

비주얼 인터데브를 이용한 웹 데이터베이스 통합 개발 사례

(A case study on the integrated development of web programming with database using visual interdev)

우 원 택 *

(Won-Taek Woo)

요 약 본 연구는 웹서버와 비주얼 인터데브를 각각 서버, 클라이언트의 개념적 틀로 사용해서 인터넷이나 인트라넷상에서 웹 애플리케이션을 개발하는 방법과 이 애플리케이션에 데이터베이스를 동적으로 통합하는 사례를 보임으로써 데이터베이스 정보를 다루는 파일과 웹 페이지를 다루는 파일을 각각 따로 개발하지 않고도 비주얼 인터데브를 이용하면 하나의 통합 환경 안에서 쉽고도 강력한 동적 웹데이터베이스를 개발할 수 있다는 사례를 보이고 향후 보다 적은 코딩 작업으로 보다 더 쉽게 웹 프로그램을 개발할 수 있는 기법과 원리를 제공하고 자 함에 그 목적이 있다.

핵심주제어 : 비주얼 인터데브, 웹프로그래밍, 웹데이터베이스 프로그래밍

Abstract The purpose of this study is to show a case integration of web programming with database using Visual InterDev as a conceptual client and Web Server as server. Through this case study, we can see how easily dynamic web database programming used in Internet and Intranet can be developed in one integrated design environment such as Visual InterDev eliminating the inconvenience of developing database file and web page separately. Some literature survey and experimental practices were done to provide future developers with the principles and methodologies of developing web database programming using Visual InterDev with much less coding efforts and more convenient and easy way compared to other web developing tools.

Key Words : Visual InterDev, Web Programming, Web Database Programming

1. 서 론

정보화 시대를 살아가는 지금 정보의 보고인 인터넷을 모르는 사람은 아마 없을 것이라 생각한다. 팔머(Palmer)는 향후 수년 동안 웹사이트의 수는

20배율로 증가할 것이며 2005년까지 웹 사이트의 수는 2억개가 되며 웹페이지의 수는 500억 페이지가 될 것으로 예상하고 있다.[1][2]

한편 국내의 인터넷 이용현황을 보면 1998년이후 계속 증가하여 4천7백만 국내 인구의 55% 이상이 인터넷에 접속하고 있으며[3] 국내 전자상거래 시장규모도 1998년 4억달러에서 2003년 96억달러로[4] 2010년에는 약 19조원으로 증가할 것으로 전

* 대구한의대학교 멀티미디어학부

망하고 있다.[5] 미국의 인터넷 이용현황을 보면 미국 성인의 인터넷 이용률이 인구의 63%에 도달하였고 2003년 하반기 가입자수 증가율은 20%에 달하고 있는 추세이다.[6] 캐나다의 전자상거래 규모는 2003년 191억 캐나다 달러로 전년대비 약 40%가 증가, 1990년 후반부터 꾸준히 증가하고 있으며 2003년 민간기업의 66%, 공공기업의 95.3%가 초고속 인터넷을 이용하고 있다.[7] 이와 같이 현대사회에서의 인터넷의 역할은 계속 증대되고 있으며 앞으로도 더욱 더 증대될 것으로 기대된다. 따라서 이러한 인터넷사회에서 생산활동을 하는 기업에서는 훌륭한 웹사이트가 필수적이므로 웹사이트 개발 요구가 더욱 증가할 것이다. 게다가 사회의 급진적인 발전속도에 부응하려면 웹사이트를 신속하게 개발할 수 있는 웹 개발 도구의 필요성도 증대될 것이며 또한 급변하는 컴퓨팅 기술환경 하에서 필요한 요소만을 선별하여 집중적인 기술을 익혀나가는 전략의 필요성도 높아지고 있다.[8] 이와같이 전자상거래의 발달과 웹의 개발환경이 더욱 복잡해 짐에 따라 웹정보시스템의 요건 분석과[9] 팀협업과[10] 전문화의 필요성도[11] 더욱 높아져 가고 있으며 이를 통합적으로 충족시켜 줄 수 있는 웹 툴들의 개발과 웹 운영체제[12], 효과적인 웹사이트 개발과 구축에 대한 연구도 활발히 이루어 지고 있다.[13][14][15] 이런 개발방법들의 일편으로 인터넷 사이트 빌더들이 소프트웨어 시장에 소개되고 있는데 마이크로소프트의 비주얼 인터데브도 이중의 하나이다. 마이크로소프트의 비주얼 인터데브는 웹사이트를 개발할 때 필요한 거의 모든 기능을 가지고 있고 CGI나 PERL과 같이 복잡한 프로그래밍 언어를 배우지 않고도 쉽게 다이나믹한 웹사이트를 개발할 수 있도록 도와준다.

본 연구에서는 현존하는 여러 인터넷 통합개발 도구들 중 마이크로소프트 인터데브를 이용하여 개발하는 사례를 보이고자 하는 것으로 웹서버와 비주얼 인터데브를 각각 서버, 클라이언트의 개념적 개발 툴로 사용해서 인터넷이나 인트라넷상에서 웹 애플리케이션을 개발하는 방법과 이 애플리케이션에 데이터베이스를 동적으로 통합하는 사례를 보임으로써 데이터베이스 정보를 다루는 파일과 웹 페이지를 다루는 파일을 각각 따로 개발하지 않고도 비주얼 인터데브를 이용하면 하나의 통합 환경 안에서 쉽고도 강력한 동적 웹 데이터 베

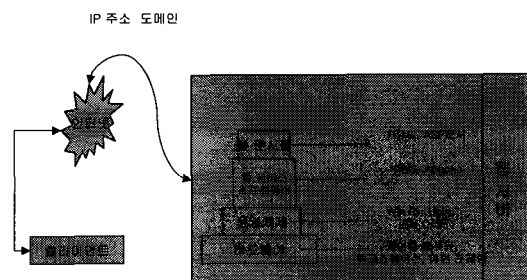
이스를 개발할 수 있다는 사례를 보이고 향후 보다 적은 코딩 작업으로 보다 더 쉽게 웹 프로그램을 개발할 수 있는 기법과 원리를 제공하고자 함에 그 목적이 있다.

2. 웹서버 구축환경과 비주얼 인터데브의 구조

비주얼 인터데브를 이용하여 동적 웹 데이터 베이스를 구축하기 위해서는 우선 웹사이트 구축환경과 구축절차 그리고 이와 관련된 애플리케이션과 비주얼 인터데브의 구조에 관한 지식이 필요하다.

2.1 웹사이트 구축환경

웹사이트 구축에 필요한 환경은 <그림 1>과 같으며 웹서비스 소프트웨어로는 Apache, IIS(Internet Information Server), PWS(Personal Web Server) 등이 있다. Apache는 Unix 환경에서 주로 사용되어 왔으나 최근에는 윈도우환경에서도 편리하게 사용할 수 있도록 개조되어 현재 Apache 2.0 버전이 나오고 있으며 IIS는 윈도우용 서버로 윈도우 XP 프로페셔널 이상의 버전에서 사용 가능하다. PWS는 실무적인 서버가 아닌 윈도우 9x 계열 또는 윈도우 NT 워크스테이션 버전의 운영체제에서 웹서버를 구축하여 운영하고자 하는 사용자를 겨냥한 제품이다. PWS는 IIS의 웹서버 기능을 대부분 포함하고 있으나 FTP기능이 지원되지 않고 연결가능 인원이 10명으로 제한되고 있는 점 이외에는 개인적인 용도, 소수집단의 정보공유, 웹 프로그래머 교육과정의 목적으로 사용할 때에는 비용이 저렴하고 구현하기 쉬운 Win 9x 시스템에 PWS를 탑재하여 사용하여도 무방하다.[16].



<그림 1> 웹사이트 구축에 필요한 환경

일반적으로 웹사이트 구축에 필요한 시스템 및 관련 소프트웨어를 살펴보면 다음과 같다.

<표 1> 웹사이트 구축에 필요한 애플리케이션

구분	개인용 시스템	서버급 시스템
운영체제	Win 9x 시스템	윈도우 NT, 2000, XP
웹서비스 소프트웨어	PWS	IIS 4.0, IIS 5.0
웹 문서 작성언어	HTML, ASP, PHP, C/C++, 스크립트언어	HTML, ASP, C/C++, 스크립트언어
데이터베이스	MS-Access	MS-SQL, Sybase, Oracle
문서편집기	Editor, Namo Web Editor, FrontPage, Visual InterDev 등	Editor, Namo Web Editor, Frontpage, Visual InterDev 등
파일송수신 서버(FTP)	Serv-U	IIS에 내장

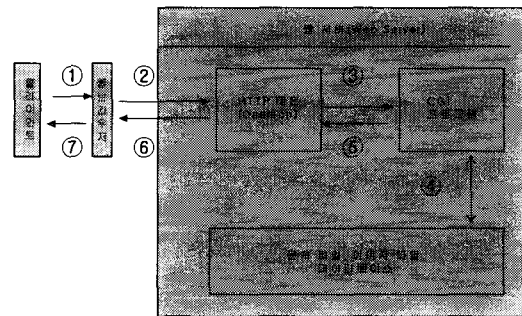
자료: 신회범, 김기용, 윤명영공저, 웹서버 구축 그리고 웹 프로그래밍, 생능출판사, 2001, p.25

웹사이트를 개발할 때 개발자가 좀 더 쉽고 편리하게 웹 페이지를 만들 수 있도록 해 주는 것이 웹 에디터인데 이 에디터에는 <표 1>과 같이 비주얼 인터데브, 나모 웹에디터, 프론트페이지 등이 있으며 자신에게 익숙한 에디터를 선택하여 사용하면 된다. ASP(Active Server Pages)를 이용하여 웹 문서를 개발하는 사람들은 비주얼 인터데브를 많이 사용한다.[16]

2.2 동적인 웹 문서 작성과 CGI

HTML 태그로만 작성된 웹 문서는 클라이언트와 서버간의 상호작용 없이 웹서버로부터 미리 정해진 내용이 일방적으로 전달되는 정적(Static)인 특성을 가지고 있다. 이러한 HTML, 웹 문서의 단점을 극복하고 동적인(Dynamic) 웹문서를 작성하기 위하여 등장한 기술이 CGI(Common Gateway Interface)이다. 즉 CGI는 HTML의 기능을 확장하기 위하여 외부 프로그램을 HTML과 연결시키는 시도로 프로그램 언어가 아닌 하나의 표준규약이다. CGI 프로그램 개발에 사용되는 언어는 제한되어 있지 않으며 웹서버 개발자는 자신에게 익숙한 C/C++, Perl, PHP, ASP, Java등 프로그램 언어를

사용하여 CGI를 개발할 수 있다. CGI언어로는 UNIX시스템에서 Apache 웹 서버를 사용하는 경우는 Perl 또는 C언어를 주로 사용하고 윈도우 NT 시스템 안에서 IIS웹서버를 사용하는 경우는 Perl이나 ASP를 주로 사용한다. 이하에서 CGI의 동작 원리와 그 순서를 보면 <그림 2>와 같다.



<그림 2> CGI의 동작원리

- ① 요청자, 즉 클라이언트는 웹 브라우저를 이용하여 입력양식(FORM)에 데이터를 입력하거나 혹은 기타방법으로 서버에 보낼 데이터를 준비한다.
- ② 브라우저는 입력받은 데이터를 TCP/IP를 통해 웹서버로 전송한다.
- ③ 웹 서버의 데몬(Daemon)은 이 데이터를 CGI환경변수(QUERY_STRING) 혹은 표준입력(Standard Input)으로 받아 CGI 프로그램에 넘긴다.
- ④ 여기서 CGI 프로그램은 서버 내에 있는 텍스트 파일, 또는 이미지 파일, 또는 데이터베이스에 접속하여 입출력 등의 일을 수행한 결과를 받는다.
- ⑤ CGI 프로그램은 작업 결과를 받아 적당한 MIME(Multipurpose Internet Mail Extension) Type으로 포매팅 한 후(즉 적당한 txt, html등의 문자유형으로 변환) 다시 HTTP 데몬으로 넘겨준다.
- ⑥ 데몬은 포매팅된 결과를 TCP/IP를 이용하여 브라우저(클라이언트)에게 넘겨준다.
- ⑦ 웹브라우저는 MIME Type에 따라 CGI프로그램의 결과를 브라우저에게 보여준다.

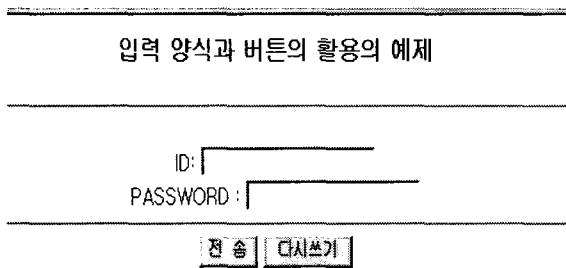
이와 같은 CGI 프로그램을 사용하기 위해서는 사용자로부터 입력(예, 회원 등록인 경우에는 회원의 신상정보를 입력)을 받아야 하는데, 이것을 가능하게 하는 것이 HTML의 <FORM>태그이다. HTML 웹 문서에서 CGI와 연계된 데이터 입출력

을 하기 위한 기본적인 입력양식과 CGI프로그램의 보기는 다음과 같다.

<표 2> 기본적인 입력양식과 CGI 프로그램

```
<HTML>
<HEAD><TITLE>입력 양식과 버튼의 활용</TITLE></HEAD>
<BODY><CENTER>
<H3>입력 양식과 버튼의 활용의 예제</H3>
<FORM>
<FORM METHOD="POST" ACTION="CGI">
  ID: <INPUT TYPE="TEXT" NAME="ID" MAXLENGTH="10"><BR>
  PASSWORD : <INPUT TYPE="PASSWORD" NAME="PASSWORD" MAXLENGTH="8"><BR>
  <INPUT TYPE="SUBMIT" VALUE="전 송">
  <INPUT TYPE="RESET" VALUE="다시쓰기">
</FORM></BODY>
</HTML>
```

<표 2>는 데이터를 입력받고 CGI 프로그램에 데이터를 전달하는 기초적인 웹 문서로 브라우저에서 그 실행결과를 보면 <그림 3>과 같다. 그림에서 ID와 PASSWORD를 입력하고 [전송]버튼을 누르면 <FORM>태그에서 설정한 내용대로 데이터를 웹서버에 전달하고, 웹 데몬은 해당 CGI프로그램에 전달한다. <표 2>에서 FORM의 ACTION 속성에 ASP와 같은 CGI프로그램을 설정하지 않고 단순히 CGI프로그램이 입력되어야 한다는 의미로 CGI를 단순 기입하였기 때문에 [전송]버튼을 누르면 에러 메시지가 나타난다.

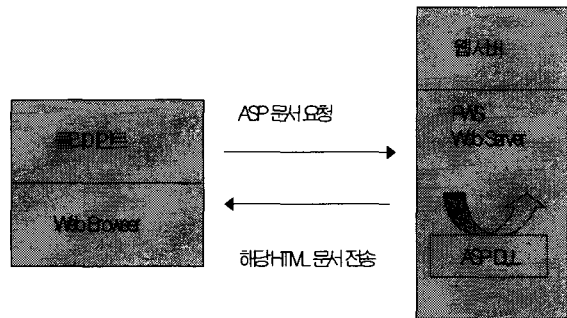


<그림 3> 브라우저로 실행 결과

2.3 데이터베이스 연동

입력양식에 입력한 데이터는 단순 텍스트파일로 브라우저에 되돌려 줄 수도 있고 데이터베이스에 저장할 수도 있다. 이렇게 할려면 ASP와 같은 별도의 CGI프로그램을 작성하여야 한다. 웹 문서와 데이터베이스의 연동은 클라이언트와 서버간의 대화방식으로 정보를 주고받는데 이를 위한 언어로는 C/C++, Perl, ASP, Java등이 있다. 이중 ASP는

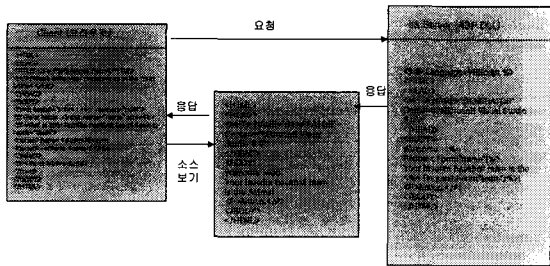
데이터베이스와 연결하는 프로그램을 작성하는데 용이하며 또한 ASP는 클라이언트가 웹서버에 동시에 접속하는 경우 스레드로 작동하므로 서버의 부담이 적다는 장점이 있다. ASP는 1995년 말에 등장한 IIS의 세 번째 버전으로 Win NT 기반의 IIS 3.0 이상 또는 Win 9x 기반의 PWS에서만 동작하는 특별한 스크립트언어이다. ASP코드는 자바 스크립트 또는 VB스크립트와 마찬가지로 HTML 문서 내에 삽입하는 일종의 스크립트언어이다. 스크립트언어는 서버사이드(Server-side)와 클라이언트 사이드(Client-side) 두 가지로 분류할 수 있다. ASP 스크립트를 포함하는 웹 문서를 클라이언트가 요청하면 서버는 ASP.DLL을 통하여 스크립트를 처리한 후 넘겨주므로 서버사이드 스크립트이다. 그러나, 자바 또는 VB 스크립트를 포함하는 웹 문서를 요청하면 웹서버는 해당 웹 페이지를 바로 클라이언트에 전송하여, 브라우저가 이를 받아 스크립트코드를 해석하여 화면에 내용을 보여준다. 이런 종류의 스크립트를 클라이언트 사이드(Client-side) 스크립트라고 한다. <그림 4>는 ASP웹문서에 대한 웹서버의 동작구조를 보여준다.



<그림 4> ASP 웹문서에 대한 웹서버 동작구조

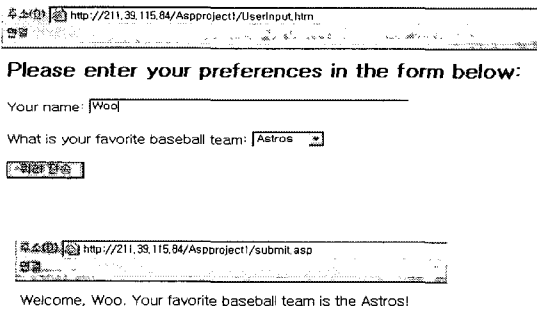
파일간의 구체적인 전송관계를 <그림 5>의 예를 통해서 보면 우선 클라이언트에 submit.asp라는 ASP파일의 실행을 요청하는 Preferences.htm이라는 웹문서가 있고 이것이 서버에 전달되면 IIS서버는 submit.asp에 있는 ASP스크립트를 ASP.DLL로 해석하여 HTML코드로 변환한 후 클라이언트에게 문서를 전송하게 된다. <그림 5>는 이러한 코드의 호출과 변환과정을 보여준다. 그리고 <그림 6>은 실제로 위의 코드를 실행한 결과로 폼에서 데이터를 입력받아 ASP로 처리한 결과를 클라이언트 화

면으로 되돌려 준 것이다. 웹서버는 처리 결과를 적당한 MIME 타입으로 포매팅(텍스트파일, 또는



<그림 5> ASP문장의 변환과 ASP.DLL의 내부적 처리

이미지파일, 또는 데이터베이스에 접속)한 후 HTTP데몬에게 넘겨서 클라이언트(브라우저)에서 보여주게 된다. <그림 6>의 사례는 데이터베이스에는 접속하지 않고 단순 텍스트 형태로 브라우저에서 처리결과를 보여주는 한 예이다.



<그림 6> ASP문장변환과 브라우저에서의 처리

<그림 6>은 비주얼 인터데브 6.0 으로 작성되어 PWS 웹서버에 탑재되어 http://211.39.115.84/asproject1/userinput.htm로 실행된 사례이며 userinput.htm과 submit.asp의 코드와 submit.asp의 실행 결과로 클라이언트에게 되돌려 주는 HTML코드는 각각 상기 <그림 5>의 첫번째 상자(Client(브라우저)), 세번째 상자(IIS Server(ASP.DLL)), 두번째 상자에 기술되어 있다.

3. 비주얼 인터데브의 구조

웹 애플리케이션은 웹서버를 실행하는 서버와 웹 브라우저를 실행하는 클라이언트라는 두 시스템으로 이루어져 있다. 웹 애플리케이션을 작성하기

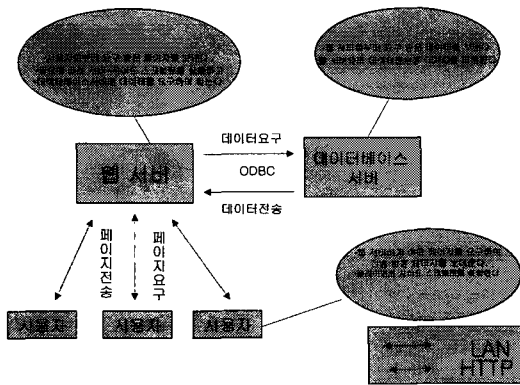
위해서는 우선 윈도 NT서버에서 웹 애플리케이션을 실행시켜주는 IIS웹서버와 데이터베이스를 실행, 연결시켜주는 MS-SQL 데이터베이스 서버와 같은 서버소프트웨어가 있어야하고 둘째, COM과 같은 구성요소 리소스[17], 데이터베이스연결, 시스템보안 및 트랜잭션 경계선을 관리하는 MTS(Microsoft Transaction Server) 구성요소 관리도구와 애플리케이션간 통신을 가능하게 해주는 MSMQ(Microsoft Message Queue Server)와 같은 운영체제 확장 서비스기능과 셋째 COM구성요소 특히 DLL을 작성할 수 있게 해주는 비주얼베이직, 비주얼 C++. 코볼등과 같은 구성요소 개발도구와 비주얼 인터데브와 같은 웹개발 도구가 필요하다. 비주얼 인터데브는 프로그래머이거나 프로그램에 대한 지식이 그리 많지 않아도 웹사이트를 쉽게 개발할 수 있도록 도와주는 도구라 할 수 있는데 실제 비주얼 인터데브에서 마이크로소프트 프론트페이지 같은 웹 전용에디터를 연결할 경우 프론트페이지가 제공하는 템플릿 기능이나 마법사기능을 이용할 수 있으므로 보다 쉽게 대화형 홈페이지나 ASP페이지를 제작할 수 있다.[18][19] 프론트페이지는 그 자체로도 매우 강력한 위지윅(WYSIWYG)편집기이나 마이크로소프트는 프론트페이지에 윈도 NT/95 기반의 웹서버를 추가함으로써 CGI스크립트를 작성할 수 있게 하여 모든 종류의 웹페이지 중에서 가장 뛰어난 기능을 갖추게 하였다.[19] 비주얼 인터데브는 말 그대로 비주얼한 통합 개발 환경을 제공한다.[20] 비주얼 인터데브를 이용하여 웹사이트를 개발하면 사용자의 웹브라우저, 웹서버, 데이터베이스 서버, 개발 워크스테이션의 4개의 구성요소가 서로 연동되어 실행되는데 이들의 연결관계는 웹 애플리케이션을 개발할 때, 실행할 때, 그리고 테스트할 때마다 각각 다르게 연동되며 각

<표 3> 인터데브와 구성요소간의 연결관계

구성요소의 연결	개발할 때	실행할 때	테스트할 때
사용자(웹 브라우저) ↔ 웹서버	-	HTTP를 통해	-
웹서버 ↔ 데이터베이스서버	-	LAN을 통해	LAN을 통해
개발시스템 ↔ 웹서버	HTTP를 통해	-	HTTP를 통해
개발시스템 ↔ 데이터베이스서버	LAN을 통해	-	연결은 유지하나 실질적 데이터 전송은 없음

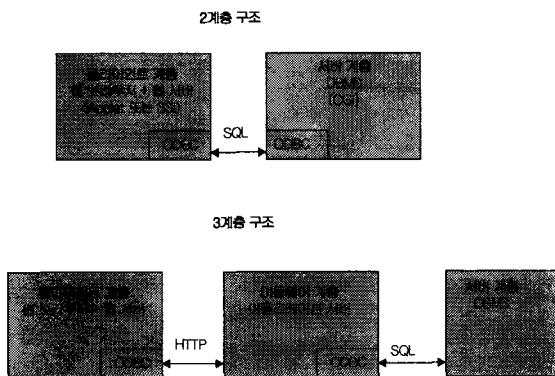
단계별 연결관계를 보면 <표 3>과 같다.

이때 개발시스템은 웹서버와 같은 시스템에 존재할 수도 있고 다른데 존재할 수도 있는데 이는 개발자가 웹 애플리케이션의 아키텍처를 몇 개의 계층으로 선택하느냐에 따라 각 계층 간의 연결구조는 달라질 수 있다. 참고로 웹 애플리케이션이 실행될 때의 구성요소간 연관관계를 보면 <그림 7>과 같다.



<그림 7> 인터데브와 웹 애플리케이션 실행시의 구성도
 자료: 은태환, 조보미공저, 비주얼 인터데브, 도서출판 대림, 1997,p.29

웹 애플리케이션 아키텍처는 N 계층으로 나눌 수 있으며 아키텍처의 계층 선택은 비용과 성능 및 관리를 포함한 많은 요소를 기반으로 주의 깊게 고려되어 선택되어야 한다.[21] 데이터베이스와



<그림 8> 데이터베이스와 웹의 연결: 2계층기술과 3계층 기술

자료: Thanaa M. Ghanem and Walid G. Arel, "Databases Deepen the Web, Computer, January 2004, pp.116-117

웹을 연결할 때는 2계층 기술과 3 계층 기술을 많이 사용하는데 <그림 8>을 한 보기를 보여주고 있다.[22]

비주얼 인터데브의 웹애플리케이션을 위한 구성요소와 작업내용을 보면 <표 4>과 같다. <표 4>에서 첫째 HTML파일은 프론트페이지나 인터데브 편집기를 이용하면 거의 모든 원하는 파일을 쉽게 작성할 수 있고 부족한 부분은 소스편집기로 직접 작성할 수도 있다. 둘째 ASP 파일은 서버사이드 스크립트("<%>"와 "<%/>")와 클라이언트사이드 스크립트("<SCRIPT>"와 "</SCRIPT>")중 서버사이드 스크립트를 포함하는 HTML파일이지만 <%@LANGUAGE = 스크립트언어 %>나 <SCRIPT RUNAT = SERVER LANGUAGE = JSCRIPT >를 이용하여 클라이언트 스크립트를 서버 스크립트로 동작하도록 하여 이용할 수 있다.

<표 4> 웹 애플리케이션의 구성요소

구성요소	확장자	작업내용
HTML파일	.htm	프론트페이지나 인터데브 소스편집기로 작성
Active Server Page 파일	.asp	서버 스크립트를 포함하는 HTML파일
전역(Global)파일	.asa	인터데브의 웹 프로젝트템플릿에 의해 생성
그림 및 멀티미디어 파일	.gif, .jpg, .bmp, .wmf, .avi 등	Image Composer, Music Producer, Media Manager 등을 이용 웹 프로젝트에 추가
ActiveX Layout 파일	.ax	컨트롤을 끌어다 놓는 방식으로 ax 파일을 만들고 이를 HTML이나 ASP파일에 추가함

셋째 전역파일에서는 애플리케이션의 범위나 세션 범위의 오브젝트를 선언하거나 이벤트 스크립트를 적는 곳으로 Application_OnStart, Application_OnEnd, Session_OnStart, Session_OnEnd 의 4개의 이벤트가 있으며 웹 프로젝트가 어떤 데이터베이스에 연결되면 Session_OnStart 스크립트가 자동으로 Global.asa 파일에 추가되어 이 정보는 Design Time ActiveX나 비주얼 데이터베이스 틀에서 이용된다. 넷째 HTML 파일 안에 들어갈 텍스트나 그림, 멀티미디어 요소들의 효과적인 배치를 위해서 ActiveX Layout 파일을 이용하는데 ActiveX Layout 파일은 HTML Layout 파일이라고도 하며 HTML 레이아웃은 여러 가지 컨트롤을 끌어다 놓는 방식으로 배치하고 그 구성을 저장할 수 있다. 이때 저장되는 파일은 .ax 확장자로 저장된다. 그런 다음 HTML 레이아웃파일을 HTML 파일이나 ASP

파일에 추가하여 웹 페이지를 꾸밀 수 있으며 하나의 웹 페이지에 여러 개의 HTML 레이아웃 파일이 포함될 수도 있고, 여러 개의 웹 페이지가 하나의 HTML 레이아웃 형식에 맞게 꾸며 질 수도 있다.

4. 비주얼인터데브와 웹데이터베이스 설계

4.1 설계개요

웹 애플리케이션과 데이터베이스를 연결하는 방법은 통상 아래 세 개의 범주중의 하나로 볼 수 있다.

- 프로그래밍언어로접속
(Programming Language-based Access)
- 스크립트언어로 접속
(Script Language-based Access)
- 애플리케이션틀로접속
(Application-based Access)

첫 번째 방법은 새로운 방법은 아니며 C, C++, Pascal, Cobol 등과 같은 프로그래밍언어 기준의 접속 방법으로 웹 환경과 통합이 잘 되지 않는 단점이 있다. 그러나 최근의 자바언어는 웹과 잘 조화를 이루는 언어로 보고 있다. 두 번째 방법은 VBScript, JavaScript, Perl, PHP, Python, ASP와 같은 언어를 이용하여 CGI를 통해서 데이터베이스 파일에 접근하는 방법으로 널리 쓰이는 방법이다. 세 번째 방법은 ColdFusion과 같은 애플리케이션 틀을 이용하여 접속하는 방법이다.[23] 본 연구는 두 번째의 방법을 이용하여 웹 애플리케이션과 데이터베이스의 통합을 시도하되 ASP개발 툴인 마이크로소프트의 인터데브와 프론트페이지를 이용한다. 본 연구에서 이들 툴을 이용하여 개발할 사례는 소매상에 음식물을 판매하는 가칭 노스윈드 식품(Northwind Foods)회사의 인터넷 애플리케이션을 구축하는 것으로 사용자의 주문내용에 대해 데이터베이스를 탐색하고 그 결과를 보여주도록 설계되었으며 <표 5>와 같은 HTML페이지와 ASP 페이지들로 구성되어 있으며 HTML코드, ASP스크립팅, 클라이언트 스크립팅, 서버스크립팅, HTML 컨트롤, ActiveX컨트롤, Design Time Control (DTC)등의 구성요소기술을 종합적으로 사용하며 FrontPage HTML편집기, 비주얼 인터데브 등의 통합개발환경(IDE: Integrated Design Environment)을 이용한다.

<표 5> 설계 파일과 기능

파일이름	기능
Default.htm	어플리케이션의 초기화면 ASP파일로 연결된다.
OrderSearch.asp	사용자로부터 탐색조건을 입력 받고 데이터베이스를 검색
OrderList.asp	검색결과를 보여주고 상세내용을 보여주는 ASP페이지로 이동
OrderDetails.asp	상세정보를 HTML파일로 보여줌

4.2 개발환경

본 사례연구에 이용된 개발환경은 <표 6>과 같으며 비주얼 인터데브에서 사용할 수 있는 구성요소는 자바언어로 Win32 애플리케이션이나 ActiveX 컨트롤을 빠르게 구축할 수 있게 해주는 WFC (Windows Foundation Class for Java)컨트롤, HTML페이지디자이너에서 사용할 수 있는 HTML 내부컨트롤, 서버의 속성과 메소드에 대한 접근 방법을 제공하는 Server Object, 웹 페이지를 디자인 할 때 이용할 수 있고 설계시에 그 속성을 설정할 수 있는 Design Time Controls(DTC), COM객체를 이용할 수 있게 해주는 ActiveX컨트롤 기능 등의 5가지가 있으나 본 사례에서는 표준HTML컨트롤, DTC, ActiveX컨트롤만을 사용하였다.

<표 6> 사용 개발환경

웹 서버	PWS
주 저작도구	Visual InterDev 6.0
연결사용자작도구	FrontPage 2000
데이터베이스서버	Microsoft Access 2000
운영체제	Window 98
사용구성요소	표준 HTML컨트롤, DTC, ActiveX컨트롤
인터데브 사용기능	WYSIWYG 편집기, Intellisense기능, Scripting Object Model, Scripting Library, Query Designer
사용스크립트언어	VBScript, JScript

비주얼 인터데브의 사용기능으로는 애플리케이션 내에서 페이지의 관계를 시각적으로 보여주는 다이어그램을 작성할 수 있는 사이트디자이너기능, 계단식배열 스타일 시트 편집기능, 속성과 메소드의 이름을 기억할 필요없이 이를 자동으로 보여주는 Intellisense 문 완성기능, 디자인타임 컨트롤과 스크립트 개체의 모든 기능이 스크립트 라이브러리 파일에 의해 저장되어 있는 Scripting Library 기능과 일련의 HTML과 ASP파일로 구성되어 폼

을 생성하고 데이터를 텍스트 상자나 라디오 단추 같은 입력 필드에 바인딩하고 웹 페이지의 기능에 대한 액세스를 애플리케이션 내에 있는 다른 페이지에게 데이터 개체로써 제공되는 데 필요한 모든 기능을 제공하는 Scripting Object Model의 기능이 있으나 본 연구에서는 사이트디자이너 기능과 CSS(Cascading Style Sheets) 편집기능은 사용하지 않았다.

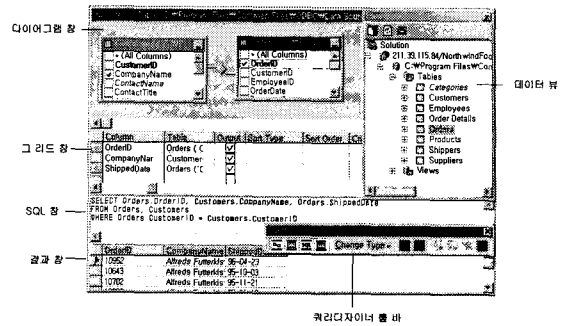
4.3 데이터베이스연결과 쿼리디자이너

웹 데이터베이스에 연결하여 사용할 데이터베이스는 Microsoft Access 2000으로 기 작성되어 있는 노스윈드(Northwind)라는 사례 파일을 이용하여 이를 셋업 하였으며 그 절차는 첫째 비주얼 인터데브 프로젝트 탐색 창의 global.asa 파일에 데이터연결을 추가하면 연결마법사가 나오게 되는데 마법사의 지시대로 그대로 따라하면 새 데이터 원본을 생성하거나 기존의 데이터원본을 ODBC 드라이버로 연결할 수 있게 해준다. 둘째로 마법사에서 나타나는 속성과 이름을 정하면 노스윈드의 데이터베이스가 새로운 데이터베이스로 설정되고 웹 페이지에서 이들의 테이블과 뷰를 이용할 수 있게 된다. 이 절차에 따라 연결 후의 프로젝트 창과 데이터 뷰 창을 보면 <그림 9>와 <그림 10>과 같다.



<그림 9> 프로젝트 탐색 창 <그림 10> 데이터 뷰 창

이제 연결된 데이터베이스의 테이블들을 상호 연결하여 웹 페이지에서 필요한 쿼리 들을 생성하기 위한 쿼리디자이너를 보면 <그림 11>과 같다.



<그림 11> 쿼리디자이너의 보기

쿼리디자이너는 SQL문을 만들어 내기 위해 쿼리를 비주얼 하게 만들어 내는 기능을 제공하는 것으로 다이어그램 창, 그리드 창, SQL 창, 결과창으로 구성되어 있다. 쿼리디자이너로 할 수 있는 작업은

- 모든 ODBC 호환 데이터베이스의 데이터를 검색할 수 있는 쿼리를 만들 수 있다.
- 결과 창에서 쿼리 결과를 미리 볼 수 있다.
- 테이블을 조인하여 다중테이블 쿼리를 만들 수 있다.
- 행을 업데이트하거나 삽입 또는 삭제하여 데이터베이스를 직접 편집할 수 있다.
- 행을 추가, 삭제 또는 복사하여 데이터베이스를 수정하는 쿼리를 만들 수 있다.
- 쿼리가 실행될 때 검색 값이 제공되는 매개 변수 쿼리같은 특수한 목적을 가진 쿼리를 만들 수 있다.

쿼리 디자이너가 지원하는 쿼리형식은 SELECT 쿼리, INSERT VALUE 쿼리, INSERT 쿼리, UPDATE 쿼리, MAKE TABLE 쿼리가 있다. 이밖에 비주얼 인터데브에서는 데이터베이스 다이어그램 디자이너라는 툴도 제공되는데 이는 데이터 베이스 구조 변경, 데이터베이스에 있는 개체정의에 영향을 주지 않고 다이어그램에서 테이블과 관계의 그래픽 레이아웃을 변경하는데 이용할 수 있다. 상기 <그림 11>은 주문리스트페이지(OrderList.asp)에서 보여줄 주문번호, 고객이름, 선적일을 보여주기 위한 쿼리를 만드는 작업으로 주문테이블(Orders)과 고객테이블(Customers)을 조인하여 결과를 얻었다. 이 작업은 상기 <그림 11>의 다이어그램 창에 두 테이블을 갖다놓으면 두 테이블간의 조인이 자동으로 생성되고 두 테이블의 속성들 중 출력하고자

하는 속성(Attribute)들만 그리드 창에서 선택하면 자동으로 SQL창에서 쿼리가 작성되고 실행시키면 그 결과가 결과 창에 보여지게 된다. 이와 같이하여 주문상세페이지(OrderDetails.asp)에서 주문번호에 따른 상품명, 단가, 수량, 할인율을 출력해줄 쿼리문을 쿼리디자이너로 만든 SQL문은 <표 7>과 같다.

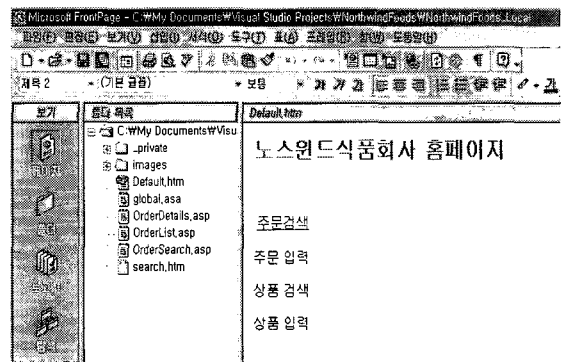
<표 7> 쿼리디자이너의 SQL문 생성 보기

```
SELECT Products.ProductName, 'Order Details'.UnitPrice,
'Order Details'.Quantity, 'Order Details'.Discount, Orders.OrderID
FROM Order Details, Products, Orders
WHERE 'Order Details'.ProductID=Products.ProductID
```

4.4 웹페이지 작성과 실행

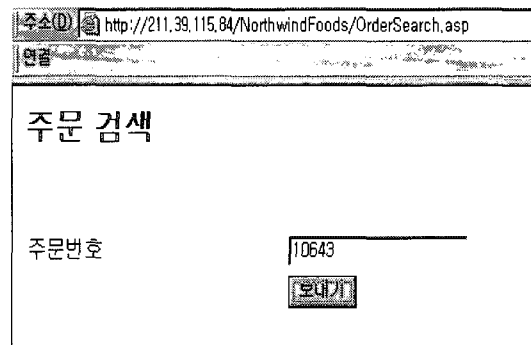
설계개요에서 전술한 바와 같이 본 연구에서 필요한 웹페이지는 디폴트 홈페이지(Default.htm), 주문 탐색 페이지(OrderSearch.asp), 주문 리스트 페이지(OrderList.asp), 주문 상세페이지(OrderDetails.asp) 들이다. 이들의 페이지들은 비주얼 인터데브의 프로젝트 탐색 창에서 HTML페이지를 추가하거나 ASP페이지를 추가하여 모두 작성할 수 있으나 본 연구에서는 FrontPage에 디터도 함께 이용하여 작성하였다. 보기로 웹의 초기화면인 디폴트 홈페이지를 비주얼 인터데브 내에서 프론트페이지를 연결하여 작성하였다. 이를 위해서는 우선 Default.htm 페이지를 인터데브의 프로젝트 탐색 창에 추가한 후 이 페이지를 View메뉴의 Open With라는 서브메뉴를 이용하여 프론트페이지로 열어서 초기화면을 작성하면 <그림 12>와 같다. 이때 프론트페이지의 웹 퍼블리싱(Publishing)전의 로컬 디렉토리 구조는 비주얼 인터데브의 로컬디렉터리(1-Tier 네트워크 구조의 예를들면 C:\My Document\Visual StudioProject\NorthwindFoods\NorthwindFoods_Local)와 같고 스크립트라이브러리 파일을 제외한 모든 파일을 같이 보여주고 있다. 비주얼 인터데브에서는 프론트페이지에서 처럼 이 파일들을 별도로 퍼블리싱 하지 않아도 웹 디렉토리로 이 파일들을 보여준다. 비주얼 인터데브의 웹 디렉터리(1-Tier 네트워크경우의 웹서버위치의 예: C:\inetpub\wwwroot\NorthwindFoods)를 본 연구의 경우를 예로들어 URL주소로 표현하면 http://211.39.115.84/NorthwindFoods이 되며 만약 프론트페이지로 웹

퍼블리싱을 하려면 이 웹 디렉토리로 하면 된다. 데이터베이스가 있는 디렉터리는 1-Tier 네트워크의 경우를 예를들면 C:\Program Files\Common Files\ODBC\Data Souces\Northwind(ACCESS)가 된다. 앞전의 웹 애플리케이션의 아키텍처에서 설명한 바와 같이 로컬디렉터리, 웹디렉터리, 데이터베이스 디렉터리를 각각의 서로 다른 서버에 이룰 위치 시켰다면 우리가 일반적으로 말하는 3-Tier 네트워크 구조가 된다.



<그림 12> 초기 홈페이지 화면의 보기

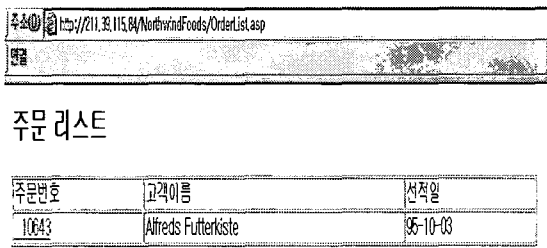
상기 초기화면에서 주문검색을 클릭하면 주문 검색 페이지(OrderSearch.asp)로 이동하는데 이 페이지에서는 방문객이 특정 주문번호를 입력하면 데이터베이스를 뒤져 그 주문번호에 해당하는 모든 주문 내용을 찾는 페이지로 <그림 13>에서와 같이 주문번호필드와 보내기 필드로 간단히 구성되어 있으나 내부적으로는 다소 복잡한 기술들이 사용되고 있다.



<그림 13> 주문 검색 페이지의 보기

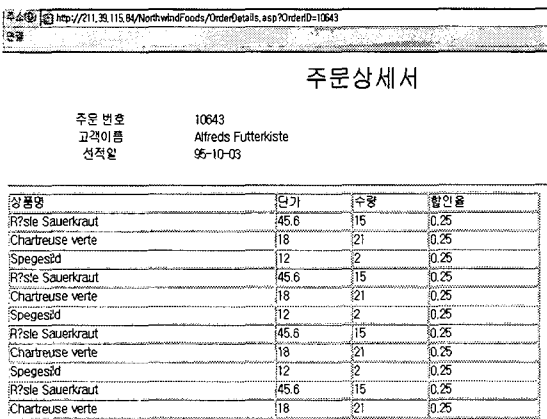
우선 주문번호 라벨 과 주문입력 텍스트박스가

하나의 양식(폼)으로 묶어져 보내기 버튼이 클릭되면 양식에 입력된 내용이 POST방식으로 다음 페이지인 주문리스트페이지(OrderList.asp)로 전송되게 되어있다. 주문리스트페이지는 주문 탐색 페이지에서 입력한 조건에 맞는 모든 주문들을 보여주는 것으로 주문번호 10643을 입력하여 검색된 리스트를 보여주는데 그 화면은 <그림 14>와 같다.



<그림 14> 주문 리스트의 보기

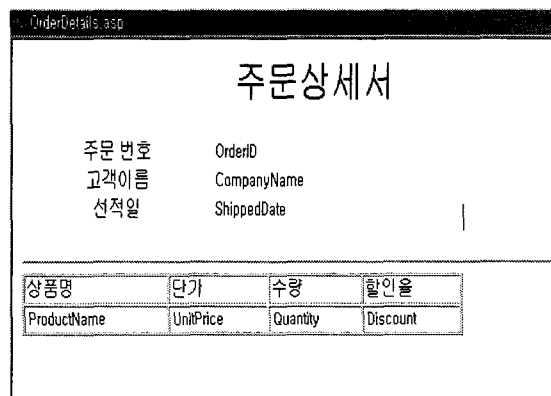
주문리스트페이지에서는 앞서 연결된 데이터베이스에서 쿼리디자이너로 추출한 레코드들을 원본 레코드로 셋업을 하고 주문 리스트 테이블의 2행의 각 열에 삽입되어 있는 레이블컨트롤과 레코드원본의 해당 필드들이 나타날 수 있도록 레이블컨트롤과 레코드원본을 연결하였다. 그리고 앞서 주문검색페이지에서 입력한 주문번호를 사용하여 레코드원본이 검색되고 그 결과가 디스플레이 된 후 다음 페이지인 주문 상세 페이지로 넘어갈 수 있도록 하기 위해 약간의 JScript코드를 추가하였다. 본 사례의 주문번호 10643을 탐색한 결과의 고객 이름과 선적일은 <그림 15>에 잘 나타나 있다. 주문 상세서 페이지는 주문 리스트 페이지에서 선



<그림 15> 주문 상세서 페이지의 보기

택된 특정 주문에 관해 상세 정보를 보여 주는 페이지로 주문에 대한 품목을 비롯하여 상세한 정보를 담고 있는 페이지가 된다.

<그림 15>는 주문 상세서의 실행 결과화면으로 실제로 데이터베이스 출력 결과는 2 페이지의 분량이나 앞부분만 추출하였다. 이 페이지에서는 두 개의 레코드원본을 셋업 하는데 한 개는 주문리스트페이지의 주문번호, 고객이름, 선적일을 한번 더 출력하는데 사용되는 주문레코드원본이고 다른 하나는 주문 상세서 페이지 하단의 상품테이블을 출력하기 위해 사용되는 상품레코드원본이다. 물론 두 개의 출력내용이 다르기 때문에 데이터베이스로부터 이를 추출하기 위한 SQL문도 서로 다르다. 앞서 쿼리디자이너를 설명할 때 보여준 SQL문으로 주문레코드원본과 상품레코드원본을 각각 추출하여 <그림 16>과 같이 설계한다.



<그림 16> 주문상세서 페이지 디자인 화면

첫 번째 주문레코드원본의 경우 상기 <그림 16>의 주문상세서 페이지의 윗 쪽에 있는 경계선이 감추어진 3행 2열의 테이블에서 2열의 각 행에 삽입한 각각의 레이블과 주문레코드원본의 해당 필드들이 연결되고, 두 번째 상품 레코드원본의 경우도 마찬가지로 상기 <그림 16>의 주문상세서 페이지의 하단에 있는 2행4열의 테이블 중에서 2행의 각 열마다 삽입된 레이블과 상품레코드원본의 해당 필드들이 연결되게 셋업 되게 한다. 다음으로 주문 리스트 페이지에서 넘어온 주문번호를 알아내고 그에 해당하는 상품을 찾을 때까지 루프를 도는 코드를 Jscript와 ASP스크립트로 혼합하여 삽입한다. 이 코드는 선택한 주문에 해당하는

모든 품목을 계속 찾으면서 보여 주게되는데 <그림 16>의 디자인 창에서 소스 창으로 변환한 후 코드를 입력하면 된다. 완성된 코드의 일부를 보면 <그림 17>과 같다.

5. 결 언

본 연구는 웹 기반 데이터베이스 통합 애플리케이션을 만드는데 어떻게 비주얼 인터데브가 다른 웹프로그래밍 툴들에 비해 간단한 코딩 작업만으로 이에 필요한 ASP와 HTML파일을 쉽게 제작할

```

<!-- Language=VBScript -->
<SCRIPT LANGUAGE="VBScript" SRC="Server/Scripts/asp.asp" -->
<!--#INCLUDE FILE="Server/Recordset.asp" -->
<!--#LANGUAGE="VBScript" RUNAT="server" -->
function _rsOrdersRecordset()
    var Orders = Request.QueryString("OrdersID")
    var DBConn = Server.CreateObject("ADODB.Connection")
    DBConn.ConnectionTimeout = Application("Northwind_ConnectionTimeout")
    DBConn.CommandTimeout = Application("Northwind_CommandTimeout")
    DBConn.CursorLocation = Application("Northwind_CursorLocation")
    DBConn.Open(Application("Northwind_ConnectionString"), Application("Northwind_RuntimeUserName"), Application("Northwind_RuntimePassword"))
    var rsOrders = Server.CreateObject("ADODB.Recordset")
    var rsConn = Server.CreateObject("ADODB.Connection")
    rsConn.ActiveConnection = DBConn
    rsConn.Source = rsOrders
    rsConn.CommandType = 1
    rsConn.CommandText = "SELECT Orders.OrdersID, Customers.CompanyName, Orders.ShippedDate FROM Customers, Orders WHERE Customers.CustomerID=Orders.OrdersID AND Orders.OrdersID = " & OrdersID
    rsConn.CacheSize = 10
    
```

<그림 17> 주문 상세서 페이지의 코드 보기

수 있는지를 보여 주는 것이 있다. 실제 본 연구를 통하여 본자와 같이 비주얼 인터데브는 사용하기 쉬우면서 강력한 기능을 제공한다. 자바 애플리케이션이나 애플릿은 크로스 플랫폼을 지원하기 때문에 그 기능적 측면에서 비주얼 인터데브보다 다소 우위에 있으나 프로그래밍이 어렵기 때문에 용이성 측면에 있어서는 한참 아래에 위치한다. 자바나 C++들은 비주얼 인터데브에 비해 상당히 복잡한 언어이다. 비주얼 인터데브에 있는 데이터베이스 도구들을 사용하면 쉽게 SQL문을 만들어 낼 수 있기 때문에 좀더 직관적이며 사용하기가 편리하다. 비주얼 인터데브는 웹 페이지와 데이터베이스 관련 부분을 하나의 환경 안에서 개발할 수 있다는 장점이 있다. Interanet Database Connector (IDC)등과 같은 이전의 솔루션들에 있어서는 SQL 정보를 다루는 파일을 하나 만들고 또 HTML페이지를 처리하는 파일을 또 하나 따로 만들어야 하는 부담이 있었다. 비주얼 인터데브는 데이터베이스와 통신할 수 있는 ADO(Active Data Object)나

DTC(Database Design Time Control)등의 컴포넌트를 제공하고 있으며 여러 가지 툴바와 메뉴 옵션들을 통하여 비주얼한 환경 하에서 SQL문을 비주얼하게 만들 수도 있고 즉석에서 그 SQL문에 대한 결과를 테스트할 수도 있고 복잡한 SQL문을 비주얼 환경에서 직접 프로그래밍할 수 있으며 데이터베이스 컴포넌트들을 관리하고 편집할 수도 있다. 특히 비주얼 인터데브 안에 포함된 데이터베이스 디자인타임 컨트롤은 ADO모델의 상위에 위치되어 있으면서 데이터베이스에 실행할 커맨드나 데이터베이스와의 연결을 위해 필요한 많은 스크립트들을 자동으로 만들어 낸다. 이 컨트롤을 애플리케이션에 추가시키고 SQL문을 쿼리디자이너를 사용하여 작성하게 되면 Recordset DTC가 SQL문을 실행시키는데 필요한 모든 스크립트들을 포함하여 이 로직을 ASP에 자동으로 삽입하게 된다. 본 연구에서는 이와같은 기능들을 사례 연구를 통하여 직접 실험해 보았으며 비주얼 인터데브가 애플리케이션에 데이터베이스를 통합시키는데 있어서는 타 옵션에 비해 보다 강력함을 경험하였다. 본 연구를 통하여 마이크로소프트 인터데브는 다른 웹프로그래밍 툴 들을 자주 이용해 보지 않은 개발자이거나 비주얼베이직이나 비주얼 C++, 프로토타입, 나모 웹에디터 등과 같은 개발 툴 들을 자주 이용해 본 개발자이거나 간에 다른 웹프로그래밍 툴에 비해 더욱 빠르게 그리고 보다 적은 코딩 노력으로 쉽게 웹프로그래밍을 작성할 있으며 현존하는 모든 component들을 별도의 노력없이 웹프로그래밍에 포함할 수 있음을 알았다. 최근 웹프로그래밍은 ASP 웹프로그래밍의 인프라 기반하에서 구현이 되는 웹서비스에 대한 기술로[24][25][26] 전이되고 있으며 거의 완전하다고 생각할 수 있는 개념의 웹서비스 기술은 윈도우 룹혼의 발표 시점쯤에 가서야 가능하다고 한다. 앞으로 웹프로그래밍에서 ActiveX? Java Applet? HTML? ASP? 등 승자가 누구건 간에 비주얼 인터데브로 구현할 수 있다는 사실에는 변함이 없다.

참 고 문 헌

[1] Jonathan W. Palmer, Web Site Usability, Design, and Performance Metrics, Information Systems Research, Vol.13, No.2, June 2002,

- pp.151-167
- [2] Jonathan Palmer, "Designing for Web Site Usability", Computer, July 2002, pp.102-103
- [3] Young F. John and G.E. Gorman, "Internet use in South Korea", Online Information Review, vol.26, no.5, 2002, pp.335-344
- [4] Sangjae Lee, Business use of Internet-based information system: the case of Korea, European Journal of Information System, Dec. 2003, pp.168-181
- [5] 홍동표, 문성배, 유선실, 박용우, 정부연, 김재경, 김민창, "국내 인터넷 쇼핑시장분석 및 전망", KISPI 이슈리포트, 2004, 6, 28, pp1-3
- [6] 한국인터넷정보센터, "미국의 인터넷 이용현황조사 결과", <http://isis.nic.or.kr/>, 2004년 7월
- [7] 한국인터넷정보센터, "캐나다의 전자상거래 및 기업정보화 현황", <http://isis.nic.or.kr/index.html>, 2004년 7월
- [8] 은태환, 조보미공저, 「비주얼 인터테브」, 도서출판대림, 1997, pp.1-2
- [9] Adam S. Huang, "Web-Based Information Systems Requirement Analysis", Information System Management, Winter 2003, pp.49-50
- [10] 배상훈, "협업과 상호 이해에 기반을 둔 전문화가 필요하다", 웹디자인, 2004, 01, Vol.49, pp.0.64-0.65
- [11] 최병호, "웹사이트 제작 프로세스의 혁신을 꿈꾸며: 제작의 현실과 대안", 웹디자인, 2004, 03, Vol.51, pp.0.54-0.55
- [12] Nijaz Bajgoric, "Server Operating Systems For E-Business: Features and Functions, Information Systems Management, Winter 2003, pp.42-48
- [13] Pairin Katerattanakul, "Framework of Effective Web Site Design for Business-to-Consumer Internet Commerce", INFOR Vol.40, No.1, Feb.,2002, pp.57-70
- [14] Susan J. Winter, Carol Sounders and Paul Hart, "Electronic Window Dressing: Impression Management With Websites, European Journal of Information System, 2003, 12, pp.309-322
- [15] L. WooKey, J. Geller, "Semantic Hierarchical Abstraction of Web Site Structures for Web Searchers", Journal of Research and Practice in Information Technology, Vol.36, No.1, February 2004, pp.23-34
- [16] 신희범, 김기용, 윤명영 공저, 「웹서버 구축 그리고 웹프로그래밍 :개인용 컴퓨터로 웹사이트 구축 실무」, 생능출판사, 2001. pp.26-27
- [17] Les Waguespack and Willian T. Schiano, "Component-Based is Architecture", Information Systems Management, summer 2004, pp.53-60
- [18] 최민석지음, 프론트페이지 2000으로 홈페이지 만들기, 정보문화사, 2000년, pp.85-100
- [19] William R. Stanek저, NetMate역, 마이크로소프트 프론트페이지 97 언리쉬드, 도서출판 대림, 1997, pp.366-377
- [20] 박노선, WBI 기반 학습자 평가 시스템 설계 및 구현, 한남대학교육대학원, 석사학위논문, 2001년 8월, pp.11-32
- [21] G. Andrew Duthie 저, 김택트랜스 편역, Microsoft Visual InterDev 6.0: Enterprise Developer's Workshop, (주)영진출판사, 1999, pp.51-73
- [22] Thanaa M. Ghanem and Walid G. Arel, "Databases Deepen the Web, Computer, January 2004, pp.116-117
- [23] Shae Jansons and Gary J. Cook, "Web-Enabled Database Connectivity: A Comparison of Programming, Scripting, and Application-Based Access", Information Systems Management, Winter 2002, pp.14-22
- [24] Billy Lin and H. Joseph Wen, "Web Services: An Analysis of the Technology, its Benefits, and Implementation Difficulties", Information System Management, Spring, 2003, pp.49-57
- [25] Chris Peltz, "Web Services Orchestration and Choreography", the IEEE Computer Society, October2003, pp.46-52
- [26] Mark Turner, "Turning Software into a Service", the IEEE Computer Society, October 2000, pp.38-44
- [27] Timothy Thompson, Rick weil, Mark D. wood, "CPXe: Web Services for Internet Imaging", Computer, October 2003, pp.54-62

- [28] Jen-Yau Chung, Kwei-Jay Lin, Richard G. Mathieu, "Web Services Computing: Advancing Software Interoperability", the IEEE Computer Society, October 2003, pp.35-37



우 원 택 (Won-Taek Woo)

1979년 2월 성균관대학교통계학
과학사

1981년 8월 연세대학교경영대학
원 석사

1991년 2월 영남대학교상경대학 박사

1993년 3월~현재 대구한의대학교멀티미디어학부

(관심분야 : 행동MIS, 지식전자상거래)