

Developing a Video on Demand Service using Mobicents JSLEE

Ray Musvibe, Alfredo Terzoli
Department of Computer Science Rhodes University,
Grahamstown, South Africa
+27 73 027 6630
g08m3079@campus.ru.ac.za
A.Terzoli@ru.ac.za

Abstract -This paper describes the development of a Video on Demand service using Mobicents JSLEE , an open source Service Delivery Platform. A discussion is given on the component architecture of Mobicents JSLEE and how the programmatic components interact. An evaluation of available open source IMS clients with video support for the envisioned service is also given. ¹

Index Terms – Mobicents JSLEE, IMS client, Service Delivery Platform (SDP's), Video on Demand.

I. INTRODUCTION

Video on Demand (VoD) services allow users to watch video content of their own choice at a time of their own convenience, unlike traditional broadcast TV, in which users can only watch TV shows at times set by the broadcaster [3] [5]. In addition, VoD services allow users to play ðtrick modesö on video streams (Pause, Rewind and Fast Forward) [1].

New services that incorporate voice, web and video have emerged [2]. Convergence has become a reality as the Next Generation Network (NGN) gradually takes shape. Consequently, suitable platforms that accommodate service creation for the various forms of media and information have to be adopted.

Telcos have traditionally relied on proprietary solutions for the provision of their services [4] but there has however been increased interest in component based, Java containers [8].

Mobicents JSLEE is an emerging open source SDP, and it is the first and only open source JSLEE certified event driven application server [2]. This work explores the viability of developing a VoD service using Mobicents JSLEE.

This paper is divided into three sections. The first section gives an overview of the Mobicents platform and discusses open

source IMS clients under related work. The second section presents current work and in the third and final section, a discussion of future work and conclusions are given.

II. RELATED WORK

A. MOBICENTS

Mobicents is essentially an application server that complements JEE to allow the convergence of voice, video and data[2].

SLEE is an already existing Telco execution environment that defines a low latency, high throughput environment for event driven applications. JSLEE is the Java implementation of SLEE [6].

The diagram below gives an overview of Mobicents JSLEE.

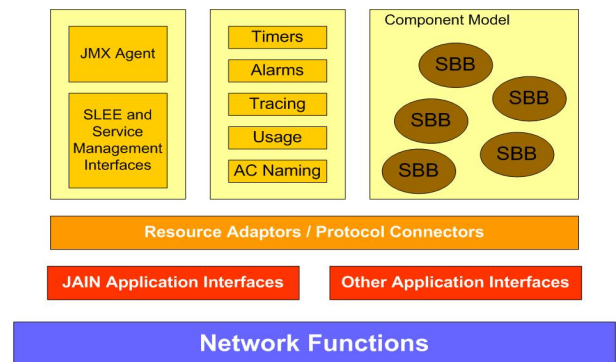


Fig 1. Mobicents JSLEE overview .

The following summarises the major components of JSLEE [7].

a. Service Building Blocks(SBB's)

SBBs contain the application logic for services deployed in the SLEE. SBBs can invoke child SBBs to handle some of the processing .

b. Resource Adaptors

Resource Adaptors allow events received from clients on the network to be fired into the SLEE in a way that it understands. Resource adaptors allow Mobicents to be protocol agnostic.

c. JSLEE facilities

JSLEE provides facilities like Timers and Alarms that are mainly useful in the monitoring of the SLEE's internal state.

¹ **Ray Musvibe** is currently studying towards his Master of Science Degree in Computer Science at Rhodes University, Grahamstown.

Alfredo Terzoli is currently the Project Director of The Telkom Center of Excellence at Rhodes University, Grahamstown.

This work was undertaken in the Distributed Multimedia Center of Excellence at Rhodes University, with financial support from Telkom SA, Business Connexion, Comverse, Verso Technologies, Stortech, Tellabs, Amotole Telecommunicaton Services, Mars Technologies, OpenVoice, Bright Ideas Projects 39, THRIP and the National Research Foundation.

d. Management Interfaces

Management interfaces allow external management clients to monitor the internal state of the SLEE.

B. IMS CLIENTS

Several open source IMS clients exist, but the majority of these do not have video capabilities. Below is an assessment of open source IMS clients with video support that can be used to provide an IMS frontend for the proposed service.

a. UCT IMS client

The UCT IMS client was designed to be used in conjunction with the Fraunhofer FOKUS Open IMS core. The UCT IMS client has ample functionality including VoD/IPTV support, XDMF support, instant messaging and presence. The major drawback is that it is designed to be used on the Linux platform, making it unavailable to the ubiquitous Windows platform [9].

b. Mercurio

Mercurio is a Microsoft Windows based IMS client with 3GPP/IMS compliance. It has support for a wide range of video codecs with presence and XDMF support. The major drawback is that there is no initial support for VoD/IPTV implementation. This functionality will have to be added [11].

c. Ekiga

Ekiga is an open source IMS client for the GNOME environment. Again this client does not have initial VoD/IPTV support and this will have to be added [10].

This evaluation clearly shows that currently, the UCT IMS client is best suited for use as an IMS client in the proposed service because of the existing support for VoD/IPTV, though it will only limit users to the Linux environment.

III. CURRENT PROGRESS

A. Service architecture

An initial step for the proposed service was also to develop an initial architecture. Fig 2 below provides the initial architecture of the proposed service.

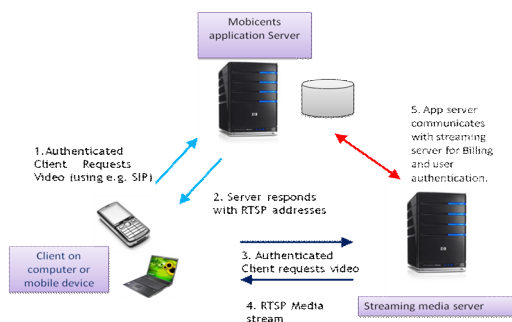


Fig 2 Initial Architecture of Proposed Service.

B. Building Application Logic

We are also presently developing the various SBBs that will contain the application logic for the proposed service.

C. Tweaking the UCT IMS client

Current work involves tweaking the UCT IMS client to best fit the proposed service.

IV. FUTURE WORK AND CONCLUSIONS

The main aim of this work is to create and evaluate a robust VoD service over the Mobicents platform.

Future work will mainly involve evaluating the performance of the proposed service, which will be deployed for testing.

Initial work has mainly involved using SIP based IMS clients, support for other protocols like HTTP will be added and evaluated.

An administrative console will also be added for the proposed service.

V. REFERENCES

- [1] Dikgole V and Ventura N, "Video on Demand Service for Next Generation Networks". SATNAC Proceedings 2008.
- [2] Durelle J, "Mobicents Communications Platform", JavaOne presentations 2008. Last Accessed March 2009. [Online]. Available: <http://people.redhat.com/vrlev/blog/JavaOne%202008%20Presentation%20-%20Jean%20Deruelle.pdf>
- [3] Koch C, "Video On Demand - Television For A New Millennium", page 1, GRIN Verlag Publishing, 2007.
- [4] Osterwalder A, Ondrus J and Pigneur Y, "Skype's Disruptive Potential in the Telecom Market: A Systematic Comparison of Business Models". HEC Lausanne working paper, 2005.
- [5] Roux, K. "Video-on-demand over satellite". IST-Africa 2007 Conference, Maputo, Mozambique.
- [6] Sun T. and Tucker D. "A SoftBridge with Carrier Grade Reliability Using JAIN SLEE". Proc. South African Telecommunications Networks & Applications Conference (SATNAC 2004), Stellenbosch, South Africa
- [7] Swee B L, Ferry D, "JAIN SLEE 1.0 specification final release", Sun Microsystems and Open Cloud. Last Accessed May 2009. [Online]. Available: http://phoenix.labri.fr/documentation/sip/Documentation/Papers/Programming_SIP/Paper_Publication_and_Draft/Java_SIP/jain_slee-1_0-fr-spec.pdf
- [8] Swee B L, Ferry D, O'Doherty P and Page D, "JAIN SLEE tutorial". Last Accessed April 2009. [Online]. Available: www.jainslee.org/downloads/jainslee-tutorial-04.pdf
- [9] <http://uctimsclient.berlios.de>
- [10] www.gnomemeeting.org
- [11] www.mercurio.net/