Making Information accessible via the WWW Infrastructure

..... A brief overview ....

Goal

Goal:

To provide an introduction to the architectural and technology issues in making information available via the Web

Overview

- Motivation
- Web Server and Architectures
- Techniques for making database information available via the Web
- Examples

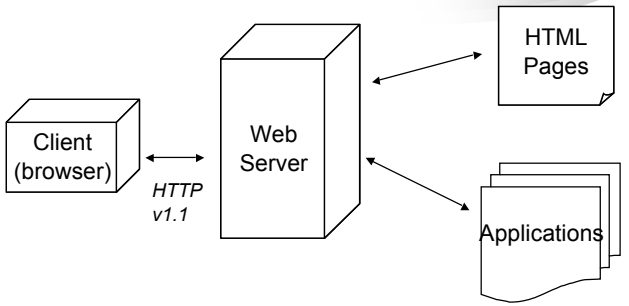
Motivation

- Platform (vendor) Independent access to DB
  - Independence of Conceptual Model, Storage Model as well as information management & hosting technology platforms
- Distributed (wide area & local area) access
- Support for flexible (G)UI
- Few client dependencies
- Security (?)
- Scalability (?)
- :
- :

Web Servers

- Responsible for transfering information to/from Client fromto Applications or Web pages
- Simple Protocol (HTTP) consisting of PUT, GET, DELETE etc…
- Support for MIME data types e.g. video, text, image, etc.

General Web Architecture



Example Web Servers

- Microsoft Internet Information Server (IIS)
- Apache
  - Multi Platform Support
- Jigsaw: java-based W3C
- PWS
- Many vendor products e.g. IBM, ORACLE, SUN, etc.

Some Web/DB Techniques

- Common Gateway Interface:
  - JavaScript/VBScript/Perl/PHP
- Active Server Pages (ASP)
- Java Servlets
- Applications: XML-based Web Interface

Common Gateway Interface (CGI)

- Historically one of the most widely used of techniques
- Implementded in all Web Servers
- CGI automatically launches program (script) if it recognises that URL requested d by the browser (client) points to a programme/script
- CGI-script implements required (application) functionality/logic using many possible scripting languages e.g. Java, C/C++, PHP, Perl, ….
- Info passed to script via: command line parameters, emvironment variables, standard input, extra path information

CGI Advantages & Disadvantages

- Advantages:
  - Generic Interface
  - Simplicity
  - Language Independence
  - Web Server Independence
- Disadvanatges
  - Web Server Possible Bottleneck
  - Lack of efficiency and txn support (multiple login/logout)
  - New process or thread for each CGI script (script may have to handle concurrency)
  - Possible Security holes due to execution of system routines by scripts

Remote Data Services (RDS)

- Improvement on Table Data Control (TDC) sending data back to the server
- Client-side technique: possible reduction of network traffic
- Client-side technique: Fat or Rich Client
- Microsoft Specific – ActiveX component in the client browser => only works with IE5
  - Quick to develop

Active Server Pages (ASP)

- Microsoft Specific => requires IIS as Web Server
- JScript/VBScript included in HTML code and evaluated at the server-side
- Dynamic HTML pages can be quickly developed

JSP and Java Servlets

- Currently very popular server side scripting technique to support thin-client web applications
- Embed dynamic content using special tags. A JSP is compiled to Java Servlets and processed by a Java-enabled Web server
- JSP engine transforms JSP tags, Java code, and static HTML content into Java code which is then automatically organised by JSP engine into underlying Java Servlets.

JSP and Java Servlets

- Servlets are automatically compiled into Java bytecode
- Each Servlet implements a particular interface and is automatically called by the server (when its interface is interacted with by a user)
  - Servlet loaded only once: quick startup
  - Java-based: platform independent
  - Supported by any major Web Server

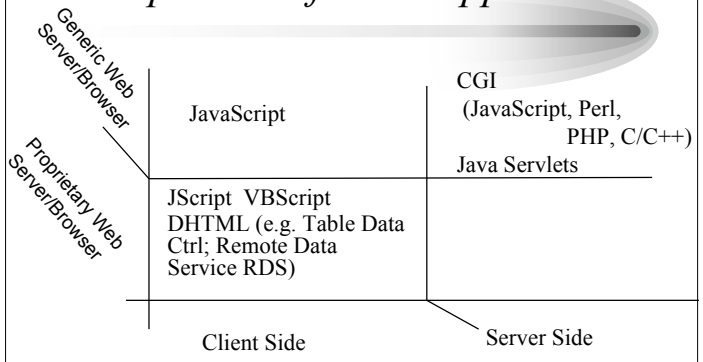
Java Servlets

- All servlets implement the 'Servlet' interface: **`init(), ServletConfig(), service(), getServletInfo(), destroy()`**
- Service() is called to respond to user requests
- Class `HttpServlet` provides default implementations for these methods
  - **`doGet()`** and **`doPut()`** get executed whenever the Web server receives a GET or PUT request
- Servlets usually extend HttpServlet

Application:XML

- XML is defacto standard format in which digital documents are transferred
- XSL used to transform XML document into anything! (typically HTML)
- Typically server side extracts data from database into XML (e.g. from local DB format) and transforms to into HTML using XSL & sends it to the client

Comparison of Some Approaches



References used

- 'Database Systems: A Practical Approach to Design, Implementation and Management'; 3rd Edition; T Connelly & C Begg, Addisson Wesley 2002 pages 955-997
- 'Internet and World Wide Web – How to Program' H.M. Deitel & T.R. Nieto, Prentice Hall, 2000
- Presentation by J. Bispal http://www.cse.msu.edu/~bisbal/