From Instant Messaging to Cloud Computing, an XMPP review

Adrian Hornsby
Nokia Research Center
Tampere, Finland
name.surname@nokia.com

Rod Walsh
Nokia Research Center
Tampere, Finland
name.surname@nokia.com

Abstract—**The eXtensible Messaging and Presence Protocol (XMPP) is a versatile protocol that demonstrates many advantages over traditional communication protocols. XMPP's origin as an instant messaging (IM) protocol ensures that it inherits a tight relation with social communities and its flexibility makes it a very good candidate for converging communications between different application domains, such as social networks, consumer electronics and cloud computing. This paper reviews the XMPP protocol, its standardization community extensions and its usage in published literature. An application domain classification, or taxonomy, is derived and used to relate complementary and competing instances of XMPP deployment.**

I. Introduction

Jabber, also known as the eXtensible Messaging and Presence Protocol (XMPP) was initially developed in 1999 as an open, presence aware Instant Messaging (IM) protocol. Over the years, and to satisfy the ever growing human desire for interaction, developers have leveraged its flexibility to adapt it to a broad range of different needs, from a relatively simple application such as instant messaging to larger systems such as cloud computing. In this paper, we first review the XMPP protocol, the extensions provided by the standardization community and its usage in the published literature from which we derived a classification of application domain. We then discuss the emerging possibilities brought by the flexibility and adaptability of the XMPP protocol.

II. The XMPP Protocol

The XMPP protocol [1] [2] [3] is the formalization, by the Internet Engineering Task Force (IETF), of the instant messaging protocol developed by the Jabber community in 1999. It uses an Extensible Markup Language (XML) streaming technology to exchange XML elements (stanza) between any two entities over a network. An XML stanza is a semantic unit of structured data that exists at the direct child level of the root <stream/> element which represents the start of an XML stream. XMPP defines three core stanza – presence, message and iq (info/query):

The <presence/> stanza is a broadcast notification mechanism through which entities can receive network availability information from an entity they have subscribed to. This is mostly used by instant messaging applications to notify whenever an entity is online, away or offline.

The <message/> stanza is a mechanism whereby one entity can send asynchronous information to another entity. The message stanza is optimized for real-time delivery but it also supports store-and-forward asynchronous delivery.

Finally, the <iq/> stanza is a mechanism enabling entities to make requests of, and receive responses from, each other.

XMPP entities are identified by a unique Jabber ID (JID) in the style "node@domain/resource" where the "node" can be a username, a chat room, a bot, etc. The "domain" is a server or component and the "resource" (not mandatory) can be a device identifier (phone, laptop), a nickname in a chat room, etc.

The domain is the only required identifier element of a JID and usually represents the server to which entities connect for XMPP messaging capabilities. The domain identifier for every server has to be a fully qualified internationalized domain name in order to enable XMPP to build up a federated, decentralized architecture as shown in Figure 1, simplifying communications across administrative domains.

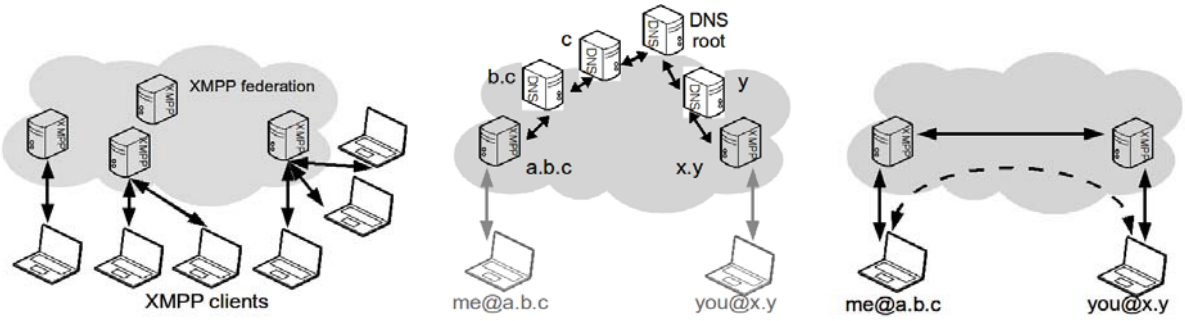

Figure 1: XMPP Federation

978-1-4244-6673-3/10/$26.00 ©2010 IEEE

In a typical XMPP session, client@domain negotiates a stream with server@domain, retrieves his contact list (roster in XMPP terminology), notifies and receives presence information to/from all his online contacts and then can start messaging with one or several of his contacts, all in a secured and authenticated manner. Indeed, the core protocol specifies that XMPP sessions provide authentication at the XML streaming layer using the IETF standard for Simple Authentication and Security Layer (SASL) [6], and provide channel encryption with the Transport Layer Security (TLS) [7].

A collection of XMPP extension protocols (XEPs) is maintained by the XMPP Standards Foundation [8]. These enlarge the capabilities of XMPP by offering, for example, asynchronous publish-subscribe (PubSub) and service discovery methods.

Social and community oriented extensions include Multi-User Chat (MUC), Personal Eventing via Pubsub (PEP), User Geolocation, Mood, Activity and Tune (currently being listened to). Entity Capabilities discovery and Ad-Hoc Commands are available for advertising and executing application-specific commands on a remote entity. In response to demand for logged status updating and blogging, the Microblogging over XMPP capability is now also available as a XEP.

The serverless messaging extension defines a peculiar communication schemes for local and ad-hoc wide-area networks using the principles of zero-configuration, allowing any entities to negotiate a serverless connection enabling the exchange of XMPP message and iq stanzas.

Even though most of the protocol and the available extensions are primarily based on textual and XML-based communication, peer-to-peer multimedia sessions control are now being fully supported by the Jingle extensions [4]. It is used for multimedia communications such as in Voice over Internet Protocol or videoconferencing, using XML data exchange for in-band signaling and the Real-Time Transport Protocol (RTP) for transporting the audio-visual data.

XMPP, with its extensions, form a push mechanisms, whereby entities can receive message or information update when ever they occur on the server avoiding the effect of polling for updates similarly to HTTP. Even though technologies such as AJAX, pubsubhubbub, Comet and HTML5 have mechanisms to overcome the effect of polling (as show in Figure 3), either by dynamically reloading the document, by "pinging" the client when new update is available, by asking for incremental updates, or by keeping long-lived HTTP connections, they are mainly emulations of push mechanisms using poll techniques. Moreover they require a notable additional implementation effort [14].

An XMPP bindings to HTTP (BOSH) has also been defined to be used in restricted environment, where firewalls, gateways or proxies inhibit messages of protocols other than HTTP. These bindings adhere to the push model, while being fully HTTP/1.0 compliant, by emulating a long-lived, bidirectional TCP connection between two entities using Comet. Therefore, BOSH provide HTTP connectivity with the benefit of XMPP - persistent connections, messaging capability, and authentication.

Published Literature in each Application Domain	Instant messaging	Interactive Social Media	Cooperative Work Flow	Internet of Things	Multi-Agent System	Cloud Computing
1. SMS extended with XMPP [12]	●	○				
2. Instant Messaging System for Mobile Networks	●					
3. IM and presence for college campuses [13]	●					
4. Jingle: Jabber does multimedia [17]	●					
5. XMPP and iDTV to make television a social medium [18]	○	●				
6. Telebuddies on the move [20]	○	●				
7. Web-based multi-player games [19]		●				
8. TuVista: Multimedia for Mobile Sports Fans [21]	○	●				
9. Distributed Collaborative Application [22]		○	●			
10. Human Collaboration Tools [23]	○		●			
11.Google Wave-protocol [11]			●			
12. Distributed measurement for hydrocarbon pollution [24]				●		
13. XMPP Event notification for large scale health care [25]				●		
14. Sensor Andrew [26]				●		
15. Federal Pervasive Sensor Networks [28]				●		
16. Support of Asynchrony in Sensor Web [29]				●	○	
17. XMPP-based WSN & the extended home [30]				●		
18. µXMPP: implementation for Contiki [31]				●		
19. Substrate for P2P Grid Computing [32]					●	
20. A Jabber based multi-agent system platform [33]					●	
21. Using XMPP for ad-hoc grid computing [34]					●	
22. Kestrel: XMPP framework for many task computing [35]					●	
23. JADE [36]					●	
24. Presence within SOA environment [39]					●	
25. XMPP for Cloud Computing in bioinformatics [37]						●
26. Blueprint for the Intercloud [38]						●

Table 1. Literature review table. (Black dots are primary classifications, grey dots are for significant secondary classes)

Table 2 (later) classifies the most interesting current XEPs according to their applicability to the application domains that are introduced in the next section.

III. Application Domains

In this section, we review use of XMPP in published literature (Table 1 provides an overview of this). We construct six application domain classes which encapsulate the state-of the-art and facilitate comparison: (1) Instant Messaging, (2) Interactive Social Media, (3) Cooperative Work Flow, (4) Internet of Things, (5) Multi-Agent System and (6) Cloud Computing.

1) Instant Messaging (IM)

IM is a form of real-time communication, often called chat, between two or more participants using typed text, wrapped and transferred via a network. Several IM applications are available such as MSN, Yahoo Messenger, ICQ, AIM and Skype, but those systems are proprietary and centralized.

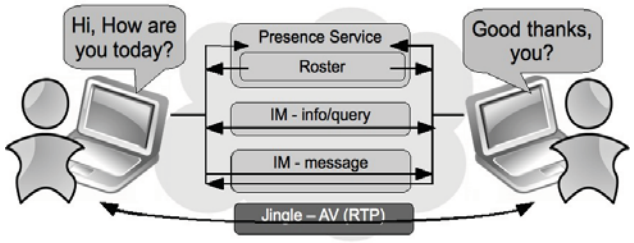

Figure 2. Instant Messaging with XMPP

XMPP, on the other hand, is an open, decentralized protocol designed for efficient real-time communication between peers. This concept is illustrated by Figure 2. It makes use of presence mechanisms to be aware of the availability of particular correspondents before exchanging messages. IM basic extensions provide support for file transfer, delayed delivery and mechanisms for capturing extended presence data such as geolocation, mood, activity and tune.

XMPP evolved from IM and presence, later adding support for multimedia session negotiation using the Jingle extension [17]. From this, XMPP has been deployed by services such as GoogleTalk and the Facebook Chat Service. XMPP is often compared to the session initiation protocol (SIP) for IM applications although, unlike XMPP, SIP began as a session setup and session management protocol, later adding extensions such as SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions) to support IM and presence. Comparing XMPP and SIP in more detail is out of the scope of this paper as various comparison are already available in the published literature [13] [18] [25]. As both these technologies are widely deployed, mappings between the two technologies is being defined in a series of SIP-XMPP interworking documents [41].

XMPP capabilities have been used in many different environment, from simple chat applications to large groups of users [13], both on the Internet and on mobile networks [15], where its usage extend the use of short message services [12] across different operators and providers.

The origin of XMPP in social, human-to-human, dialogues is a key property that makes it easy to develop social applications around by reusing existing contact lists (social graphs and "social residue") and offering the IM as an ancillary user and administrative channel. This social affinity can be subtly inherited to other XMPP application domains.

2) Interactive Social Media

Traditional media applications, such as television, are usually not associated with interactivity. Little interaction typically occurs between users visualizing content on similar platforms, fixed or mobile. Only moderate interactivity is added with, for example, SMS voting to affect the course of a live program. Hoekman et al. used XMPP to augment the traditional media delivery with social interaction [18], enabling users to: (1) share viewing experiences in real-time (as suggested in Figure 4); (2) give recommendations to their online buddies; (3) further expand their community by joining chat rooms with people having common TV program interests.

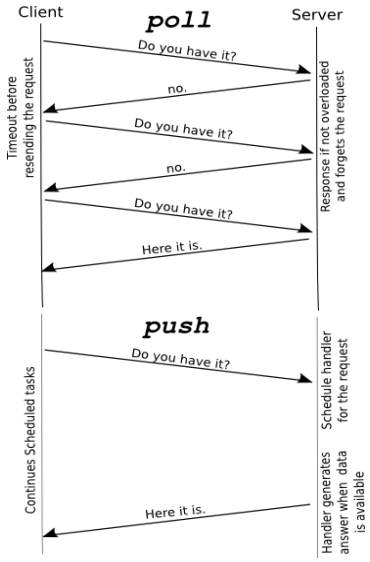

Figure 3: Poll vs Push

The flexibility and social affinity of XMPP provide an interactive interface to existing media services such as interactive TV (iTV) [18], mobileTV [20] [21] and even multi-player games [19].

XMPP is also well-suited for publishing and blogging [9] and in particular when used together with Atom feeds. Posts are in plain text. People learn and can reply to posts only if they have permission to view them. A feed is discovered based on identity of a user or of a service. Furthermore, the push nature of XMPP overcomes the problems of polling for updates via HTTP, which causes scaling issues in existing microblogging services (see Figure 3).

Recently, the Internet has witnessed the emergence of the Social Web characterized by people interaction with each other, brought together through shared interest. Social websites such as Facebook, MySpace and LinkedIn promote the sharing of an online identity (persona or profile), exposing data between users or a list of buddies (friends), and allow the exchange of messages - real-time or similarly to emails. All those features are natively supported by XMPP through stanzas, rosters, extended presence, PubSub, etc. In essence, using XMPP as the

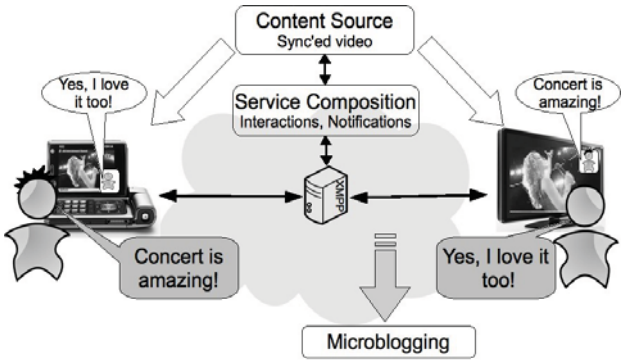

Figure 4: Interactive Social Media

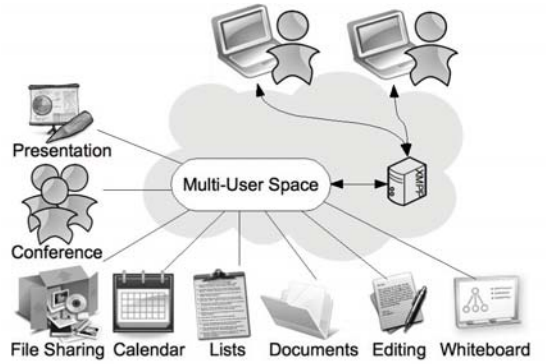

Figure 5: Cooperative Work Flow

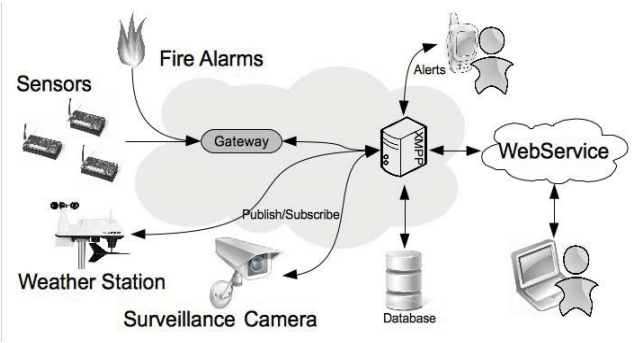

Figure 6: The Internet of Things with XMPP

social infrastructure means that friends can chat, share photos, publish ideas and play games from different platforms without having to create a different account, invite friends and build up the social network – all services can share one common user account, adhering to the OpenID mechanism.

3) Cooperative Work Flow

XMPP functionalities have been used as the foundation for communication and collaboration platforms [22] [23] – enabling instant messaging between users and near real-time interaction with hosted documents for multiple collaborators (as suggested in Figure 5). Synchronous and simultaneous modification support enables people to communicate and work together in new, convenient and effective ways: IM and presence services offer dynamic communication and the group notification functionality (through MUC or PubSub) distributes document consistencies to participating collaborators. In the case of Google Wave protocol [11], XMPP is used to build up a federation of servers in order to maintain consistency of hosted documents across several Wave servers. These systems are generalized and illustrated in Figure 5.

4) Internet of Things (IoT)

Recent additions to IP - 6loWPAN and the IPSO (IP for Smart Object) alliance - have made it feasible to connect wireless sensors and other low power devices to the Internet: enabling the IoT vision. Thus XMPP can: serve the event-driven paradigm of sensors; bridge multiple sensor networks; and, bring sensorial information directly to end-user applications [30] [31].

XMPP acts as a catalyst for real-time communication and interaction not only for human-to-human messaging but also for human-to-objects, actuators and sensors. Those objects represent an event node which is an XMPP addressable entity allowing for clients to subscribe to data feeds (as suggested in Figure 6). This also enables content aggregation, in gateways or in web-applications, and dynamic service construction, where information is combined to form new knowledge; e.g. temperature is combined with humidity and/or pressure (dew point), or motion detection combined with light sensing (occupancy). This enables end-users to get real-time information about, for example, the environment (exposure to pollution and UV). XMPP is well adapted to sensorial information distribution as it provides support for service discovery, data feed subscription, group distribution (based on MUC or geolocation), and data-logging (on the server side).

Labidi et al. [25] use XMPP as a notification middleware for real-time and large scale Integrated Ambient Systems and, in particular, for ubiquitous healthcare monitoring. The system uses Publish-Subscribe between a “Subject” and the “Observer” (usually more than one); the observer keeping track the subject's physiological and contextual data, extracted by body sensors. This system makes use of presence mechanisms to direct notifications of change in the data to available observers (using presence), which can initiate a fast and appropriate response.

Several authors provide XMPP-based infrastructures for Internet-scale sensing and actuation across a wide range of heterogeneous devices [24], [26], [28] and [29]. These systems are build around XMPP where transducers are modeled as event nodes in a publish-subscribe architecture. The extensibility, flexibility and ease to use of XMPP enables application developers to directly use physical data from sensors and actuators without having to manage low-level interfaces. However, the performance of XMPP in complex, highly constrained environment remains unclear.

5) Multi-Agent System (MAS)

MASes consist of dispersed cooperating elements which are used for problems that benefit from outsourcing tasks to specialized agents: typically harnessing underutilized resources of computers to tackle computationally challenging problems [32]. Many Grid computing systems are a form of MAS, where coordination of the resources is not centralized, using open, standardized and flexible protocols and interfaces. Typically, a MAS architecture has three entities: (1) users, (2) managers and (3) workers [33] [34] [35]. Users are generally simple agents which publish job requests to managers which in turn selects the most appropriate available worker to perform the job for the user (as illustrated in Figure 7) When the worker finishes the job, the manager returns to result to the user.

In MASes, XMPP has been mainly used for presence notifications, allowing systems to track, for example. resources (machine) presence and state in real-time [34] [35] or track web service availability [39] . Since XMPP is easily extensible, it is also used to transfer resource descriptions and

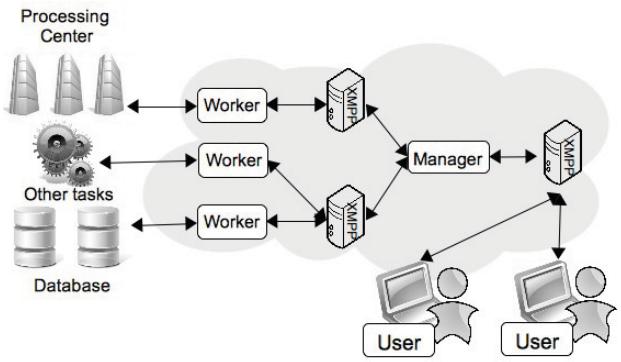

Figure 7: Multi-Agent System with XMPP

profiles as well as job descriptions. The MUC capability act as a market place and is used in a many-to-many fashion way, enabling easy organization and distribution of roles and resources. In [34], the authors have developed their own sub-protocol, custom XML name spaced content, to transmit jobs to workers using the iq stanza. In [37], a similar approach using iq was selected, but with the option to adhere to the Ad-Hoc command format.

Enhanced interaction and cooperation with existing web services using SOAP, popular in enterprise for exchanging XML structured data, is also possible with XMPP [8] providing both synchronous and asynchronous message delivery, without the need for complex protocols, such as WS-Routing and WS-Referral.

6) Cloud Computing

Clouds are often seen has the next evolution of MAS; from a system to solve resource demanding problems towards a more general-purpose infrastructure. Valquero et al. [40] offer a definition of cloud computing including these key characteristics: *a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services);* that *can be dynamically reconfigured to adjust to a variable load (scale)* and *optimum resource utilization.* Proponents of the cloud computing paradigm suggest that service applications, storage, and other resources will shift from local user desktops and company data centers to large, remote server farms accessible by users over the Internet.

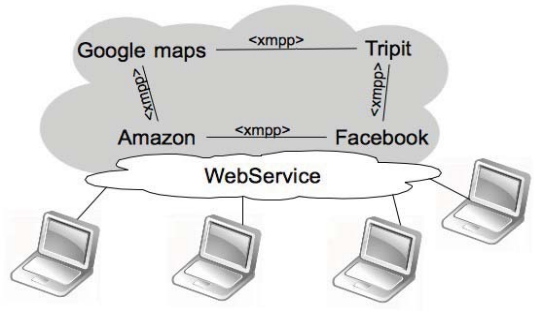

Figure 8: Inter Cloud Service connection using XMPP

XEP usage in Application Domain	Instant messaging	Interactive Social Media	Cooperative Work Flow	Internet of Things	Multi-Agent System	Cloud Computing
MUC	●	●	●	●	●	●
Jingle suite	●	●	●	●		
Extended Presence	●	●				
File Transfer	●	●	●			
Serverless Messaging	●			●		
Delayed Delivery	●	●				
PubSub		●	●	●	●	●
Microblogging		●				
BOSH		●			●	
SOAP over XMPP					●	
RPC			●			
Whiteboard			●			
Service Discovery				●	●	●
Ad-Hoc Commands					●	●
IO Data						●

Table 2. XMPP Extension usage in application domains (Class numbers, 1 to 6, as in the domain applications description) [8]

In [37], the authors demonstrate how XMPP together with the IO Data extension augment the traditional SOAP and REST approaches extremely well; their drawbacks being the lack of interoperability and discoverability of services, the inability to notify and indicate the availability. They show how XMPP for cloud services is a very compelling solution as the services are discoverable without external registry, its pushes mechanism eliminates unnecessary polling and IO Data extension enable in-band semantic description of services. In [38], the authors use XMPP for control plane presence and dialog protocol between different clouds (as suggested in Figure 8).

IV. Discussion and Conclusion

This review reveals the diversity of ways XMPP serves consumer electronics; from instant messaging, social media, collaborative work flows, internet of things to cloud computing. It also shows how XMPP is a powerful agent for IT convergence, and represents a timely opportunity for several currently separate consumer technology fields.

Since its initial release in 1999, XMPP has proven mature, scalable and very flexible, with a large open-source community and code libraries in many different programming languages, enabling developers to quickly build XMPP-powered applications [5]. Its semantics build on three core stanzas – presence, message and iq. XMPP provides a general layer to serve human-to-human, human-to-machines and even machine-to-machine synchronous and asynchronous communications. XMPP provides an efficient multi-user push mechanism for web applications and services, where clients register with servers to receive messages; when new data is available, the client is notified by the server; avoiding the polling problem. This preserves bandwidth and gives very low latency interactions, letting users as well as applications

interact efficiently on the same data in real-time. This is a latency, scalability and complexity benefit compared with polling and long-polling adaptations of HTTP-only methods.

XMPP is a mature and flexible foundation for applications and services that should leverage social components for their users; utilizing existing social relations without having to incentivize users in building new or redundant social graphs.

The six application domain classes proposed can be used to identify synergies between similar applications, and how specific XEPs and other mechanisms can be best utilized. They also offer an insight into the community development of XMPP extensions by clusters of application and people. Perhaps the most exciting consequence of this, is the insight of yet unconnected exploitations of this real-time Internet protocol. We expect these new connections to prove to powerful tools in the growing convergence of Internet, mobile and consumer electronics technologies and products.

References

[1] P. Saint-Andre, ed., Extensible Messaging and Presence Protocol (XMPP): Core, RFC 3920, 2004

[2] P. Saint-Andre, ed., Extensible Messaging and Presence Protocol (XMPP): Instant messaging and Presence, RFC 3921, 2004

[3] P. Saint-Andre, Streaming XML with Jabber/XMPP, IEEE Internet Computing vol. 9 no. 5, 2005

[4] P. Saint-Andre, Jingle: Jabber Does Multimedia. IEEE Multimedia, vol. 14 no.1 , 2007.

[5] XMPP foundation, online at http://xmpp.org/

[6] J. Myers, Simple Authentication and Security Layer, RFC 2222, 1999

[7] T. Dierks, C. Allen, Transport Layer Security, RFC 2246, 1999

[8] XEP – XMPP Extension Protocols, online at http://xmpp.org/extensions/

[9] Microblogging service using jabberIM, online at http://identi.ca/doc/im

[10] Amazon S3 services online at http://aws.amazon.com/s3/

[11] Wave Protocol, online at http://code.google.com/p/wave-protocol/

[12] Heng-Te Chu et al. A novel design of instant messaging service extended from short message service with XMPP, Proc of 5th IEEE International Conference on 3G Mobile Communication Technologies, 2004

[13] Chatterjee S. et al. Instant Messaging and Presence Technologies for College Campuses. IEEE Network vol. 19 (3) pp. 4 – 13, 2005

[14] Pohja M. Server Push with Instant Messaging. Proc. of the ACM Symposium on Applied Computing , 2009

[15] Yunxiang Gao et al. A Study on Jabber-based Instant Messaging System for Mobile Networks. IEEE International Symposium on Knowledge Acquisition and Modeling, 2008.

[16] J. Afonso et al. Development of an Integrated Solution for Intrusion Detection: A Model Based on Data Correlation. Proc. of the International Conference on Networking and Services, 2006.

[17] Saint-Andre. Jingle: Jabber Does Multimedia. IEEE Multimedia vol. 14 (1) pp. 90 – 94, 2007

[18] Hoekman et al. XMPP and iDTV or How to Make Television a Social Medium, Proc. of the 4th IEEE Consumer Communications and Networking Conference, 2007.

[19] Ohring P., Web-based multi-player games to encourage flexibility and social interaction in high-functioning children with autism spectrum disorder. Proc. of the 7th International Conference on Interaction design and children, 2008.

[20] Luyten et al. Telebuddies on the move: social stitching to enhance the networked gaming experience. Proc. of 5th ACM SIGCOMM workshop on Network and system support for games, 2006.

[21] Bentley F., Groble M. TuVista: meeting the multimedia needs of mobile sports fans. Proc. of the 7th ACM international conference on Multimedia, 2009.

[22] Roczniak et al. Design of Distributed Collaborative Application through Service Aggregation. Proc. of the 10th IEEE International Symposium on Distributed Simulation and Real-Time Applications , 2006.

[23] Hayden T., Ward C. Human collaboration tools for net-centric operations. Proc. of the International Symposium on Collaborative Technologies and Systems, 2005.

[24] Makowski L.,Michalski,A., Communication with nodes of distributed measurement system for hydrocarbon pollution monitoring using Jabber protocol, Proc. of the IEEE Instrumentation and Measurement Technology Conference, 2007.

[25] Labidi et al. XMPP based Health Care Integrated Ambient Systems Middleware. Developing Ambient Intelligence, Springer Paris, 2008.

[26] Rowe et al. Demo abstract: The Sensor Andrew infrastructure for large-scale campus-wide sensing and actuation. Proc. of the International Conference on Information Processing in Sensor Networks, 2009.

[27] Black et al. Impress: Towards Next-Generation Ubiquitous and Pervasive Computing Systems. 2007.

[28] Mittlboeck et al. Federal Pervasive Sensor Networks Serving Geographic Information, Proc. of the 5th International Symposium on LBS and Telecartography. 2008

[29] Liping Di et al. Support of Asynchrony in Sensor Web, Proc. of the 8th annual NASA Earth Science Technology Conference, 2008

[30] Hornsby A., Belimpasakis P., Defee I., XMPP-based wireless sensor network and its integration into the extended home environment, Proc of the 13th IEEE International Symposium on Consumer Electronics, 2009

[31] Hornsby A., Bail E. µXMPP: Lightweight implementation for low power operating system Contiki. Proc. of the International Conference on Ultra Modern Telecommunications, 2009.

[32] Schulz et al. A Network Substrate for Peer-to-Peer Grid Computing beyond Embarrassingly Parallel Applications. Proc. of the International Conference on Communications and Mobile Computing, 2009.

[33] Gregori, M et al. A jabber-based multi-agent system platform. Proc. of the 5th International Joint Conference on Autonomous Agents and Multi-agent Systems, 2006.

[34] Weis, G. and Lewis, A. 2009. Using XMPP for ad-hoc grid computing - an application example using parallel ant colony optimization. Proc. of the IEEE International Symposium on Parallel and Distributed Processing, 2009

[35] Stout, L., Murphy, M. A., Goasguen, S. 2009. Kestrel: an XMPP-based framework for many task computing applications. Proc. 2nd Workshop on Many-Task Computing on Grids and Supercomputers, 2009

[36] Roczniak and Saddik. JADE: jabber-based authoring in distributed environments. Proc. of the 13th annual ACM International Conference on Multimedia, 2005.

[37] Wagener et al. XMPP for cloud computing in bioinformatics supporting discovery and invocation of asynchronous web services. BMC Bioinformatics, 2009.

[38] Bernstein et al. Blueprint for the Intercloud-Protocols and Formats for Cloud Computing. Proc. of the 4th International Conference on Internet and Web Applications and Services, 2009.

[39] Konieczny et al. Establishing presence within the service-oriented environment. Proc. of the IEEE Aerospace conference, 2009.

[40] Vaquero et al. A break in the clouds: towards a cloud definition. ACM SIGCOMM Computer Communication Review, vol. 39 (1), 2008.

[41] P Saint-Andre, A. Houri, J. Hildebrand, Interworking between the Session Initiation Protocol (SIP) and the Extensible Messaging and Presence Protocol (XMPP): Core, draft-saintandre-sip-xmpp-core-01 (work in progress), March 2007