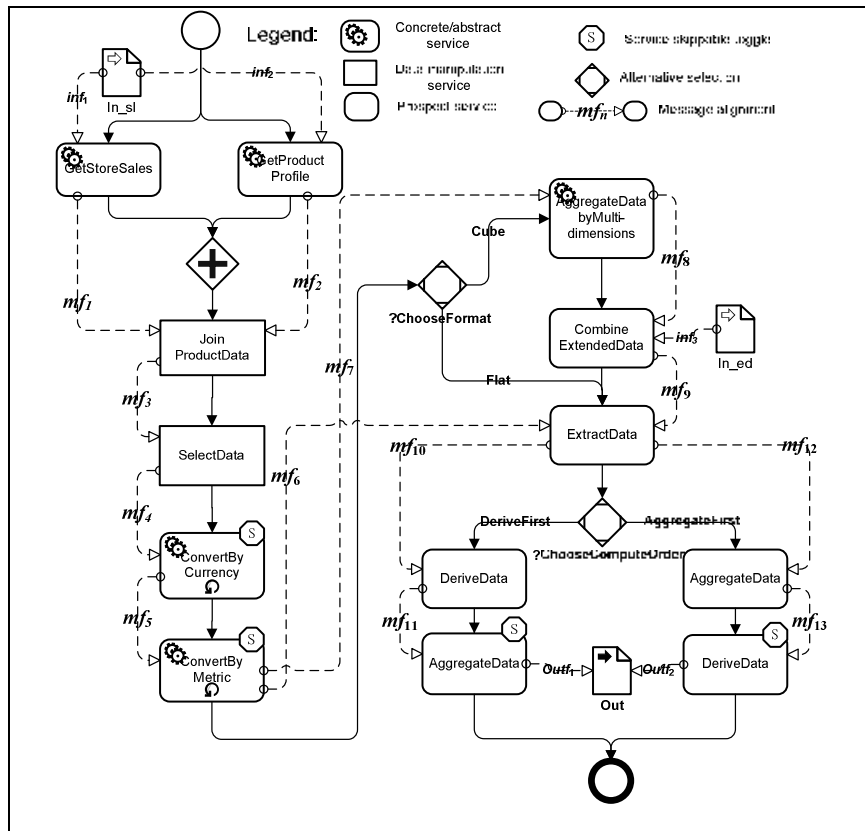


圖 2 搭配圖 1 服務範型的可調適工作流程樣版(Lee et al.)

(三) 研究方法

依據文獻與經驗，我們以地政整合系統的實際案例繪製出登記系統的各個工作流，然後歸納出 Service Pattern。然後透過與地政系統開發專業人士訪談討論後，設計出使用 General Service Pattern 為基礎的工作流題型。並且選擇於資訊領域相關業界人士，其學經歷均有大專相關科系，實際業界工作三年以上之受訪者予以受測。最後根據其受測結果給予該商業領域資歷十餘年的專業人士評斷其受測結果，最後整理受測結果及相關意見，統計歸納出結論。

1. 受測對象

Stump 和 Rabinowitz(1981)以專業年資劃分生涯發展階段，擢升階段 2-10 年間會進入該領域專業的階段。故本研究選擇的受測對象，均為在資訊領域擔任系統分析師以上的職位，專業領域年資三年以上，一共 18 位受測者。其所在領域有資訊公司員工，醫院、電信公司、電子製造業、傳統製造業的資訊人員、區域分布也位於高雄、台南、台中、新竹、大陸等地。所有的測試者彼此都是在獨立的情形下受測。每個受測者均是在不同時間、不同地點拿到試題。且試題方面組合方面也會有所不同。故受測者無法彼此討論，且有充分的受測時間以完成受測內容，達到獨立受測的環境。

2. 受測題目設計

為了確保其商業流程經驗的出發點為一致，避免受測者自身經驗延續的持久性效應，故選擇了受測者完全陌生的作業流程領域：地政整合系統，一個一般資訊人員較少機率接觸到，且獨特的商業流程。我們與前述的專家討論後，選擇了地政整合系統中的土地登記系統的實際案例來做為受測內容領域。與專家加以討論彙整之後，整理列表出登記流程中相關的 Web Service，繪製出三個基本架構的服務範型：土地樣版、建物樣版以及權利樣版。在建立出較高階層的 General Service Pattern 之後，接著在該領域中依據實際的案例與參數建立出三個不同實例的工作流題目：土地分割(以下簡稱 P1)、他項權利移轉(以下簡稱 P2)、變更更正(以下簡稱 P3)。同時利用前述的服務範型，針對這三個實例，再與前述專家一同繪製出正確答案的工作流，作為接下來受測之後評量結果正確性的依據。

同時間為了避免題型之間學習效應，導致受測結果有所偏差沒有辨識成果之能力，我們也設計不同的輔助工具讓受測者使用，以比較服務範型與其他不同工具的成效。三個工具分別為：從無到有，憑自身能力繪製出題目所指定的工作流(以下簡稱 T1)、給予相似案例參考，繪製出題目所指定的工作流(以下簡稱 T2)、給予三個服務範型，請受訪者依照服務範型繪製出題目所指定的工作流(以下簡稱 T3)。其中 T3 即為前述已經設計好基本架構的服務範型：土地樣版、建物樣版以及權利樣版。T2 是選定與 P1、P2、P3 同樣領域但是不同實際案例的工作流，利用 T3 的服務範型，依據實際案例的參數來繪製出相似案例的流程圖。其工作流程的相似度約四~五成，並且工作流中每個 web service 的順序並無不同，但是在具體或抽象的服務選用、忽略、路徑選擇、流程參數上會有所不同。

於是三個題目要給予受測者測試之前，利用三個題目與三個工具互相搭配，分別給予不同的受測者不同排列組合的題型。排列組合如下：

$$\begin{aligned} Q1 &= \{(T1, P1), (T2, P2), (T3, P3)\} & Q4 &= \{(T1, P3), (T2, P1), (T3, P2)\} \\ Q2 &= \{(T1, P2), (T2, P1), (T3, P3)\} & Q5 &= \{(T1, P2), (T2, P3), (T3, P1)\} \\ Q3 &= \{(T1, P1), (T2, P3), (T3, P2)\} & Q6 &= \{(T1, P3), (T2, P2), (T3, P1)\} \end{aligned}$$

受測問卷的第一題一定是採用 T1 的方式，第二題採用 T2 的方式，第三題採用 T3 的方式，每張受測問卷都搭配不同組合的題目，如同上面的組合。整份受測問卷準備完畢後，就會隨機選取符合受測資格的受測人員予以受測，每組給予三位受測者受測，一共 18 位。

3. 受測過程

受測前會依照試題的章節，花費約一個小時的時間循序漸進的介紹 BPMN 及 General Service Pattern 的相關知識及地政整合系統現有 Web Service 的簡介，使受測者有能力進行接下來的測驗。結束說明之後，開始給予充分的時間，請受測人員依據分配的題目及對應工具，以 BPMN 繪製題目要求的工作流程圖，並記錄完成各題目的花費時間。每個受測者獨自於不同時段，獨立空間受測，避免相互參考的情況，且受測期間擁有充分的休息時間，不會有疲勞效應。

4. 指標評量

完成實驗之後，我們會記錄每小題的繪製時間來做為衡量指標。除了時間之外，完成的工作流程圖會給予前述專家針對其正確性加以評斷給分，分數範圍介於 0 與 100 之間。利用正確性分數與時間加以分析使用 Pattern 的效益。

(四) 資料分析

1. 資料分析

在受測者結束實驗之後，我們將繪製指定工作流所花費的時間，使用二因子變異數分析的 ANOVA 模型來檢定使用不同工具是否有所顯著的差異。先進行過 Levene 檢定，確定三個母體之變異數相同之後，其時間的 ANOVA 結果如表 1 所示：

表 1 繪製時間 ANOVA 表

繪製時間 ($\alpha=0.05$)	F	P
題型	0.487	0.618
工具	3.685	0.033*
題型*工具	0.705	0.593

檢定後發現工具的檢定值為 3.685， $p\text{-value} = 0.033 < 0.05$ 。代表不同測試工具造成答題花費時間之不同有顯著差異。於是進一步使用 Tukey HSD 進行多重比較，如表 2 所示：

表 2 繪製時間多重比較表

Tukey HSD	工具 (I)	工具 (J)	平均差異 (I-J)	顯著性
	1	2	9.83	0.328
		3	18.50*	0.025*
	2	1	-9.83	0.328
		3	8.67	0.419
	3	1	-18.50*	0.025*
		2	-8.67	0.419

Tukey HSD 檢定可看出工具 T3 與 T1 的顯著性 $0.025 < 0.05$ ，平均差異 18.50 達正向顯著，代表著工具 T3 與工具 T1 有顯著的差異。顯示使用 Pattern 的工具之下的確能有效地減少系統分析師、系統設計師組合 WS 所花費的時間。

同樣的，我們也針對了繪製結果的正確性加以統計，使用二因子變異數分析的 ANOVA 模型來檢定。同樣經過 Levene 檢定母體之變異數相同之後，其正確性的 ANOVA 表如表 3：

表 3 受測結果正確性 ANOVA 表

正確性 ($\alpha=0.05$)	F	P
題型	0.456	0.637
工具	16.344	0.000*
題型*工具	0.722	0.581

檢定後發現工具的檢定值為 16.344， $p\text{-value} = 0.000 < 0.05$ 。代表不同測試工具造成答題花費時間之不同有顯著差異。於是進一步使用 Tukey HSD 進行多重比較，如表 4 所示：

表 4 受測結果正確性多重比較表

Tukey HSD	工具 (I)	工具 (J)	平均差異 (I-J)	顯著性
	1	2	-21.94*	0.001*
		3	-31.11*	0.000*
	2	1	21.94*	0.001*
		3	-9.17	0.240
	3	1	31.11*	0.000*
		2	9.17	0.240

Tukey HSD 檢定可看出工具 T3 與 T1 的顯著性 $0.00 < 0.05$ ，平均差異 31.11 達正向顯著，代表著工具 T3 與工具 T1 有顯著的差異。

接著我們將受測結果的花費時間除以正確性的分數來做探討。由於在前面的檢定已確定在不同工具下會對花費時間與正確性結果有所影響，故將其數值分別依照工具 T1、T2、T3，針對個別數值加以計算出平均數，可得到以下表 5 的結果：

表 5 時間/分數結果

	T1	T2	T3
時間/分數的平均數	2.2274	0.9325	0.6606

由兩段結果可以看出，以花費時間除以正確性評分之後，其比值均為 $T1 > T2 > T3$ 。換言之，在得到相同正確性評分的情況下，採用 T3 需要花費的時間比 T2 和 T1 都低。

2. 數據探討

經過多重比較之後發現無論是時間或是正確性，使用 T3 與 T2 之間的差異並不顯著，且在正確性分數的 Tukey HSD 檢定中，工具 T2 和 T3 同時達到顯著。我們從受測者的意見與建議中，得到以下結果：

(1) 為對 BPMN 塑模工具不熟悉，及對商業流程領域更為陌生。

雖然我們給予 BPMN、General Service Pattern 及地政整合系統的相關說明與教育，但受測前期的理解時間變長與理解程度有限的情形仍會發生。受測時常

發現受測者會回頭檢視 BPMN 可使用圖形與地政整合系統基本可用 Web Service。這導致了 T2 與 T3 的差異不顯著。

(2) 相似案例與題目相似性差異程度，受測者對其認知差異不大。

T2 提供相似案例的工作流，相似程度與題目正確流程的差異約四至五成左右。但由於受測地區(台灣)的資訊人員大多是以相似案例的方式建置、撰寫系統，故其自身的經驗會提高了工具 T2 的正確性，減少了作業時間。而對於 Pattern 建置方式較為陌生，於是降低了工具 T3 的正確性與增加作業時間，於此兩點原因拉近了工具 T2 與 T3 之間的程度，導致了不明顯的情況。

(五) 結論

面對開發腳步越來越快的現今社會，如何有效縮短設計開發時程是越來越重要的課題。再用既存 WS 和 WSC，有助於縮短系統分析師或系統設計師規劃商業流程的時程，因為利用已經驗證正確可靠的 WS 來加以組合應用，不僅縮短開發和測試時間，也提升了工作流程中每個環節的正確性；而 WSC 的再用，也達到善用既存商業流程之最佳實務(best practice)的效益。

本研究採用實驗法評估再用 WSC 的方法在組合 WS 的耗費時間與 WSC 的正確性等指標的表現，結果顯示再用 WSC 的二個方法都比只再用 WSC 的方法好，且具有統計的顯著性；另外，以服務範型為基礎的方法，其平均表現則優於再用類似 WSC 的方法，茲彙總本研究的貢獻如下：

1. 量化指標的評比結果可供企業導入 SOA 之參考

企業導入 SOA 著眼於提升其敏捷性，亦即較快的開發速度並正確地滿足新需求，故本研究提出了組合 WS 的耗費時間與 WSC 的正確性等兩個明確的衡量指標，評量只再用 WS 和同時再用 WS 與 WSC 等方法的表現。越少的耗費時間且有較高的產出品質，才是具有敏捷性的開發方法。經由本研究的實驗，得到資深系統分析師及系統設計師在使用服務範型所設計出具有商業邏輯的工作流程，其平均花費的時間和繪製出的工作流程正確性都比其它二種方法好。因此，企業導入 SOA 時，不應僅只著重於再用 WS，也應該將一些常用或核心的商業流程設計成服務範型，作為開發新 WSC 滿足新需求的主要方法。

2. 實驗設計方法可作為未來 SOA 相關實證研究之參考

一般的實驗設計需要考量一些內外部影響因素，例如：學習效應、互相參考、疲勞效應、持久性效應…等等因素，而 SOA 相關實證研究的受測者又需具軟體開發經驗，如何有效排除前述因素造成實驗的干擾，必須在受測對象的篩選、受測題目的設計、受測過程的安排，及受測結果的評量等方面整體性規畫。彙總第三節研究方法的描述，本研究依據 Stump 和 Rabinowitz 研究的建議，招募具有三年經驗的開發人員，設計三種功能無關且難易度不同的題目供受測人員作答，作答之前先進行一個小時的訓練並不限其作答時間，最後，請領域專家針對作答

結果進行評分。參考前述的實驗設計，未來的 SOA 相關實證研究，可降低一些內外部因素對實驗的不良影響。

另外，本研究的商業領域故意選擇受測者陌生不熟悉的政府部門地政作業流程，受測者藉由使用服務範型繪製出工作流程的正確性能達到滿意的結果。這表示在實務上可從服務範型中得知主軸的商業流程架構，在面對各式各樣不同需求與狀況的現實商業情境實例中，得以引申基礎流程建置出符合實例的工作流。在使用服務範型時使用者思索正確標準的工作流，對於培養新進系統分析師或系統設計師有相當程度的幫助。

本研究因受限於測試樣本數過少，導致實體化服務範型與參考類似 WSC 的資料分析不具有統計的顯著性，在未來可以招募更多受測人員，並改進人員的訓練方式，以進一步評量前述方法的差異。再者，商業流程通常是複雜及多變的，如何設計一個服務範型以涵蓋較大的功能範圍，或是何種功能的細膩程度 (granularity) 較適當，也是值得未來研究的方向。

參考文獻

- Aggarwal, R., Kunal, V., Miller, J., and Milnor, W. "Constraint Driven Web Service Composition in METEOR-S," in: *2004 IEEE International Conference on Services Computing*, 2004, pp. 23-30.
- Alonso, G. *Web services: concepts, architectures and applications* Springer, 2004.
- Chafle, G., Dasgupta, K., Kumar, A., Mittal, S., and Srivastava, B. "Adaptation in web service composition and execution," *IEEE International Conference on Web Services (ICWS 2006)* 2006, pp 549-557.
- Cummins, F. *Building the Agile Enterprise: With SOA, BPM and MBM* Morgan Kaufmann, 2008.
- Erl, T. *Service-oriented architecture: concepts, technology, and design* Prentice Hall PTR, 2005.
- Fu, J., Bastani, F.B., Yen, I.L., and Hao, W. "Using Service Patterns to Achieve Web Service Composition," *2009 IEEE International Conference on Semantic Computing* 2009, pp 402-407.
- Geebelen, K., Michiels, S., and Joosen, W. "Dynamic reconfiguration using template based web service composition," in: *Proceedings of the 3rd workshop on Middleware for service oriented computing*, ACM, Leuven, Belgium, 2008a, pp. 49-54.

- Geebelen, K., Michiels, S., and Joosen, W. "Dynamic Reconfiguration Using Template Based Web Service Composition," 3rd Workshop on Middleware for Service Oriented Computing, 2008b, pp. 49-54.
- Gil, Y., Groth, P., Ratnakar, V., and Fritz, C. "Expressive reusable workflow templates," *the 5th IEEE International Conference on e-Science (e-Science 2009)* 2009, pp 344-351.
- Gil, Y., Ratnakar, V., Kim, J., Gonzalez-Calero, P., Groth, P., Moody, J., and Deelman, E. "Wings: Intelligent Workflow-Based Design of Computational Experiments," *IEEE Intelligent Systems* (26:1) 2011, pp 62-72.
- Lee, C.H., Hwang, S.Y., and Yen, I.L. "A Service Pattern Model and its Instantiation Using Rule-Based Reasoning," *UTD Technical Report UTDCS-05-12*, Feb 2012.
- Rosen, M. *Applied SOA: service-oriented architecture and design strategies* Wiley, 2008.
- Yang, J., and Papazoglou, M. "Web component: A substrate for web service reuse and composition," Springer, 2006, pp. 21-36.

Abstract

Service-oriented architecture is a crucial technology to aid enterprises in improving their agility. By reusing and composing existing Web services, enterprises can quickly build workflow-based Web service compositions to fulfill new requirements at lower cost. There have been academic works that propose different methods of reusing existing service composition to improve the reusability of Web services and the efficiency of composing Web services. However, there is no evidence to confirm which is better. In this paper, we apply experiment method to evaluate three methods, namely creation from scratch, creation from similar service composition, and instantiating service pattern, by considering composition time and composition accuracy in the context of lands management. The results show that instantiating service pattern outperforms the other methods.

Keywords : Web service composition, Service pattern, BPMN, ANOVA, Lands management