

軟體品質標記語言的設計及應用

Design and Applications of Software Quality Markup Language

張鴻全

Hun-Chuan Chang

元智大學資訊工程研究所

中壢遠東路 135 號

Yuan-Ze Universit

Chung-Li, Taiwan

范金鳳

Chin-Feng Fan

元智大學資訊工程研究所

中壢遠東路 135 號

Yuan-Ze Universit

Csfanc@saturn.yzu.edu.tw

易俗

Swu Yih

核能研究所核儀組

龍潭郵政 3-11 號信箱

Institute of Nuclear Energ

Research, Taiwan, R.O.C.

摘要

本研究以 ISO 9001(ISO 9000-3)為實例，提出以「產品生命週期」與「缺失生命週期」的概念結構來發展「品質標記語言」及相關的標籤處理技術。在應用上我們提出 ISO 認證分級制度的新構想及方法，利用品質標記語言所建構的超文件系統，可有效的檢索出決定分級所需的相關資料，以輔助分級制度的推行。此外，我們也開發了 ISO 稽核與文件維護所需的工具，並提供「多重觀點」(multiple views)的文件導覽功能。本研究所發展的方法可提高超鏈結建構的效率，並提供趨近概念層次的電子文件閱讀方式。

關鍵詞：超文件、品質標記語言、缺失生命週期、ISO 認證、概念為基礎的超文件。

Abstract

This research used ISO 9001(ISO 9000-3 as guidelines) as a case study. We proposed "product life cycle" and "defect life cycle" as the semantic framework for designing "Quality Markup Language" (QML) along with its related tag processing techniques. We suggested that QML could be used to retrieve information to classify ISO certification into different levels. A tool for QML has been constructed to support ISO auditing and document maintenance. Besides, this tool provides document navigation from different viewpoints. Our developed techniques facilitate efficient hyperlink construction and turn the information retrieval towards a concept level.

Keywords: hypertext, Quality Markup Language, defect life cycle, ISO certification, concept-based hypertext.

1 緒論

現行的電子文件技術，偏重在文件的版面配置與檢索效率，對於閱讀理解效率的提昇並沒有很大的幫助。超本文(hypertext)解決了人們閱讀文件時互相參考的困擾，提昇了資料搜尋的效率，但每異動一個鏈結，即需以人力的方式加以維護，因此超鏈結(hyperlink)的維護需要不少人力。另一方面，這類超文件由作者預先安排的鏈結方式，並不一定適合所有使用者的需要。因此如何增進閱讀理解效率、並減少維護時的人力負擔是電子文件技術重要的研究方向。

針對以上議題，我們發展了以概念為基礎的超文件技術，首先萃取文件內容的語意概念，利用標記語言來標記此類概念，並發展進一步的標籤處理技術，以完成一檢索便利的超文件系統，提供進一步的應用。我們以 ISO 9001 應用於軟體產品為實例(以 ISO 9000-3 為指引)，提出軟體的「產品生命週期」(Product Life Cycle)與「缺失生命週期」(Defect Life Cycle)的語意結構(semantic framework)，以萃取此應用領域的語意概念，並發展了「品質標記語言」及相關之標籤處理技術。

在此品質標記語言的應用上，本文提出了 ISO 認證分級制度的新構想及方法。利用所發展的軟體品質標記語言及相關工具，可有效的檢索出決定分級所需的相關資料，輔助分級制度之推行。本研究所完成的工具亦支援 ISO 稽核及文件維護的作業，並提供「多重觀點」(multiple views)的功能，即不同的使用者可以不同的觀點來應用相同的標籤，可增加電子文件的應用效率及彈性。

本文第二節介紹我們的品質系統語意結構及品質標記語言，第三節討論我們所提出的標籤處理技術，第四節說明品質標記語言的應用，接著介紹工具，最後為結論。

2 品質標記語言 QML

我們提出以「缺失生命週期」(defect life cycle)與「產品生命週期」(product life cycle)，來表達此品質系統的語意結構(semantic framework)(如圖 1 所示)，用此語意結構我們可以萃取此領域的主要概念及其結構，以發展品質標記語言。

我們定義「缺失生命週期」為缺失起源(defect source)、缺失預防(defect prevention)、缺失偵測(defect detection)與缺失移除(defect removal)等缺失生命週期階段。ISO 9001 品質系統強調品質保證與管理，也就是要減少缺失的發生，軟體產品在相關的需求、設計與撰寫程式碼等過程中，缺失可能產生，為了達成品質要求，必須執行缺失預防、偵測與移除等品質管制與品質保證措施。當偵測到缺失後，必須執行缺失移除程序。完成了缺失移除程序後，仍必須再一次執行缺失偵測程序，以確保缺失已經完全移除，且無新引進的缺失。品質保證的目標是使剩餘缺失(residual defect)減至最少，所以在缺失偵測至缺失移除階段以反覆(迴圈)的方式進行。

「產品生命週期」則包含需求、設計、程式碼編寫/產品製成、測試等各階段，同時在產品生命週期各階段中實施管理責任、型態管理、內部品質稽核與問題解決等支援活動，以提昇產品之品質。ISO 的各項活動皆以這些產品生命階段來進行。上述缺失偵測移除等品質相關活動需在產品的發展階段中進行。我們提議合併使用「產品生命週期」以及「缺失生命週期」來表達 ISO 品質文件系統的主要概念及內涵。

根據以上的語意基礎結構(semantic framework)，我們進一步發展了「品質標記語言」(Quality Markup Language, QML)。在品質標記語言中，我們根據缺失生命週期，定義了品質相關的概念，即缺失預防、偵測、移除的概念。這些概念又可細分為技術(techniques)、資源(resource)與經驗(experience)等下層概念(如圖 2 所示)。這些概念可以實際應用在檢核分級評審相關的資訊上。而 ISO 9001 的 20 項條款亦可與產品缺失週期相互對應。例如 ISO 的第一條管理責任及第二條品質系統屬於缺失預防、偵測及移除，第十條檢視測試屬於缺失偵測，第十四條矯正預防活動屬於缺失預防及移除等。

圖 1. 語意結構

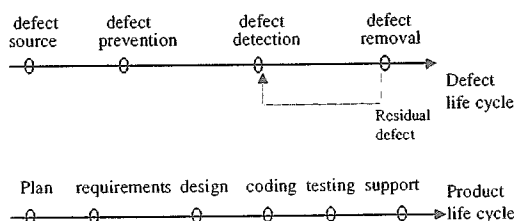
品質標記語言(QML)亦針對產品之生命週期，定義了合約、計畫、需求、設計、程式碼編寫、測試、支援活動等上層概念，ISO 9001 的 20 項條款亦可由此產品生命週期來區分，如 ISO 的第一條管理責任屬於支援活動，第三條合約審查屬於生命週期的合約階段，第四條設計管制屬於設計階段等。除了上述概念結構外 QML 亦包含一般性的標註以表達文件的格式或主題的呈現方式，其萃取方法則與現行一般標註的制定相同。圖 2 為 QML 部份的上層概念標籤。圖 3 為 QML 以 SGML 的形式表達。

3 標籤處理技術(tag processing techniques)

品質標記語言定義後，ISO 9001、ISO 9000-3、廠商品質系統文件與軟體專案文件等可以用 QML 標籤加以標記，這些標籤需要轉成超鏈結，才能進一步支援以概念為單位的檢核。

我們發展的標籤處理技術，包含 (1)將品質標記語言的標籤轉換成地址性的超鏈結(address-based hyperlink) (2)閱讀時提供即時展現(pop up)相關文件的相同標籤部份(3)不同使用者在閱讀點選相同標籤時，可以提供檢視不同觀點的相關文件部分。這些處理技術主要仰賴內建的索引資料庫。標註了 QML 的文件需先納入資料庫，納入時可利用所建的工具，將文件中標籤的名稱索引鍵入資料庫。標籤鏈結的資料結構示意圖如圖 4 所示。此資料結構可支援瀏覽模式下各項功能的實作。

首先介紹 ISO 文件系統瀏覽部分的設計。我們針對 ISO 文件系統提出雙十字形的邏輯關聯關係做為概念導航的結構，包括縱向邏輯結構與橫向邏輯結構兩部份：(1)縱向：抽象至具體的關係，即 ISO 9001、ISO 9000-3、品質系統與軟體專案間之關係。(2)橫向：主題與因果關係，即前述之「缺失生命週期」



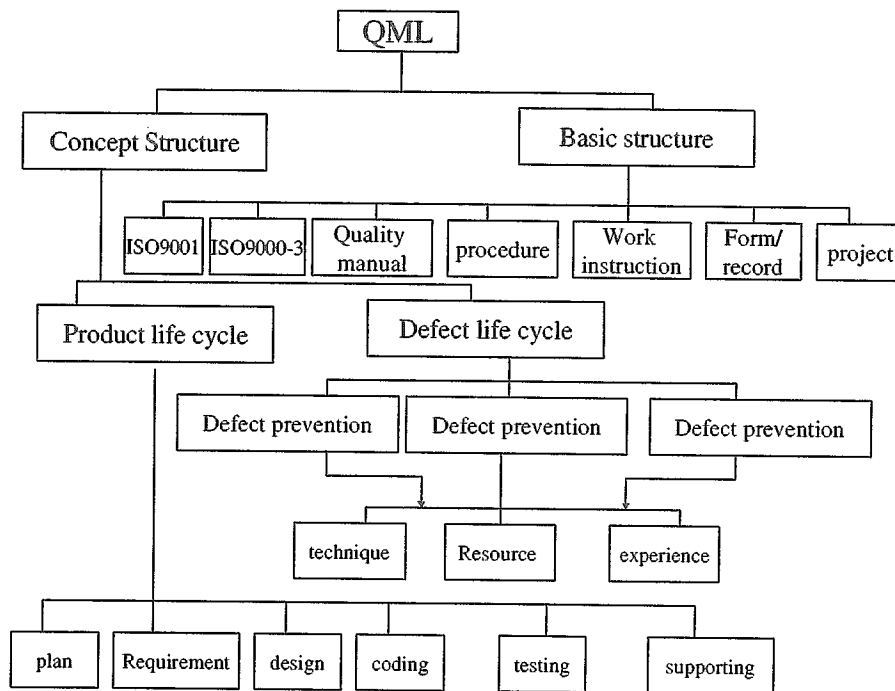


圖 2. QML 的上層概念結構

與「產品生命週期」。縱向邏輯結構的目的在於表達條款之間抽象與具體、目標方法等的關聯關係。橫向邏輯結構的目的在表達文件內的目的、主題、或因果關係。這兩組橫向結構提供文件追溯與相依的關係(如圖 5 所示)。這個縱向及橫向的雙十字型的結構可以作為指導閱讀的概念導航圖(concept navigation map);藉由這兩種關聯資訊之輔助,閱讀者便可經由概念導航圖位參考座標引導其閱讀過程。此導航示意圖如圖 6 所示。

在上述瀏覽設計下,我們的標籤處理進行如下:

- (1) 將品質標記語言的標籤轉換成地址為主的超鏈結:

文件內的標籤索引資料,首先需內建於資料庫中,使用者點選導航圖中的橫軸與縱軸上的標籤,系統至索引標籤料庫內,擷取文件檔案名稱及其索引標記,再透過輔助工具提供的瀏覽器,以較醒目的方式顯示相關的文件。

- 2) 閱讀提供即時展現(popup)相關文件且具相同標籤的功能:

當閱讀者瀏覽閱讀文件時,可提供選擇開啟在橫軸或縱軸上與此文件相關(相鄰)文件的相同標籤處。在執

行上可利用上述已建立之標籤索引及文件間之雙十字結構來實作此項功能。

- 3) 閱讀時不同使用者點選相同標籤時,

可以提供多重觀點的相關文件:

多重觀點(multiple views)的功能指不同的使用者在使用相同的標籤時,可以不同的方式加以瀏覽(如圖 12 示意圖所示),例如發展者在點選檢視 QML 中 Requirement 中的某依功(Requirement)時,本輔助工具可由 Functional Requirement 中的追溯屬性(attribute traceback)提供追溯至「規格」與「程式碼撰寫」的相關文件部份。而稽核者點選檢視此標籤時,本工具則由 Functional Requirement 向上找到 QML 中 Requirement 項的「參照 ISO 標準」(ISO Standard Reference)的屬性,以提供檢索出 ISO 標準或品質系統文件的相關文件內是否有針對設計文件的內容與描述頒訂相關之作業程序與格式規定,以稽核此文件內容的完整性。標籤處理技術支援多重觀點的(Functional Requirement)時,本輔助工具可由 Functional Requirement 中的追溯屬性(attribute traceback)提供追溯至「規格」與「程式碼撰寫」的相關文件部份。而稽核者點選檢視此標籤時,本工具則由 Functional Requirement 向上找到 QML 中

```

<!ELEMENT QML -- (ConceptStructure | BasicStructure)>
<!ELEMENT ConceptStructure -- (DefectLifeCycle & ProductLifeCycle) |(DefectLifeCycle|
  ProductLifeCycle)>
<!ELEMENT DefectLifeCycle -- (DefectPrevention | DefectDetection | DefectRemoval)>
<!ELEMENT DefectPrevention -- (Techniques | Experience | Resource)>
<!ELEMENT DefectDetection -- (Techniques | Experience | Resource)>
<!ELEMENT DefectRemoval -- (Techniques | Experience | Resource)>
<!ELEMENT ProductLifeCycle -- (Contract | DevelopmentPlan | Requirement | Design | Coding |
  Test | Accept | OperationMaintenance | SupportingActivities )>
<!ELEMENT Contract -- (Plan, CustomerRequirement, FeasibilityEvaluation,
  SubmitProposal, ContractSignature, ContractUnderControl, AssessmentOfSubcontractor,
  Verification)>
<!ELEMENT Plan -- (Scope & Environment & Reference & Objectives & Glossary & Schedules|
  Activity & Resource & Responsibility & Warrant? & Cost? & ChangeProcedure? &
  QualityControlMeasures? & WisdomPropertyRight? & AccessTest? & RiskAnalysis?)>
<!ELEMENT Verification -- Plan, (Review| Test | FormalAnalysis), Records>
<!ELEMENT Review -- (TechnicalReview | ManagementReview | JointReview |
  InspectionWalkThrough) & Checklist & Meeting>
.....

<!ELEMENT Requirement -- (FunctionalRequirement+ & NonfunctionalRequirement* &
  Verification)>
<!ELEMENT FunctionalRequirement -- (Input, Process, Output)>
<!ELEMENT NonfunctionalRequirement -- (PerformanceReq|DesignConstraint|InterfaceReq|
  ResourceReq) & (SecurityReq?, SafetyReq?, IntegrityReq?, ReliabilityReq?)>
<!ELEMENT Design -- (SoftwareArchitecture & DesignUnit+ & Verification)>
<!ELEMENT DesignUnit -- (ModuleName & InputParameter & OutputParameter & Caller & Callee &
  PseudoCode & Algorithm)>
.....:

<!Attlist Functional Requirement traceback ID #Required>
<!Attlist Functional Requirement id ID #Implied>
<!Attlist Functional Requirement dependency ID #Required>
<!Attlist Functional Requirement isoref ID #Required>
--Nonfunctional Requirement, Design unit and module have the same attribute.
.....

<!ELEMENT BasicStructure -- (ISO9001 & ISO90003 & QualityManual & Procedure & WorkInstruction
  & FormRecord & SoftwareProject)
<!ELEMENT ISO9001 -- (ManagementResponsibility & QualitySystem & ContractReview & .....)>
<!ELEMENT ISO90003 -- (ManagementResponsibility & QualitySystem & .....)>
<!ELEMENT QualityManual -- (ManagementResponsibility & QualitySystem & ContractReview & ...)>
<!ELEMENT Procedure -- (ContractReviewProcedure & IQAProcedure & ServiceProcedure.....)>
<!ELEMENT WorkInstruction -- (ProjDocControl|WI & QSDocControl|WI & .....)>
<!ELEMENT FormRecord -- (ContractReviewForm & CustomerReviewValForm & .....)>
<!ELEMENT SoftwareProject -- (SoftwareDevPlan & Contract & .....)>

```

圖 3 QML 以 SGML 表達

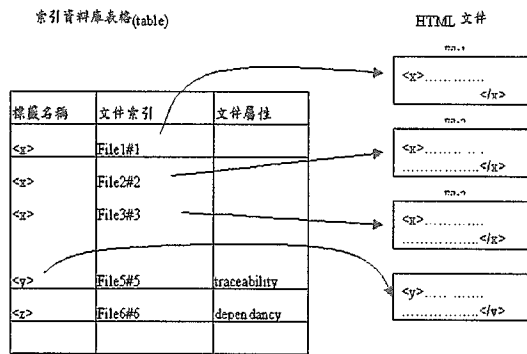


圖 4 標籤鏈結資料結構示意圖

Requirement 項的「參照 ISO 標準」(ISO Standard Reference)的屬性，以提供檢索出 ISO 標準或品質系統文件的相關文件內是否有針對設計文件的內容與描述須訂相關之作業程序與格式規定，以稽核此文件內容的完整性。標籤處理技術支援多重觀點的功能，將電子文件技術提昇至具有彈性，有部份智慧的應用層次。

4 品質標記語言的應用:ISO 認證分級

我們所發展的品質標記語言及標籤處理技術可應用於下列兩方面：(1) 輔助 ISO 認證的分級 (2) 輔助一般 ISO 的稽核及專案的維護。後者可以使用不同觀點的功能，而且其導航圖不需用到缺失生命週期，可以簡化為十字形的導航圖。

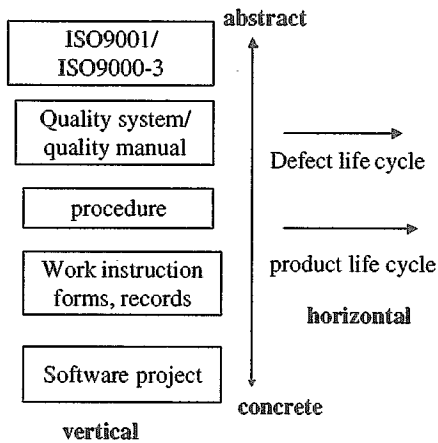


圖 5 ISO 品質系統縱向及橫向邏輯結構

ISO 9001 認證制度，只有認證『通過』或『不通過』，對於通過認證的軟體廠商，並無細分其優劣層級，無 CMM(Capability Maturity Model)[10]般的分級制度，使軟體廠商有明顯的改進程序(improvement process)可以依循，這是 ISO 常為人詬病的方面。應用我們所提出的「缺失生命週期」與「產品生命週期」的概念標籤，可檢索出決定 ISO 分級的相關資訊，輔助分級制度之推行。為了實施 ISO 認證分級，我們訂定了 ISO 分級評估項目與評分量化標準，並且運用雙十字導航圖來檢索決定分級的資訊，可以客觀地給與軟體廠商在軟體發展能力上的分級。

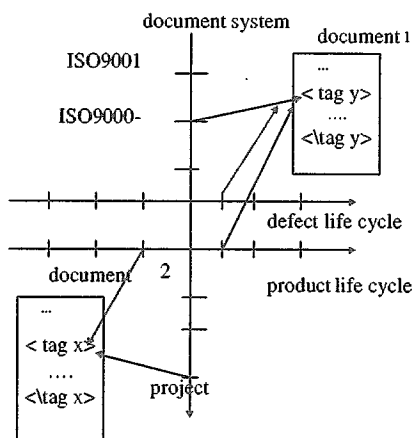


圖 6 標籤鏈結功能示意圖

部份我們所提出的評估項目如表 2 所示，每項目又包含數個子項目，涉及程序、計畫、技術、資源與經驗等，例如以評分項目的第一項為例：軟體專案發展階段用什麼方法或技術來偵測問題與缺點？針對本項評估所列的第一小項，經由雙十字概念導航圖的引導，可檢索軟體專案發展階段中與「缺失偵測」下面的「技術」標籤所標註的文件部份。同時我們亦可合併「缺失生命週期」與「產品生命週期」來檢索個別階段的問題偵測措施。第二小項：對於此作業，是否有訂定相關的作業程序或工作指導書？可檢索「缺失偵測」所標註的部份，第三小項：是否有足夠的資源、時間、工具以支援此項活動？可由「缺失偵測」下面的「資源」標籤來萃取相關的資訊，第四、五小項：參與本項專案活動人員，是否有足夠的經驗？組織是否有類似的專案活動經驗？由「缺失偵測」下面的「經驗」標籤來讀取與個人或組織相關的經驗資訊。

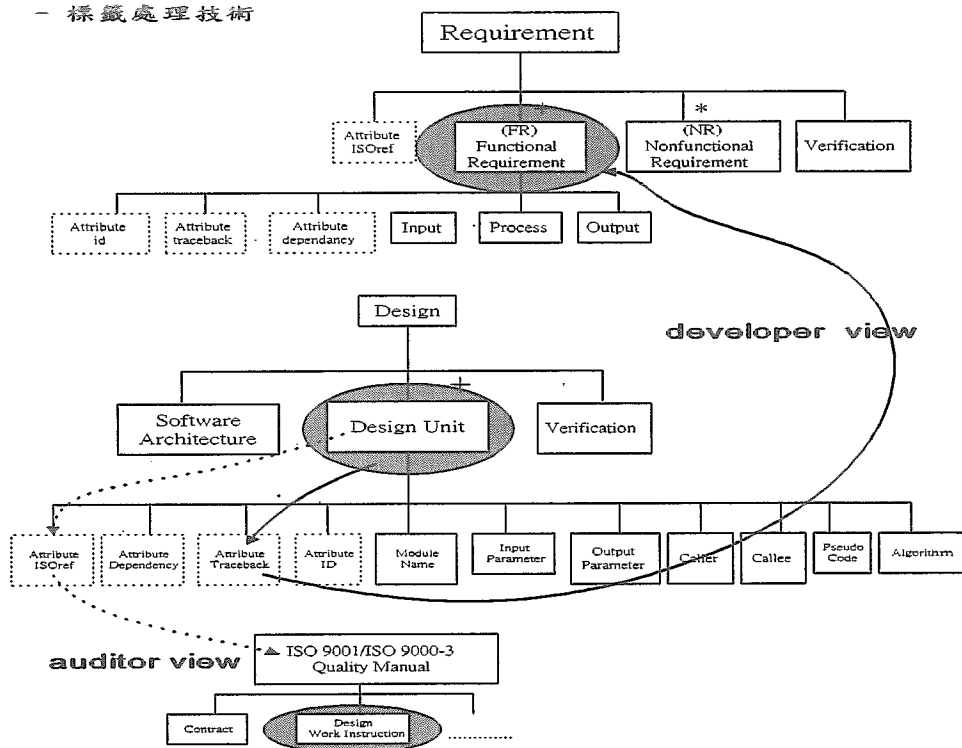


圖 7 多重觀點標籤處理技術示意圖

經由雙十字概念導航圖的引導可檢索出與評估項目相關的文件部份，再依據各評估項目影響軟體品質的強與弱，配合各評估項目的權重參數(weight)，最後以權重參數計算所得的總和來評斷通過 ISO 認證的軟體廠商的優劣層級(level)，以評估軟體廠商的發展能力成熟度等級。量化配分的方法可如下：

1. 每子項分數

- a. (此題不適用不計分)
- b. 較差(40分)
- c. 尚可(70分)
- d. 優(90分)
- e. 特優(100分)

2. 計算各評估項目之平均分數：

以 n 項評估項目分別計算平均分數

$$AS_j = Q_j / N_j \quad (j=1 \text{ 至 } n)$$

AS_j 評估項目 j 之平均分數

Q_j 本評估項目得分總和

N_j 本評估項目題總數

3. 計算總平均分數

$$TA = \sum (AS_j \times W_j) / \sum W_j \quad (j=1 \text{ 至 } n)$$

TA : 總平均分數

AS_j : 第 j 項的平均分數

W_j : 第 j 項的加權值(weight)

4. 軟體發展能力評分值分級

- a. 等級一(最差)：0~50
- b. 等級二：50~70
- c. 等級三：70~90
- d. 等級四(最佳)：90~100

5 電腦輔助工具

我們針對上述 QML 及其應用發展了一套電腦輔助工具，本工具採用 Microsoft Visual Basic 開發，資料庫使用 Microsoft Access 97 的 mdb 資料格式，各類相關文件採用全球資訊網中通用的 HTML 格式，在 HTML 規格方面包含了 HTML 4.0 中的階層樣式表(CSS Cascading Style Sheet)。此工具分為四個部分，第一部份是將文件納入資料庫的工具，第二部份為文件閱讀工具，第三部份為輔助 ISO 認證分級制度工具，第四部分為稽核品質系統文件輔助記錄工具。

閱讀工具(如圖 8 所示)提供使用者「基礎模式」(basic structure)及「概念導航圖模式」來閱讀文件。「基礎模式」提供傳統 ISO 品質系統文件章節內容之引導閱讀。閱讀者在選取「概念導航圖模式」時可點取概念導航圖以檢索要檢視的文件內容。此瀏覽模式又提供進階的功能以及多重觀點瀏覽功能，如第三節所述。我們發展的多重觀點功能，在應用實作上分為兩個模式，即發展者模式(developer mode)與稽核者模式(auditor mode)。此功能對發展者提供電子文件追溯的功能，對稽核者提供 ISO 相關標準參考檢索功能。本工具亦提供 ISO 認證分級評估之輔助記錄及稽核記錄工具。

6 結論

本文提出品質標記語言的設計及應用，其主要的目的為增進電子文件的閱讀效率，並節省維護超鏈結的人力。本研究提出的方法步驟如下：

(1)首先萃取應用領域的語意概念及其結構。本研究以 ISO 軟體品質文件系統為應用案例，提出以軟體「產品生命週期」與「缺失生命週期」來表達 ISO 文件系統深層的語意概念及精神內涵。

(2)以上述概念結構，發展領域導向標記語言。本研究在此階段發展了品質標記語言，以標記此領域的主要概念及其結構，以供進一步的處理應用。

(3)接著此文件系統的文件依上述標記語言加以標註，並建立相關工具及標籤處理技術。在工具導航功能上我們提出一雙十字邏輯導航結構。雙十字形邏輯結構的縱軸表達文件系統中的抽象與具體、約束與建議、目標與方法等關聯關係，橫軸以產品生命週期(product life cycle)與缺失生命週期(defect life cycle)來表達文件內的目的、主題、或因果關係，雙十字結構適用於具有邏輯結構的複雜系統。

(4)接著發展標籤處理技術，將概念的標籤轉成地址性的超鏈結，以節省建構與維護超鏈結的人力，並且可發展進階的標籤處理技術，以提昇電子文件的閱讀檢索效率。本研究發展的進階標籤處理技術，包括不同的使用者在使用相同的標籤時，可以用多重的觀點(multiple views)加以瀏覽。

(5)在品質標記語言(QML)的應用上，本文提出 ISO 分級制度的構想及方法，並利用工具輔助 QML 文件擷取分級資訊，此構想與方法可改善現行 ISO 二分法(通過或不通過)之缺點。

未來的發展可以加入專家系統與人工智慧，以更一步提昇電子文件閱讀效率。

誌謝

本研究承國科會之贊助，計劃編號 NSC 88-2213-E-155-002.

項次	評分項目
(1)	1. 軟體專案發展階段用什麼方法或技術來偵測問題與缺點? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?
(2)	1. 軟體專案發展階段用什麼方法或技術來完成更正行動? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?
(3)	1. 軟體專案是否有訂定風險管理? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?
(4)	1. 軟體專案是否採用 Formal development (Structure/CO Methodology、Requirement Verification... etc.)? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?
(5)	1. 軟體專案是否有採用 prototyping、scenario tree、joint review 等技術, 以確定需求? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?
(6)	1. 程式碼撰寫是否要求程式撰寫標準(coding standard), 以確保程式之可維護性? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?
(7)	1. 是否有指派專人負責執行軟體測試, 以確保測試之獨立性與公正性 (IV&V)? 2. 對於此作業, 是否有訂定相關的作業程序或工作指導書? 3. 是否有足夠的資源、時間、工具以支援此項活動? 4. 參與本項專案活動人員, 是否有足夠的經驗? 5. 組織是否有類似的專案活動經驗?

表 3 ISO 認證評估的部份項目

參考文獻

- [1] ISO 9001:1994(e) Second Edition 1994-07-01 Quality Systems – Model for Quality Assurance in Design , Development , Production , Installation and Servicing.
- [2] ISO 9000-3 First edition Quality management and Quality Assurance Standards – Part 3 Guidelines for the Application of ISO 9001 to the Development , Supply and Maintenance of Software · 1991.
- [3] D. Connolly , J.Boask , Extensible Markup Language , <http://www.w3.org/xml/>.
- [4] P.Murra -Rust, "Chemical Markup Language," CML,Version 1.0, <http://www.venus.co.uk/omf/cml/doc/tutorial.html>, Jan 1997.
- [5] P.Ion and Miner, "Mathematical Markup Language," W3CWorking Draft , <http://www.w3.org/pub/www/TR/wdmath/>, May1997.
- [6] World Wide Web Consortium(W3C) , "HTML 4.0 Reference Specification," <http://www.w3.org/MarkUp/>.
- [7] C.F.Goldfarb,*The SGML Handbook*,Oxford Univ.Press,1990.
- [8] CALS/IET 的運用實例, <http://www.csd.org.tw/html/0005.htm>.
- [9] Trellix, <http://www.trellix.com/>.
- [10] Mark C. Paulk, et al. ,"Key Practices of the Capability Maturity Model ,"Version 1,1 Technical Report CMU/SEI-93-TR-025 ESC-TR-93-178 February 1993.
- [11] E. van Herwijnen , *Practical SGML* , Kluwer Academic Publications , 1991.
- [12] 交談式電子技令漫談, http://www.asiatek.com.tw/guide/prd_5-6-3.htm.

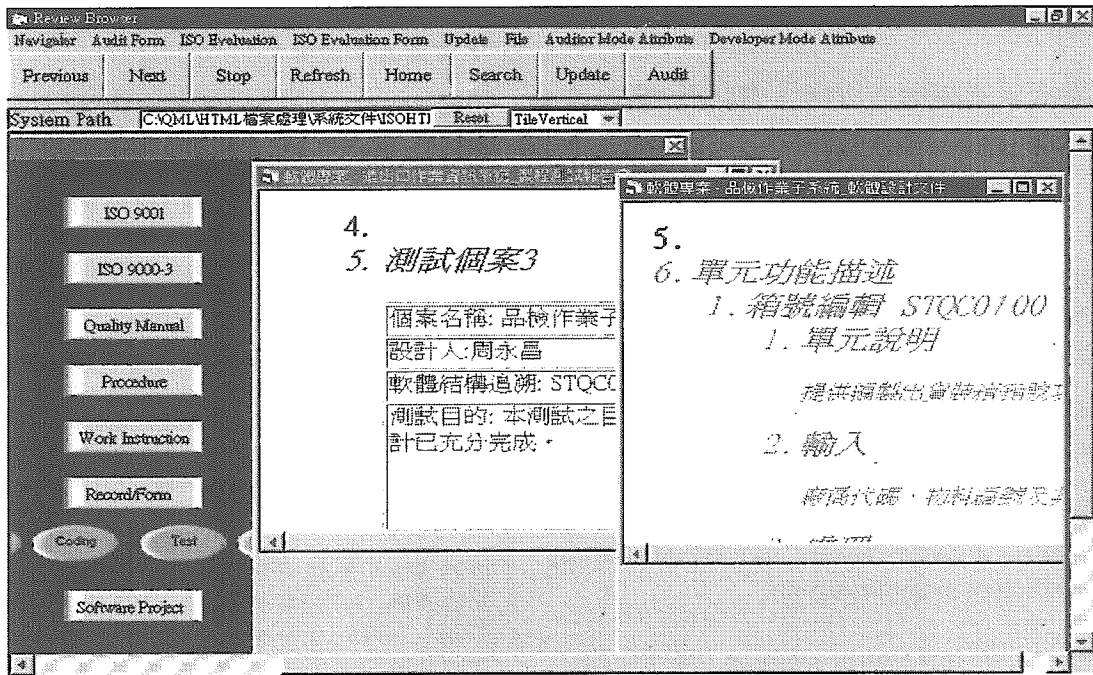


圖 8 閱讀工具