Unit 2: Issues of Web Technology (4 Hrs.)

Contents

Architectural Issues of Web Layer

Web design is the process of planning and creating a website. Text, images, digital media and interactive elements are used by web designers to produce the page seen on the web browser. Web designers utilize markup language, HTML is used for structure and CSS for style JavaScript for behavior of web pages that can be read by web browsers.(about tech)

The three layer of web design:

Structure or content layer:

The content or structure layer is what visitor are coming to get when they come to your Web page. Content can consist of text and images and includes the hyperlinks that your visitors need to navigate around your Web site. In Web development, HTML makes up the content layer and it also structures the Web document.

Style or presentation layer:

The style or presentation layer is how a structured HTML document will look to a site's visitors. This layer is defined by the CSS or styles that indicate how your document should be displayed and on what media types.

Behavior layer:

The behavior layer is the layer of a Web page that can respond to different user actions or make changes to a page based on a set of conditions. For most Web pages, the behavior level would be the JavaScript interactions on the page.

It is important to keep the layers separate using external files. External style sheet is the best way to separate your content from design. And the same is true for using external JavaScript files.

Some of the benefits of separating the layers are :(web designing plus)

Shared resources: When you write an external CSS file or JavaScript file, you can use that file by any page on your Web site. There is no duplication of effort, and whenever the file changes, it changes for every page that uses it without you making more than one change.

Faster downloads: Once the script or stylesheet has been downloaded by your customer the first time, it is cached. Then every other page that is downloaded loads more quickly in the browser window.

Multi-person teams: If you have more than one person working on a Web site at once, you can divide up the workload without worrying about permissions or content management. You can also hire people who are style/design experts to work on the CSS while your scripters work on the JavaScript, and your writers work in the content files.

Accessibility: External style sheets and script files are more accessible to more browsers, because they can be ignored more easily, and because they provide more options. For example, you can set up a style sheet that is displayed only for screen readers or a script library that's only used by people on cell phones.

Backwards compatibility: When you have a site that is designed with the development layers, it will be more backwards compatible because browsers that can't use technology like CSS and JavaScript can still view the HTML. Your Web site can then be progressively enhanced with features as browsers support them.

Hypertext Transfer Protocol(HTTP)

HTTP stands for Hypertext Transfer Protocol. It a TCP/IP based communication protocol, that is used to deliver resource on the World Wide Web. This resource could be HTML files, image files, query results, etc.

The default port is TCP 80, but other ports can be used as well. It provides a standardized way for computers to communicate with each other. Most typical use of HTTP is between Web browser (client) and Web server. There are three basic features which make HTTP simple but powerful protocol they are :(Tutorailspoint)

HTTP is connectionless: HTTP client initiate an HTTP request after a request is made, the client disconnects from the server and waits for a response. The server must re-establish the connection after it processes the request

HTTP is media-independent: Any type of data can be sent by HTTP as long as both the client and server know how to handle the data content. How content is handled is determined by the MIME specification.

HTTP is stateless: This is a direct result of HTTP's being connectionless. The server and client are aware of each other only during a request. Afterward, each forgets the other. For this reason, neither the client nor the browser can retain information between different requests across the web pages.

The following diagram shows where HTTP protocol fits in communication:

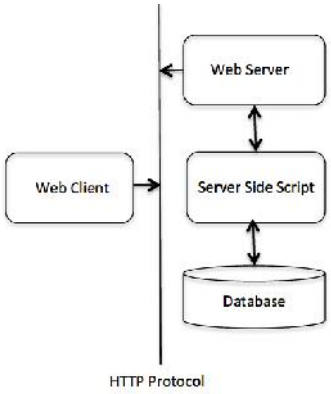

fig: Basic architecture

The HTTP protocol is a request/response protocol based on the client/server based architecture where web browsers, search engines, etc. act like HTTP clients, and the Web server acts as a server.

Http uses client-server architecture model and stateless protocol to exchange resources .An HTTP client opens a connection and sends a request message to an HTTP server; the server then returns a response message, usually containing the resource that was requested. After delivering the response, the server closes the connection. HTTP requests and responses use a generic message format of RFC 822 for transferring the data. This generic message format consists of the following four items.

1. An initial line
2. Zero or more header fields followed by CRLF
3. A blank line(i.e., a line with nothing preceding the CRLF) indicating the end of the header fields
4. An optional message body like file, query data or query output.

An initial line:

initial line will have the following generic syntax:

initial -line = *Request-Line* | *Status-Line*

let's see the examples of start line in case of request and response:

GET /index.htm HTTP/1.1 (This is ***Request-Line*** sent by the client)
HTTP/1.1 200 OK (This is ***Status-Line*** sent by the server)

Header field: Header lines provide information about the request or response, or about the object sent in the message body. The header lines are in the usual text header format, which is: one line per header, of the form "Header-Name: value", ending with CRLF.

Example:

Host:www.example.com
Last-Modified: Fri, 31 Dec 1999 23:59:59 GMT
Content-Type:text/plain
Content-length:60

Message body: A message body is the one which carries the actual HTTP request data (including form data and uploaded, etc.) and HTTP response data from the server (including files, images, etc.).

HTTP message header: (tutorialspoint)

There are four types of HTTP message headers:

General-header: These header fields have general applicability for both request and response messages. General header contains different fields they are Cache-Control, Connection, Date, Pragma, Trailer, Transfer-Encoding, Upgrade, Via, Warning. Some example of general header fields are :

Cache-control: no-cache
Upgrade: HTTP/2.0, SHTTP/1.3, IRC/6.9, RTA/x11
Connection:close
Connection :keep-alive
Via: 1.0 fred, 1.1 nowhere.com (Apache/1.1)

Client Request-header:

These header fields have applicability only for request messages.client request header contains different fields they are Accept,Accept-Charset,Accept-Encoding,Accept-Language,Authorization,Cookie,Expect,From,Host,If-Match,If-Modified-Since,If-None-Match,If-Range,If-Unmodified-Since,Max-Forwards,Proxy-Authorization,Range,Referer,TE,User-Agent.

Example:
Accept: text/plain; q=0.5, text/html, text/x-dvi; q=0.8, text/x-c
Accept-Charset: iso-8859-5, unicode-1-1; q=0.8
Accept-Encoding: compress, gzip
Accept-Language: da, en-gb;q=0.8, en;q=0.7
From:webmaster@w3.org
Host:www.w3.org
If-Modified-Since: Sat, 29 Oct 2000 19:43:31 GMT
Range: bytes=0-499

```
Accept: audio/*; q=0.2, audio/basic
```

SHOULD be interpreted as "I prefer audio/basic, but send me any audio type if it is the best available after an 80% mark-down in quality."

Server Response-header:

These header fields have applicability only for response messages.different fields of response header contains Accept-Ranges,Age,ETag,Location,Proxy-Authenticate,Retry-After,Server,Set-Cookie,Vary,WWW-Authenticate fields.

Example:

Range: bytes = 0-499
Accept-Ranges: bytes
Accept-Ranges: none
Age: 1030(delta-second)
Server: Apache/2.2.14 (Win32)
WWW-Authenticate: BASIC realm = "Admin"
Vary: Accept-Language, Accept-Encoding

Entity-header:

These header fields define meta information about the entity-body or, if no body is present, about the resource identified by the request.entity header contains:Allow,Content-Encoding,Content-Language,Content-Length,Content-Location,Content-MD5,Content-Range,Content-Type,Expires,Last-Modified fields.

example

Allow: GET, HEAD, PUT
Content-Encoding: gzip
Content-Length: 3495
Content-MD5 : 8c2d46911f3f5a326455f0ed7a8ed3b3
Content-Range : bytes 0-499/1234
Content-Type: text/html; charset = ISO-8859-4

https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html

FTP

FTP stands for file transfer protocol. It is used to send files from one system to another under user commands. Both text and binary files are accommodated and the protocol provides features for controlling user access. It is an old protocol and is used less than it was before the World Wide Web came along. Today, its primary use is uploading files to a Web site.

To transfer the file FTP makes TCP connection to the target system. These allow used ID and password to be transmitted and allow the user to specify the file and file action desired. Once file transfer is approved, a second TCP connection is set up for data transfer. The file is transferred over the data connection, without the overhead of headers, or control information at the application level. When the transfer is complete, the control connection is used to signal the completion and to accept new file transfer commands.

FTP can be run in active or passive mode, in *active mode* the client sends the server the IP address and port number on which the client will listen, and the server initiates the TCP connection. In *passive mode* the client sends a PASV command to the server and receives an IP address and port number in return.

Data transfer can be done in any of three modes:

Stream mode: Data is sent as a continuous stream, relieving FTP from doing any processing. Rather, all processing is left up to TCP. No End-of-file indicator is needed, unless the data is divided into records.

Block mode: breaks the data into several blocks (block header, byte count, and data field) and then passes it on to TCP.

Compressed mode: Data is compressed using a single algorithm (usually run-length encoding).

Things to remember

- Web design is the process of planning and creating a website. Text, images, digital media and interactive elements are used by web designers to produce the page seen on the web browser.
- web design consists three layers they are **:**Structure or content layer,Style or presentation layer,Style or presentation layer.
- HTTP stands for Hypertext Transfer Protocol.It a TCP/IP based communication protocol, that is used to deliver resource on the world wide web
- HTTP is connectionless, media independent and stateless protocol
- This generic message format consists of the following four items.
    1. An initial line
    2. Zero or more header fields followed by CRLF
    3. A blank line(i.e., a line with nothing preceding the CRLF) indicating the end of the header fields
    4. An optional message body like file, query data or query output.
- There are four types of HTTP message headers:**General-header,Client Request-header,Server Response-header:Entity-header:**
- FTP stands for file transfer protocol.It is used to send files from one system to another under user commands.FTP can be run in active or passive mode
- Data transfer can be done in any of three modes:**Stream mode,Block mode,Compressed mode.**

Tier Technology

Client-Server architecture

Client-server architecture consists of two kinds of computers: clients and servers. Clients are the computers that that do not share any of its resources but requests data and other services from the server computers and server computers provide services to the client computers by responding to client computers requests. Normally servers are powerful computers and clients are less powerful personal computers. Web servers are included as part of a larger package of internet and intranet-related programs for serving e-mail, downloading requests for FTP files and building and publishing web pages.

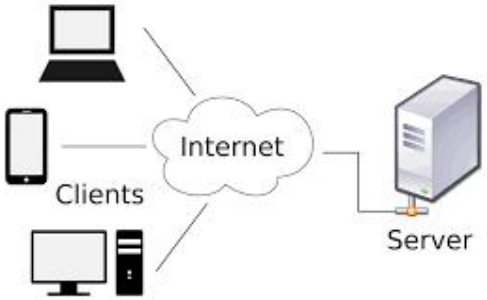

fig: Client-Server Architecture

Advantages of client-server architecture:

- It reduces network traffic by providing a query response to the user rather than transferring total files.
- All the files are stored at server. So that, management of files becomes easy. Also it becomes easier to find files.
- Easy to implement security policies.
- From various platforms in the network, server can be accessed remotely.
- The client/ server model improves multi-user updating through a graphical user interface (GUI) front end to the shared database.

Disadvantages of Client-server architecture

- Failure of the server causes the whole network to be collapsed
- Expensive to install and manage this type architecture
- need professional to maintain the servers and other technical details of the network.
- When there are too many requests from clients, the server gets overloaded.

Client/Server architecture can be of different model based on the number of layers it holds. Some of them are:

2-Tier Architecture

It is used to describe client/server systems where the client requests resources and the server responds directly to the request, using its own resources. This means that the server does not call on another application in order to provide part of the service. The direct communication takes place between client and server.

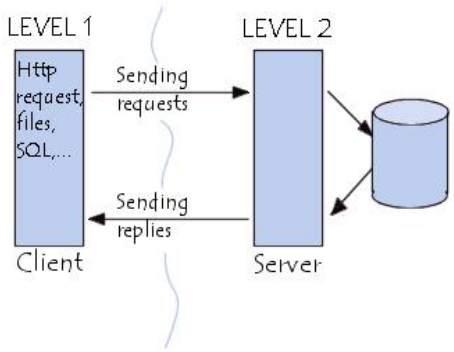

Fig:2 tier architecture

The Two-tier architecture is divided into two parts:

1. **Client layer (Client Tier)**
2. **Data layer (Data Tier)**

On client application side the code is written for saving the data in the SQL server database.Client sends the request to the server and it processes the request & sends back with data. The main problem of two-tier architecture is the server cannot respond multiple request same time, as a result it cause a data integrity issue.(software testing class)

Advantages:

- Easy to maintain and modification
- Communication is faster

Disadvantages:

- In two-tier architecture, application performance will be degraded upon increasing the users.
- cost ineffective

3-Tier Architecture

In 3-tier architecture, there is an intermediary level, meaning the architecture is generally split up between:

1. Client layer
2. Application layer
3. Data layer

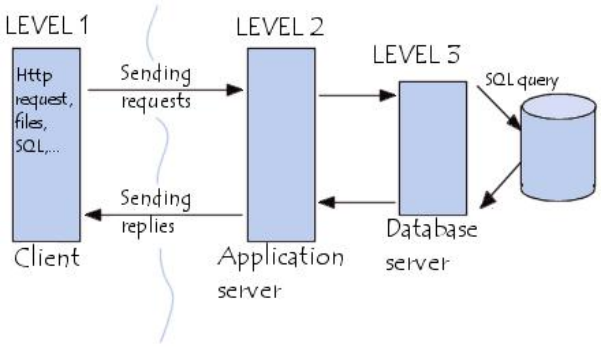

Fig:3- tier architecture

Client layer which contains UI part of our application. This layer is used for the design purpose where data is presented to the user or input is taken from the user.

In **application layer** all business logic written like validation of data, calculations, data insertion etc. This layer acts as an interface between Client layer and Data Layer. This layer is also called the intermediary layer helps to make communication faster between client and data layer and

In **data layer** actual database is comes in the picture. Data Layer contains methods to connect with database and to perform insert, update, delete, get data from the database based on our input data.

Advantages:

- client doesn't have direct access to the database business logic is more secure
- Managing data is independent of the physical storage
- possible to make changes on the one layer without affecting the other two
- When one tier fails there is no data loss, because you are always secure by accessing the other tier.

Disadvantages:

- It is more complex structure
- difficult to built a 3-tier application
- difficult to set up and maintain

N-Tier Architecture

In N-tier architecture (with N more than 3) the middle tier is split up into new tiers. The application tier is broken down into separate parts. What these parts differ from system to system.

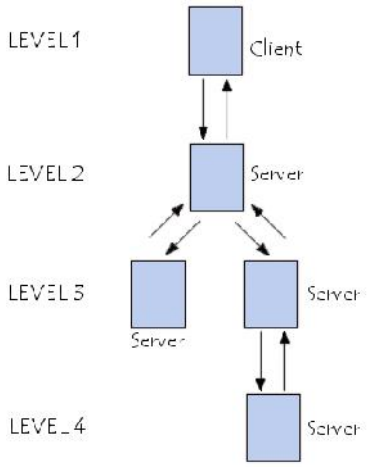

fig: N-Tier Architecture

Advantages:

- Changes to the user interface or to the application logic are largely independent of one another, allowing the application to evolve easily to meet new requirements.
- Network bottlenecks are minimized because the application layer does not transmit extra data to the client, only what is needed to handle a task.
- The client is insulated from database and network operations. The client can access data easily and quickly without having to know where data is or how many servers are on the system.
- Database connections can be 'pooled' and thus shared by several users, which greatly reduces the cost associated with per-user license
- The organization has database independence because the data layer is written using standard SQL which is platform independent. The enterprise is not tied to vendor-specific stored procedures.
- The application layer can be written in standard third or fourth generation languages, such as ASP, PHP with which the organization's in-house programmers are experienced.

Disadvantages:

- It is more difficult to program and test an N-tier architecture due to its increased complexity.

Things to remember

- Client-server architecture consists of two kinds of computers: clients and servers. client request for service and server provide service to the client.
- Client/Sever architecture can be of different model based on the number of layers it holds. Some of them are: 2 tier ,3 tier and n tier .
- In 2 tier Architecture the client requests resources and the server responds directly to the request, using its own resources.i.e,The direct communication takes place between client and server.
- The Two-tier architecture is divided into two parts: client layer and Data layer.
- In 3-tier architecture, there is an intermediary level, meaning the architecture is generally split up between:
    1. client layer
    2. application layer
    3. Data layer
- client layer contains UI part,application layer contains business logic.and data layer contains actual database.
- In N-tier architecture middle tier is split up into new tiers. The application tier is broken down into separate parts. What these parts are differs from system to system.