SIP-based Location Aware Services

N.Priggouris[*], S.Paskalis[*], D.Vali[**], S.Hadjiefthymiades[*], L.Merakos[*]
nprigour@noc.uoa.gr, paskalis@di.uoa.gr, dvali@oteresearch.gr, shadj@di.uoa.gr, merakos@di.uoa.gr

[*] Communication Networks Laboratory
University of Athens, Dept. of Informatics and Telecommunications
TYPA Bldg., Panepistimioupolis, Athens 15784, Greece

[**] Mobile & Radio Communications
OTE Research
Pelika & Psaron, Psalidi Amaroussiou, Athens 15122, Greece

Abstract

In this paper, we investigate the prospects of using Session Initiation Protocol (SIP) in context—aware services. We focus on the location-based paradigm in which the users' location is the contextual parameter used for service adaptation and delivery. We present the general steps required to achieve this and elaborate on an architecture that uses a SIP-enabled Value Added Service Provider as the main component responsible for the provision and delivery of the service. The pilot implementation carried out exploits the general ability of SIP to embed user specific information in the message body of the exchanged messages.

1 Introduction

Session Initiation Protocol (SIP) [RFC2543] is a simple, Internet–based session control protocol that was introduced mainly in order to establish and terminate calls over the Internet (VoIP). However, its simplicity and versatility has made it a prime candidate in the development of new services (e.g. Instant Messaging [RFC2778]), as well as a key technology enabler for supporting new signalling capabilities within next generation services. Many competitive Telecom Operators are already using SIP for initial service provision and SIP based solutions are emerging rapidly all over the globe.

On the other hand, context-aware computing is a mobile computing paradigm in which applications can discover and take full advantage of contextual information (such as user geographical location, time of day, type of terminal/device, user activity, etc.). The applications are then delivered to the user customized to its current context. In supporting such types of transactions, SIP appears as an ideal solution. It can be used initially for the context discovery (by properly using certain header fields or information carried in the message body of a SIP request) and later on for the connection establishment and termination of the service.

In the present article, we explore the capabilities of using SIP in support of context–aware applications. We focus on an architecture where SIP is used in the control plane to signal the current context of the user to the information provider, and the delivered content specific information is based on the user's current location.

The rest of the paper is organised as follows. In Section 2 we survey related work. In Section 3 we describe the architecture of the proposed system. Starting with a high level description of the system, we continue with a more detailed analysis of each individual component and conclude by presenting the complete interaction and message exchange between them. We present the details of a pilot system implementation in Section 4. Finally, Section 5 presents some conclusions and possible enhancements of our work.

2 Related Work

Context-aware computing is a mobile computing paradigm in which applications can discover and take full advantage of contextual information (such as user location, time of day, type of terminal/device, user activity, etc). Context-awareness is a prerequisite for pervasive computing systems as pointed out in [Sat01]. Obtaining and using the user's contextual information helps in minimising the disturbance caused by a computer assistant.

An important subset of context-aware services for which much attention has been drawn, is the location-based paradigm. Most of the context-aware prototype systems use location as the contextual parameter on which they adjust so as to deliver the requested service. Location-based services are expected to comprise a large fraction of the new services offered by next generation telecommunication operators (3G) [And01].

Location dependent technology was pioneered and prototyped in research labs. Since the experiments were carried out in indoor environments, RF technology was used for the location information detection. Xerox' ParcTab [Wan95] and the ORL Active Badge System [Wan92] used office-wide wireless infrared networks to provide the necessary location information.

The location discovery of a user on a global scale imposes different requirements, though. One of the most well known solutions is the Global Positioning System (GPS) [Eng99]. Although its accuracy in outdoor conditions is very good, it is ineffective indoors. Moreover the cost of the equipment would render its deployment unaffordable in low-cost embedded devices.

The most attractive alternative seems to be the utilisation of existing information available on the Internet. [Pad01] presents an overview of the existing techniques, which can be coarsely distributed in the following categories:

- incorporating location information in DNS records [RFC1876]
- using the whois database [RFC954]
- using the traceroute tool [traceroute]
- extensive IP address to location mapping

The DNS location information proposal has not caught up to the critical mass necessary for effective global coverage. The whois database query is a very popular method, although it exhibits some drawbacks. The most crucial concerns allocations of large blocks of IP addresses pointing to the same geographical point and the possibility of stale entries. The traceroute usage could be more efficient, although it shares its own set of shortcomings, since a router name may not contain location information or be formatted in a non-standard manner. Extensive IP address to location mapping requires co-operation between the majority of ISPs.

3 SIP-based context-aware service

3.1 Service Description

In our implementation, the context-aware information delivery is a subscription-based service. It adopts the push-based paradigm. In other words, a server initiates the context aware service provision and the relevant information is delivered to the user. The user must have subscribed to the information delivery service in order to receive the generated announcements. The information delivered to the end user is adapted to its current geographical location. The whole process can be decomposed in the following steps:

1. *Service Subscription*: Provide the user with the ability to subscribe, unsubscribe or modify his subscription to the service.
2. *Context Retrieval*: Obtain the IP address of the user through SIP signalling.
3. *Context Resolution*: Find the current location of the user on the base of its current IP address.
4. *Information Retrieval/Customisation*: Obtain the proper information according to the user location.
5. *Information Delivery*: Send data to the user through a network connection.

3.2 Architecture Overview

There are various approaches suitable for implementing a service that delivers context aware information. Based on the functionality and entities provided by SIP, we can use e.g. a SIP proxy server enhanced with the capability of initiating requests to user agents (UAs) and retrieving information from proper database structures. Another solution may involve the development of a special user agent capable of obtaining and analysing the user's current context – through a sequence of SIP messages – and then deliver the requested information. We choose the second approach since it is more straightforward, transparent and does not require modification of existing SIP elements. For the same reasons, we use the widely deployed types of SIP messages (INVITE, OK etc.) instead of introducing new types specialized to our case. The contextual information is embedded in the SIP message body using Session Description Protocol (SDP) [RFC2327].

The proposed architecture for the service is depicted in Figure 1. We assume the presence of three main entities:

- a SIP-enabled context—sensitive Value Added Service Provider (VASP): VASP is the main component within the suggested architecture, responsible for the provision and delivery of the service.
- a SIP Service Provider: The SIP Service Provider (SISP) incorporates a SIP Registrar server along with a SIP Redirect server and a SIP Proxy server. The Registrar server stores internally information about the user's location and possible contact address. All calls from/to the SISP are handled by the Proxy Server.

– a SIP-enabled User terminal: The SIP—enabled User terminal corresponds to a common SIP UA enhanced with the ability to communicate via UDP or TCP connections.

Moreover, in Figure 1 the following entities are depicted:
– Content DB: Stores information that will be used during content adaptation.
– User DB: A database that stores the profiles of user after their subscription to the service. This entity can be co-located with VASP.
– Resolve Context Server: This entity resolves the current context of the user. For example, in our proposed service it finds the current location of the user based on the provided IP address.

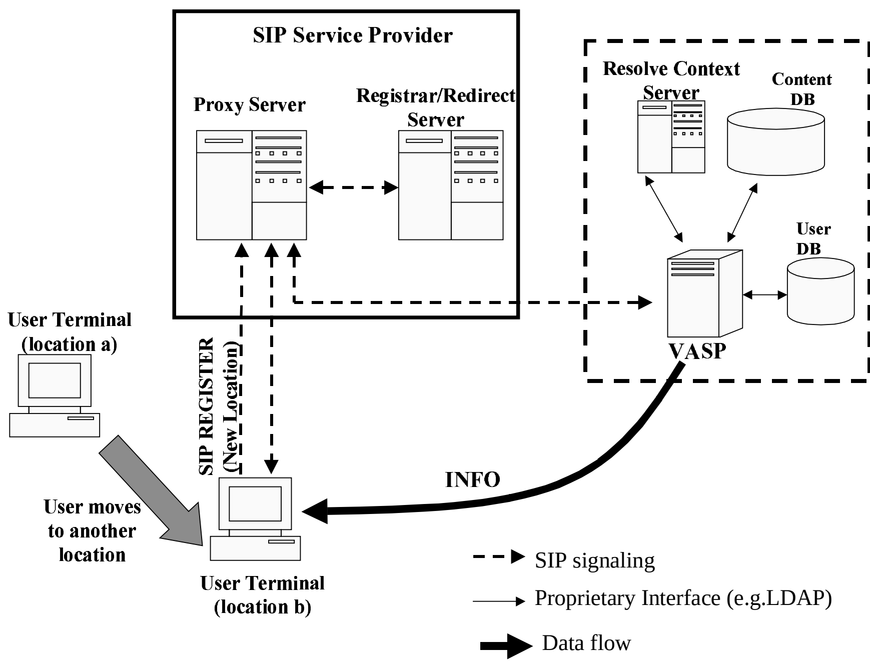

Figure 1: System Architecture

3.3 System Operation

3.3.1 VASP functionality

VASP incorporates the functionality of a SIP user agent along with the ability to communicate with a content Database, through a proprietary interface, in order to retrieve the proper content that corresponds to the user's current context (geographical location in our case). It is also possible that VASP, after the exchange of the required SIP signalling, interacts with another server (e.g. a geo—location server) and resolves the exact context of a specific user that will be used later on for the generation of the content. Of course, the procedures of context resolving and content adaptation can be handled by internal entities of the VASP but such an approach is less modular and flexible.

Moreover, VASP should provide a proper interface to the end user for subscribing/unsubscribing to a desired service. This can be achieved by the completion of a proper web form by the potential end user. The results of these actions will be securely posted to VASP, which will add/remove the user to/from a User Profile Database.

3.3.2 VASP Internal Structure

Figure 2 presents the internal structure of the VASP. VASP consists of the following entities:

- Content Generator: The Content Generator is the main component where the service logic resides. It is the co-ordinating component for service delivery, capable of communicating with all other VASP and non-VASP entities. The Content Generator retrieves the customised content based on the users' profile and current context and initiates the "push" of the customised content to the user.
- Subscription Manager: The Subscription Manager receives the user requests posted to VASP and stores information about each user in the User Profile DB. It also interacts with the Time Scheduler component, which is responsible for service triggering.
- Time Scheduler: Schedules Triggering Events at specific times.
- Transcoder: Performs media transcoding when necessary.
- UA+ SIP stack: A common User Agent that interacts with the Content Generator. It is responsible for the complete handling of SIP signalling from/to VASP.

In short, the interaction between the aforementioned entities is described in the following:

According to subscription details, the Scheduler will generate events for the Content Generator. These events are associated with individual users' profiles. The reception of a trigger event forces the Content Generator to initiate a SIP INVITE request towards the specified user with proper connection information in the SDP message body. The INVITE message propagates to the end user, which responds with a 200 OK containing connection parameters as well as contextual information pertaining to the user in the SDP part. Such propagation is handled automatically by SIP, hence the advantage of employing SIP for the initiation of a push-based service. The SDP conveyed information is passed back to the Content Generator, which in turn contacts the User Profile DB, Resolve Content Server and Content DB and obtains the proper content according to the user's current context. Then, the obtained data can be delivered through an application protocol (e.g. in XML format) over a typical network connection (TCP or UDP). The presence of the transcoder unit is optional. Its role is bounded in transforming the data content based on the capabilities of the end user specified in the SDP part of the final OK message (e.g. if the UA can, at the considered time accept only voice data a textual content must change accordingly).

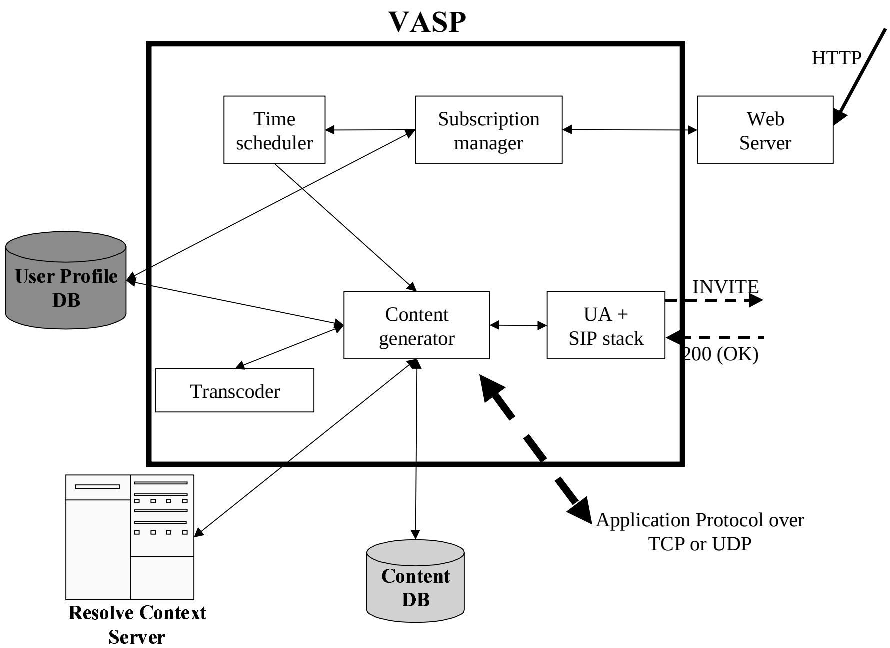

Figure 2: Internal Structure of VASP

3.3.3 VASP and system operation

VASP operation, as well as system operation in general, can be divided in two distinct and somewhat independent phases called *subscription* and *invocation.* The subscription of users to specific services is handled exclusively by the Subscription Manager and is achieved through a web browser. The potential service user completes a properly structured form. The subscription record should at least contain the fields: username, user info, service type and service timing. Table 1 gives a more detailed description of each field. All the information provided during subscription is stored in the User Profile DB. The username should identify uniquely the user within the Profile DB. If subscription is successful, a password is return to the end user that might be used for authentication at a later stage, during service modification or cancellation. Moreover, the Time Scheduler is properly informed and schedules a “trigger” event for the specified user based on the “service timing” field. The complete component interaction during the subscription phase is demonstrated in Figure 3.

Table 1

Fields	**Description**
Username	A username capable of uniquely identifying the user. This should be the same name used during SIP Registration
User Info	Specific Information pertaining to a user (e.g. email address, terminal type, default contact address etc.)
Service Type	Type of requested information (only meteorological info

	defined for pilot implementation)
Service Timing	Information about how often the service will be delivered to the user

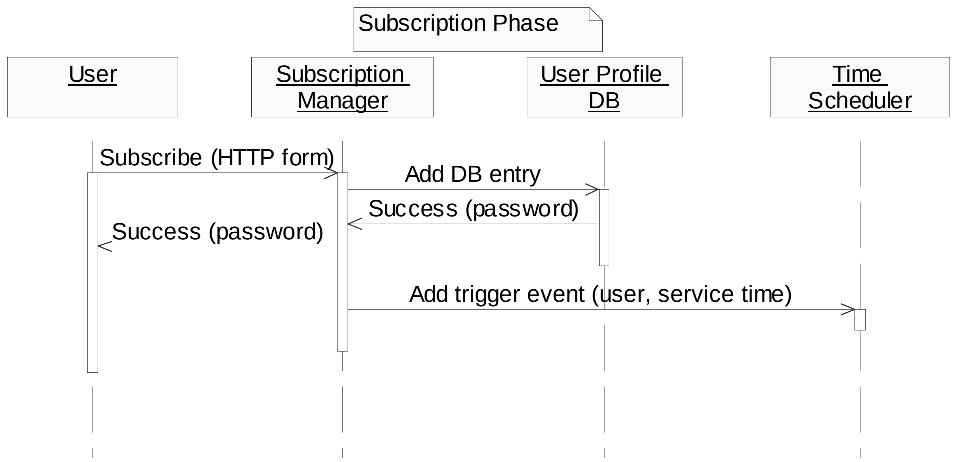

Figure 3: User Subscription Phase

The invocation phase is initiated upon the arrival of a "trigger" event to the Content Generator. This VASP component is responsible for the sequence and synchronization of all the messages that follow. Figure 4 shows the series of messages during the invocation phase. The Content Generator stimulates the VASP UA, which constructs and sends a SIP INVITE message to the user. The message body contains various connection parameters as well as the name of the VASP so that the end user can discriminate the originator. Furthermore, it contains the type of requested contextual info. This INVITE message propagates through a number of proxies and/or redirect—registrar SIP servers to the proper destination corresponding to the end user. A prerequisite for the correct operation of the service is that the end user re-registers with the SIP provider, by sending a SIP REGISTER message, every time it changes location (IP address). The end user provides the requested contextual information in the SDP part of the final response OK message. For example, in our proposed service of location-based information delivery the end user embeds its current IP address in the reply to VASP. After that, the Content Generator entity, assumes all responsibility for interacting with the others components in order to obtain and deliver the desired information. The whole transaction is terminated using SIP signalling. Upon completion of data delivery, VASP sends a SIP BYE message to the corresponding user. This action marks the completion of the service.

4 Pilot Implementation

A pilot implementation of the described context-aware service delivery was carried out using the VOCAL software from Vovida [vovida]. The software provides the whole functionality of a SIP Service Provider (Registrar/Redirect, Proxy) as well as a software UA capable of initiating calls through a proxy server. The VASP implementation was based on the framework depicted in Figure 2. However, the transcoder unit was omitted since the main focus of the pilot implementation was to demonstrate the feasibility of the proposed architecture for context-aware information delivery.

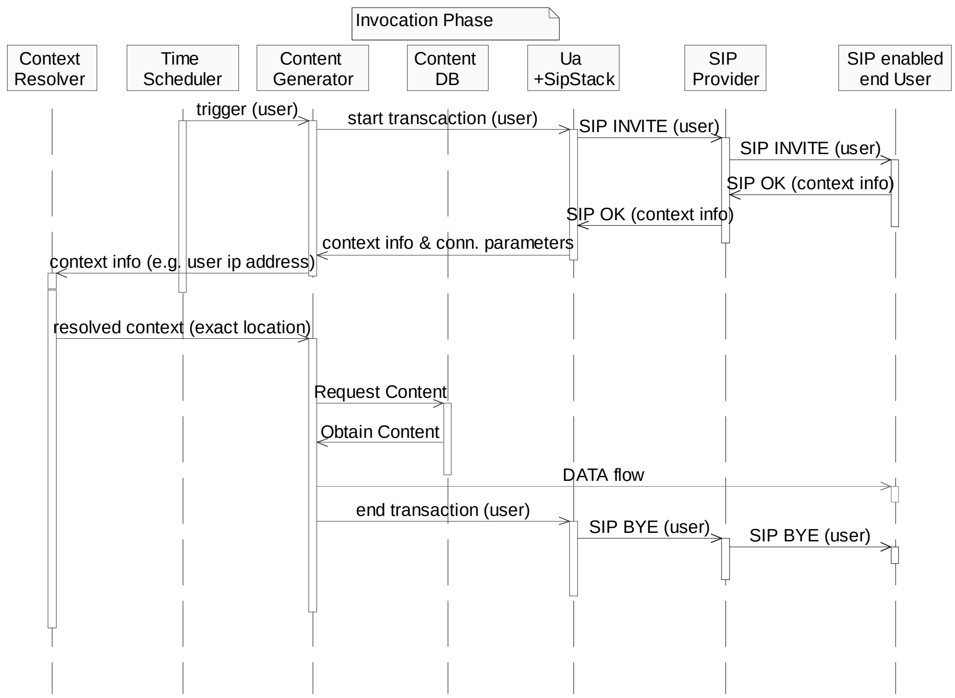

Figure 4: Service Invocation Phase

The Subscription Manager was implemented using Java Servlets in the Jakarta-Tomcat Application Server. The Time Scheduler was implemented as a crontab record [cron] executing a predefined command at a specified time. It takes as an argument the user's global identification name as well as the service invocation time. The user has registered to the SIP provider using this name. This name should also be provided to the user name field during the service subscription phase.

The Content Generator was integrated with the VOCAL UA component. More precisely, the software UA provided by Vovida was modified to take as a command line argument a user name. Upon starting up it sends a proper INVITE message to the user specified in the command line. Subsequently, it enters a stand by mode waiting for a response from the other end (user). After receiving the OK response message, it extracts the user's IP address from the SDP field of the message body. Using this information and through a proper Java API, it contacts NetGeo database located in http://www.caida.org/tools/utilities/netgeo/ [Moo00]. NetGeo is a database and collection of Perl scripts used to map IP addresses, domain names and AS (Autonomous Systems) numbers to geographical locations. It is created by queries to the whois databases. The information structure returned contains the fields: {longitude, latitude, city, state, name, country}. There are also some other fields of minor importance to our service. To find the latitude/longitude values for a domain name, NetGeo first searches for a record containing the target name in its own database. The NetGeo database caches the location information parsed from the results of previous *whois* lookups, to minimise the load on *whois* servers. If a record for the target domain name is found in the database, NetGeo returns the requested information, e.g., latitude and longitude. If no matching record is found in the NetGeo database, NetGeo performs one or more *whois* lookups using the InterNIC and/or RIPE (http://www.ripe.net) *whois* servers, until a *whois* record for the target domain name is found. After obtaining a record from a *whois* server, the NetGeo Perl scripts parse the *whois* record and attempts to extract the city, state (or province, district, etc.), and country from the text of the *whois* record.

After the exact user location is obtained, the Content Generator queries the Content DB. For our pilot implementation this database contains meteorological information. According to the country field value obtained by the NetGeo server, the meteorological information corresponding to that particular country is returned. The final results of the query are sent to the end user through a TCP connection.

The pilot implementation of the considered system was demonstrated in a workshop that took place in Berlin. The demonstrated scenario was the following: a SIP user registers, at first, from Athens and receives Athens-specific meteorological info and the same user registers afterwards from Berlin and receives Berlin-specific meteorological info. The demo illustrated the successful content (meteorological info) adaptation to the user's context (geographical location).

5 Conclusions & Future Work

We have designed and implemented a prototype location based service using SIP as the signalling protocol. Given the momentum that both SIP and location based services play major role in next generation telecommunication networks, their combination in service delivery seems a natural choice. However, our proposal introduces some differentiation from previous approaches. We adopt a push—based model for the service delivery, in which the user subscribes once and is paged periodically with the requested information. The context information is passed using SIP during the initial signalling phase. This accelerates the whole process of service delivery.

Through our service design and implementation, the feasibility of a SIP-based location aware service was proven, the simplicity and power of the SIP protocol was experienced and the advantages of SIP-based service development were verified in practice.

As future work, we will investigate the use of newly proposed SIP methods (SUBSCRIBE, MESSAGE) defined specifically for presence and instant messaging in our service. Another enhancement could be the delivery of voice and/or video information to the user (instead of solely textual information), with the aid of a transcoder, providing that the system can retrieve the SIP device's capabilities. Moreover, one could extend the context scope of our service, so as to deliver customised context based not only on user's geographical location but also on time of day, date, sentimental situation, etc.

6 Acknowledgements

The work presented in this paper was carried out in the framework of the EURESCOM Project P1111 "Next-Generation Open Service Solutions over IP (N-GOSSIP)" (http://www.eurescom.de/public/projects/p1100-series/p1111). Our pilot implementation was demonstrated in the Eurescom P1111 project Workshop that took place in Berlin.

7 References

[RFC2543] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg, "SIP: Session Initiation Protocol", RFC 2543, March 1999.

[RFC2778] M. Daw, J. Rosenberg, H. Sugano, " A Model for Presence and Instant Messaging", RFC 2778, February 2000.

[Sat01] M. Satyanarayanan, "Pervasive Computing: Vision and Challenges", in IEEE Personal Communications, August 2001.

[And01] C. Andersson, "GPRS and 3G Wireless Applications", Wiley, 2001.

[Wan95] R. Want, B. Schilit, N. Adams, R. Gold, K. Petersen, D. Goldberg, J. Ellis, and M. Weiser, "An Overview of the PARCTAB Ubiquitous Computing Experiment". IEEE Personal Communications, December 1995.

[Wan92] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The Active Badge Location System". ACM Transactions on Information Systems, 10(1), pp. 91-102, January 1992.

[Eng99] P. Enge and P. Misra, editors, "The Global Positioning System", Proceedings of the IEEE, January 1999.

[Pad01] V. Padmanabhan, and L. Subramanian, "An Investigation of Geographic Mapping Techniques for Internet Hosts", in Proceedings of SIGCOMM '01, San Diego, California, August 2001.

[RFC1876] C. Davis, P. Vixie, T. Goodwin, and I. Dickinson, "A Means for Expressing Location Information in the Domain Name System", RFC 1876, January 1996.

[RFC954] K. Harrenstien, M. Stahl, and E. Feinler, "NICNAME/WHOIS", RFC 954, October 1985.

[traceroute] V. Jacobson, ftp://ftp.ee.lbl.gov/traceroute.tar.gz

[RFC2327] M. Handley, V. Jacobson, "SDP: Session Description Protocol", RFC 2327, April 1998.

[vovida] http://www.vovida.org/

[cron] P. Vixie, ftp://ftp.vix.com/pub/vixie/vixie-cron-3.0.1.tar.gz

[Moo00] D. Moore, R. Periakaruppan, and J. Donohoe, "Where in the World is netgeo.caida.org?", in Proceedings of INET 2000, Yokohama Japan, July 2000.