行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※
※　　　　　　　　　　　　　　　　　　　　　　　　※
※　　　　　　**以 XML 設計中文電子支票**　　　　　　※
※　　　　　　　　　　　　　　　　　　　　　　　　※
※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：☑個別型計畫　　□整合型計畫
計畫編號：NSC 90－2213－E－009－059－
執行期間：　90 年　08 月　01 日至　91 年　07 月　31 日

計畫主持人：　黃景彰
共同主持人：

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

執行單位：　國立交通大學資訊管理研究所

中　華　民　國　91　年　9　月　　　日

行政院國家科學委員會專題研究計畫成果報告

以 XML 設計中文電子支票

計畫編號：NSC 90－2213－E－009－059－
執行期限：90 年 8 月 1 日至 91 年 7 月 31 日
主持人：黃景彰
計畫參與人員：徐鵬雲、邱雅文（研究生）
執行機構及單位名稱：交大資管所

一、中文摘要

付款機制是發展電子商務的重要項目。現行的付款機制包括電子現金、線上信用卡付款機制、小額付款與電子支票。其中，以電子支票的發展最為緩慢，而其主因在於電子支票的發展牽涉到各國的法律及商業運作習慣不同。在本研究中，我們針對我國的法規現況及商業運作的特點設計符合 XML 語法的電子支票並提出一實作的系統架構。
關鍵詞：電子支票、可擴展標示語言、資訊安全

Abstract

Payment schemes are essential to the development of electronic commerce. The schemes developed include electronic cash, on-line payment using credit cards, micro-payment, and electronic check. Among these, the scheme of electronic check is less developed, because it involves distinctive laws and business practices in different judicial territories. In this research, the authors have designed a set of XML-based syntax for Chinese electronic checks, and also proposed operational framework for systems to be adaptive to the judicial conditions and business practices of our country.
Keywords: electronic check, XML, information security

二、緣由與目的

網際網路的興起帶動電子商務的發展。但在兩年前，由於網路經濟的泡沫化，讓許多人士開始擔心電子商務的發展就此結束與停止。然而，在歷經網路泡沫化的洗禮，電子商務的發展愈趨成熟。依據路透社所發佈的報導，全球上網人口的逐年增加，預估至 2006 年，全球電子商務將加速成長至 6 兆 5010 億美元 (資策會電子商務應用推廣中心，民 91 年 a)。此外，根據 ISM 與 Forrester Research 發公佈的企業利用電子商務進行線上採購的比率已有明顯的成長 (資策會電子商務應用推廣中心，民 91 年 b)。這些數據再再地顯示出網路商務的發展與重要性，網路銀行、網路下單、網路競標與購物、企業快速回應系統、供應鏈管理、客戶關係管理等已成為電子商務的主要應用。

在商業活動中，金融服務是不可缺少的必備條件，自然地在電子化的市場中也不例外。在電子商務的環境裡，已有多種的支付工具可應用於網際網路上的交易。其中，有類似於傳統使用現金、類似傳統使用的支票、利用網路銀行轉帳或利用信用卡付款等等的電子支付方式。根據亞太區國際網路評量機構 (亞太評量) 2001 年九月上旬針對臺灣所做的線上消費金融的使用調查中顯示，信用卡是網路消費最普遍的付款方式，約有 65% 的使用者用它作為網路交易的支付工具 (iamasia, 2001)，雖然如此，信用卡卻不見得是最安全的方式。因此，電子支票 (electronic check) 在近年來也逐漸發展，希望建立有效、安全、低風險的付款方式，並解決線上信用卡付款的不足。

目前已在運作的電子支票系統首推美國 FSTC (Financial Services Technology Consortium) 組織所發展的 eCheck 系統，此一系統是採用自訂的標示語言(Markup Language) — FSML (Financial Service Markup Language)。然而，FSML 與企業所採用的 XML(eXtensible Markup Lanauage) 並不相容，將不利於與企業內的資訊系統整合。再者，支票的使用常因國情、社會、使用習性的不同而有差異，所以 FSTC 所開發的 eCheck 系統並不適用於我國。因此，本計畫將針對我國企業與民眾的特性，發展適用於我國的電子支票系統，並期望此研究能提供網路上一個安全可靠的電子付款機制，帶動國內電子商務的發展。

三、文獻探討

本研究將設計一符合我國使用者習性的電子支票，而此一電子支票系統將以 XML 來達成資料的共享。因此，在 3.1 小節中，將先針對 FSTC 的電子支票架構進行介紹。在 3.2 小節中，則針對 XML 的相關技術做簡介。而在 3.3 小節，則將介紹 XML 簽章技術，以做為實作電子支票的背景知識。

3.1 FSTC 電子支票架構

電子支票的設計概念(圖 1)首見於 Milton M. Anderson 於 1998 年發表的”The Electronic Check Architecture” (Anderson, 1998)，文中提出 4 種不同的電子支票流程，並引以為 FSTC 開發電子支票系統的基本架構。

電子支票的運作起始於付款方接收到收款方所傳送的發票或帳單，而付款方的帳務系統在付款日來臨時，會交由系統依發票或帳單上的資訊產生一份電子支票，其上包括類似紙本支票記載的訊

息，如收款人、金額、到期日、帳號等。付款方再輸入一識別碼(PIN)來開啟智慧卡形式的電子支票簿，並利用智慧卡內的個人私密金鑰對電子支票進行簽章。在簽核電子支票後，付款方將這份含有簽章訊息的電子支票連同憑證、發票包裝起來，利用 e-mail 或 www 的型態傳送給收款方。收款方在驗證簽章與電子支票無誤後，拿掉發票資訊並記錄應收帳款。當收款方欲向銀行要求提示付款，需將自己所收到的電子支票與存款單一同進行簽章，而此一動作與付款方簽核電子支票的動作相同。待簽核完畢後，就將加簽後的電子支票及收款方的憑證一併傳送給收款方銀行。收款方銀行再依票據清算程序與付款方銀行進行清算，最終付款方銀行會通知付款方交易完成。

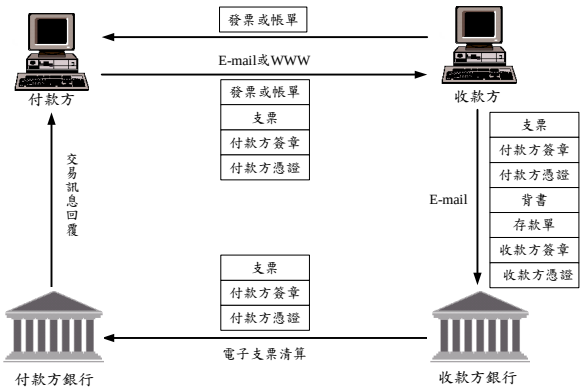

圖 1 電子支票概念

基於此概念衍生 4 種電子支票流程：電子支票基本流程(Electronic Check Basic Flow)、電子支票 Lockbox 流程(Electronic Check Lockbox Flow)、電子支票兌現與移轉流程(Electronic Check Cash and Transfer Flow)、保付電子支票流程(Certified Electronic Check Flow)。

1. 電子支票基本流程

基本流程(圖 2)是類似於傳統支票的交易模式，為一般最常見的交易模式，其運作流程為上述概念之簡化。

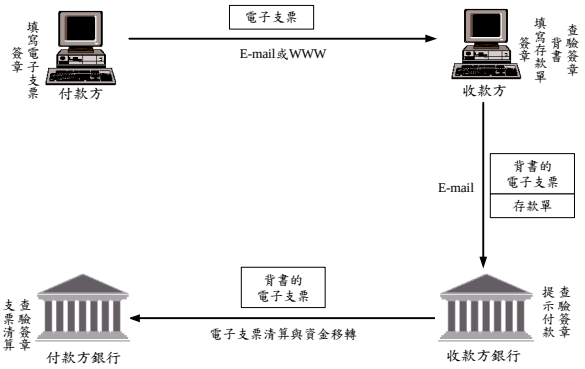

圖 2 電子支票基本流程

2. 電子支票 Lockbox 流程

此種流程(圖 3)之重點在於電子支票並不傳送到收款方，而是透過第三者稱之 Lockbox，Lockbox 代替收款方處理電子支票的簽章驗證，並將背書後的電子支票連同存款單直接傳送給收款方銀行，同時回傳發票訊息給收款方。

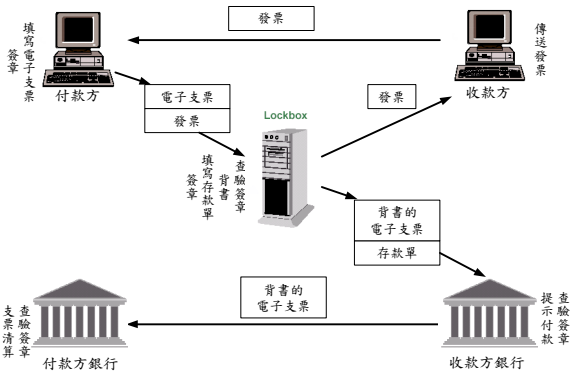

圖 3 電子支票 Lockbox 流程

Lockbox 流程的好處在於收款方不需要安裝新的電子支票等相關軟硬體，就可接收到電子支票的付款，以降低使用電子支票所需付出的成本。此外，Lockbox 的功能若能直接由收款方銀行擔任，則可直接與銀行內部的系統整合，以減少背書與提示付款的動作。

3. 電子支票兌現與移轉流程

電子支票兌現與移轉流程(圖 4)為付款方先傳遞電子支票給收款方，但收款方在背書後就將電子支票直接交付付款方以進行付款提示，雙方銀行再透過金融網路進行資金移轉。此流程適用於收款方銀行尚未採行電子支票系統或收款方未有開立電子支票帳戶的情況。

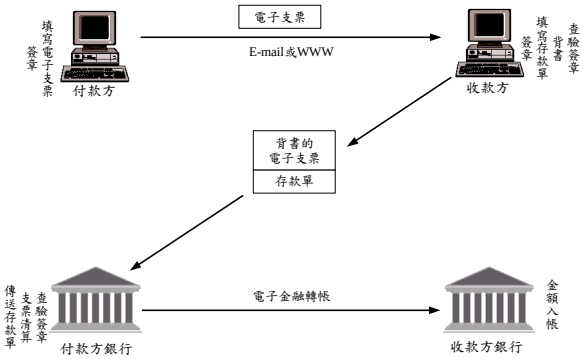

圖 4 電子支票兌現與移轉流程

4. 保付電子支票流程

在保付電子支票流程(圖 5)中，付款方必須先把電子支票傳送給自己的銀行，付款方銀行驗證其簽章並查驗付款方的帳戶狀況及是否保留足夠的支付金額，再對電子支票加上銀行簽章，以表示此電子支票無誤而且對可以付兌核保付電子支票後，可將支票傳回給付款方或直接交由收款方，以進行後續的流程。此一保付電子支票流程與我國保付支票有相同的意義與效果。

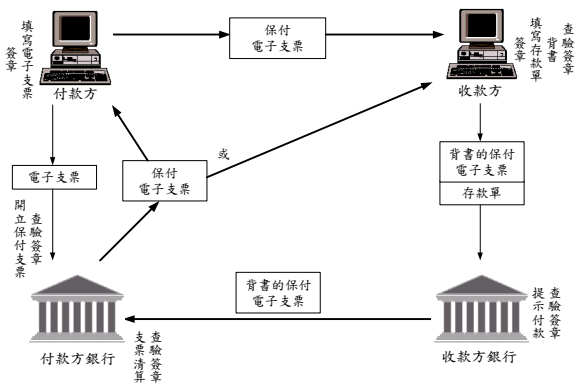

圖 5 保付電子支票流程

3.2 XML 技術簡介

XML (eXtensible Markup Language) 是 SGML (Standard Generalized Markup Language) 的子集合，由 XML 工作小組所開發。此一工作小組是在 1996 年由 W3C (World Wide Web Consortium) 所組成，其主要的工作在於讓 SGML 能像現有的 HTML (HyperText Markup Language) 一樣，可在 Web 上提供服務、接收及處理資料等(W3C, 1998; 徐子淵, 民 87)。

XML 具有自我定義(Self-Defined)與自我描述(Self-Description)的特性，讓 XML 比一般的 HTML 更具備了擴充性及結構性，很容易地被使用者與應用程式了解。此外，XML 的規格開放，可以讓不同的作業平台與應用程式進行資料交換，以減少不必要的平台轉換成本。總而言之，XML 設計的目標在於友善、便利與易於了解的統一規格。相較於 HTML 與 SGML，XML 具有撰寫便利、閱讀方便但又不失其結構性及擴充性的優點。

XML 文件由實體(Entity)的儲存單元所組成，而實體又可包含剖析過(Parsed)或未剖析過(Unparsed)的資料。已剖析過的資料由字元組成，其中一些字元組成字元資料，另一些字元組成標籤(Tag)。標籤包含了文件儲存格式(storage layout)及邏輯結構(logical structure)的描述。因此，XML 可說是提供一種可用於規範及限制儲存格式、邏輯結構的機制。

若要編寫一份格式「形式合法」XML 文件(Well-Formed Document)，需滿足以下幾個條件(W3C, 1998; 徐子淵, 民 87; 陳長念、陳勤意, 民 90)：

1. 選用正確的存檔編碼格式
2. XML 文件的第一列必須是 XML 宣告
3. XML 文件中只能有一個根標籤
4. 開始的標籤與結束的標籤缺一不可
5. 空的標籤需要有「/」符號
6. 所有的標籤需滿足巢狀排列
7. 英文字母的大小寫是有差異的
8. 屬性值需使用雙括號包圍
9. 文件中直接或間接引用的每個剖析過實體都是形式合法

當 XML 文件有引用 DTD 時，該文件除了必須滿足 Well-Formed 的條件外，還必須遵循該 DTD 所定義的法則。

DTD 的全名為 Document Type Definition，主要是定義 XML 的文件結構、元素的資料屬性、元素間的從屬關係及合法性。由 DTD 的內容中，我們可以看到根標籤應包含那些子標籤，而子標籤是否還含括再下一層的子標籤。除此之外，DTD 還記載元素的資料屬性、使用次數等屬性方面的設定。以下將以一例說明 XML 文件與 DTD 文件：

XML 文件

```
<?xml version=”1.0”?>
<customer order_number=”A9999”
<first_name>John</first_name>
<middle_initial/>
<last_name>Smith</last_name>
<shipping_address>
      <street>123</street>
      <city>New York</city>
      <state>NY</state>
      <zip>12345</zip>
      <country>USA</country >
</shipping_address >
</customer>
```

DTD 文件

```
<!ELEMENT customer (first_name, middle_initial?,
            last_name, shipping_address)>
<!ATTLIST customer order_number ID #REQUIRED>
<!ELEMENT first_name (#PCDATA)>
<!ELEMENT middle_initial (#PCDATA)>
<!ELEMENT last_name (#PCDATA)>
<!ELEMENT shipping_address (street, city, state, zip,
            country)>
<!ELEMENT street (#PCDATA)>
<!ELEMENT city (#PCDATA)>
<!ELEMENT state (#PCDATA)>
<!ELEMENT zip (#PCDATA)>
<!ELEMENT country (#PCDATA)>
```

在上述這份 XML 文件中，記錄著一筆訂單資料，而 XML 文件裡所使用到的標籤及標籤屬性則定義在 DTD 文件。由 DTD 文件，我們可以知道根標籤為 customer，其下的子標籤為 first_name、middle_initial、last_name、shipping_address。而在 shipping_address 下又有 street、city、state、zip、country 四個子標籤。此外，這些元素的資料格式為 #PCDATA。透過 DTD 文件我們可以了解這份 XML 文件的內容定義及標籤出現先後順序及其屬性格式。

當我們或機器看到一份 XML 文件後，會先去檢查是否符合 Well-Formed，再去透過 DTD 來檢查是否符合定義，以確定 XML 文件是否 valid。

3.3 XML 簽章介紹

W3C 於 1999 年 10 月提出工作草稿"XML-Signature Requirement"，開始 XML 簽章的研究發展。W3C 於 2001 年 8 月提出"XML-Signature Syntax and Processing"的建議文件，並於 2002 年 2 月 12 日成為 W3C 提議標準(Recommendation)。目前，也被 IETF 列為 RFC3275 號。該份標準指出 XML 簽章提供資料真確性(integrity)、訊息鑑別(message authentication)、簽章者身份鑑別(signer authentication)、簽章者不可否認(non-repudiation)服務，而簽章本身則可以應用於任何型式的數位化資訊內容及資料物件(W3C, 2002)。

XML 簽章是以 XML 來呈現一份簽章，而其設計原則 — 描述如何對數位化的文件內容進行簽章，並將金鑰與文件內容建立關聯。然而，在 XML 簽章的設計裡，並未考量到為金鑰與組織或個體建立關聯。W3C 所提出的 XML-Signature Syntax and Processing 文件 (W3C, 2002) 中，有針對 XML 簽章的基本結構進行定義。

```
<Signature ID?>
    <SignedInfo>
        <CanonicalizationMethod/>
        <SignatureMethod/>
        (<Reference URI?>
            (<Transforms>)?
            <DigestMethod>
            <DigestValue>
        </Reference>)+
    </SignedInfo>
    <SignatureValue>
    (<KeyInfo>)?
    (<Object ID?>)*
</Signature>
```

XML 對於 XML 簽章的標籤定義如下：符號"?"表示可能會有零個或是一個此種資料；符號"+"表示可能會有一個或是多個此種資料；符號"*"表示可能會有零個或是多個此種資料。資料結構與原則也如同 XML，而整個 XML 簽章的內容是以<Signature>與</Signature>兩標籤包起來，其中包括有 <SignedInfo> 、 <SignatureValue> 、 <KeyInfo>、<Object>等元素。

＜SingnedInfo＞的結構包含了 XML 簽章所需的一些相關內容，如標準化演算法(即 CanonicalizationMethod 的部分)、簽章演算法(即 SignatureMethod 的部分)及其他相關內容(即 Reference 的部分)。SignedInfo 結構中各元素內容意義分別說明如下：

(1) CanonicalizationMethod：XML 文件標準化的方式。在 XML 文件要簽章前，先將文件標準化(Canonical)，以保證同一份 XML 文件的內容格式一致，簽章運算時，數位簽章才會一致。

(2) SignatureMethod：定義產生及驗證數位簽章的簽章演算法。

(3) Reference：定義簽章所用訊息摘要演算法(即 DigestMethod 部分)，以及所要簽章資料的摘要值(即 DigestValue 部分，此內容必須以 base64 的型式顯示)。此外，也包含資料在進行訊息摘要運算前的程序與步驟(即 Transforms 部分)。

在＜Signature＞中，其他的元素還有＜SignatureValue＞、＜KeyInfo＞、＜Object＞，其意義分別如下：

(1) SignatureValue：此為放簽章值的元素，其內容是文件經簽章後，再經過 base 64 轉換後的內容。

(2) KeyInfo：此為驗證簽章所需金鑰的相關資訊。其中，所包含的資訊有金鑰名稱、金鑰值、金鑰取得方式、X.509 資訊、PGP 資訊等。

(3) Object：此元素內容可以是任何資料，就像 MIME 型態、MIME 的 ID 或編碼屬性。

針對上述的 XML 簽章結構，我們以下列的例子來做說明。

```
[S01] <Signature Id="MyFirstSignature"
       xmlns="http://www.w3.org/2000/09/xmldsig#"
       >
[S02]   <SignedInfo>
[S03]     <CanonicalizationMethod
            Algorithm="http://www.w3.org/TR/2001/
            REC-xml-c14n-20010315"/>
[S04]     <SignatureMethod
            Algorithm="http://www.w3.org/2000/09/x
            mldsig#dsa-sha1"/>
[S05]     <Reference
            URI="http://www.w3.org/TR/2000/REC-
            xhtml1-20000126/">
[S06]       <Transforms>
[S07]         <Transform
                Algorithm="http://www.w3.org/TR/2
                001/REC-xml-c14n-20010315"/>
[S08]       </Transforms>
[S09]         <DigestMethod
                Algorithm="http://www.w3.org/200
                0/09/xmldsig#sha1"/>
[S10]         <DigestValue>j6lwx3rvEPO0vK
                tMup4NbeVu8nk=</DigestValue>
[S11]     </Reference>
[S12]   </SignedInfo>
[S13]   <SignatureValue>MC0CFFrVLtk=Avty0opiP
          YUIN...</SignatureValue>
[S14]   <KeyInfo>
[S15a]    <KeyValue>
[S15b]      <DSAKeyValue>
[S15c]        <P>...</P><Q>...</Q><G>...</G>
[S15d]      </DSAKeyValue>
[S15e]    </KeyValue>
```

[S16] </KeyInfo>
[S17] </Signature>

[S02-S12] 簽章所需的一些相關內容，如標準化演算法(即 CanonicalizationMethod 的部分)、簽章演算法(即 SignatureMethod 的部分)、訊息摘要演算法(即 DigestMethod 部分)、摘要值(即 DigestValue 部份)等其他相關內容。由[S09]我們可以得知，此一 XML 簽章是採 SHA-1 的訊息摘要模式，而其摘要值在[S10]中標示。

[S03] 是將 XML 文件標準化的方式。[s04] 則是將 SignedInfo 轉換成標準化的內容後，再透過 SignatureMethod 所定義的演算法來產 SignatureValue。由[S04]我們可以得知，此一 XML 簽章是採 SHA-1 的訊息摘要模式，並以 DSA 方式進行簽章工作。

[S05-S11] 每一個 Reference 都包含訊息摘要演算法及經由演算法計算出來的結果。此外，也包含資料在進行訊息摘要運算前的程序與步驟。

[S14-S16] KeyInfo 指驗證簽章所需金鑰的相關資訊。其中，[S15a-S15e] 是記載此一 XML 簽章所採用之 DSA 金鑰的相關資訊。

由於電子支票的運作過程皆需利用簽章來達成權利與義務的轉移，所以簽章的動作在電子支票的運作中佔有相當重要的地位。因此，本研究將採用 XML 簽章的方式來記錄電子支票中所進行簽章動作的相關資訊。

四、研究成果與討論

本研究計畫在設計電子支票系統時，考量數位化的電子支票易於複製，為避免支票因重覆而產生諸多的爭議，所以採行集中保管的運作機制。也就是電子支票是存放於金融機構的資料庫內，由金融機構統一保管，使用者若要進行電子支票相關動作皆需透過金融機構的網站來進行。

本研究計畫利用 XML 定義適合我國國情的電子支票，並設計一具備開戶、簽發支票與帳戶查詢功能的電子支票系統雛形。首先，將針對以 XML 制定電子支票標籤進行介紹。其次，再介紹本研究所設計的電子支票運作流程。最後，以系統畫面簡介本研究的系統雛形。

4.1 電子支票系統標籤設計

民國七十六年六月二十九日財政部臺財融字第七六０八一七五七０號公告、法務部法檢字第七四二三號公告第一百四十一條、第一百四十二條，公告票據法於中華民國七十五年十二月三十一日屆滿而廢止。票據法在我國實行五十八年後，終於宣佈廢止。然而，票據法對我國匯票、本票及支票的發展確有不可憾動之勢，即便是公告廢除但現行票據的型態、格式與運作依然不變。因此，本研究還是參考票據法的規範設計電子支票標籤。根據票據法第 125 條 (票據法, 民 76) 之規定，支票上應記載表明其為支票之文字、一定之金額、付款人之商號、受款人之姓名或商號(未載受款人者，以執票人為受款人)、發票地(未載發票地者，以發票人之營業所、住所或居所為發票地)、發票年月日、付款地，並由發票人簽名。

參考目前紙本支票上的記載，支票尚需記載支票號碼與帳號，而此處的帳號是指付款人的銀行帳號。而在金額記載方面，為避免國人書寫錯誤，一律同時有國字與阿拉伯數字兩種金額的記載，並且依票據法規的規範，當兩者金額不等時，以國字書寫之金額為準，而在電子支票系統中雖已加入系統自動檢查及使用者再確認的機制，可消除此項紙本支票使用上可能發生的錯誤，但考量符合國人的使用習性，所以還是保留兩種書寫方式。此外，參考 FSML (Jeff, 1996) 的設計，為了增加金融機構對付款人的了解，遂添加付款人帳戶資訊的標籤<account>。

綜整上述所提，一張電子支票的根標籤定義為<eCheck>，其下分為支票內容<checkdata>、付款人帳戶資訊<account>及簽章<signature>三大子標籤，此三個子標籤又有更細部的標籤設計。電子支票的簽章是採行 XML 簽章的技術，將支票內容與付款人帳戶資訊一起進行簽署。

```
<eCheck>電子支票
    <checkdata>支票內容
        <chkNo>支票號碼</chkNo>
        <chkDate>發票年月日</chkDate>
        <CustAcct>帳號
        <chkPayTo>收票人</chkPayTo>
        <currency>幣別</currency>
        <chkAmount>支票金額</chkAmount>
            <Amount>阿拉伯數字書寫之金額
            </Amount>
            <NT>國字書寫之金額</NT>
        </chkAmount>
        <bankName>發票銀行</bankName>
        <bankAddr>付款地</bankAddr>
    </CheckData>
    <account>付款人帳戶資訊
        <acctTitle>付款人</acctTitle>
        <acctType>帳戶型態</acctType>
        <bankName>銀行名稱</bankName>
        <bankCode>銀行代碼</bankCode>
        <bankAdd>銀行住址</bankAdd>
        <bankPhone>銀行電話</bankPhone>
    </account>
    <signature>數位簽章</signature>
</Check>
```

4.2 電子支票系統運作流程

本研究所設計的電子支票系統是將電子支票儲存於金融機構的資料庫，當使用者要進行開票、支票查詢工作皆需透過金融機構始能運行。此一設計概念，雖然易造成金融機構沉重的工作負擔，但可避免因電子支票的複製所造成爭議及成本損失。

本系統雛形的參與個體有金融機構、發票人與收票人。本系統裡共有三個基本功能，分別為(1)產生個人金鑰對、(2)開立電子支票、(3)電子支票帳戶查詢。以下將針對每個功能進行說明。

(1) 產生個人金鑰對

使用者第一次使用此一電子支票系統需先申請一組金鑰對，以下將針對產生個人金鑰對的流程進行介紹(圖 6)。

使用者需先通過系統的身份鑑別(步驟 1)。待通過鑑別後，使用者可要求產生金鑰對(步驟 2)，而此時系統會要求使用者填寫相關個人資訊(步驟 4)。填寫完畢後，系統會對個人資訊進行比對，在查核完畢後就產生一組金鑰對，並顯示於使用者端(步驟 5)。使用者在得知此一金鑰對後，即按下"確認"鍵，代表已認可此組金鑰(步驟 6)。金融機構會將使用者個人資料及金鑰對傳送給憑證機構(certificate authority)，以要求核發使用者的憑證(certificate) (步驟 7)。憑證機構在確認使用者的身份及金鑰組後，便可核發使用者憑證，並將此一憑證傳送給金融機構。金融機構在接到憑證後，即可將使用者憑證儲存於資料庫內，以供後續簽發電子支票之用(步驟 8)[1]。

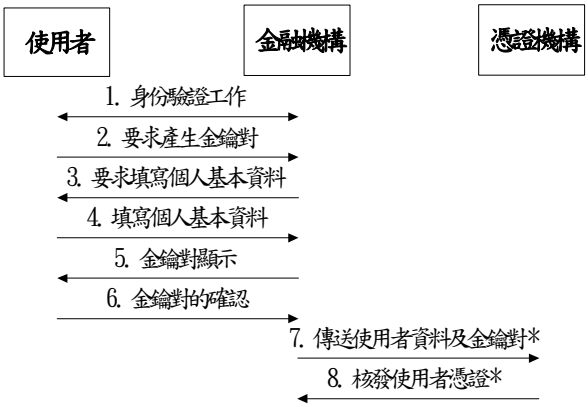

圖 6 產生個人金鑰對流程圖

(2) 開立電子支票

發票人在開立電子支票前，需先申請一金鑰組才可開立電子支票，以下將針對開立電子支票的流程進行介紹(圖 7)。

發票人須先登入金融機構的電子支票系統並通過身份鑑別(步驟 1)。在通過身份鑑別後，發票人可要求開立電子支票(步驟 2)。金融機構在收到發票人的要求後，即顯示一張尚未填寫基本資料的電子支票(步驟 3)，發票人只需填寫收票人、發票年月日、支票金額等並傳送給電子支票系統(步驟 4)。系統在整理發票人的開票資料後，就將一份內容完整但未簽章過的電子支票顯示在發票者端，讓發票人對支票內容做再次地確認(步驟 5)。若發票人確認支票內容無誤，即可由金融機構端對此份電子支票進行簽章動作，並顯示於發票人端的螢幕上，以做為最後的確認(步驟 6)。發票人在確認一切無誤後，就可要求金融機構儲存與傳送電子支票。最後，發送支票開立結果給發票人，並以 e-mail 方式通知收票人已接到一張新的電子支票，並請收票人至自己的帳戶內查詢(步驟 7)。

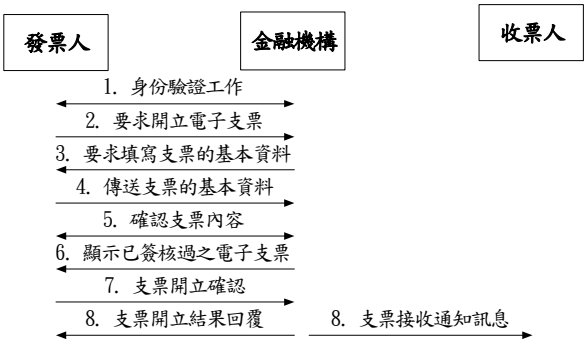

圖 7 開立電子支票流程圖

(3) 電子支票帳戶查詢

使用者可以透過電子支票系統的查詢功能(圖 8)，查閱帳戶內的支票使用現況、交易記錄等。使用者需通過電子支票系統的身份驗證(步驟 1)，再要求查詢使用者的個人帳戶內資訊記錄(步驟 2)。系統會以表格方式，顯示使用者要求查詢的結果(步驟 3)。目前規劃可供查詢的內容為接收到之電子支票列表、開立之電子支票列表、使用者個人帳戶資料列表。

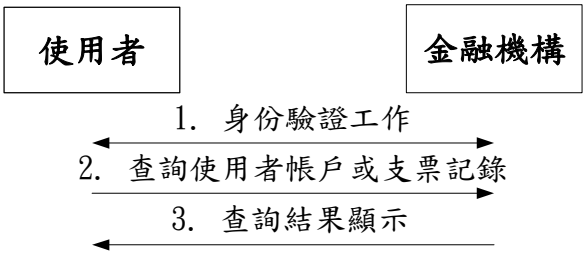

圖 8 電子支票帳戶查詢流程圖

4.3 電子支票系統雛形展示

本節將針對本研究所設計的電子支票系統雛形作介紹。第一小節將對系統開發規劃、系統功能作介紹，第二小節則詳述系統各模組的功能，最後就以系統畫面展示系統雛形開發的成果。

4.3.1 系統規劃

本研究所設計所展示的系統為網頁型態的電子支票系統雛形，而開發的環境在 Microsoft Windows 2000 作業系統上架設 IIS Server，後端資料庫使用 Microsoft SQL 7.0，使用 ASP、VBScript、XML 等程式語言撰寫而成。以下為本系統運作流程(如圖 9)，為本研究所建置的系統與後端資料庫間的運作。

[1] 憑證機構最主要的功能在於核發及管理憑證，在本研究裡憑證機構則在產生金鑰階段介入，故憑證機構在本研究中為支援角色。再者，金融機構可協助使用者產生金鑰對，亦以擔負起註冊機構(registration authority)之職責。所以，本系統雛形的開發將不另外設計憑證機構，而單純僅以金融機構所產生的金鑰來簽核支票，而不加入憑證的申請(步驟 7~8)。

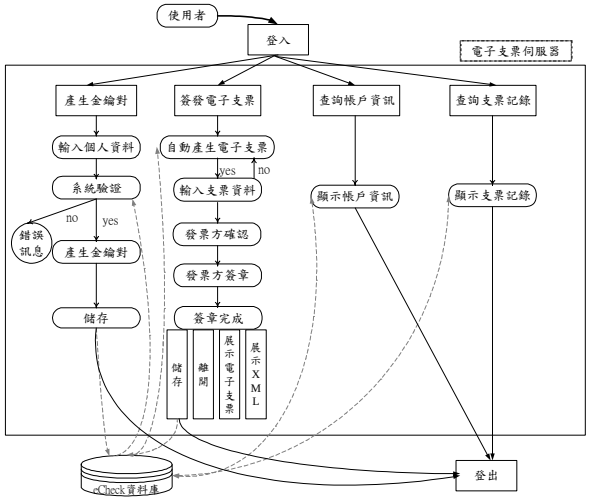

圖 9 系統運作流程圖

4.3.2 系統模組介紹

本研究所開發的系統雛形共可分為五個模組，分別為(1)使用者身份查驗模組、(2)金鑰對產生模組、(3)電子支票簽發模組、(4) 個人帳戶及支票記錄查詢模組、(5)電子郵件傳送模組。

(1) 使用者身份查驗模組：使用者在使用電子支票前需到銀行進行開戶的動作，並取得一組識別碼及通行碼。使用者若要使用電子支票服務前，必須先通過使用者身份查驗模組的查核。

(2) 金鑰對產生模組：本研究所採行的簽章方法是 RSA 簽章法，因此需要預先產生一組 RSA 金鑰。RSA 金鑰的產生是當使用者第一次登入系統時，由系統透過亂數產生使用者的私密金鑰、公開金鑰，並且存入資料庫中，以做為往後簽核電子支票之用。此外，每個使用者只能操作此功能一次，若系統中已儲存使用者回的金鑰，系統會使用者的請求。

(3) 電子支票簽發模組：此模組主要是讓系統連接後端資料庫，自動產生一張電子支票，而上面已記載等同於紙張空白支票所擁有的訊息，讓使用者在輸入支票資訊後確認、簽章並儲存。本系統對電子支票的簽核是先利用 MD5 的單向赫序函數對電子支票做訊息摘要，再從資料庫取出使用者的私密金鑰對訊息摘要進行簽章運算。最後，將訊息摘要連同簽章依 XML 簽章的格式，附加在電子支票後面。

(4) 查詢模組：使用者可透過查詢模組查詢銀行中的支票存款帳戶及個人簽發過、接收到的支票記錄。

(5) 電子郵件傳送模組：發票人在開立、簽署電子支票後，會將電子支票儲存於後端資料庫中。在此時同時，系統會透過電子郵件傳送模組將此一訊息告知收票人。

4.3.3 系統雛形說明與展示

(1) 登入

本系統的首頁提供身份驗證的功能(圖 10)，讓使用者需輸入個人的識別碼及通行碼，待通過後才能執行電子支票系統的服務。

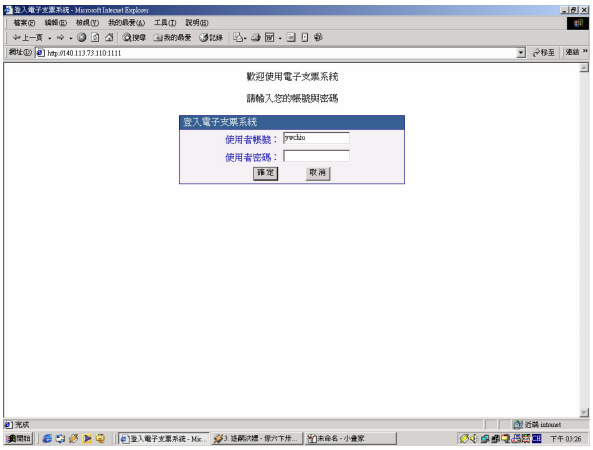

圖 10 電子支票系統登入畫面

若使用者輸入錯誤的識別碼及通行碼，則會出現錯誤提示訊息(圖 11)。

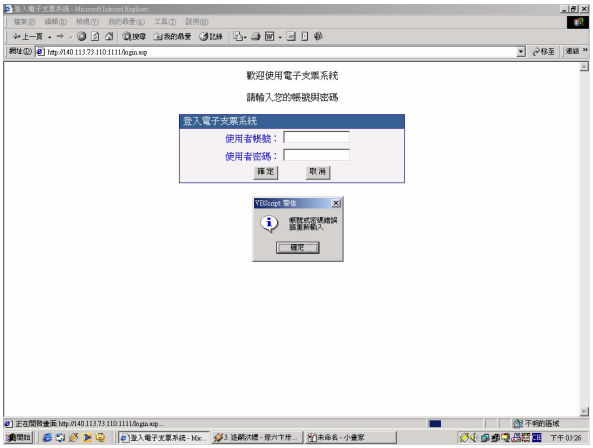

圖 11 登入失敗—錯誤提示訊息

合法的使用者在登入電子支票系統後，即進入系統的主畫面(圖 12)。上半部為主選單，下半部則顯示歡迎與提示訊息。主選單的選項包括有「首頁」、「簽發電子支票」、「查詢個人帳戶」、「查詢支票記錄」、「產生個人金鑰」及「登出」的功能。

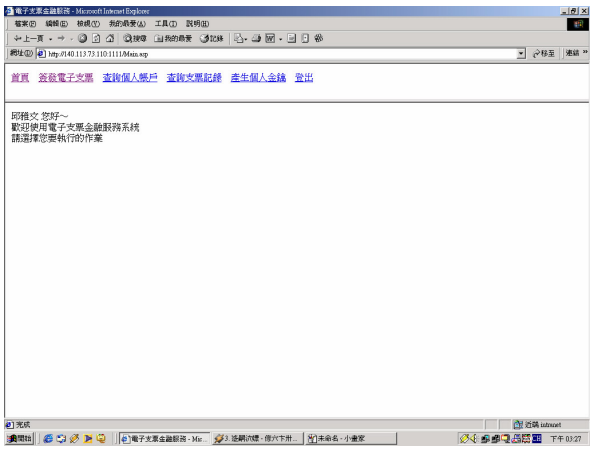

圖 12 系統主畫面

(2) 產生個人金鑰對

第一次使用電子支票系統的客戶，必須先產生個人金鑰。使用者在進入產生個人金鑰的畫面，可以看到系統要求使用者先輸入帳號、姓名及身份證字號，以做為確認使用者身份之用。使用者在填寫相關資料後，需按下”產生金鑰對”按鈕(如圖 13)，系統比對使用者輸入的資料是否與後端資料庫所儲存的資料一致。若不一樣則會出現「輸入錯誤，請重新輸入」的訊息(圖 14)。

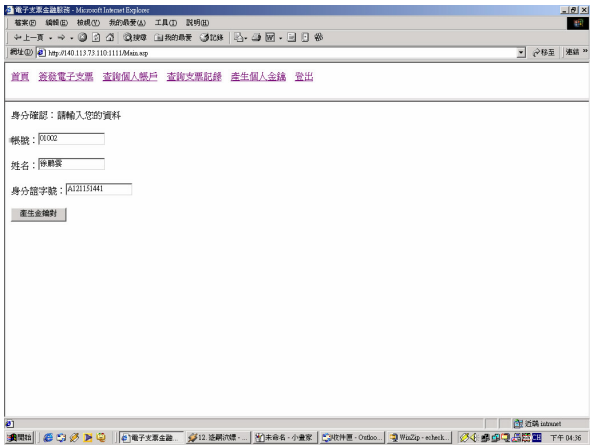

圖 13 產生個人金鑰產生畫面

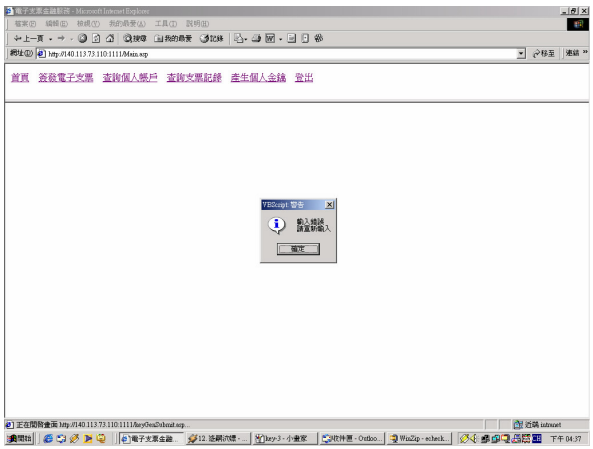

圖 14 資料有誤—錯誤提示訊息

若使用者身份確定無誤後，系統會亂數產生一組金鑰對(圖 15) — 公開金鑰(Public Key)及私密金鑰(Private Key)，並顯示於畫面上。不論是公開金鑰，亦或是私密金鑰，皆可看到有一”+”符號。對公開金鑰而言，在”+”前為 e 值，在”+”後則為 n 值；對私密金鑰而言，在”+”前為 d 值，在”+”後則為 n 值。當使用者按下”確定”鍵後，系統會顯示「新增個人金鑰成功!」的訊息，並且將金鑰對存入系統的資料庫中。

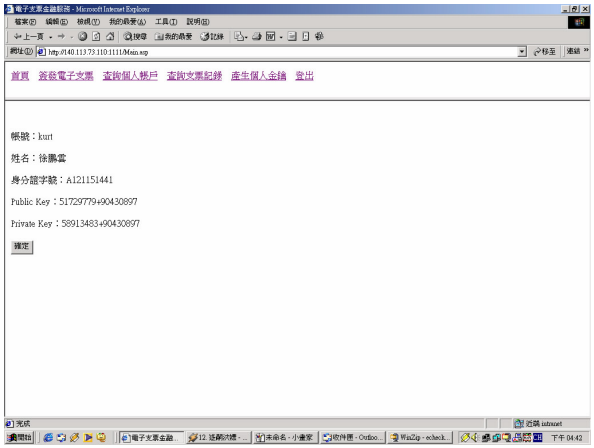

圖 15 產生個人金鑰確認畫面

如果使用者已在本系統存有金鑰對，而又再點選此功能，下方的視窗會顯示「個人金鑰已存在，請選擇其他功能！！」的訊息(圖 16)。

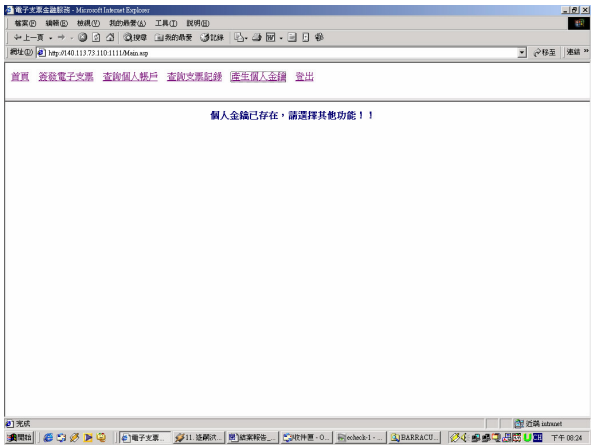

圖 16 拒絕重覆申請金鑰畫面

(3) 簽發電子支票

使用者選擇簽發電子支票功能時，系統會立即連接後端資料庫，選擇一筆未簽發的支票資料，擷取支票號碼、使用者的帳號及付款銀行等資料，並且顯示在螢幕上(圖 17)。

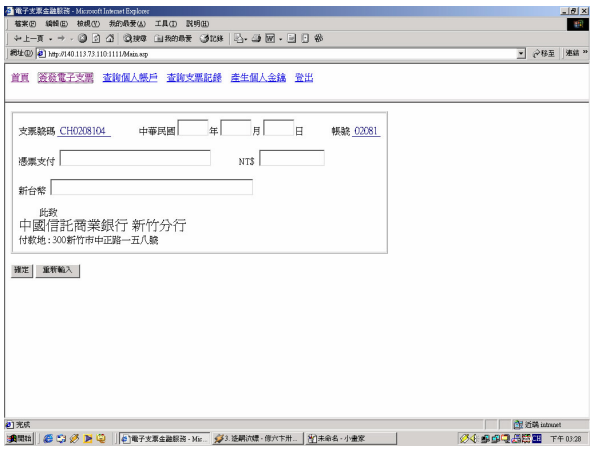

圖 17 簽發電子支票畫面

發票人只需將收票人、日期、金額填入(圖 18)並按下"確認"即會進行下一步驟。若使用者要取消此筆資料，只要按下"重新輸入"，將會消除全部填寫的資料。

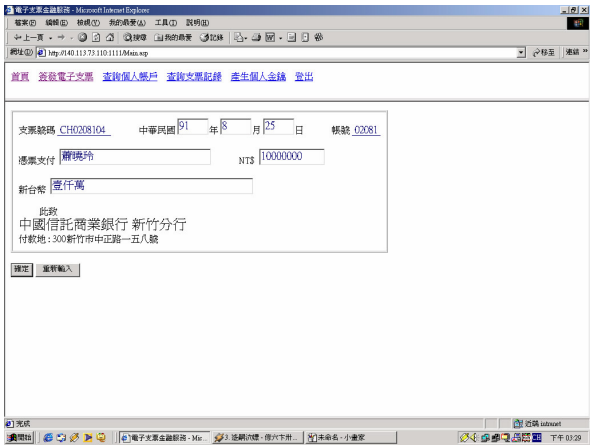

圖 18 填寫電子支票畫面

發票人在填寫資料完成並按下"確認"鍵後，系統會將支票的資料再次顯示於螢幕上，供發票人做再次地確認。若資料有誤，就可點選"重新輸入"，回到上一頁；若資料無誤，就可點選"確定簽章"(圖 19)。在確定要進行簽章後，系統會先將電子支票的內容轉換為 XML 格式，並以 MD5 產生訊息摘要，再由資料庫取出使用者的私密金鑰簽署訊息摘要以產生數位簽章。

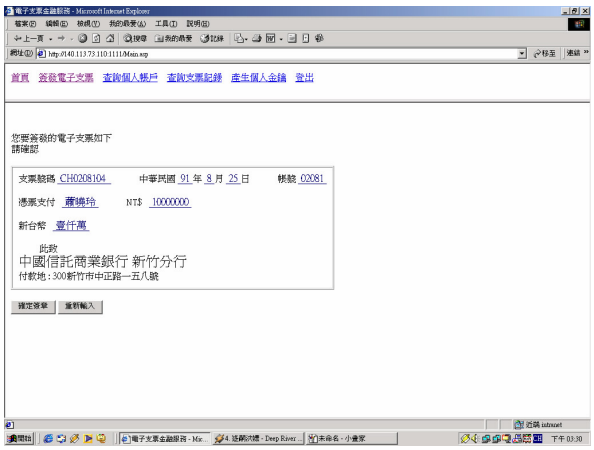

圖 19 電子支票確認畫面

簽章動作完成後，系統會顯示圖 20 的畫面，使用者可選擇"檢視電子支票"、"檢視電子支票 XML 格式"、"確定儲存"或直接"離開"。

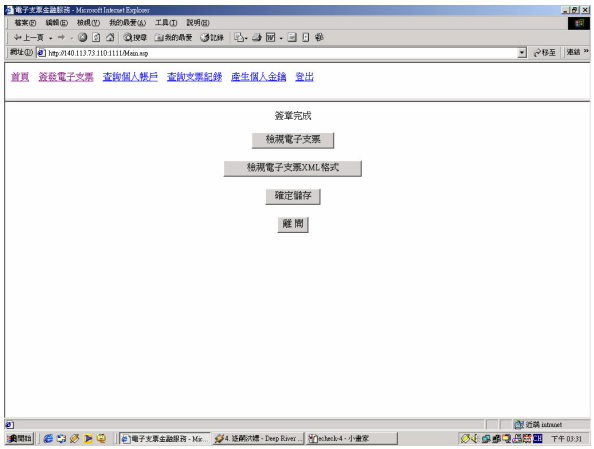

圖 20 簽章完成畫面

發票人若選擇"檢視電子支票"，螢幕上會跳出新的視窗並顯示已簽章的電子支票(圖 21)，畫面中發票人數位簽章的區塊正是經過發票人私密金鑰簽章過的簽章值。

圖 21 電子支票的數位簽章

若是選擇"檢視電子支票 XML 格式"，螢幕上會跳出新的視窗並以 XML 格式來呈現已簽章的電子支票(圖 22)。

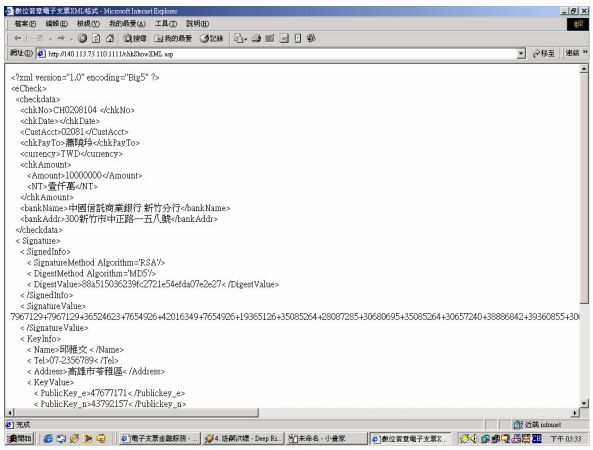

圖 22 電子支票的 XML 格式

發票人在檢視完畢已簽章的電子支票後，需按下”確定儲存”鍵。系統會將會連接後端的資料庫，以進行一連串的支票處理程序。待執行完成後，系統會依收票人所登錄的 E-mail 位址，寄發 E-mail 來通知收票人。系統在確認信件寄出後，即顯示「成功寄送電子郵件通知收票人取票」的訊息(見圖 23)。對於收票人而言，則會收到如圖 24 的電子郵件，以通知收票人已接收到新的電子支票。

若發票人未選擇”確定儲存”就離開，系統不會將資料寫入資料庫，即之前簽發的電子支票無效。

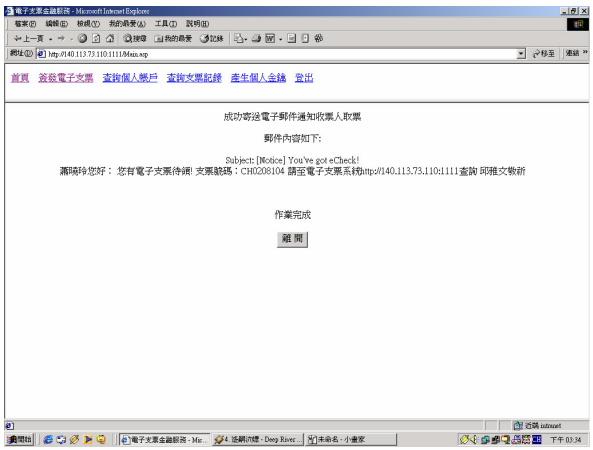

圖 23 發票者端電子支票簽核成功畫面

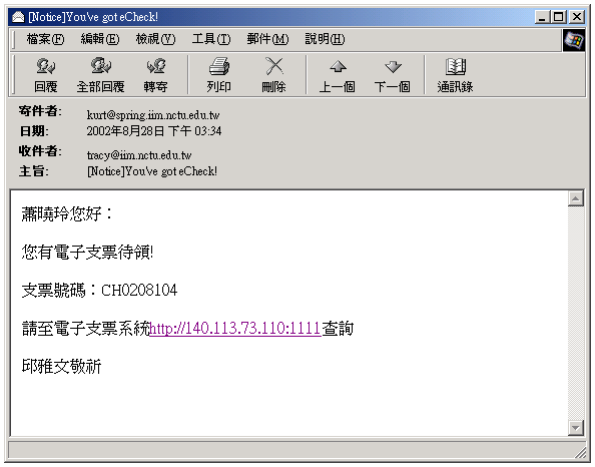

圖 24 通知收票者領取電子支票

(4) 查詢個人帳戶

使用者若選擇”查詢個人帳戶”功能，系統會連結後端資料庫，顯示使用者帳號、存款餘額、支票簿號碼(圖 25)。

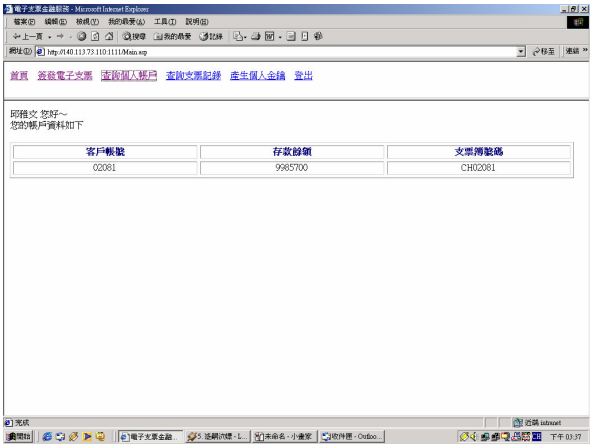

圖 25 查詢個人帳戶畫面

(5) 查詢支票記錄

使用者若選擇”查詢支票記錄”功能，系統會連結後端資料庫，並依使用者已簽發及使用者已接收到的電子支票分別顯示。顯示的資料包括：支票號碼、日期、帳號、憑單支付、阿拉伯數字金額、國字金額、銀行名稱及銀行地址(圖 26)。

圖 26 查詢支票記錄畫面

4.4 結論與討論

依據 eCheck 組織針對支票歷史發展所發布的文件(eCheck, 1995)，最早有支票的文獻記載出現於西元一千五百年左右。時至今日，支票依然為商業交易中不可或缺的支付工具。支票不同於現金的使用，支票的使用可便利使用者攜帶、清點，減少安全顧慮。然而，在網路時代裡，眾多電子付款系統紛起，唯獨電子支票的發展最為緩慢，深究其原因，大多因配合支票的相關法律未獲修改，而造成對電子支票的發展裹足不前。

本研究將跳脫出修法的限制，由技術角度出發，配合我國國情、社會、使用習性設計適用於我國的電子支票系統，並以實作方式完成一系統雛形，做為我國推動電子支票的先鋒。我們所設計的

系統雛形，具有申請金鑰對、開立電子支票、帳戶查詢功能。除此之外，為促進金融機構與各企業金流系統的整合，遂採行具有跨平台資料交換功能的 XML 語言，以便利企業與金融機構間的資料傳輸，降低資料格式轉換的成本。

五、計畫成果自評

本計畫的研究方向與研究方法與原計畫建議書所提相當一致，在研究成果方面，也達成預期的目標 —— 制定 XML 標籤、設計電子支票架構並發展電子支票系統雛形，以作為未來發展電子支票系統的參考。如此，相信可積極推動國人的使用，俾使電子支票成為國內商業活動中付款的主要工具。

依據 ISM 及 Forrester Research 的調查，2002 年第二季企業在線上採購直接物料的比率從第一季的 53%躍升至 64.4%；間接物料的比率也從第一季的 78.1%上升至 84.2% (資策會電子商務應用推廣中心，民 91 年 b)。企業間的交易漸漸地轉移到網際網路上進行，而網際網路上的支付工具越顯重要。對於每個企業而言，大多希望能採用與企業內部 ERP 系統或財會系統連接的網際網路支付工具。然而，目前所發展的電子支票系統雖可做為 B-2-B 交易的支付工具，但對於與企業資訊系統整合以達成金流自動化的目標尚有一段距離。因此，本研究將繼續深入探討企業內的資訊系統應如何有效率地與金融機構的電子支票系統進行連接，以促進 B-2-B 交易處理的效率。

在本計進行過程中，已深入研究電子支票架構、XML 相關技術，撰寫一學術論文(註一)，並已刊登於 ICIM2002。此外，亦有學生以此方向為題發展一碩士論文(註二)。

[註 1] 黃景彰、邱雅文. (民 91 年). 一種網際網路上的延遲付款機制 — 電子支票. Proceedings of the 13th International Conference on Information Management, 論文集(I) - 581.

[註 2] 邱雅文. (民 91 年). 一種網際網路上的延遲付款機制 — 電子支票, 國立交通大學資訊管理研究所.

六、參考文獻

Anderson, M. M. (1998). The electronic check architecture. Retrieved September 15, 2000 from the World Wide Web: http://www.echeck.org/library/wp/ArchitectualOverview.pdf

eCheck. A brief history of checking. (1995). Retrieved September 15,2000 from the World Wide Web: http://www.echeck.org/library/history.html

Jeff, K. (1996, February). FSML – Financial Service Markup Language. Version. 1.50. Retrieved September 15, 2000 from the World Wide Web: http://www.echeck.org/library/ref/fsml-v1500a.pdf

World Wide Web Consortium (W3C). (1998, February 10). Extensible Markup Language (XML) 1.0 specification. Retrieved July 15, 2002 from the World Wide Web: http://www.w3.org/TR/1998/REC-xml-19980210

World Wide Web Consortium (W3C). (2002, February 12). XML-Signature syntax and processing. Retrieved July 15, 2002 from the World Wide Web: http://www.w3.org/TR/2002/REC-xmldsig-core-20020212

iamasia，(2001, October)，iamasia 發表台灣地區最新線上金融調查報告。於 2001 年 12 月 5 日，由 http://www.iamasia.com/presscentre/pressrel/press_tc_1011.cfm 取得。

王毓仁，(民 85 年)，支票使用實務(第五版)，台北市：書泉出版社。

徐子淵(譯)，(民 87 年)，可擴展標示語言(XML)規格書，於民國 89 年 11 月，由 http://www.twtec.org.tw/XML_spec.htm 取得。

陳長念、陳勤意，(民 90 年)，活用 XML，台北市：知城數位科技。

票據法，財政部臺財融字第七六０八一七五七０號公告，(民 76 年 6 月 29 日)。於民國 91 年 8 月 13 日，由 http://www.ntifo.org.tw/law/law203.htm 取得。

資策會電子商務應用推廣中心，(民 91 年 a)，2002 年第二季企業線上採購比率大幅躍升。於民國 91 年 8 月 13 日，由 http://www.find.org.tw/0105/news/0105_news_friendly_print.asp?news_id=2230 取得。

資策會電子商務應用推廣中心，(民 91 年 b)，亞太地區網路及電子商務在 2006 年將有亮麗表現。於民國 91 年 8 月 13 日，由 http://www.find.org.tw/0105/news/0105_news_disp.asp?news_id=2247 取得。