Abstract: Internet telephony now enables a wealth of new communication services. Traditional telephony services, such as call forwarding, call waiting or call transfer can be enhanced with directory services, Web, or e-mail. Programmability of Internet telephony services is a crucial issue for providing these services. There is a strong connection between programming Internet telephony services and the protocols that are used for their delivery. Among them, the signalling protocols play an important role. A number of protocols have been defined for Internet. In this paper, we concern ourselves only with SIP. On an example of the University Infoline service, we try to show a category of new service possibilities related to enhanced “click-to-call” services.

Key-Words: IP telephony, VoIP, SIP, programming services, click-to-call

1 Introduction
The importance of Internet within the world telecom industry has grown rapidly in several last years. There is evident renaissance of communications visible mainly in the Internet world and feasible thanks to a suite of new protocols, which are based on completely different principles as the centralized control systems used in the traditional circuit-switched systems. SIP – Session Initiation Protocol, a signalling protocol developed to set up, modify and tear down multimedia sessions over Internet, belong definitely among the key IP protocols. Thanks to it close relations to the HTTP 1.1 and SMTP protocols, the traditional telecom signalling and control models for telephony could be moved towards the Internet and web-based protocols.

SIP does not provide only seamless integration of telephony and conferencing with many other WWW and messaging applications, but benefits also from new forms of communications and new service features, like presence, instant messaging or mobility. One of the main SIP advantages is linked to the common Internet heritage with the HTTP and SMTP protocols. This common origin makes SIP an ideal bedfellow for different e-commerce applications. The University Infoline service, which is introduced in this paper, is a typical example of a voice-enhanced e-commerce service. The paper is organized as follows: After this introduction, we provide a brief overview of the Session Initiation Protocol. In the next part, we overview existing tools for programming SIP services and features. The main part of the paper addresses the University Infoline service, which has been developed and implemented in the Department of Information Services at the University of Zilina. On an example of this service, we show a category of new service possibilities, which are related to enhanced “click-to-call” services.

2 SIP (Session Initiation Protocol)
SIP is relatively a new signalling protocol developed to set up, modify and tear down multimedia sessions over the Internet. SIP was specified by the IETF MMUSIC WG as a proposed standard in 1999 [1]. In contrary with H.323, the ITU-T signalling standard for multimedia sessions over packet-based networks, SIP presents the IETF, or Internet approach to voice and video over IP. SIP provides signalling and control functionality for a large range of multimedia communications. The main functions are: location of parties, invitation to service sessions, and negotiation of session parameters. To accomplish this, SIP uses a small number of text-based messages, which are exchanged between the SIP peer entities, i.e., SIP User Agents in user terminals. SIP User Agents may operate as a client or server, depending on the role in any particular call. Messages can traverse the network entities like proxy servers or redirect servers, which are used for support, like address resolution, routing calls to other entities, etc.

SIP is a protocol that was designed to work hand in hand with other core Internet protocols like HTTP and SMTP. Many functions in a SIP-based network rely upon complementary protocols, including IP. Since SIP itself only defines the initiation of a session, all other parts of the session are covered by other protocols, which may come from other applications or functions not necessarily designed for real time multimedia over IP. The text-coding scheme of the SIP has been undertaken from the Web-browsing scheme.
The session itself is then described in two levels. The SIP itself contains the parties' addresses and protocol processing features, but the description of the media streams exchanged between the parties of a session is defined by another protocol. The Session Description Protocol (SDP) [2] is used for this. The SDP is not a protocol in its sense, but rather a structured, text-based media description format that can be carried in the SIP message body. The message body is transparent to SIP, thus, any other session description can be carried (e.g., a weblink). From this point of view, SIP sessions are not limited to telephony calls or conferences only, but they can include also information retrieval or broadcast sessions depending on the session description.

Apart from the baseline SIP RFC, the architecture has been completed by other IETF drafts, e.g., for call control of supplementary services, programming of services and features, management of personal preferences, and interworking issues. The IETF standard work is still in progress with continuous refinement of the RFCs and drafts by specialized IETF WGs.

3 Programming SIP services and features

For defining services, SIP took a different approach as compared to standard telephony. Implementation of SIP services can be done in general:

- Using SIP baseline protocol mechanisms
- Defining extensions (headers, methods-SIP call control framework) like in [3], [4], or by
- Dedicated programming tools (SIP – CPL [5], SIP- CGI [6], SIP- servlets [7], Java applets [8], JAIN APIs [9], Parlay [10]).

From another point of view [11], SIP services can be programmed either by the trusted (such as administrators), or by the untrusted (such as end users) users.

On the example of the University Infoline service, we show the implementation of SIP service by the use of Java programming language and the proprietary SIP API by the trusted users.

Independently on a programming tool, service creation has the meaning of the implementation of a service logic that guides behaviour of each of the system elements and controls a specific message flow or reacts on a message request. In the University Infoline service, a service logic is implemented on the SIP application server (which assembly the functions of the SIP registrar, SIP proxy, Third Party Call Control module) side. Thus, the service logic is a Java programme, which directs the application server’s actions based on inputs from the SIP messages exchanged between SIP elements, the service logic and localization database and the data got from the Web page.

4 University Infoline Service

4.1 Description

University Infoline service has been developed by the Department of Information Networks at the University of Zilina. The service belongs to the group of click2call services and enables web browsing simultaneously with the establishment of SIP (Session Initiation Protocol) calls with persons who are responsible for a content of the web page (University / Educational Center / Restaurant / Infoline etc.) The service is implemented as a part of distance learning environments of the departmental network, but in general, it may be implemented and used as any infoline service.

In the service example of the University Infoline, a student retrieves courses, which are listed in the web page of the University web space. Browsing a list of courses, a student wishes to get the detailed information on the course, provisions, etc. When he clicks on the hyperlink, a window with the course description appears together with the presence information of a tutor and a proposal to establish a call to a tutor of a course. If he accepts the offer, he becomes the SIP caller. Tutor becomes a callee.

In the frame of the proposed service, we handle different states of a SIP call, which may occur between a student and a tutor according to the tutors’ personal preferences. Under “tutors’ personal preferences”, we understand the possibility of a tutor to setup the time of day when he/she is available and when he/she wishes to accept calls initiated from the web on his personal SIP UA (User Agent). Thus, the following states of a tutor has to be distinguish and handle:

1. Tutor is online and available to accept calls initiated by a student from the web
2. Tutor is online, but busy with another call
3. Tutor is online, but not available to accept calls from the web
4. Tutor is offline

In the case 1., the SIP call between the tutor and the student is established.

In the case 2. - 4., some options are made available for a student how to proceed. The options include providing a tutor contact addresses, i.e. e-mail address and SIP address of a called tutor, which can be used by a student to contact a tutor manually later and/or the possibilities, which allow a student to leave his/her contact addresses to the infoline system (e-mail or SIP address of a student, to which the system will sent the e-mail or
establish a SIP call in the case a called tutor becomes online and available.

4.2 Provision, activation, deactivation, registration of the service

From a student’s point of view, the service is provided through the student’s web interface in the moment, when he decides to select some course from the list. The web page loaded from the university web server provides to him the presence information of a tutor (tutor is online or offline). The service is activated in a moment when a student made a SIP call to a tutor (he made a “click” action). In this moment, the Java applet is loaded from the university web server. The Java applet is an important component of the service and is used to initiate a SIP call itself through an opened socket connection with the SIP proxy. Java applet provides the student with the information about connection proceeding and allows him to leave his contact data in the case a tutor is not available. The service is deactivated, when the SIP call is finished successfully, or when a student contact data validity elapsed and the system deleted it. The service does not require any kind of student’s registration.

From a tutor point of view, the service is provided in a moment when a University Infoline administrator creates an account for him (SIP and Web account). Activation of the service is made in the moment when the tutor starts his SIP UA and registers in the system. The tutor can profit from all the service characteristics only if he sets his personal preferences through the tutor’s web interface. The service is deactivated when a tutor closes his SIP UA.

4.3 Operation

The service operates in the following way (Figure.1). In the first step, the Infoline administrator has to create the account for the tutor and, of course, the web page of a course provided by the tutor. The account is created through the web interface (1) and the tutor’s data is written and stored in a database through the JDBC interface (2). The administrator informs the tutor about his username and password, which is required for the user authentication to the system. They enable the tutor to access his service web part account to edit his personal information or time availability (3)(4). The same username and password are used by the tutor when he does SIP registration and authentication of his SIP UA in the SIP service server (5). SIP server collects and stores the localization data of tutors in the database (6). By now, the University Infoline has courses provided for students, and the system has tutors responsible for their content and ready for consultation with students. Courses are available via the public university web server, which is accessible to all students.

When a student selects a course from the list, the HTTP request is sent to the web server (7). The web server answers with the detailed course description together with the SIP presence information (SIP online, offline) of the tutor. The presence information is get from the database (8), built during the SIP registration phase. This

![Fig.1 University Infoline service operation](image-url)
presence information can be asynchronously refreshed through the web page on demand. When the student wishes to consult the content of the course through a SIP call, he inserts his current SIP address to an edit box on the page and clicks on the “Call” button. The HTTP request is sent then from the student’s web browser to the web server, which sends the web page back to the student. The web page is presented in a new web browser window. To this window, the Java applet is loaded (9). In a moment, the applet loading is completed, the applet establishes a communication socket with the SIP server (service logic) and sends the information about the caller and callee to it (10). The service logic accepts a request for a call, and now, it has to get the information on the tutor’s state. For that, the service logic inquires the database (11). In the case the tutor is online and available, the service logic has to find out, if the tutor is not busy by another call. Based on the conditions, the service logic performs different actions:

1. Tutor is online and available for calls initiated from the Web
2. Tutor is online and available for calls initiated from the Web (but he is busy with another call)
3. Tutor is online and not available
4. Tutor is off-line

In the case 1, the service logic asks the 3rd party call control to establish a call between the student and the tutor (12).

In the case 2, the service logic instructs the Java applet to allow the student to leave his contact data. If the student does not fill out any contact data (or he closes the Web page with the Java applet), the service is deactivated. If the student gives his contact data together with the related time periods, he has to send them back to the service logic in the SIP proxy server. (13). The service logic has to record the data to the database using a standard JDBC interface (14). Since the tutor can be contacted by more than one student, their contact data will be kept and handled according to the FIFO mechanism. The service logic starts to monitor the tutor’s status based on the time periods set by the student. If the tutor becomes free (during this period), the service logic informs the student according to the preferred way of contact. If the call establishment was set as the preferred way, the service logic establishes a call by the means of the 3rd party call control (15). If there was the e-mail notification set by the student as the preferred way, the e-mail will be sent to the student when the tutor becomes available (16). If there were both the means set as the preferred ones, and the tutor becomes available within the valid period for the both, the call is established and the e-mail notification is sent to the student as well.

In the case 2 and 3, the procedure is the same as in the Figure 1, up to the step 11. In this moment, the service logic finds out that the tutor is off-line or not available, thus, it proceeds in the same way as it does in the Figure 1, starting from the step 13.

4.4 Components

The following components will be involved in the service:

- **SIP client**: generic SIP User Agent.
- **SIP server**: StarSIP Application server is used. The server involves SIP Registrar, Proxy, Third Party Call Control (TPCC) module. It provides API for programming of services, which utilize the SIP server functions.
- **Database**: SQL database, which supports an access through the JDBC driver. The database is supposed to be shared between the SIP server and the Web server. We are using MySQL database.
- **Web browser**: MS IE v 4.0 or higher will be used. It provides Web interfaces to a student, tutor and administrator. Web browser has to support for running of the Java applets.
- **Web server**: server, which supports JSP technology for dynamic generating of the HTML pages and providing for the database access through the JDBC driver. We are using Tomcat web server.

For the maximal portability of the service all service components and service logic were written in Java and JavaServer Pages. All the used servers (Tomcat, StarSIP) were chosen to be platform independent. The only requirement is the existence of the Java Virtual Machine (JVM) for the platform.

4 Conclusion

Efficient programming of new communication services is a key issue for Internet telephony. With SIP, services can be created that combine elements from telephony and other web applications such as email, messaging, the Internet and video streaming. Using SIP, services like “click-to-call” become possible, in which, the user profiles can be managed through a web interface. On an example of the University Infoline service, we present a category of new enhanced “click-to-call” services, which allow a user browsing through the web pages not only to establish a SIP call, but which enable, via the service logic, to handle different states of a call based on the preferences of calling and called parties. The service is simply integrable to any type of voice-enhanced e-commerce services.
References: