Int. J. Advanced Networking and Applications
Volume: 05 Issue: 01 Pages:1853-1857 (2013) ISSN : 0975-0290

Remote Access using Remote Frame Buffer Protocol based on Virtual Network Computing Architecture in Mobile Cloud

Maddali M. V. M. Kumar
Assistant Professor, Department of MCA, St. Ann's College of Engineering & Technology, Chirala, India.
Email ID: mvmkumar.mtech@gmail.com

----------**ABSTRACT**----------
Mobile Cloud Computing is the emerging area with the usage of Cloud Computing in mixture with mobile devices. The principle of mobile cloud computing physically splits the user interface from the application logic. Although they suffer from intrinsic resource limitations, mobile devices have become very prevalent. Mobile cloud computing provides a elucidation to come across the intensifying functionality demands of end-users, as all application logic is proficient on reserved servers and only user interface functionalities reside on the mobile device. The mobile device turns as a remote display, taking user input and translation the display updates acknowledged from the reserved server. The key challenge for mobile cloud computing is the boundaries inherent to the mobile devices like processing power, memory, network bandwidth and storage capacity as compared to a fixed device like PC, and short battery lifetime and interaction latency present additional major challenges for the remote display on mobile devices of cloud applications. In this paper a number of adequate solutions that have been suggested to tackle the main issues accompanying with the remote display on mobile devices of cloud facilities.

Keywords - **Mobile Cloud, Proxy, RFB Protocol, Smart VNC, WNIC**

Date of Submission: June 08, 2013 Date of Acceptance: August 13, 2013

I. INTRODUCTION

Mobile Cloud Computing (MCC) is a combination between mobile network and cloud computing, thereby providing optimal services for mobile users. In mobile cloud computing, powerful configuration is not needed in mobile devices since all the data and difficult computing modules can be dealt with in the clouds. Mobile Devices like Smartphone, Tablet PCs are progressively increasing a vital part of human life as the most effective and proper communication tools which do not have constrain of time and place. Mobile users accumulate rich knowledge of various services from mobile applications such as iPhone Apps, Google Apps, etc., which run on the devices and/or on remote servers via wireless networks. The rapid improvement of Mobile Cloud Computing grow into a dominant development in the IT technology as well as commerce and industry fields.

However, the mobile devices are facing many challenges in their resources (e.g., battery life, storage, and bandwidth) and communications (e.g., mobility and security) [2]. The limited resources ominously impede the heightening of service qualities. Mobile Cloud at its humblest states to an infrastructure where both the data storage and the data processing occur outside of the mobile device. Mobile cloud applications transmit the computing power and data storage left from mobile phones and into the cloud, receiving applications and mobile computing to not just smartphone users but a much broader range of mobile subscribers. Mobile Cloud as a new paradigm for mobile applications whereby the data processing and storage are relocated from the mobile device to strong and federal computing platforms located in clouds. These integrated applications are then retrieved over the wireless connection based on web browser or a thin native client on the mobile devices.

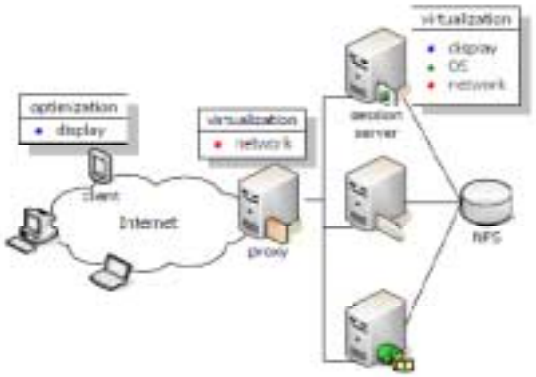

Figure 1: Mobile Desktop Architecture

II. STRUCTURE OF VNC

Time considering portability and generality propose a VNC based architecture. It involves of VNC servers

Volume: 05 Issue: 01 Pages:1853-1857 (2013) ISSN : 0975-0290

consecutively on one or more remote computers, a Smart VNC (SVNC) proxy, and a Smart VNC onlooker on a cellular phone. A VNC server directs remote desktop shows as bitmap images in RFB protocol. A SVNC proxy converts (crops, shrinks and resample) the display image and then transmits the renewed image to a SVNC viewer in response to a user request that was received from that SVNC viewer. The transfer is performed in this Compact RFB (CRFB), streamlined RFB protocol. Then, the SVNC viewer displays the transferred images. Key events received by the SVNC viewer are transmitted to a SVNC proxy that coverts them and sends them to the server. When the user first tries to connect to a remote computer, he must specify his user name and password for confirmation as well as the host name of the computer that is running a VNC server. If confirmation succeeds, the SVNC proxy creates a session with the VNC server and the SVNC viewer starts user services. To suppress network traffic, encoding is amended depending on contexts.

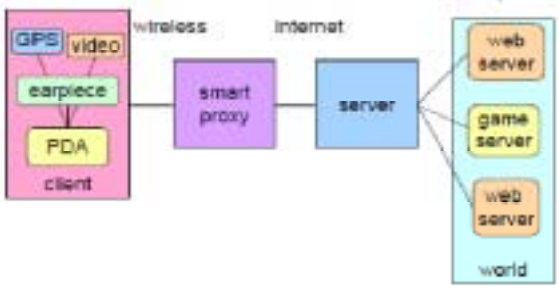

Figure 2: Smart VNC Proxy

However, while the user is handling the remote desktop access such as scrolling and touching the showing device, the display images are grey-scaled to decrease the number of bytes essential to encode the image. Only an onlooker component is implemented on the mobile device, operating as a remote display for the applications running on reserved servers in the cloud. Any remote display framework is composed of three components: a server side element that intercepts encodes and transmits the application graphics to the client, a viewer element on the client and a remote display protocol that transfers display updates and user events between both end points.

III. APPLICATIONS OF VNC

VNC has a widespread range of applications together with system administration, IT support and help desks. It can also be used to provision the mobile user both for hot desking within the enterprise and also to provide remote access. The system permits several connections to the same desktop, in case a priceless tool for joint or shared working in the place of work or classroom.

Figure 3: VNC Application Model

Higher version of industry-standard VNC is Enterprise Edition developed for use in business atmosphere and across the Internet. Premeditated and built from the ground up by the innovative inventors of VNC, Enterprise Edition provides robust and straightforwardly administered security with a slightest of fuss.

IV. PROBLEM STATEMENT

Cellular phones have shown a dramatic improvement in their functionality to a point where it is now possible to have cellular phones execute Java programs. As a result, cellular users throughout the world are now able to read and write e-mail, browse Web pages, and play Java games using their cellular phones. This trend has prompted us to propose the use of a cellular phone as a device for remotely controlling computers. Virtual Network Computing is a graphical desktop sharing system providing remote control via network. It supports a controlling functionality by usage of a graphical screen update from a controlled device and capturing a mouse or a keyboard. VNC system is based on RFB (Remote Frame Buffer) protocol to transmit all information between connected devices. Transmission is running on one port from range 5900-5906 using TCP/IP protocol. VNC system is required two types of applications for proper work - server application for a machine under control and client - for a supervisor (controlling) device. Client side is called viewer because of its functionality. Controlling machine is responsible for viewing a shared desktop or screen in general and capturing and converting all user activity into the RFB protocol messages. On the other side, server must to interpret all events received from client and inject them into self-system. Server should also response to graphic screen update request by sending back a desktop view to connected client. The cellular user can see and manipulate the desktop on the cellular phone. The same cellular phone to talk someone, the user must terminate the network connection.

Volume: 05 Issue: 01 Pages:1853-1857 (2013) ISSN : 0975-0290

Figure 4: Mobile Access thru VNC Architecture

4.1 RFB Protocol

Remote Frame Buffer (RFB) is a humble protocol for remote access to GUI because it works at the frame buffer level. It is appropriate to all Windows applications. RFB is the protocol used in VNC the remote end point where the user sits (i.e. the display plus keyboard or pointer) is called the RFB client or viewer. The frame buffer originates like windowing system and applications from the end point where alterations are known as the RFB server.

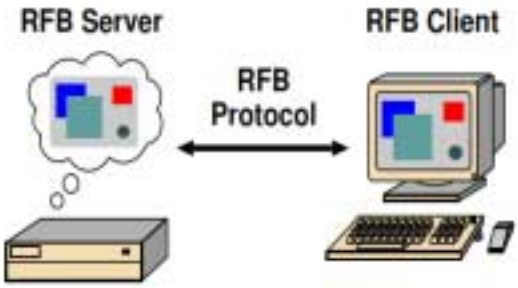

Figure 5: Remote Frame Buffer (RFB)

Although RFB started as a moderately simple protocol, it has been enriched with added features (such as file transfers) and more refined firmness and security techniques as it has established seamless cross-compatibility maintaining between the various altered VNC client and server enactments. The best RFB version is used the clients and servers to exchange a connection and the most suitable firmness and safety possibilities that they can both support.

4.2 Display Protocol

The display side of the protocol is based all over the place a single graphics primitive: “put a rectangle of pixel data at a given x & y position”. A classification of these rectangles makes a frame buffer update (or simply update). An update signifies a modification from one valid frame buffer state to another. The rectangles in an update are usually disjoint but this is not necessarily the case. The protocol update is demanded by the client that is server sent an update is only to the client in reply to an obvious entreaty from the client. This gives the adaptive quality protocol. The slower the client and the network are, the lower the rate of updates develops. With typical applications, changes to the same area of the frame buffer tend to happen soon after one another.

4.3 Input Protocol

The input side of the protocol is built on a multi-button pointing device and standard model of a keyboard. Input events are humbly sent to the server by the client every time the pointing device is moved or every time the user presses a pointer button or a key. These input events can also be fused from other non-standard I/O devices. For example, a pen-based handwriting recognition engine might make keyboard events.

4.4 Pixel Data Representation

Preliminary communication between the RFB client and server implicates a conciliation of the layout and encoding with which pixel data will be sent. This conciliation has been deliberated to make the job of the client as easy as possible. The lower most stroke is that the server must at all times be able to supply pixel data in the form the client wishes. However, if the client is able to cope equally with several different layouts or encodings, it may elect one which is easier for the server to produce. The furthermost shared pixel layouts are 16-bit or 24-bit ‘true color’, where bit fields within the pixel value interpret directly to Red, Green and Blue intensities and 8-bit color map where an indiscriminate mapping can be used to interpret from pixel values to the RGB intensities. Encoding refers to how a rectangle of pixel data will be sent on the wire. The encoding types defined at present are Raw, RRE, CopyRect, ZRLE and Hextile. In exercise we generally use only the ZRLE, Hextile and CopyRect encodings from the time when they provide the finest firmness for typical desktops.

A simple thin-client with ATM connectivity called a Videotile to be used in RFB as a remote display technology developed by Olivetti Research Laboratory, in order to hold onto the device as modest as possible RFB is used in liking to any of the current remote display technologies. More enduring and RFB found a second use when VNC was developed and

Volume: 05 Issue: 01 Pages:1853-1857 (2013) ISSN : 0975-0290

released as open source, the RFB description available on the web. Subsequently, RFB has been a free protocol which everyone can use. In 2002, Olivetti Research Laboratory was closed due to some of the crucial people at the rear of VNC and RFB established RealVNC Ltd to endure improvement of VNC and to conserve the RFB protocol.

V. User Interfaces on Mobile

The following steps give a description of the process methodology and the Figure. 6 shows snapshot of the VNC viewer as the user is accessing the desktop of a remote Windows system. Currently displayed area of the desktop called the viewport.

Figure 6: VNC Viewer

5.1 Process Methodology

5.1.1 The VNC server PC is run, which is connected to the Internet.

5.1.2 The mobile having GPRS is activated which is used for viewing other PCs, which is connected to VNC Server. In this case the PC will act as Server and mobile act as client. Connection happens through Wide Area Network.

5.1.3 Using J2ME Wireless Tool-Kit, the mobile checks PC, which is to be viewed, and it will check its IP address and password for authentication.

5.1.4 After authentication if the IP address of the PC is valid, it will connected to the server and we can view the PC using the mobile and if it is not valid the connection will not happen.

5.1.5 After connecting the PC to the server, all the operations of the PC can be performed using the mobile. For example opening a file, deleting, cut, copy, paste

5.2 VNC Servers

Inscription a VNC server is slightly tougher than inscription a client for a number of causes. The protocol is envisioned to make the client as humble as possible, so it is habitually up to the server to execute any indispensable conversions. For example, the server need to provide pixel data in the layout the client wants. A UNIX machine can run a sum of XVNC servers for dissimilar users, each of which signifies a distinct VNC desktop. Each VNC desktop is identical a virtual X display, with a root window on which numerous X applications can be displayed. The Windows Server (i. e. Win VNC) is a slight more problematic to create, because there are fewer places to insert hooked on the system to monitor display updates, and a less clearly well-defined model of multi-user procedure. The current server humbly mirrors the real display to a remote client, which means that the server is not 'multi-user'. It does, however, provide the foremost user of a PC with remote access to their desktop. Also create humble servers, which produce displays other than desktops using a simple toolkit. A "VNC CD player", for example, makes a CD player user interface using the protocol straightly without any reference to a windows system or frame buffer. Such servers can run on very humble hardware, and can be retrieved from any of the normal viewers.

5.3 VNC Clients

Inscription a VNC viewer is a simple job, as it should be for any thin-client machine. It necessitates only a trustworthy transport (generally TCP/IP), and a way of displaying pixels (either directly inscription to the frame buffer, or going through a windowing system). This includes the original RFB client (Videotile), an X-based client (which runs on Linux, Digital Unix and Solaris workstations), a Win32 client, which runs on Win Family, a Macintosh client, and a Java client, which runs on any Java – enabled web browser.

5.4 Panning & Zooming

The user can move the viewport horizontally and vertically. The viewport can be widened (zoom out) to browse its contents and narrowed (zoom in) to see the display in greater detail.

VI. Conclusion & Future Work

By physically separating the user interface from the application logic, the principle of mobile cloud computing allows to access even the most demanding applications in the cloud from intrinsically resource-constrained mobile devices. In this paper, surveyed contemporary remote display optimization techniques unambiguously tailored to the short mobile device battery lifetime, the varying and imperfect bandwidth accessibility on wireless links and the interaction latency. Although each of these solutions adequately addresses specific challenges of mobile cloud

Volume: 05 Issue: 01 Pages:1853-1857 (2013) ISSN : 0975-0290

computing, an overall approach is currently lacking.

Future work should therefore be enthusiastic to the design of a global framework, incorporating all the accessible solutions and inspiring the most appropriate solutions reliant on the current device, network and cloud server status. Implementation extending to speed up the frame rate, to incorporate more smart triangulation, to provide integrated panning and zooming of the view port, to shorten basic operations, apply speed-dependent instinctive zooming to provision incremental apprising of the VNC viewer image. The same RFB protocol will be used for the data transfer. The VNC architecture will be used for execution of the system. This system will provide mobility for users for governing their computer desktops over internet. More conveniences and features for accessing applications running on remote desktop from mobile handheld devices will be provided. Thus the extended scope of this system will prove to be helpful in providing mobility and accessing the remote desktop over the internet.

REFERENCES

[1.] Guan, Le; Ke, Xu; Song, Meina; Song, Junde,*"IEEE A Survey of Research on Mobile Cloud Computing"* pp : 387 – 392, May 2011.

[2.] Pieter Simoens, Filip De Turck member,, Bart Dhoedt member, Piet Demeester Fellow, *"IEEE Remote Display Solution for Mobile Cloud Computing" Vol: 44* ,pp : 46 - 53 ,Aug 2011.

[3.] D. Kovachev, D. Renzel, R. Klamma, and Y. Cao, *"Mobile community cloud computing: emerges and evolves,"* Proc. 1st Intl. Workshop on Mobile Cloud Computing (in conjunction with) 11th Intl. Conf. on Mobile Data Management (MDM' 10), Kansas, MO, 2010, pp. 393-395.

[4.] Mahadev Satyanarayanan, *"Mobile Computing: The next decade" Proc. 11th Intl. Conf. on Mobile Data Management (MDM'10)*, Kansas, MO, 2010.

[5.] G. Huerta-Canepa and D. Lee, *"A virtual cloud computing provider for mobile devices,"* in Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond (MCS), no. 6, 2010.

[6.] Y. Sun and T. T. Tay, *"Analysis and reduction of data spikes in thin client computing,"* Journal of Parallel and Distributed Computing, vol. 68, no. 11, pp. 1463–1472, NOV 2008.

[7.] P. Simoens, F. A. Ali, B. Vankeirsbilck, L. Deboosere, F. De Turck, B. Dhoedt, P. Demeester, and R. Torrea- Duran, *"Cross-Layer Optimization of Radio Sleep Intervals to Increase Thin Client Energy Efficiency," IEEE Communications Letters*, vol. 14, no. 12, pp. 1095–1097,DEC 2010.

[8.] H. Kawashima, K. Koshiba, K. Tuchimochi, K. Futamura, M. Enomoto, and M. Watanabe, *"Virtual PC - type thin client system,"* NEC Technical Journal, vol. 2, no. 3, pp. 42–47, SEP 2007.

[9.] A. Rudenko, P. Reiher, G. J. Popek, G. H. Kuenning *"Saving portable computer battery power through remote process execution SIGMOBILE Mobile Computer Commun."* Rev., 2 (1998), pp. 19–26.

Authors Biography

Maddali M. V. M. Kumar received his Master of Technology in Computer Science & Engineering from Jawaharlal Nehru Technological University Kakinada. Present he is working as an Assistant Professor in Department of MCA, St. Ann's College of Engineering & Technology, Andhra Pradesh and having 1.8 Years Industrial and 8.5 Years Teaching Experience, He is a Life Member in CSI & ISTE. He has published 05 papers in National, International Conferences and Journals. His research focuses on the Mobile and Cloud Computing, Network Security.