

以非同步訊息佇列技術建構之隨選資訊遞送系統

Implementation of an On-Demand Notification System using Asynchronous Message Queue

黃宗祺

Tsung-Chi Huang

rick@ilab.csie.ntut.edu.tw

陳英一

Ing-Yi Chen

ichen@csie.ntut.edu.tw

黃致豪

Chih-Hao Huang

nelson@ilab.csie.ntut.edu.tw

國立台北科技大學資訊工程研究所

National Taipei University of Technology

Department of Computer Science and Information Engineering

摘要

本論文主要研究主題乃是以非同步訊息佇列 (Asynchronous Message Queue) 技術以及通知標記語言 (Notification ML) 為基礎, 整合多種有線及無線訊息遞送通道, 發展一隨選資訊遞送系統 (On-Demand Notification System)。此系統能將重要訊息以最適合的通道, 傳送至使用者手上。具體的研究成果包括 (1) 資訊遞送系統 (Notification System) 平台的開發建置, (2) 透過通知標記語言 (Notification ML) 轉換訊息格式, 以達成通道訊息的通用性 (3) 多種有線及無線訊息遞送閘道器 (Notification Gateway) 的整合應用。以此系統為基礎, 本研究期能建立一通用的訊息平台, 提供一標準的訊息通道介面, 以整合更多資訊遞送方式。在此架構下所開發的庫存警示系統, 目前已應用於物流系統中, 能在貨品低於安全庫存量時, 主動以電子郵件、簡訊、無線應用程式通訊協定訊息及 IBM Lotus Sametime 即時傳訊服務通知管理者。

關鍵詞: 非同步訊息佇列、通知標記語言、資訊遞送系統

Abstract

The goal of this research is using Asynchronous Message Queue technology and Notification Markup Language to implement an On-Demand Notification System that integrates variety of message delivery channels. Under this architecture, a stock warning subscription system has been implemented and applied to a delivery system. When the amount of product is

less than the limit that manager has assigned, the On-Demand Notification System will notify manager through email, short message and any available message channel.

Keywords: Asynchronous Message Queue, Notification Markup Language, Notification System

一、背景

在網際網路與行動通訊越來越普及的今日, 資訊的更新與流動更加快速。如何能在眾多的訊息中, 掌握最新的資訊, 並在第一時間取得此訊息, 便成為一件重要的事。

以物流產業為例, 當訂單陸續大量湧現, 會造成貨品庫存量迅速下降。此時, 系統如果能在最短的時間內, 以最適合的訊息通道主動通知管理者, 便能及時協助進行補貨作業。

隨著行動電話與個人掌上助理 (PDA) 等行動設備的日益普及, 使得工作地點不再侷限在辦公室。這些行動裝置所帶來的便利性, 讓個人的訊息聯絡管道, 除了電子郵件系統之外, 更增加了如簡訊服務 (Short Message Service)、多媒體簡訊服務 (Multi-Media Service)、無線應用程式通訊協定訊息 (WAP Push Message) 以及各種即時傳訊服務 (如: Microsoft MSN、IBM Lotus Sametime 等) 等新方式。如何能夠利用這些新型態的訊息遞送通道, 讓訂閱者隨時隨地收取即時訊息, 為本研究的重點。

第三代無線通訊網路的建設, 使得通訊頻寬大幅的提升, 訊息內容以及傳遞方式也越來越多樣化。目前的訊息系統, 勢必也將整合更

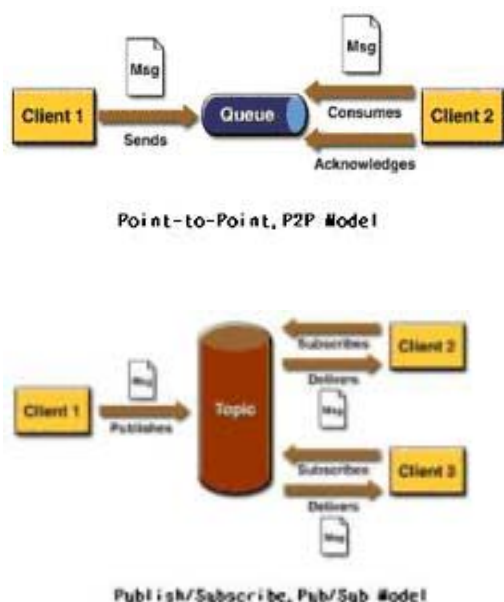
多的遞送通道。本研究旨在建立一通用的訊息遞送平台，提供共通的遞送通道介面，以整合目前以及未來的傳訊方式。藉著此平台的通用特性，使本研究能應用於不同的行業中，以最低的發開成本，達成訊息遞送的目的。

本研究使用非同步訊息佇列 (Asynchronous Message Queue) 技術為核心，並以標準的 Java 訊息服務應用程式介面 (Java Message Services Application Program Interface) [1~2]，開發出具備訂閱功能的訊息遞送應用程式介面 (Notification Application Program Interface)。在此基礎下，本研究結合多種有線及無線的訊息遞送通道與閘道器，發展出一套隨選訊息遞送系統 (On-Demand Notification System) [3]。

二、研究方法

2.1 系統架構

本系統的核心使用 J2EE (Java 2, Enterprise Edition) 標準 Java 訊息服務 (Java Message Services, JMS) 所開發而成。Java 訊息服務規範了兩種傳訊模式，分為點對點 (Point-to-Point, P2P) 及出版/訂閱 (Publish/Subscribe, Pub/Sub) [1~2] 兩種方式，本系統實作了規範中的出版/訂閱的訊息模式。

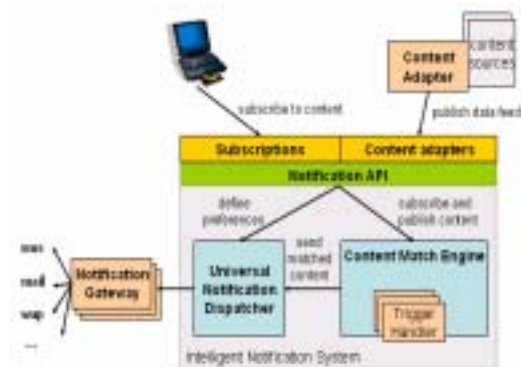


圖一 點對點及出版/訂閱模式

對出版/訂閱的訊息模式而言，是由出版者 (Publish)、主題 (Topic) 與訂閱者 (Subscriber) 所組成。出版者負責將訊息出版到主題，此訊息並不會指定特定的接收者，

而是先儲存在主題的佇列中，再由主題比對所有訂閱者訂閱時所提供的條件值。當符合訂閱條件時，此訊息才會送交給訂閱者。

在出版/訂閱模式的架構之下，本研究以非同步訊息佇列技術為核心，架構出一套隨選資訊遞送系統 (On-Demand Notification System)。下圖為此系統之架構圖。



圖二 資訊遞送系統架構圖

為了達成隨選訊息通知的特性，除了負責接受訂閱資訊以及比對資料的元件外，還需要具備與訊息伺服器連結的通道模組，讓訊息可藉由各種管道送出。在此架構之下，本系統規劃出以下四項主要元件。

A. 內容比對引擎 (Content Match Engine)

內容比對引擎的主體為非同步訊息佇列 (Asynchronous Message Queue)，主要負責紀錄使用者的訂閱資訊，以及接收內容配接器 (Content Adapter) 所擷取的資訊。當內容比對引擎接收到新的資訊時，會將所有使用者的訂閱條件比對一次。符合比對條件時，內容比對引擎會呼叫觸發處理器 (Trigger Handler) 元件。

每個觸發處理器 (Trigger Handler) 會對應個別獨立的主題資訊。觸發處理器負責組成發送給通用訊息配送器 (Universal Notification Dispatcher) 的訊息內容。此訊息內容可由內容配接器所提供的資訊重新組合而成，且必須符合通知標記語言 (Notification ML) 的規格。通知標記語言是一種用來將通知訊息格式化的標記語言，為 XML 的一種格式，下表說明了主要使用的標示。

表一 通知標記語言

標示	說明
<message></message>	標示為通知標記區塊
<to></to>	訊息收件人
<from></from>	訊息傳送者
<subject></subject>	訊息的主題
<text></text>	文字訊息主體

B. 訊息遞送應用程式介面 (Notification API)

訊息遞送應用程式介面以同步訊息佇列技術為核心，並由標準的 Java 訊息服務應用程式介面 (Java Message Services Application Program Interface) 所延伸出來，扮演著訂閱介面以及內容配接器與內容比對引擎的溝通介面。此應用程式介面主要分為訂閱 (subscriptions) 與內容配接器 (content adapters) 兩大部分。前者所提供的程式庫能讓開發者設計出特定主題的訂閱介面，將訂閱者的訂閱資訊註冊在內容比對引擎中；後者所提供的程式庫讓開發者發展主題資訊來源的配接器，將資訊內容輸入內容比對引擎中，以作為資訊比對的依據。

C. 通用訊息配送器 (Universal Notification Dispatcher)

當使用者進行訂閱動作時，一方面會將訊息訂閱條件註冊在內容比對引擎中，另一方面會將使用者的訊息通道喜好設定 (Channel Preferences) 登錄在通用訊息配送器中。訊息通道喜好設定 (Channel Preferences) 主要的功能在負責定義針對不同的訂閱主題，使用者可以選擇各種不同的訊息接收管道。

通用訊息配送器在此系統中扮演著內容比對引擎與訊息遞送閘道器之間的溝通介面。通用訊息配送器主要接收來自觸發處理器 (Trigger Handler) 的通知標記語言 (Notification ML) 訊息，並根據上述的訊息通道喜好設定，將通知標記語言訊息轉送至不同的訊息遞送閘道器 (Notification Gateway)。

D. 訊息遞送閘道器 (Notification Gateway)

訊息遞送閘道器 (Notification Gateway) 則是此系統與各個訊息伺服器的溝通介面。本研究中實作了電子郵件 (SMTP) 以及簡訊服務 (Short Message Service) 兩種通道。電子郵

件訊息閘道器會與郵件伺服器連結，簡訊服務閘道器則會與電訊公司的訊息中心連線。當前方的通用訊息配送器發出通知標記語言訊息時，訊息遞送閘道器會將此通知標記語言訊息進行轉碼動作，轉換成符合特定閘道的訊息規格。



圖三 訊息遞送閘道器與轉碼程式

每一個閘道器都有一個專屬的轉碼程式，用來製作訊息遞送通道專屬的訊息格式。轉碼程式也可配合 XSL 樣式表 [4~5] 來執行轉碼動作，可將通知標記語言訊息依據 XSL 樣式表所規範的格式，轉換成目標訊息伺服器所需的規格。

本研究中所開發的兩種閘道器，在簡訊服務通道的轉碼動作方面，由於電訊公司的訊息中心所接受的訊息格式，為單純的純文字文件，所以採用的是基本的字串擷取方式，只將通知標記語言訊息 <text></text> 區塊內的文字取出即可。

另一方面，電子郵件通道為了符合標準的訊息規範，所以採用配合 XSL 樣式表的方式，以簡化轉碼的複雜度。下圖為電子郵件 XSL 樣式表的基礎範本。

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform" >
  <xsl:output omit-xml-declarations="yes" method="text"/>
  <xsl:template match="message">
    <xsl:value-of select="text"/>
    <xsl:for-each select="url">
      <xsl:text>
        </xsl:text><xsl:value-of select="."/>
      </xsl:for-each>
      <xsl:text>
        convert to mail template
      </xsl:text>
    </xsl:template>
  </xsl:stylesheet>
```

圖四 電子郵件 XSL 轉換樣式表

上述的四項元件，組成了資訊遞送系統 (Notification System) 的發展平台，本研究已依據此平台實作出為物流業所設計的庫存警示系統。

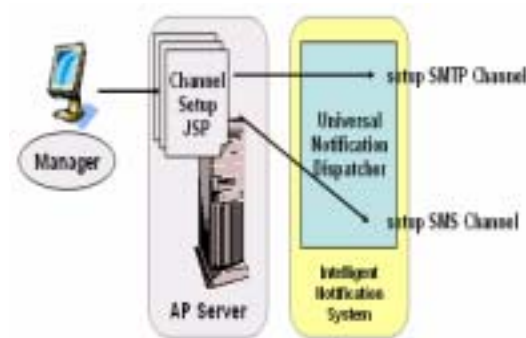
2.2 系統實作

本研究運用以非同步訊息佇列技術為核心的資訊遞送平台，實作出一套物流業的庫存警示系統。

完整的訊息訂閱過程包含三部分：

- (1) 設定個人通道資訊
- (2) 註冊訂閱資訊
- (3) 觸發訂閱條件值與發送訊息

訂閱者首先必須透過應用程式伺服器 (Application Server) 所提供的 Channel Setup 頁面，來設定個人專屬的 SMTP 與 SMS 通道。利用 SMTP 通道設定畫面，設定訊息發送時的接收者電子郵件地址，SMS 道設定畫面則是設定接收者的手機號碼。



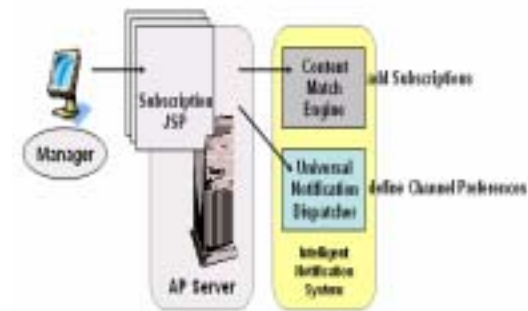
圖五 設定訊息遞送通道

接下來訂閱者透過 Subscription 頁面，向內容比對引擎註冊庫存訂閱的資訊。註冊的主題 (Topic) 名稱為 "Stock" (庫存)。在此主題之下包含兩個子項目："stockName" 與 "stockQuantity"，分別為貨品名稱與目前庫存量。註冊的觸發條件值 (Trigger Filter) 則類似以下的範例：

Filter = (stockName = "productA") AND
(stockQuantity < 30)

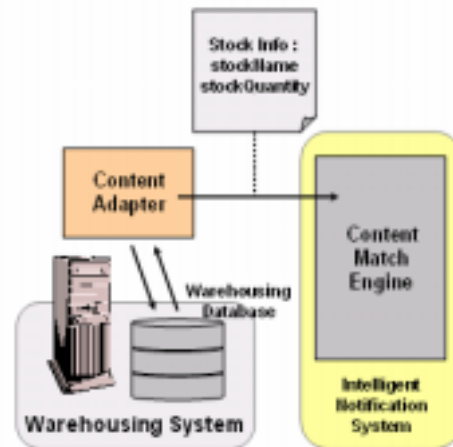
觸發條件值敘述句須符合 SQL 92 規範中 "WHERE" 的條件句型語法。上述的條件句型即代表 "當 productA 庫存量低於 30 個時，發送訊息給訂閱者"。訂閱者向內容比對引擎

註冊訂閱資訊的同時，也必須同時向通用訊息配送器 (Universal Notification Dispatcher) 登錄：當條件觸發時，該選用何種遞送通道送出訊息。



圖六 註冊訂閱資訊

內容配接器 (Content Adapter) 會定時向倉儲系統 (Warehousing System) 查詢目前的庫存資訊 (stockName、stockQuantity)，並輸入內容比對引擎中。每輸入一筆資料，內容比對引擎會比對所有註冊的訂閱資訊。當訂閱條件值符合輸入的資料時，觸發處理器 (Trigger Handler) 將會被內容比對引擎所呼叫，並利用內容配接器所輸入的資訊，組合成通知標記語言 (Notification ML) 訊息，向通用訊息配送器發出請求。



圖七 內容配接器 (Content Adapter)

通用訊息配送器 (Universal Notification Dispatcher) 收到觸發處理器發出的請求後，會依據訂閱者所登錄的訊息通道喜好設定 (Channel Preferences)，向指定的訊息遞送閘道器 (Notification Gateway) 傳送通知標記語言訊息內容。

訊息遞送閘道器將標記語言訊息轉碼過後，會透過不同的訊息伺服器，向訂閱者發出庫存不足的警示訊息。

三、研究成果與結論

本研究所發展的隨選訊息遞送系統，架構在非同步訊息佇列技術上，整合了多種有線及無線訊息通道，解決了即時資訊的取得問題。本研究目前業已完成下列成果：

(1) 資訊遞送系統 (Notification System) 平台的開發建置。

(2) 透過標準通知標記語言 (Notification ML) 以及轉碼程式(Transcoder) 達成訊息轉換的目的。

(3) 多種有線及無線訊息遞送閘道器 (Notification Gateway) 的整合應用，除了電子郵件通道以及簡訊服務通道外，並實作出無線應用程式通訊協定訊息 (WAP Push Message) 通道 (須搭配 WAP Gateway) 與即時傳訊服務 (IBM Lotus Sametime) 通道。

在此架構下所開發的庫存警示系統，目前已經應用在實際物流系統上。此系統能在貨品庫存量低於設定值時，以電子郵件、簡訊、無線應用程式通訊協定訊息以及 IBM Lotus Sametime 即時傳訊服務通知管理者進行補貨動作，使貨品能維持在安全存量之上。

圖八與圖九為訂閱者設定個人專屬的訊息通道的設定畫面，在此畫面下可設定個人的手機號碼與電子郵件地址。圖十為庫存警示訂閱介面，可在畫面中輸入庫存警示的條件值並選擇偏好的訊息遞送通道。



圖八、九 設定訊息通道



圖十 訊息訂閱介面

四、謝誌

本文蒙國科會 (NSC 91-2213-E-033-028) 專題研究計畫補助，特此致謝。

五、參考文獻

- [1] P. Th. Eugster, R. Guerraoui, and J. Sventek. "Distributed asynchronous collections: Abstractions for publish/subscribe interaction." In 14th European Conference on Object Oriented Programming (ECOOP 2001), pages 252-276, June 2001.
- [2] P. Pietzuch and J. Bacon, Hermes "A Distributed Event-Based Middleware Architectur" on DEBS'02 with IEEE ICDSCS, 2002.
- [3] Yih-Farn Chen, Huale Huang, Rittwik Jana, Sam John, Serban Jora, Amy Reibman, Bin Wei, "Personalized Multimedia Services Using a Mobile Service Platform", Proceedings of the IEEE Wireless Communications Networking Conference, Florida, March 17-21, 2002.
- [4] Daniel Deveaux, Yves Le Traon. "XML to Manage Source Engineering in Object-Oriented". IEEE Computer, May 2002.
- [5] Tayeb Lemlouma, Nabil. "Adapted Content Delivery for Different Contexts". Nov 2001.