

導入語義網技術於法規檢核系統同步化之研究  
以全民健康保險藥品給付規定及臨床醫令用藥檢  
核為例

張儀興

南台科技大學資訊管理系

(06)2533131-4300

yhchang@mail.stut.edu.tw

張簡稜剛

南台科技大學資訊管理系碩士班、成大醫院資訊室

(06)2533131-4300

fhopec@gmail.com

# 導入語義網技術於法規檢核系統同步化之研究

## 以全民健康保險藥品給付規定及臨床醫令用藥檢核為例

### 摘要

本文導入語義網技術於法規檢核來實現藥品檢核規則自動更新的功能，其主要設計理念為以語意網的 OWL 模型來實作法規檢核系統，在法規的知識表徵上，系統將所有的藥品規則視為醫療狀態的類別，因此判別用藥的合法性，便可使用分類運算達成，這簡化了推理機的設計；利用 OWL 支援類別繼承的語意，故能形成認知經濟性效應，縮小醫療法規機器編碼長度；最後 OWL 的語法能表示大部份規則型態，使得系統可以使用一個文件檔，一種推理機模型便能表示所有的法規。OWL 簡化了健保給付規定自動化更新到醫院資訊系統的設計，讓醫令系統成為更能即時反應健保法規的決策支援系統。對醫院而言，這大幅節省法規變動時，資訊系統異動成本。其次，對醫事管理單位而言，則能以較少的成本，落實其法規控管於醫療資訊系統上，對抑止醫療資源浪費，及提升病人安全是一大利器。

**關鍵詞：**醫療資訊管理、語意網、決策支援系統

### 1.前言

目前的醫療系統已從傳統式的會計、保險申報系統，提昇至目前能診斷支援決策、病人安全、電子病歷等各式各樣的應用領域。當 IT 能支援醫師作決策後，那麼醫院、政府也可以應用 IT 影響醫師作決策。醫療行為對人類生命影響重大，所以政府都會以法令介入管理，以維護大眾的生命權。傳統法令的執行是藉由相關管理單位及專家，定期檢查醫療執行的記錄，病歷，是否違法，來約束醫師的醫事行為。電子化為政府開啟一個新的控管通道，藉由醫療應用程式來控管醫事行為。當醫院流程電子化程度越高，電子控管已比傳統的法規執行更為確實方便有效。

自從全民健保實施後，醫療資源濫用一直是嚴重的問題，健保局為了控管醫療資源濫用，制定給付規則，這迫使醫院需管制醫師不能開立太多的非給付醫令，否則會造成醫院很大的損失，進而院方必須能讓醫師開立醫令當下，也能考量健保規則。對醫師而言，其天職應專注於診療過程上，而非著重在會計行為。所以一個良好的診療系統必須在醫師開立醫令時，能給予明確的規則資訊，以輔助醫師能專注其診療行為。因此作為醫師最主要的決策支援系統，臨床醫令系統必須能適時隨著健保給付規則更動。但健保局因政策改變或新醫學研究等經常異動規則，使得臨床醫令程式的維護工作變得相當沈重，而同樣的法則，卻必須在每個醫院的系統實作，非常浪費人力。如何建立一個可隨時更新給付規則的規則庫，卻又不用常更動到應用程式，是急迫需求的。

### 2.文獻探討

#### 2.1.國內的醫令決策支援系統現況

醫令輸入系統原本其主要的目的在於節省醫師重複手寫之工作，使醫師能有更充裕時間提供病患診療服務，現今的醫令輸入系統除具備上述交易處理的功能外，更加強了臨床決策支援功能，即時提供醫師用藥警訊[4]，使系統能大大的提昇病人安全。例如藥物開立上，現代的醫令系統大部份支援三類藥物檢核，如以下條列[2]：

- 1.根據醫療狀態對藥物作出限制，像是病人的性別、身高、體重及各種檢驗值等等。
- 2.重覆用藥的檢核。
- 3.藥物交互作用的檢核[3][5]。

#### 2.2 醫療資訊的共享現況

醫療資訊量大，且其判定規則相當複雜，是故整合醫療資訊一直是熱門的主題，早在 1997 年，學者便提出建立臺灣醫療資訊交換中心之構想[6]，健保 IC 卡發行後，IC 卡成為共享醫療資訊的一個重要媒介，也有學者以此作為實作病人安全的檢核的控管資訊[2]。

而目前國內醫令決策支援系統對規則庫的建構方式均是使用傳統式的關聯式模型[2][3][5]，由於每個資料表的關連都是隱含在外鍵內，這使得設計者必須先設計出檢核規則的實體關係模型，再轉換成關聯式模型。同時規則的機器形式表徵難以一致化，在[2]中，由於其嘗試綜合醫令的限制條件，並強調系統對規則的可擴充性，於是獨立出一組後設的限制條件檔用來描述規則的限制條件，但此種作法其解析資料值邏輯並不存在其資料中，而是寫在應用程式裡，是故一定要利用額外的文件來描述資料的組織及語意，並交由程式師實作到系統內，這對經常改變的健保法規環境要建立出一個共有法規服務是很困難的，本文便試著利用語意網模型來加強資訊的共享。

## 2.3 系統更新藥品檢核規則流程現況

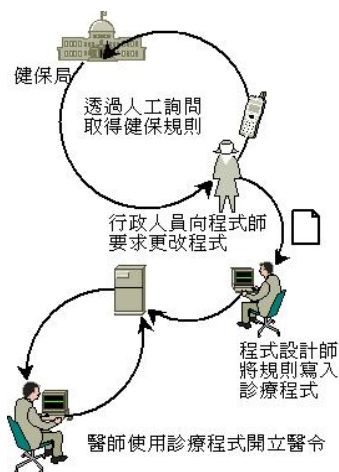


圖 1.系統更新藥品檢核規則流程現況

圖 1 描述了目前醫院系統健保局藥品規則異動時，系統更新藥品檢核規則的流程。當健保局藥品規則異動時，會以公文傳送到各相關單位，行政人員接獲公文後，依情形歸納出規則後，在會議提出，並請醫院資訊室修改，程式師修改院內程式，

再佈署上線，此時新規則才算落實到某間醫院的資訊系統中，在此流程下，規則異動時，同樣的檢核程式，卻必須在每個醫院的系統實作，非常浪費資源，故本文目標為建構一個系統，其能將上述的藥品規則更新的人工過程自動化。

## 3.系統設計

### 3.1.自動化工作流程

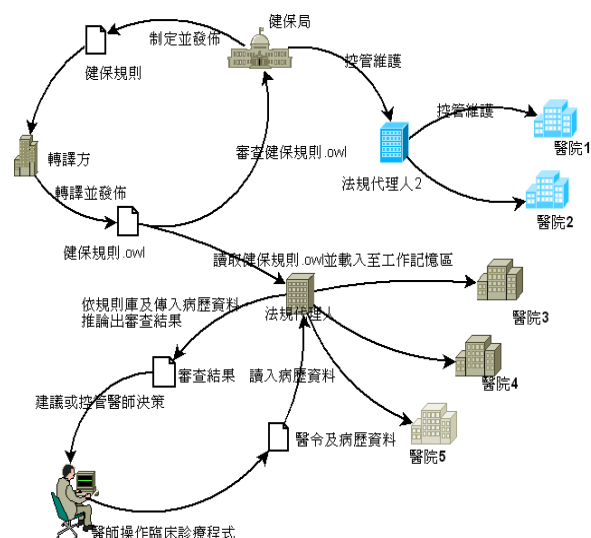


圖 2.藥品法規自動同步資料流程圖

圖 2 是本文提出的藥品法規自動同步系統的資料流程圖，如圖系統分成四個方面，分別是健保局、轉譯方、法規代理人以及醫院，每一方面執行特定獨立的功能，並以公眾網路連結起來。由於功能分解並委任到不同獨立元件的機制，可減少醫院實作重覆的法規控管程式碼，以及健保局控管的幅度，這四方面的功能描述如下：

#### (1) 健保局

負責制定藥品給付規定、審核轉譯方所轉譯的機器法規及法規代理人的推理引擎是否符合於其制定的法規意義。機器法規為一串電子資料，能有效率由機器直譯並執行自然語言法規所描述的邏輯。本系統使用的機器法規其格式為 W3C 所制定的 OWL。

#### (2) 轉譯方

A、將自然語言法規轉譯成法規的 OWL 文件。

B、與局方人員審核其轉譯機器法規其合法性，以維護病患權益。

C、負責維護機器法規伺服器能持續在 Internet 運作，以供法規代理人能夠取用機器法規。

### (3) 法規代理人

A、定期至轉譯方更新機器法規檔案。

B、實作判別用藥行為是否合法功能的推理引擎。

C、定義藥品法規自動同步系統協定(MRAS 協定)，讓醫院的原生應用程式系統只要遵行此協定，便能取得法規代理服務。上述的原生應用程式系統是指醫院內部使用的醫療資訊系統。

### (4) 醫院

指醫院本身，受健保法規所制約，若醫院想加入此系統必須實作 MRAS 協定，以取得法規代理服務。並在其原生 HIS 系統下，依法規服務所傳回的結果來執行健保法規管制項目。

## 3.1.1 系統特色

與傳統的流程相較，本系統將法規更新及檢核服務功能自醫院資訊系統分離出來，獨立成為法規代理人，並使醫院的臨床醫令系統其檢核藥物規則的功能能委由其代理，因此具有下列特色：

A、加入系統的醫院可以不用負責法規庫的更新及直譯功能的更新，同時也可保證醫院系統能隨時取得到最新的健保規則。

B、對健保局或相關管理單位而言，其只要確保法規代理人的規則庫及推理引擎是符合健保規則，即能確信委任此代理人服務的醫事機構的用藥檢核系統是合乎現行法令的，藉由第三方的法規代理人，健保局能大幅度減少管控範圍，使得健保局可以大幅降低管理成本。即大幅降低健保法規異動的系統更新成本。

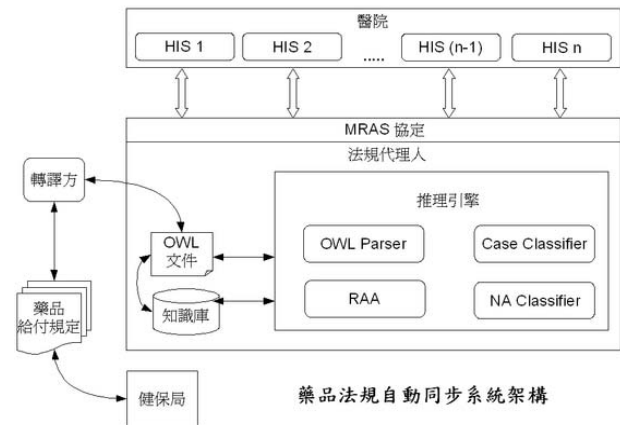


圖 3.藥品法規自動同步系統架構

## 3.2 系統功能

依據圖 2，本文設計藥品法規自動同步系統，MRAS，其架構及模組如圖 3，包含 OWL、知識庫、推理引擎、WARS 協定等 4 部份，說明如下：

### (1) OWL

健保法規的機器儲存形式，能以檔案形式儲存，故可以容易進行重製、交換及保存。另外此機器法規要能輕易的由程式直譯出來，並能表達所有健保藥品法規的語意。

### (2) 知識庫

知識庫是存放健保規則的推理事實，能直接由法規推理引擎讀入以檢核用藥合法性，一般是先由 OWL Parser，解析 OWL 載入記憶體後，再利用 OWL 推理引擎，將模型內的隱函規則一併推理出來，並存放在知識庫中，以加速法規推理引擎的效率。

### (3) 推理引擎

法規代理人本質是健保藥品法規的專家系統，依一般專家系統的架構，法規代理人至少需具備兩類元件，分別是知識庫以及推理機[9]以下則為設計的 4 個模組：

A、OWL Parser 模組解析 OWL 載入記憶體，作為推理引擎的知識庫。

B、OWL 推理引擎模組能依載入記憶體的 OWL 模型，將其模型內的隱函規則一併推理出來，並存放在知識庫中。隱函規則指的是未明確

出現在原本的 OWL 模型中,但可藉由 OWL 模型所定義的推理法則所推論出來的規則事實。

C、 RAA 模組能依據 HIS 經由法規服務 API 所傳入的用藥計算出判別用藥合法性所需要的醫療狀態資訊。

D、 Case Classifier 模組能依據 HIS 所傳入的用藥及醫療狀態作分類,若歸屬類別屬於合法的醫療狀態類別,則判別用藥合法,反之為違法用藥。

E、 NA Classifier 模組能修正原本 OWL 模型不支援的數值區間類別的歸屬問題,來支援 Case Classifier 的分類。

#### (4) MRAS 協定

讓醫院之原生應用程式系統能以此 API 來取得其法規代理服務,並傳入藥物檢核推理所必要的資訊,本文會在節 3.2 詳細說明。

相關設計說明如下：

### 3.3.MRAS 協定設計

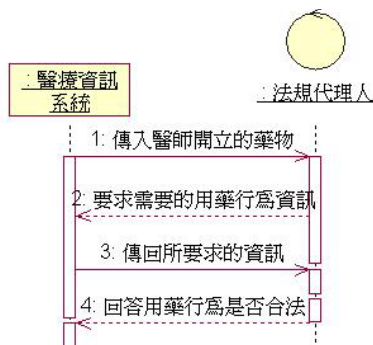


圖 4.MRAS 協定循序圖

由於不同的用藥所要求的病歷資訊大不相同,若是每次都把整個藥品給付規定所要求的病歷資訊都上傳至法規代理人,則會造成太多冗餘的資訊傳遞,因此,由圖 4 可知 MRAS 協定設計成執行二次詢問,第一次詢問此用藥需要那些病歷資訊,由此系統只要上傳必要的資訊給法規代理人便可完成第二次關於用藥合法性的詢問,這可大幅減少許多不必要的流量。依此 MRAS 協定定義了五個訊息來實現上述的二次詢問,說明如下：

(1) USEMED：醫令系統告知法規代理人指明那個用藥需要判定合法性。

(2) REQ\_PROPS：法規代理人指明判定傳來用藥合法性所需要的醫療狀態。

(3) MED\_STATUS：醫令系統用來傳入個案醫療狀態,之後法規代理人便據以完成用藥合法性的判斷。

(4) LEGALITY：法規代理人用來回覆用藥是否合法。

(5) INVALID：雙方用來指示訊息錯誤。

### 3.4.健保法規之形式化知識本體設計

在人工智慧領域中,建構本體(Ontology)的目的是知識分享 (Knowledge sharing),而形式化的目的是資訊能讓電腦有效處理,本節提出以 OWL 建構健保法規之形式化知識本體的方法,以期能將健保法規能簡單地讓電腦處理、儲存、交換來自動化系統健保法規的更新。

本系統所要建構的知識本體類型是領域式知識本體,所有的概念只存在某個專業知識領域,在本文則為健保藥品給付規則。由於系統要解答的問題是「是否開立這顆藥給此病患會違反健保規則?」,其基本的解題情境如下:醫師將病患的狀態及處方輸入系統,系統經運算後能回答醫師此處方是否違反健保規則,是故醫療規則的本質是根據醫療狀態的限制,來決定目前的「醫療狀態」是否違法。

#### (1) 醫療狀態

「醫療狀態」就是醫師執行醫療行為所有與法規有關的事實,由圖 8 也可以看出「醫療狀態」可再區分為病患狀態以及醫令狀態,醫令狀態包含用藥、用藥量、開藥醫師等等資訊。病患狀態又可細分為性別、年齡、身高、體重、疾病狀態。基本上「醫療狀態」應包含所有醫療行為應紀錄的資訊,這些資訊有法律、研究及會計等用途。

#### (2)「醫療狀態限制」描述健保法規

「全民健康保險藥品給付規定」是健保局制定的一組法規,主要利用申報給付與否來控制醫事服務機構用藥行為 [7],所以為對用藥行為的限制,而用藥行為也是一種「醫療狀態」,是故法規是一

種「醫療狀態限制」，系統可以使用「醫療狀態限制」來描述健保法規。

「醫療狀態限制」本質上是對「醫療狀態」的分類，因為每個「醫療狀態限制」定義一個允許的「醫療狀態」集合。是故審查運算便可形式化為輸入的「醫療狀態個案」是否屬於描述法規的「醫療狀態限制」，若醫師傳進來的「醫療狀態」不屬於法規的分類範圍內，則視為不符合規定，其邏輯如圖 5 所示。

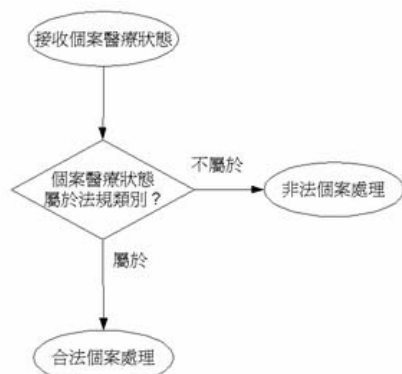


圖 5.個案分類運算

### (3) 使用屬性值限制來定義「醫療狀態限制」

由於法規可由「醫療狀態限制」來表示，而「醫療狀態限制」本質上為「醫療狀態」的類別，是故採用的形式知識本體語言必須能夠支援定義類別的語法，且其模型能輕易的執行某執行個體是否屬於某定義類別的運算，以期能利用此形式知識本體語言來模擬法規審查運算。

#### 第 1 章神經系統藥物

##### 1.1.疼痛解除劑

##### 1.1.1.非類固醇抗發炎劑外用製劑給付規定

定:(88/09/01、92/2/1)

1.外用非類固醇抗發炎軟膏，限不適合口服非類固醇抗發炎製劑之軟組織風濕症或關節炎病患使用，每月至多以處方一支(每支最大包裝量不得大於 40gm)為限。

#### 圖 6.神經系統藥物的藥品給付規定

圖 6 為神經系統藥物的藥品給付規定，假設我們要列舉所有合法的「醫療狀態」來定義上述法令的類別，幾乎是不可能的，所以我們應該使用屬性值限制來定義類別，例如圖 7 的虛擬碼，便是使用屬性值限制法來定義上述的法規類別，而且可以很輕易的就可以轉成 OWL 文件。

```
class "規定 1.1.1.1" subclass of 「醫療狀態限制」
"用藥"值="外用非類固醇抗發炎軟膏"
"藥品用法"值<>"口服"
"每次用量"值<"一支"
"一支"的"每次用量"值<"40gm"
"使用週期"值="一個月"
"疾病"值 in {"軟組織風濕症","關節炎"}
end class
```

圖 7.以屬性值限制來定義神經系統藥物規定

### (5) 健保規則的本體類別階層設計

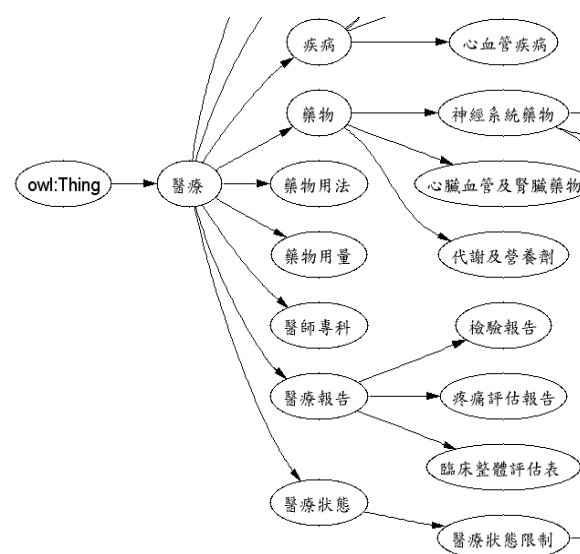


圖 8.健保規則代理人類別階層圖

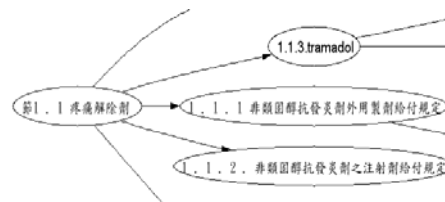


圖 9.疼痛解除劑的類別階層圖

由圖 8 可看出，第一層為「醫療」的子類別，來表示此類別以下的概念類別是屬於醫療業的領域知識。於是像是「疾病」、「藥物」、「藥物用法」、「藥物用量」、「醫師專科」、「醫療報告」、「醫療狀態」等在法規內容出現的概念，都繼承「醫療」來表示這些概念專屬於醫療領域。

「醫療狀態」指每個人就醫的狀態，定義許多屬性，像是用藥、用藥量、開藥醫師、病患性別、病患年齡、病患身高、病患體重，來精確描述醫事活動。前文提到「醫療狀態限制」本質上是對醫療狀態的分類，所以將「醫療狀態限制」定義為「醫



療狀態」的子類別。因為「醫療狀態限制」可用來描述「健保法規」，故將「全民健康保險藥品給付規定」作為「醫療狀態限制」的子類別。而「全民健康保險藥品給付規定」以下類別階層，則大致上依照規定本文的章節階層而定，如圖 9 所示的類別階層就和健保規則本文章節階層類似。

### 3.5.推理機設計

(1) RAA 模組：為了能實作 MRAS 協定，推理機必需能夠根據「用藥」屬性，來推論出使用者需要提供的病歷資訊。RAA 模組的運算邏輯如圖 10 的 SPARQL 碼[11]所描述，此模組定義了 medres，代表用藥與使用者所傳入的藥物相同的所有「醫療狀態限制」類別；再由 medres 定義其所有子類別 medcls，則 medcls 表示所有限制此類藥物的類別，再求出 medcls 所繼承的所有屬性限制類別 restrict，由 restrict 類別的 onProperty 屬性值，表示限制這個用藥的所有屬性限制。上述架構可以發現在 OWL 模型中要組合不同的限制，是由多重繼承關係來實作。

```
usedmed= 醫療狀態.用藥
SELECT?requireProp
WHERE{
?medresowl:allValuesFrom usedmed.
?medclsrdf:subClassOf?medres.
?medclsrdf:subClassOf?restrict.
?restrictrdf:typeowl:Restriction.
?restrictowl:onProperty?prop.
}
```

圖 10.RAA 模組運算邏輯

(2) CaseClassifier 模組：CaseClassifier 模組其運算邏輯如同圖 5 所示，其能夠剖析 OWL 文件，並依照 OWL 文件所描述的本體模型，對傳入的「醫療狀態」個案作分類運算來模擬法規審查運算。

(3) NA Classifier 模組：OWL 沒有定義數值區間類別的建構子，例如「病患年齡」值小於「70」的「醫療狀態限制」類別，便無法直接利用 OWL 來定義。因此要進行數值區間類別的分類，必須擴充 OWL 模型的語義，本系統由 NA Classifier 模組實現擴充語義。

NA Classifier 模組其運算邏輯如同圖 11 所示，能從所有的「醫療狀態限制類別」中，對每個「醫

療狀態個案」的「數值區間屬性」，找出其「對應之數值區間屬性限制值」；再來判斷「個案數值屬性值」是否在「對應之數值區間屬性限制」值之間，若是，則此「醫療狀態個案」是屬於某個「醫療狀態限制」，也意指此「醫療狀態個案」是合法的。

```
boolean result=true
for 「個案數值區間屬性值」 in 「醫療狀態個案」.
「屬性集合」 do
find 「醫療狀態限制類別」. 「對應之數值區間屬性限制值」
if 「個案數值區間屬性值」 between 「對應之數值區間屬性限制」 then
result and true
else
result and false
end
return result
```

圖 11.數值區間屬性限制運算邏輯

(4) OWLParser：此模組解析 OWL 文件載入記憶體，並推理出 OWL 文件中所有的隱含規則，作為推理引擎的知識庫。

## 4.雛型系統實作

本文依系統設計建構簡易的雛型來實現上述的法規自動更新系統，本雛型有兩個測試，需求醫療狀態測試，及判斷用藥合法性測試。

請輸入用藥：

判斷用藥非類固醇抗發炎劑之注射劑合法性需要輸入以下的醫療狀態

- 病患能接受的藥物用法
- 患有疾病
- 醫令用藥
- 連續使用天數

圖 12.求出合法性判斷所需的醫療狀態

需求醫療狀態測試會要求輸入一顆藥，系統則會傳回判斷此藥物合法性運算所需要輸入的屬性值。圖 12 中顯示使用者輸入欲檢測的藥物「非類固醇抗發炎劑之注射劑」，按下「查詢」後，系統則會回傳此藥物限定的四個屬性，分別為病患能接受的藥物用法、患有疾病、醫令用藥及連續使用天數。

病患狀態	
病歷號：	0001 姓名：李德旺
疾病：	中風 患病月數：
檢驗報告：	

用藥資訊	
用藥：	非類固醇抗發炎劑之注射 用法：注射
用藥片數：	用藥天數：4
併用藥：	

用藥合法！

圖 13.通過合法性判斷的醫療狀態

病患狀態	
病歷號：	0001 姓名：李德旺
疾病：	中風 患病月數：
檢驗報告：	

用藥資訊	
用藥：	非類固醇抗發炎劑之注射 用法：注射
用藥片數：	用藥天數：11
併用藥：	

用藥不合法！

圖 14.藥物使用日期不合法的醫療狀態

病患狀態	
病歷號：	0001 姓名：李德旺
疾病：	急性上呼吸道感染 患病月數：
檢驗報告：	

用藥資訊	
用藥：	非類固醇抗發炎劑之注射 用法：注射
用藥片數：	用藥天數：4
併用藥：	

用藥不合法！

圖 15.病患疾病不合法的醫療狀態

病患狀態	
病歷號：	0001 姓名：李德旺
疾病：	患病月數：
檢驗報告：	

用藥資訊	
用藥：	非類固醇抗發炎劑之注射 用法：
用藥片數：	用藥天數：11
併用藥：	

缺少以下屬性，請補輸入！

- 病患能接受的藥物用法
- 患有疾病

圖 16.缺乏需求屬性的不合法醫療狀態

圖 13 顯示判斷用藥合法性測試的畫面，其有一組表單讓使用者輸入必要屬性值以建立一個醫療狀態個案，當表單輸入完成後，系統便將表單上的醫療狀態傳入至 MRAS 來回答用藥是否合法。圖 13 就是一個合法的醫療狀態個案。而圖 14 是一個

與圖 13 相同藥物但不合法的醫療個案，因為其「用藥天數」違反了「使用本類藥品，每次不可連續超過五天」的規定，圖 15 主要是違反了「不可患有急性上呼吸道感染類別的疾病」，圖 16 是缺乏了「用藥天數」、及「疾病」兩個需求屬性值，而無法作判斷，同時系統會提醒使用者補入漏掉的需求屬性值，上述的測試可以看出 OWL 可以處理大部份的法規邏輯。

## 5.結論

目前大部份的醫療資訊系統其藥品檢核系統的設計，主要是使用關連式資料庫，並以表格記載不同法規所需的條件參數，由於條件資料表示及處理邏輯的不一致，造成必須為不同的法規撰寫不同的推理引擎。也要撰寫許多的維護程式去維護這些參數表格，來達到系統所需的彈性。

本研究將違反法規的運算，簡化成醫師傳入的醫療狀態個案，是否屬於合法的醫療狀態類別，是故推理引擎只要實作出個案分類運算，之後法規更新便只要更新類別定義資料即可達成。本系統採用 OWL 模型，實作 OWL 模型中的個案分類運算，當法規異動時，只要更動表示類別關係的 OWL 檔就好，由於法規異動被封裝到 OWL 檔，是故法規異動完全不會更動到推理引擎。又由於 OWL 或等效的邏輯模型，已隱含了大部份藥品法規所會用到的邏輯於其模型的類別定義語意中，是故可以上述模型來表達大部份的藥品法規。

當資料與推理引擎完全分開，則可以輕易的將每項工作切給不同實體來負責，像是健保局僅負責制定給付規定及審核轉譯的法規 OWL 文件其語義是否合於其制定的法規。而法規代理人則提供推理引擎及法規更新的服務給各醫院，以往這些異動工作往往由醫院負責，造成各醫院資訊人員極大的負擔；對健保局而言，法規異動則要很長的推廣期，複雜的審核系統，才能確信法規已實施到各個醫院中。

在健保醫療資源逐漸吃緊的狀態下，若能讓健保法規自動發佈並實現在各醫院的法規檢核系統，除



了讓健保局能確實有效地執行其法規政策以妥善管理醫療資源，也讓醫院也能減少其更動資訊系統上的成本。這對落實藥品法規以杜絕醫療資源浪費，以及節省醫院資訊系統異動成本上，都有很大的幫助，進而能推動電子法規控管至更大範圍的電子化政府的目標。

## 參考文獻

- [1]林建良.以知識本體提供代理人建構共通之協商環境－以生產排程協商為例.中原大學資訊研究所碩士論文.2005.
- [2]尤進泰.健保 IC 卡內存醫令與醫師處方用藥安全檢核系統設計實作.成功大學工程科學研究所碩士論文.2004.
- [3]賴昆汕.藥品交互作用資料庫應用系統設計與實作.臺灣大學資訊工程研究所碩士論文.2003.
- [4]李彥良.Web-based 診間醫令之開發與建置.臺北醫學大學醫學資訊研究所碩士論文.2001.
- [5]湯進聖.醫院醫囑藥物交互作用提示系統.國立陽明大學公共衛生研究所碩士論文.2002.
- [6]簡文山,李友專,唐大鈿,且胡俊弘."建立臺灣醫療資訊交換中心之藍圖".醫療資訊雜誌.54-66.中華民國醫療資訊學會.
- [7]中央健康保險局.全民健康保險藥品給付規定.
- [8] Alfred V. Aho, Ravi Sethi, and Jeffrey D Ullman. Compilers, principles, techniques, and tools. Addison-Wesley Pub. Co. 1986.
- [9] Peter Jackson. Introduction to Expert Systems. Addison-Wesley. 1990.
- [10] John F. Sowa. Knowledge Representation. Logical, Philosophical, and Computational Foundations . Course Technology. Aug 16 1999.
- [11] Eric Prud'hommeaux and Andy Seaborne. SPARQL Query Language for RDF. <http://www.w3.org/TR/rdf-sparql-query/> .