An Architectural Framework for Delivering SIP-AS Multimedia Services Based on JADE/OSGi Technology

Renato B. Cabelino Ribeiro[1(✉)], Magnos Martinello[1],
Celso Alberto Saibel Santos[1], and Rosane Bodart Soares[2]

[1] Departamento de Informática, UFES, Vitória, Brasil
renato@ifes.edu.br, {magnos,saibel}@inf.ufes.br
[2] Departamento de Engenharia Elétrica, UFES, Vitória, Brasil
rosane@ele.ufes.br

Abstract. This paper proposes a scenario of service-oriented architecture based on OSGi technology, in combination with multi agents systems developed using JADE environment. A key part of the scenario is the new architecture for SIP application server as part of the IMS Core network. As a proof of concept, a simulated environment for a televoting service was implemented. The major contributions of this work are the identification and analysis of jitter, variation of jitter, packet loss, load capacity and CPU utilization of the JADE/OSGi SIP-AS. Test results validate the approach and show good overall performance.

Keywords: IMS · SIP-AS · JADE · OSGi · Multimedia services · Televoting

1 Introduction

The world of digital communications is quickly moving away from a dependence on access methods to service-oriented providers. This paper introduces a service-oriented architecture combined with multi-agents systems as a hosting platform for telecommunication supplementary services on IP Multimedia Subsystem (IMS).

The JADE platform, integrated with the OSGi framework, is the proposed agent-based development environment [1, 2]. This approach allows for a more flexible and dynamic form of service provisioning over the IMS architecture, allowing services to be negotiated on demand according to the current environment requirements (services rules, QoS requirements, interaction parameters, etc.).

This article presents the implementation of a SIP Application Server (SIP-AS) on OSGi service-oriented architecture, integrated with JADE framework for the creation and provision of multimedia services on IMS. In Sect. 2, the related works are described. The technology is presented in Sect. 3. Section 4 describes the design and implementation of the televoting service, and the result analysis. Conclusions are described in Sect. 5.

S. Mumtaz et al. (Eds.): WICON 2014, LNICST 146, pp. 122–128, 2015.
DOI: 10.1007/978-3-319-18802-7_17

2 Related Works

This section analyzes exclusively SIP approaches that have correlation with customized telecommunications services (for example, additional services, toll free phone services, Televoting, local number portability) or with the framework presented in this work. Oliveira et al. [8] propose two approaches for implementing number portability service in IMS networks, tested on an AS (Application Server) according to the standards of the General Regulations of Portability. In the first approach, the AS performs the function of local number portability without call states control, routing every call originated by IMS to AS, or those based on a numeric phone context through an Initial Filter Criteria (IFC). In the second approach, the number portability service acts as a back-to-back user agent (B2BUA), i.e. in a leg termination call aimed to a ported user, configures an IFC to conduct the call routing for an AS, which acts on behalf of the user ported and initiates a new call to the correct destination. For the implementation of AS, SIP Servlet technology was created by the authors.

Munadi et al. [9] propose the design and implementation of VoIP services with OpenIMS and ASTERISK, interconnected by an ENUM server which develops numerical mapping function between the two servers. The authors observed the proposed environment according to: (a) performance measures for each server; (b) the Post Dial Delay (PDD) and (c) of the same CPU consumption. The values measured and analyzed in (a) identify the service time consumption on the part of the SIP signaling system. In (b), three scenarios were tested where the Traffic Analyzer WIRESHARK was used in order to capture and analyze traffic from the User Agent Caller from its application until its acceptance by the counterparty in the call, which allowed the analysis of the PDD in each test performed. For (c), were used the TOP utility from the operating system itself in order to obtain the maximum CPU value throughout the experiments.

Li et al. [10] implement two IMS services – a chat room (SIP-IM) and Presence services, in a SIP-AS. SIP-AS architecture used is based on Mobicents SIP Servlet component (MSS). In addition, we used the OpenIMS Core to IMS Core Network implementation. The authors have developed a use case diagram and class diagram for the services analyzed. For the test scenarios, XML templates were made to present the requirements and the design of both services.

3 Technology Base: IMS and JADE/OSGi Integration

IMS is an evolving definition of an architecture that addresses the continuing demands and frustrations of users and enterprises. The ultimate goal of the architecture is to define a model that separates the services offered by fixed-line, mobile, and converged service providers from access networks used to carry those services [3].

The layered approach proposed by IMS increases the importance of the application layer as services are designed to work independently of the access network. IMS is designed to bridge the gap between them [4]. It offers more flexibility for telecom operators to manage different services with distinct requisites (e.g.: bandwidth, latency, jitter, etc).

JADE is a middleware for the development and execution of peer-to-peer applications based on the agent paradigm that can easily work and interoperate on traditional or wireless network environments. JADE internal architecture is currently the only architecture entirely compliant with FIPA standards [5]. According to [6] and [1], the JADE platform can offer the following: Graphical interface which allows monitoring, debugging and logging; Components which can be distributed over the network; Mobility and cloning of agents as well as multi-tasking scheduling; Lifecycle management, name and yellow pages services, point-to-point message transport service, speech-act message structure and ontology service; Interoperability with other platforms that offer support to FIPA standards.

The development of applications using OSGi can be accomplished through the combination of collaborative, reusable modules associated with descriptive information on their metafiles which include service-related input that must be instantiated/imported to achieve a consistent execution of the modules [7]. Also, the services provided by OSGi implement a JAVA interface for registering on local service registries. Through this centralized control model, the modules (or bundles) can verify their service dependencies.

Furthermore, the OSGi services platform offers developers the means to maximize the use of platform independent resources and dynamic updating of JAVA modules, allowing development of services for devices with limited computing resources, widely used in corporate environments. New services registration as well as research and maintenance of pre-existing services (including their uninstallation from the system), services status notifications and follow-ups on bundles lifecycle can be carried out in a simple and efficient way.

4 Design and Implementation

Figure 1 introduces the scenario for implementation Televoting service (or another multimedia service) through the utilization of the JADE platform along with the OSGi framework, running on SIP-AS.

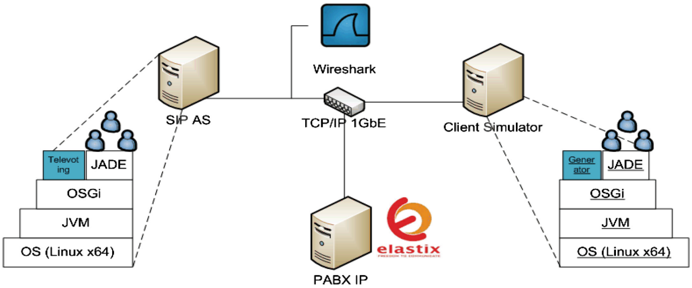

Fig. 1. Network Topology.

4.1 Televoting Service Lifecycle

Upon start, the OSGi Televoting service bundle loads its configuration parameters from a file named televoto.conf and automatically creates a number of attendant agents on the JADE platform. Each agent registers itself on IP PBX as an extension and become operational. All registered agents are grouped in a unique number. When a call arrives at that unique number, the IP PBX redirects it to one of the registered agents and the call is then processed. The rules for redirection (first available, ring all, last called, etc.) depend on the IP PBX distribution. In the proposed scenario the programmed rule is first available. As described above, televoting agent register along IP PBX and became ready to work. Once created, they turn visible on JADE GUI (inside MAIN container) and your control is now managed by JADE framework.

4.2 Validation Tests

We have developed a sequence of tests to validate integration of JADE/OSGi as a SIP-AS. The parameters observed in the experiment are: jitter, variation of jitter, packet loss and CPU load of the SIP-AS. To support this scenario, the equipment presented in Fig. 2 and detailed in Table 1 below, were used.

Table 1. Test equipment's specification.

ITEM	EQUIPMENT
SIP-AS, Client Simulator	Dual Intel Xeon with 4 core/processor, 20 GB RAM, Intel Gigabit Ethernet, OS Linux Server 12.04 x64
IP PBX	Single Intel Xeon with 4 core, 8 GB RAM, Intel Gigabit Ethernet, ELASTIX Custom Distro
Switch	H3C-2928 24 ports Gigabit Ethernet
Network Analizer	Intel Core2 Duo, 4 GB RAM, Atheros Gigabit Ethernert, OS Microsoft Windows XP SP3

In each test, the client simulator performed a load of calls to its counterpart in the televoting service (a client for each service agent). This load of calls was parameterized in the configuration file of client simulator such that it is executed one or more times, depending on the amount of redials parameter set.

With this approach, we identified the capacity of the SIP-AS in handling calls faster without the need of integration with the IMS Core. During testing, all network traffic was captured by the Analyzer for subsequent analysis.

The methodology used in the tests was developed according to the following profile: (a) the whole SIP-AS infrastructure is initialized; (b) the packet capture is initialized in the WIRESHARK; (c) the client Simulator is initialized, running 100 concurrent calls to the Televoting service and (d) at the end, the entire environment is shutdown. In each test, the client Simulator is reconfigured to generate additional concurrent calls as shown in Table 2, up to a total of 1000 calls, in order to identify its impact with respect to jitter

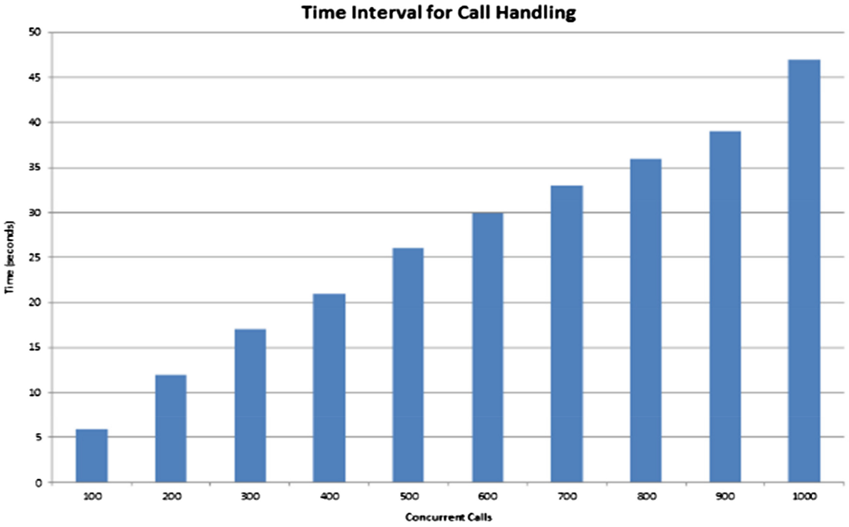

Fig. 2. Time interval for handling *n* calls.

and jitter variation. Note that the average value of jitter, as well as your variation (MAXDELTA), increases as the load of concurrent calls grows.

Table 2. Mean values of Jitter and its MAXDELTA.

Concurrent Calls	Jitter (ms)	MAXDELTA (ms)
100	0,72	111,80
200	1,74	213,07
300	3,03	354,79
400	3,96	401,76
500	4,94	485,36
600	5,93	522,16
700	7,29	629,50
800	9,71	784,57
900	9,91	716,44
1000	11,98	853,66

In the jitter column, we observe that even with a high load of concurrent calls, the values remain at an acceptable threshold. However, the same does not occur with the MAXDELTA values, because the higher the value presented more occurrences of gaps in the audio message from the televoting were perceived on the Wireshark RTP-analysis.

About CPU usage analysis, the minimum and maximum load values were between 60 and 80 percent. When added more processing resources to SIP-AS, the average value was reduced by half, leaving around 30 to 35 percent.

The values of packet loss measured were insignificant in all tests and did not influence the communication on media plan of televoting service. As noted in Fig. 2, the time spent to attend the n concurrent calls in Televoting service presented a behavior near the linearity, while the average of call handled (per second) showed practically stable after the amount of 400 concurrent calls.

5 Conclusions

We present the design and the development of a service oriented architectural framework to support the provisioning of IP Multimedia Services (IMS). Specifically, we follow a totally different approach from the already existing approaches, presented in the literature, which are about the design and the development of functional solutions based on the combination of the OSGi and JADE technologies, by focusing in the same time to the deployment of innovative techniques/solutions that enable the multimedia sharing over IP between IP enabled devices that are able to participate a televoting service.

The tests performed indicate good performance results in concurrent call handling. In addition, the system response time in the processing of televoting calls is close to those found in the same service offered in PSTN Intelligent Network (IN). Server CPU consumption is high and can be attenuated with the vertical scaling (provisioning of more hardware resources, already tested as shown in subsection 4.2) or horizontal (distribution of the service on a clustered JADE/OSGi). In a future work, we are programming the system for automatic scaling calls (load balancing, etc.) as well as the mobility of service agents and model integration with OpenIMS Core.

References

1. Bellifemine, F., Caire, G., Greenwood, D.: Developing MultiAgent Systems with JADE. Wiley, Inglaterra (2007)
2. OSGi Alliance. OSGi Service Platform: Core Specification, Release 4, Version 4.2. Technical report (2009). Disponível em http://www.osgi.org/download/r4v42/r4.core.pdf
3. Salchow Jr., K.: Introduction to the IP Multimedia Subsystem (IMS): IMS Basic Concepts and Terminology. http://www.f5.com/pdf/white-papers/ims-introduction-wp.pdf
4. Poikselkä, M., Mayer, G.: The IMS: IP Multimedia Concepts and Services, 3rd edn. Wiley, Chichester (2009)
5. Bellifemine, F., Caire, G., Poggi, A., Rimassa, G.: JADE White Paper, disponível em: http://jade.tilab.com/papers/2003/WhitePaperJADEEXP.pdf
6. Wooldridge, M.: An Introduction to MultiAgent Systems, 2nd edn. Wiley, Chichester (2009)
7. Ribeiro, R.B.C., Soares, R.B.: The application of JADE and OSGi technologies in the telecommunications services architecture. In: EUROCON 2011 (2011)

8. Oliveira, R.G., et al.: An application server approach for number portability in IMS networks. In: IEEE - SBrT - ITS 2010 - International Telecommunications Symposium, 6–9 September, Manaus, Amazonas, Brazil (2010)
9. Munadi, R., et al.: Design and implementation VoIP service on open IMS and asterisk servers interconnected through enum server. Int. J. Next-Gener. Netw. (IJNGN) **2**(2) (2010)
10. Li, K., et al.: Two exploring experiments on IMS service based on SIP AS. Int. J. Multimedia Ubiquit. Eng. **9**(4), 375–386 (2014)