

# 行政院國家科學委員會專題研究計畫 成果報告

## 有使用者意圖感知能力以服務為導向系統之研究(第2年) 研究成果報告(完整版)

計畫類別：個別型  
計畫編號：NSC 96-2218-E-343-002-MY2  
執行期間：97年08月01日至98年07月31日  
執行單位：南華大學資訊工程學系

計畫主持人：李俊宏

計畫參與人員：碩士班研究生-兼任助理人員：吳啟仲  
碩士班研究生-兼任助理人員：黃彥傑  
碩士班研究生-兼任助理人員：陳昱志  
碩士班研究生-兼任助理人員：洪偉智  
碩士班研究生-兼任助理人員：巫育麟  
大專生-兼任助理人員：陳韋臣  
大專生-兼任助理人員：張純榕  
大專生-兼任助理人員：陳君儀  
大專生-兼任助理人員：吳松展

報告附件：出席國際會議研究心得報告及發表論文

處理方式：本計畫可公開查詢

中華民國 98 年 11 月 02 日

行政院國家科學委員會補助專題研究計畫 成果  
報告

有使用者意圖感知能力以服務為導向系統之研究

計畫類別：■ 個別型計畫      □ 整合型計畫  
計畫編號：NSC 96-2218-E-343-002-MY2  
執行期間： 96 年 8 月 1 日至 98 年 7 月 31 日

計畫主持人：李俊宏

計畫參與人員：吳啟仲 洪偉智 黃彥傑 陳昱志 巫育麟 陳韋臣  
張純榕 陳君儀 吳松展

成果報告類型(依經費核定清單規定繳交)： 完整報告

本成果報告包括以下應繳交之附件：  
出席國際學術會議心得報告及發表之論文各二份

執行單位：南華大學資訊工程系

中 華 民 國 98 年 10 月 31 日

1. 前言.....	3
2. 研究目的.....	3
3. 文獻探討.....	6
3.1. 本體論(Ontology).....	6
3.2. 人工智慧規劃(AI-Planning).....	7
3.3. 網路服務選擇系統.....	8
3.4. 服務組合(Service composition).....	8
3.5. 案例式推理(Case-based Reasoning).....	9
4. 研究方法.....	10
4.1. 領域本體論與個人化本體論之建立.....	10
4.2. 目標模型與目標模型之建立.....	16
4.3. 使用者意圖擷取機制.....	18
4.4. 使用者意圖滿足機制.....	19
4.4.1 使用者意圖滿足機制之實現.....	20
4.4.2 系統執行流程說明.....	21
4.5. 系統整合.....	22
5. 結果與討論.....	23
6. 文獻列表.....	24
<b>計畫成果自評</b> .....	<b>27</b>
<b>可供推廣之研發成果資料表</b> .....	<b>28</b>

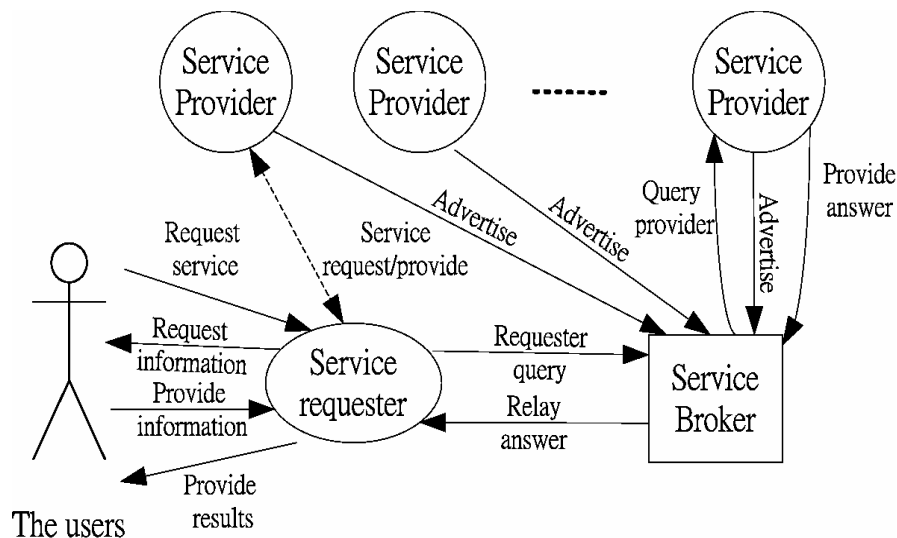
## 1. 前言

網路服務(Web Service)協定之訂定，以及其語意網(Semantic Web)之結合，可望帶來新一波的網路革命以及大量的商機。不管是國內外對此一領域相關技術的發展均極為重視。而一個計算機系統的長期目標與挑戰是如何提供使用者更多樣化和更豐富的服務，並且讓使用者在呼叫與使用這些服務時能夠如同請人類助理幫忙般的方便。在這些新技術激勵下引出了本研究的想法。

經歷二年之開發與研究，本計畫於第一年主要開發以服務為基礎之人機界面相關軟體元件並探討過程中之相關問題。我們將所開發出來的軟體系統稱為具意圖感知能力的服務導向的系統(Intention-Aware Service-Oriented System, IASOS)。IASOS將會擷取使用者使用網路服務之意圖。並根據系統能力，規劃一系列動作，以滿足使用者之意圖。以便利系統之使用。而在第二年我們則以第一年的經驗為基礎發展以IASOS架構為基礎的SOA系統開發方法。

## 2. 研究目的

● 網路服務(Web Service)[13] 為一種能將分散在不同系統資源進行整合的技術，這項技術的基礎主要是建立在以可延伸性標示語言 (XML)、簡單物件存取協定(SOAP)等一些標準的協定上。網路服務的設計者可用任何的程式語言開發工具和作業系統來描述與撰寫網路服務。根據W3C對於服務導向架構(Service Oriented Architecture)的定義[14]，在服務導向架構中主要由三個角色所構成，分別是服務提供者(Service Provider)、服務仲介者(Service Broker)及服務請求者(Service Requester)，如圖一所示。



圖一. 一個服務導向架構圖

服務提供者 (Service Provider) 為服務的擁有者，也是服務的執行平台，並且提供服務功能讓服務要求者來使用。服務請求者 (Service Requester) 需

要某些功能來幫助它完成工作，於是發出請求希望服務提供者所提供的服務能夠滿足它。服務仲介者（Service Broker）扮演媒介的腳色，提供了一個平台讓服務提供者可以發布(Publish)所提供服務的相關描述。而服務仲介者也可以接受來自服務要求者的請求，請求者經由仲介者來幫助它發現所需要的服務並且獲得相關鏈結(Binding)訊息。

在Sycara等人的研究中，甚至將多重智慧型代理人系統[15，16]中的仲介者的觀念引進到SOA的架構中，讓仲介者可以自行與服務提供者合作，取得最後的結果，並將最後的結果直接給請求者。大大的提昇了仲介者的能力。由圖一可看出網路服務的運作流程，三個角色之間彼此的互動關係、運作過程如下：

一、 服務提供者除了開發服務外給需求者用外，還要讓請求者能夠發現它。所以為了讓所提供服務被知道，服務提供者必須將與服務相關的資訊，透過網路向服務仲介者發佈。

二、 當服務請求者需要找到服務來幫助它時，向服務仲介者發出相關需要的查尋訊息，仲介者會將符合的服務提供者資訊回傳給請求者。

三、 最後，請求者經由仲介者回傳的訊息，找到所要的提供者的位置資訊。接著與提供者作互動，並發出服務呼叫的訊息使用所提供的服務。

然而在圖一中我們也可以發現，最後的結果是提供給使用者的。而使用者一開始或許對所要求的服務並沒有特別的概念，所以初次提出的服務需求描述，有可能與服務提供者所提供的服務描述不太接近。另外也很有可能使用者要操作系統許多次，才能累積足夠的經驗，順利的找到所要的服務。並利用服務解決問題。

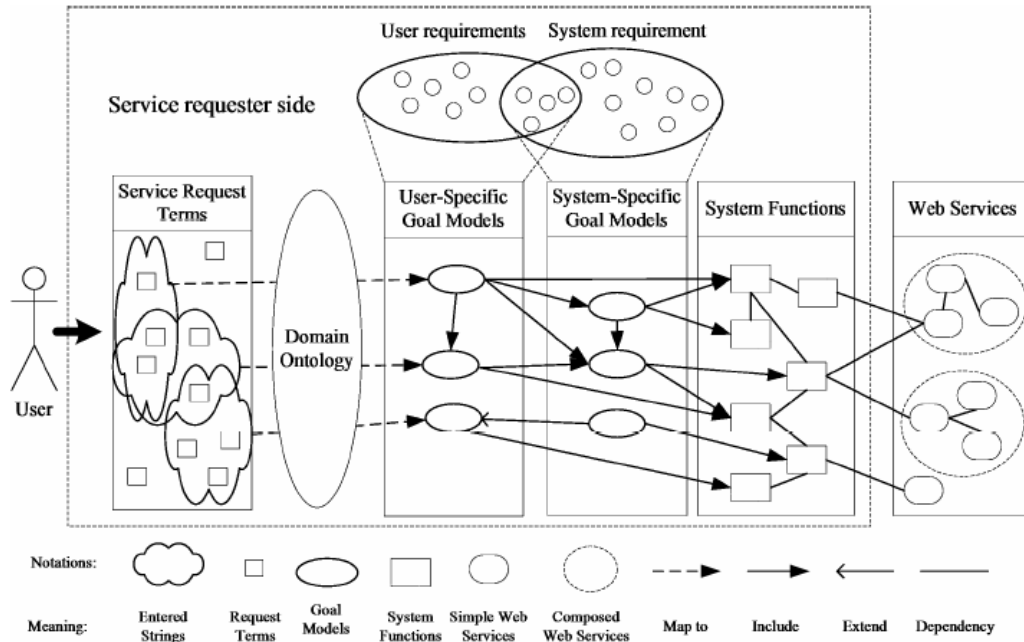
如果系統可以有一些背景知識並且能和使用者的，服務仲介者，或服務提供者互動，理解使用者的真正需求，協助使用者找到相對應的服務，並結合系統本身設計好的功能，將所找到的服務與系統功能串起來，提供給使用者，如此便能提供一個更方便的網路服務使用環境。探討這種軟體系統如何建構的相關問題即為本研究計劃最主要的目的。

因此，本二年期研究計劃的目的旨在探討如何建構一個系統，使其能夠對使用者所輸入的服務要求字串加以解釋，並結合位於系統外部的網路服務(External Web services)、以及系統本身所提供的內部系統功能(Internal system functions)，產生一系列的動作來滿足使用者的需求。本研究同時也探討建立一個以軟體工程為基礎之IASOS開發方法。我們所提出的整個系統運作的概念如圖二所示。

一個來自於使用者對服務的要求係由服務要求的關鍵字所組成。我們假設使用者對服務的要求的關鍵字代表使用者使用系統的意圖。我們將一組目標模型，以及他們之間的關係所連結起來的資料結構稱為目標結構。而藉由事先定義好的領域本體論以及目標模型的資訊，使用者所輸入的服務要求將會被對應到目標結構中的一個使用者目標模型。目標結構扮演者將服務的要求與系統能力對應起來的中間者角色。

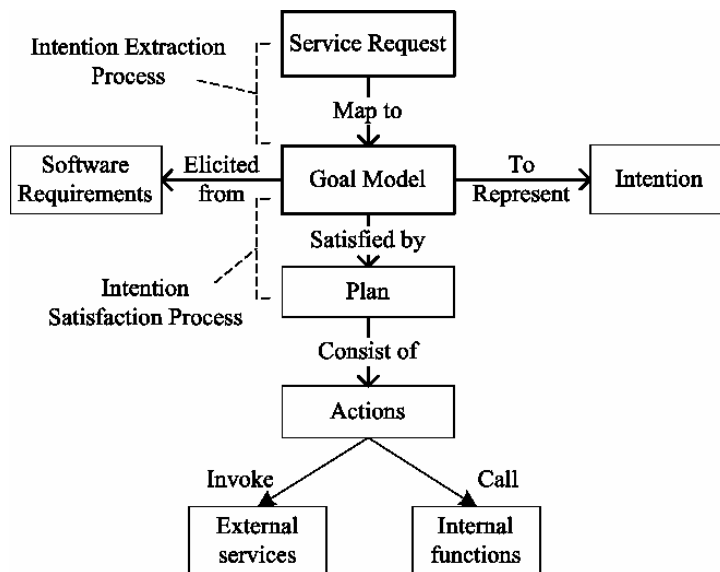
當一個原始的目標模型被選出來代表使用者的意圖時，其相關的子目標模型也會被挑出來代表使用者隱含的意圖。基於這些目標模型中的資料，系統可以產生一個由一系列動作所組成的計劃。而系統將可以經由計劃的執行來滿足使用者的需求。當被產生的計劃被執行時，系統可以與服務仲介者或提供者互動，以便

完成計劃。此外系統也可以與使用者互動以向使用者取得對於服務需求的更多資訊。



圖二. 將使用者對服務的要求轉成系統功能之過程。

整個具意圖感知能力系統的處理問題過程將會有如一個軟體需求重新使用的方式。系統的使用者輸入一個句子代表使用者的需求，然後系統根據需求分析結果所產生的目標結構，找出可以代表使用者意圖的目標模型來準備計劃的產生。如果系統本身無法滿足使用者的需求，系統可以呼叫外部的服務仲介者或提供者來擴充系統本身的能力並且合作滿足使用者的需求。圖三說明了整個研究想法的主要因素。



圖三. 使用者服務要求至系統行動對應圖。

在我們要解決的問題中，一個服務要求(Service Request,  $SR$ )被定義為由使用者所輸入的字串中的字元(Term,  $T$ )所形成的一個向量。換句話說， $SR = \{T_1, T_2, \dots, T_n\}$ ，其中  $T$  代表字串中的關鍵字元，而  $n$  代表這些關鍵字的數量。我們假設使用者對服務要求的意圖隱含於  $SR$  之中。我們定義目標模型  $GM$  為系統所能執行的功能的一個抽象描述。意圖擷取過程  $IEP$  的問題便可以被描述  $IEP: SR \rightarrow GM$ 。上式代表當系統被給予一個  $SR$ ，經過  $IEP$  的程序後， $SR$  可以被對映到一個定義好的系統功能的抽象描述  $GM$ 。

另外我們定義一個計畫(Plan,  $P$ ) 是由一系列的系統底層動作(Action,  $Act$ ) 所組成。而一個  $Act$  則是用來描述單一的或是複合的系統動作。一個複合的系統動作由數個單一的系統動作所組成。在 IASOS 中這些系統動作可以被用來呼叫系統內部功能或外部的服務(internal functions or external services)。一個內部功能可以被用來服務使用者或者是與外部服務結合，以產生更複雜的複合性服務一個外部服務則是由外部軟體物件所提供的功能，例如由網路服務提供者所提供的服務或是其他軟體代理人所提供的服務。在本研究中，我們利用代理人的架構來使得計劃的產生與執行能夠滿足個人的需求與讓服務的提供更多的彈性化。

我們利用自然語言處理技術(Natural Language Processing, NLP)，配合系統本身預先設計好的領域知識(Domain Knowledge)，以找出使用者輸入的字串的代表意義以完成「使用者意圖擷取」的過程[3,6,7,8]；而「使用者意圖滿足」則利用人工智慧規劃技術(Artificial Intelligence Planning, AI Planning) [18]，配合系統內部功能與外部服務來產生一個計畫的程序。

在本研究計劃中，我們由軟體需求工程[17, 18]的角度出發來探討這樣的系統建構的方式。我們也建立一個有意圖感知能力的系統的軟體發展方法，和一個系統架構，來讓系統設計者能夠依循我們所提供的方法，和原始的系統骨幹建構出不同領域的應用[1]。

### 3. 文獻探討

#### 3.1. 本體論(Ontology)

Ontology一開始起源於哲學，後來Ontology這個詞漸漸的用於其它的領域，所以對於每個領域對於Ontology的定義都有一點不同。在[19]中對Ontology的定義為對某個領域內所用到的概念給予明確且正式的敘述，並且描述了概念與概念之間的關係。[20]的作者認為Ontology中定義了分享資訊所需要的共同字彙，包含給予字彙清楚的定義以及關係的描述，進而讓機器可解讀這些概念與關係。經由概念被清楚的定義與描述，當雙方在談論事情時，都能知道所要表達的東西為何，也可讓機器了解人所想要表達的想法，甚至是讓機器與機器相互了解。在[19] **錯誤! 找不到參照來源**。中列出了幾項為何要發展Ontology的理由為：分享人於人或智慧型代理人(Agent)之間對於資訊架構的共同協議；領域知識的再利用；將領域知識與運作知識分開。簡單來說，我們可以用 Ontology來表達對於某一領域的知識。

而在網路上，為了使用一個網路服務，軟體實體需要一個電腦可以理解的服務描述，而這此服務的描述都可以在OWL[21]的Ontology架構下完成。而

OWL-S[22]語言則是在網路服務的語意描述方面的一個網路服務的 Ontology 基礎。

本研究也探討了利用 Personal Ontology 來代達使用者個人對於使用網路服務系統時的使用偏好。在[23]中作者提到每個人生活中所擁有的個人資訊非常的多，過多的個人資訊有時反而是一件不好的事，因為當我們需要找到需要的次資訊時，卻不太容易馬上找的到它。經由ontology的建立，除了讓我們更容易組織資訊外，更加強了資訊與資訊間的語義化(Semantic)的關係。[24, 25]中利用 Personal Ontology 來表示使用者在瀏覽網頁時的習慣與興趣，然後系統藉由 Personal Ontology 來推薦使用者有興趣瀏覽的網頁。[26]中用 Personal Ontology 協助會議排程代理人進行會議排程決策工作，來達到個人化會議排程之需求。[27]中用 Personal Ontology 來表示使用者個人的資訊與在食物方面的偏好，經由對應 Personal Ontology 與 Domain Ontology 的來幫使用者找到網路服務。在[26]中同時也利用了物件導向本體論 (Object-Oriented Ontology) 方法，來幫助建立了不同領域的 Personal Ontology。

### 3.2. 人工智慧規劃(AI-Planning)

為了達成某個預先設定目的，我們須要執行一連串的動作後，才能達成該目的，而將這一連串動作組合起來的推論過程，我們稱之為規劃。而在人工智慧的領域中，研究如何使計算機技術來進行自動化規劃的研究，我們稱之為人工智慧規劃(AI Planning)。在[28]中，我們利用階層式工作網路規劃(Hierarchical Task Network Planning, HTN) [29] 演算法來組合不同的服務，藉以產生新的複合式服務。而在本研究中我們延續此一方法來完成使用者意圖滿足的問題。

一個關於規劃的問題描述我們通常分成三個部份來討論，即狀態(States)、動作(Action)以及目標(Goal)。在一個規劃中，起始狀態和目標通常是作為規劃系統的輸入，在規劃的過程中維護狀態的表達是很平常的事。因為大部份的動作(Action)會改變目前的狀態。一般的規劃中，一個運算子通常會有下列三個部份：動作的描述(Action Description)、前條件(Precondition)和運算子的作用。一個完整的規劃是每一個步驟的前條件都在之前的步驟就成立，如果某條件是一個步驟的作用，且沒有其它的步驟會使它不成立，則稱此步驟為完成某步驟的條件。在進行一個規劃時，我們可以從起始狀態開始，一次使用一個動作，直到我們達成目標內敘述的所有狀態。

近年來人工智慧領域中的自動化規劃技術(automatic Planning)[29]普遍地應用於自動化網路服務組合(Service Composition)[30]中，如馬里蘭州大學所展的 MIND[31]、以及M.Pistor[32]等系統。但是這些系統本身專注於搜集位於系統外部的網路服務資訊，並將服務提供者(Service Provider)所提供的服務利用人工智慧規劃(AI Planning)[1]的技術將這些可取得的網路服務組合起來，藉以滿足使者的需求，然而這些系統的設計上並沒有注意到，其實有些使用者的需求有時可以結合系統本身所提供的服務以及系統外部的網路服務來提供更具彈性的服務模式。另一個問題是大部份網路服務組合系統並沒有記錄以前所使用過的服務提供者相關資訊及以前完成執行過的規劃來加速系統提供服務的速度和便利性。



### 3.3. 網路服務選擇系統

由於在網路服務的環境中，隨時都可能新的網路服務被創造出來，所以我們需要存在一個公開的平台，來讓服務提供者能夠發布新的服務讓大家知道，而需要服務的請求者也可以透過這個平台來查尋找到所想要的服務。在網路服務的重要角色與整個流程中有個很重要的步驟，就是的網路服務的選擇。當服務請求者向服務仲介者查尋服務後，通常會有多於一筆符合查詢條件的服務資料回來，此時我們要如何選擇最好或最適合的網路服務來滿足請求者的需要，這便時網路服務選擇想要完成的任務。

在[33]中將服務的種類分為二類，分別是企業對企業（business-to-business）與企業對使用者（business-to-consumer），在執行網路服務選擇時兩者的重點又不太一樣。網路服務選擇並不是一件容易的工作，在[34]中就提到說因為網路服務本身就是一個開放式的環境，所以我們很難去預測網路服務品質(QoS)。若如果還需要進一步考慮使用者話，會使網路服務選擇成為一個很大的挑戰。在[35]就提出我們需要以使用者的觀點來看選擇網路服務，了解使用者真正所要的需要，因此可能要考慮到比語意(Semantics)更深的語用學（Pragmatics）。在[36]中也提出達到個人化網路服務選擇不是那麼容易的，因為以使用者為中心的通常會使用選擇的工作變得複雜，並且最後決定服務是否要被執行時，通常得讓使用者自己去決定，因為使用者可能會有自己特別的偏好。雖然個人化網路服務選擇是一個困難的工作，但在一些網路服務相關文獻中，皆認為個人化的網路服務環境是我們必須努力達成的目標。

經由比較服務選擇的相關研究與系統，讓我們知道了網路服務選擇的因難在哪，並且也可以看到目前的系統大多並沒有真正考慮到使用者個人對於服務領域的偏好問題。在本研究中我們同時探討透過個人知識本體來讓機器自動化的處理與使用者相關的問題。

### 3.4. 服務組合(Service composition)

網路服務是以網路的開放標準為基礎，使用者可以利用各個網路服務所提供的服務，來達成使用者自己本身的需求。由於單一服務已經無法完全滿足使用者的需求，所以組合多個網路服務以滿足使用者的需求是最近幾年來網路服務的發展趨勢。利用人工智慧的規劃方法來組合網路是一個普遍被採用的方法。例如 Evren Sirin [31]等人開發的服務組合系統是以 HTN為主要的規劃方法來組合相關的網路服務。由於 HTN和 OWL-S都屬於階層式的架構，該系統利用此特性來設計一個OWL-S 到 HTN程式的轉換器。利用此轉換器將可使用的網路服務提供者所提供的 OWL-S檔轉換成 HTN相關的檔案。當使用者利用一個使用者介面輸入使用者所需要的服務後，系統會將這些資訊傳入 HTN規劃器。規劃器會利用由 OWL-S轉換來的資訊進行規劃，最後再將規劃的結果利用規劃器到 OWL-S的轉換器，將規劃完的結果轉換成一個OWL-S，用以描述如何進行網路服務組合。

表1是一些網路服組合相關的文獻，經過整理過後所產出的系統能力比較表。在表1中我們是以它們是否具有突發狀況處理能力、目標達成能力、網路服務記憶能力以及內外部服務組合能力這四種能力作為各系統之間的能力比較依據。由表1中可以看到，全部份的系統都具有目標達成能力，也就是這些網路服

務組合系統，都能利用系統已知的網路服務提供者所提供的功能，將這些功能組合成能滿足使用者需求的網路服務組合。但只有少部份的系統具有突發狀況處理能力，而且大部份的系統並沒有記憶網路服務的功能。所以當使用者有類似的網路服務需求時，大部份的系統都必需要重新取得網路服務提供者的資訊，才能組合出符合使用者服務需求的功能。此外大部份的系統也不具有內外部服務組合能力，所以他們無法提供更具彈性的服務組合模式。

使用系統本身所提供的內部服務主要有幾個優點，第一個是可以節省搜尋網路服務的時間以及資料傳輸時間，第二點是可以有效掌控各種突發狀況。由於尋找適合的網路服務需要透過網路服務仲介者才有辦法取得，若所需要的功能可由系統本身所提供的服務來滿足的話，則可以有效的減少搜尋網路服務時間及資料傳輸時間。此外由於內部服務為系統內部所提供的功能，所以各種可能會發生的突發狀況都可以由系統內部所掌控，進而可提出有效的應對措施。

表 1. 網路服務組合系統功能比較表

網路組合系統	網路服務組合方式	突發狀況處理能力	目標達成能力	網路服務記憶能力	內外部服務組合能力
E. Sirin [31]	HTN Planning	No	Yes	No	No
M. Pistore [32]	Planning As Model Check	Yes	Yes	No	No
U. Kuter [37]	HTN Planning	Yes	Yes	No	No
B. Limthanmaphon [38]	CBR	No	Yes	No	No

本研究希望能藉由階層式工作網路規劃方法加上Case Based Reasoning (CBR) 機制[39]，讓本系統在使用者首次有某些服務需求時，能使用本系統所提供的階層式工作網路規劃方法，結合系統內部及系統外部的服務來提供使用者一個更具彈性的服務組合。當使用者下次有類似需求時，使用者便選擇使用CBR機制以加速產生一個可以滿足使用者需求的服務組合。

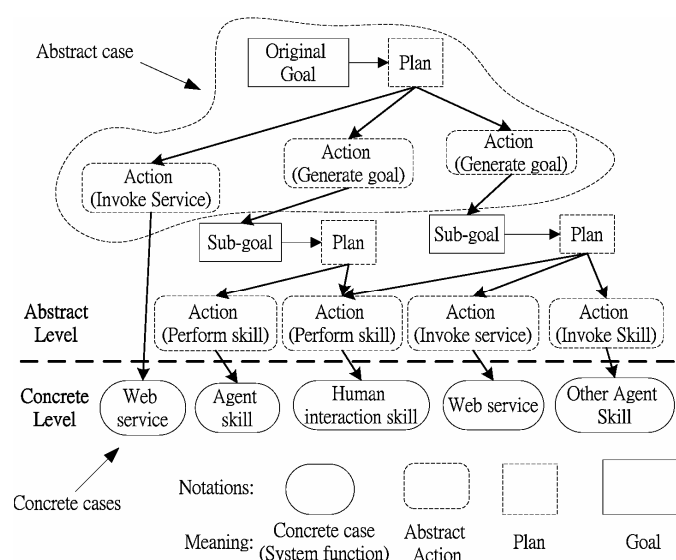
### 3.5. 案例式推理(Case-based Reasoning)

案例式推論(Case Base Reasoning)是一個解決問題方式，案例式推論理論是以類似人類解決問題的方式進行問題的解答。一般人類在解決目前所遇到的問題時，通常會以過去解決類似問題的方式去解決目前所遇到的問題。案例式推論(簡稱 CBR)基本上它與人工智慧領域的其它方法有很多方面的不同，第一個不同點在於它不僅依賴有關問題領域的常識之外，它還能跟據問題的狀況去運用特殊領域的知識去解決問題。另一個不同點在於 CBR 每解決一個新的問題後，可以經由學習的方式將每次解決問題的經驗保留下來，並使用它來解決未來的問題。

利用 CBR 技術來加強網路服務之提供有不少之方法被提出。B. Limthanmaphon 等人以 CBR 作為主要的工作原理來進行網路服組合的工作 [38]。而在[40]中，作者則探討了應用 CBR 於語意網路服務(Semantic Web Services)

中的相關議題。利用 CBR 進行服務組合(Services Composition)也是一個應用 CBR 於網路服務研究的議題之一[41]。在[42]所提出來的的方法則藉由階層式工作網路規劃方法(Hierarchical Task Network, HTN) [29]，加上 CBR 機制，讓系統在使用者首次有某些服務需求時，能使用系統所提供的 HTN 方法，結合系統內部及系統外部的服務來提供使用者一個更具彈性的服務組合。而當使用者下次有類似需求時，系統可以便選擇使用 CBR 機制以加速產生一個可以滿足使用者需求的服務組合。

在[8]中，我們提出一個類似[42]的方法來加強代理人為基礎的網路服務系統。在此方法中我們將案例分成抽象層與實用層不同的層次以提昇案例推理的效能。此方法的概念圖顯示於圖四中。在本研究中我們將進一步實現此一方法，讓服務的要求者可以記得過去與使用者以及其他服務提供者互動的經驗，以提供更貼近使用者需求的服務。



圖四 應用案例推理於網路服務系統中之概念圖

#### 4. 研究方法

在第一年我們著重在如何開發意圖擷取工具、與意圖滿足工具，最後再結合這二項工具產生 IASOS 的基本架構。而第二年我們則著重在 IASOS 軟體建構的方法論上。其間我們也考慮了個人化的 IASOS 的設計。因此在本章節中首先描述我們所發展的個人化本體論與目標模型，接著我們描述使用者意圖擷取機制以及使用者意圖滿足機制，最後說明以目標模型導向的 IASOS 軟體開發方法。

##### 4.1. 領域本體論與個人化本體論之建立

建構 Personal Ontology 時，我們用 concept 代表使用者對於某個領域的偏好，例如“price”這個 concept 應用購物的領域時，我們通常都用來指物品價錢或花費的多寡，可以用來當作購買物品時的一個要素。concept 可以跨越於不同的領域，對於不同的領域來說，同一個 concept 可能代表了不同的意義，就如同一字多義一樣，用於不同的狀態下，描述了不一樣的意思，例如“price”也可能代表的是某項產品，而不再是原來價格的含義了。

不過在這裡有了另外一個與concept相關的問題產生，就是我們如何知道所談論的某一個concept，它所代表的含義為何呢？這時我們需要更多與此concept相關的輔助資訊與證據，來讓我們專注在concept的特定含義上。就上面的“price”的例子而言，當我們看到的描述是“price : 500 NTD”時，很容易就聯想到此時price這個concept所代表的含義為價錢，當看到的描述是“product name : price”時，就很容易了解到price所代表含義為一項產品。

另外，我們即使是同一個領域下的同一個 concept，對於不同的使用者來說，對於 concept 的描述、量測與形容上，仍會有程度上不同的差別，例如：“expensive book”，A 與 B 都知道這樣的文字是在描述貴的書籍，但對於 A 來說，書超過 500 元就已經很貴了，但對 B 來說，書要超過 1000 元才是屬於貴的書。

根據以上的想法，每一個領域會包含了許多的 concept，而 concept 本身又需要由其它更多的 concept 來組成。本論文分別從 Domain 的角度與 User 的角度，將這些 concept 區分為六種型態。先從領域來看的話，分別為：

- 一、Domain Independent Concept (DIC)：指的是 concept 會被用於不同的領域，不過在每一個領域之下，concept 所代表的含義皆是相同的，即是只有一種且唯一的定義，concept 在每一個領域下的含義皆為相同。
- 二、Domain Dependent Concept (DDC)：concept 在不同的領域之中，可能會擁有不同的含義，類似文字上的一字多義，此時 concept 需要更多其它的資訊來輔助，concept 在不同的領域下會有不同的 properties 被用於描述此 concept，因此可以解到 concept 在目前的領域所代表的含義為何，例如上面提過的“price”例子。
- 三、Domain Specific Concept (DSC)：此 concept 只會出現在某個特定的領域，為此領域特別的限定的用法。concept 只會擁有唯一的含義，為一對一的關係，例如：“ISBN”可能只會在書的領域被用到，代表的是國際標準圖書編號。

而從使用者的角度來看，對於不同環境背景下的使用者，是否存在一個對於 concept 的公認準則，讓我們可以來用於表達個人對於 concept 的認知，例如“expensive book”，user 對於書的貴或便宜，是否存在一個相對應的標準，讓使用者不會有不同的解釋，就像是否有一條公認的規則，如一本書的價錢大於 1000 元的話，就是 expensive book，而這個標準，是每個人都會遵循的。在這樣的定義下，又可將 concept 分為：

- 一、User Independent Concept (UIC)：每位使用者對於 concept 都存在著一個共通的標準，並且每個人都循此標準，就像“大雨，豪大雨”這樣的 concept，氣象局用此 concept 用來表示雨勢大小，氣象局有明確定義大於多少的雨量才可以被稱為大雨，所以存在著一套標準，會對映到一個值。用數學的角度來看的此問題，大雨如果是一個輸入，大家都使用公用的規則來處理這種輸入，而這個規則，就是所謂的標準。對每一個使用者來說，於同樣的輸入經過使用者處理後會有一樣的輸出，所以我們稱這類的 concept 為 User Independent Concept。
- 二、User Dependent Concept (UDC)：一些屬於比較模糊的 concept，即便是 concept 在同一個領域下，還是會因人而異的 concept，每個人會因自身的條件，造成對 concept 不同程度上的見解，例如說：“expensive book”，expensive 的標準因人而不同，每個人心中都擁有自己的規則，expensive 所對映到的值因

人而異。用數學的角度來看，在已定義好的輸入種類與輸出種類下，我們用自己所擁有的規則來處理這些輸入，並得到輸出。如果價錢是輸入，貴或便宜是輸出的話，大家都用不同的規則來處理這種輸入，就有可能會得到對於書的價錢是屬於貴或便宜等不同的看法了。所以 concept 會因使用者的不同，而產生了不同的對映關係，我們將此類的 concept 稱為 User Dependent Concept。

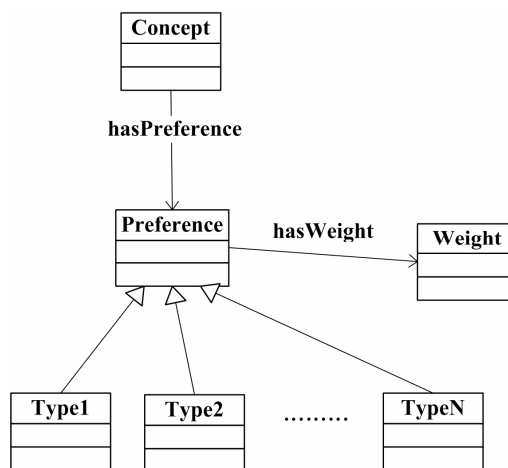
三、User Specific Concept (USC)：若從使用者的角度來看的話，使用者對此 concept 給予特別的含義，而且這種含義其它使用者並不遵循，我們就把此 concept 稱為 User Specific Concept。它與 User Dependent Concept 並不一樣，不一樣的地方在於 User Dependent Concept 所描述的是一種程度上的差別，而 User Specific Concept 是根本上含義的差別。就像對於某個變數的處理，雖然是一樣的輸入，可是輸出的種類卻非常的不一樣，例如：“price”這樣的描述，對於其它人所代表的皆是價格，不過這樣的描述對我來說，所代表的是我家小狗的名字。因為只存在於使用者自己的心中，並且只對自己有所其特殊意義的 concept，我們將之稱為 User Specific Concept。

根據以上的定義，分別從 Domain 與 User 的角度，我們可以將每一種 concept 作分類，並且每一種 concept 同時可能會擁有二種特性，所以一共會有九種可能的組合。

#### 4.1.1 Concept 與 Concept 之間的關係

上面一個小節已描述過我們所定義的分類標準與對於 concept 要如何將之分類，接下來我們會說明為何要做如此的分類，還有 concept 與 concept 之間關係的詳細說明。concept 與 concept 的關係說明如下：

一、hasPreference：指的是擁有特性為偏好的 concept，如下圖一，其 hasPreference 的定義域(domain)為任意的 concept，在本論文中通常為服務領域，值域(range)為可能的偏好，偏好同時含有權重(weight)，權重可經由使用者自訂，或者是利用 Data Mining[8]等方法來獲得，權重可以用來當成使用者對於偏好的程度差別，可以隨著使用者的使用記錄來調整權重值。

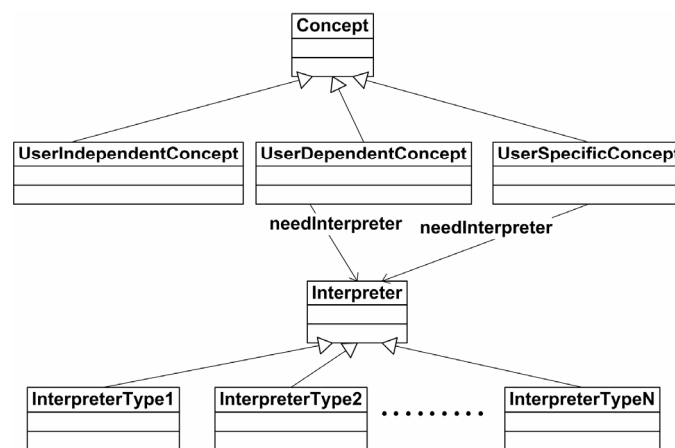


圖五. hasPreference relation

二、hasInformation：指的是連結使用者與使用者 相關資訊類 concept 用的關係，

例如：Credit Card。

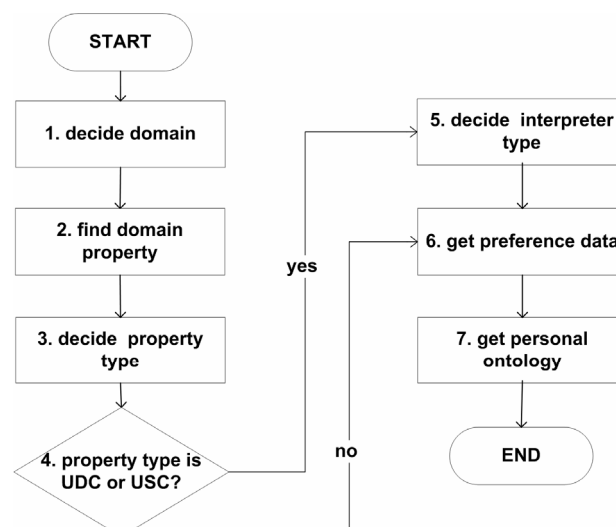
三、needInterpreter：連結 User Dependent Concept 與 Interpreter 之間的關係，如下圖二所示，needInterpreter 的定義域為 User Dependent Concept 或 User Specific Concept，值域為 Interpreter 類的 concept，Interpreter 主要是用來幫助我們了解使用者對於 concept 的含義，因為當每個人對於同一個 concept 卻有不同的了解時，如“expensive book”，可能我的大於 1000 才是貴的書，不過你大於 500 就認為是貴的書了，因此需要額外的使用者資訊或另外的處理方式，才能了解使用者想法並且解讀出對於使用者來說所代表的真正含義。例如使用者的意圖若為想要買機票，我們從 Goal Model 中取得的與飛機相關的參數為 cheap，經由 Interpreter 的解釋，就可知道 cheap 對使用者所代表的價錢範圍為多少了。



圖六、needInterpreter relation

#### 4.1.2 Personal Ontology 建構流程

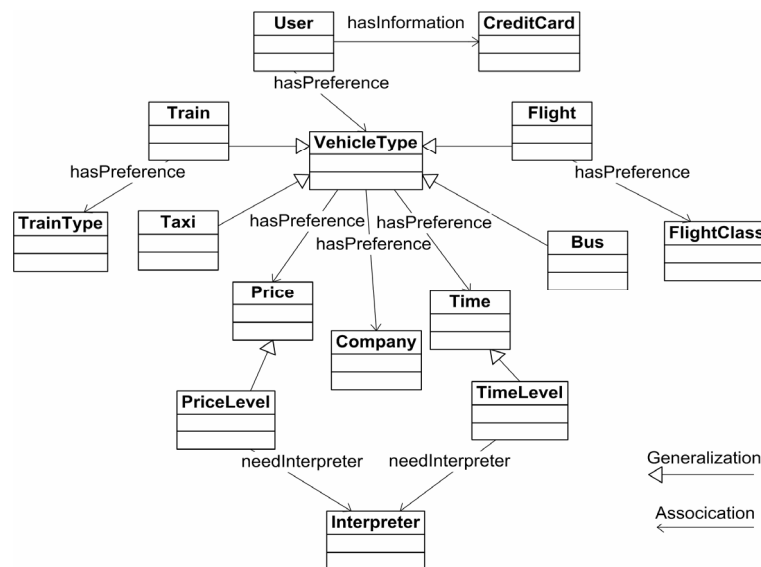
下圖為整個構 Personal Ontology 的流程，接下來針對流程中的每一項個別做詳細的說明，說明如下：



圖七、Personal Ontology 建構流程

- 一、先決定所要應用的 Personal Ontology 是屬於哪一個領域。
- 二、針對領域的不同，利用專家的領域知識或是用有系統的方法(如:Data Mining) 來找出使用者在此類領域所關心的要點有那幾項，而這些要點便成為我們用來表示使用者偏好部份，本論文中用 concept 來代表這些要點，也是整個 Personal Ontology 架構中的核心，也是最困難的部份。因為使用者關心的要點種類會隨著領域的不同而來改變，雖然在本論文中自動化找出使用者關心的要點並不是主要的重點，不過也把這項工作視為未來需要加強的地方。
- 三、再來必須針對已找出的 concept，依照我們所提出的分類方法，依所代表的含義來做分類。
- 四、若分類為 User Dependent Concept 或 User Specific Concept 的話，表示同一個 concept 可能會有不同使用者代表不同意義的狀況發生，所以我們需要更多的資訊來幫助我們。例如用貴、便宜來表示使用者買機票偏好的話，因為使用者對於貴、便宜會有不同程度上的感覺，所以此類 concept 為 UDC。如果不是這二種分類的話，直接跳到第六點。
- 五、User Dependent Concept 與 User Specific Concept 需要進一步的靠 Interpreter 來幫助我們解讀出使用者的想法，所以需要參考 concept 的內容，並且需要依靠領域專家的知識，來決定怎樣形式的 Interpreter 較適合，例如：就上面的價格例子，我們可以簡單的以 rule 的方式來表示，if preference = expensive, then price > 3000。
- 六、獲得使用者個人資料與對每一類偏好的權重，而資料與權重取得的方法有很多種，可以根據使用者過去的記錄如 Data Mining 等方式，或者以網頁問卷的方式讓使用者點選。
- 七、最後根據獲得的資料，系統對於不同的使用者產生屬於個人的 Personal Ontology，然後系統就可以根據 Personal Ontology 來幫使用者自動的找到適合的服務。

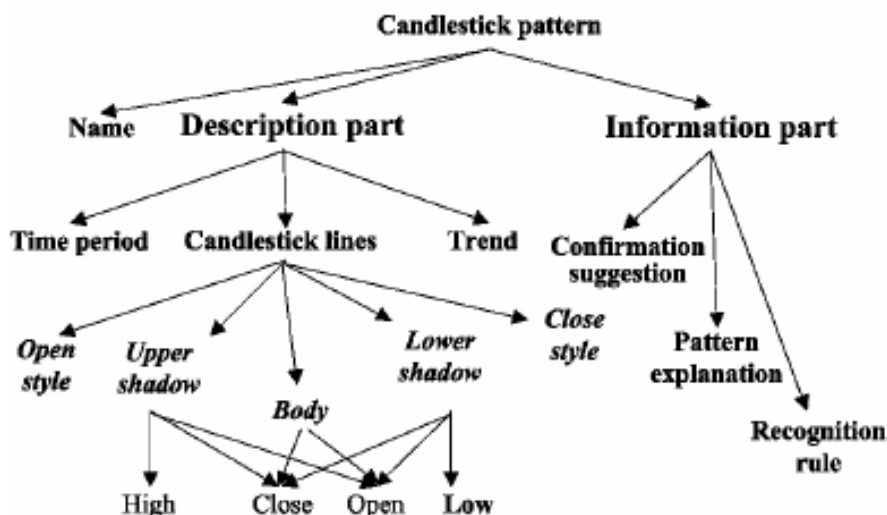
圖八是我們以交通工具為領域，所建立的簡略 Personal Ontology。



圖八、簡略交通工具領域 Personal Ontology

### 4.1.3 Personal Ontology 的其他應用

在[4, 12]中，我們利用模糊理論以及知識本體，定義出一個 K 線理論[43]的知識，而寫好的電腦程式便能利用這個定義好的知識本體來記錄對投資人有意義的 K 線樣式。所定義之知識本體簡圖如圖九所示。



圖九. K 線樣式之知識本體簡圖

一個 K 線樣式有三個屬性：Name 為特定 K 線樣式的名稱，Description Part 描述總成 K 線樣式的 K 線，Information Part 則記錄此一 K 線樣式的人為建議。一個 K 線樣式的例子如表 2 所示。

表 2. K 線樣式實例

Pattern name: 多頭吞噬	Pattern information part
Pattern description part	<b>Confirmation suggest:</b> 須確認本樣式是有效。
<i>Followed trend:</i> Down trend	<b>Confirmation information:</b> 隔日之開盤價不能低於前一日的收盤價...
<b>K 線</b> <i>Candle line 0</i> <i>Open style:</i> VERY OPEN_LOW <i>Close style:</i> VERY CLOSE_HIGH <i>Upper shadow:</i> null <i>Body:</i> ABOVE MMIDDLE <i>Body color:</i> WHITE <i>Lower shadow:</i> null	<b>Recognition rule:</b> 1. 走勢在一個下降趨勢中。當日開低。 2. 第二天的實心線整個蓋過前一天的實心線...
<i>Candle line 1</i> <i>Open style:</i> BELOW OPEN_HIGH <i>Close style:</i> BELOW CLOSE_HIGH <i>Upper shadow:</i> null <i>Body:</i> ABOVE SHORT <i>Body color:</i> BLACK <i>Lower shadow:</i> null	<b>Pattern explanation:</b> 在下降趨勢中，為一黑一紅組合稱之多頭吞噬。多頭吞噬與空頭吞噬意義大於貫穿與烏雲罩頂，但首先需決定大盤的趨勢停損可放在多頭吞噬排列最低價...
<i>Interested time period:</i> DAY	

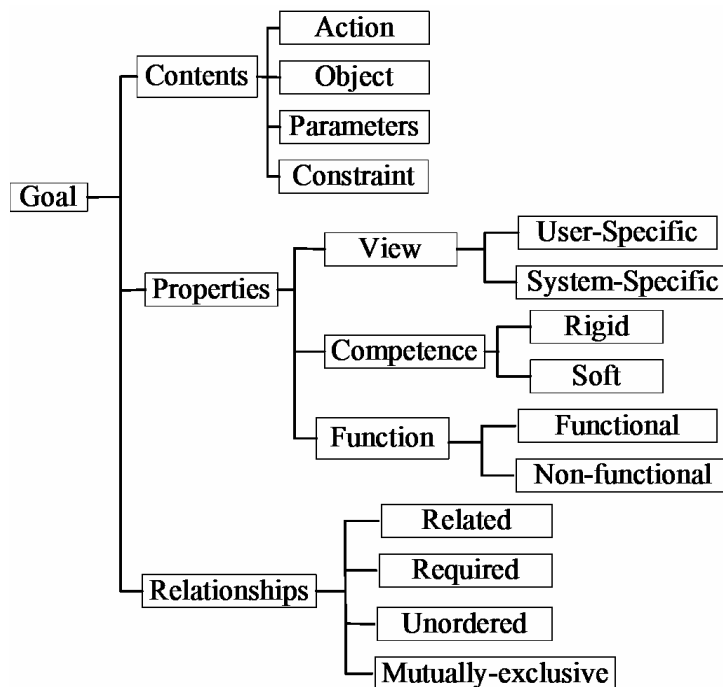


在 K 線樣式的描述部份可以使用決策樹等演算法來由時間序列的資料中找出相關的 K 線樣式，再驗證被找出來的 K 線樣式的有效性，最後由投資者自行為這些有投資意義的 K 線樣式加上描述部份。由系統所辨識出來的 K 線樣式資料不但是能夠解釋其意義且對於使用者是非常有意義之參考資訊。藉助電腦的高效能運算與快速資料處理的優點，來縮短投資者或是交易員在尋求投資標的物的過程，進而成為交易員或是投資者的一個決策輔助工具。交易員及投資決策者藉由辨識出來的 K 線樣式，可以很清楚的理解目前市場的趨勢與並利用作為風險的評估。如此逐步的累積投資的知識和經驗，相信對投資者在進行投資決策時會有相當大的助益，並且可以與他人分享投資的經驗。

#### 4.2 目標模型與目標模型之建立

如何建構使用者意圖擷取的背景知識在我們的方法以及一些本體論方法中扮演一個重要的角色。在本研究中對所使用的目標模型有三個特殊的要求。首先目標模型必須足夠的資訊來讓使用者的輸入能對應到目標模型。其次目標模型要提供一個分類的基礎給其設計者。最後目標模型要包含對應到其他相關目標模型的資訊來讓系統找出一個使用者要求下可能對應的所有目標模型，以進行規劃使用者意圖滿足的步驟。

在之前的工作中[44]，我們提出一個目標結構用於表示使用者的意圖。目標將引導系統產生一個規劃然後這個規劃的執行將會被預期來滿足使用者的需求。目標結構顯示在圖十中。一個目標由三個部分所組成：內容(contents)用來表示詢問所包含的詞彙；特性(properties)用來描述目標包含哪些屬性；規劃(plan)用來連結到哪一個規劃需要被執行以滿足這個目標。使用這樣的方式，使用者意圖可以這樣的目標結構所描述，而系統則可以執行相對應的規劃來滿足使用者的意圖。

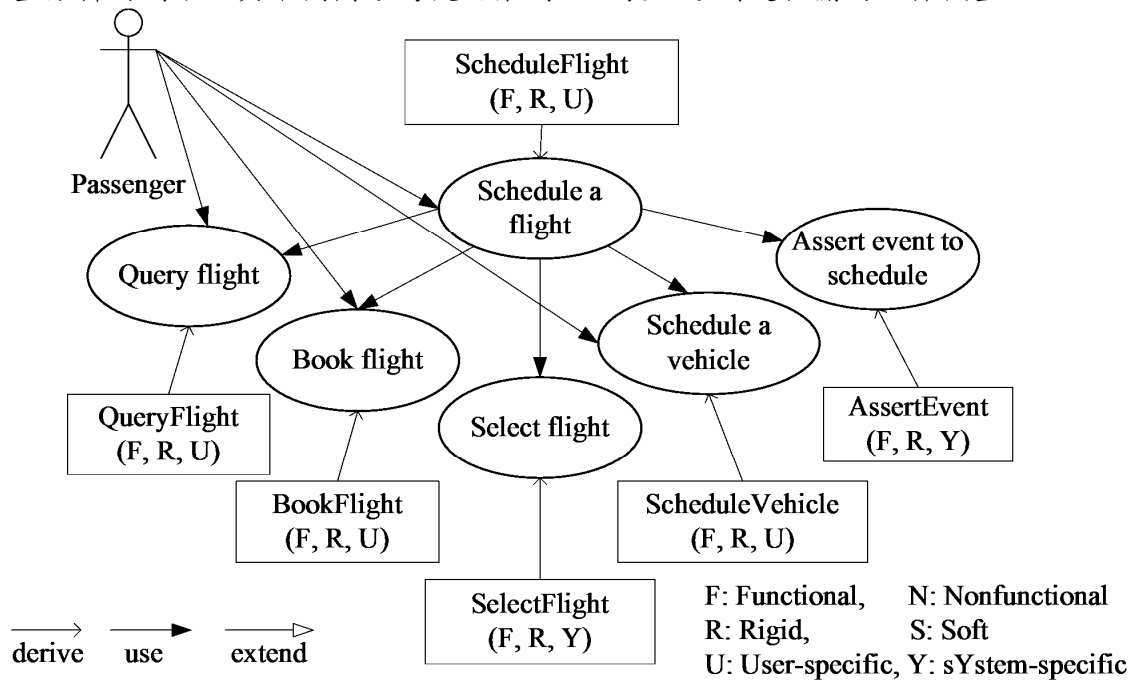


圖十. 目標模型

內容包括行動(action)、限制(constrain)和參數(parameters)。在詢問字串中，動詞被解釋成必須被執行以滿足目標的行動，形容詞和副詞則成為執行行動時的限制，而名詞則被當作行動的參數。比如說，詢問的字串為“Schedule a meeting on June 20 in the afternoon”。動詞“schedule”代表使用者要系統去執行的動作；名詞“meeting”則描述一些使用者感興趣的事物；“June 20”表示相關的參數；而“in the afternoon”則是對於時間的限制。

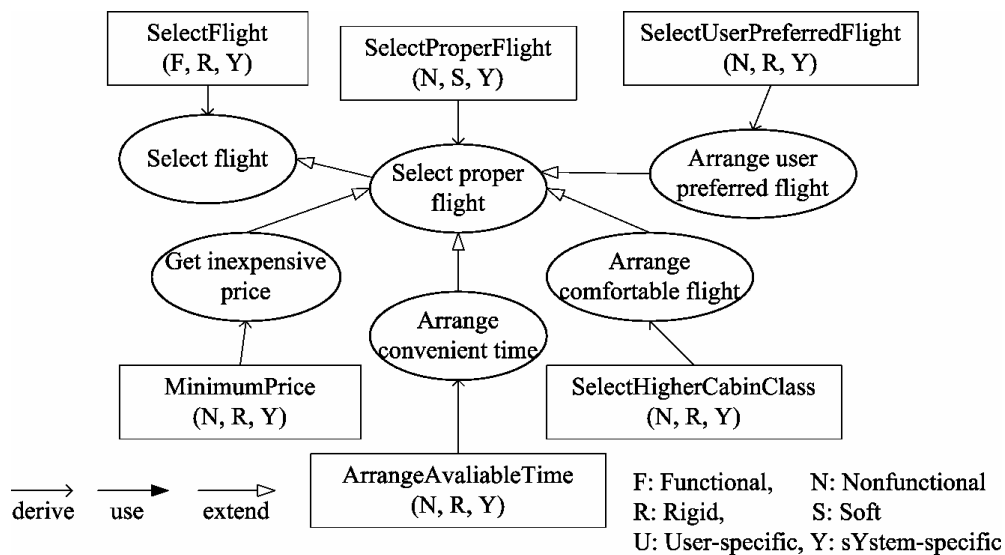
權限(competence)與觀點(view)這兩種特性提供目標分類的基礎。硬性目標(rigid goal)表示必須要永遠都要被滿足的需求；而軟性目標(soft goal)則表示期望被滿足而可以滿足到一定程度即可的需求。參與者使用系統的目的稱之為參與者具體目標(user-specific goal)，而系統具體目標(system-specific goal)則表示有關於系統提供的服務的需求。由於輸入字串反映了使用者的意圖，因此由輸入字串所衍生出來的目標皆為參與者具體目標。

關係是一個記錄相關於此目標的其他子目標。藉由使用目標的資訊以及系統執行的狀態，系統可以規劃出一些有用的策略來達成使用者的目標與查詢網路服務。這些策略構成一系列主動的行動來試圖滿足目標模型。如果一些成功或是失敗的結果回傳時，這些資訊將被傳給系統。假如使用者主要的目標失敗時，由于規劃所得到的部分結果將顯示給使用者參考。下圖是一個安排飛機行程的目標模型分析的例子。其中橢圓形為使用案例，而長方形則是相關的目標模型。



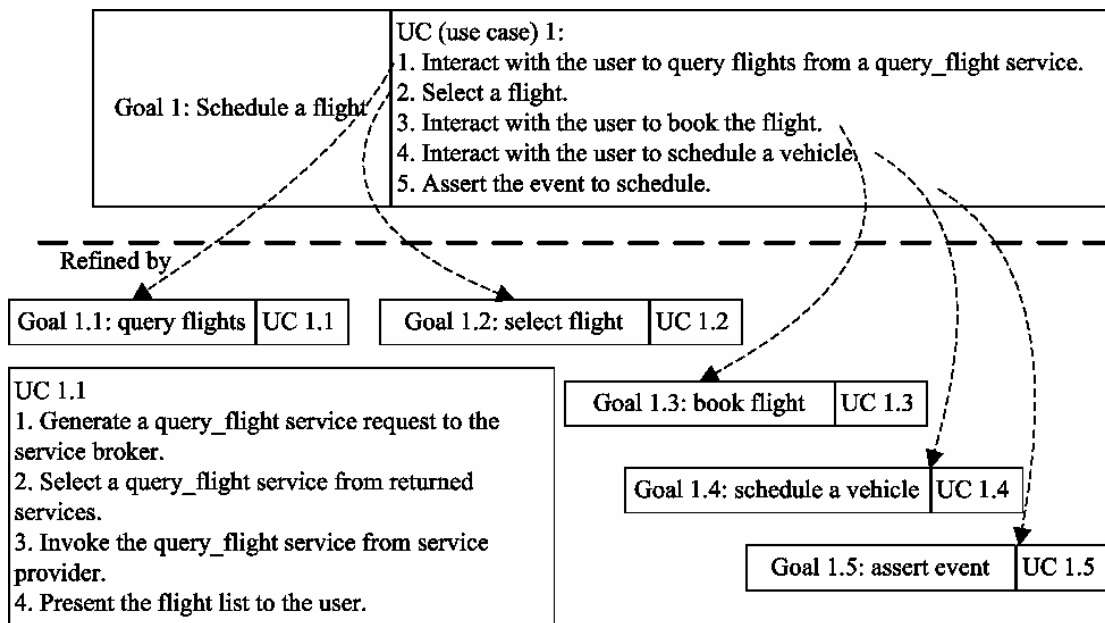
圖十一. 安排飛機行程的目標模型分析

由基本的使用案例可以找出基本的目標模型然後再進一步找出擴充的子目標模型。我們將“Select flight”的擴充子目標分析結果於圖十二中表示。



圖十二. "Select flight"的擴充子目標分析

依據目標導向使用者案例之方法(Goal-driven Use Case)，本研究將目標模型建立過程分為三個步驟：找出使用者與使用案例；利用使用案例找出擴充之目標模型；分析目標模型之間的關係。下圖說明如何從"Schedule a flight"使用案例找到其他相對應子目標模型的過程。

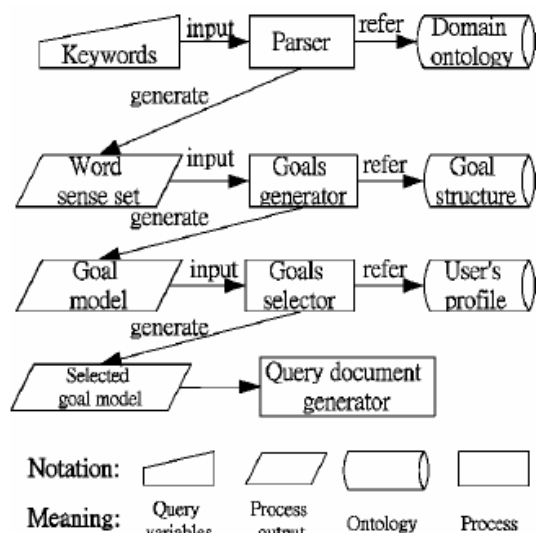


圖十三. 由"Schedule a flight"使用案例找到其他相對應子目標模型的過程

### 4.3 使用者意圖擷取機制

目標基礎的使用者意圖擷取方法的概念是簡單的。首先，系統確認輸入字串中的動詞來當作需要被執行的行動以滿足使用者的意圖。因為應用的領域為網路服務架構，所以假如輸入字串中沒有動詞，則目標的動詞則自動被設定為"query"。第二步，系統透過關鍵字比對的方式在目標模型本體論中找出所有相關的目標模型。最後，系統選出一個最適當的目標模型來表示使用者的意圖。

圖十四表示使用者意圖擷取過程的步驟。解析器、目標產生器、以及目標選擇器被設計來滿足擷取的過程。領域本體論、目標模型、以及使用者模型用來支援擷取過程。最後，被選擇的使用者模型會被使用以產生查詢文件。網路服務代理人利用查詢文件來詢問網路服務。不同於使用原始的輸入字串當做網路服務查詢的關鍵字，使用者意圖擷取工具產生的查詢文件不單只有原始的輸入字串，同時也包含的系統本身的資訊。



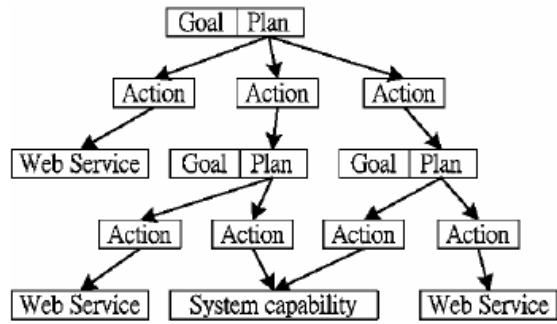
圖十四. 使用者意圖擷取過程

#### 4.4 使用者意圖滿足機制

由於不明確的服務需求字串已經被解析而且一個相關連的目標模型被產生以描述需求，下一個問題便是如何滿足目標。我們使用人工智慧規劃技術(Artificial Intelligence Planning, AI Planning) 來滿足所找到的目標的機制。

規劃是一個記錄如何達成目標的配方。藉由使用目標的資訊以及系統執行的狀態，系統可以規劃出一些有用的策略來達成使用者的目標與查詢網路服務。這些策略構成一系列主動的行動來試圖滿足目標模型。如果一些成功或是失敗的結果回傳時，這些資訊將被傳給系統。假如使用者主要的目標失敗時，由于規劃所得到的部分結果將顯示給使用者參考。

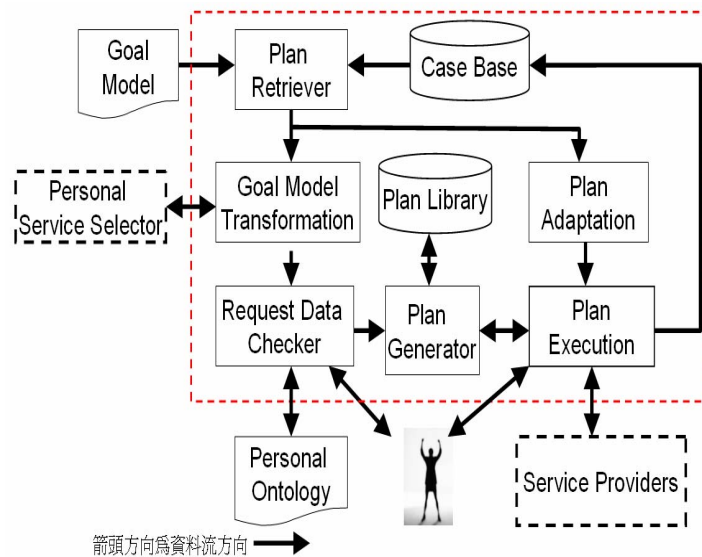
正如同規劃可以被目標所呼叫，一個目標集合也可以在規劃執行時被產生。目標與規劃的階層架構以動態的方式被建構，而這個階層架構也依系統狀況而定。使用者輸入的字串產生一個初始目標且被連結到一個初始規劃，然後子目標與子規劃被系統反覆地產生。滿足使用者目標的過程是一種決策的過程。哪些子目標需要被產生以及哪些子規劃需要被呼叫都決定於系統的狀態以及環境的資訊。圖十六表示一個目標與規劃配對階層架構的快照。一個規劃由幾個行動所組成。一個行動可以是呼叫一個網路服務、產生另一個目標、或是呼叫一個系統模組，來完成行動的需求。在目標與規劃配對階層架構中要避免形成一個無窮迴圈，且一個規畫的執行步驟與時間也要侷限在一個範圍內。在目標結構本體論中，目標與規劃配對需要被完整定義。



圖十六. 目標與規劃配對階層架構

#### 4.4.1 使用者意圖滿足機制之實現

由以上的想法我們開發了一個使用者意圖滿足之相關軟體模組。模組架構圖如圖十七所示，圖中紅色虛線所圍住的部份是本系統所擁有的功能，由於本模組的目地在證明本系統能產生出一個有效的規劃以滿足使用者的服務需求，故Plan Execution這個內部功能只是一個簡單的規劃執行器，目地是用來驗證系統產出計劃的可行性。



圖十七. 使用者意圖滿足相關軟體模組架構圖

在圖十七的系統架構圖中，本系統內部有六個主要的功能方塊，以下分別介紹這六個部份的功能：

- Plan Retriever

主要的功能是由案例庫中找出以前是否有類似的案例，若沒有找到相似的案例，系統會將目標模型傳給Request Data Checker功能方塊。若有相似的案例存在，則系統會把案例顯示出來供使用者選擇，若使用者選擇使用舊有的案例，Plan Retriever會將目標模型及使用者選擇的案例傳至Plan Adaptation功能方塊。

- Goal Model Transform

主要的功能由目標模型中的Action及Object來判斷使用者需要的服務類型，若需要使用外部服務，則將目標模型中的Object以及針對此Object預設的輸入、輸出相關參數傳給系統外部的Personal Service Selector，以取得可以滿足使用者需求的外部網路服務。

- Request Data Checker

主要的功能是檢查執行服務所需要資訊是否齊全，若是資訊有短缺，則先向系統外部的個人化本體論(Personal Ontology)取得所需的資料，不足的部份再要求使用者輸入，Request Data Checker除了檢查服務所需要的資料外，為了使產生出來的規劃能更符合使用者個人化需求，Request Data Checker還會向個人化本體論取得使用者對服務的偏好，使產出的規劃能更符合使用者個人需求及偏好。

- Plan Generator

主要的功能主要是將前面所取得的服務以使用階層式工作網路規劃方法將這些服務串接起來，以滿足使用者的服務需求。在這個部份我們是以JSHOP2[31]為本系統的規劃引擎，利用事先定義好的Problem Domain，JSHOP2 規劃引擎會根據計劃初始狀態來組合出一連串的动作。

- Plan Adaptation

主要的功用是比對案例中的目標模型與新進目標模型之間的差異，若兩個目標模型內容相同，則直接將案例中所儲存的相關資訊及計劃傳給Plan Execution，若是兩者之間有差異，則使用預定的Rule去修改案例中所儲存的相關資訊，之後再將修改過後的資訊及計劃傳給Plan Execution。

- Plan Execution

由於本研究主要的目地在計劃的產出，故本架構中的Plan Execution主要的目地是要用來驗證計劃的可執行性，Plan Execution的詳細執行過程及演算法則不在本研究討論的範圍內，當規劃被執行完後，會在最後加入一個評估機制，以評估本次計劃執行結果，以決定此次計劃及相關資訊是否要存入案例庫中。

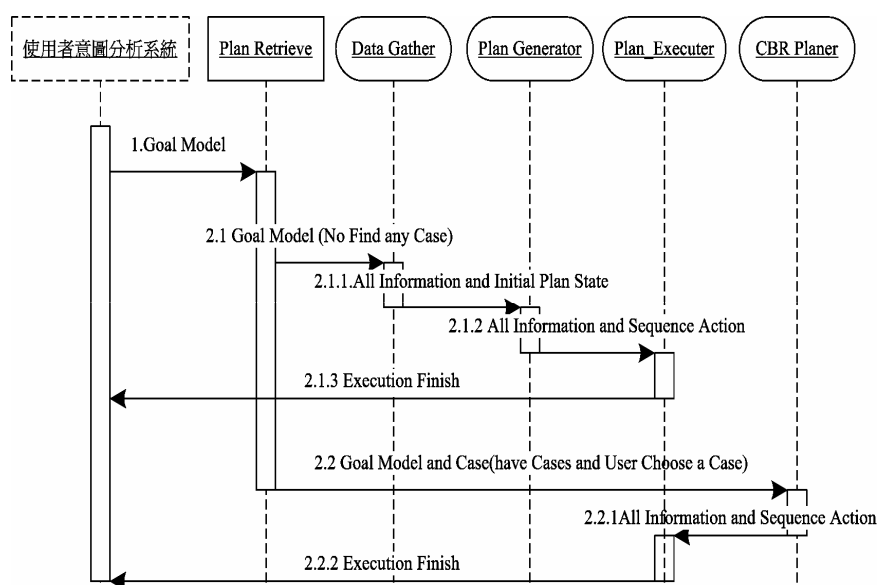
#### 4.4.2 系統執行流程說明

本系統的執行程序如圖4-2所示，圖中虛線方塊代表外部系統所提供的功能，由於系統內部功能方塊繁多，故我們把結構圖中的功能方塊依其功能性分別封裝成四個物件，並以圓角方塊表示這些被封裝物件。系統收到使用者意圖分析的機制所傳入的目標模型之後，先執行Plan Retrieve，由Plan Retrieve到案例庫中尋找以前是否存在類似案例，若找不到類似案例，則將目標模型傳至Data Gather中，如圖中步驟2.1所示，而3.2中所討論的狀況1，即是執行此步驟。若找到類似案例，而且使用者選則要使用舊案例，則系統會將目標模型以及使用者所選擇的案例一起傳至CBR Planner中，如圖中步驟2.2所示，而3.2中所討論的狀況2，即是執行此步驟。

當目標模型傳入Data Gather後，Data Gather主要的功能包涵了結構圖中的Goal Model Transform和Request Data Checker兩個部份的功能，其內部循序圖如圖4-3所示，在Goal Model Transform功能中，主要是分析目標模型內容來判斷使用者需要那些類型的服務，若這些服務有一部份是系統內部無法提供的服務，

系統會利用 Personal Service Selector 取得適合的服務，如 3.2 所討論的狀況 1，即是系統內部無法提供與飛機航班相關的服務，只好向 Personal Service Selector 取得與飛機航班相關的服務。取得適合的服務之後，再將執行 Request Data Check，Request Data Check 功能會先檢查使用這些服務所需要的資訊是否充足，若資訊不足將向使用者或是個人化本體論取得使用服務所需要的資料，Request Data Check 也會向個人化本體論取得使用者偏好，最後再將這些資料送入 Planner Generator。

當 Data Gather 把服務資訊和規劃的初始狀態送入 Planner Generator 後，Planner Generator 的執行流程如圖 4-4 所示，Initial Plan State Transform 主要功能是将相關的服務及資訊描述轉換成 JSHOP2 可接收的規劃初始狀態，轉換完後再使用 JSHOP2 產生出一個計劃，此計劃內容為一連串動作的組合，最後 Initial Plan State Transform 會將產出的計劃與 Data Gather 所傳入的資訊一起傳給 Plan Executer。



圖十八 使用者意圖滿足相關軟體模組循序圖

#### 4.5 系統整合

在[1]中，我們介紹了一個以服務為基礎具使用者感知能力的通用系統架構。在圖十九中，我們呈現了一個實現 IASOS 的基本架構與範例。

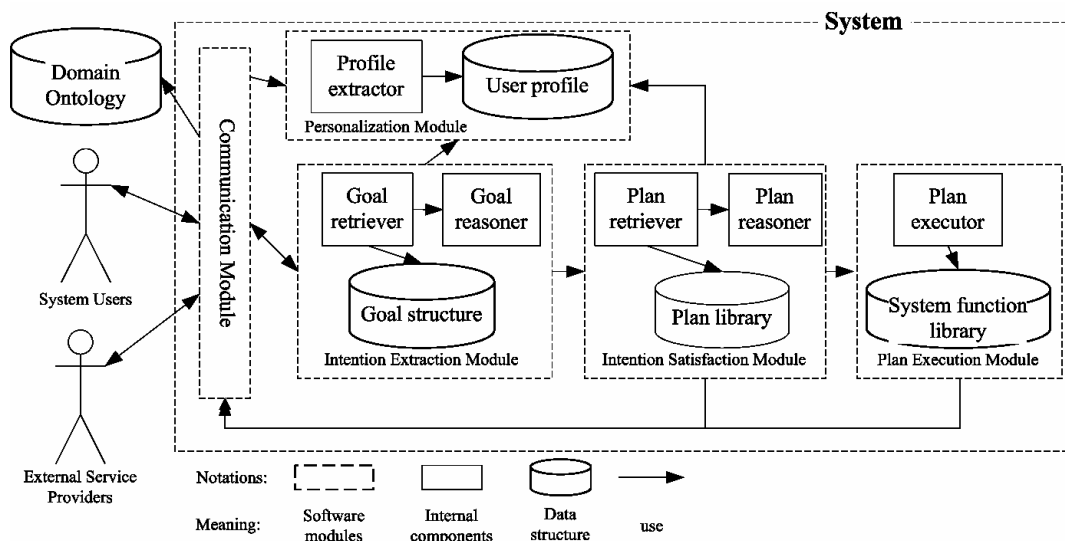
本架構的功能可以被分為三個部份：

1. **個人化的使用者意圖擷取**：本部份由目標結構、目標取回器模組、以及目標推理器模組所組成。目標結構是一個儲存目標模型以及目標模型之間關係的資料結構。目標取回器模組實現了一個以案例為基礎的意圖擷取的演算法，以便依使用者所輸入的詞句取出相關的目標模型。目標的推理器會由這些被取出的目標模型中，根據使用者的偏好，推理出那一個目標模型最能代表使用者的意圖，並加以修正成足以代表使用者意圖的目標模型。

2. **使用者的意圖滿足**：本部份由計劃函式庫，計劃取回器模組、以及計劃推理器模組所組成。計劃函式庫包含了舊有的，曾經成功地被使用來滿足使用者意圖的計劃。為了降低計劃函式庫所佔用的空間，計劃函式庫被設計成階層式的資料結構，以使在不同計劃中的子動作可以被重覆使用。計劃的取回器模組使用被辨識出來的目標模型以及目標模型中的描述部份來取出相關的舊計劃，準備用來滿足新的使用者要求。計劃的推理器模組使用它相關的知識庫來修改由計劃取回器所取回的舊計劃，以便被用來滿足使用者的要求。

3. **計劃的執行**：本部份包括有計劃執行者與系統功能函式庫。計劃執行者執行由計劃推理器模組所送來的計劃，而系統功能函式庫儲存實際系統功能函式的模組。當計劃的執行期間，相關的模組便會被呼叫並執行。

溝通界面模組提供一組函式，以便讓系統可以經由這些函式來與使用者界面與外部的服務配對者進行溝通。系統同時也可以連結到本體論管理程式例如 Protege，來進行本體論資料結構之儲存、查詢、以及更新等動作。



圖十九. 一個實現 IASOS 的基本架構與範例

## 5 結果與討論

本研究的貢獻在於我們提出了如何分析與設計一個具意圖感知與滿足能力的系統。我們分析了由如何使用目標模型來代表使用者的意圖，並且將使用者所輸入的關鍵字連結到設計好的目標模型，並使用人工智慧的規劃方法來滿足使用者意圖的方法與概念。

我們所提出的方法可以結合系統的內部及外部服務來滿足使用者需求。在實作上我們利用人工智慧領域中規劃(Planning)的技術來結合這兩種服務，並產生出一個由一系列可滿足使用者需求的系統動作所構成的計劃。我們也利用案例式推理(Case-Based Reasoning)的技術將所產生的計劃及計劃相關資訊儲存在系統內部的案例庫中(Case Base)，以便當使用者有類似需求時，系統能更快速產生出一個規劃。



當一個系統的設計者要實現新的具意圖感知能力的系統時，設計者的主要工作便會成為設計目標結構、計劃的函式庫、系統的功能函式庫以及目標推理器與計劃推理器的知識庫便可以了。其他大多數的功能模組便可以重覆的被使用。

除了發展 IASOS 的系統架構與流程外，我們也試圖將 IASOS 的概念應用到各個領域，如：股市投資決策系統，智慧型運輸系統等等。

## 6 文獻列表

1. **C.H.L Lee** and A. Liu, "A Goal-Driven Approach for Service Request Modeling," *Int. Journal of Intelligent System*, 2009. (SCIE, EI)
2. **C.H.L Lee** and A. Liu, "Service Quality Evaluation by Personal Ontology," *Journal of Information Software and Engineering*, vol. 25, no. 5, 2009. (SCI, EI).
3. **C.H.L Lee** and A. Liu, "A Study on Intention-Aware Service-Oriented Systems," *Journal of Software Engineering Studies*, 2008.
4. **C.H.L Lee** and A. Liu, "Applying Fuzzy Candlestick Pattern Ontology to Investment Knowledge Management," *Int. Journal of Internet Technology*, vol. 9, no. 4, pp. 307-315, 2008. (SCIE, EI).
5. **C.H.L. Lee**, "Modeling Personalized Fuzzy Candlestick Patterns for Investment Decision Making," *Proc. of the Asia-Pacific Conference on Information Processing (APCIP-2009)*, Shenzhen, China, July, 2009.
6. **C.H.L. Lee** and Alan Liu, "Modeling Explicit and Implicit Service Request for Intelligent Interface Design," *Proc. of the Second International Workshop on Intelligent Interfaces for Human-Computer Interaction (IIHCI-2009)*, Fukuoka, Japan, March, 2009.
7. **C.H.L. Lee**, Alan Liu, and K.Y.R. Cheng, "Service Request Interpretation Using Case-Based Reasoning," *Proc. of the 4th IEEE Symposium on Service-Oriented Systems Engineering (SOSE2008)*, Jhonglii, Taiwan, Dec, 2008.
8. **C.H.L. Lee**, Alan Liu, and K.Y.R. Cheng, "A Case-Based Planning Approach for Agent-Based Service-Oriented Systems," *Proc. of the IEEE SMC 2008*, Singapore, 2008.
9. **C.H.L. Lee**, K.Y.R. Cheng, and Alan Liu, "Designing Intelligent Agents for Traffic Delay Compensation," *IDPT 2008*, Taichung, 2008.
10. **C.H.L Lee** , "Toward Intention-Aware Services Provision," *Proc. of IEEE TENCON 2007*, Taipei, 2007.
11. 李俊宏、劉立頌, "具使用者意圖感知能力的服務導向的系統之研究," 第三屆台灣軟體工程研討會 (TCSE2007), July, 2007, 台中, 台灣.
12. 李俊宏、洪偉智, "一個輔助投資知識管理之 K 線知識本體," 國家計算機會議 NCS2007, 台灣, 台中.
13. H. Wang, J.Z. Huang, Y. Qu, and J. Xie, "Web services: problem and future directions," *ELSEVIER J. Web Semantics*, vol. 1, no. 3, April 2004, pp. 309-320.
14. W3C, "Web Services Architecture," <http://www.w3.org/TR/2002/WD-ws-arch-20021114/>, November 2002.

15. K. Sycara et al, "Automated discovery, interaction and composition of Semantic Web services," *ELSEVIER J. Web Semantics*, vol. 1, no. 1, 2003, pp. 27 – 46.
16. M.P. Singh and M.N. Huhns, *Service-Oriented Computing: Semantics, Processes, Agents*, John Wiley & Sons. Ltd., New York, 2005.
17. J. Lee, N.L. Xue and K.Y. Kuo, "Structuring requirement specifications with goals," *Information and Software Technology*, vol. 43, 2001, pp. 121-135.
18. J. Lee and N.L. Xue, "Analyzing user requirements by use cases: a goal-driven approach," *IEEE Software*, vol. 16, no. 4, 1999, pp. 92-101.
19. N. F. Noy and D. L. McGuinness, "Ontology Development 101: A guide to Creating Your First Ontology," Technical Report KSL-01-05, Stanford Medical Informatics, Stanford University, 2001.
20. B. Chandrasekaran, J.R. Josephson, and V.R. Benjamins, "What Are Ontologies, and Why Do We Need Them," *IEEE Intelligent Systems*, vol. 14, 1999, pp. 20-26.
21. W3C, "OWL Web Ontology Language Overview," <http://www.w3.org/TR/owl-features/>, Feb. 2004.
22. <http://www.daml.org/services/owl-s/>
23. M.N. Huhns and L.M. Stephens, "Personal Ontologies," *IEEE Internet Computer*, 1999, pp. 85-87.
24. J. Chaffee and S. Gauch, "Personal Ontologies for Web Navigation," In Proc. 9th international conference on Information and Knowledge Management (CIKM'00), 2000, pp.227-234.
25. C-S Lee, Z-W Jian, and L-K Huang, "A Fuzzy Ontology and Its Application to News Summarization," *IEEE Transactions on Systems, Man and Cybernetics Part B*, vol. 35, no. 5, Oct. 2005, pp. 859-880.
26. C-S Lee and C-Y Pan, "An Intelligent Fuzzy Agent for Meeting Scheduling Decision Support System," *Fuzzy Sets and Systems*, vol. 142, no. 3, 2004, pp. 467-488.
27. J. Pathak, N. Koul, D. Caragea, and V. Honavar, "A framework for semantic web services discovery," Seventh ACM International Workshop on Web Information and Data Management (WIDM 2005), Bremen, Germany, November 4, 2005, pp. 45-50.
28. C.H.L Lee and A. Liu "User intention satisfaction for agent-based semantic Web services systems," Proc. of The 12th Asia-Pacific Software Engineering Conference (APSEC'05), Taipei, Taiwan, Dec, 2005.
29. S. Russel and P. Norvig, *Artificial Intelligence: A Mordern Approach 2ed.*, Prentice Hall, London, 2003.
30. M. Nikola, M. Miroslaw, "Current Solutions for Web Service Composition," *Proc. of IEEE Internet Computing Online*, Vol. 8, no. 6, 2004, pp. 51-59.
31. E. Sirin , B. Parsia, D. Wu, J. Hendler and, D. Nau, "HTN Planning for Web Service Composition Using SHOP2," *Journal of. Web Semantics*, vol. 1, no. 4, 2004, pp.377–396.
32. M. Pistore, F. Barbon, P. Bertoli, D. Shaparau, and P. Traverso, "Planning and monitoring web service composition," Proc. of International Conference on Artificial Intelligence, ethodologies, Systems, and Applications (*AIMSA*), 2004, pp.106-115.

33. S. Kalepu, S. Krishnaswamy, and S. W. Loke. "Verity : A QoS Metric for Selecting Web Services and Providers," Proc. of the 4th Intl Conf. on Web Information Systems Engineering Workshops (WISEW'03), Dec. 2003, pp. 131–139.
34. S. Kalepu, S. Krishnaswamy, and S. W. Loke, "Reputation = f(User Ranking, Compliance, Verity)," Proc. of the 2004 IEEE International Conference on Web Services (ICWS 2004), San Diego, July 6-9, 2004, pp. 200-207.
35. Raghuram M. Sreenath and Munindar P. Singh, "Agent-Based Service Selection," *Journal on Web Semantics (JWS)*, vol. 1, no. 3, April 2004, pp. 261-279.
36. W.-T. Balke and M. Wagner., "Towards Personalized Selection of Web Services," In Proc. of the 12th International World Wide Web Conference (WWW 2003), Budapest, Hungary, 2003.
37. U. Kuter, E. Sirin, D. Nau, B. Parsia, J. Hendler, "Information gathering during planning for web service composition," Proc. of 3rd International Semantic Web Conference (ISWC -2004), Hiroshima, Japan, November 2004.
38. B. Limthanmaphon and Y. Zhang, "Web Service Composition with Case-Based Reasoning," Proc. of the Fourteenth Australasian database conference on Database technologies 2003, pp.201-208.
39. A. Aamodt, E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches," *AI Communications*. Vol. 7 no 1, 1994, pp. 39-59.
40. J. A. Recio-Garcia, .et. al, "The WINGS of jCOLIBRI: A CBR Architecture for Semantic Web Services, " Proc. of Third International Conference on Next Generation Web Services Practices, NWeSP 2007, pp. 7-12.
41. R. Cheng, et. al, "Using Case-Based Reasoning to Support Web Service Composition" Lecture Notes in Computer Science, vol. 3994, 2006, pp. 87-94.
42. K. H. Huang and A. Liu, "Service Composition Using Planning and Case-Based Reasoning," Proc. of the Nineteenth International Conference on Software Engineering and Knowledge Engineering (SEKE'07), PP. 610-615.
43. G. L. Morris, *Candlestick Charting Explained: Timeless Techniques for Trading Stocks and Futures*, 2nd. ed., McGraw-Hill Trade, pp. 8-139, 1995.
44. C.H.L Lee and A. Liu "Modeling the query intention with goals," Proc. of The First Int. Workshop on Ubiquitous Smart Worlds (USW2005) (Conjunction with AINA2005), Taipei, Taiwan, Mar. 2005, pp. 535-540.

## 計畫成果自評

### **第一年成果：建立 IASOS 的基本軟體元件與完整架構**

在第一年我們依照既定行程進行研究開發完成了使用者意圖擷取與使用者意圖滿足相關軟體元件之設計與發展。相關成果亦發表於國內外知名研討會當中。而其中整個 IASOS 的想法發表於 *International Journal of Software Engineering Studies* 期刊中。而關於應用個人本體論於網路服務的成果更是發表於知名 SCI 期刊 *Journal of Information Software and Engineering* 當中。

綜合而言第一年的三個目標：建立用以描述使用者需求的目標模型；建立使用者意圖擷取機制；建立使用者意圖滿足機制；均能順利完成，且能將研究成果至相關研討會與國際學者進行討論。

### **第二年成果：發展以 IASOS 架構為基礎的 SOA 系統開發方法**

第二年我們以第一年的經驗為基礎發展以 IASOS 架構為基礎的 SOA 系統開發方法。相關成果亦發表於國內外知名研討會當中。而其中整個 IASOS 開發流程的想法已經被知名 SCI 期刊 *Int. Journal of Intelligent System* 所接受，目前正準備出版當中。此外我們也發現研發成果的其他應用，我們將個人化本體論應用於股票投資領域，成果也被另一 SCIE 期刊 *Int. Journal of Internet Technology* 所接受並發表。

綜合二年成果，本研究之成果除了原本所預期之外，同時又多了些其他的應用成果，總合成果可謂豐碩。在二年時間，我們共發表了四篇期刊論文以及八篇國內外研討會論文。在參與相關研討會時，我們的研究成果也獲得其他參與學者的興趣與肯定。在欣喜之餘，主持人同時也相當感謝國科會對學術新進人員之支持，讓本研究得以順利進行並完成。

# 可供推廣之研發成果資料表

可申請專利       可技術移轉

日期：2009 年 10 月 31 日

<p><b>國科會補助計畫</b></p>	<p>計畫名稱：有使用者意圖感知能力以服務為導向系統之研究</p> <p>計畫主持人：李俊宏</p> <p>計畫編號：NSC 96-2218-E-343-002-MY2 學門領域：WEB 技術</p>
<p><b>技術/創作名稱</b></p>	<p>一個以目標導向開發服務導向系統之方法</p>
<p><b>發明人/創作人</b></p>	<p>李俊宏</p>
<p><b>技術說明</b></p>	<p>中文：我們提出一個目標導向的方法來模型化服務導向系統中的使用者要求意圖。服務要求意圖可由使用者輸入擷取，並且由事先定義好的目標模型來模型化。我們將這個問題稱為使用者意圖擷取。若一個服務導向系統助有使用者意圖擷取能力，並且能產生一些行為來滿足擷取的意圖，系統能為使用者提供一個更加便利而且有效率服務。我們由軟體需求工程的角度出發來設計這樣的系統。在系統建構的階段，系統需求規格書中的功能性與非功能性需求將會藉由以目標為基礎的需求分析方法得到。而一系列的目標模型將會從這些需求中被擴充出來。系統的設計者可以根據這些需求分析的結果，建構出系統所能提供的能力。基於所提供的意圖擷取方法，使用者模糊不明確的意圖將會被取出並且對應到電腦可以計畫的目標模型。</p>

	英文：We proposed a goal-driven approach to model the service request intention in service-oriented systems. The service request intention can be extracted from the user input and modeled by predefined goal models. We identify this problem as the service request intention extraction. If a service-oriented system has the abilities of user's intention extraction and can make some activities to satisfy the extracted intention, the system can provide a more convenient and efficient service for the user. We start the system construction from the view of goal-driven requirements engineering. The requirements specification is generated by the goal-based requirements analysis in which the functional and nonfunctional requirements will be extended with goal models. A set of computable goal models that represents the user requirements is selected and refined as the basis of system services. The designer can also design related system services based on the requirements specification. Based on the proposed intention extraction approach, the user's vague and imprecise intention will be extracted and mapped to computer understandable and computable goal models for representing the intention.
可利用之產業 及 可開發之產品	網路服務系統開發業  具使用者輸入界面之網路服務系統
技術特點	能結合一般的網路服務系統提供更好更有效率的使用者界面
推廣及運用的價值	加速網路服務系統之開發

- ※ 1. 每項研發成果請填寫一式二份，一份隨成果報告送繳本會，一份送貴單位研發成果推廣單位（如技術移轉中心）。
- ※ 2. 本項研發成果若尚未申請專利，請勿揭露可申請專利之主要內容。
- ※ 3. 本表若不敷使用，請自行影印使用。

## 【出席國際學術會議報告】

2009 年人機互動智慧型界面國際研討會

2009 International Workshop on Intelligent Interfaces for Human-Computer Interaction, IIHCI2009 (conjunction with CISIS2009)

南華大學資訊工程學系 助理教授 李俊宏

### 一、參加會議經過

此次筆者所獲邀參加之「2009 年人機互動智慧型界面國際研討會」(International Workshop on Intelligent Interfaces for Human-Computer Interaction, IIHCI2009) 為 International Conference on Complex, Intelligent and Software Intensive Systems, CISIS2009 中的一個 workshop。整個研討會時間為 2009 年的三月十六日起至三月十九日止，一共是四天的時間，與於日本福岡市 (Fukuoka, Japan) 之福岡工業大學所舉行。共有來自多個國家相關學者近三百人參與了研討會，大會除了安排學術論文於各場發表會中進行口頭以及壁報形式的發表外 (Oral & Poster Presentations)，也同時在安排了 Osaka 大學的 Shojiro Nishio 教授擔任 Keynote Speaker，針對網路上大量資訊之處理與整合進行演講；而亦有不少國內數所教學及研究機構如中央大學及北科大之專家學者前往參與並發表相關論文，整個會議盛況之熱烈由此可見。

由於福岡為日本九州重要的國際都市，由台灣前往之交通尚稱方便，然而其相關之生活開銷較台灣高一些。在經費、時間、及即時赴會等因素的考量下，本人選擇了三月十五日當天從台北桃園機場搭乘班機前往福岡。本人的論文被安排在會議的第一天 (三月十六日) 發表，為了多擷取一些專業領域上新的知識，個人在第一天至大會報到，報告完論文後再留下來聆聽其他相關場次的論文發表，由於筆者也是第一次至日本福岡工業大學，順道利用論文發表之餘參觀比較日本大學與台灣的大學之不同。

本次會議中筆者的論文“Modeling Explicit and Implicit Service Request for Intelligent Interface Design”中，由於提出利用目標模型以及軟體工程方法進行智慧型界面之設計，對於專注於人機界面應用的其他學者來說，此乃一創新之應用，且能與其研究結合。因此本論文得到與會的相關專長之學者專家的興趣與討論，筆者也因此受邀擔任此一 workshop 來年之委員，對我們的研究是一大肯定。另一方面，筆者在會議中還利用時間參與了一些其他口頭報告與壁報展示的論文場次，對於與本人研究相關的研究亦特別的加以關注，並與它們的作者相互交換一些心得與經驗，實在是一個難得的機會。

## 二、與會心得

由於IIHCI與CISIS/ARES 二個大型研討會一起舉辦，而研討會主題包含相當廣因此其整體規模甚大，論文的接受率平均也在50%以下，具相當學術水準。由於我國在相關領域之學術界或產業界人數與先進國家相比較少，使得參與此水準學術會議的先進在人數上顯得較為欠缺。另一方面也感受到幾個如中國大陸、南韓及日本等國家地區上相關研發的投入及規模，實在值得我們警惕。

但是無論如何，本人藉由所參與交流之會議過程中與相關學者專家之交流，也確實地吸收了不少的心得與經驗，特別是同一場次發表論文之義大利學者對於個人所提方法在應用上的建議讓作者獲益良多。

行萬里路讀萬卷書，第一次至日本福岡參加國際研討會，除了學術研究上的收獲，也擴增了視野。在四天的研討會進行中，筆者也抽空買了火車的九州daypass到九州其他地方參觀，順道了解日本為何能成為世界強國之一。令筆者最深刻的是各個地方乾淨的街道與和善的人民。台灣的人民在友善程度不會不及日本，然而整齊、乾淨和秩序卻是我們還可以向日本學習的。

## 三、感言

在研討會結束後，個人認為獲益最大的應是協助承辦本次研討會的福岡工大。除了舉辦國際研討會所帶來的正面廣告效益外，這類型國際研討會更有助於學校之國際化，當來自各國學者穿梭於校園之中時，對學生也是一大激勵。對台灣而言，舉辦這類型學術研討會除了學術上的效益，更有助於台灣國際形象之提昇。或許國內大學在爭取相關類似國際研討會時能更加積極，而政府單位也可以提供更多資源協助這些國際活動之爭取。

很感謝國科會在計畫內提供本人出席國際會議開銷費用，雖然不可能對我們出國的實際開銷完全補足，但在精神上的鼓舞及使用上的彈性仍然是非常值得我們所肯定的。

## 四、攜回資料名稱及內容

Proceedings of the International Conference on Complex, Intelligent and Software Intensive



Systems, CISIS2009。

論文集由 IEEE Computer Society 出版同時收錄於 IEEE Xplore.

# Modeling Explicit and Implicit Service Request for Intelligent Interface Design

Chiung-Hon Leon Lee

*Department of Computer Science and  
Information Engineering  
Nanhua University  
Dalin, Chia-Yi, 620, Taiwan  
chlee@mail.nhu.edu.tw*

Alan Liu

*Department of Electrical Engineering and  
Center for Telecommunication Research,  
National Chung Cheng University  
Min-Hsiung, Chia-Yi, 621, Taiwan  
aliu@ee.ccu.edu.tw*

## Abstract

*We propose a goal-driven approach to model explicit and implicit service requests for intelligent interface design. This approach starts the system construction from the view of software requirements engineering. Requirements of the system are generated by the goal-based requirements analysis in which functional and nonfunctional requirements will be represented by a set of goal models. These goal models are used to represent possible explicit and implicit requests in the user's service requests. The goal models will be future selected and refined by performing a proposed service request interpretation process. The process is used to extract the user's explicit and implicit service request in an input service request phrase. A service request interpreter is implemented to demonstrate the proposed approach.*

## 1. Introduction

How to bridge the gap between human users and computer-based systems is a long-term research issue in Human-Computer Interaction (HCI), Human-Robot Interaction (HRI) [1], and natural language understanding [2]. If a computer system has ability to understand the user request from a simple command or direction, the system can provide more convenient and efficient services [3].

In the real world, it is easy for a human assistant to understand the request of his boss from a simple direction. For example, a direction might be "I will go to Fukuoka to attend IHHCI workshop." A smart assistant can understand that the boss might want him to arrange travel and living issues for him. He could interact with the boss and use Internet to get more information about the workshop, assert this event into boss's schedule, and handle the related travel schedule for the boss such as booking flight, arranging a taxi

from airport to the workshop venue, and reserving a room etc.

However, to a computer system, because it lacks of common sense and background knowledge, how to interpret the user's requests becomes a nontrivial task. Beside, how to infer or 'guess' implicit user requests behind the user's simple direction also needs more auxiliary information such as the user's preference, system capabilities, and context information.

In the research fields of HRI or HCI, there are several ways to facilitate the context information gathering and user intention identification such as visual recognition of hand and body gesture, conversational interaction, force-feedback tactile glove, or fusing the multimodal input [3, 4], etc. However, in many software applications, such as the network search engines and Web service systems [5], a keyboard and a mouse are most general input devices for user-computer interaction. How to create an interface smart enough to infer implicit service request becomes an interesting research challenge.

In this paper, we proposed a goal-drive approach to model and extract user's explicit and implicit service requests. The approach starts the system construction from the view of software requirements engineering [6]. We assume that any service request from the user should not exceed the boundary of the system application domain. For example, the user should not request a travel scheduling system to buy a book from a bookstore for him. The target of acquiring precise requirements in software engineering is similar to the target of extracting a service request intention from system services users. The difference is that the user does not provide too much information about what he wants when requesting system services. A requirement engineer can elicits the requirements from a user by frequently interactions but a system user cannot stand a computer system to ask him too much problems.

In our approach, requirements of the system are generated by the goal-based requirements analysis in which the functional and nonfunctional requirements will be represented by a set of goal models [7]. These goal models will be future selected and refined as the basis of the interface inference knowledge. Based on these predefined goal models, the system could infer user's implicit intention from user entered service request phrase. The relationships of service request, goal models, and software requirements are shown in Fig. 1. We believe that such a mechanism can provide alternative solution for services access enhancement in different applications such as multiple service-robots systems and semantic Web services systems [8, 9].

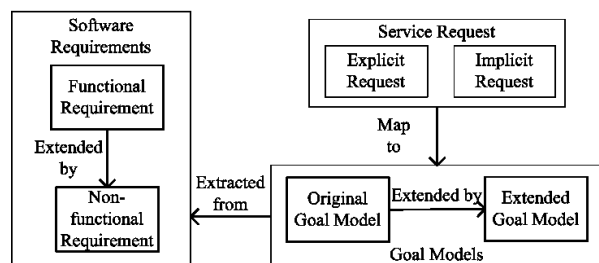


Fig. 1. The relationships among service request, goal models, and requirements.

In our approach, a service request  $R$  is represented by a vector space which consists of a set of terms extracted from the user's entered string. In other words,  $R = \{T_1, T_2, \dots, T_n\}$ , in which  $T$  indicates terms in the service request and  $n$  is the number of terms. The goal model  $G$  is used to represent user's implicit and explicit intention in service requests. All goal models are elicited from software requirements and also are an abstract description of the capability which the system can perform. The goal can also be an aim (an object or goal) that guides actions (system capabilities) to achieve the user's request.

We called the process of mapping a service request to goal models an intention extraction process  $IEP$  which can be described as  $IEP: R \rightarrow G$ . It means that given a service request  $R$ , after the  $IEP$ , an abstract description of system capability  $G$  can be obtained. The mapped goal model not only represents the explicit intention in the service request but also relates to a set of extended sub-goal models to represent implicit intention in the request.

A plan  $P$  is composed by a series of system operations. An action  $A$  describes a single or a composite system function. We called the process of generate and execute a plan to satisfy related goals an intention satisfaction process  $ISP$  which is represented as  $ISP: G \rightarrow P$ . It means that given an original goal  $G$ , after the  $ISP$ , the system should derive a plan  $P$  to

achieve  $G$ . How to generate and execute the plan and monitor the execution of the plan are also important issues when constructing an intelligent system. However, in this paper we focus on the discussion of how to model service request and it's application.

This paper is organized as follows. First, we introduce a goal model for service requests representation. In Section 3, a rule-based request interpretation approach is proposed. A service request interpreter to implement the proposed approach is shown in Section 4. Finally, we give a conclusion for this paper.

## 2. Goal Representation

A service request will be an aim (an object or goal) that guides actions (system functions) to achieve it. Goals identification is a crucial factor in the elicitation of software requirements. In [10], the authors proposed a structure for analyzing software requirements based on a verb and its parameters. We adapt their method and proposed a goal model for representing service requests. The goal will guide the system to generate a plan and the execution of the plan will be expected to satisfy service requests.

There are three requirements on the goal model in our approach. First, the goal model should contain enough information for mapping the user's entered string to a computable goal model. Second, the goal model should have a classification basis for the system designer to analysis their relationships. Third, the goal model should contain the information of related sub-goal models and plan to facilitate the retrieval process for the goal models and plans. Based on these considerations, the basic attributes of a goal model are shown in Fig. 2. A goal is composed by four parts: *contents* to represent service request variables, *properties* to describe attributes of the goal, *relationships* to link original goal, related goals, and sub-goals, and a *plan* linkage will guide the system to retrieve and modify related plan to achieve the goal. Using this approach, a service request can be describe by goal models. A goal structure consists of goal models. The goal structure represents a space in which possible service requests could be identified.

### Contents

The contents consist of *action*, *object*, *constrain*, and *parameters*. The *verb* in the service request string will be interpreted as an *action* which should be performed to satisfy the goal. *Object* indicates the object which will be affected or generated by the *Action*. The noun in the sentence might be an *Object*. The adjective and adverb play the role to constrain the

execution of the *action*. The nouns are assigned to parameters which will be the parameters of the action.

For example, considering a service request like “Schedule an inexpensive flight to Fukuoka on March 15,” the service request itself will be the *Contents*, the term “schedule” will be an *Action*, “flight” is the *Object* of the *Action*, “March 15” and “Fukuoka” are parameters of the *Object*, and “inexpensive” is the *Constraint* of the flight type selection.

Different from the requirement representation in software engineering, a service request string is not a complete sentence. It is composed by different request variables. In our approach, the position of the request variable is not limited. This makes the system implementation more difficult but lets the users have more freedom to use the system like using the Web search engine.

The system uses the variables and synonyms of variables in the input string to match the pre-defined goal structures. If the same verb can be found in the pre-defined goal structure, the matched goal model will be selected as the candidate goal model.

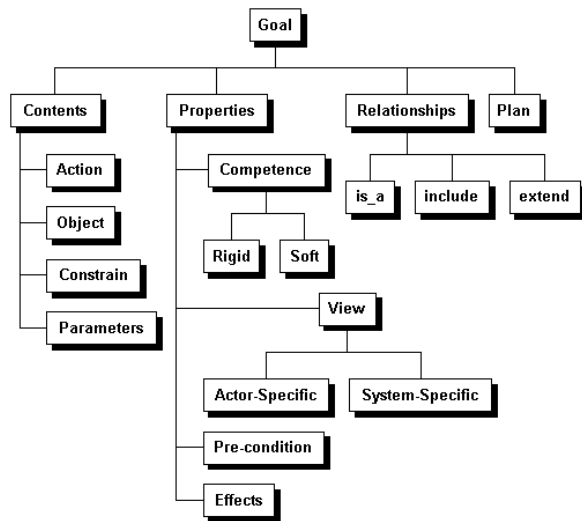


Fig. 2. The goal model.

## Properties

The properties provide a classification basis of the goals. We use them to analyze the interaction among goal models when constructing the goal models. The works proposed in [11] is adapted to define the properties of the goal model. Four properties are considered as the classification basis of goal models: *View*, *Competence*, *precondition*, and *effects*.

*View* concerned whether a goal model is *actor-specific* or *system-specific*. An *actor-specific* goal model is an objective of the human user in using the system. A *system-specific* goal model is requirements

on functions that the system provides to support the *actor-specific* goal. The original goal derived from the user’s request is always an *actor-specific* goal because it reflects a service request that the user wants the system to complete. Because the purpose of goal models in our approach is to map a service request to a series of system activities, we take the point of view from whether the goal is used to represent the requirement from human user or a function of the system to aid the achievement of the user-specific goals. For example, “Schedule a flight” is a *actor-specific* goal and “Select an inexpensive flight” is a *system-specific* goal to extend the *actor-specific* goal.

The definition of *Competence* property classifies the goal as soft or rigid. A rigid goal must always be satisfied. A soft goal is desired to be satisfied and can be satisfied to a degree. If a goal is soft, there is a set of evaluation criteria to evaluate the satisfaction degree of the goal. For example, “Rent a car” is a rigid goal, because the system has to rent a car for the user. There are only two results of this goal: satisfied or fail. Taking another example, “Rent a convenient car” is a soft goal, because there is no clear cut about the term “convenient”. If the system wants to determine whether a car is convenient or not, it needs to refer a set of evaluation criteria such as the car type, user preference, price, etc.

Finally, the pre-condition is used to indicate the achievement of a goal to be possible. The effects display whether a goal has been achieved or completed and related information.

## Relationships

There are three relationships defined to support the linkage of goal models and sub-goal models: *is\_a*, *include*, and *extend*. The designer can specify a specialization relationship between goal models by *is\_a* relationship. The concept of *include* is similar to the composition relationship of a class in object-oriented design. The extend goal model can be used to make the original goal achieved in a ‘better’ situation. Fig. 3 shows a scheduling flight example to demonstrate these relationships. *Schedule\_Flight* is the original goal and composed of *Query\_Flight*, *Select\_Flight*, and *Book\_Flight* sub-goals. *Select\_Proper\_Flight* is used to enhance *Select\_Flight*. *Schedule\_Flight* inherit all properties of *Schedule\_Vehicle*.

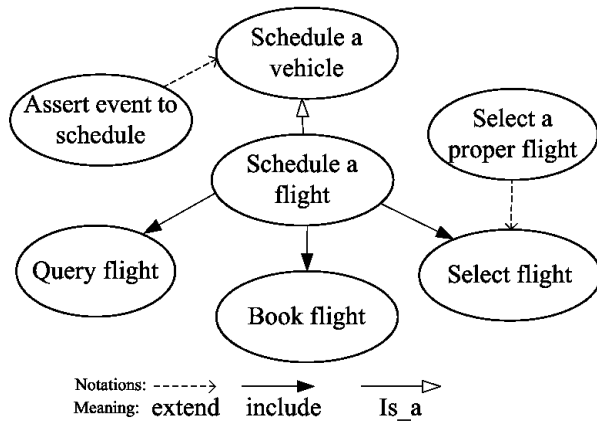


Fig. 3. An example of goal model relationships.

### Plan

The plan is a recipe that records how to achieve the goal. By using the information of the goal model and the execution state of the system, the system can plan some useful strategies to achieve the user's service request. These strategies constitute a serial of actions which will try to satisfy purpose of the goal model. If some successful or failed results have been returned, these messages will be passed to the system. If the main goal of the user is failed, the partial result completed by the sub-plans can also be display to the user.

Just likes a plan can be invoked by the goal, a set of goals can also be generated when performing the plan. The goal-plan hierarchy can be constructed in static way or dynamic way depended on the system statement.

The string acquired from the user initiates an original goal and the goal will link to an original plan, the sub-goal and sub-plan is generated iteratively by system. The process of goal satisfaction is a decision processes. What sub-goal should be generated or which sub-plan should be invoked depends on the system status and the information from the environment.

Fig. 4 shows a one shot of the goal plan hierarchy. A plan is composed by several actions. An action can invoke an external services performed by other systems, generate another goal, or call a system function for achieving a related goal.

Forming an endless loop in the goal-plan hierarchy should be avoided and the execution steps and time of a plan should be limited in a boundary. The goal-plan couple in the goal structure ontology should be well defined.

A process of design the goal-plan couple is defined as follows.

1. Identifying the actor.  
Three actors can be identified: the system user, external service providers, and the system self.

2. Identifying the service request of the actor.
3. Designing plans to achieve the goal.  
The system designer determines a serial of actions to achieve the related goal.
4. Repeating step 1-3 until all possibilities has been found.
5. Analyzing the relationship among goal, plan, and action.

5.1 The endless loop of the goal-plan couple should be check carefully.

5.2 The side-effect between goal and actions should be check carefully. Sometimes a goal will be ceased by some actions.

6. Using the analyzing result reconstruct the goal-plan couples.

The aim of goal-plan couple analyzing is to discover and elicit the possible service request of the user and give a original plan to achieve the goal.

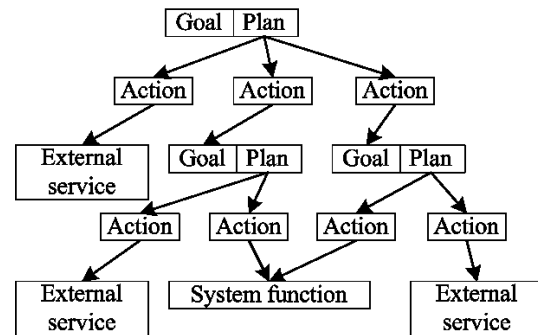


Fig. 4. The goal-plan hierarchy.

### 3. Service Request Interpretation

There are different ways to interpret a service request. In this paper, we suggest a rule-based service request interpretation process. First, the system identifies the verb in the input string as the action which should be performed to satisfy the request. If there is no verb in the string, the verb of the goal will automatic assigned by using the information in the *object* and *parameter* field.

Second, the system tries to find the related goal models in the goal structure by keyword matching. Finally, the system pick up a most likely goal model from candidate goal models to represent the user intention.

Request variables abstraction and goal identification are two main tasks in the service request interpretation process. The former is to find the position of the query terms in the domain ontology and to annotate the variables in relation information. The latter is to identify the goal by comparing the annotate request variables and the goal structure. A user profile is also used as the background knowledge to judge the goal which is reasonable to the user of not. Three

processes compose the request interpretation procedure: input string parsing process, goal generating process, and the goal selecting process.

Fig. 5 shows the procedure of service request interpretation process. Three modules parser, goal generator, and goal selector are designed to perform the process. The domain ontology, goal structure, and user preference are used to support the process.

Different from the way of using the original input string as service request keyword, our approach generate the goal model not only by the original input string but also by the domain, user, and system information.

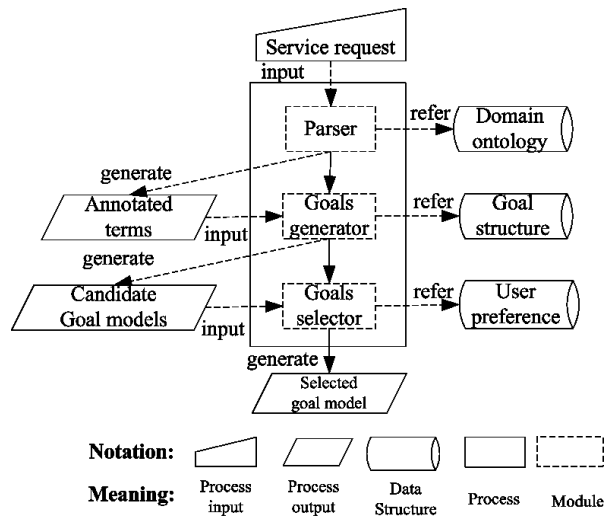


Fig. 5. The service request interpretation process.

## 4. Implementation

User's service request interpretation is a knowledge-intensive and non-trivial task. The interpretation processes is like a serial decision making process. We apply the proposed techniques introduced in Section 3 to implement a service request interpreter (SRI). Based on this interface, the designer can design an intelligent service requester agent. The requester agent is a software entity which performs the tasks such as accepting the service request string from the user, extracting the user's explicit and implicit intention from the request string, generating and executing a plan to satisfy the user's requests, etc. The benefits of embedding the service requester agent with SRI are that the user-system interaction times can be reduced and the system usage will become more conveniently.

An implementation of the proposed method is shown in Fig. 6. There are seven modules in the SRI, one for communicating with other sub-components (communication interface), three for performing the

request interpretation process (service request parser, goal generator, and personalized goal selector), one for user profile processing (profile extractor) and two for data storage (goal structure and user profile). The arrow in the end of line indicates dependency relationship between components. The arrow from component A to component B means that A can use functions or data provided by B. For example, Goal Generator can use the data stored in the Goal Structure, but Goal structure cannot use any function of Goal Generator actively.

There are two lines from the goal selector to communication interface. One line is used to indicate the fact that if there are more than one goal model have been selected by the goal selector, the requester agent has to interact with the user letting the user decide which goal should be selected. The other line is used to indicate the selected goal will be sent to the plan executor component of the requester agent. The plan executor will call a planner to generate a plan for goal achievement.

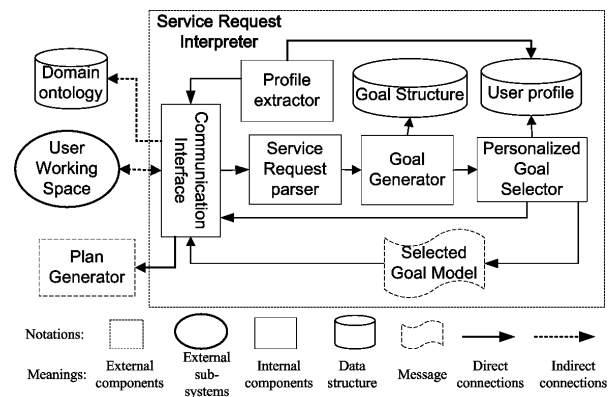


Fig. 6. A design of service request interpreter.

The control flow of the SRI is simple. First, the requester agent receives the service request string from the user working space and forwards the string to the SRI. The SRI performs the service request interpretation process and returns the selected goal model to the plan generator. The requester agent performs the plan which generated by a planner to satisfy the user's request. The successful performed goal will be passed to the profile extractor for constructing the user profile.

## 5. Conclusions

How to model and extract implicit service request in a simple service request phrase is a difficult task. We model the explicit and implicit service request with goals makes the request clear and comparable. The background knowledge is essential to represent the possible goals. Using the information of domain

ontology, goal structure, and user profile the system can parse the request variables, generate candidate goals, and select an original goal to represent the user's service request.

Sub-goal and sub-sub-goal can be generated statically or dynamically when a plan is running. The process of link a plan to a goal and generate sub-goals from a plan is like a decision making process. A tool is introduced to implement the request interpretation process and be used as the basis to a service request agent. We believe that our approach can give an alternative solution when design an intelligent interface for better human computer interaction.

## 6. Acknowledgment

This research was partially supported by the National Science Council in Taiwan through Grant NSC- 96-2218-E-235-002-MY2.

## 7. References

- [1] C. Breazeal, "Social Interactions in HRI: The Robot View," *IEEE Trans. Systems, Man, and Cybernetics*, May 2004, pp. 181-186.
- [2] J. Allen, *Natural language understanding 2<sup>nd</sup> ed.*, The Benjamin/Cummings Publishing Company, Inc., Redwood City, 1994.
- [3] I. Marsic, A. Medl, and J. Flanagan, "Natural communication with information systems," *Proceedings of the IEEE*, vol. 88 , no. 8 , Aug. 2000, pp. 1354-1366.
- [4] S. Iba, C.J.J. Paredis, and P.K. Khosla, "Intention aware interactive multi-modal robot programming," *Proc. of the IEEE RSJ Conf.* 2003, pp. 3479-3484.
- [5] M.P. Singh and M.N. Huhns, *Service-Oriented Computing: Semantics, Processes, Agents*, John Wiley & Sons. Ltd., New York, 2005.
- [6] R. Pressman, *Software Engineering 6/E*, McGraw Hill, New York, 2005.
- [7] J. Lee and N.L. Xue, "Analyzing user requirements by use cases: a goal-driven approach," *IEEE Software*, vol. 16, no. 4, 1999, pp. 92-101.
- [8] H. Wang, J.Z. Huang, Y. Qu and J. Xie, "Web services: problem and future directions," *ELSEVIER J. Web Semantics*, vol. 1, no. 3, April 2004, pp. 309-320.
- [9] N. Gibbins, S. Harris, and N. Shadbolt, "Agent-based semantic Web services," *ELSEVIER J. Web Semantics*, vol. 1, no. 2, Feb. 2004, pp. 141-154.
- [10] C. Rolland, C. Souveyet, and C. Achour, "Guiding goal modeling using scenarios," *IEEE Tran. On Software Engineering*, vol. 24, no. 12, Dec. 1998, pp. 1055-1071.
- [11] J. Lee, N.L. Xue and K.Y. Kuo, "Structuring requirement specifications with goals," *Information and Software Technology*, vol. 43, 2001, pp. 121-135.

**【出席國際學術會議報告】**  
**2009 年亞太信息處理國際研討會**  
**2009 Asia-Pacific Conference in Information Processing,**  
**APCIP2009**

南華大學資訊工程學系 助理教授 李俊宏

一、參加會議經過

此次筆者所獲邀參加之「2009 年亞太信息處理國際研討會」(Asia-Pacific Conference in Information Processing, APCIP2009)乃是於2009 年的七月十八日起至七月十九日止，一共是二天的時間，於中國大陸深圳市 (Shenzhen, China) 之深圳大學所舉行。共有來自亞太地區多個國家研究信息處理超過一百多人參與了此項盛會，大會除了安排學術論文於各場發表會中進行口頭的發表 (Oral Presentations) 外，也同時在整個議程中安排了台灣的張真誠教授擔任Keynote Speaker；而亦有不少國內數所教學及研究機構之專家學者前往參與並發表研究心得。

由於深圳為中國極具代表性的現代化都市，由台灣前往之交通尚稱方便，相關之生活開銷也與台灣相去不遠。然而由於台灣至香港機票票價以及班次遠較台灣至深圳便宜且選擇較多，在經費、時間、即時赴會以及觀察對岸發展等前題因素的考量下，本人乃選擇了七月十七日當天從高雄小港機場搭乘班機前往香港經內地火車轉至深圳入關。雖然本人的論文係被安排在會議的第二天(七月十九日)發表，為了多擷取一些專業領域上新的知識，個人還是在第一天便至大會報到。在早上聆聽完張真誠教授之演講，並與張教授打招呼後，大會突然宣布由於七月十九日當遇到颱風來襲，所有行程均被壓縮至第一天進行。而個人之報告便被移至第一天下午進行。在全程參予了大會在其它幾天的數個與專長相關之場次論文發表過程後，發現由於行程的突然壓縮與改變造成一些口頭報告的人員並不了解情形，而筆者與一些大陸學者交流後，他們也為行程之混亂感到抱歉。本次會議中筆者的論文“Modeling Personalized Fuzzy Candlestick Patterns for Investment Decision Making”由於是以股票投資當作應用領域，現場不少學者詢問的不是專業的問題而是是否可以用筆者所開發的系統賺錢，此乃專業問題之外的趣事一件。此外，筆者在會議中還利用時間參與了一些其他口頭報告的論文場次，對於與本人研究相關的工作亦特別的加以關注，並與它們的作者相互交換一些心得與經驗，了解目前二岸三地資訊處理的相關研究議題。

二、與會心得

近年來可以看到中國大陸承辦資訊相關研討會的次數有逐年增長的趨勢。一方面是中國在國際經



濟上所扮演的角色愈來愈重要，另外也可以看出來中國對資訊相關產業與研究的重視。筆者為第一次至中國大陸參加國際研討會，雖然本次APCIP2009也有與國際電子電機協會合作，進行出版論文集等相關事務。然而由議程之安排以及接待的精緻性來說，其與台灣承辦的國際研討會相比還是有可加強之處。

但是無論如何，本人藉由所參與交流之會議過程中與中國大陸學者專家之交流，也確實地吸收了不少的心得與經驗。另一方面也感受到中國大陸對於資訊產業相關研發的投入及規模，實在值得我們警惕。

由於研討會時間的壓縮，筆者便利用多出來的時間至深圳市區參觀電子資訊相關產品的賣場。在參觀了手機、筆電、以及書局的賣場後，深深的為中國所謂的規模所震驚。在一個較台灣加樂福大賣場為大的場地中各式的手機陳列販賣，人來人往川流不息的場面相當驚人。而在書的大賣場中也有不亞於電子產品賣場的人氣，而且往往可以看到不少人抱著一堆書準備結帳，這種場面或許在台灣只能在書展中才能看到。以台灣目前推動閱讀的成效來看或許中國大陸也有值得我們借鏡的地方。

### 三、感言

行萬里路讀萬卷書。第一次至中國大陸參加國際研討會，雖然行程不是順利，收獲卻也不少!在專業的成長以及眼界的寬廣都有很好的收獲。很感謝國科會在計畫內提供本人出席國際會議開銷費用，雖然不可能對我們出國的實際開銷完全補足，但在精神上的鼓舞及使用上的彈性仍然是非常值得我們所肯定的。

### 四、攜回資料名稱及內容

Proceedings of the Asia-Pacific Conference in Information Processing, APCIP2009。

論文集同時收錄於 IEEE Xplore.

# Modeling Personalized Fuzzy Candlestick Patterns for Investment Decision Making

Chiung-Hon Leon Lee

Department of Computer Science and Information Engineering  
Nanhua University  
Dalin, Chia-Yi, 620, Taiwan  
chlee@mail.nhu.edu.tw

**Abstract**—Candlestick theory is one of widely used technical analysis methods in stock and commodity investment domains. The investors can make their investment decision by observing the change of the candlestick lines and discovering specific candlestick patterns. A candlestick pattern is composed of some candlestick lines. Because different investors have different interpretation of a candlestick pattern, we model different parts of a candlestick line with fuzzy linguistic variables to create a fuzzy candlestick pattern. We also proposed a personal ontology for the candlestick pattern interpretation and decision making. The user can use data mining algorithm such as decision tree to mine some candlestick patterns for investment decision making and the mined candlestick patterns could be stored in a database for different user's future reuse. Our approach can be future used with other financial time series prediction results to provide users more information for investment decision making.

**Keywords**—candlestick theory; fuzzy candlestick pattern; personal ontology; data mining.

## I. INTRODUCTION

Candlestick chart analysis is one of the most widely used technical analysis techniques and many investors believe that it is definitely viable and effective for stock and commodity market timing and analysis [1]. The candlestick patterns are empirical models of investment decision and reflect the psychology of market. The investors make investment decisions by the identified candlestick patterns.

Figure 1 shows an example of the candlestick chart for the stock market. The open, close, high, and low prices in a specific trading time period are recorded in the candlestick lines from  $t_1$  to  $t_{10}$ . The time interval from  $t_1$  to  $t_{10}$  is fixed. On the time  $t_3$ , the price closes at a lowest price and continues the downtrend from  $t_1$  to  $t_2$ . On the  $t_4$ , the opening price is lower than previous closing price, but the price closes at the highest price and leaves a long lower shadow. This situation might be interpreted by an experienced investor as the candlestick line on the day from  $t_1$  to  $t_3$  reflecting a downtrend of the stock price, because there are many investors who want to sell the stock, making the closing price much lower than the opening price. However, the downtrend might reverse itself on  $t_4$ , because there might be investors wanting to buy the stock in the trading period that makes the price close at the highest price and leave a long lower shadow. In other word, the candlestick lines at  $t_3$

and  $t_4$  can be interpreted that the downtrend is bouncing back.

At  $t_9$ , the closing price is higher than the opening price, but the long upper shadow indicates that there are some investors start to sell their stocks. At  $t_{10}$ , the opening price is much higher than the previous closing price, but it closes at lowest price and lowers than the close price on previous day. The lines at  $t_9$  and  $t_{10}$  can represent a reverse, because the downtrend is broken at  $t_{10}$ .

A candlestick pattern is composed by one or more candlestick lines and the trend before the pattern. By the trading experience, the investor tries to identify the candlestick patterns to help themselves to make the investment decisions such as to buy, sell, or hold the stock. There are many existing defined candlestick patterns which are widely used by the investors. For example, in Figure 1, the candlestick line on  $t_4$  and the trend formed by  $t_1$ ,  $t_2$ , and  $t_3$  are defined as a pattern which is called Hammer to represent the downtrend is reversed. Another pattern called Bearish engulfing is also illustrated in Figure 1 and is composed by an uptrend and the candlestick lines on  $t_9$  and  $t_{10}$ .

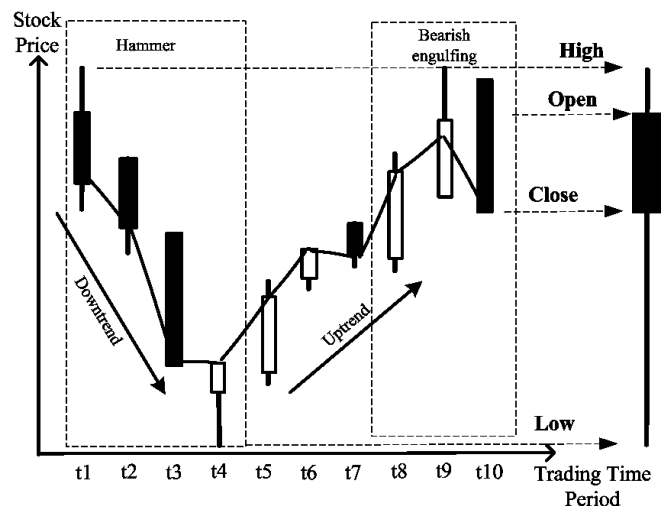


Figure 1. An example of the candlestick chart.

There are many investors making their investment decision by observing the candlestick chart. However, how to identify an effective pattern from the candlestick chart and

how to explain the pattern needs investment experiences in many years to a human investor.

In [2], the author proposed a fuzzy candlestick pattern discovery approach in which the different part of a candlestick line is modeled by using fuzzy linguistic variables and the financial time series data is transferred to fuzzy candlestick lines. The system developer can use data mining algorithm such as decision tree to mine some fuzzy candlestick patterns for investment decision making. However, because different users have different explanation of candlestick patterns and the fuzzy candlestick pattern not only be used to model stock price but also be used to model stock index, fixed pattern definition cannot satisfy different financial time series representation. Figure 2 shows the concept that two users have different interpretation for a concept. For example, when an investor says a candlestick line has “long” upper shadow, to the other investor, the “long upper shadow might be “middle” upper shadow.

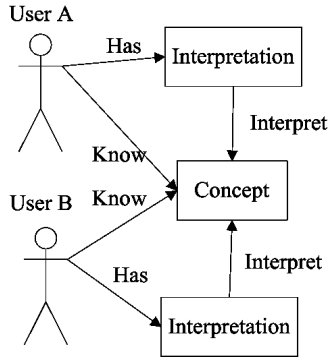


Figure 2. Different users have different interpretation for a concept.

Personal ontology is an approach used in artificial intelligence domain to store information about each individual user. In [3,4], the author proposed an approach that uses personal ontology to record the habit and interests of a user in browsing Web pages. A system then uses such information for recommending the user to view certain Web pages. The authors in [5] use personal ontology in an agent-system for a meeting scheduling system. In [6], personal ontology and domain ontology were used in keeping personal information and food preferences for finding Web services.

In this paper, we combine the concept of personal ontology and fuzzy candlestick pattern to propose a personalized fuzzy candlestick patterns for flexible financial time series modeling. The paper is organized as follows. In Section 2, how to transfer stock trading data into candlestick pattern is introduced. The personalized fuzzy candlestick pattern is proposed in Section 3. Section 4 gives the conclusion of this paper.

## II. CANDLESTICK PATTERN MODELING

The concept of the candlestick line during a trading time period can be found in Figure 1. The first trading price during a trading time period is called open price; the last trading price is called close price; the highest price is called high price, and the lowest price is call low price. The

candlestick chart in the right side of Figure 1 represents the stock price variation from  $t1$  to  $t10$ .

The candlestick line contains a box to makes up the difference between the open and close price. The box is called the body of a candlestick line. The height of the body is the range between a trading day's open price and the day's close price. In this paper we use black color to represent that the closing price was lower than the opening price. When the closing price is higher than the opening, the body is represented by white color. A candlestick line may have small thin lines above and below the body. These lines are called shadows and represent the highest and lowest prices reached during the trading time period. The height of the upper shadow is the range between the high price and the higher price among the open and close prices in a specific time period. The height of the lower shadow is the range between the low price and the lower price among the open and close prices. By the definition of the upper-shadow, lower-shadow, and body, we can transfer the stock price variation in a specific time period into crisp candlestick line by following equations.

$$\begin{aligned} L_{upper} &= [high - \max(open, close)]/open \\ L_{lower} &= [\min(open, close) - low]/open \\ L_{body} &= [\max(open, close) - \min(open, close)]/open \end{aligned} \quad (1)$$

The character “L” of the equation indicates the length of the upper shadow, lower shadow, and body. The terms of open, close, high, and low are the prices in an interested time period. The function of **max** is used to calculate the greater value between the open price and the close price while and the function **min** is for the smaller value between them.

Four fuzzy linguistic variables EQUAL, SHORT, MIDDLE, and LONG are defined to indicate the fuzzy sets of the shadows and body length. Figure 3 shows the fuzzy membership function  $\mu(x)$  of the linguistic variables. The time period and the range of body and shadow length can be defined by users for different applications.

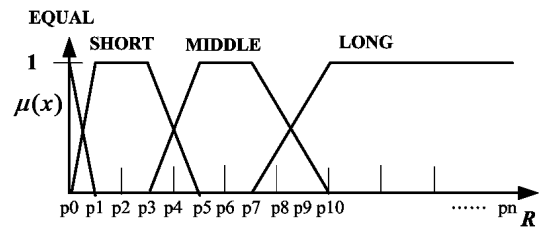


Figure 3. The fuzzy sets of the length of the body and shadows.

A right linear membership function is used to model the EQUAL fuzzy set and is defined by the following formula. The parameters (a, b) are equal to (0, p1).

$$right\_linear(x : a, b) = \begin{cases} 1 & x < a \\ (b - x)/(b - a) & a \leq x \leq b \\ 0 & x > b \end{cases} \quad (2)$$

The LONG fuzzy set is defined by the following left linear membership function. The parameters (a, b) are equal to (p7, p10).

$$\text{left\_linear}(x : a, b) = \begin{cases} 0 & x < a \\ (x - a)/(b - a) & a \leq x \leq b \\ 1 & x > b \end{cases} \quad (3)$$

The membership function of SHORT and MIDDLE is a trapezoid function and the following formula is used.

$$\text{trapezoid}(x : a, b, c, d) = \begin{cases} 0 & x < a \\ (x - a)/(b - a) & a \leq x < b \\ 1 & b \leq x < c \\ (d - x)/(d - c) & c \leq x < d \\ 0 & x \geq d \end{cases} \quad (4)$$

Four parameters (a, b, c, d) of this function to describe the linguistic variables SHORT and MIDDLE are (p0, p1, p3, p5) and (p3, p5, p7, p10).

The body color is also an import feature of a candlestick line and can be simply defined by three terms BLACK, WHITE, and CROSS. The situation where open price equals close price has specific meaning in the candlestick pattern, so a ‘‘CROSS’’ term is defined to describe this situation. In this case, the height of the body is 0, and the shape is represented with a horizontal bar. The definition of body color is defined as follows.

- If open - close > 0 then the body is BLACK.
- If open - close < 0 then the body is WHITE. (5)
- If open - close = 0 then the body is CROSS.

The relationship between candlestick lines also plays an important role in a candlestick pattern. Figure 3 shows the membership function of the linguistic variables the define relationship between two candlestick lines. The candlestick line in the bottom of Figure 3 is the candlestick line of previous trading time. The unit of X axis is the trading prices in previous trading time period and the unit of Y axis is the possibility values of the membership function.

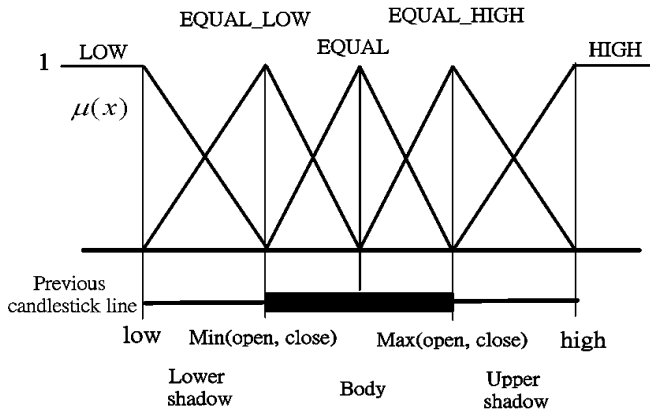


Figure 4. The fuzzy sets of the relationship between candlestick lines.

### III. CONSTRUCTING PERSONALIZED FUZZY CANDLESTICK PATTERN

#### A. Fuzzy Candlestick Pattern Modeling

The description of a candlestick pattern consists of three parts: a pattern name, time period, candlestick lines and a description of previous trend. A candlestick line includes five parts: Open style, Close style, Upper shadow, Body, and Lower shadow. An ontology for candlestick pattern is shown in figure 4.

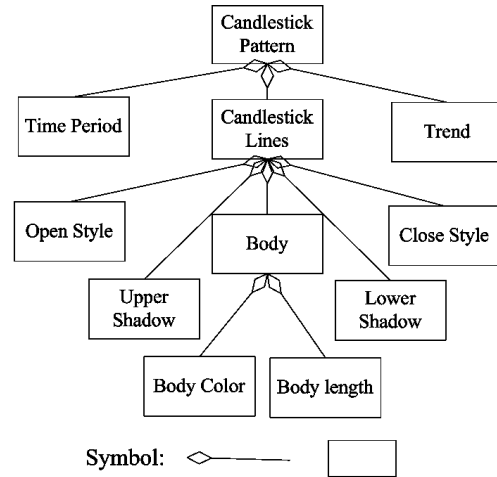


Figure 4. Candlestick Pattern Ontology.

For constructing a personal ontology, we use a concept to represent the interpretation of a user for some concepts. For example, the term 'long' for the upper shadow of a candlestick line might associate with a left linear membership function in parameter (3.5, 5). A candlestick pattern is shown in Table 1 and its interpretation is shown in Table 2.

Table 1 An example of the candlestick pattern.

<b>Pattern name:</b> Bullish Engulfing
<b>Time period:</b> one day
<b>Previous trend:</b> Down Trend
<b>Candle lines</b>
<b>Candle line0</b>
<b>Open style:</b> OPEN_LOW
<b>Close style:</b> CLOSE_HIGH
<b>Upper shadow:</b> null
<b>Body:</b> LONG
<b>Body color:</b> WHITE
<b>Lower shadow:</b> null
<b>Candle line1</b>
<b>Open style:</b> OPEN_HIGH
<b>Close style:</b> CLOSE_LOW
<b>Upper shadow:</b> null
<b>Body:</b> SHORT
<b>Body color:</b> BLACK
<b>Lower shadow:</b> null

Table 2. An interpretation for the candlestick pattern.

<b>User name:</b> Big Bull
<b>To interpret:</b> candlestick line
<b>Concept parameters</b>
<b>Equal:</b> (0, 0.5)
<b>Short:</b> (0, 0.5, 1, 1.5)
<b>Middle:</b> (1, 1.5, 3, 3.5)
<b>Long:</b> (3.5, 5)

### B. Applying Fuzzy Candlestick Pattern

Since the fuzzy candlestick patterns are modeled, we can use the pattern model to identify specific patterns in the candlestick chart and mine identical patterns from the candlestick chart. It is obvious that a pattern can be transferred into the fuzzy rules such as follows for pattern recognition.

```
IF line0_open_style = OPEN_LOW
and line0_close_style = CLOSE_HIGH
and line0_body = LONG
and .....
THEN The pattern = Bullish Engulfing.
```

A pattern recognition rule consists of the crisp part and the fuzzy part. The crisp part includes the previous trend of the pattern and the body color. The others of the rule are the fuzzy part such as the body and shadow length and the open and close style. From our experimental results, well arranged identification rule will reduce the pattern recognition processing time.

Comparing with the processing time of the fuzzy part, the crisp part takes less processing time. For example, the body color includes three possibilities: BLACK, WHITE, and CROSS. For judging the value of the body color, the pattern recognition module only needs to compare the value of open price and close price. The pattern identifying time can be reduced if the judgment of the crisp part is placed before the process of the fuzzy part.

There are different methods could be used to recognize a fuzzy candlestick pattern in the candlestick chart. For example, the concept of Hamming distance [7] could used to measure the similarity among fuzzy candlestick patterns.

The candlestick patterns mining process can be divided into some steps. First step is to define the parameters in the personal ontology. Next are to acquire time series from the database and transfer it into fuzzy candlestick patterns. There might be more than one fuzzy set matched for a single crisp value when finding the value of the linguistic variable. For disambiguates, the fuzzy set with biggest membership value will be selected. The amount candlestick lines which to compose the candlestick pattern are assigned by the user.

Then, based on the following trend, we can use some data mining algorithms such as the ID3 classification algorithm [8] to classify the fuzzy candlestick patterns.

The mined pattern can be easily transferred into fuzzy rules like follows.

```
IF the previous trend = Down Trend,
AND Line 1 body = EQUAL WHITE,
.....
AND Line 0 body = MIDDLE BLACK,
.....
THEN the following trend = STRONG BULLISH.
```

Finally, using the simple mechanism of symbolic matching process, the investor can validate the efficiency of the selected patterns and add comments for the mined patterns.

## IV. CONCLUSION

Because different users have different explanation of candlestick patterns and the fuzzy candlestick pattern not only be used to model stock price but also be used to model stock index, fixed pattern definition cannot satisfy different financial time series representation. In this paper, we combine the concept of personal ontology and fuzzy candlestick pattern to propose a personalized fuzzy candlestick patterns for flexible financial time series modeling.

We believe the personalized fuzzy candlestick pattern can give the user more flexible way to model financial time series and uses the modeled pattern to make more correct investment decisions.

## ACKNOWLEDGMENT

This research was partially supported by the National Science Council in Taiwan through Grant NSC-96-2218-E-235-001-MY2.

## REFERENCES

- [1] G.L. Morris, "Candlestick Charting Explained: Timeless Techniques for Trading Stocks and Futures 2nd edition", McGraw-Hill Trade, 1995.
- [2] C.H.L Lee, A. Liu, and Wen-Sung Chen, "Pattern Discovery of Fuzzy Time Series for Financial Prediction," IEEE Transaction on Knowledge and Data Engineering, vol 18, no. 5, 2006, pp. 613-625.
- [3] J. Chaffee and S. Gauch, "Personal Ontologies for Web Navigation," In Proc. 9th international conference on Information and Knowledge Management (CIKM'00), 2000, pp. 227-234.
- [4] C-S Lee, Z-W Jian, and L-K Huang, "A Fuzzy Ontology and Its Application to News Summarization," (SCI) IEEE Transactions on Systems, Man and Cybernetics Part B, vol. 35, no. 5, Oct. 2005, pp. 859-880.
- [5] C-S Lee and C-Y Pan, "An Intelligent Fuzzy Agent for Meeting Scheduling Decision Support System," (SCI) Fuzzy Sets and Systems, vol. 142, no. 3, 2004, pp. 467-488.
- [6] J. Pathak, N. Koul, D. Caragea, and V. Honavar, "A framework for semantic web services discovery," Seventh ACM International Workshop on Web Information and Data Management (WIDM 2005), 2005, pp. 45-50.
- [7] G.J. Klir, and B. Yuan, "Fuzzy sets and fuzzy logic theory and application", Prentice Hall, Upper Saddle River, NJ, 1995.
- [8] Ian H.W. and Eide F., "Data Mining – practical machine learning tools and techniques with Java implementations", Morgan Kaufmann, San Francisco, 2000.