ISSN: 2321-7782 (Online)
e-ISJN: A4372-3114
Impact Factor: 6.047

Volume 5, Issue 7, July 2017

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Web OS - 'Moving OS to the Web'

P. Revathi[1]
Research Scholar,
Research Department of Computer Science,
Mazhrul Uloom College, Ambur,
TamilNadu – India

Dr. M. Mohammed Ismail [2]
Head & Associate Professor,
Department Of Computer Science,
Mazharul Uloom College, Ambur, Vellore District,
TamilNadu – India

Abstract: ***This Paper "Moving OS to Web Using Cloud Computing" aims at providing the users with the most easy and the efficient way of using an OS, which is stored in a Server and accessing it from anywhere (at Office or home). The only thing you need to do is that you should have a system with minimum configuration with an internet connection and having a browser for accessing the OS present in the Server. In this Paper we are going to introduce the Web OS – 'Moving OS to the Web' concept using Cloud Computing.***

Using Web OS we will be using the concept of Software as a Service which has a Fabulous advantage, which is, we can install a new software in one server and can access the software from other systems. The Advantages of Web OS are it runs on any Internet enabled computer or device. This is important for mobile workers or people who don't have their own computers and must work out of Internet cafes, libraries, or schools. Also, Web OS users can work, log out, and then log in later from a different computer.

The future enhancement to this Paper is to make the concept of Cloud computing more efficient and reliable; we are going to include Grid computing for providing much more performance for the systems which access the WebOS. This is done since more users login at the same time and the Server might not be able to provide the equal performance to all the systems, so when we add Grid Computing by combining other systems to the server we can achieve the performance by getting the performance from the systems that are connected to the server and providing them to the Systems that are accessing the Web OS.

Keywords: Application Management – Centralized – Concurrent Users – Event Handler – Grid Computing – Internet Technologies – Load Balance – Platform Independent – Scalability — Storage Space – Virtualization – VOS.

I. INTRODUCTION

In this Paper, Moving OS to web using cloud computing implies creating a Virtual OS (VOS). This document is an introduction to the application development for the VOS web Operating System. A little bit PHP and MySQL knowledge are required to read this document, and it is not necessary to have experience programming with CSS, JavaScript or XHTML, although knowing these technologies will help you understand better VOS and its internal functioning

II. WHAT IS VOS

From a technical point of view, VOS is a platform for web applications, created with the idea to make easy the application development. There are currently a lot of web-related technologies, such as PHP, XHTML, CSS and JavaScript, so it is required to master a lot of languages and understanding numerous concepts to be able to create web applications. In addition, every web browser has a different interpretation of the code and every PHP version and configuration works slightly different from the others.

VOS intends to cover those and other problems derived from the web development, offering the programmers a homogeneous platform to develop their web applications, using only PHP code and leaving to the system the resource management, the communication with the browser, the security, etc.

III. VOS Basic Structure

Before studying the VOS components, we must know its basic structure. The platform is created over a client-server architecture, in which VOS is the server, and the client is usually a web browser.

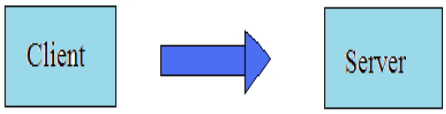

Fig: 3.1 Client-Server Model

Note: Unless specifying the contrary, this manual will assume the client is a web browser.

IV. Cloud Computing

Cloud Computing is a nebulous term covering an array of technologies and services including; Grid Computing, Utility Computing, Software as a Service (SaaS), Storage in the Cloud and Virtualization [10]. There is no shortage of buzzwords and definitions differ depending on who you talk to. The common theme is that computing takes place 'in the cloud' - outside of your organizations network [2].

V. Purpose of Moving OS to Web

Due to the increase in the technological advancements, Cloud Computing is becoming the hottest topics. In Cloud Computing, users work with Web-based, rather than local storage and software. How about having our own PC with such a feature? , So that you can access your own PC where ever you want, i.e., from your home or office with the same OS, Hard disc and Software's by making the PC available on the Internet. Doesn't that sound good? Yes, it's possible with Cloud Computing [4]. You can shift an entire OS to the Web using the Concept of Cloud Computing [3].

It functions much like a traditional operating system, although it doesn't include drivers for computer hardware is becoming a subject of increasing interest. One contributing factor is Internet technologies' increasing bandwidth, which enables the efficient movement of applications and data via the Internet to and from Web Oss. We can Design this Web OS using Cloud Computing Architecture, which makes computing easy, fast and efficient. With this approach, users can work with their applications from multiple computers. In addition, organizations can more easily control corporate data and reduce malware infections. Also, cloud computing makes collaboration easier and can reduce platform-incompatibility problems [5]. Cloud Architectures address key difficulties surrounding large-scale data processing. In traditional data processing it is difficult to get as many machines as an application needs. Second, it is difficult to get the machines when one needs them. Third, it is difficult to distribute and coordinate a large-scale job on different machines, run processes on them, and provision another machine to recover if one machine fails . Fourth, it is difficult to auto scale up and down based on dynamic workloads. Fifth, it is difficult to do away with of all those machines when the job is done. Cloud Architectures solve such difficulties.

VI. User Interface

Since we moved the OS to the internet, the browser of the each client act as the user interface for the particular user. User generates the request to server via browser and the program gets executed over the server side and response is sent back to the user via browser. Only minimum hardware and software resources is enough for the client to access the VOS through browser, since all the required amount of hardware and software are installed in server side [6].

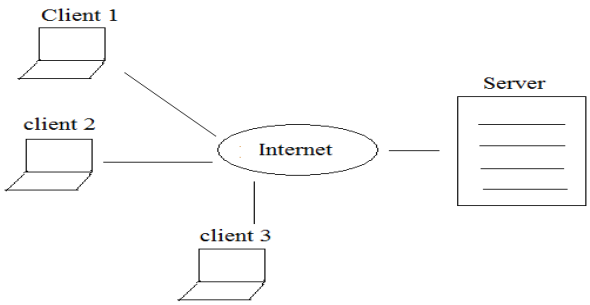

Fig: 6.1 Multiple User Interaction

The above Figure: [6.1] shows the client server interaction through the web browser installed in client side. Since web browser act as user interface, the VOS is platform.

VII. System Architecture

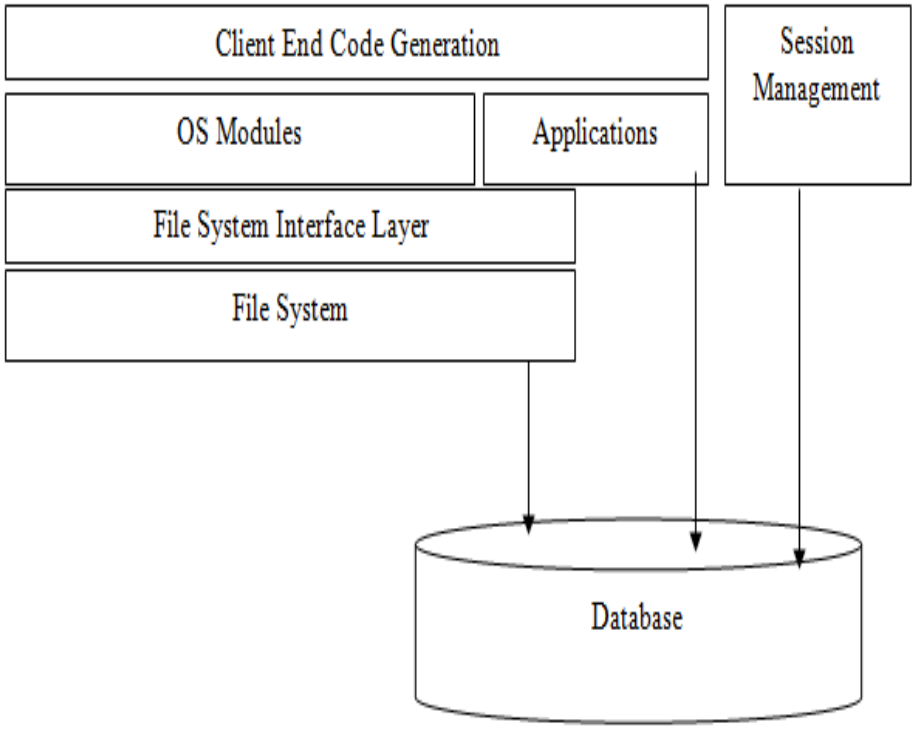

Fig: 7.1 Virtual Operating System Design Diagram

The File System Interface Layer forms the core functionality of the Paper. With every aspect depending on files for their working, it is essential to implement this module flexible and powerful way. The main part of File System Interface Layer is managing the fs table.

The User Interface Management manages the look and feel of the system. This is completely implemented using JavaScript and cascaded style sheet. This also includes the generation of response for each request by client. This forms the largest portion of the Paper.

Application Management manages the applications associated with the user. Also it provides options for users to add/remove applications from their environment. This module is designed in such a way that it provides future addition or modification of applications.

Recycle Bin Management forms as a sub part of the file system. This manages a special kind of folder which marks a virtual link to the deleted file and also providing facilities for files to be copied back to original location from where they are deleted.

Session Management application is developed with having large scale use in mind and hence we use sessions for handling concurrent users. The session also is designed well to take care of the security issues.

Event Handler is the client end part of code and forms sub module of the user interface. This module is responsible for receiving the events from various objects and pack them is xml requests and port it to server or handled locally(usually within applications) and simulating the actual desktop in a virtual way.

VIII. PROGRAMME TESTING

The logical and syntax errors have been pointed out by performing the program testing on WEB OS Application. An improperly defined field dimension or omitted keywords are common syntax errors found during this testing phase. These errors are shown through error messages generated by the computer. A logic error on the other hand deals with the incorrect data fields, out-off-range items and invalid combinations. Condition testing exercises the logical conditions contained in a module. The possible types of elements in a condition include a Boolean operator, Boolean variable, a pair of Boolean parentheses, a relational operator or on arithmetic expression. Condition testing were performed to focus the testing on each condition in the program the purpose of condition test is to deduct not only errors in the condition of a program but also other errors in the program.

IX. FILE SYSTEM MANAGEMENT

This module forms the core functionality of the Paper. With every aspect depending on files for their working, it is essential to implement this module flexible and powerful way. This module is used to maintain the each files and directories status of all the clients. To maintain the status of all the client's files and directories we use *file system (fs)* table, the table has the following fields:

- ✓ Fileid - This is the actual name with which file is stored. Because multiple users may want to have same file name. So this field is maintained unique as primary key. This field doesn't have information of type of file.
- ✓ Filename - the filename visible to the external world.
- ✓ Dirfile - This field specifies the file type of the file.
- ✓ Deleted - This flag determined if file is deleted (present in trash) or not.
- ✓ Loc - This gives the virtual location of the file
- ✓ Actualloc - This gives the actual location of the file in server, which is never exposed to the client end, thus enhancing the security.
- ✓ Username - This determines the user with whom the file is associated

X. RECYCLE BIN MANAGEMENT

This forms as a sub part of the file system. This manages a special kind of folder which marks a virtual link to the deleted file and also providing facilities for files to be copied back to original location from where they are deleted.

Session Management:

This application is developed with having large scale use in mind. So there will be more number of clients accessing the server OS or VOS and their own data at the same time, in order to separate the activity of one user from other users we are using the session concept or session management, that is to maintain the concurrent users. The session also is designed well to take care of the security issues.

XI. Event Handler and Application Management

Event Handler

This module is the client end part of code. This module is responsible for receiving the events from various objects and pack them is xml requests and port it to server or handled locally(usually within applications) and simulating the actual desktop in a virtual way.

Application Management

This module manages the applications associated with the user. Also it provides options for users to add/remove applications from their environment. Client can add the applications which are needed by the client from the list of applications. It is not necessary to have all the applications. This module is designed in such a way that it provides future addition or modification of applications. This module also helps in by reducing the usage of time and resources required by the particular client.

XII. Web Technologies

PHP:

PHP is a general-purpose scripting language that is especially suited for web development. PHP generally runs on a web server, taking PHP code as its input and creating web pages as output. It can also be used for command line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platform, and can be used with many relational database management systems. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. It can automatically detect the language of the user From PHP 4, the PHP parser compiles input to produce byte code for processing by the Zend engine, giving improved performance over its interpreter predecessor.

Originally designed to create dynamic web pages, PHP's principal focus is server-side scripting, and it is similar to other server-side scripting languages that provide dynamic content from a web server to a client, such as Microsoft's Active Server Pages, Sun Micro system's Java Server Pages, and mod_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promote Rapid Application Development (RAD). Some of these include CakePHP, Symfony , Code Igniter, and Zend Framework, offering features similar to other web application frameworks [9].

XHTML:

The Extensible Hypertext Markup Language, or XHTML, is a markup language that has the same depth of expression as HTML, but also conforms to XML syntax.

While HTML prior to HTML 5 was defined as an application of Standard generalized Markup Language (SGML), a very flexible markup language, XHTML is an application of XML, a more restrictive subset of SGML. Because they need to be well-formed, true XHTML documents allow for automated processing to be performed using standard XML tools—unlike HTML, which requires a relatively complex, lenient, and generally custom parser. XHTML can be thought of as the intersection of HTML and XML in many respects.

XHTML is "a reformulation of the three HTML 4 document types as applications of XML 1.0". The W3C also continues to maintain the HTML 4.01 Recommendation and the specifications for HTML 5 and XHTML 5 are being actively developed. In the current XHTML 1.0 Recommendation document, as published and revised to August 2002, the W3C comments that, "The

XHTML family is the next step in the evolution of the Internet. By migrating to XHTML today, content developers can enter the XML world with all of its attendant benefits, while still remaining confident in their content's backward and future compatibility.

JAVASCRIPT:

JavaScript is a Scripting Language used to enable programmatic access to objects within other applications. It is primarily used in the form of client-side Java script for the development of dynamic websites. JavaScript is a dialect of the ECMA Script standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but be easier for non-programmers to work with.

JavaScript, despite the name, is essentially unrelated to the Java Programming Language, even though the two do have superficial similarities. Both languages use syntaxes influenced by that of C syntax, and JavaScript copies many Java names and naming conventions. The language's name is the result of a co-marketing deal between Netscape and Sun, in exchange for Netscape bundling Sun's Java runtime with their then-dominant browser. The key design principles within JavaScript are inherited from the self and Scheme programming languages. "JavaScript" is a trademark of Sun Microsystems. It was used under license for technology invented and implemented by Netscape Communications and current entities such as the Mozilla Foundation.

AJAX:

AJAX (Asynchronous JavaScript and XML), is a group of interrelated web development techniques used to create interactive web applications or rich internet applications. With Ajax, web applications can retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page. The use of Ajax has led to an increase in interactive animation on web pages. Data is retrieved using the XMLHttpRequest Object or through the use of Remote scripting in browsers that do not support it. Despite the name, the use of JavaScript and XML is not actually required, nor do the requests need to be asynchronous. The acronym AJAX has thus changed to the term Ajax, which does not represent these specific technologies [8].

XIII. WEB OS

The proposed system is client-server architecture. In which server plays the important role, it will contain the OS and all other software's and hardware's. The only interface for the client is web browser; the client needs to have minimum hardware resources in order to access the web browser. Client can access his/her own data through web browser from the server which is connected through internet. The OS is provided to client as Software as a Service (SaaS) and the storage space is also provided as the Hardware as a Service (HaaS). The storage space for the OS and clients data are in server resources. Client uses the virtual file system to access the files and folders of the users which are stored in server space [1].

To accommodate more number of users the VOS uses the session management to separate from each user. Also VOS uses file system tables which will details about the files of the particular users [5].

Below Figure [13.1] shows the proposed system diagram. Here all the clients access the OS and other software from the server connected via internet.

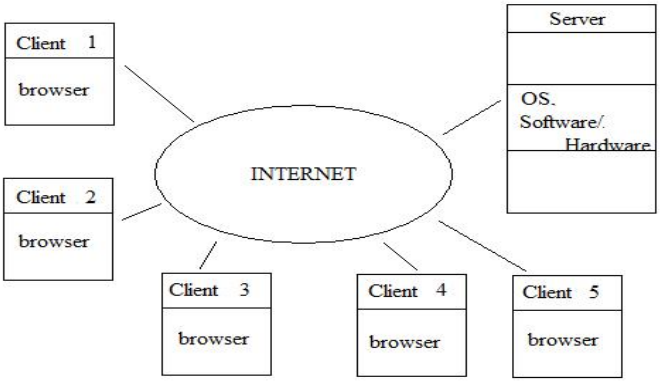

Fig: 13.1 Virtual Operating System client-server architecture

XIV. CONCLUSION

Instead of building your applications on fixed and rigid infrastructures, Cloud Architectures provide a new way to build applications on on-demand infrastructures. Without having any upfront investment, we were able to run a job massively distributed on multiple nodes in parallel and scale incrementally based on the demand (users, size of the input dataset). With no idle time, the application infrastructure was never underutilized. Whether you call it Cloud Computing or utility computing, the omnipresent power of high-speed internet connections and linkages to databases, applications and processing power will change the way we work in communications. We may use Cloud Computing tools without being aware of what they actually are. We may learn our way into Cloud Computing applications that seem strange today. We will find entirely new applications in the Cloud to help send messages more effectively to target audiences.

Directions for future work:

a) **Grid computing** concept can be added with the **Cloud computing** to obtain efficient load balancing between large numbers of client using at the same time. This is accomplished by sharing the resources of idle system connected in the network.

b) Adding more number of applications so that the user updates with the new and latest software accessed as a service.

c) Adding Crypto graphical Things like Encryption, Decryption of File Storage system. Advanced Security Systems like OTP etc…

References

1. Neal Leavitt, *'Is Cloud Computing Really Ready for Prime Time?'* Pg. 15-20, January, 2009.
2. Carl Hewitt, '*ORGs for Scalable, Robust, Privacy-Friendly Client Cloud Computing'*, Pg. 96-99, September, 2008.
3. Lizhe Wang , Jie Tao , Marcel Kunze , Alvaro Canales Castellanos , David Kramer , Wolfgang Karl , '*Scientific Cloud Computing: Early Definition and Experience*', Pg. 825-830, September, 2008.
4. George Lawton, '*Moving OS to the Web*' – Journal of Computer science Volume 41, Issue 3, Pg. 16-19, May, 2008.
5. Huan Liu , Dan Orban , '*GridBatch: Cloud Computing for Large-Scale Data-Intensive Batch Applications*', Pg. 295-305, May , 2008.
6. Michael Miller, 'Cloud Computing: Web-Based Applications that Change the Way You Work and Collaborate Online', 2008.
7. Philip K. Mckinley , Farshad A. Samimi , Jonathan K. Shapiro, Chiping Tang , 'Service Clouds: A Distributed Infrastructure for Constructing
8. Nicholas C. Zakas, Jeremy McPeak, and Joe Fawcett, 'Professional Ajax', 2007.
9. Luke Welling, Laura Thomson, 'PHP and MySQL Web Development', 2007.

AUTHOR(S) PROFILE

P.Revathi, She is a Research Scholar, Has received M.Sc Degree in 2016. Her research interest is in the area of Data Mining that maximize innovative patents and Network Security and Cryptography and Cloud Computing.

