International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-2, Issue-9, September 2014

Web Operating System- An Overview

Shakshi Maheshwari

***Abstract*— With the increasing use of high-speed Internet technologies during the past few years, the concept of cloud computing has become more popular. In cloud computing, users work with Web-based, rather than local, storage and software. These applications are accessible via a browser and look and act like desktop programs. 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.**

Now a growing number of organizations are adding to the cloud concept by releasing commercial and open source Web-based operating systems. While the idea isn't new, the proliferation of users and applications distributed over the Web, including those at scattered corporate sites, has made it more interesting, relevant, and, vendors hope, commercially viable. It also includes many of a traditional OS's capabilities, including a file system, file management, and productivity and communications applications. As is the case with Web-based applications, the Web OS functions across platforms from any device with Internet access.

***Index Terms*— iCloud, EyeOS**

I. INTRODUCTION

With traditional computer operating systems, you'd have to install applications to your own computer. The applications would exist on your computer's hard disk drive. They would run by accessing the processing power of your computer's central processing unit (CPU) by sending electronic requests to your computer's OS.

A Web OS[5][6][7] allows you to access applications stored not on your computer, but on the Web. The applications exist wholly or in part on Web servers within a particular provider network. When you save information in an application, you might not store it on your computer. Instead, you save the information to databases connected to the Internet. Some Web operating systems also give you the option to save information to your local hard disk drive.

Because Web operating systems aren't tied to a specific computer or device, you can access Web applications and data from any device connected to the Internet. That is, you can do it as long as the device can run the Web operating software (whether that's a particular Web browser or client).

Fig 1: The AstraNOS operating system login screen

Why need WebOS?

Web operating systems simplify a user's experience when accessing applications hosted on remote servers. Ideally, a Web OS behaves like a desktop OS. The more familiar and intuitive the system, the faster people will learn how to use it. When a person chooses to run a certain application, his or her computer sends a request to the system's control node -- a special server that acts as a system administrator. The control node interprets the request and connects the user's client to the appropriate application server or database. By offloading applications, storage and processing power to a remote network, users don't have to worry about upgrading computer systems every few years.

Web operating systems can also make it easier to share data between computers. Perhaps you own both a Mac computer and a PC. It can be challenging to share data between the two different computers. Even if you use file formats that are compatible with both Mac computers and PCs, you could end up with a copy of the same file on each machine. Changing one copy isn't reflected on the other computer's copy. Web operating systems provide an interface where you can use any computer to create, modify and access a single copy of a file saved on a remote database. As long as the Web OS you're using can cross platforms, meaning it works on both Macs and PCs, you'll be able to work on the file at any time using either of your computers.

Likewise, Web operating systems can simplify collaborative projects. Many Web operating systems allow users to share files. Each user can work from the file saved to the system's native network. For many users, this is an attractive alternative to organizing multiple copies of the same file and then incorporating everyone's changes into a new version.

Manuscript received September 21, 2014.
Shakshi Maheshwari, University of Mumbai, India

Fig 2: YouOS is one of the most popular Web OS on the Internet

Which technologies do they work on?

With so many different Web operating systems either currently available or in development, it should come as no surprise that programmers use different approaches to achieve the same effect. While the goal of a Web OS is to provide an experience similar to using a desktop OS, there are no hard and fast rules for how to make that happen. The two most popular approaches rely on Flash technologies or Asynchronous JavaScript and XML (AJAX) technologies.

Flash is a set of technologies that enable programmers to create interactive Web pages. It's a technology that uses vector graphics. Vector graphics record image data as a collection of shapes and lines rather than individual pixels, which allows computers to load Flash images and animation faster than pixel-based graphics.

Flash files stream over the Internet, which means the end user accessing the file doesn't have to wait for the entire file to download to his or her computer before accessing parts of it. With Flash-based programs like YouTube's video player, this means you can start watching a film clip without having to download it first.

AJAX technologies rely on hypertext markup language (HTML), the JavaScript programming language, Cascading Style Sheets (CSS) and eXtensible Markup Language (XML). It's a browser technology. The HTML language is a collection of markup tags programmers use on text files that tell Web browsers how to display the text file as a Web page. CSS is a tool that gives programmers more options when tweaking a Web site's appearance. Programmers can create a style sheet with certain attributes such as font style and color, and then apply those styles across several Web pages at once. JavaScript is a programming language that allows applications to send information back and forth between servers and browsers. XML is a markup language, which means programmers use it to describe the structure of information within a file and how it relates to other information.

The "asynchronous" aspect of AJAX means that AJAX applications transfer data between servers and browsers in small bits of information as needed. The alternative is to send an entire Web page to the browser every time something changes, which would significantly slow down the user's experience. With sufficient skill and knowledge, a programmer can create an AJAX application with the same functions as a desktop application.

How do WebOS operate?

Consider the case of computers we have. The computer consists of many applications to work with such as one can use calculator to calculate, calendar to be scheduled, clock, games, and many other applications. Apart from these apps we also have diverse data like movies, memories, music, and files etc which we store in computer hard disk. We communicate with the computer through the user interface which is right now before your eyes (if you are viewing this page on a computer). If we want to share any data, internet is engaged. Sharing can be done through many websites available on the web. So this is how a local computer with a normal user works. Now consider yourself as a computer user right now working at office. Suddenly your boss calls you to show some random file to him (sometimes they like people feeling inconvenient). But you forgot to bring it. The file is in the hard disk of your home PC and it could not be teleport from there to your office computer in air. This is the situation when most brain thinks if there could be any method by which they could access their local content anywhere. This problem was resolved by the programmers by introducing the concept of "Web OS".

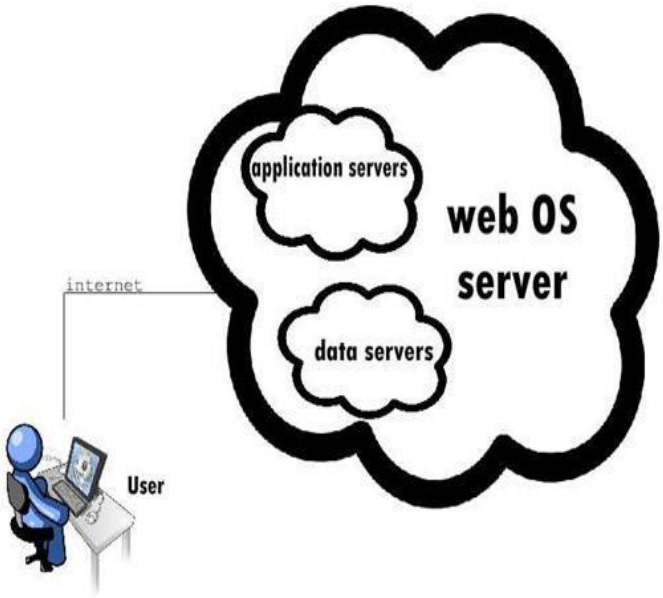


Fig 3: Web OS structure

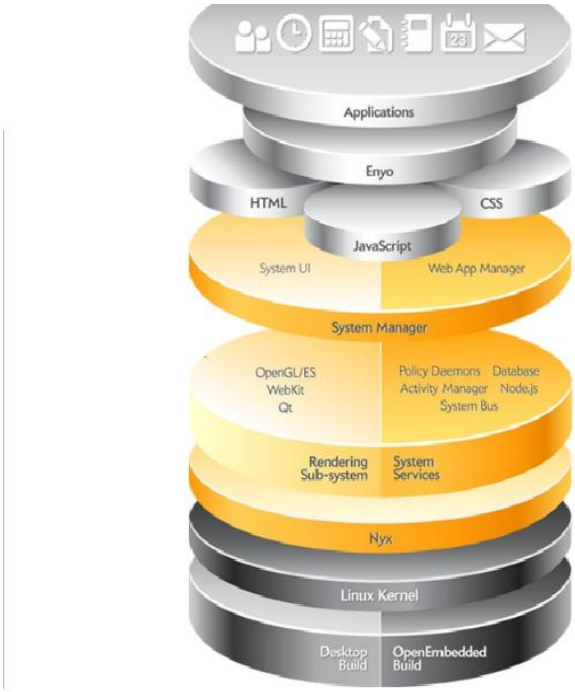
WebOS are the dynamic computers. The applications, hard disk, operating systems are all present at the servers from where they are operated. The web OS service provider has different spaces for application access and database. The user is provided with a graphical user interface which feels like the one at your PC. This operating system consists of application section like calendar, clock, calculator, document editors etc then there is a section for data storage where user can store data, and there are many other sections depending upon the web OS. Whatever content user wants to store is stored at the hard disk at servers. As the terminology itself says, the web OS make use of the web to connect and upload files to the client server.

As the files are now been stored at the server, user can now utilize them remotely anywhere on the earth. It just needs internet and a computer and everything is done. User has to switch on computer, start the internet and go to respective web

OS website or download the web OS application (in case it is not on the remote computer), and browse the data.

II. ARCHITECTURE

The architecture of Web OB is as described in Figure 4.
Core Apps include the following Enyo1.0 applications: accounts, calculator, calendar, clock, contacts, email, and memos, and the Node.js-based services required to run them. Enyo is an object-oriented JavaScript application framework emphasizing modularity and encapsulation.
Web OS System Manager is a key webOS component responsible for:
Managing the application and and service interface for physical devices, such as keys, accelerometer and vibrator.
Managing the running of applications and passing of messages between applications, Installation and removal of applications, Managing display and notify applications and services, Managing the sharing of system resources between different applications and services, the dock mode status, the security policy and access to a locked device. It also provides for display of notifications and system menus. It also renders webOS card view, lock screen, status bar, system menus, virtual keyboard, notifications, and launcher, in addition to other system management features that are viewable in the System Manager User Interface. WebAppMgr is provided by System Manager and is responsible for running Enyo applications.



Open webOS uses a version of QtWebKit to render web pages and apps. Code-named "Isis Project", it uses a client-server model that separates the rendering process from the user interface. This architectural approach delivers smooth scrolling and a responsive user experience. The Isis Browser uses the highly portable Qt framework with the goal of delivering a cross-platform browser in the future. The code may be found in the Isis Project repository on GitHub.
Rendering Subsystem: Rendering is achieved through a set of policies orchestrated by luna-sysmgr, relying on Qt, qtWebKit and OpenGL/ES and supporting hardware rendering.
Novacom: Novacom and Novacomd provide a generic communication toolset to allow communication between a host and an embedded device using sockets over USB. New devices can be supported by adding a new vendor ID in the USB device stack.
System Services policy features are implemented on top of the Platform Portability Layer. They include:
Powerd
Sleepd
Storaged
DB8: DB8 is the database service provider for webOS components. The initial release provides a partial implementation on top of the LevelDB database engine. Future releases will provide an implementation based on LevelDB.
Luna-service2: Luna-service2 provides a bus-based IPC mechanism used between components in Open webOS. Luna-service2 is composed of a client library and a central hub daemon. The client library provides API support to register on the bus and communicate with other components. The hub provides a central clearinghouse for communication. Utilities for monitoring and debugging the bus are included.
Node.js: Node.js is an open source project which can be found at http://nodejs.org. The Node.js release in Open webOS provides minimal extensions to enable access to the webOS system bus and extended system access for Node.js services. This includes the node_spawner tool used to launch Node.js services within webOS.
The Open webOS platform portability layer (PPL), code-named "Nyx Project", is used to isolate the upper layers of webOS from dependencies on the hardware and the core OS upon which it is running. It is implemented as a shared library that exposes a uniform client API and expects to call into a series of platform-dependent modules that implement the API for a particular device. Open webOS is intended to be built on any standard Linux kernel, with the Platform Portability Layer providing platform abstraction.

III. ICLOUD

Fig 5: icloud Operating System

iCloud is a cloud storage and cloud computing service[1][2][3] from Apple Inc. launched on October 12, 2011. As of July 2013, the service has 320 million users.[4]
The service allows users to store data such as music and iOS applications on remote computer servers for download to

multiple devices such as iOS-based devices running iOS 5 or later,[4] and personal computers running OS X 10.7.2 "Lion" or later, or Microsoft Windows (Windows Vista service pack 2 or later). It also replaces Apple's MobileMe service, acting as a data syncing center for email, contacts, calendars, bookmarks, notes, reminders (to-do lists), iWork documents, photos and other data. The service also allows users to wirelessly back up their iOS devices to iCloud instead of manually doing so using iTunes. The users are also able to share photos, music, and games instantly by linking one anothers account via airdrop wireless.

A. *ARCHITECTURE OF iCloud*

From an application's perspective, iCloud consists of one or more "special folders" whose contents iCloud synchronizes with files stored at a central location. This special folder is called a ubiquity container. An application can have one or more ubiquity containers, each of which is assigned its own unique container ID when you enable an application to use the service. As a user adds or modifies application data, iCloud pushes the changes to a central server, which in turn pushes them to other devices that have signed up to share it. An application doesn't need to query iCloud for updates to its ubiquity containers but instead simply queues itself as an observer. When notified of new content, the application takes steps to integrate it into its local data stores.

To make this mechanism perform efficiently, the contents of files in a container are broken into *chunks*. Whenever you change a file in a ubiquity container, the synchronization mechanism pushes the bits that have changed, not the entire file. The same thing happens when an application is notified of changes made on other devices: the application running on your device receives only the bits that have changed and integrates them into the files in its ubiquity container.

The synchronization of data across devices is managed by a background process on each device known as the *daemon*. The daemon is not under the control of the developer, who is responsible for managing the main thread of a program. The daemon is an independent process, whose job is to detect changes to a resource (for example, a document or database) and send these changes to a central iCloud server. The daemon acts as a sort of middle man to the file system on a device. This is summarized in Figure 6, *Architecture of iCloud*, which diagrams the flow of data between an application, its containers, and iCloud.

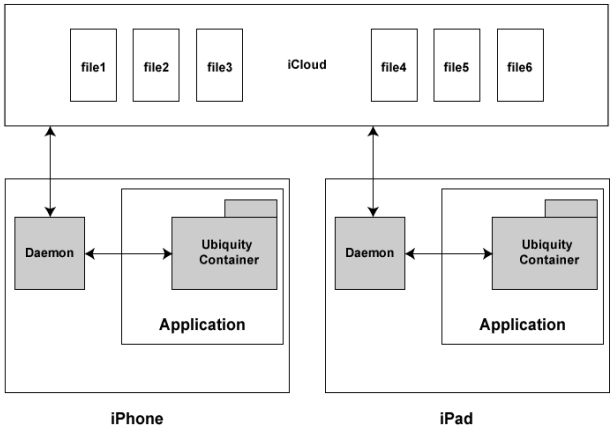


Fig 6: Architecture of iCloud

B. *Real Applications of iCloud*

The cloud-based system allows users to store music, photos, applications, documents, bookmarks, reminders, backups, notes, iBooks, and contacts, and provides a platform for Apple's email servers and calendars. Third-party iOS and OS X app developers are able to implement iCloud functionality in their apps through the iCloud API.[5]

Backup and restore

iCloud allows users to back up the settings and data on iOS devices running iOS 5 or later.[5] Data backed up includes photos and videos in the Camera Roll, device settings, app data, messages (iMessage, SMS, and MMS), ringtones, and Visual Voicemails. Backups occur daily when the device is locked and connected to Wi-Fi and a power source. In case of a malfunction of any Apple device, during the restoration process, iCloud offers to restore all data along with App data only if the device was synced to iCloud and backed up.

Find My iPhone

Find My iPhone, formerly part of MobileMe, allows users to track the location of their iOS device or Mac. A user can see the device's approximate location on a map (along with a circle showing the radius depicting the margin of error), display a message and/or play a sound on the device (even if it is set to silent), change the password on the device, and remotely erase its contents.[6] The feature was first announced on June 10, 2009 and was included in iOS 3.0 software update as a feature for paying MobileMe users.[7] Find My iPhone was made free of charge with the iOS 4.2.1 software update on November 22, 2010, but only for devices introduced in 2010.[8] An iOS app was also released by Apple on June 18, 2010, which allows users to locate their device from other iOS devices running iOS 4 or later software.[9][10] In iOS 5, Find My iPhone was continued as a feature for iCloud. iOS 6 introduced Lost Mode, a new feature that allows the user to mark a device as "lost", making it easier to protect and find. The feature also allows someone that finds the user's lost iPhone to call the user directly without unlocking it. Similar phone finder services under various names are available for other families of smartphones.

iCloud Drive

iCloud Drive is iCloud's file hosting service for devices running iOS 8, OS X Yosemite, or Windows 7 or later. This feature allows users to save photos, videos, documents (Keynote, Pages, and Numbers), and music, and other apps data on iCloud... Users can start their work on one device and continue on another device. By default, users will get 5 GB of storage for free, but this space will be expandable via subscription.

Photo Stream

Photo Stream is a service supplied with the basic iCloud service which allows users to store the most recent 1,000 photos on the iCloud servers up to 30 days free of charge. When a photo is taken on a device with Photo Stream enabled, it is automatically uploaded to the iCloud servers; from there, it is automatically pushed to the rest of the user's registered devices. Photos in Photo Stream will automatically be

removed from other devices after the user reaches the 1,000 photo or 30 day limit. Users who utilize Photo Stream on their Mac or PC can choose to have all photos permanently saved on their computer - their photos in Photo Stream will not be removed from the computer when they are dropped out of Photo Stream after the user reaches the 1,000 photo or 30 day limit. The service is also integrated with Apple TV, allowing users to view their recent photos wirelessly on their HDTV.[11]

IV. EyeOS

EyeOS[4] is the Open Source Cloud Computing's Web Desktop. This operating system can be used by individuals as well as organizations. You can either create an account on eyeOS server or download it and run it from your own server. eyeOS acts as a platform for web applications written using the eyeOS Toolkit. It includes a Desktop environment with 67 applications and system utilities, including Word Processor, Address Book, PDF reader, and tons of applications developed by the community.

Fig 7: EyeOS

A. *ARCHITECTURE- eyeOS*

eyeOS architecture can be fully clusterized. This enables us to balance the eyeOS load as well as the office conversion to gain in horizontal scalability and availability.

Kernel
Responsible for managing resources via call services to the system

Connectivity
- Web API Rest Server – Enables communication between applications and eyeOS via APIs

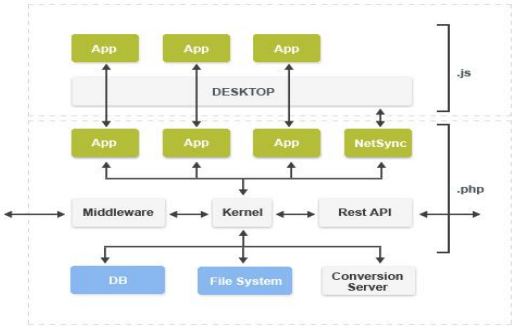


Fig 8: Architecture of EyeOS

- Web Netsync Server – Enables real time communication. It is isolated from the Web API Rest Server since it consumes the server's resources in a different way than the web server

Data Storage
- File System – Enables data storage. NAS: Network Access Storage
- DB – Metadata storage: groups, user profiles, preferences, privileges, etc.

File formatting
- Office conversion – Service that converts documents and files into browser friendly formats. Benefit: eyeOS always handles files in the best format for its environment
- Multimedia conversion – service that coneverts multimedia files into browser friendly formats.

Administration
- Middleware – Manages the eyeOS server via a web front end without having to access the Linux layer below and eliminates the need to run advanced commands.

B. *REAL APPLICATIONS OF EyeOS*

I asked Eduardo Pérez of EyeOS how people are currently using EyeOS - what are some of the common uses? Eduardo told me that most EyeOS users see it as "a virtual web desktop", but he also said there are three kind of users:

1) People who **download the code** and use it in their own server.
This person commonly uses EyeOS to share files and as a communication tool for families and small work groups.

2) People who use the **miniserver** (a small windows-only program that installs apache, php and eyeOS in your PC, so you can run it locally).

This type of user does the following things with EyeOS:

- Play games
- Test out the system
- Develop / translate new applications
- Word processing

3) The users at the **Public Server** at http://eyeOS.info:

These users, over 50,000 of them, commonly use EyeOS for these tasks:

- Share files
- Office tasks - e.g. creating private word documents, using the calendar and agenda. They do this so they can access the files from different computers. One user mentioned that he creates docs at home and then prints them out in the office, using eyeEdit.
- Upload and download files in places were FTP is not allowed (e.g. the office) and use eyeOS to move files.
- Chat / IM - eyeMessages (their IM app) has a lot of traffic right now, according to Eduardo.
- Play games

V. CONCLUSION AND FUTURE WORK

Since the inception Operating systems have been in use but with the introduction of Web Operating System the dependency on the traditional desktops have been eradicated. Though the Web OS have been serving as a boon but their market value is deprecating. Right now, Web operating systems aren't as robust as their desktop counterparts. But some people believe that Web operating systems provide just enough functionality to compete with more traditional desktop software suites. If Web OS providers can address issues like the functionality gap and data security concerns, we might see a dramatic shift in computer network systems.
A common concern about Web operating systems is that they require users to trust a third party to keep potentially sensitive data secure. For many users, this is a leap of faith. Will the provider be able to fend off hackers? It's in the provider's best interests to employ advanced security measures to keep client data safe. As distributed computing systems become more popular, we'll likely see a battle between hackers and security specialists.

REFERENCES

[1]Switched On: Apple's cloud conundrum". Engadget. June 13, 2011. Retrieved June 13, 2011.
[2] "Fourth time's a charm? Why Apple has trouble with cloud computing". ArsTechnica. June 8, 2011. Retrieved June 13, 2011.
[3]"4th Time a Charm for Apple? From iDisk to .Mac to MobileMe to iCloud". Wired. May 31, 2011. Retrieved June 13, 2011.
[4] eyeOS. "Open Source Web Desktop Environment, commonly known as Web Operating System (Web OS) or Web Office".
[5] "WebOS: Operating System Services for Wide Area Applications".
[6]Peter Kropf and John Plaice and Herwig Unger. "Towards a Web Operating System".
[7] Richard MacManus. "What is a WebOS?".
[8]iCloud: Backup and Restore Overview". Apple. February 17, 2014.
[9]Apple - iCloud - Find your missing Apple device.". Apple.
[10]Snell, Jason (Nov 22, 2010). "Apple makes Find My iPhone free for some iOS 4.2 users". Macworld. IDG. Retrieved July 2, 2012.
[11] Apple's iCloud Connects People and their Photos". InfoTrends InfoBlog. June 8, 2011. Retrieved August 30, 2011.

Shakshi Maheshwari, Bachelor of Engineering in Computer Science from Mumbai University Currently working as the Systems Engineer for NSE on behalf of TCS(Tata Consultancy Services).