# Development of an IMS Compliant, Cross Platform Client Using the JAIN SIP Applet Phone

Walter T. Muswera, Alfredo Terzoli.

Department of Computer Science and Information Systems
Rhodes University, P. O. Box 94, Grahamstown 6140
Tel: +27 46 6038111, Fax: +27 46 6225049
email: {g09m3278@campus. ru.ac.za; A.Terzoli@. ru.ac.za}

Abstract - Several open source SIP/IMS Clients are currently in use in the Rhodes University Convergence Research Group. However, there is no single Client that provides researchers with all the required functionality needed to test the applications they have developed. Integrating advanced features into a single Client can help to speed up testing as well as provide a variety of innovative communication services. In this paper we present an overview of two popular SIP/IMS Clients currently in use. We also discuss the features and design architecture of a Client called JAIN SIP Applet Phone (JSAP) which will be used as a basis for the development of a new IMS compliant Client. We then analyse what functionality JSAP lacks and propose features that need to be implemented.

Index Terms - SIP/IMS Client, IMS Compliant.

# I. INTRODUCTION

The IP Multimedia Subsystem (IMS) is a service delivery framework approved by the 3GPP that enables the development and/or delivery of IP based multimedia applications, such as Presence, Video Sharing and Instant Messaging (IM) [2]. IMS is the catalyst for convergence and enabler for service driven development and delivery of new applications. It enables the convergence of different wireless and fixed access technologies for the creation, delivery and consumption of multimedia services [3].

The Rhodes University Convergence Research Group (RUCRG) is mainly concerned with current trends in the move towards converged service platforms for Next Generation Networks (NGN) and the Internet. Research covers areas such as complex service orchestration, policy frameworks for service development, development of toolkits for services such as IPTV, LBS and VoD using open standards. These applications are built on platforms such as the Mobicents Application Server, Opensource IMS Core, Kamailio and Asterisk. These projects therefore rely heavily on availability of SIP based and IMS compliant Client software for proper testing and evaluation.

SIP is an application layer protocol for initiating, modifying, or terminating communication and collaborative sessions over Internet Protocol (IP) networks. SIP is the chosen session control protocol for IMS [3]. Furthermore, there is need for a Client that is not limited to any particular operating system platform. Several open source clients are available but none has all the required features to test the applications being developed.

This paper describes the development of a Java based, IMS compliant Client using the JAIN SIP Applet Phone (JSAP) as the foundation of the new Client. A review of some existing SIP/IMS Clients currently in use in the

RUCRG is provided. A discussion is also provided of the current status and architecture of the JSAP and the possible extensions that need to be implemented.

## II. CHOICE OF TECHNOLOGY

The need to produce a client that is portable across different platforms resulted in Java being the language of choice. Java is both a compiled and interpreted language. Any computer that has the Java Virtual machine (JVM) installed can interpret compiled Java bytecode. Furthermore, the Java Community Process (JCP) through the Java APIs for Integrated Networks (JAIN) initiative, define Application Programming Interfaces (API) for using Java technologies to provide next generation telecommunications services [2].

# III. SIP/IMS CLIENTS

Most SIP/IMS Clients can only be used on specific platforms [4] and support a limited range of video and audio codecs [3] [7]. An assessment is given below of the feature sets of two open source clients that are currently in use in the RUCRG.

# A.SIP Communicator

SIP Communicator is an open source audio/video software phone and Instant Messenger. It was built using the OSGi framework. It supports audio calls with SIP and Jabber, as well as video calls with SIP. Furthermore, it supports Instant Messaging with a variety of protocols such SIP and Jabber. Firewall traversal and basic NAT is implemented using STUN [1]. However it is built on top of JMF as a media API which Sun ceased to support in 2003 and requires a variety of plugins to support various codecs.

# B. UCT IMS Client

The UCT IMS client is a free open source implementation of a 3GPP IMS Client [5]. It supports a variety of IMS applications such as IM, Presence, VoD/IPTV, as well as the XCAP protocol among others. However, it was designed to be used on the Linux platform thus making it unavailable to other major platforms such as Windows and Mac OS.

The evaluation shows that Clients exist that possess some of the required features needed for testing applications but there is no single client that allows researchers to perform testing without having to switch between clients or adjust their systems.

# IV. JSAP DESIGN AND ARCHITECTURE

The JSAP is an open source project which possesses some of the basic features such as voice and text instant

messaging which are required in an SIP/IMS compliant client, but is in need of several extensions in order to realize full IMS compliance.

JSAP uses the Java Media Framework (JMF) as well as JAIN SIP (a low level Java API) for SIP signaling. Presence in JSAP is implemented in a peer to peer manner but ideally should also support Client/Server. The video implementation in the JSAP is not fully functional and requires attention. For example, JMF fails to initialize video capture devices in Ubuntu but works under Windows. JMF also lacks support for some of the new high quality well compressed codecs such as Theora.

Figure 1 below gives an overview of the architecture of the JSAP with the dotted lines showing the various components of the client that need to be added or modified to make it IMS compliant.

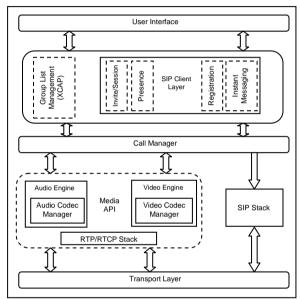


Figure 1: JSAP Architecture

## V. CURRENT WORK

An investigation into alternative open source media APIs to the now unsupported JMF is currently underway. The process involves testing functionality by building prototypes and comparing audio/video/streaming support as well evaluating support for these media APIs. The aim is to integrate the most appropriate media API into the client. An overview of the projects being investigated is given below.

## A. Freedom for Media in Java (FMJ)

FMJ is a new open source project which was started to implement and extend JMF [7]. FMJ is API compatible with latest JMF (2.1.1e) meaning that one can use existing JMF codes and run applications. However several areas of the project under development and at times workarounds are needed if existing JMF codes do not work.

## B. GStreamer

GStreamer is framework for creating media applications. It is based on plugins that provide the various codecs and other functionality. Plugins can be linked and arranged in a pipeline making it possible to write any type of streaming multimedia application [4]. *GStreamer-Java* is a Java interface to the GStreamer framework. It uses Java Native Access (JNA) to access GStreamer functionality. GStreamer itself and these java bindings are portable across operating

systems [6].Similar to IMS the GStreamer-java project is still under active development

## VI. CONCLUSION

The main focus of this work is to provide a comprehensive IMS compliant Client that is portable across platforms. The hope is that the Client will be useful to the RUCRG as well as to other researchers. Furthermore there is a need to make the Client extensible and interoperable with leading SIP/IMS servers.

## VII. FUTURE WORK

Work that still needs to be done in the future includes:

- Evaluation of FFMPEG, JFFMPEG, VLC and JVLC media APIs.
- Integration of XCAP (XML Configuration Access Protocol) support in JSAP is also needed
- Analysis of other major open source Clients and compare the feature sets to get an idea of what kind of SIP features they have that may be good to include in our Client.
- Integration of SIP Extensions for IMS.

## VIII. REFERENCES

- Ivov, E. SIP Communicator (2007). Last Accessed April 2010 [Online]. Available: jres.org/planning/slides/132.pdf
- NMS Communications. SIP/IMS Client Applications for Operators, Terminal Vendors, and Equipment Vendors. Last Accessed May 2010 [Online]. Available: http://www.nmscommunications.com/NR/rdonlyres/9218414C-8C7C-4543-A898-BA5A5E5D7A5A/0/MP\_IMS\_Overview.pdf
- Oguejiofor, E.; Bazot, P.; Georges, B.; Huber, R.; Jackson, C.; Kappel, J.; Martin, C.; Subramanian, B.S. and Sur, A. Developing SIP and IP Multimedia Subsystem (IMS) Applications (2007) pages 27-39 Available: http://www.redbooks.ibm.com/abstracts/sg247255.html
- Taymans, W.; Baker, S.; Wingo, A.; Bultje, R. and Kost, S. GStreamer Application Development Manual(0.10.25.3) http://gstreamer.freedesktop.org/documentation/
- Waiting, D.; Good, R.; Spiers, R. and Ventura, N. "The UCT IMS client," tridentcom, pp.1-6, 2009 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops, 2009
- Gstreamer-java Java Interface to the Gstreamer Framework. Last Accessed May 2010 [Online] Available: http://code.google.com/p/gstreamer-java/
- Home FMJ Last Accessed April 2010 [Online]. Available: http://fmj-sf.net/
- JMF 2.1.1 Supported Formats. Last Accessed May 2010 [Online]. Available:
  - http://java.sun.com/javase/technologies/desktop/media/jmf/2.1.1/form ats.html#Capturers

**Walter Muswera** received his honours degree in 2009 from Rhodes University and is presently studying towards his Master of Science Degree in Computer Science at the same institution. His research interests include converged telecommunication networks.

**Alfredo Terzoli** is Professor of Computer Science at Rhodes University, where he heads the Telkom Centre of Excellence in Distributed Multimedia. He is also Research Director of the Telkom Centre of Excellence in ICT for Development at the University of Fort Hare. His main areas of academic interest are converged telecommunication networks and ICT for development.

The work reported in this paper was undertaken in the Telkom Centre of Excellence in Distributed Multimedia at Rhodes University, with financial support from Telkom, Comverse, Tellabs, Stortech, Amatole Telecom Services, Bright Ideas 39 and THRIP.