

©2008 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Re-configurable VoIP Interactive Response

Alex Talevski, Elizabeth Chang, Fellow, IEEE and Tharam Dillon, Fellow, IEEE

Digital Ecosystems and Business Intelligence Institute (DEBII),
Curtin University of Technology, GPO Box U1987, Perth, Australia 6845
e-mail: (Alex.Talevski, Elizabeth.Chang, Tharam.Dillon)@cbs.curtin.edu.au

Abstract— Organizational alliances are rapidly being formed as a means for effective cooperation with a common goal within a targeted value chain. The combination of such communication, coordination and cooperation leads to new organisational forms and scenarios within the Digital Ecosystem space that require technological support. Convergence refers to the move towards concurrently coupled telecommunications services with enterprise and internet data.

Voice over Internet Protocol (VoIP) is a telecommunications solution that uses a data network such as the internet to transport voice traffic. VoIP data is transported alongside other data over an existing network. VoIP has introduced significant cost savings, functionality and convenience that is functionally comparable to traditional telecommunication offerings. However, current IT&T solutions have failed to use this powerful technology beyond simple telephony applications.

This paper presents a flexible VoIP Interactive Voice Response (IVR) interface that offers simplified convergence of telecommunications services that are coupled with enterprise and internet data.

I. INTRODUCTION

Extended enterprises are rapidly being formed as a collaborative approach within a targeted value chain. The combination of such communication, coordination and cooperation leads to new organisational forms and scenarios within the Digital Ecosystem space that require technological support [1, 2]. Convergence refers to the move towards the use of a single united interaction medium and media as opposed to the many that we use today [3]. Such a solution enables telecommunications services that are concurrently coupled with enterprise and internet data. The ability to visualize a concept via images, graphs, tables and procedures while communicating over the telephone greatly enhances interaction. Communication becomes pleasing, meaningful, effective and efficient. Feature-rich services allow for actions to be taken on the move with greater precision and faster response to market drivers. Therefore, data and telecommunications convergence promises a wide range of possible solutions that will increase productivity and flexibility, reduce costs, and provide new opportunities and revenues for extended enterprises.

Converged voice and data services have rapidly emerged as a popular alternative to existing telecommunications networks and computer services. Many sources [4 – 7] indicate that converged voice and data networks are growing from only 100,000 US households in 2004 to more than 12 million by 2009. However, flexible access to

converged telecommunications and data services are virtually non-existent. Convergence solutions are largely isolated to static environments where fixed Personal Computers (PC) and network connections are used in conjunction with various customized software tools that simulate pseudo united voice and data streams. Generally, data presented on the internet and in enterprise applications is not available on voice networks and devices and vice-versa.

A. Voice over Internet Protocol

Voice over Internet Protocol (VoIP) also termed IP Telephony is a telecommunications solution that uses a data network (such as the internet) to transport voice traffic. VoIP is commonly used by people as an alternative telecommunications medium [8]. VoIP has introduced an alternative telecommunications medium that offers significant cost savings, functionality and convenience that is functionally comparable to traditional offerings. Telephone calls can be transmitted with little or no loss in functionality, reliability, or voice quality.

B. Interactive Voice Response Telecom

Interactive Voice Response Telecom (IVR) systems provide computer controlled telephone answering and routing functions as well as facilities for the collection and provision of information. Interactive menus allow callers to input data using the telephone keypad and/or simple voice prompts. IVR devices gather information and route calls based on user defined steps, commands and responses to prompts as required.

Industry has failed to make the most of VoIP and IVR technologies. Use of VoIP beyond simple telecommunications oriented applications has been limited. Existing systems rarely provide an integrated approach where more than communication services are provided. Due to the versatile nature of today's enterprise, a flexible, feature-rich, adaptive and widely accessible converged solution is required.

Section 2 provides a re-configurable software overview. Section 3 details the adopted reconfigurable plug-in framework. Section 4 outlines the re-configurable VoIP interactive response solution. Section 5 documents the implementation. Section 6 concludes the paper.

II. RE-CONFIGURABLE SOFTWARE

Component-based software engineering is a way of raising the level of abstraction for software development so

that software can be built by easily reusing previously designed, implemented and refined software building blocks.

Recent research [9] has outlined the requirement for a component-based approach in order to solve software complexity and flexibility issues. Component-based approaches achieve loose coupling among interacting software components where disparate components interact using a common interaction protocol and architectural constraints. Such architectures define how components are located, integrated, managed, and monitored.

A. Component

Components are the smallest possible reusable pieces of software. They are self-managing, independent and ideally can be reused in multiple environments. Components extend the existing object principles by strengthening the role of an interface. A component interface separates the component implementation from its interaction. It contains a collection of provided and required operations and attrib-

utes that govern the services that a component provides. Such services are functions that are well-defined, self-contained, and may or may not depend on the context or state of other services [10]. An attribute is a named property value that defines the characteristics and state of a component. An operation is the implementation of a specific service that represents the dynamic behavior of a component. Operations are defined by their input and output parameters and pre and post conditions.

By strategically abstracting a component's internals through an interface, components become well isolated and standardized. Composite component architectures are formed from layered components where components at the lower layers provide component services to the components above them.

The component Meta data model illustrated in Figure 1 represents an explicit Meta specification of a single software component and its interface.

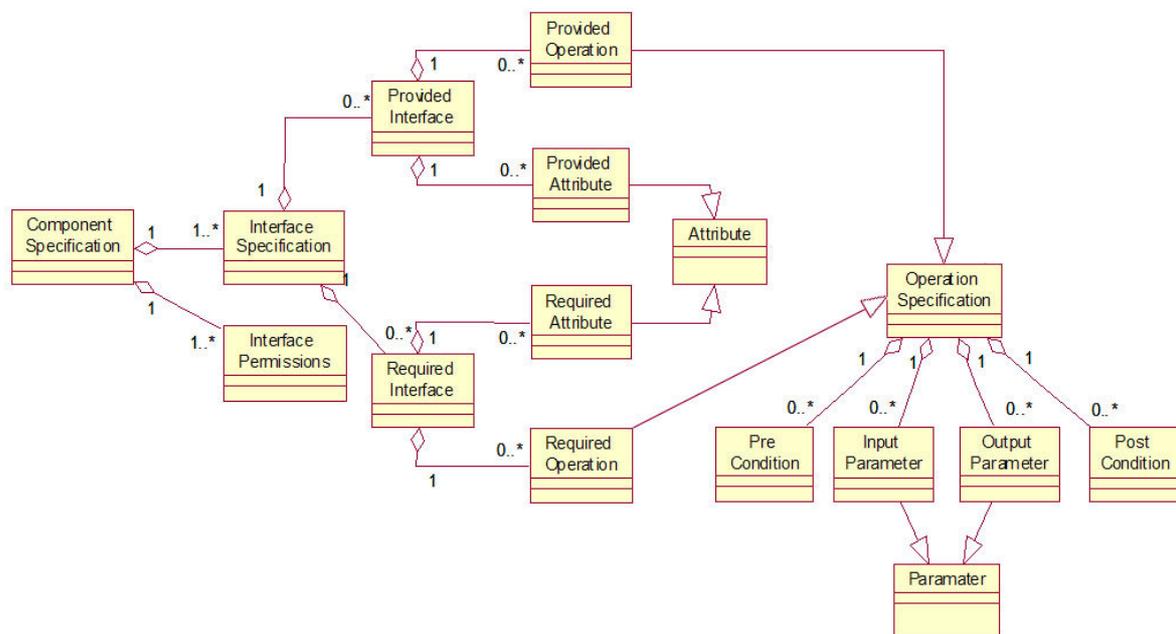


Fig. 1 Conceptual Meta Model of a Component

A reconfigurable component architecture defines the following component interface;

- Plug-in Discovery – Plug-in discovery is used to find, identify, describe and use available voice plug-ins. In order to locate plug-ins, the service consumer broadcasts a request for a lookup service. Service providers respond to this request with a lookup proxy.
- Plug-in Identification – The lookup proxy is used to define voice plug-in characteristics.
- Plug-in Query – A consumer is also able to query a provider lookup proxy for available services.
- Plug-in Invocation – When a service is selected by a consumer the consumer and provider bind together and the selected service is invoked.

B. Tailorability

Tailorable software allows generic software to satisfy the specialized, rapidly changing, unclear and / or evolving changes in system requirements. It provides a means for the straightforward creation and modification of software at runtime based on multiple levels of detail and complexity.

Using a reconfigurable plug and play component-based framework [11] as a basis for the creation and modification of software, it possible to construct, customize, integrate and evolve software in a simplified way.

We have developed a reconfigurable plug and play component-based software framework and platform to

ease software construction, customization, integration and evolution. This framework and platform constitute the skeletal support and software tools that are used as the basis for constructing component-based applications. They are directed at: increasing software quality and performance, reducing software development and maintenance costs and easing runtime software modification.

III. RECONFIGURABLE PLUG-IN FRAMEWORK

Using a conceptual Meta model framework, components, their interfaces and their interconnections are defined as a Meta specification. This explicit model supports software tailoring by explicitly defining application constructs, their composition and interaction. External entities (other applications and constructs) may access this Meta data in order to evaluate the composition of a tailored system or use the services that it provides.

A component collaborates with other components through its configuration and composition interconnections. Such interconnections denote the specific component associations. A consumer / provider relationship is employed where the services of a server component are utilized by a client component.

A. Configuration

A component may expose a set of required and provided attributes which characterize a components configuration, operation parameters and state.

Attribute connectors are used to connect required and provided attributes. Once connected, a required attribute always requests the value it requires from the provided attribute that it is connected to. Attribute adaptors are connectors that translate component interaction. An attribute adaptor can be used to convert an incompatible provided attribute so that it satisfies system [12].

B. Composition

A component implemented by combining the functionality provided by other components is referred to as a composite component. The process of developing component composites is referred to as component composition. Component composition can be seen as a way to master software complexity. It is an iterative process where the definition of increasingly complex applications is performed by recursively composing components.

A composition is typically formed from many interconnected components that are constructed in a layered and hierarchical manner. Connectors and adaptors provide a level of indirection that reduces dependencies among components (Figure 2). Interconnected compositions are coupled using operation and attribute connectors and adaptors.

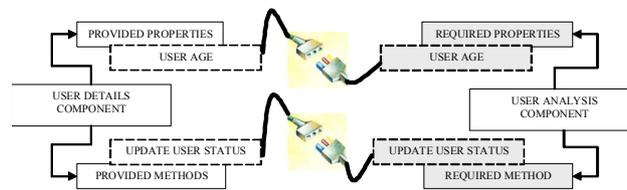


Fig. 2 Plug-in a component

Like attribute connectors, operation connectors are used to connect required and provided operations. Once connected, a required operation always calls the provided operation that it is connected to. Operation adaptors may be used to connect an incompatible provided operation.

C. Interaction

Whenever a component passes a message to another, the two components are said to be synchronized. Component-based applications represent groups of interconnected component plug-ins that interoperate to form a composite architecture. The interaction between client and server is predefined as an interaction protocol. We have identified two forms of component interaction:

Explicit Invocation (Direct Connection) – Attribute and operation connectors and adaptors are used to directly connect components required and provided services. Once connected, the required interfaces of one component may be satisfied by the provided interfaces of another.

Implicit Invocation (Indirect Connection) - Implicit invocation gives developers more flexibility in building systems out of existing components whose interfaces are incompatible. A communications server or whiteboard (Figure 3) is used where components post messages regarding changes in their internal state that may be of concern to others that are not directly connected to them. The whiteboard buffers all messages that have been posted as a sequence of events since the instantiation of an application so that components entering the application after a particular time are able to synchronize with the current application state.

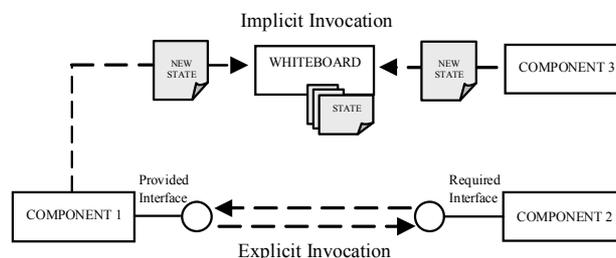


Fig. 3 Implicit and Explicit Invocation

IV. RE-CONFIGURABLE VOIP INTERACTIVE RESPONSE

The reconfigurable VoIP IVR interface solution allows flexible access to telecommunications and internet services. It provides the glue between enterprise business applications and multi-modal forms of business communication.

A reconfigurable plug-in host behaves as an IVR endpoint. As illustrated below (Figure 4), voice plug-in discovery, query, identification and invocation are used to situate, define, describe and utilize available services. Once a voice plug-in has been identified it is dynamically assigned to a keypad number. All of the available plug-ins and their associated keypad number are listed in the IVR main menu. A user enters a number using Dual Tone Multi Frequency (DTMF) tones to invoke a voice plug-in's services. Upon activation, and during execution, each voice plug-in governs user interaction and the provision of its services.

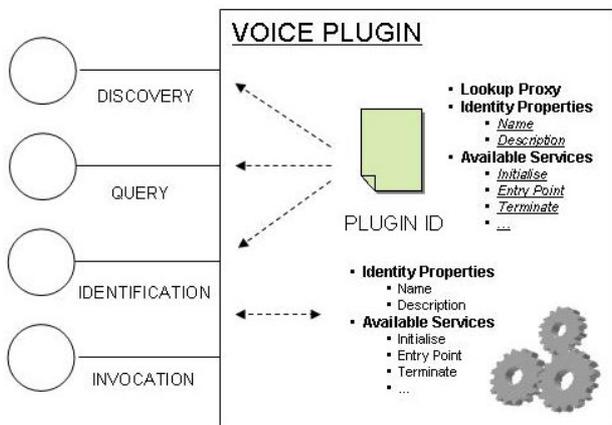


Fig. 4 Voice Plug-in

The solution is constructed from a number of lower level voice plug-ins that are fully composed component hierarchies (Figure 5). These plug-ins are constructed from multiple lower-level generic components which are reused in other module compositions. Customised voice plug-ins are constructed by mixing and matching hierarchically interconnected components.

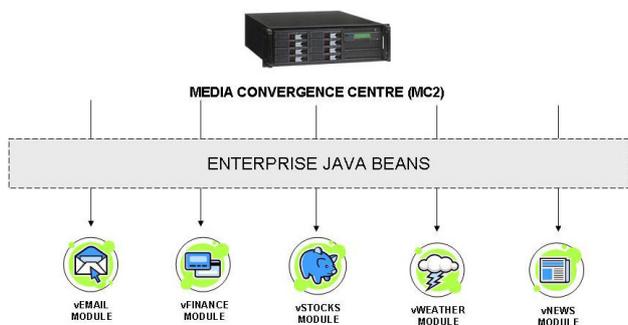


Fig. 5 VoIP Interactive Response Solution

The high-level diagram below illustrates the vEmail

plug-in. This plug-in was hierarchically constructed from multiple lower-level generic components which are reused in other module compositions.

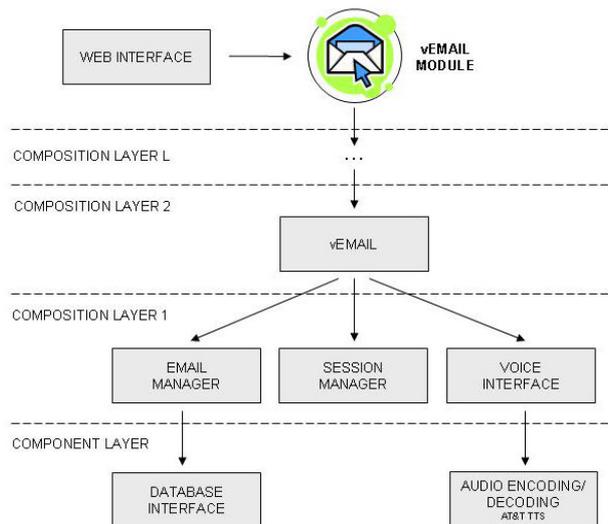


Fig. 6 vEmail Plug-in

V. IMPLEMENTATION

The software platform is an N-Tier distributed computing architecture that employs the following technologies;

- Asterisk – Asterisk [13] is an open source software PBX that can be programmed to create custom applications. Our system uses the java-based Application Gateway Interface (AGI) to trigger custom classes that handle incoming connections.
- Java - Java provides a platform independent environment that has wide support. Java [14] was used to interface the tools mentioned and implement the solutions in this system. Enterprise Java Beans are a suitable for the development of distributed and heterogeneous component systems. These technologies were selected because the Asterisk AGI has wide Java support and many existing libraries and frameworks that interface the soft-switch functions of Asterisk.
- AT&T TTS - The AT&T Text-To-Speech (TTS) [15] engine generates high quality synthesized voice from text. It is integrated through the Asterisk AGI interface. Using the AT&T TTS is it possible to adopt different dialogue files to simulate accents from different nationalities.
- Speex - This product [16] provides Automatic-Speech-Recognition (ASR) functions. Asterisk's extended scripting commands make use of Speex to take voice commands from a user.
- Hibernate – Hibernate [17] is a high performance object / relational mapping service for Java. It uses interfaces that have defined via mapping documents to convert between the Object Oriented (OO) to the Relational Database Management Systems

(RDBMS).

- MySQL - In order to persist hibernate objects we use the MySQL RDBMS [18].

A. Implemented Plug-ins

vEMail Plug-in - Emails can be accessed at any time and followed up instantly. The vEMail voice plug-in hosts a Post Office Protocol (POP) email service where high priority emails can be forwarded for voice access. The vEMail voice plug-in reads out each email using a clear voice. The user may interact with the vEMail voice plug-in by telephone key tones. vEMail allows the automatic browsing of emails without user intervention. It is possible to reply to emails with predefined email templates and forward messages to predefined contacts immediately. Users may manage their email messages by saving, moving and deleting selected items. It is also possible to customize the way that the system performs and to manage emails contacts and template messages via an easy to use web interface.

vStocks Plug-in - Live Australian Stock Exchange (ASX) values can be heard at the user's convenience. The vStocks service reads out detailed information on each user's individually predefined stocks. Stock list navigation is performed using telephone key tones. vStocks allows the browsing of stock data without user intervention. The vStocks service is able to announce each stock's trade date, time, change, previous close, day

high, day low, and volume. Users may customize the stock properties they wish to hear to suit their individual preferences.

vWeather Plug-in - Live Bureau of Meteorology (BOM) weather forecasts can be accessed at any time. The vWeather service reads out detailed weather information for a user's predefined city or town. Weather forecasts are read out for up to one week in advance. Forecast information, days high and days low are given for each day. Users may customize their city to suit their individual preferences and travel arrangements. vWeather allows the weekly weather forecast to be read out without user intervention.

vNews Plug-in - Live Rich Site Summary (RSS) News feeds can be heard at a preferred occasion based on a user's preference. The vNews service reads out each news item as requested by the users telephone key tone interaction or automatically.

Figure 7 illustrates a high level composition of the solution with sample voice plugin services (vEMail, vFinance, vStocks, vWeather, and vNews). The illustrated services interact with the user using voice and keypad prompts and access other components, services and repositories as required.

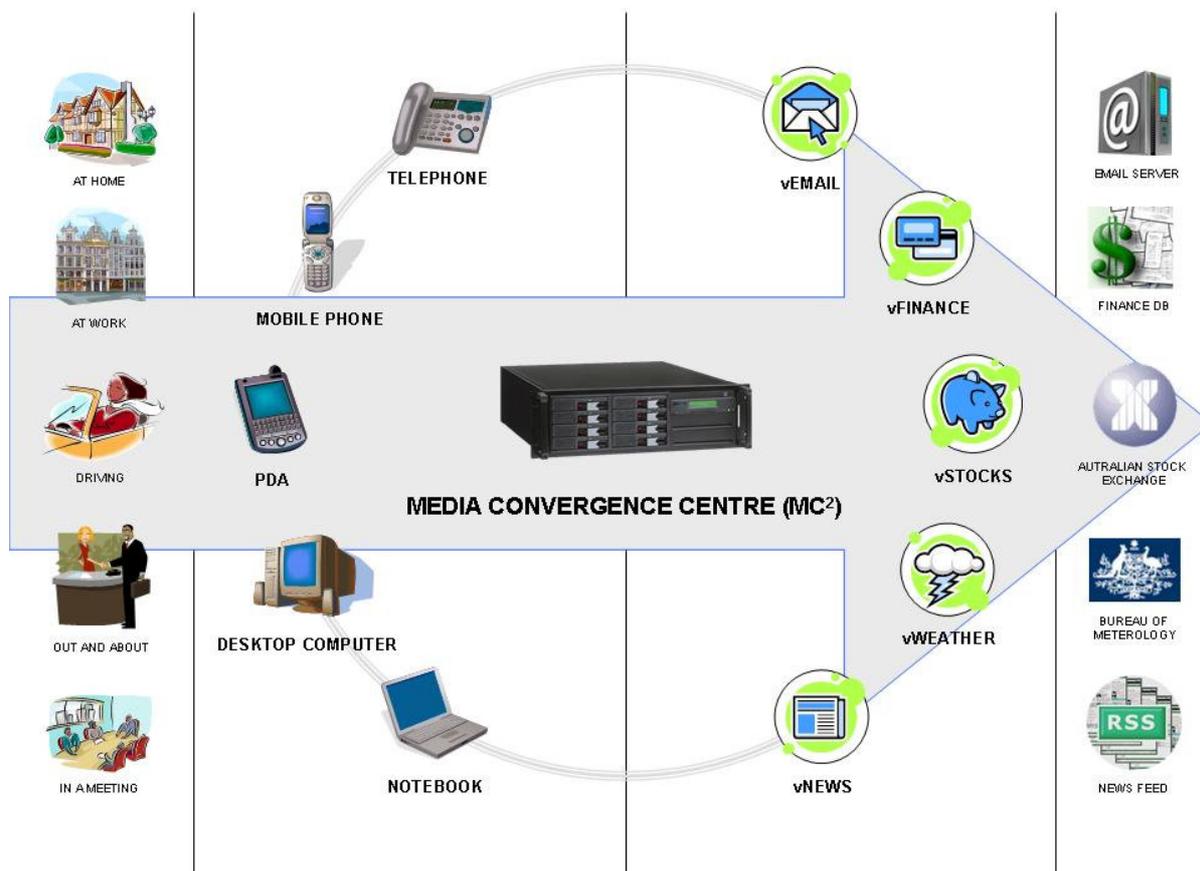


Fig. 7 Reconfigurable VoIP Interactive Response Solution

VI. CONCLUSION

Widespread use of mobile computing devices, telephones along with Internet services has resulted in a broad range of ways to access information and to communicate. However, true voice and data convergence has been isolated to certain environments. The versatile approach in this paper presents an Interactive Voice Response (IVR) driven solution that uses a reconfigurable plug and play framework as a basis for the simplified convergence of telecommunications services that are coupled with enterprise and internet data.

This combination reduces development and integration complexity, effort and cost while enhancing service effectiveness, flexibility and convenience for professionals on the move.

This evaluation of the tailoring platform and framework proved to be very successful. The reconfigurable solution demonstrated positive results and potential to tackle other problems and environments.

VII. REFERENCES

- [1] C. R. Strathmeyer, "An Introduction to Computer Telephony", IEEE Communications Magazine, 35(5), May 1996, pp. 106–11.
- [2] B. Benner, "Computer Telephony Integration (CTI) Industry", On-line at: <http://faculty.ed.umuc.edu/~meinkej/inss690/benner/CTIpaper.htm> (2006).
- [3] R. Darlington, "What is multimedia convergence and why is it so important", On-line at: <http://www.rogerdarlington.co.uk/Multimediaconvergence.html>(2007)
- [4] S. Phil, F. Cary, You Don't Know Jack About VoIP, Queue, 2004, 2(6), p. 30-38.
- [5] W. Stallings, Data and Computer Communications (Seventh Ed.), Pearson Educational International, 2004.
- [6] Deloitte, "Getting off the Ground: Why the move to VoIP is a decision for all CXOs", On-line at: <http://www.deloitte.com/dtt/research/0,1015,sid%3D2245&cid%3D64027,00.html> (2004).
- [7] M. Grant, "Voice Quality Monitoring for VoIP Networks", Calyptech Pty. Ltd., Melbourne, 2005.
- [8] M. C. Hui, H.S. Matthews, "Comparative analysis of traditional telephone and Voice-over-Internet Protocol (VoIP) systems", In Proceedings of the IEEE International Symposium on Electronics and the Environment, 2004
- [9] R. Schutte, "SOA is changing software", On-line at: <http://www4.gartner.com/resources/111900/111987/111987.pdf> (2003).
- [10] "Service-oriented architecture (SOA)", On-line at: http://www.service-architecture.com/webservices/articles/service-oriented_architecture_soa_definition.html (2003).
- [11] A. Talevski, E. Chang, T.S. Dillon, "Re-Configurable Web Services for Extended Logistics Enterprise", IEEE Transaction on Industrial Informatics , vol. 1, no. 2, May 2005, pp. 74-84.
- [12] D. Rine, N. Nada, K. Jaber, "Using Adapters to Reduce Interaction Complexity in Reusable Component-Based Software Development", In Proceedings of the Symposium on Software Reusability, Los Angeles, California, May 1999
- [13] "Asterisk Home", On-line at: <http://www.asterisk.org/> (2006)
- [14] "Java Sun Home", On-line at: <http://java.sun.com/> (2006)
- [15] "AT&T TTS", On-line at: <http://www.research.att.com/viewProject.cfm?prjID=315> (2006)
- [16] "Speex Home", On-line at: <http://www.speex.org/> (2006)
- [17] "Hibernate Home", On-line at: <http://www.hibernate.org/> (2006)
- [18] "MySQL Home", On-line at: <http://www.mysql.com/> (2006)