Modern Web Technologies

Leonidas Akritidis, Dimitrios Katsaros and Panayiotis Bozanis

Department of Computer & Communication Engineering, University of Thessaly, Volos, Greece

Abstract. Nowadays, World Wide Web is one of the most significant tools that people employ to seek information, locate new sources of knowledge, communicate, share ideas and experiences or even purchase products and make online bookings. The technologies adopted by the modern Web applications are being discussed in this book chapter. We summarize the most fundamental principles employed by the Web such as the client-server model and the http protocol and then we continue by presenting the current trends such as asynchronous communications, distributed applications, cloud computing and mobile Web applications. Finally, we conduct a short discussion regarding the future of the Web and the technologies that are going to play key roles in the deployment of novel applications.

1 Introduction

During the past few years we have witnessed a massive evolution in the applications hosted on the World Wide Web. The obsolete, static Web sites have been replaced by innumerable, novel services that changed dramatically the manner that users navigate, purchase, communicate, think and make decisions. New types of dynamic applications have been developed by using the modern technologies and their participatory features have made them extremely popular, since hundreds of millions of people use them on a daily basis. Blogs, social networks, forums, search engines, wikis, media sharing services and office suites are only a small subset of these applications, which are collectively known as *Web2.0*.

Therefore, understanding the technologies that support the continuous expansion of the Web is of significant importance. Since the field of Web technology constantly changes and evolves, researchers and developers are facing the challenge of being early informed in order to investigate and propose novel solutions to newly posed problems. Note that some of the techniques characterizing the Web since its birth still exist and development must always obey to the traditional rules set by them. One representative example of these technologies is the HTTP protocol which, essentially, has been left unchanged since 1995.

However, the majority of the technologies have either been evolved, or completely replaced by novel ones. In this chapter we present the state-of-the-art technologies that are now vital in modern Web computing. Robust design and efficient development of systems and applications deployed on the Web is a topic

of critical importance, since it determines the acceptance of a system by the users and affects the commercial success of the product. Here we discuss such topics along with some of the tools and issues involved in this development.

At first, in Sections 2 and 3 we describe the basic networking models adopted by the developers when building Web applications. The Client-Server model is a basic computer networking architecture, which was established before the Web explosion and it sets two types of devices; clients and servers. On the other hand, the Peer-To-Peer model was developed later in order to interconnect users and every device in such a network is treated equally.

Next, we present some of the basic characteristics of *hypertext*, the most popular manner of publishing and distributing information across the entire Internet. At first we provide some definitions and then we discuss how hypertext is transferred and formatted by using HTTP (Section 5) and HTML (Section 6), respectively.

The description of the basic principles of the XML language follows in Section 7. XML is a tool gaining rapid acceptance by both users and developers and it is mainly employed in order to transfer information in a fast and effective manner across different platforms. It is definitely becoming the method of choice for the most modern Web applications such as news portals, blogs, forums and even electronic stores publishing their product lists.

Javascript is one of the most popular scripting languages encountered on the Web and millions of Web sites and applications use it for various purposes. In Section 8 we present the basic characteristics of the scripting languages and especially Javascript. A family of modern technologies, AJAX, is later described in Section 9. Nowadays, AJAX is a quickly expanding tool, since it offers important solutions to the stateless nature of the HTTP and the traditional Client-Server model. By submitting requests to a Web server asynchronously, the AJAX-enabled Web pages can modify their content without requiring to refresh their display. This family of technologies which is one of the most important Web2.0 features, employs Javascript for scripting and XML for transferring data between the client and the server. AJAX is currently being used by numerous Web services, such as email managers, global and planetary mapping services, instant messengers, Web search engines and others.

Finally, we discuss how Web applications are constructed and deployed. In Section 10 we present some of the most popular programming and storage tools and in the sequel, we describe some characteristic applications of the Web, such as social networks, Web communities, office suites, and mobile software.

Since the number of people using the Web constantly increases, one of the main trends encountered today is the utilization of large clusters of computers in order to handle the tremendous workloads. The first category of these applications, discussed in Section 12, includes software which is distributed across the computers of the users themselves and exploit their free resources in order to solve complex scientific problems. In the second category, we mainly encounter the popular *cloud computing* solutions (Section 13), which employ thousands of

interconnected servers usually hosted in one or more data centers with the aim of addressing the huge traffic that millions of users produce.

Concluding in Section 16, we provide a brief discussion regarding the presented technologies and the future of the Web.

2 The Client-Server Model

The Client-Server model is one of the most popular architectures for computer networking. It utilizes two types of devices to address the communication requirements of the terminals of a network; Clients and Servers. The model can be used on the Internet as well as local area networks. Examples of client-server systems on the Internet include email services, Web browsers, Web servers and FTP clients and servers. It was originally developed to allow multiple users to share access to database applications and offers improved scalability because connections can be made as needed, rather than being fixed.

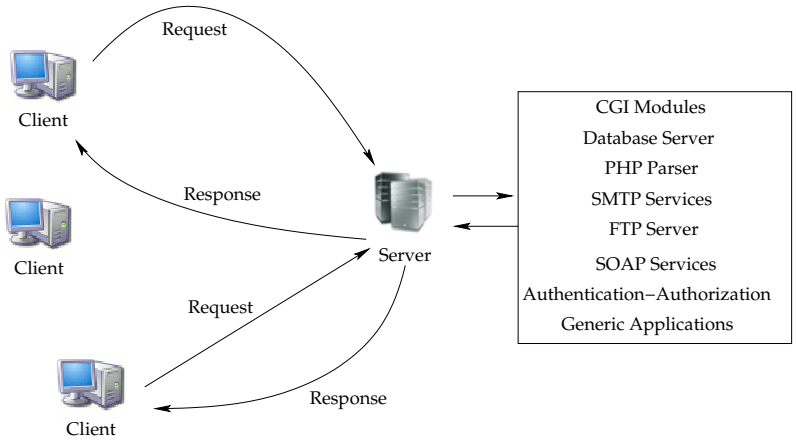

Fig. 1. Typical Architecture of the Client-Server Model.

A Client is a device, typically a personal computer or a mobile device with network software applications installed that request and receive information over the network. On the other hand, a Server typically hosts applications (including other servers), or stores files and databases. Server devices often feature higher-powered equipment, greater processing performance and larger storage capabilities than clients.

Figure 1 illustrates a typical communication of several remote machines according to the Client-Server model. Network clients submit requests to a server machine by transmitting messages. On the other side, servers respond to their clients by processing each request and by sending the appropriate response messages. In a typical Client-Server environment one server is utilized to support

the traffic originated by multiple clients, whereas numerous servers can be linked together in order to handle and address increased processing load.

The Client-Server model is employed by some of the most widespread applications on the Internet. Such applications include FTP, email and Web services and in each of these, the clients employ software of special type (an Internet Browser, an email management program or an FTP client), allowing them to connect and receive data from the corresponding servers.

3 The Peer-To-Peer (P2P) Model

Peer-To-Peer networks, often abbreviated to P2P, is a network architecture alternative to the Client-Server model. However, instead of requiring central coordination by a machine or software such as a server, all the devices (peers) are treated equally. In such an architecture, every peer makes a portion of its resources directly available to other peers, hence each network participant is both a supplier and consumer. The resources that the peers may share include network bandwidth, processing power and disk storage.

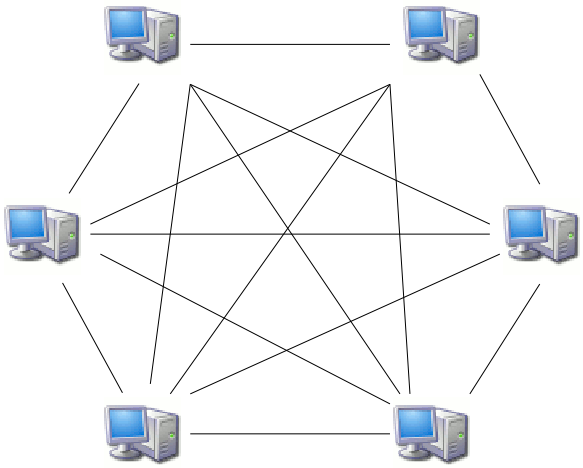

Fig. 2. Typical Architecture of a Peer-To-Peer network.

This network architecture became really popular when file sharing services (such as Napster, kazaa eMule and others) appeared on the Internet. Their distributed architecture and the fact that no server is required, provides improved scalability, since computers may dynamically enter or leave the network without significant impact.

Depending on how the connections among the peers are established, Peer-To-Peer networks are divided into two categories: The structured and the unstructured networks. The first category includes topologies where the connections are

fixed, whereas in the second category we encounter architectures that do not provide any algorithm for organizing or optimizing these connections.

The most popular application employing Peer-To-Peer technologies is the file sharing services, where millions of users join a community in order to request or provide media files, documents and software. There are also other important types of applications such as instant messaging, online chat and voice over IP where this network architecture finds remarkable utilization.

4 Hypertext

Currently, the information on the Web is mainly provided through documents of special type, which provide inter-linking capabilities. Consequently, every Web document (also known as Web page) may contain references to other documents and resources and the user is able to immediately access these resources by simply following the corresponding links.

All this functionality is provided through *hypertext*, a term coined by Ted Nelson around 1965. Hypertext is a specific form of text containing dynamic references to other resources. Although it is an old invention, it still remains the main way of information propagation within the Web. Apart from running simple text and links, hypertext may also contain headings, lists, tables, images and other presentational devices.

Other means of interaction could also be present, such as a form to complete and submit.

5 Hypertext Transfer

The requirement for retrieving inter-linked resources containing hypertext, led to the establishment of *HTTP (HyperText Transfer Protocol)*. The utilization of HTTP imposed a set of rules and specifications allowing hypertext documents to be transmitted and received. It is a generic, stateless protocol which can be applied on many tasks beyond its use for hypertext, such as name servers and distributed object management systems.

The specifications of the protocol obey the request-response standard, which is typical in client-server computing. The client is usually an application (i.e. the user's browsing software) running on the machine of an end user, whereas the server is a computer program executed by the machine hosting the Web site or service. The communication is conducted through HTTP *requests* submitted by the client and HTTP *responses* returned by the server. The responses are accompanied by specific code numbers which indicate the status of the corresponding response. In other words, response codes reveal whether the request has been served successfully, or whether an error has occurred (including the type of the error).

HTTP is usually implemented on top of the TCP/IP protocol which controls the reliable data transfers. However, this is not a constraint, since HTTP can operate by using any other protocol which guarantees reliable transports.

Nowadays, there are two major versions of the protocol, HTTP/1.0 and HTTP/1.1. According to [13] the latter improves the former by providing a more efficient caching mechanism, and more effective bandwidth usage (the introduced range requests allow a client to request portions of a resource). Furthermore, while HTTP/1.0 included some support for compression, it did not provide adequate mechanisms for negotiating its use, an issue addressed by the newer version.

6 Hypertext Markup

The requirement for building Web documents of unique and personalized style, layout and formatting, led to the introduction of a special markup language which can be used by Web developers to construct their pages according to their personal preferences. Moreover, the requirement for publishing information that can be globally distributed, made the need for a universally understood language imperative.

HTML, which stands for HyperText Markup Language, is now the predominant markup language for Web pages. It provides a set of command (*HTML tags*) to create structured documents by denoting structural semantics for text. Such semantics include tables, paragraphs, headings, lists, links and numerous others. The need for interaction between a Web application and the user is mainly satisfied by the introduction and usage of forms. These forms are now used for conducting transactions with remote services, making reservations, ordering products, etc

Finally, HTML allows the integration of scripts which are usually executed by a Web client (browser) and Cascading Style Sheets (CSS) which is a standard of parallel commands for explicit presentational markup.

There are several releases of HTML, but the most popular among them is HTML 4.01 [1] also recommended by the W3C. In addition to the text, multimedia, and hyperlink features of the previous versions, this version supports more multimedia options (i.e. embedded videos), scripting languages, style sheets, more printing facilities, improved support for right to left and mixed direction text, frames and enhancements to forms, offering improved accessibility for people with disabilities. HTML 4.01 is also more oriented towards the internationalization of the Web documents since they can be written in every language and be transported easily around the world.

7 XML

As we aforementioned in Subsection 6, HTML was designed to display Web pages with focus on the manner that the data is presented to the user. The need for a language that would focus on transportation and storage of data led to the introduction of *Extensible Markup Language* or *XML*. Note that XML is not a replacement for HTML, since the latter is mainly designed for displaying

information, whereas XML was created to structure, store, and transport information. The precise design goals of the XML as set by the World Wide Web Consortium [2] emphasize on the wide, fast and easy usability that the standard must provide.

XML is a simple, very flexible text format derived from SGML (Standard Generalized Markup Language). It was originally designed in order to provide convenient, uniform and platform-independent publishing of information. Nowadays it is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere.

The documents of this type are composed of markup and content and have both a logical and a physical structure. There are six kinds of markup that may occur in an XML document: elements, entity references, comments, processing instructions, marked sections, and document type declarations. An entity may refer to other entities to cause their inclusion in the document and furthermore, the logical and physical structures must nest properly.

To obtain the data contained in an XML document, we usually employ especially designed applications known as *XML parsers* which are capable of translating the structure of the document and generating data nodes. There are many types of XML parsers with various capabilities. One of their most remarkable feature is their ability to examine whether the document is well formed. In general, XML parsers are more strict than HTML renderers and the most sophisticated among them perform *validation* of the XML structure and syntax. Hence, a document which does not conform to the XML grammar or does not contain a proper document type declaration, is not considered valid and cannot be parsed.

7.1 RSS Feeds

Really Simple Syndication (or *RSS*) is one of the most common applications of the XML language. It is a family of *feed* formats mainly employed by authors to publish frequently updated information. Such information includes news headlines, multimedia content, blog posts, product catalogs (accompanied by the corresponding availability data and prices), airline tickets and numerous others. The RSS documents are built by using standard XML syntax and their popularity increased rapidly after 2005.

An RSS document (which is called a *feed*) includes text and meta-data such as publishing dates, authorship and others. The generation and delivery of information by using such feeds is beneficial for both publishers and readers. More specifically, publishers are allowed to syndicate content automatically, whereas readers are able to subscribe to temporal updates from preferred sites, or to aggregate different RSS feeds originating from multiple sources.

A reader can access and extract the information stored in RSS feeds by using specific software called an *RSS reader* or *aggregator*. Most modern Web browsers include built-in RSS readers, whereas some new mobile devices have their own reading software available. A standardized XML format allows the information to be published once and viewed by many different programs. The user subscribes to a feed by informing the corresponding reading software about

the desired feed. In turn, the reader checks the subscribed feeds regularly for new publications and updates and informs the user accordingly.

8 Scripting

HTML is a powerful markup language offering the Web developers various tools in order to format the Web pages that they build. However, it lacks programming capabilities such as setting variables, computing values, handling files etc. To cover this drawback, HTML provides the option of composing external *scripts* that is, fragments of code written by programmers in order to solve specific problems.

Scripts can be separated into two main categories: *server-side* and *client-side*. The first category includes programs executed by the server and only the result of the programm processing is returned to the end user. Furthermore, this type of scripts may, or may not require compilation before they can be executed. For example, PHP scripts do not require compilation, but there is a dedicated parser running as a server module that handles runtime or syntax errors. Therefore, in this case, no executable file is created, but the opposite holds for server-side programs written in other programming languages (i.e. ASP.NET).

In the second category we encounter scripts executed by the client (that is, the user's browser) and the code is integrated within the Web page itself. All client-side scripts do not require compilation or parsing. This has the cost of harder and slower debugging, since when an error is triggered, no informative messages are generated and the execution is being terminated silently. Therefore, the programmer has to examine and debug the entire code in order to detect the specific portion causing the problem.

Nowadays, the most popular language on the Web offering client-side scripting is *Javascript*. Javascript is used in millions of Web pages to add functionality, validate form data, build visual effects, open pop-up windows and there are many other applications. Another significant field, where it finds application, is the development of enhanced user interfaces and dynamic Web sites.

Furthermore, it offers event-driven programming that is, it allows the programmer to write code which is executed before or after a specific event is triggered. Hence, a Web page may be modified without having to send any data to the server and receive its response. For this reason, Javascript is ideal for building robust Web applications with modern user interfaces, and it is one of the main characteristics that the services of the next generation of Web include.

9 Asynchronous Transfers and AJAX

The usage of the traditional request/response methods that the classic client-server model imposed, prevented Web sites and browsers from providing a fast and responsive user experience. For example, filling and submitting an online form was inconvenient and time consuming, since all the requested information had to be entered and then submitted to the machine hosting the service (Web

server). Then the server performed a validation of the form data and if problems were detected, the user was obliged to refill and resubmit the same form. The flow of information and the resulting experience was inconstant and disconnected, reflecting the stateless nature of HTTP.

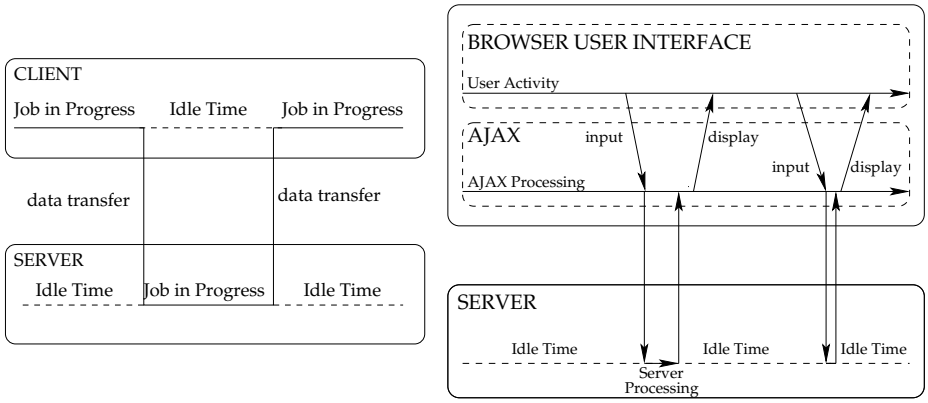

Fig. 3. AJAX asynchronous data transfer model (right) vs traditional client server workflow model (left).

The introduction of Java applets offered a different route: this route included asynchronous loading of content and allowed client-side code to load data asynchronously from the Web server after a web page was fetched. Moreover, the IFrame component of the HTML language and some introduced ActiveX controls also enabled this to be achieved. More specifically, these ActiveX controls included a special object, namely XMLHttpRequest[1], which was designed in order to submit requests to a server asynchronously.

AJAX (shorthand for *Asynchronous JavaScript and XML*) is a modern Web technology introduced by the World Wide Web Consortium (W3C) in 2006[2]. It is a group of interrelated web development techniques used on the client-side to create interactive web applications. The applications employing AJAX technologies are capable of retrieving data from the server asynchronously in the background, without having to interfere with the display and behavior of the currently loaded page.

In Figure 3 we illustrate how the usage of AJAX technologies in modern Web applications offers uninterrupted user experience. The left diagram of this Figure depicts how the user and the remote server communicate according to the traditional client-server model. Note the idle times in the side of the client; these are the times required to transfer the desired data between the client and the server and also concern the time the server consumes to process this data

[1] http://www.w3.org/TR/XMLHttpRequest/
[2] http://www.w3.org/standards/webdesign/script

and return a response to the end user. In the AJAX environment, the user experiences no idle times, since the asynchronous communications between the browser and the server offer continuous work flow.

In the sequel, let us provide some of the main benefits deriving from the usage of AJAX techniques. At first, in many cases, some pages on a Web site include content that is common among them. The usage of traditional methods required that this content would have to be reloaded on every request. However, by employing AJAX an application can request only the content that needs to be updated. Therefore, we manage to drastically reduce both the usage of valuable bandwidth usage and load time.

In addition, the utilization of asynchronous requests allows the interface of the client's browser to be more interactive and to respond quickly to inputs. Several portions of pages can be reloaded individually and the users may perceive the application to be faster or more responsive, even if the application has not changed on the server side.

Finally, with AJAX we can minimize connections to the Web server, since external files such as scripts and style sheets only have to be requested once. Programmatically, this means that the local variables will retain their values, because the main container page need not be reloaded.

For all these reasons, the usage of these techniques has led to a significant increase to the applications providing interactive and dynamic user interfaces[2][3]. Some of the most common services employing AJAX techniques are:

- Mailbox management applications, where the entire user interface is designed to allow composition, reading and deletion of messages without refreshing the display. Moreover when a new message arrives it is added to the in-box automatically without requiring the user to refresh the page.
- The new technology allowed the introduction of modern Web instant messengers. These services are constructed in such a way that allow their users to exchange their messages instantly. Their main characteristic is that each time a message is sent or received, only its content is loaded by the client and the entire interface remains unchanged.
- Global Maps Services employ the asynchronous features of AJAX to allow their users navigate through the surface of the planet. They also provide magnification potentials by directly accepting data from satellites.
- The novel translation services now operating on the Web offer their users new functionality. They are capable of accepting words or even sentences and paragraphs written in a specific language and, as the user types, they translate the content into another language.
- A huge amount of other smaller services is now built by using asynchronous technologies. Such services include result retrieval in the major commercial search engines, spelling correction, auto-complete features (as the user types his/her query, current search engines fetch similar entries from their query logs and present them on the fly below the text box),

Nevertheless, the remarkable new features and functionality introduced by the AJAX technologies do not come without costs. The main drawback is that

the interfaces constructed by using AJAX are substantially more difficult to develop than static pages. Pages dynamically created using successive AJAX requests do not automatically register themselves with the browser's history engine and this may raise problems regarding the user's navigation on the Web. For example, it is possible that when a user clicks the "Back" button of the Web browser, he/she will not return to an earlier state of the AJAX-enabled page, but may instead return them to the last full page visited before it. Dynamic web page updates also make it difficult for a user to bookmark a particular state of the application.

Web crawlers are computer programs developed by the search engines in order to browse the Web in a methodical, automated manner. However, since the majority of the Web crawlers do not execute Javascript code, applications indexed by search engines should provide means for accessing the content actually retrieved with AJAX. Note that any user whose browser does not support Javascript or XMLHttpRequest, or simply has this functionality disabled, will not be able to properly display and use pages which depend on AJAX. Similarly, devices such as mobile devices and screen readers may not have support for the required technologies.

Finally, like other web technologies, AJAX has its own set of vulnerabilities that developers must address. Developers familiar with other web technologies may have to learn new testing and coding methods to write secure AJAX applications.

10 Application Deployment

In this subsection we present some of the main tools and technologies employed by the modern Web applications.

10.1 Database Servers

Currently, XML is the dominant technology to publish and distribute semi-structured information on the Web. However, there are types of applications that require their data to be organized in a more robust and structured manner. Such applications include electronic stores, social networks, forums, search engines and others which usually have to deal with massive amounts of data.

Database Management Systems (or DBMS) is a tool developed to offer efficient organization, storage, management and retrieval of an application's data. These systems usually reside in dedicated machines and offer database services to other computers and applications. Instead of having to write computer programs to store and extract information, user can ask simple questions in a supported query language. Thus, many DBMS packages provide a structured query language (SQL) and other application development features.

Within a typical DBMS, data is organized into *records* which is a collection of data regarding a physical entity (i.e. an employee, a book, or a product). Each record consists of numerous user-defined *fields*, that are able to store information

of different types (text, binary data, integers and floats, time stamps, dates and several others which vary across different DBMSs). Records of the same type are again grouped within *tables*. Databases provide an efficient manner of separating the application logic from the data logic, therefore, different applications can cooperate with the same database.

One of the most important characteristics offered by Database Management Systems is the indexing feature. An index is an auxiliary data structure usually implemented in a form of a tree such as B-Tree, to allow fast and efficient data access and retrieval. The indexes also allow effective sorting of the returned records and offer fast organization (i.e. grouping of records).

Other features commonly offered by database management systems include:

- Restricted access to resources and attributes. Each user of the system is assigned privileges which determine whether he or she has read or write access to several resources and attributes of the database. These privileges are assigned by individuals, or groups of individuals maintaining elevated authority across the entire system.
- Data safety and integrity are of critical importance for every informational system, hence copies of attributes are required to made regularly in case of equipment failure. DBMS usually provide utilities to facilitate the process of extracting and disseminating attribute sets.
- Data retrieval by submitting queries. Instead of composing special software to obtain and format the data stored within a database, most modern systems accept structured queries which usually follow the simple syntax of a structured query language. By submitting queries we request attribute information from various perspectives and combinations of factors (i.e. who are the male clients that purchased a specific product?). Queries can also be submitted to the database in order to insert, update or delete data, according to the privileges each user is granted.

The introduction of World Wide Web in 1995 imposed new challenges for database systems. Researchers realized that the traditional database management techniques were becoming too complex and there was a need for automated configuration and management. For example, online transactions have become extremely popular with the evolution of the e-business world. Consumers and businesses are able to purchase products and make payments securely on corporal Web sites.

In addition, Web search engines have even been remarkably influenced by database management. Using technologies similar to the ones employed by current database systems, these services are able to accept user queries and locate data across the entire the Web.

10.2 Hypertext Preprocessor - PHP

PHP is one of the most widespread scripting languages used to deploy Web applications. The rich features it offers combined with the natural easiness and

the open source characteristics, have made it the second most popular scripting language encountered on the Web [4]. Although there is a general intuition that PHP is mainly preferred for constructing small or medium sized applications, several large-scale Web sites serving hundreds of millions of users worldwide, such as Facebook, Wikipedia and Wordpress have been developed with it. Currently, PHP is installed on over 20 million sites and 1 million Web servers [3].

It was originally designed for the development of Web applications, in order to produce dynamic pages. PHP scripts can be embedded into HTML and they generally run on a Web server (server-side scripting), which needs to be configured properly to execute PHP code. It can be deployed on most Web servers and on almost every operating system and platform.

PHP scripts are phrased by following a C-style coding syntax and all the allocated resources are released after the script execution by an automatic garbage collection mechanism. Since its fifth version, it also supports the object-oriented programming style by adopting principles such as abstract data types and information hiding, inheritance and polymorphism. Moreover, it includes features such as variables, arrays and associative arrays setting and manipulation, conditional statements, loops, function setting and file handling. Apart from these classic characteristics, PHP allows programming of the HTTP protocol by providing access to HTTP sessions and cookies and, furthermore, by implementing secure file uploads.

One of the most robust features of PHP is its native support to MySQL, SQLite and PostgresSQL database systems. Through built-in functions and classes, PHP scripts can easily connect to database servers, submit queries and retrieve data. The combination of PHP and MySQL is one of the most common techniques currently employed by the developers when building Web applications.

10.3 Active Server Pages - ASP/ASP.NET

Another popular server-side scripting technology that is competent to PHP is *Active Server Pages*. It has been introduced by Microsoft and provides to the developers robust tools in order to create dynamic and interactive Web applications. Similarly to the PHP documents, an ASP page is a standard HTML document which contains server-side scripts. The scripts are processed by a properly configured Web server which sends the processing output to the user's browser.

In contrast to PHP, ASP is not a scripting language, but rather a technology used to produce dynamic pages when a browser requests ASP files from a Web server. The default scripting language employed for scripts composition is VBScript, although alternative languages like JScript (Microsoft's version of Javascript) can also be used. When an ASP script is called, the server processes the requested file from top to bottom and executes the commands it contains. It the sequel, it generates and formats a standard Web page and sends it to the browser.

During 2002, Microsoft released a large set of coded solutions to common programming problems. This library, known as the *.NET Framework*, includes

solutions regarding user interface design, data access and processing, database connectivity and development of dynamic Web applications, whereas the programmers are able combine its classes with their own work.

In addition, the library includes a virtual engine able to execute the software written specifically for the framework. The applications developed with the .NET Framework are deployed in a special environment which manages their runtime requirements. This runtime environment, which is known as the *Common Language Runtime (CLR)*, allows the programmers to work without considering the capabilities or the specifications of the specific machine that will execute their program. The CLR also provides other important services such as security, memory management, and exception handling.

Along with the release of the .NET Framework, Microsoft also introduced ASP.NET, an enhanced version of ASP used to produce dynamic Web sites, applications and services. The new development framework is built upon the CLR and allows programmers to compose software by employing any of the supported languages such as the VB.NET (Visual Basic .NET), C#, J# and others. Moreover it offers the ability to construct applications by using an event-driven user interface model, in contrast to the conventional scripting environments such as ASP and PHP.

10.4 Java Server Pages - JSP

Java Server Pages or simply *JSP*, is another technology used to deploy dynamic Web applications, by allowing Java code to be embedded into the content of a regular static page. The code is not pre-compiled, but it is actually being compiled on the server at each page request similarly to PHP. The Web pages that are created by using JSP are loaded in the server and handled by a special Java server packet, called the *J2EE* Web Application.

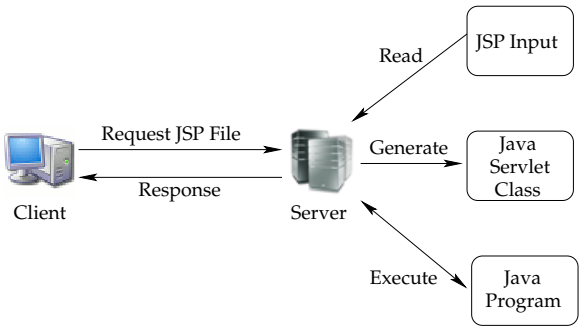

Fig. 4. JSP page translation and processing phases.

The processing of the a JSP page is performed in two phases. At first, we employ a typical JSP compiler which converts the input file into a *servlet*, that

is, a particular Java class that responds to HTTP requests. In the sequel, the servlet can be either compiled by a Java compiler and generate a standard Java program, or be converted to a directly executable byte code. Figure 4 illustrates the procedure of translating and processing JSP pages.

JSP is currently an alternative method to PHP and ASP, allowing developers to construct dynamic Web sites and services by writing their code in Java. Although it provides rich features and offers almost equivalent possibilities, it is not as popular as the other two aforementioned technologies.

11 SOAP

A protocol which gained attention during the past few years is the *Simple Object Access Protocol*, or simply SOAP. It is a simple XML-based protocol which allows the applications to exchange structured or semi-structured information over HTTP. Its messages follow the standard XML syntax, whereas the trasmission/receive procedure is handled by other application protocols, such as the HTTP.

SOAP messages consist of three parts:

- An envelope which describes the content of the message and instructions about how this content should be processed,
- a set of rules containing the data types defined within the application and
- a convention which represents procedure calls and responses

The main advantage of SOAP is that it allows easier and more robust communication between proxy servers and firewall applications along with the language and platform independence. In addition, the transmitted packets not only include the content of the message, but also sufficient information describing how this content should be processed by the receiver. However, the verbose XML format can render SOAP relatively slower than other solutions.

Since one of the most common purposes of Web services is to exchange XML data, SOAP is rapidly becoming the generally accepted protocol for XML-based systems communication. For example, Web search engines APIs make wide use of SOAP. In addition, numerous stock quote services, weather services or news portals, employ it in order to transmit and receive data formatted in the XML language.

12 Distributed Applications

Distributed computing is one of the most discussed and hot topics in computer science. It refers to partitioning a large or complex problem into several smaller parts and assigning each of these parts to a machine that belongs to a wider cluster of processing nodes (also called workers). When each of the processing nodes finishes its computations, the distributed solutions are merged to form the final solution of the problem. In such a distributed environment, a central

coordinator is usually employed in order to synchronize and send messages to the processing nodes.

Of course, distributed computing is not a pure Web technology. However, there are some projects which utilize the machines of the Web users in order to solve large scientific problems. These projects exploit the free (or idle) resources (mainly the processing power) of thousands or even millions machines of Web users in order to compute the solution of a small fraction of a huge problem.

Folding@home is one of the world's largest distributed computing projects developed with the official goal of "understanding protein folding and related diseases". It does not rely on powerful supercomputers for processing the available data; instead, the primary contributors to the project are many hundreds of thousands of personal computer users who have installed a client program. The client runs in the background, utilizing the unused resources, whereas it periodically connects to a server in order to retrieve new data and continue the calculations, or send back the produced results.

Seti@Home is a similar project which exploits the computers of Web users with the aim of performing Search for Extraterrestrial Intelligence (SETI) by analyzing radio signals. Similarly to the Folding@Home project, the users download and install a client software which is capable of processing data generated by radio telescopes. The client exploits the unused resources of the machine it is installed on and proves the viability and practicality of the distributed grid computing concept.

13 Cloud Computing

Cloud computing is a recent trend in Computer Science that moves computing and data away from desktop and portable PCs into large data centers. It refers to applications delivered as services over the Internet, as well as to the actual cloud infrastructure. Currently, the main technical characteristics of cloud computing services include virtualization, grid computing technologies, service-oriented software management of large facilities and power efficiency.

Within a cloud computing environment, applications and services are provided in the form of platform-as-a-service (PaaS), software-as-a-service (SaaS), or infrastructure-as-a-service (IaaS). We can distinguish two different architectural models for the clouds: the first one is designed to scale out by providing additional computing instances on demand. Clouds can use these instances to supply services in the form of SaaS and PaaS. The second architectural model is designed to provide data and compute-intensive applications via scaling capacity. In most cases clouds provide on-demand computing instances by adopting a "pay-as-you-go" economic model.

Cloud computing is a disruptive technology with profound implications not only for Internet services but also for the entire IT field. Its emergence promises to streamline the on-demand provisioning of software, hardware and data as a service, achieving economies of scale in IT solutions' deployment and operation.

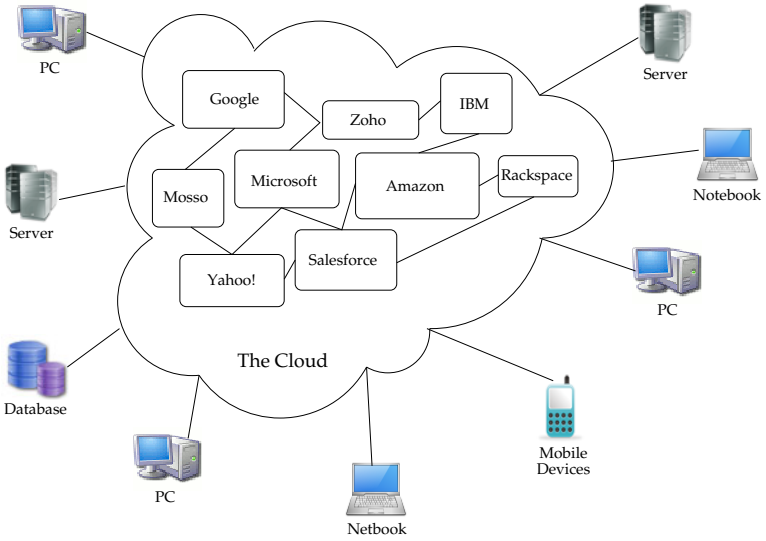

Fig. 5. The Cloud.

Still, several outstanding issues exist, particularly related to service-level agreements (SLAs), security and privacy, and power efficiency. Other open issues include ownership, data transfer bottlenecks, performance unpredictability, reliability and software licensing issues. Several companies have already built Internet consumer services such as search, social networking, Web email, and online commerce that use cloud computing infrastructure.

14 The Mobile Web

During the past few years, the mobile devices have played a key role in the market of telecommunications. By offering significant features such as small sizes, light weights, efficient power consumption, ability for direct user-to-user communication, affordable prices and remarkable processing power, they have attracted hundreds of millions of consumers. The market of mobile devices is expected to experience an impressive growth over the following few years [5] and recently, numerous manufacturers have included Web browsing capabilities to their products.

According to the inventor of the Web Tim Berners-Lee, "the Mobile Web initiative goal is to make browsing the Web from mobile devices a reality". By employing browsers particularly designed for mobile devices (mobile browsers), mobile Web access is becoming increasingly popular. Since a large desktop system is not required any more to access the Web, users are provided with the ability to work online at all times in all situations. Curren mobile Web browsers retain the main functionality offered by the respective desktop applications, such as basic browsing, form completion and submission and generic transactions.

However, there are still some problems that need to be confronted and interoperability is the most significant among them. It derives from the the existence of many different platforms with various operating systems and browsers. In addition, the limited size of these devices and the small display sizes raise important usability issues.

Within the Mobile Web, the information is published and delivered via lightweight pages written in XHTML or WML (Wireless Markup Language). The new versions of the mobile browsers raise these limitation by supporting a wider range of Web formats, including variants of HTML commonly found on the desktop Web. In addition, W3C have published a set of recommendations [6] to Web site creators and developers who desire their applications to be fully accessible from mobile devices.

15 Web 2.0 Applications

Web 2.0 is a widespread term which reveals the evolution we have witnessed in the World Wide Web and the applications hosted on it. The definition of Web 2.0 [10] does not refer to an update to the technical specifications characterizing the Web, but rather to fundamental changes in the manner that application developers and users exploit it. Therefore, the Web is currently treated as a platform, where new applications are built upon it, similarly to how applications are developed and deployed upon the desktop platform.

The new version of the Web is usually connected with web applications that offer participatory and sharing features. More generally, the main characteristic that a Web 2.0 application has, is its user-centered design. That is, the information is not simply provided to the users, but the users contribute to it by expressing and publishing their own knowledge, experiences and opinion. In such an environment, the users are not limited in a traditional passive role, but they dynamically determine the content of a Web site.

This design has led to the introduction of numerous novel services such as Web communities, social networks, media sharing services, wikis, blogs, forums, online auctions and numerous others. In this Section we provide brief descriptions of the most popular Web 2.0 sites, applications and services. Of course, we do not intend to provide a complete directory of the most significant Web 2.0 applications, but we rather exhibit some of their major features which made them extremely popular during the past few years.

15.1 Web Communities

The novel features that Web 2.0 introduced resulted in an significant growth of Web locations where people connect and share ideas, knowledge and experiences. Such locations are currently known as Web communities and have became extremely popular during the past few years. Examples of Web communities include social networking sites, forums and community blogs.

Blogs are locations on the Web where individuals (the bloggers) express opinions or experiences about a subject. Such entries are called blog posts and may contain text, images, embedded videos or sounds and hyperlinks to other blog posts and Web pages. On the other hand, the readers are provided with the ability to submit their own comments in order to express their agreement or disagreement to the ideas or opinions contained in the blog post. The comments are usually placed below the post, displayed in reverse chronological order. The virtual universe that contains all blogs is known as the *Blogosphere* and accommodates two types of blogs [11, 12]: a) *individual blogs*, maintained and updated by one blogger (the blog owner), and b) *community blogs*, or multi-authored blogs, where several bloggers may start discussions about a product or event.

In a physical community, people use to consult others about a variety of issues such as which restaurant to choose, which medication to buy, which place to visit or which movie to watch. Similarly, the Blogosphere is a virtual world where bloggers buy, travel and make decisions after they listen to the opinions, knowledge, suggestions and experience of other bloggers.

15.2 Social Networks

A social network is a group of individuals sharing common experiences, knowledge and ideas. These individuals are usually grouped within social structures such as communities or neighborhoods. The introduction of Web 2.0 along with the participatory features of the applications it established have led to the creation of numerous social networking Web sites which function as online communities for their members.

Many of these online community members share common interests, beliefs, knowledge, hobbies, religion, or politics. The users who are granted access to a social network are free to construct their profile by filling information regarding their name, email, education and geographic location or describe their habits and personal interests. File uploading is also one of the provided features and these files usually include online games, documents and personal photographes. Furthermore, the members of such a community are free to read the profile pages of other members and contact them.

According to sources published in Wikipedia [7] there are typically several hundreds of such social Web sites. The most popular among them is MySpace[3] and Facebook[4] which accommodate about 471 and 350 millions of registered users respectively. A large fraction of these users connect to their favorite social networking service on a daily basis.

15.3 Office Suites

The vast majority of computer users are somehow familiar to an office suite. Almost everybody have used at least once, a word processor to create a textual

[3] http://www.myspace.com
[4] http://www.facebook.com

document or a spreadsheet software to create documents comprised of enriched and dynamic data.

One vision of the 21st century computing is that a large portion of the applications that now operate offline, will be transferred on the Internet and their users will be able to create, store and distribute information online. By using cloud computing techniques, the dominating Web companies are redesigning their applications in order to provide such functionality. The most ambitious of the existing projects, is the creation of an operating system capable of operating entirely on the Web, within a browsing software.

Another project which has already been released, is the online office suites offered by Web sites in the form of software-as-a-service. Such suites basically include a word processor and a spreadsheet, whereas some of them also offer drawing utilities, graphics editors, presentation applications and even media players.

Nowadays, there are numerous services offering office productivity. The most popular among them is Ajax13[5], Google Docs and Spreadsheets[6], Thinkfree Office Online[7] and Zoho Office Suite[8]. Each of them has its own strong points but generally online office suites offer satisfying capabilities at low (or no) cost, whereas they do not require to download and install any software. Moreover, the users can access their documents from almost any computer with a connection to the Internet, regardless of which operating system they use. Finally, in 2009 Google introduced Google Wave[9], a web-based communication and collaboration tool using richly formatted text, photos, videos, maps, and the like—currently (Jan, 2010), this application is available only after one gets invited.

Nevertheless, there are still some significant disadvantages which indicate that such tools are only at their infant stages. For instance, there are accessibility issues arising from the fact that in case the remote server or network is unavailable, the content will also be unavailable. Moreover, such applications usually require high bandwidth Internet connections, otherwise speed is limited dramatically. Even in that case, an online office application cannot compete an offline opponent in terms of response speed. Finally, although basic functionality is provided, current online office suites do not provide the more advanced features available on their offline counterparts.

15.4 File and Media Sharing Services

The large communities that have been created on the Web have led to the generation of specific services allowing their register users to share files of any type. The most popular forms of file sharing include applications, electronic forms of books, documents, audio files and videos.

[5] http://us.ajax13.com/en/
[6] http://docs.google.com/
[7] http://www.thinkfree.com
[8] http://www.zoho.com
[9] http://wave.google.com

YouTube[10] is currently one of the most popular locations on the Web where users can publish, watch, share and comment videos. The users of the service are divided into two categories: The unregistered users who can just watch videos and the registered ones, who are permitted to upload an unlimited number of files. The latter are also provided with the ability to publish comments about the presented material and judge the quality of the content by voting.

Although each registered user can upload an unlimited number of videos, he/she is not free to publish those which contain defamation, pornography, copyright violations, and material encouraging criminal conduct. These restrictions are all described in the terms of use of the service and videos violating these terms are immediately erased from the database of the site.

15.5 Real-time Web

The advent of Twitter[11] in 2007, introducing the micro-blogging concept, i.e., the posting and delivering of short messages up to 140 characters long to author's "followers", emphasized the need for the *real-time web*. That is, new technologies for rapid dissemination of information as soon as it gets published by its author on the web. Examples of real-time web are frienfeed[12] and notifixious[13], while the *Extensible Messaging and Presence Protocol (XMPP)* and the *Simple Update Protocol (SUP)* are two protocols for developing instant-messaging-like applications [8, 9]. Many believe that this instant-messaging perspective is the next big thing of web.

16 Discussion

In this chapter we have briefly described the core technologies employed by the modern applications of the Web. Knowledge of the modern Web technologies is a key issue for both developers and researchers. The former need to be informed in order to apply the most robust tools when they build their applications, whereas the latter are expected to deeply examine the key issues regarding these technologies in order to provide efficient solutions to newly posed problems.

From one perspective, Web is a society which constantly evolves. At the time these lines are written, novel services are being released and fresh software is composed. Cloud computing services promise to solve current issues and elevate computing. According to the most optimistic judges, the moment at which computers will not require an operating system to work and every transaction will be accomplished through a Web browser, is very close.

[10] http://www.youtube.com
[11] http://twitter.com
[12] friendfeed.com
[13] http://notifixio.us

References

1. http://www.w3.org/TR/html4/
2. http://www.w3.org/TR/REC-xml/
3. http://www.php.net/usage.php
4. http://www.cio.com/article/446829/
5. http://www.abiresearch.com/products/service/
6. http://www.w3.org/TR/mobile-bp/
7. http://en.wikipedia.org/wiki/List_of_social_networking_websites
8. http://xmpp.org/about/
9. http://code.google.com/p/simpleupdateprotocol/
10. Tim O'Reilly (2005-09-30). "What Is Web 2.0". O'Reilly Network. 2006-08-06.
11. N. Agarwal and H. Liu. Blogosphere: Research issues, tools and applications. *ACM SIGKDD Explorations*, 10(1):18–31, 2008.
12. L. Akritidis, D. Katsaros and P. Bozanis. Identifying Influential Bloggers: Time Does Matter Proceedings of the*IEEE / WIC / ACM International Conference on Web Intelligence (WI)*, 76–83, 2009
13. Krishnamurthy, B. and Mogul, J.C. and Kristol, D.M. Key Differences between HTTP/1.0 and HTTP/1.1 Computer Networks-the International Journal of Computer and Telecommunications Networkin, vol. 31, pp 1737, 1999.