

行政院及所屬各機關出國報告
(出國類別：實習)

IOIS J2EE/J2ME研發平台實習

出國報告

服務機關：中華電信研究所

出國人職稱：助理研究員

姓名：黃文怡

出國地區：新加坡

出國期間：92年9月7日至92年9月20日

報告日期：92年11月18日

H6/009204151

系統識別號:C09204151

公務出國報告提要

頁數: 20 含附件: 否

報告名稱:

IOIS J2EE/J2ME 研發平台實習

主辦機關:

中華電信研究所

聯絡人/電話:

楊學文/03-4244218

出國人員:

黃文怡 中華電信研究所 92860專案研究計畫 助理研究員

出國類別: 實習

出國地區: 新加坡

出國期間: 民國 92 年 09 月 07 日 -民國 92 年 09 月 20 日

報告日期: 民國 92 年 11 月 18 日

分類號/目: H6/電信 /

關鍵詞: IOIS,J2EE,J2ME,研發平台

內容摘要: 本計畫(專案860計畫)自2001年起協助總公司開發IOIS線路供裝維運管理系統(eLEAMIS)，系統已經陸續於北中南三區上線(中區七月份全區上線；南區及北區於十一月全區上線完畢)，因應總公司之政策，系統間的資訊整合、元件的再利用扮演系統開發的重要角色，此外，網頁的個人化(Personalization)也是未來程式開發的重點。本次實習的BEA WebLogic Platform及BEA WebLogic Workshop 是目前市場佔有率相當高的J2EE系統開發平台，提供軟體開發人員快速建置客戶導向及具有個性化和用戶特徵的使用者介面及內容的網站，並可透過J2EE元件整合不同系統的資訊，以提供使用者的需求。藉由此次研習，學習建置具有個性化的網站及學習J2EE相關技術，以提升未來程式開發與系統整合所需的技能。

本文電子檔已上傳至出國報告資訊網

摘 要

本計畫(專案860計畫)自2001年起協助總公司開發IOIS線路供裝維運管理系統(eLEAMIS)，系統已經陸續於北中南三區上線(中區七月份全區上線；南區及北區於十一月全區上線完畢)，因應總公司之政策，系統間的資訊整合、元件的再利用扮演系統開發的重要角色，此外，網頁的個人化(Personalization)也是未來程式開發的重點。本次實習的BEA WebLogic Platform及BEA WebLogic Workshop 是目前市場佔有率相當高的J2EE系統開發平台，提供軟體開發人員快速建置客戶導向及具有個性化和用戶特徵的使用者介面及內容的網站，並可透過J2EE元件整合不同系統的資訊，以提供使用者的需求。藉由此次研習，學習建置具有個性化的網站及學習J2EE相關技術，以提升未來程式開發與系統整合所需的技能。

IOIS J2EE/J2ME 研發平台實習

出國報告書

1. 目的

本計畫(專案 860 計畫)自 2001 年起協助總公司開發 IOIS 線路供裝維運管理系統(eLEAMIS)，系統已經陸續於北中南三區上線(中區七月份全區上線；南區及北區於十一月全區上線完畢)，因應總公司之政策，系統間的資訊整合、元件的再利用扮演系統開發中的重要角色，此外，網頁的個人化(Personalization)也是未來程式開發的重點。BEA WebLogic Platform 及 BEA WebLogic Workshop 是目前市場佔有率相當高的 J2EE 系統開發平台，提供軟體開發人員快速建置客戶導向及具有個性化和用戶特徵的使用者介面及內容的網站，並可透過 J2EE 元件整合不同系統的資訊，以提供使用者的需求。職奉准至新加坡 BEA 公司參加「IOIS J2EE/J2ME 研發平台—BEA WebLogic Platform」研習課程，研習內容包括 J2EE 架構、用 BEA WebLogic Portal 及 Portlets (包括 BEA WebLogic Personalization Server)開發和部署於 Portal 的電子商務應用等相關技術。期望藉由此次研習，學習建置具有個性化的網站及學習 J2EE 相關技術，以提升未來程式開發與系統整合所需的技能。

2. 過程

行程概要：

整個行程從 92 年 09 月 07 日出發，至 92 年 09 月 20 日返國，共計 14 天。其受訓過程如下表：

日期	主題
09/07	啟程
09/08~09/12	IOIS J2EE 研發平台研習
09/12~09/14	整理文件
09/15~09/19	BEA WebLogic Workshop 研習
09/20	回程

實習內容：

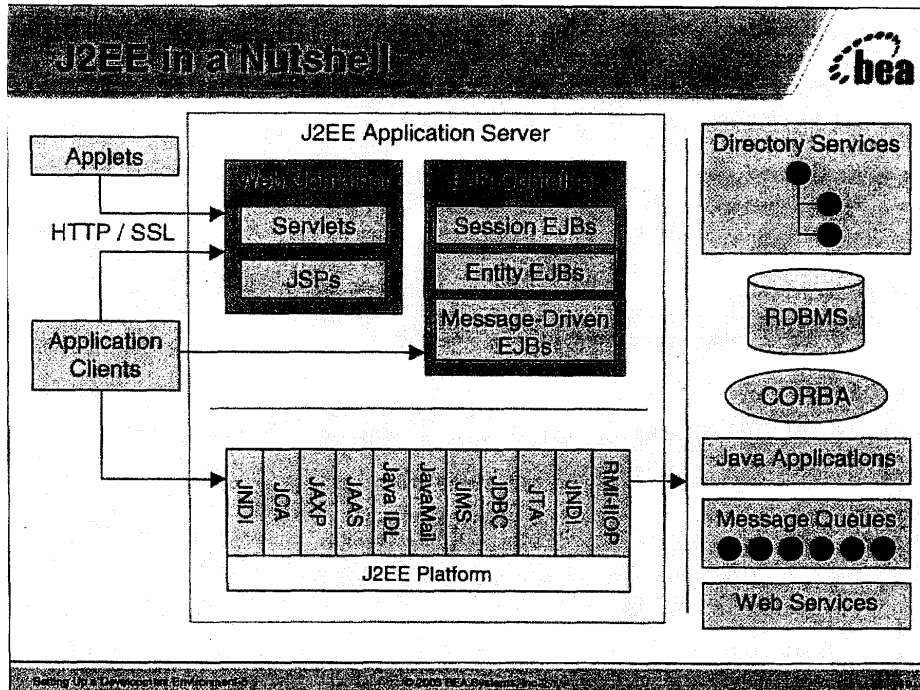
2.1 J2EE 技術簡介

Java 2 Platform, Enterprise Edition (簡稱 J2EE) 是目前業界的一種標準，它的目的是用來簡化且規範應用系統的開發與部署，進而提高可移植性、交易安全與再利用價值。它被設計成專門用來解決多層式(multi-tier)企業解決方案的開發、部署以及管理上的問題，降低應用系統開發的成本及複雜度。

J2EE 主要是由昇陽與 IBM 等廠商協同業界共同擬定而成的技術規範，它的架構如圖一，架構內的元件、服務架構及技術層次，均有共通的標準及規格，在 J2EE 的架構下，開發人員依循業界制定的共同規範進行應用系統開發；而所有依循 J2EE 架構的不同應用系統之間，因為共通的標準存在良好的相容性，解決過去企業後端使用的資訊產品彼此之間無法相容，導致企業內部(或外部)難以互通的窘境。

J2EE 主要的元件如下：

- (1) 應用程式元件(Application components)—包含Servlets及JSP網頁、Applets、Application clients、EJBs。
- (2) 應用程式容器(Container)—應用程式容器提供應用程式元件執行時的環境。
- (3) 資源管理驅動元件(Resource manager drivers)—資源管理驅動元件是一個系統層面的軟體元件，透過它可以與外部的資源互相連結溝通。
- (4) 資料庫(Database)—J2EE平台中包含資料庫，應用程式元件、EJBs、客戶端應用元件可透過JDBC存取該資料庫的資料。



圖一 J2EE 架構圖

2.1.1 J2EE 服務

J2EE平台包含下列的服務，J2EE提供的各項服務背後均有相對應的API支援，透過這些APIs 達到系統的各项需求：

- (1) **HTTP**—用戶端程式透過HTTP通訊協定與Web Server連結取得用戶所須資訊，HTTP客戶端的API 定義在java.net 類別中；HTTP Server端的API 則透過servlet及JSP介面定義。
- (2) **HTTPS**— HTTPS是利用HTTP 通訊協定及網路安全SSL通訊協定定義出來的， 透過與HTTP相同的用戶端及Server端的API支援此項服務。
- (3) **Java Transaction API (JTA)**—應用程式容器透過JTA提供交易服務。
- (4) **RMI-IIOP**—遠端程序呼叫 (Remote Procedure Call,RMI)係將通訊細節抽象化至程序呼叫的層次，透過客戶端的 stub 與遠端物件取得聯繫，RMI-IIOP API 是由支援RMI 通訊協定的API與支援CORBA IIOP通訊協定的API 組合而成，用戶端程式可以直接透過RMI-IIOP連結 EJB Container 以取得 Middle-Tier所提供的服務。

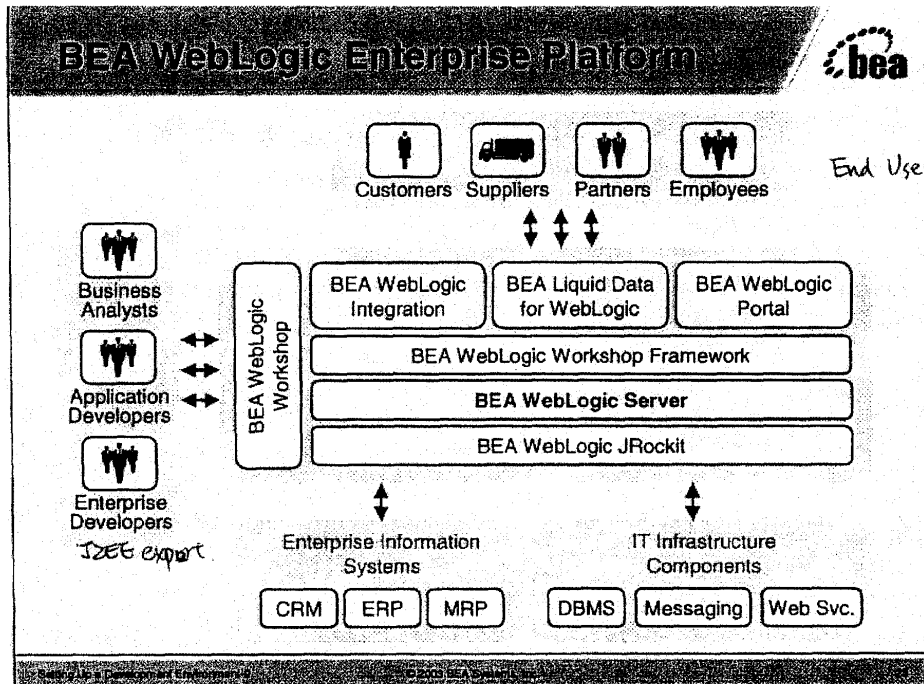
- (5) **JavaIDL**—JavaIDL 允許 J2EE 應用程式元件透過IIOP通訊格式啟動外部的CORBA物件。
- (6) **JDBC**—JDBC API 提供與資料庫系統連接的服務。
- (7) **Java Message Service (JMS)**—JMS是一套標準企業 API,用來作為 Java 程式和 MOM (Message-Oriented Middleware, 訊息導向中介軟體)之間的介面。而訊息的內容可以是任何東西,包括文字、Java 物件、XML 文件...等。
- (8) **Java Naming and Directory Interface (JNDI)** —JNDI是一套標準企業 API,它是用來對環境中資源命名,以及從此命名取得該資源的一項技術。
- (9) **JavaMail**—J2EE平台提供 JavaMail API 及JavaMail service provider,透過它們讓應用程式元件可以寄送網路信件。

2.2 BEA WebLogic 企業研發平台

BEA WebLogic 研發平台提供一個高度整合、高可靠度且可高度執行的解決方案，它結合一般的應用程式開發介面的優點，加入簡化的、可延伸的發展環境及管理框架。

BEA WebLogic 研發平台的應用程式架構(如圖二)，包含：

- (1) 一組整合的技術框架，這些框架提供程式發展者開發系統時的全面性部署環境，將系統各項需求透過各自問題導向的起點延申為整個系統的架構，在元件在利用及 J2EE 標準下達到系統多樣的需求。
- (2) 一個單一的執行期框架(run-time framework)，為企業整個軟體平台開發提供了介於開發者與複雜 J2EE 系統基礎結構之間的抽象層。開發者可免於處理耗時的、API 層次的基礎結構程式碼、元件組態及部署的細節，而自由地使用視覺化的開發環境，在必要的部分撰寫程序性的 Java 程式碼，並以簡明、宣告的註解存取進階的功能。WebLogic Workshop 執行期框架便是透過上述方式，產生標準的 EJB, JMS 以及 JDBC 程式碼，並以此程序管理與 J2EE 結構相關的設計及實作要點，故其應用程式之實做，皆立基於一可靠的、具規模度且安全的企業級架構之上。
- (3) 一個簡單但完整的管理環境，讓程式發展人員及系統管理透過此環境很快地瞭解其分散式及多層次系統中的各項元件。
- (4) 符合 J2EE 標準，提供高可靠度,高度相容、可延申、高度執行基礎的開發平台。..



圖二 BEA WebLogic 企業研發平台

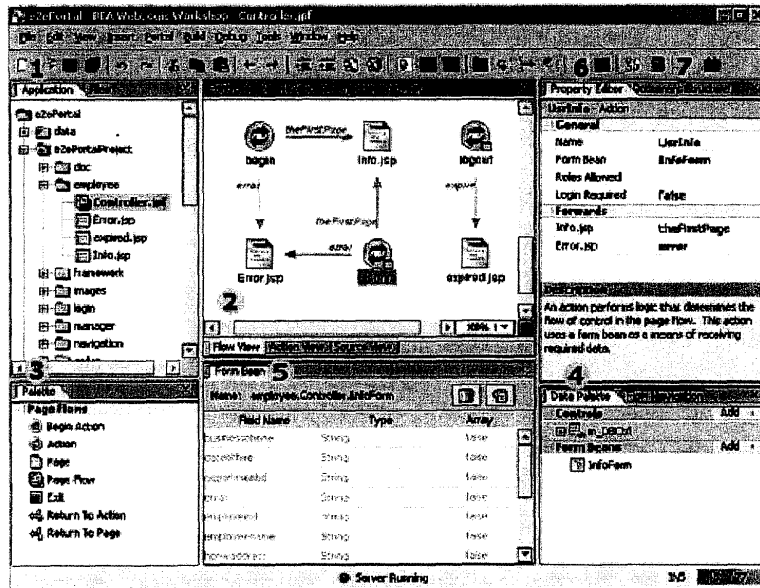
2.3 WebLogic Workshop 簡介

WebLogic Workshop透過視覺化的開發環境(如圖三)開發Portal應用程式，它的整合式開發環境包括：視覺化的開發元件(visual designers)、瀏覽視窗(navigation windows)、執行期框架(run-time framework) 以及 Java 控制項(Java Controls)。

如圖三，WebLogic Workshop開發環境中包括：

- (1) 應用程式視窗(Application Window)—顯示應用程式開發者開發的所有的檔案及資源。
- (2) 流程管理視框(Flow View Canvas)—流程管理框架內含一個圖形化的介面，讓使用者可以透過拖拉方式輕鬆地建立及修改 Java 網頁流程(Page Flows)。Java 網頁流程是相關的 JSPs 及 Java 控制元件(controller)的組合，流程中將顯示網頁間的互動關係。
- (3) 套件視窗(Palette Window)—顯示所有的程式開發者可利用的套件。
- (4) 物件視窗(Data Palette)—提供使用者方便的去取用可利用的控制元件和函數。

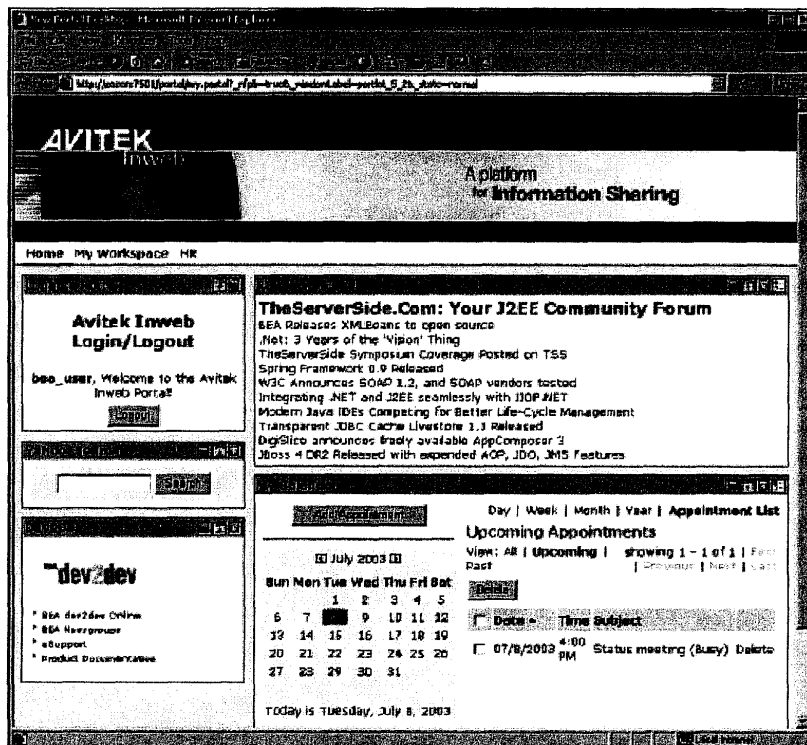
- (5) Form Bean 編輯框架—程式設計者可以在此框架內檢視及設定所選定的 Form Bean 中的控制函數的屬性。
- (6) 屬性編輯框架—程式設計者可以在此框架內檢視及設定所選定的物件的屬性。
- (7) 文件架構視窗—文件架構視窗顯示相關的文件及其檔案架構。



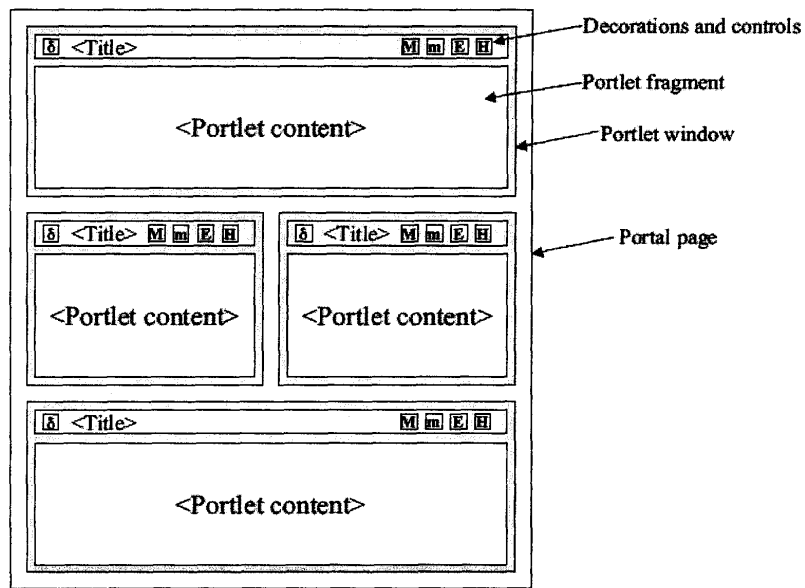
圖三 WebLogic Workshop 開發環境

2.3.1 Portals 及 Portlets

Portal是一個Web基礎的應用程式，如圖四，Portal提供一個可個人化、單一整合入口的網站，並解決單一登入(single sign on)的問題，其背後可透過J2EE各項標準整合不同主機或來源的資料，顯示於同一網頁中，Portal中可包含許多的Portlets(如圖五)，每一個Portlet提供一項功能，Portlets間可互相傳遞參數溝通以達到整合不同系統資訊的需求。Portal的個人化功能讓不同的客戶可以客制化他的網頁，納入其所需的Portlets，使其網頁只包含個人所需資訊，同樣地，開發人員也可透過Portal個人化的功能，客制化不同層級的使用者的網頁。



圖四 Portal 範例



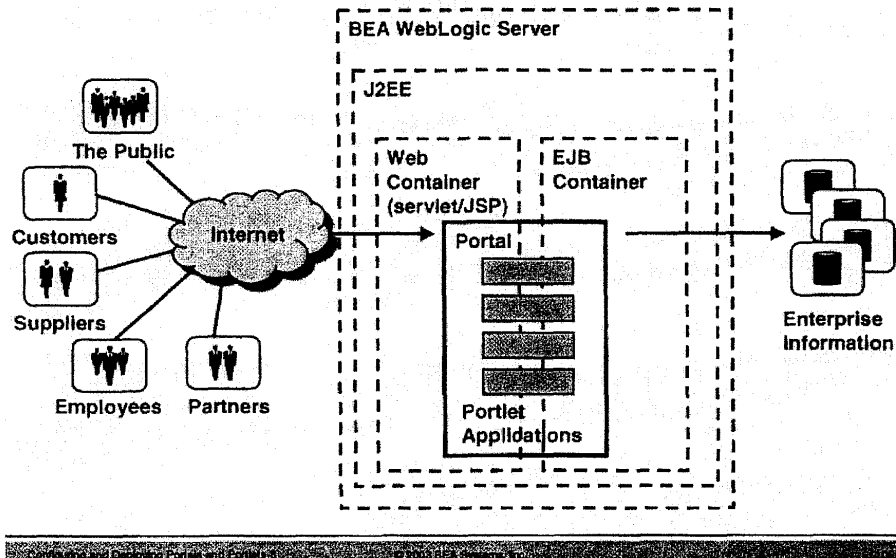
圖五 Portal 與 Portlets

Portlets 是以 Java 程式為基礎的 Web 元件，使用者可透過 Portlet 容器管理其 Portlets，Portlet 容器管理 portlet 的生命週期（啟動→處理 Request 及執行動作→結束）並呼叫相對應的 portlet 函式。

在 BEA Workshop 中，我們可以在不同的 portlets 間傳遞訊息，藉由一個 portlet 的變動觸發另一個 portlet 內容的更新，藉此開發人員可以整合各個已經存在的企業內部系統的資源，並可將這些系統的內容整合在一個網頁中。

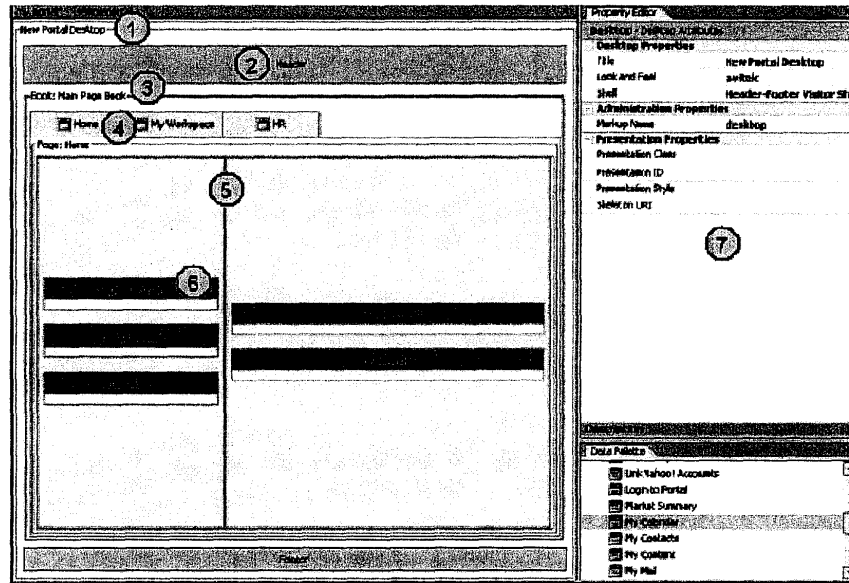
此外，BEA Workshop 中提供一組 portlet APIs 以達到企業應用系統所需的網站個人化、畫面部署及網路安全控管等需求。

BEA WebLogic Portal 就像一個 J2EE 平台—它提供符合 J2EE 標準的服務，它的架構如圖六，圖中說明我們透過 WebLogic Server 進行 BEA WebLogic Portal 的管理，客戶端(包含供應商、員工、客戶及合作夥伴等)透過 Internet 進入 WebLogic Server 內讀取 Portal 網頁，Portal 應用程式透過 EJB 容器存取資料庫及外部其他資源。



圖六 BEA WebLogic Portal 平台

2.3.2 Portal 設計視窗及 Portal 頁面



圖七 Portal 設計視窗及Portal 頁面

圖七顯示Portal 設計視窗，包含：

- 頁面

頁面(desktop)即整個Portal網頁的最外層，內部可包含其他的Portal組件，客戶端透過Internet連結至Web Server所見到的就是整個頁面。BEA WebLogic Portal透過兩個XML檔案- skins 及 skeletons 來控制頁面的整個外觀、顏色及配置，開發人員可以建置多組的skins及skeletons，Portal管理人員及客戶端可以選取其所需的skin及skeleton來改變Portal頁面的外觀及配置。

- 頁首及頁尾

頁首及頁尾是透過shell定義，shell是一個XML檔案，檔案中指名頁首及頁尾指向的JSP或HTML網頁(包含顏色、圖形及個人化內容)。如同頁面中的skins及skeletons，開發人員可以建置多組的shells，Portal管理人員及客戶端可以選取其所需的shell來改變Portal頁面的頁首及頁尾。

- **最上層書頁**

最上層書頁(The top-level book)，內部可包含其他的書頁及portlets，它定義出客戶端最原始瀏覽書頁風格，其下的不同書頁中您可以選擇不同的書頁風格。在屬性編輯視窗下，開發人員可以定義不同的書頁風格，而Portal管理人員及客戶端可以選取其所需的書頁風格。

- **書頁**

書頁(Pages and sub-books)用來容納portlets，在屬性編輯視窗下，開發人員可以定義不同的書頁風格，而Portal管理人員及客戶端可以選取其所需的書頁風格。

- **畫面配置**

畫面配置決定書頁及portlet的擺放位置，它經由一個XML檔案定義其網頁的切割方式。在屬性編輯視窗下，開發人員可以定義不同的畫面配置方式(例如:圖七中書頁切成左右兩個框架)，並決定每一個portlet的位置，而Portal管理人員及客戶端可以選取其所需的畫面配置方式。

- **Portlets**

Portlets是以Java程式為基礎的Web元件，它由開發人員開發來提供系統所需要的功能，透過WebLogic Workshop的Portlet精靈或Portlet設計視窗，開發人員可以輕易的產生所需的Portlet元件，再透過Portlet設計視窗及屬性編輯視窗，可以變更portlet的屬性及模式，這些資訊會被儲存在.portlet XML檔案中。

WebLogic Portal架構讓開發人員可以重複利用同一個portlet多次，而只要原始的portlet程式變更，所有利用到的相關畫面也會跟著變更，如此一來可以達到元件的再利用。

- **屬性編輯視窗**

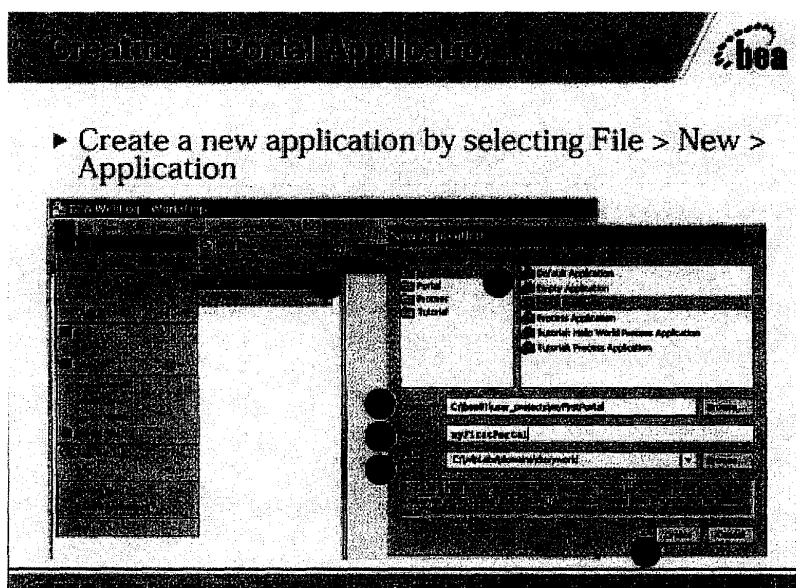
程式設計者可以在此框架內檢視及設定所選定的物件的屬性。

2.3.3 建置一個 Portal 網頁

除了符合一般傳統的整合式開發環境開發Portlets的特性外，WebLogic Portal 提供完整管理Portal生命週期的發展環境，並在WebLogic Workshop環境中可以很容易地建立一個Portal網頁，其步驟如下：

(一)、建立一個Portal應用程式

- (1) 在WebLogic Workshop環境中選取File>New>Application
- (2) 選取Protal Application
- (3) 視窗中顯示Portal應用程式將被建立的目錄.
- (4) 寫下這個Portal應用程式的名稱，這個名稱會被顯示在上述的目錄下.
- (5) 視窗中的Server欄位顯示這個Portal 應用程式建立後基本上會被部署的Server
- (6) 按下”create”按鈕產生Portal應用程式

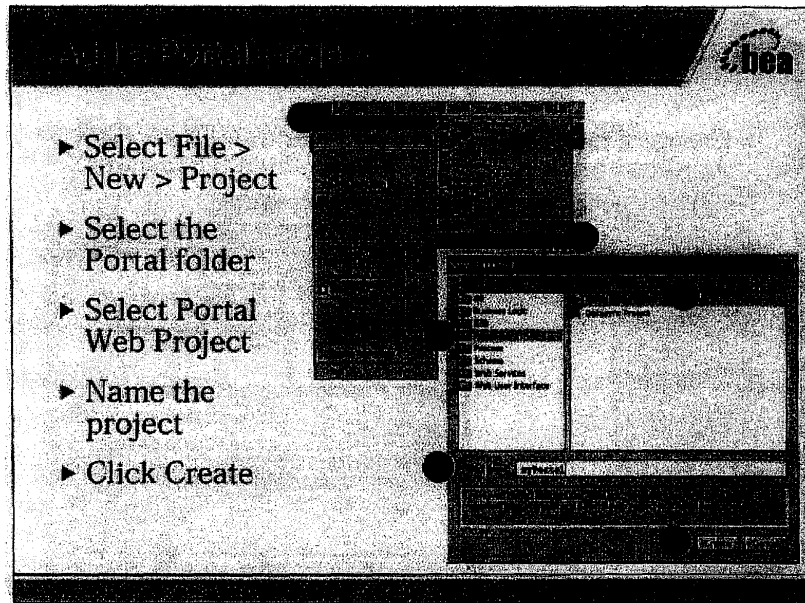


圖八

(二)、建立一個Portal專案

- (1) 在WebLogic Workshop環境中選取File>New

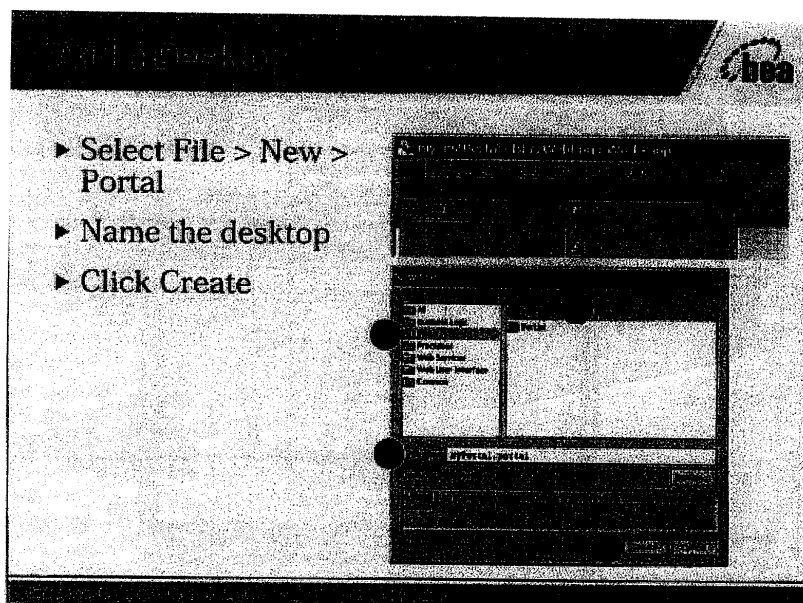
- (2) 選取Project
- (3) 選取Protal 目錄
- (4) 選取Portal Web Project
- (5) 寫下這個Portal專案的名稱
- (6) 按下”create”按鈕產生Portal專案



圖九

(三)、加入一個頁面

- (1) 在WebLogic Workshop環境中選取File>New>Portal
- (2) 選取Protal 目錄
- (3) 選取Porta 檔案
- (4) 寫下這個Portal檔案的名稱
- (5) 按下”create”按鈕產生Portal檔案



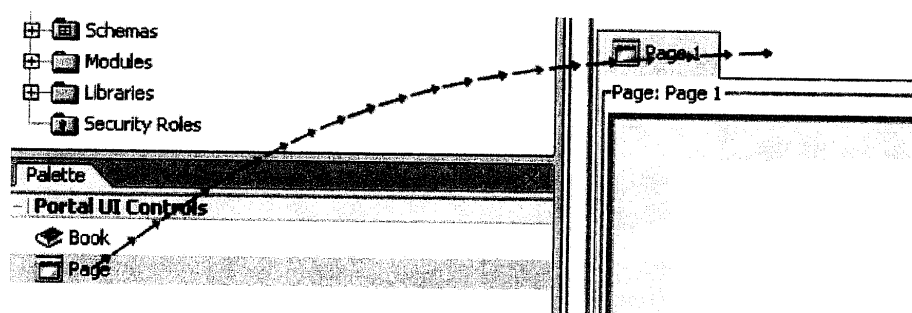
圖十

建立完Portal應用程式後，我們可以

- 加入一個書頁(Page)到Portal中
- 加入一個已經存在的portlet到Portal中

2.3.3.1 加入一個書頁到 Portal

(1) 在左下方的物件視窗中，將書頁物件拖拉到Portal頁面中。



圖十一

(2) 點選到新的書頁，在屬性編輯視窗中修改書頁的抬頭屬性。

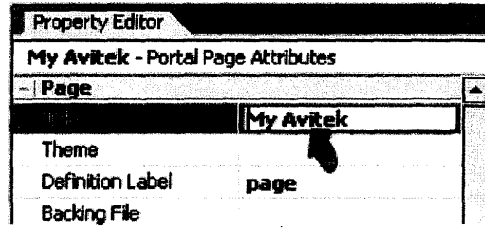
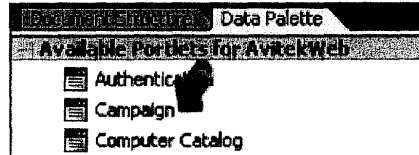


Figure 2.3.4.1. Customizing the Page

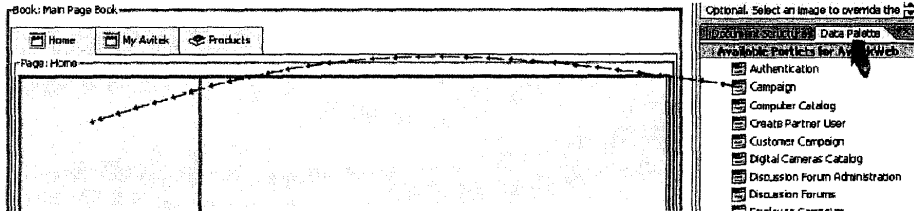
圖十二

2.3.3.2 加入一個已經存在的 portlet 到 Portal

- (1) 在Portal編輯畫面上，點選要將portlet加入的頁面。
- (2) 在物件視窗中，將portlet拖拉到該頁面中。



圖十三



圖十四

3. 心得:

職自進入中華電信研究所以來，即從事 Web Application Server 端的程式開發，為因應全省北中南各區不同的作業流程及使用者介面操作習性，還有使用者對其它系統資料的查詢或整合需求，往往要花費很大的力氣進行整合、開發及配置；一個好的開發平台除了在開發階段能幫助軟體開發人員能更快速的完成程式開發，更須要提供良好的 J2EE 各項元件整合環境，才能達到元件再利用以提高軟體開發人員的效率。

BEA WebLogic Platform 及 BEA WebLogic Workshop 提供軟體開發人員方便的程式開發介面，可以快速建置具有個性化和用戶特徵的使用者介面的網站，並可整合 J2EE / J2ME EJB,JMS,JDBC 等元件整合不同系統的資訊，以滿足使用者的需求。

J2EE(J2ME)所牽涉到的技術相當廣泛，舉凡 JDBC、EJBs、RMI/IIOP、JTA、JMS、JMX 等等，每一項技術都需要深入的了解才能有效的規劃系統的各項元件及程式架構，BEA WebLogic Workshop 提供使用者方便的程式開發介面，讓系統開發人員專注於 J2EE 各項技術的研究與企業邏輯的設計與實作上，才能開發出執行效能佳且兼具穩定性及擴充性的企業應用系統。

4. 建議

BEA WebLogic Platform 及 BEA WebLogic Workshop 提供軟體開發人員方便的程式開發介面，可以快速建置具有個性化和用戶特徵的使用者介面的網站，但其背後所牽涉的 J2EE 及 WEB 相關技術日新月異，還有系統繁複的企業邏輯，對系統開發人員是相當大的挑戰，這也使得團隊的合作更趨重要，透過團隊間的知識共享及資源分享才能共同創造最佳的系統，這個團隊含括系統開發人員、系統使用者及系統規劃人員，系統的開發與整合有賴大家的共同努力。

參考網頁:

- <http://edocs.bea.com>
- <http://dev2dev.bea.com>
- <http://education.bea.com>
- <http://newsgroups.bea.com>