

# ASP System구축을 위한 액티브 서버 환경연구

The Active Server Environment for ASP System development

박진호, 정항녕, 박진석, 최성

Jin-ho, Park, Hang-nung, Jung Jin-suk, Park, Sung Choi

남서울대학교 컴퓨터학과

Dept of Comput Science NAMSeoul University

## 요약

ASP는 동적인 대화형 웹 서버 응용 프로그램을 작성하고 실행하기 위해 사용할 수 있는 '서버 측 스크립트' 환경이다. ASP를 사용하면 HTML 페이지와 스크립트 명령 및 ActiveX 구성요소를 조합하여 대화형 웹 페이지나 강력한 웹 기반의 응용 프로그램을 만들 수 있다. 사용자가 HTML에 익숙한 사람이라면 ASP 스크립트가 동적인 페이지의 작성을 위한 좋은 방법임을 알 수 있을 것이다. 또한 HTML 파일 작성에 필요한 자바 스크립트나 비주얼 베이직 스크립트는 물론이고 그 외 HTML 파일의 내용을 요구에 맞게 배치하기 위해 사용하는 CSS1의 기본적인 내용 정도는 알고 있어야 한다고 본다. 뿐만 아니라 CGI 프로그래밍과 웹 서버와 클라이언트의 상호 작용 관계, 그리고 웹 서버 관리하는 방법 등의 내용을 완벽하게 갖추고 있다고는 생각하지는 않는다. 하지만 앞으로 진행해 나가면서 직접 설명을 하지는 않겠지만 이와 같은 기술들이 모두 사용이 될 것이고, ASP는 다른 웹 애플리케이션 언어보다 쉽고, 웹에서의 많은 응용과, 무한한 가능성을 내포하고 있다는 것을 말하고 싶다. 그러므로 ASP에 접근하는 방법에 있어서 운영하는 방법을 연구하였다.

## I. 서론

ASP란 동적인 홈페이지를 만들 수 있는 페이지를 말한다. 확장자 명이 xxx.asp로 웹 프로그래밍을 위한 기술로서 Windows NT의 IIS(Internet Information Server)

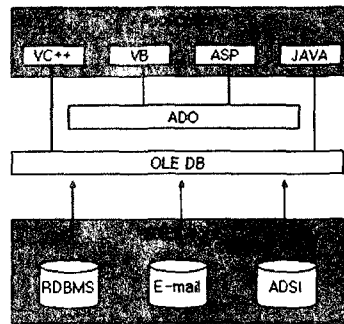
나 Windows 95/98의 PWS(Personal Web Server) 서버와 같은 웹 서버에서 지원되는 ISAPI를 기본으로 하는 솔루션이다. ASP 페이지는 DB(DataBase)와 연동할 수 있다. CGI ActiveX Control, ISAPI 같은 프로그램을 이용하지 않고도 ASP만으로 DB(DataBase)와 연동할 수 있어 더욱 매력적이다. 즉, 어떤 데이터를 DB에 저장도 하고, 데이터를 가져와서 보여줄 수 있어 스크립트 언어로서는 적합하다고 본다.

## II. ASP 객체

### ADO(ActiveX Data Object)

ADO는 데이터를 취급하기 위한 객체이다. 즉 데이터베이스 대상이라고 할 수 있다.

ADO는 OLE-DB 제공자를 통해서 데이터베이스 서버에 접근하여 데이터를 액세스 할 수 있도록 해준다. 이 ADO를 연결할 때 사용하는 객체를 ADO Connection 객체라고 한다. ASP에서 데이터베이스와 연결해서 사용하는 곳에서는 어디서나 이 ADO 연결객체를 볼 수 있다. ADO 객체는 데이터베이스를 다루는 객체이다. 데이터베이스를 다루는 기본 과정은 먼저 데이터베이스에 연결하고, 명령을 실행하고, 실행한 결과를 사용자에게 보여주는 것이다.



[그림 1] ADO, OLE DB, 저장소와의 관계

위의 [그림 1]은 ADO와 OLE DB가 애플리케이션과 데이터 저장소 사이에서 어떠한 역할을 하는지를 보여주는 그림이다.

### 1. ADO 객체 구성

ADO 객체 모델은 데이터베이스를 사용하는데 필요한 객체를 모아놓은 것이다. 그 중에서 다음에 세 가지를 주로 사용해서 데이터베이스를 처리한다.

◎Connection 객체 : 데이터 제공자에 대한 연결 정보를 가지는데 Connection 객체라고 연결만 하고 끝나는 것이 아니라 이 객체를 이용해서 SQL를 실행시키고 그 결과를 레코드 셋으로 받아와서 보여주기기도 한다. 사실 이 객체만 있어도 데이터베이스에

연결해서 명령을 실행하고 그 결과를 가져와서 볼 수 있다.

◎Command 객체 : Connection 객체와 마찬가지로 단순히 명령만 실행하는 것이 아니라 생성된 Connection객체가 있다면 이것을 이용해서 데이터베이스를 연결하면 되고 없더라도 연결 문자열을 넘겨주면 자동으로 연결을 시도하고, 명령을 실행한다.

◎Recordset 객체 : Connection 객체와 Command 객체를 이용해서 명령을 주고 실행한 다음 받아오는 결과는 Recordset 객체이다. 이 객체를 이용해서 각 레코드의 필드값을 페이지에 보여주는 것이 일반적인 방법. 다른 객체 없이 Recordset객체만 있어도 데이터베이스에 연결해서 명령을 실행한 다음 결과를 가져올 수 있다.

## 2. ADO Connection 객체 사용

ADO연결 객체가 없다면 레코드 셋 자체의 정의를 이루어지지 않는다. 즉, 데이터베이스와 연결이 되어야만 레코드들을 가져올 수 있다. 예를 들어 테이블을 생성한 다음 '윈도우 ODBC 등록'을 하였다면 기본적으로 'DSN'과 'UID', 'PWD'가 있을 것이다. 만약 독립된 DB서버를 사용하면 웹서버와 연결되어 있어야 한다.

### [알고리즘1]

```
Set DBCon=server.CreateObject
("ADODB.Connection")
DBCon.Open"dan=asp;uid=asp;pwd=asp"
sql1 = "select * from sample"
SET url = DBCon.Execute(sql1)
```

[알고리즘1]에서 먼저 첫째 줄은 ADO연결 객체를 생성하는 것이다. 그리하여 그 이름을 'DBCon'이라고 정했다. 둘째 줄은 첫줄에서 생성한 객체를 여는 곳으로 'DSN', 'UID', 'PWD'가 모두 'asp'인 DataBase Instance를 열겠다 라는 말이다. 셋째 줄은 자신이 필요한 레코드 및 그 값을 얻을 수 있는 적절한 sql문을 만들고 넷째 줄은 레코드 셋을 만든다는 뜻이다. 즉, 셋째 줄에서 정의한 SQL문을 수행(.Execute)을 하되 위에서 생성한 'DBCon'이라는 연결 객체를 사용한다는 것이다.

## III. 웹서버에서 ASP를 실행시키는 방법

ASP 프로그램은 서버에서만 실행이 되며 그 결과는 클라이언트(웹 브라우저)에서만 실행될 수 있다. 좀더 자세히 설명을 하자면 보통 C로 짠 프로그램과 같이 도스나 윈도우의 프롬프트 상태에서 실행시킬 수 있는 것이 아니라 클라이언트가 서버에 있는 ASP 애플리케이션의 실행을 요구를 하면 웹 서버는 그 요청을 받아 들여 해당되는 ASP 애플리케이션을 실행시

킨 뒤 오직 그 결과만 요청한 클라이언트에게 전달한다. 즉, 실행은 서버에서 이루어지고 그 결과만 클라이언트에게 던져 주는 것이다. 이것은 CGI 방식과 상당히 흡사하다. 이러한 관점에서는 ASP도 하나의 CGI 방식이라고 봐도 상관이 없겠다. 그러면 웹서버가 ASP를 실행시킬 때 그 파일이 ASP 파일인지 아닌지를 어떻게 구분할까? 웹 서버는 이 구별을 위해 아주 전형적인 방법을 쓰고 있다. ASP의 코드 내용을 보통 HTML 태그와 같이 섞어 코딩한 다음에 확장자를 ".asp"로 저장을 한 뒤 적당한 웹 서버의 홈페이지 디렉토리에 넣어 두면 클라이언트가 참조할 때 서버가 알아서 실행한다.

ASP는 HTML 태그와 같이 사용될 수 있는데 HTML 태그와 혼동을 피하기 위해 <% 와 %> 사이에 ASP 내용을 넣는다.

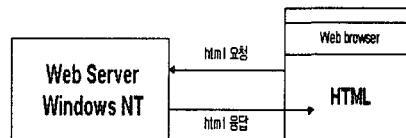
### [알고리즘2]

```
<html>
<body>
<%
response.write("<H1 style='text-align=
center;color=blue'>안녕하세요</H1>")
%>
</body>
</html>
```

ASP도 [알고리즘2]와 같은 퍼미션 설정이 때로는 필요한데 방법은 유닉스와 다르나 목적은 같다고 보면 된다. 유닉스에서 CGI 프로그램을 실행시키기 위해서는 CGI 구동이 가능한 디렉토리에 적당한 퍼미션을 주고 그 안에 CGI를 두고 실행 퍼미션을 열어 주어야 한다. 윈도우 95에서 PWS로 ASP를 실행하려면 단순히 디렉토리에 실행이나 스크립트 실행을 체크해 주면 되고 윈도우 NT에서 IIS로 ASP를 실행시킬 때도 마찬가지로 디렉토리에 프로그램을 실행시킬 수 있는 퍼미션을 열어 주어야 한다. 윈도우 NT는 윈도우 95보다 보안 기법을 좀더 제공해 웹에서 많이 쓰이는 Basic Authentication과 그외 여러 인증 방법을 제공해 특정인에게만 그 스크립트를 실행 할 수 있도록 기능을 제공한다.

## IV. HTML의 구동원리

이 ASP라는 페이지는 Web을 프로그래밍 할 수 있도록 해 주는, 서버에서 동작하는 페이지로써 기존의 HTML 페이지와는 상당히 다른, 동적인 구성을 가질 수 있게 해 준다.



[그림2] HTML 구동원리

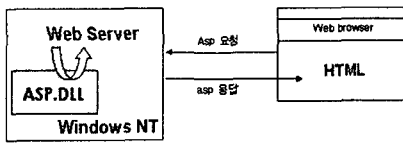
[그림2]은 클라이언트가 서버의 HTML 페이지를 요청할 경우의 동작내용을 보여주고 있다. 클라이언트란 어떤 정보나 서비스를 요청하는 사용자들을 의미한다. 즉, 웹을 서핑하는 모든 사용자들은 클라이언트들인 것이다.

그들이 어떤 서버로(예를 들면, 조선일보 서버) 접속해서 show.html을 요청한다면, 조선일보 서버는 show.html을 접속한 사용자 클라이언트의 브라우저에게 건네주게 된다.

이 html 페이지는 태그, 스크립트 등으로 구성된, html 언어로 제작된 페이지이다. 서버가 그 html 코드를 사용자의 브라우저로 넘겨주면 서버의 역할은 그것으로 끝난다.

그렇기에 웹에 접속하고자 하는 사용자들은 반드시 html 코드를 해석할 수 있는 브라우저가 있어야 한다. 서버의 역할은 클라이언트가 요청하는 html 페이지를 찾아서 사용자에게 넘겨주기만 하면 되는 것이다.

ASP의 경우는 사용자가 어떤 페이지를 요청했다고 가정했을 때, 그 페이지를 html 페이지처럼 사용자의 브라우저에게 그냥 보내 주지는 않는다.



[그림3] ASP 구동원리

[그림3]와 같이 사용자에게 보내주기 이전에 서버 내에서 어떤 처리를 하고 그 처리 결과를 태그로 만들어, html 형식으로 다 만든 뒤, 그 만들어진 페이지를 사용자의 브라우저에게 보내주는 것이다

### V. ASP의 객체, 컴포넌트, ActiveX

프로그래머가 비주얼 베이직 스크립트나 자바 스크립트로 웹 애플리케이션을 만들 때 많은 어려움을 느끼는 데 이는 스크립트 언어 자체가 서버의 부담을 줄이고자 클라이언트 쪽의 보안 체계가 허용하는 한도 내에서 최대한 할 수 있는 작업을 수행하고자 만들어졌기 때문이다. 그래서 파일에 접근하거나 데이터베이스에 접근하는 것은 상당히 어렵다. 이러한 단점을 가진 스크립트 언어가 ASP 객체와 컴포넌트와 결합해서 서버 쪽에서 실행됨으로써 막강한 기능을 가진 언어로 재 탄생하게 되었다. 물론 보안 문제도 충분히 해결이 된다. 하지만 장점이 있으면 단점도 있기 마련이다. 이러한 스크립트 언어가 탄생할 때는 서버의 부담을 줄이고자하는 목적으로 클라이언트에서 실행이 되었는데 다시 서버와 결합이 되므로 서버에게는 일이 늘어나 이전 보다 많은 부하가 걸릴 것이다. 그러나 그렇게 걱정할 정도는 아니다. 그 정도는 충분히 감수할 만하고, 또 기존의 CGI와 비교를 해 보면 훨씬 낡

은 환경이니 말이다.

ASP 객체와 컴포넌트는 마이크로소프트사의 ActiveX 컴포넌트이다. ASP에서 기본적으로 제공되는 주요 객체는 Application, Session, Request, Response, 그리고 Server가 있다. 이러한 객체들은 ASP 수행 중에 언제든지 사용이 가능하다.

ASP에서 제공하는 컴포넌트는 프로그래밍 중에 바로 사용할 수 없으며 그 컴포넌트에 대한 객체를 직접 만들어 사용해야한다. 이러한 컴포넌트를 사용하면 스크립트에서 지원이 안되는 기능들을 쉽게 이용할 수 있다.

### VI. ASP를 사용하기 위한 환경

윈도우98에는 PWS 4.0이 기본적으로 들어있다. 이 버전은 NT의 Option Pack에 해당하는 것으로 엄밀히 말하자면 IIS는 아니다.

윈도우NT4.0 환경에서는 기본적으로 네트워크 기능이 월등하므로 훨씬 사용 환경이 안정되어 있다. 먼저 IIS 3.0으로 업그레이드해야 사용할 수 있는데 이는 서비스팩3을 설치하면 IIS2.0에서 3.0으로 업그레이드 가능하다. 그리고 ASP는 내장되어 있지 않으므로 따로 인스톨을 해야한다.

다른 방법으로는 옵션 팩을 설치할 수도 있는데 옵션 팩에는 2가지 종류가 있는데 윈도우9.x 계열은 PWS 4.0이, 윈도우 NT환경에서는 IIS 4.0이 설치된다.

이외에도 옵션 팩에는 로컬 검색을 구현해 주는 Index Server, 사이트 관리와 상거래 시스템을 담당하는 사이트 서버의 일부분 기능으로 구성된 Site Server Express, 뉴스 서버인 NNTP Server, 그리고 메일을 보내는 기능을 담당하는 SMTP Server, 인증 기능을 담당하는 Certificate Server, Radius 그리고 트랜잭션이 가능한 컴포넌트 관리를 해주는 MTS(Microsoft Transaction Server) 등 여러 가지 프로그램이 포함되어 있다. 이러한 일련의 프로그램들은 모두 ASP와 연동이 가능해 여러 가지 막강한 기능을 구현할 수 있다.

### VII. Server 객체의 Execute 메소드

IIS 에서 새로이 등장한 2개의 메소드 중 하나인 서버(Server) 객체의 Execute 메소드 이다.

Execute 메소드는 .asp 파일을 호출하여 마치 이 파일이 호출한 ASP 스크립트의 일부인 것처럼 처리한다. Execute 메소드는 많은 프로그래밍 언어에서 프 로시저 호출과 유사하다. 즉, ASP 페이지를 마치 하나의 함수처럼 사용할 수가 있다고 하는 것이다.

[알고리즘3]은 asp페이지를 만들기 위한 예제 소스를 나타낸 것이다.

in.asp파일이 중간에 하나의 함수나 프로시저처럼 실행이 되어져, 페이지의 사이에 그 결과가 끼여 들어가 고, 계속 페이지가 완성되는 것을 볼 수가 있다.

