D2.2.5

Report on identified technological risks and recommendations

Version 1.0

WP2.2 Environment & Project Context

Dissemination Level: Public

Lead Editor: Julian Gallop, CCLRC

11 December 2006

Status: Final

SIXTH FRAMEWORK PROGRAMME

PRIORITY IST-2002-2.3.1.18

Grid for complex problem solving

Proposal/Contract no.: 004293

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. **"Collective Work"** means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
b. **"Derivative Work"** means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
c. **"Licensor"** means the individual or entity that offers the Work under the terms of this License.
d. **"Original Author"** means the individual or entity who created the Work.
e. **"Work"** means the copyrightable work of authorship offered under the terms of this License.
f. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;

b. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats, but otherwise you have no rights to make Derivative Works. All rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights set forth in Sections 4(d) and 4(e).

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any reference to such Licensor or the Original Author, as requested.
b. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.
c. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work, You must keep intact all copyright notices for the Work and give the Original Author credit reasonable to the medium or means You are utilizing by conveying the name (or pseudonym if applicable) of the Original Author if supplied; the title of the Work if supplied; and to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work. Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.
d. For the avoidance of doubt, where the Work is a musical composition:
    i. **Performance Royalties Under Blanket Licenses**. Licensor reserves the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work if that performance is primarily intended for or directed toward commercial advantage or private monetary compensation.

ii. **Mechanical Rights and Statutory Royalties**. Licensor reserves the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions), if Your distribution of such cover version is primarily intended for or directed toward commercial advantage or private monetary compensation.

e. **Webcasting Rights and Statutory Royalties.** For the avoidance of doubt, where the Work is a sound recording, Licensor reserves the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions), if Your public digital performance is primarily intended for or directed toward commercial advantage or private monetary compensation.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
c. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Context

Activity 2	Environment & Requirements definition
WP2.2	Environment & Project Context
Dependencies	Assumes an awareness of the technologies in D2.2.4 Needed as input for Activities 3 and 4 and 5

Contributors:

Contributors, including contributors to ID2.2.5
Angelo Gaeta (CRMPA)
Antonio Cuevas Casado (UPM)
Brynjar Viken (Telenor)
Christian Morariu (UniZH)
David Hausheer (UniZH)
Ignaz Mueller (HLRS)
Julian Gallop (CCLRC)
Kleopatra Konstanteli (NTUA)
Matteo Gaeta (CRMPA)
Nuno Inacio (IT-Aveiro)
Patrick Mandic (USTUTT)
Robert Piotter (HLRS)
Isabel Alonso (TID)
Vasiliki Andronikou (NTUA)

Reviewers
Antonis Litke (NTUA) – reviewed ID2.2.5
Per-Oddvar Osland (Telenor)

Approved by:

Stefan Wesner, USTUTT

Quality Manager

Version	Date	Authors	Sections Affected
0.9	26/6/2006	Julian Gallop	Complete draft of internal report ID2.2.5
1.0	20/7/2006	Julian Gallop	All changes accepted
1.1	21/7/2006	Brynjar Viken	Some editorial changes
1.2	31/7/2006	Julian Gallop, Brynjar Viken	Changes to respond to internal review by Antonis Litke. Comments retained.
1.3	17/8/2006	Julian Gallop	ID2.2.5 issued after internal review.
1.4	8/11/2006	Julian Gallop	Creating D2.2.5. Incorporated revised chapter 4 from Brynjar Viken.
1.5	5/12/2006	Julian Gallop	Incorporate contributions from NTUA and USTUTT; partial restructuring according to TN proposals; overall

			editing for clarity and correctness.
1.6	11/12/2006	Julian Gallop	Fixed comments from TN and CRMPA.

Table of Contents

List of Figures

List of Tables

1. Introduction and Executive Summary

This report builds on the Akogrimo State of the Art Report (D2.2.4) [106], but changes the emphasis. Where D2.2.4 was descriptive and in some cases introductory, this report D2.2.5 aims to provide an assessment of the available technologies, in general avoiding giving the introductory or tutorial material.

Although this report provides recommendations and motivation for a number of choices, the final decision on technologies to be adopted is made as part of the architecture, design and implementation processes.

Chapter 2 explains how the technological issues are discussed by considering the risks involved. Chapters 3 to 6 discuss the specific technological issues, one layer per chapter.

In this report a large number of technological issues are discussed. In most cases, a summary is provided as an Assessment Summary at the end of sections describing significant issues in Chapters 3 to 6. For those reasons, a summary is not repeated here in this introductory section.

2. Methods

Experts within Akogrimo assess the risks of certain choices. The risks of competing technologies are considered. Examples of risks could be: insufficient functionality, uncertain availability, uncertain acceptance by key players, dead end, no planned transition to the future, and so on. The report goes on to make recommendations regarding the risks.

A risk of an undesirable event happening is generally considered to be a combination of two things: the likelihood of that event; and its impact. In this assessment, it is not necessary to analyse this to a great precision. It is likely to be sufficient to know that (for instance) a particular undesirable event is very unlikely but the potential impact is serious.

The kind of undesirable event that may be considered in this report can include: not interoperable, not scalable, not sufficiently available, insufficient performance.

No alternative is risk-free. This report suggests what the risks are. In designing the architecture and prototype implementation, one makes a choice and then needs to be aware of an appropriate countermeasure. One set of possible alternative countermeasures is: prevent, reduce, accept, transfer or plan a contingency. For instance one can reduce interoperability problems by testing the multiple systems and restricting the prototype to well understood situations.

The assessment includes information on: functionality, standards, interoperability, implementations and performance/scalability.

3. Mobile Network Layer

This chapter assesses the main technologies that can be used to deploy a mobile network. Mobility is approached in different ways depending on the kind of mobility addressed, whether terminal mobility, user mobility, session mobility, inter-domain mobility or inter-technology mobility.

3.1. Access technologies

3G refers to the collection of third-generation mobile technologies (UMTS, Edge..) that are designed to allow mobile operators to offer integrated data and voice services over mobile networks (GSM/GPRS: has been used for quite a few years now and soon other more powerful technologies will be used instead).

WiFi refers to the 802.11b wireless Ethernet standard that has become the preferred technology for wireless local area networking in both business and home environments ([19], [22]). Extensions to the 802.11 standard are developed to achieve higher bit rates (802.11a, 802.11b and 802.11g - [19]) and improved security, to add quality of service (QoS) features (802.11e) and to provide better inter-operability.

WiFi, WiMax [20] and 3G [21] are wireless access technologies with different properties offering high bandwidth services and targeting the same groups. One of the key distinctions is that 3G and other mobile technologies use licensed spectrum, while WiFi uses unlicensed shared spectrum. Obviously, wired access networks are still an option. In general, wired access has better bandwidth but does not offer terminal mobility.

Different technologies have their pros and cons, thus it is most likely that heterogeneous access technologies will co-exist, which implies the need to interconnect them. A main trend seen in network technology is fixed/mobile convergence, allowing the end-user to use a wide range of services and applications across heterogeneous access technologies, both mobile and fixed.

3.1.1. Assessment summary

Although Akogrimo is an IP-based architecture and therefore independent of the access technology, a choice of access technology should be made for prototyping. This should be based on criteria such as cost, maturity and availability of software. WiFi is a mature technology that has low cost, uses unlicensed spectrum and software is available for different operative systems (Linux, Windows, etc.). It is therefore recommended that WiFi is used as access technology, in combination with wired access, for Akogrimo prototyping.

3.2. Terminal mobility

3.2.1. Protocols and implementations

Device mobility could be solved at different layers. However if below layer 3, it would depend on the access technology used, which would limit the concept of mobility considerably. If this was solved above, layer 3 performance would be notably decreased. In layer 3, the alternative for MIPv6 would probably only be the HIP protocol. However for the present MIPv6 seems to be more established and approved than HIP.

IPv6 was created to escape from the short comings of IPv4; there is no other IP-like alternative that could be used and it may progressively be used more and more.

The Mobile IPv6 protocol supports terminal mobility. A network node with Mobile IPv6 may move from network to network while maintaining connectivity. Mobile IPv6 is defined in RFC 3775 [17].

Several implementations exist, but the MIPL Linux implementation was the one chosen for Akogrimo due to available features, supported hardware, and source code availability.

- Linux supports a wide variety of hardware platforms, although the most commonly used is the Intel x86 architecture.
- MIPL supports the latest Mobile IPv6 specification.
- Despite the risk of bugs in an implementation of a new RFC, the MIPL implementation in particular, was found to be adequate. Since the MIPL project has available source code, there is less of a risk that development and bug fixing will come to a halt.
- New releases of MIPL may bring more stability. Feature-wise, it already implements the Mobile IPv6 RFC. New releases also mean a new kernel version for MIPL enabled machines, so developers must make sure that no incompatibilities with other software components are introduced, or that those incompatibilities are dealt with accordingly.

3.2.2. Assessment summary

For the reasons given above, Akogrimo uses Mobile IPv6 and specifically the MIPL Linux implementation.

3.3. Session-based mobility

Session Mobility is the seamless transfer of media of an ongoing communication session from a source device (the element that is currently carrying a session) to a target device (the entity which will support the session after the transfer).

3.3.1. Session Initiation Protocol (SIP)

This protocol manages session mobility by means of two main mechanisms:

- *Third-Party Call Control (3PCC)*, it is also called Mobile Node Control mode. This mechanism uses a SIP External Entity (SIP-EE) which controls the communication, and is different from finally involved into the data transfer. The SIP-EE which controls the communication is the Source terminal itself. The Source device arranges separate SIP sessions with the Target and the Remote device, but media stream is established between both of them. There are some risks involved:
    - This mechanism requires that this SIP External Entity (SIP-EE) remains active to maintain the sessions and perform any further actions, like termination or renegotiations.
    - There is a risk of a synchronization problem between the SIP-EE (Source device) and Target device. This problem can cause packet loss, and the Source device may stop the transmission until the Target device restarts the transmission.
        - Recommendation: one solution to this problem could be to have one Remote device User Agent (UA) which supports several simultaneous dialogues. That means, the Remote device is listening to both device (source

and target) and it only frees the resources associated with the Source, when it starts receiving packets from the Target device.

- *SIP REFER method.* Sessions could be transferred also using the SIP REFER method RFC 3515 [25], which avoids the requirement of having the source device active to maintain the session. This mechanism is also known as Session Handoff mode. This is useful if a user may need to transfer a session completely because the battery on his mobile device is running out. Alternatively, the user of a static device (like a video wall) who leaves the serving area may wish to continue the communication in his mobile device, so it performs a session transfer to it. In such cases, this approach is really useful.

3.3.2. H.323

Like SIP, it's also possible to make a Call Transfer service with H.323. ITU H.245 specification is in charge of precisely specifying this service [23]. Also, H.323 includes a call transfer specification [24].

3.3.3. Assessment Summary

Other aspects of SIP and H.323 are considered in section 4.4 "Signalling". For the reasons discussed in that section, SIP now has clear advantages compared to H.323 in the telecommunications/internet market.

3.4. Interdomain mobility

Mobile IPv6: This protocol incorporates interdomain mobility, or global mobility, by design. Prior agreements between network operators are, of course, necessary for allowing a foreign user in a network, but after setting up network infrastructure accordingly; Mobile IPv6 handles all mobility related issues by itself.

Diameter: This is an AAA protocol designed by the IETF [26] that allows a user to roam across different administration domains and is the leading protocol for this in terms of the commercialization of IP networks. The biggest threat would be that in the future non-IP architecture is used, which seems quite unlikely.

PANA: The Protocol for carrying Authentication for Network Access (PANA) [27] [28] was designed by the IETF in order for a client to authenticate, via a back-end AAA infrastructure, against other networks. It is designed for clients to not need to speak every kind of AAA protocol when they roam across networks using different ones. PANA uses IP so it is access technology independent. Risks for the use of this protocol are low. In the future probably newer protocols for this purpose will appear but for now this is the only one that we have at IP level.

EAP: The Extensible Authentication Protocol (EAP) [29] is defined by the IETF in order to carry different authentication methods depending on the requests. These methods may be based on shared secrets, certificates or any other authentication fundamentals. The use of EAP is quite widely extended and it can carry any kind of authentication procedure. Risks of becoming unavailable are low.

3.4.1. Assessment summary

The protocols described fulfil different roles and are all used in the architecture and prototype.

3.5. End-to-end security

The security goals stated in ISO 7498-2 are: Confidentiality (of entities, data, and traffic flow), Entity Authentication (unilateral or mutual), Data Authentication (connection-less or connection-oriented): data origin Authentication + data Integrity, Access Control and Non-Repudiation

The choices regarding security depend on the layer at which is applied. Thus applying it at the Application Layer has these advantages: it's closer to the user, enables transparent secure channels independent of the respective application, enables more sophisticated controls.
Applying security at a lower layer has these advantages: it's application independent, and it hides traffic data. But it's vulnerable at intermediate points.

3.5.1. End to end security using SSL/TLS and IPSec

3.5.1.1. SSL/TLS

SSL/TLS, Secure Sockets Layer (SSL) and Transport Layer Security (TLS), is a protocol with connection-oriented data confidentiality and integrity, and optional client and server authentication [30]. It's in between Application Layer and TCP Layer, and thus can be used to secure other applications than HTTP too (IMAP, telnet, ftp..). Since SSL/TLS works on Transport Layer it's transparent for the application and can be used for all TCP-based applications without modifying them.

SSL/TLS [6] provides:

- entity authentication
- data confidentiality
- data authentication
- data integrity

SSL/TLS drawbacks:

- The main drawback of this protocol is that it does not support applications, which run over UDP.
- It does not provide Non Repudiation.
    - SSL/TLS secure the communication channel, but not the exchanged messages
    - SSL/TLS does not use digital signatures in the first place (except for client authentication)
    - For electronic business more advanced security protocols are needed.
- It does not provide protection against traffic analysis (i.e. deducing information from traffic analysis even when the messages are encrypted [31])
- It does not secure many-to-many communications (multicast).

3.5.1.2. IPSec

IPSec (IP security) is a standard for securing Internet Protocol (IP) communications by encrypting and/or authenticating all IP packets, it's mandatory for IPv6 and optional for IPv4. IPSec provides security at the network layer**.**

IPSec is a set of cryptographic protocols for securing packet flows and key exchange. Of the former, there are two protocols: Encapsulating Security Payload (ESP), providing authentication, data confidentiality and message integrity; and Authentication Header (AH), providing authenti-

cation and message integrity, but without offering confidentiality. Currently only one key exchange protocol is defined, the IKE (Internet Key Exchange) protocol.

There are two performance modes with IPSec, Transport-Mode or Tunnel-Mode. The first of them is related host-to-host communications; while the second one is between security gateways.

IPSec [4] provides:

- Access control
- Connectionless integrity
- Data origin authentication
- Rejection of replayed packets (a form of partial sequence integrity)
- Confidentiality
- Limited traffic flow confidentiality

3.5.1.3. Assessment summary

IPSec versus SSL/TLS [7]

- IPSec operates at the network layer (layer 3), however SSL/TLS operates from the transport layer up (OSI layers 4 - 7). **This makes IPSec more flexible, as it can be used for protecting both TCP and UDP-based protocols**
- Operating at the network layer **increases IPSec complexity and processing overhead**, as it cannot rely on TCP (layer 4 OSI model) to manage reliability and fragmentation.
- **SSL/TLS easier to implement,** impact on usability. It is independent from MIPv6 however it usually needs the applications to be changed to support it and it doesn't provide a solution when using UDP.
- **IPSec protects packages, while SSL/TLS protects sessions.**
- IPSec: it is supposed to be **mandatory when using IPv6**. As any other security method, it has a drawback on usability. In addition, for now the interaction of IPSec with other IP layer protocols such as MIPv6 is not totally clear in all aspects.
- **IPSec is technically superior**
    - Rogue packet problem: TCP doesn't participate in crypto, so attacker can inject bogus packet, no way for TCP to recover

3.5.2. OpenPGP and S-MIME

Security services can be added to each communication link along a path, or appropriate security material can be wrapped around the data being sent, so that it is independent of the communication mechanism. This latter approach is often called "end-to-end" security and it has become a very important topic for users. The two basic features of this type of security are privacy (only the intended recipient can read the message) and authentication (the recipient can be assured of the identity of the sender). The technical capabilities for these functions have been known for many years, but they have only been applied to Internet mail recently. There are currently two actively proposed methods for providing these security services: S/MIME and PGP (both in its early incarnation as PGP/MIME, and as the new OpenPGP standard) [8]

S/MIME and OpenPGP are both secure and tested methods of protecting data. The primary difference between the two methods of protection is how they are implemented for the end user.

S/MIME requires certificates, and this requires the existence of a third-party certificate service. OpenPGP, on the other hand, is based on individually determined levels of trust. If someone sends you a key, and you trust it, you can communicate with this person. There is no third party involved, and therefore OpenPGP is probably limited for a dynamic and flexible architecture.

The trusted third parties in S/MIME are both an advantage and disadvantage when we consider the security model. The advantage of the existence of the third party certification server is that individual users are thoroughly audited by the certifier before their identity is determined. The disadvantage is that you need to trust the third party certification server to do their job. So, S/MIME has a smaller impact on usability than PGP with the correspondent trade-off on security.

3.5.2.1. Assessment summary

The most appropriate choice between S/MIME and Open PGP depends on the situation and whether a third-party certificate service is beneficial.

3.6. Network access security

In a mobile network it is extremely important to provide a secure connection between a user and the network. In order to achieve this goal, most access technologies offer their own encryption methods to provide message privacy.

3.6.1. GPRS/UMTS Access Network

The Access Network security mechanism allows the identification of a user on the radio access link by means of a Temporary Mobile Subscriber Identity (TMSI/P-TMSI). A TMSI /P-TMSI has local significance only in the location area or routing area in which the user is registered. Outside that area it should be a accompanied by an appropriate Location Area Identification (LAI) or Routing Area Identification (RAI) in order to avoid ambiguities. The association between the permanent and temporary user identities is kept by the Visited Location Register (VLR/SGSN), in which the user is registered.

The TMSI/P-TMSI, when available, is normally used to identify the user on the radio access path, for instance in paging requests, location update requests, attach requests, service requests, connection re-establishment requests and detach requests.

The security architecture in GPRS/UMTS networks is completely explained for more detail in 3GPP TS 33.102 v7.00 [2]

3.6.2. WLAN

One of the major problems of **WLAN network** is security. There are three technologies in charge of solving this issue. The following table shows as the technology has evolved to improve security.

	WEP (1999)	**WPA (2003)**	**WPA2 (IEEE 802.11i) 2004**
Encryption	RCA (24 bit IV)	RCA (48 bits IV)	AES
Key Rotation	None	Dynamic Session Keys	Dynamic Session Keys
Key Distribution	Manually typed into each device	Automatic distribution Available	Automatic distribution Available
Authentication	Uses WEP key as au-	802.1x & EAP	802.1x & EAP

	thentication		
Integrity	CRC	MIC	MIC

Table 1: Comparison of WLAN security technologies

Based on the table above, the WLAN [3] security standards can be summarised as follows:

- **WEP**
  This technology has major vulnerabilities regarding security, which the later approaches are intended to solve. WEP is based on RCA with static keys and initiation vector (IV) modified for each transmitted package. This enables hackers to recover the keys

- **WPA**
  Before the 802.11i solution, the WLAN vendors tried to improve the WEP failure by means of Wi-Fi Protected Access (WPA). The most important characteristics of WPA are:
    - Encryption key management: TKIP
    - Doubled IV to 48-bits
    - Better protection from replay & IV collision attacks
    - Per-packet keying (PPK)
    - Protects against key-recovery attacks
    - Broadcast key rotation

  However and due to backward compatibility reasons, TKIP continues to use a weak codification mechanism (RC4).

- **802.11i (WPA 2)**
  WPA2 is the name used by Wi-Fi Alliance in order to refer to the implementation of all 802.11i mandatory components. The 802.11i most important features are to add: IEEE 802.1x with EAP (Extensible Authentication Protocol), RADIUS, Kerberos, and encryption based on Rijdael algorithm AES. This mechanism also includes secure IBSS, secure fast handoff, secure de-authentication, disassociation and roaming support. (IBSS - Independent Base Service Set - is a mode of operation in 802.11i which allows modes to authenticate each other even when out of range of wireless access point)

3.6.2.1. Assessment summary

To summarise, it should be pointed out that 802.11i enables one overall security chain, that is: connection, credentials interchange, authentication and encryption much more secure and effective against hacker attacks. It is also sufficiently mature to be used in the prototype.

3.6.3. Virtual Private Network (VPN)

A highly heterogeneous network will probably make use of many different access technologies in order to provide access to different kinds of users. It is desired that no matter what the access technology utilized is, a certain degree of security can be assured. By means of a VPN (virtual private network), users can access a network securing their access independently of the access technology utilized. In order to do this, several protocols [5] can be used: IPSec, SSL and PPTP. PPTP will not be considered further because it is considered insufficiently secure.

3.6.3.1. IPSec VPN

The majority of IPSec VPN [4] solutions require third-party hardware and / or software. In order to access an IPSec VPN, the workstation or device in question must have an IPSec client software application installed. This is both a pro and a con.

- **Pro IPSec VPN**: IPsec tunnels between an MN and an AR are the only way to provide network access control transparently to any layer and independently of the access technology used. IPsec is mandatory for IPv6. So in principle this technology will have to be supported for all terminals
- **Con IPSec VPN**: the con for a commercial implementation is that it can be a financial burden to maintain the licenses for the client software and a nightmare for technical support to install and configure the client software on all remote machines - especially if they can't be on site physically to configure the software themselves. However most IPSec implementations are free and Linux comes with one by default.

3.6.3.2. SSL (Secure Sockets Layer) VPN

- **Pro SSL VPN**: SSL is a common protocol and most web browsers have SSL capabilities built in. Therefore almost every computer in the world is already equipped with the necessary "client software" to connect to an SSL VPN.
    - Another pro of SSL VPN's is that they allow more precise access control. First of all they provide tunnels to specific applications rather than to the entire corporate LAN. So, users on SSL VPN connections can only access the applications that they are configured to access rather than the whole network. Second, it is easier to provide different access rights to different users and have more granular control over user access.
- **Con SSL VPN**: Bigger overhead than IPsec.

3.6.3.3. Assessment summary

SSL VPN's have been gaining in prevalence and popularity; however they are not the right solution for every instance. The fact that IPSec has been chosen to be mandatory when using IPv6 makes this technology the most interesting future candidate.

3.7. Policy-based Network Management (PBNM)

Policy based management is present in many systems, for instance COPS-PR (COPS - Common Open Policy Service - usage for Policy Provisioning (COPS-PR)) [12] in DiffServ networks; another example is that the IRTF (Internet Research Task Force) included policy as one of the "pillars" of its AAA architecture. However, the goal of policy based network management is to encompass, under a single model, the policies of all the systems forming the network. The idea is to help bridge part of the conceptual gap between a human policy maker and a network element that is configured to enforce the policy. When a human business executive defines network policy, it is usually done using informal business terms and language. For example, a human may utter a policy statement that reads: "traffic generated by our human-resources application should have higher priority for making it through to its destination compared to traffic generated by people browsing the web during their lunch breaks" (abstract policy). While this statement clearly

defines QoS policy for the network, the network itself cannot enforce it. Translation to "network terms and language" (concrete configuration) is required.

Clearly this wide gap implies several translation levels, from the abstract to the concrete.

3.7.1. Network management policy specification

At the abstract end are the business policy rules. CIM [9], defined by DMTF, can be employed by Akogrimo. CIM facilitates a formal representation of network business rules, thus providing the first concretization level: formally representing humanly expressed policy. Working at a so high "level" raises several concerns:

- First, policy conflict. This is better explained by an example. We have two policies: policy 1 "users under 18 years old, have a 50% discount in mms between 20:00 and 8:00" and policy 2 "gold profile users have a 60% discount in mms between 21:00 and 9:00". If a user is gold and under 18, which policy should be applied? Mechanisms handling these conflicts must be designed, for instance assigning priorities to policies, or policy "set-union" rules.
- Another issue is that CIM alone can not be used, since we fall into the risk of having a nice handling of "human defined policies" with a nice user interface but lacking the ways to enforce them in the network elements.

CIM has a wide support, is well documented and many free source code tools exist. Besides it is based on XML and UML models, easy to understand and handle. So it can be a good choice for Akogrimo. There are several open code CIM servers. Two of the most widely used are OpenPegasus [11] and OpenWBEM [10]. Both offer higher benchmarking than other open source products. OpenWBEM uses MOF format to represent the policies while OpenPegasus employs XML which is more verbose and thus makes OpenPegasus to have higher memory requirements. OpenPegasus has more documentation available than OpenWBEM while OpenWBEM development is more active.

Even if CIM has a lot of support, there are models, like WSDL [15] more appropriate for Web Services and thus for Grid related issues. Indeed, the Akogrimo Grid infrastructure targets WSDL for Grid policy issues. There are already some WSDL products available, for instance, UDDI [14] which is a complete Web Service, WSDL-based commercial solution. Also there are available Java APIs and SDK [16]. However there are not many open source products so the development of the WSDL solution may, in some aspects, need to begin from scratch.

More than the availability of open source products, the risks of having divergent high level policies (CIM vs. WSDL) must be considered.

3.7.2. Network management policy enforcement

For policy enforcement issues more system-related policy models should be used, for example COPS-PR [12] based on PIBs [13]. There are open source COPS SDKs, such as the one developed by Intel. A bottom-up approach could be followed and reach the level of formal representation of human defined policies. As an example, A4C in Akogrimo has available a "complete" PBM solution, covering from the high level, human oriented and with friendly GUI, policies to the enforcement level of these policies. However, this bottom-up approach faces the risk of having isolated policy systems, non coherent and diverging at the top level. Conflicting (mainly concerning different systems) and self contradictory policies may arise and – both these situations require careful policy construction.

3.8. Quality of Service (QoS) in a mobile network

Implementation of Quality of Service depends mainly on the QoS Broker and the Access Routers. These components were developed using C++ and a number of supporting open source libraries. They also make extensive use of Linux kernel's network and QoS support.

3.8.1. Risks

3.8.1.1. Supported platforms

The QoS Broker and Access Routers have been extensively tested with various Linux distributions and kernels. Some issues may occur due to wrong library versions, but they are not dependant on a specific distribution or kernel.

All tests have been done using x86 architectures. Porting to other CPU architectures should not be difficult, but would probably imply some work.

3.8.1.2. Availability of features

At the time of writing, the QoS Broker incorporates most of the desired functionality. Notably missing is support for Web Services reservation. The QoS Client that will be used by applications in the mobile terminal for making explicit QoS requests, is also not finished as of yet, although that is not part of the QoS Broker itself.

The access router software component is mostly finished.

3.8.1.3. Malfunction of software (bugs)

Any of these software components are highly complex programs, and as such are bound to have bugs. The high number of dependencies on libraries and the Linux kernel means that on one hand, new features and bug fixes may become available at little or no cost, but on the other hand, changes to one of those dependencies may require changes in the programs.

3.8.2. Assessment summary

The software supporting QoS is still experimental and bears the resulting risks, but the approach appears to be the best available to achieve the prototype.

3.9. IP Multimedia Subsystem (IMS)

IP Multimedia Subsystem (IMS) is a next generation networking architecture originally defined by the 3GPP for 3G mobile phone systems in UMTS networks.

IMS is independent of the access network used, supports different network technologies, provides roaming and user mobility. IMS was also designed to allow the offering of any kind of IP-based service, and is in fact strongly oriented towards multimedia services.

The Akogrimo project follows an all-IP approach, with IPv6 acting as a convergence layer between the core network and any access technologies an operator might want to use. Functionalities of both approaches are similar, each one having its own strong points as well as weaknesses.

The main Akogrimo components from a network point of view are:

- A4C Server: acts as a Policy Information Point (PIP). Its functionality resembles the combination of the HSS and I-CSCF.
- QoS Broker: acts as a Policy Decision Point (PDP). Some of its functionality resembles the combination of P-CSCF and S-CSCF.
- Access Router: acts as a Policy Enforcement Point (PEP). Part of its functionality is close to the GGSN.

Multimedia service provisioning is supported by a SIP Server, and different services could be provided by different entities should the need arise.

Akogrimo's all-IP approach defines a universal architecture for heterogeneous environments capable of supporting any kind of service. By having separate PDP and service provisioning servers, the Akogrimo network provides the ability of supporting any application signalling protocol, thus not being tied to a single protocol (SIP) as is the case of IMS. This approach allows a simpler and more flexible architecture, which is suitable for any kind of service provisioning, whereas IMS is optimized for multimedia services.

3.10. References for Mobile Network Layer

[1] Beaujean, Ch., Chaher, N., et al., Implementation and Evaluation of an End-to-End IP QoS Architecture for Networks Beyond 3rd Generation, http://www.it.uc3m.es/cgarcia/articulos/IST_Mobile_Summit_2003.pdf

[2] 3GPP, "3G security; Security architecture", 3GPP TS TS 33.102 v7.0.0 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security Architecture (Release 7) , http://www.3gpp.org/ftp/Specs/html-info/33102.htm

[3] M.G.Nystrom, Wireless Security Primer, http://xianshield.org/presos/Wireless%20Security%20Primer.ppt, March 2005

[4] Kent S. and Atkinson R., Security Architecture for the Internet Protocol, RFC 2401, 1998, http://www.ietf.org/rfc/rfc2401.txt

[5] Bradley T, VPN's: IPSec vs. SSL, http://netsecurity.about.com/cs/generalsecurity/a/aa111703_2.htm

[6] T. Dierks, C. Allen, The TLS Protocol Version 1.0, RFC 2246, 1999, http://www.rfc.net/rfc2246.html

[7] B.Preneel, Internet Security Protocols, June 2005, http://www.iaik.tugraz.at/teaching/00_angewandte%20kryptografie/internet.pdf

[8] S/MIME and OpenPGP, http://www.imc.org/smime-pgpmime.html

[9] Common Information Model (CIM) Standards http://www.dmtf.org/standards/cim/

[10] Open Web Based Enterprise Management http://www.openwbem.org/

[11] OpenPegasus http://www.openpegasus.org/

[12] COPS Usage for Policy Provisioning (COPS-PR). K. Chan, RFC3084, March 2001, http://www.ietf.org/rfc/rfc3084.txt .

[13] "Framework Policy Information Base for Usage Feedback" D. Rawlins, RFC3571, August 2003, http://www.ietf.org/rfc/rfc3571.txt .

[14] Oasis UDDI http://www.uddi.org/

[15] Web Services Description Language (WSDL) http://www.w3.org/TR/wsdl

[16] Java API's for WSDL http://www.serviceoriented.org/java_apis_for_wsdl.html

[17] "Mobility Support in IPv6", D.Johnson, C.Perkins, J.Arkko, RFC3775, 2004, http://www.ietf.org/rfc/rfc3775.txt .

[18] Handley M. et al. SIP: Session Initiation Protocol. RFC 2543, 1999, http://www.ietf.org/rfc/rfc2543.txt

[19] Wireless Fidelity (Wi-Fi) – Specifications. http://www.irit.fr/~Ralph.Sobek/wifi/

[20] "The IEEE 802.16 Working Group on Broadband Wireless Access Standards - developing the IEEE 802.16 **WirelessMAN**® Standard for Wireless Metropolitan Area Networks", IEEE, http://www.ieee802.org/16/

[21] 3G tutorial – Overview of the Universal Mobile Telecommunication System (UMTS), draft (2002), http://www.umtsworld.com/technology/overview.htm

[22] Lehra, W., McKnight, L. 'Wireless Internet access: 3G vs. WiFi', Telecommunications Policy, (2003) Volume 27: pp. 351–370

[23] Recommendation H.245, Control protocol for multimedia communication, International Telecommunication Union (ITU), http://www.itu.int/rec/T-REC-H.245/en

[24] Recommendation H.450.2, Call transfer supplementary service for H.323, International Telecommunication Union (ITU), http://www.itu.int/rec/T-REC-H.450.2/en

[25] "The Session Initiation Protocol (SIP) Refer Method – SIP REFER", R.Sparks, RFC3515, April 2003, http://www.ietf.org/rfc/rfc3515.txt

[26] Calhoun P. et al., Diameter Base Protocol. RFC 3588, http://www.ietf.org/rfc/rfc3588.txt

[27] J. Bournelle et al., Using PANA in the Mobile IPv6 Integrated Case, Internet draft - Work in Progress., draft March 2006, http://mirror.switch.ch/ftp/mirror/internet-drafts/draft-ietf-pana-pana-11.txt

[28] Forsberg D. et al., Protocol for Carrying Authentication for Network Access (PANA). Internet draft - Work in Progress., draft March 2006, http://mirror.switch.ch/ftp/mirror/internet-drafts/draft-bournelle-pana-mip6-01.txt

[29] Aboba B. et al., Extensible Authentication Protocol (EAP). RFC 3748, June 2004, http://www.ietf.org/rfc/rfc3748.txt

[30] What's the difference between SSH and SSL/TLS?, The Snailbook, http://www.snailbook.com/faq/ssl.auto.html

[31] Traffic analysis, Wikipedia, http://en.wikipedia.org/wiki/Traffic_analysis

4. Mobile Network Middleware Layer

Middleware is a set of software components that provide interconnection, integration and interoperability of distributed systems and services. It offers a uniform interface shielding applications from unnecessary details, heterogeneity and complexity of underlying computer platforms and networks. A middleware platform also provides common services to software applications (such as authentication, credit card payments, directory services, etc). For Telcos middleware is seen as a key enabler for the open services market. Further, middleware also offers solutions to resource sharing and fault tolerance requirements [32] [73].

Hence, middleware provides application developers with a higher level of abstraction built using the primitives of the underlying systems. This facilitates faster development of applications and easier porting of applications to other platforms.

Akogrimo mobile network middleware comprises a range of technologies intended to enhance the Grid infrastructure in terms of mobility support and cross-layer integration. It should offer open and unified interfaces for Grid application developers to the variety of underlying technologies, providing a glue between Grids and mobile networks.

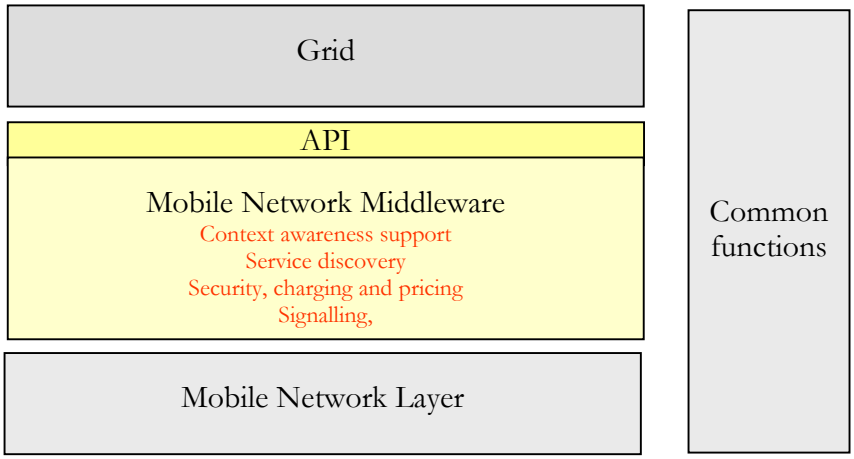

Figure 1: Layered architecture showing technologies for Mobile Network Middleware

As shown in Figure 1, Akogrimo mobile network middleware layer addresses the following functionality:

- **Context-awareness support**: Akogrimo introduces the mobile Grid, where interactive services involving mobile and nomadic users are of key importance. By knowing the situation of Akogrimo users, Grid applications may more efficient choose the right people or services to participate in a workflow, and better adapt service behaviour.
- **Security, charging and pricing**: to make the system user friendly and to make the incurred usage costs transparent, there should be an A4C infrastructure covering all layers, allowing e.g. single sign-on, and flexible composition of value chains.
- **Signalling**: offering management of multimedia sessions while hiding user mobility. Although network management is not a core middleware topic, in Akogrimo it has some relevance to the middleware layer, and is hence briefly mentioned.
- **Service discovery**: Akogrimo should provide a service discovery infrastructure making it possible to find and access service and devices being offered.

In the following, an assessment of relevant technologies for the Akogrimo mobile network middleware layer is presented. We assess these technologies according to parameters such as features, standards, interoperability, maturity, acceptance and available implementation.

4.1. Network middleware technologies

This section presents an evaluation of middleware technologies that are considered relevant for Akogrimo. Technologies in the following areas are addressed:

- Architecture, interfaces and protocols for communication between distributed processes
- Platforms and tools for implementation and deployment of middleware service

4.1.1. Java Platform

Functionality: The Java Platform [58] includes the Java Platform Standard Edition (Java SE), the Java Platform Enterprise Edition (Java EE, formerly J2EE), and the Java Platform Micro Edition or Java ME. Java EE [59] is a programming platform for developing and running distributed multi-tier architecture Java applications, based largely on modular software components running on an application server. Java EE includes several API specifications, such as JDBC, RMI, e-mail, JMS, web services, XML, etc, and defines how to coordinate them. Java EE also features some specifications unique to Java EE for components.

Standards: Sun Microsystems manages the technical specifications for Java.

Interoperability: Java virtual machine and implementations of the standard libraries are available for most platforms, making it possible to run Java programs on different underlying operating systems.

Implementations: A variety of Java EE application servers are available [59], including:

- BEA WebLogic
- JBoss Application Server
- Sun Java System Application Server
- WebSphere Application Server by IBM

Performance/Scalability: Depends on performance of application server. However, JRE introduces some overhead that may cause problems for applications that have strict real-time requirements.

4.1.2. Common Object Request Broker Architecture (CORBA)

Functionality: CORBA emerged as a promising middleware architecture for object communication in potentially heterogeneous and distributed environments. Wireless CORBA intends to provide wireless access and terminal mobility using CORBA.

Standards: CORBA [39] and Wireless CORBA [40] are specifications from the Object Management Group.

Interoperability: Using the standard protocol IIOP, a CORBA-based program from any vendor, on almost any computer, operating system, programming language, and network, can interoperate with a CORBA-based program from the same or another vendor, on almost any other

computer, operating system, programming language, and network.[1] Thousands of sites now rely on CORBA for enterprise, internet, and other computing [41].

Implementations: As the architecture developed and established itself as an industrial standard, a number of various implementations appear. A list of available implementations is found at [55].

Performance/Scalability/Security: The Object Management Group maintains an entire web-site devoted to user design wins and success stories [54], and in many cases these are large scale operations.

4.1.3. Java Remote Method Invocation (RMI)

Functionality: Java RMI [47] is a mechanism that allows methods of remote Java objects to be invoked from other Java virtual machines, possibly on different hosts.

Standards: Java Remote Method Protocol is the Java technology-specific protocol for looking up and referencing remote objects. Java RMI has evolved towards becoming more compatible with CORBA. In particular, there is now a form of RMI called RMI/IIOP ("RMI over IIOP") that uses the Internet Inter-ORB Protocol (IIOP) of CORBA as the underlying protocol for RMI communication [79].

Interoperability: Java RMI enables invoking methods of remote objects on another Java virtual machine running on almost any operating system.

Implementations: Part of the Java platform.

4.1.4. Web Services

Functionality: The W3C defines a Web Services as follows [57]: "A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards." Web Services have emerged as the most promising candidate for realising Service Oriented Architecture (SOA).

Standards: Specifications are mainly standardised by OASIS and the W3C. Web Services relates to multiple specifications (WSDL, SOAP RFC 3288 [45], etc.).

Interoperability: The approach is based on open interfaces, providing interoperability independent of underlying operative system, programming language and vendors. A major challenge is that many standards exist. The WS-I organisation has improved interoperability between different web service implementations, by developing a series of profiles to further define the standards involved [56]. Many Web Services protocol specifications are in an advanced status (SOAP, WSDL, UDDI) and they can be considered a good basis to guarantee interoperability. However the same cannot be stated about the existing implementations of these specifications. Some implementations are not complete and this requires some careful assessment by developers to work around the specific interoperability issue.

Implementations: Platforms and tools for Web Services are widely available, such as:

[1] SOAP to CORBA bridging software: http://soap2corba.sourceforge.net/

- The **Microsoft .NET Framework** [46] is an integral component for building and running software applications Web Services.
- **IBM WebSphere** [50] provides middleware to set up, operate and integrate e-business applications across multiple computing platforms using web technology. It has been constructed using open standards such as the Java 2 Platform, Enterprise Edition (J2EE), XML and Web Services standards.
- **Apache Tomcat and Axis**
- Light weight implementations such as Jetty web server

There are multiple vendors offering technologies for realising Web Services, however, it has to be investigated if service platforms are mature enough to offer telco grade performance. In general, Web services protocols have text-based formats that may suffer from poor performance compared to other distributed computing approaches such as RMI and CORBA. Hence, Web Services may not be suited for distributed applications with strict real-time requirements.

4.1.5. Web Service Resource Framework

Functionality:

Often clients of a system need to access the internal state of some entities, e.g. an airline reservation system needs to access information about reservations. Web service specifications do not model the state of a service, i.e. a Web Service does not maintain any state between invocations by itself. To deal with internal state, workarounds are made by e.g. having the web service read from a database. Rather than implementing state by having Web Service access an internal database, Web Service Resource Framework (WSRF) offers standards for implementing stateful web services [74][75].

Standards: Web Service Resource Framework (WSRF) is a family of OASIS-published specifications for Web Services. Major contributors include the Globus Alliance and IBM.

Four specifications known as the Web Service Resource Framework (WSRF) defines interfaces and behaviour for representing and handling state in distributed systems based on Web Services. WS-Notification (WSN) is a group of specifications related to WSRF which defines interfaces and behaviours allowing clients to subscribe to changes in the state of an entity and receive notification when changes occur based on topics [74].

Interoperability: WSRF/WSN is consistent with the recommendations of the WS-Interoperability Basic Profile. This means that any WS-I compliant Web Service client can interact with any service that support the WSRF specifications. That is, different implementations of WSRF/WSN have a base level of interoperability on XML, SOAP, WSDL and HTTP. On the other hand, there are large differences in security, programming models and performance[74].

Implementations:

Several implementations of WSRF are available. In [74] a comparison of five WSRF and WSN implementations is made.

- Globus Toolkit 4
    - GT4 Java WS core
    - GT4 C WS core)
- WSRF.NET: an implementation of WSRF and WS-Notification based on the .NET framework.

- pyGridWare: Python WSRF implementation.
- WSRF::Lite perl based implementations.

Performance testing in [74] showed that GT-C was fastest in every tests. WSRF.NET and GT-Java were comparable with "no security". Further, implementations support WSN in varying degree:

- GT4 Java WS core and pyGridWare:
    - Does not implement WS-BrokeredNotification
    - Supports only flat topic spaces
    - Only basic subscription: the precondition, selector, and subscription policy elements are ignored.
- GT4 C WS core:
    - Does not implement producer-side notification.
- WSRF::Lite does not support any Notification specifications.

In addition, implementations of WSRF and WSN are available from Apache Software Foundation: Apache Pubscribe and Apache WSRF.

4.1.6. Jini

Functionality:

Jini technology [48] is a service oriented architecture that defines a programming model which both exploits and extends Java technology to enable the creation of adaptive distributed systems. Jini technology can be used to build adaptive systems that are secure, scalable, evolvable and flexible as typically required in dynamic computing environments.

It comprises technologies such Java Spaces Technology and Jini extensible remote invocation (Jini ERI). Jini offers functionality such as service discovery and mobile code [82].

Standards:

The specifications has been released under the Apache 2.0 license and offered to the Apache Software Foundation's Incubator.

Interoperability:

Jini is agnostic with respect to communication protocols (RMI/CORBA/SOAP/HTTP/...) but is implemented most of the times using RMI[83]. To enable Web Service service and a Jini client to interoperate, typically a bridge that translates from one framework to the other, in both directions, is required [81]. JISGA [80] extends a Jini system into an OGSA-compliant infrastructure for Grid computing by introducing Web service techniques.

Implementations:

There are several implementations available [82]:

- Starter Kit has been released under the Apache 2.0 license and offered to the Apache Software Foundation's Incubator
- The Blitz Project - Open source, BSD licensed, JavaSpaces implementation
- GigaSpaces - Jini based commercial grid platform

4.1.7. Middleware Access to Network Management

The actual mechanisms and protocols used by the network layer to realise different QoS levels DiffServ, IntServ, MPLS, etc.) or to gather data for network management (SNMP, RMON, CMIP, CMIS..) are not directly relevant for the middleware layer. Mobile network middleware traffic consists mainly of signalling sessions between components in the core network (or wired part of the access network) and some traffic to the mobile terminals. Such signalling sessions do not need high bandwidth but require a low delay.

The middleware layer may need to access QoS mechanisms and management information through appropriate interfaces.

4.1.8. Assessment Summary

Technologies from Akogrimo D2.2.4 [106] such as Distributed Component Object Model (DCOM) [43], JXTA [49], Microsoft Message Queuing (MSMQ) technology [51], BEA MessageQ [52], JORAM [53], Open Source Message Queue (OSMQ) [70] and xmlBlaster [71] are not recommended for the Akogrimo project because they have not gained much acceptance, are proprietary solutions, have limited interoperability and/or scalability.

Jini, CORBA and Java RMI are candidates with good functionality that have been accepted in many industries. However, they offer limited interoperability.

For Akogrimo it is vital to use open interfaces and platforms allowing transparent communication between programs implemented in different languages running on different operating systems. Additionally, given the position of Grid Services within the Akogrimo project and the role of WSRF within Grids, it is clear that the middleware platform should based on Web Service technology. Web Services/SOAP is also a good candidate for management interfaces mentioned in Section 4.1.7.

Web Service based technology is emerging as the prime candidate for realising Service Oriented Architecture, it has committed support from major vendors and will likely have a wide acceptance. The technology offers good interoperability.

A drawback is that it is unclear if telco grade service platform based on Web Services currently can be supported. Further, given the numerous standards there is a risk of interoperability issues between different implementations of Web Services standards

Web Service Resource Framework (WSRF) can be a candidate for implementing stateful middleware Web Services. In particular, WS-Notification can be used to implement services that need topic-based subscribe/notify mechanisms such as the Akogrimo Context Manager.

The Java platform is widely accepted, supports implementation of Web Services and facilitates portability across different platforms. Therefore, Java should be a prime candidate for prototype realisation of Akogrimo mobile network middleware. There is a variety of tools, web containers and application servers available for developing and deploying Java applications. For proof of concept testing within the Akogrimo project, it is sufficient to based the implementation and deployment on tools that are freely available (e.g. Eclipse and Netbeans for development, Apache Tomcat and Axis for deployment, Globus Toolkit 4 for WSRF).

4.2. Context-aware middleware

A context-aware service is a service that is able to adjust its behaviour to provide maximal utility within the current situation of the user. On one hand, standalone applications could gather, proc-

ess and store context information by itself. An obvious drawback of such approach is that every application has to deal with details of every underlying context source to gather and process context information.

During the last years many layered context-ware systems and frameworks have evolved from which a common layered architecture is identifiable when analysing their design [76]. The layers identified are sensor layer (in this setting a sensor is any data source that can provide usable context information), data retrieval layer, pre-processing layer (reasoning and interpretation), and finally storage and management layer.

A general requirement for context-ware middleware is modularity and flexibility such that new context sources can be added or removed easily as required by different application domains. In [76] it is noted that data retrieval layer often is implemented in reusable software components which makes it possible e.g. to replace a RFID system with a GPS system without major modifications. Access by clients may happen synchronously or asynchronously. Given the fact that context information may change rapidly, asynchronous mode is more suitable in most cases.

There are many aspects of context information:

- **Available networks:** What networks are available and what kind of services do they offer in terms of bandwidth, QoS guarantees etc. What tariffs apply?
- **Terminal capabilities:** What are the I/O and computational capabilities of the terminals that are currently available to the user?
- **Location and geography:** Where is the user located and what are the conditions at this location? Which infrastructure and what services are available there, and which threats and annoyances?
- **Activity:** What is the user doing?
- **Physiological and mental state:** What is the bodily state of the user and what is his mood?
- **Time and date.**

The definition of context is open-ended and cannot be completely specified for all purposes; the set of data needed will ultimately depend on the application domain. Based on requirements from Akogrimo scenarios [69], the network middleware keeps track of user context in terms of presence, user location and device capabilities. Hence, in the following relevant technologies for these areas are assessed.

4.2.1. Presence technologies

Functionality: Presence information is the basis of instant messaging (IM) and refers to information about the state of users such as availability, reachability and other information set by the user (e.g. mood, interests, etc.). This allows users to detect whether their friends/colleagues are online and if it is possible to communicate with them. In addition information about planned future availability of the user given in a calendar can be included. Presence information is dynamic and may change frequently. The changes can occur manually by user interaction or automatically based on available context information (e.g. location, user talk on the phone etc.).

Standards: There are multiple protocols for handling presence information such as:

- XMPP (Extensible Messaging and Presence Protocol) [2] (IETF)
- SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions) (IETF)
- OMA Wireless Village
- Parlay/OSA PAM
- Parlay X Presence
- A number of proprietary protocols

Interoperability: Eurescom project P1402 [85] tested interoperability of IM technology in fixed and mobile environments. Products from selected vendors implementing different presence protocols (XMPP, SIP SIMPLE and Wireless village) were tested. The tests showed there was basic or no interoperability between different protocols.

The interoperability between protocols and vendors could occur by means of gateways between system suing different presence protocols. A bi-directional protocol mapping for use by gateways between XMPP and SIP SIMPLE is presented in IETF Internet-draft [84]. Such gateways are not readily available.

Public IM networks such as AOL, Yahoo, and MSN have closely guarded their subscribers and shied away from standards, although some small steps at interoperability through individual agreements aimed at corporate users have been taken [86]. Some software clients, such as Trillian from Cerulean Studios, offer access to several IM services by supporting several underlying protocols.

Implementations: There are several implementations, both commercial and open, available:

Presence servers: Jabber, Colibra

Presence clients: SIP Communicator

4.2.1.1. Assessment

For Akogrimo, SIMPLE [67] is the prime candidate for supporting Presence, due to the use of SIP as a main technology in the project. SIMPLE does not support planned availability. RFC 3856 [68] defines how SIP is able to provide presence/context information and defines the needed components.

4.2.2. Positioning technology

Functionality: Several technologies may be used for determining position of mobile terminals. Some key technical features of such technologies are:

- **Accuracy:** What is the deviation between the position given by the positioning system and the actual position? What is the deviation between repeated measurements for the

[2] IMPP (Instant Messaging and Presence Protocol) has been succeeded by XMPP, APEX (Application Exchange) and PRIM (Presence and Instant Messaging Protocol) were early presence protocol proposals to IETF.

same (stationary) terminal? What is the deviation between two terminals in the same location?

- **Coverage:** What geographical area does the system cover? Can it be used indoors? Does it require line-of-sight between the terminal and base stations or satellites?
- **Speed:** How long must a terminal be in a given position for that position to be measured?

The available technologies for positioning can divided into the following categories[3][77]:

1) Satellite based positioning

   The main satellite based positioning system is GPS [78] which is funded by and controlled by the U.S. Department of Defense (DOD). The European Galileo system should be operational by 2008.

   GPS offers world-wide coverage but works poorly in areas where the signal is weak, such as indoor and in urban areas. Originally, GPS was typically accurate to about 15 meters but improvements in receivers and satellites has improved the accuracy the recent years (possible to achieve very high precision). To improve GPS coverage, accuracy and to reduce terminal processing requirements several ground-based augmentation systems, i.e. using servers on the ground to support the mobile terminal, have been developed (Differential GPS (DGPS), Assisted GPS (AGPS), indoor GPS™). Using network of ground-based transmitters in combination with GPS can provide accuracy of a few centimetres both outdoor and indoor [92].

2) Cellular network based positioning

   Cellular network based positioning, based on GSM and UMTS, typically offers an accuracy in the range from 60 meters to several kilometres depending on position method being used. Positioning is available only in areas where the cellular network has coverage, i.e. offers both indoor and outdoor coverage but does not cover remote areas.

3) Wireless short-range network based positioning

   Wireless short-range network based positioning has an accuracy in the range from 0.5 meter 80 meter depending on positioning method. To achieve a high accuracy it may be required to install additional Bluetooth or WLAN interfaces in the network. Positioning can be made available in areas with WLAN or Bluetooth networks, usually inside buildings and in urban areas. There is no standardized way of making such measurements, and current solutions rely on proprietary client software.

4) Tag (badge) based positioning

   Smart tags or RFID tags are intended to replace bar codes for tagging objects with a machine-readable label. RFID is a concept rather than a standard, and there are several incompatible competing technologies. The maximum distance between a tag and a

[3] A solution using indoor illumination for positioning has also been proposed: LuxTrace - Indoor Positioning Using Building Illumination, "http://www.vs.inf.ethz.ch/res/papers/bohn_puc_2006_luxtracert.pdf#search=%22accurate%20indoor%20Positioning%20%20%22"

reader can from less than a meter and up to a few meters, depending on the implementation.

Interoperability: A global industry initiative, Location Interoperability Forum (LIF), was formed in 2000 with the intention to develop and promote industry common solutions for location based services. Mobile Location Protocol (MLP) is a specification that defines an interface for position of mobile terminals independent of underlying network technology and positioning method. The approach allows applications to seamlessly access location data from different positioning technologies.

4.2.2.1. Assessment

The choice of location technology is ultimately determined by the application requirements with respect to coverage, accuracy and indoor/outdoor usage. Actually, the applications can often benefit from using a combination of different location technologies. Hence, to facilitate different application needs, the design of a multi-purpose mobile network middleware platform must be flexible enough to support different location technologies in combination.

For Akogrimo realisation, the following factors affect the selection of location technology:

- The demonstrator must be based on IPv6 WLAN access.
- The demonstrator must support indoor positioning.
- The demonstrator must be portable and easy to set up.
- The demonstrator should offer positioning with an accuracy of a few meters

As the access network will be based on IPv6 WLAN, using cellular networks (GSM and UMTS) for positioning are not recommended. Further, since indoor positioning is required, GPS cannot be used without ground-based augmentation system. However, GPS with ground-based augmentation introduce a relative high cost compared to other solution which probably does not improve the demonstrator is such a way that it can be justified

One drawback with WLAN and Bluetooth positioning is that, to compute terminal location accurately, data from multiple stations must be processed. That is, several fixed Bluetooth devices or additional WLAN access points (or specialised hardware) must be installed for accurate positioning. For a demonstrator a disadvantage is that such a solution has an additional cost and does not provide the required portability.

RFID allows the user to be positioned by carrying a badge only. Other advantages of RFID are low hardware costs, easy to prototype/demonstrate services, easy to move demonstrator to a different location and that the implementation resources needed are limited. A drawback with RFID is lack of interoperability between readers and badges from different vendors. RFID (passive) technology seems to be a good candidate for Akogrimo prototyping.

4.2.3. Determining terminal capabilities

Functionality: Part of the information to be made available as context is the capabilities of the terminal at the present time.

Standards: There are several standards available for describing the capabilities of a terminal such as Composite Capabilities/Preference Profiles (CC/PP) [87], User Agent Profile (UAProf)

[89], OMA Device Information (Devinf) (previously SyncML) [90] and UPnP device description [88].

- The W3C has defined the Composite Capabilities/Preference Profiles (CC/PP) framework [87]. The typical use of CC/PP is to adapt a web page to a handheld terminal with limited resources. CC/PP is based on the Resource Description Framework (RDF), which is a metadata modelling language. CC/PP can specify such data as screen size, number of colours, supported Java VM versions etc. The model is extensible, allowing new classes and attributes to be added. An HTTP request may contain an URI referring to a CC/PP profile stored on an arbitrary WEB server, eliminating the need for transferring the profile over the wireless link. If the profile changes, the client may send only the differences since the last request.
- The UAProf standard, defined by the Open Mobile Alliance (OMA), is a subset of CC/PP, and is specifically aimed at WAP sessions. From [89] the schema for WAP (Wireless Access Protocol) User Agent Profile (UAProf) contains the following: HardwarePlatform, SoftwarePlatform, BrowserUA, NetworkCharacteristics, WapCharacteristics and PushCharacteristics. Additional components can be added to the schema to describe capabilities pertaining to other user agents.
- OMA Device Information (Devinf) (previously SyncML) [90] has been implemented directly using XML. The DevInf device description comes in a four parts: the device, the content types it can accept, its data store and any extensions it supports [91].
- The UPnP device description, maintained by the UPnP forum, is expressed in XML and includes vendor-specific, manufacturer information like the model name and number, serial number, manufacturer name, URLs to vendor-specific web sites, etc. The description contains a list of any embedded devices or services, as well as URLs for control, eventing, and presentation [88].

Interoperability: Depending on the actual device profile, it can be possible to translate between different device description standards. This requires detailed inspection and understanding of classes and attributes of standards involved.

4.2.3.1. Assessment Summary

From the scenarios in D2.3.1 [69] the following Device information is needed to provide general services.

- Screen Size ++ (pixels, color depth etc.)
- Operating System (version)
- Memory
- Network connections
- Browser
- Installed SW and players

SyncML and UPnP descriptions will likely need some extensions to be used. UAProf covers the elements needed from scenario descriptions and can be extended if required. For Akogrimo realisation CC/PP or UAProf is recommended for describing Device context (profile).

4.3. Service Discovery

The relation between SIP and Service Discovery converges in two points. First, use of SIP as the main infrastructure for local Service Discovery provisioning and the second one is publishing the available services on the user side as part of the Presence Context information through SIP.

The risks involved in doing **Service Discovery by means of SIP** are as follows:

1. It is a new completely mechanism, so, no standard.
2. The PUBLISH and SUBSCRIBE messages body is to be defined, so, no standard
3. If XML is used inside PUBLISH message body, this means all associated risks to XML, such as: XML processing is very CPU, memory, and I/O or network intensive. Besides of this, is hard of creating and hard of parsing.

Besides the mechanism above (by means of SIP) there are other protocols which could be used for Service Discovery, with the following technological risk:

- **SLP**: Limited query semantics. Limited expression syntax (just pairs). Not pushed by industry lately in favour of other protocols.
- **Zeroconf:** Made for ad-hoc environments and therefore hard to secure. Limited expressivity of services description.
- **Bluetooth:** Not an IP-based protocol. Very constrained for this reason.
- **UPnP:** uses HTTP over UDP (known as HTTPU and HTTPMU for unicast and multicast), even though this is not standardized. May overload the network.
- **SD** specification of UPnP abandoned since 2000. Not very flexible. Few narrowly defined solutions.
- **UDDI**: Not exactly suitable for dynamic services.

4.4. Signalling

In the Mobile Network Middleware Layer there are several signalling protocols which make providing Grid-based services possible.

This section includes:

- A comparison of SIP and H.323
- An introduction to RTP and RTCP
- A comparison of RTSP against HTTP and SIP

Each technology/protocol has its technological risk, and it's possible that between two similar technologies one of them is functionally better than the other, but is unlikely to be accepted, so, if we choose which is likely to gain wide acceptance, somehow we have to work with its limitations.

4.4.1. Session Initiation Protocol (SIP)/H.323

There are several differences between these protocols. While ITU from telecommunications world wanted create a complex protocol (H.323), IETF from the Internet world, created a simple but powerful protocol (SIP).

The primary reason for the existence of two non-interoperable signalling protocols is that both the telecommunication and the Internet world wanted to have protocols meeting their traditions. ITU wanted to have a sophisticated norm utilizing their other sophisticated norms, whereas IETF defined a protocol well fitting its puzzle of simple and powerful tools. Internet telephony is located on the border of the both worlds and it is difficult to predict which approach will gain the most popularity eventually. However, if the technical aspects discussed in this section and introduction of novel integrated services will have the last word, **SIP's chances are high**

The following table (based on [61]and [62]) shows some differences between them.

Characteristic	**SIP**	**H.323**
Use in 3GPP	YES	NO. many expect H.323 to disappear with deployment of 3GPP networks
Complexity	Adequate: HTTP-like protocol	High: ASN, use of several different protocols (H.450, H.225.0, H.245)
Scalability / Load-Balancing	SIP enables scalability by means of several Proxies, at DNS level or by means of SIP redirections.	H.323 has the ability to load balance endpoints across a number of alternate gatekeepers in order to scale a local point of presence. In addition, endpoints report their available and total capacity so that calls going to a set of gateways, for example, may be best distributed across those gateways.
Reliability	SIP enables handling of device failure by means of re-REGISTER messages so that the Proxy server is aware of this. Besides this, one device could be registered into several SIP Proxies, so, if one SIP proxy fails the others would go on working There is also a new SIP PING method currently, which provides an indication to both ends of a session (User Agent Client and User Agent Server) that signalling messages can still flow between them. The SIP **PING method** is intended to confirm that the endpoints are alive and verify that a signalling path is still valid.	H.323 has defined a number of features to handle failure of intermediate network entities. For example, if a Gatekeeper fails, the protocol is designed to utilize an alternate Gatekeeper. If a call is being routed through intermediate signalling entities fails, H.323 has the wherewithal to re-route the call to an operational entity so that call is not disrupted.
Message Encoding	SIP messages are encoded in ASCII text format, suitable for humans to read. Textual encoding is easy to	H.323 encodes messages in a compact binary format that is suitable for narrowband and broadband connections

	read. Textual encoding is easy to extend, debug and process by text-processing tools As a consequence, the messages are large and less suitable for networks where bandwidth, delay, and/or processing are a concern. This is the reason for appearance of alternatives such as **SigComp**, in order to get more compressed messages (nowadays used into 3G mobile network) Another issue which is currently being analysed is the problem with some SIP messages which are getting so large that they are reaching the MTU (maximum transmission unit) of the network, risking router fragmentation. As a direct result, it has been suggested within the SIP community that UDP be deprecated and TCP would be used instead.	rowband and broadband connections. Some Internet protocols are binary, e.g., IP, TCP, UDP, ICMP, DNS, LDAP, SNMP, RADIUS, NTP, DHCP, and SSH, and for those that are text, there are or have been efforts to provide a binary form, e.g., HTTP and XML, so there is a concern even within these communities that text is not always appropriate. Moreover, consider where these encodings are used. The textual encoding of control information--signalling--makes sense for protocols whose main purpose is the transmission of truly human-readable textual content, such as email or web pages, but not for pure signalling applications, such as a control protocol for voice communication, where the encoding is rarely encountered by humans.
Instant Messaging Support	YES	NO
Extensibility - Vendor Specific	SIP is extended by adding new header lines that may be used by different vendors to serve different purposes. SIP provides more specifications than H.323 in order to do something like this in a standard way.	H.323 is extended with non-standard features in such a way as to avoid conflicts between vendors. However, debugging binary extensions is difficult since they are unknown to protocol analyzers and unreadable to humans
Extensibility - Standard	SIP is extended by the standards community to add new features in such a way as to not impact existing features. Several extensions have been added to current SIP version, and continue to be backward compatible.	H.323 is extended by the standards community to add new features to H.323 in such a way as to not impact existing features. However, new revisions of H.323 are published periodically, which introduce new functionality that is mandatory, yet done in such a way as to preserve backward compatibility.
Scalability - Call Signalling	When using a SIP proxy to, for example, perform address resolution for the SIP device, the proxy is required to handle at least 2 full message ex-	When an H.323 gatekeeper is used, it may simply provide address resolution through one RAS message exchange, or it may route all call signalling traf-

	changes for every call.	fic. In large networks, the direct call model may be used so that endpoints connect directly to one another.
Inter-domain call routing	hierarchically by DNS.	statically by Annex G
Addressing	Any URL including E-mail address, H.323, http, and E.164 URLs.	host (without username!), gatekeeper-resolved alias (arbitrary case-sensitive string, e.g. E-mail address), E.164 telephone numbers
Accounting / Billing	If the SIP proxy wants to collect billing information, it has no choice but to stay in the call signalling path for the entire duration of the call so that it can detect when the call completes. Even then, the statistics could be wrong because the call signalling may have been delayed. Since the accounting based on time could be wrong due to signalling delay, then, one solution could be accounting based on volume of data. That is, the entity which controls the volume of data (Access Router) sends this information to the entity in charge of billing (A4C)	Even with H.323's direct call model, the ability to successfully bill for the call is not lost because the endpoint reports to the gatekeeper the beginning and end time of the call via the RAS protocol. As with SIP, it could happen that the last message is not received, so, would be necessary to add further mechanism in order to bill.
Call Setup	A call can be established in as few as 1.5 round trips.	A call can be established in as few as 1.5 round trips.
Media Transport	RTP/RTCP, but most implementations use UDP	RTP/RTCP, but most implementations use TCP, so the usage of TCP results in higher call set-up time
Capability Negotiation	SIP has means of exchanging capabilities by means of UPDATE messages or re-INVITE message. Besides of this, it's possible to send for instance device capabilities information inside of SIP PUBLISH message.	H.323 entities may exchange capabilities and negotiate which channels to open, including audio, video, and data channels. Individual channels may be opened and closed during the call without disrupting the other channels.
Video and Data Conferencing	SIP has limited support for video and no support for data conferencing protocols like T.120. However currently, there is a draft regarding data conferencing (MSRP).	H.323 fully supports video and data conferencing.
Synchroni-	SIP doesn't control the conference;	H.323 provides control for the con-

zation	RTP/RTCP protocols are in charge of this task.	ference as well as lip synchronization of audio and video streams.
Services	SIP enables mechanism such as Third Party Call Control, or configuration control through XCAP. Besides of this, one may provide ad-hoc services through other means, such as XML, SOAP.	Services may be provided to the end-point through a web-browser interface using HTTP or a feature server using Megaco/H.248. In addition, services may be provided to an end-point as it places a call, as a call arrives, or during the middle of a call by a gatekeeper or other entity that routes the call signalling. As a result, H.323 is better suited to providing new services.
Web-Integration	Integration with other Internet services (e.g. a caller may send an E-mail to an unreachable callee)	

Table 2: SIP / H323 comparison

Some further indications that by comparison with H.232, SIP is making inroads into telecommunications/internet market are the following:

- H.323 is not evolving, remains obsolete.
- H.323 is more complex than SIP.
- The providers are launching SIP solutions.
- H.323 is more addressed to internet.
- SIP was the selected protocol by 3GPP standard in order to integrate into 3G mobile networks. So, SIP will be used in NGN/IMS. One of the reasons for selecting it was its ability to detect user presence, thus making it easy to manage multiparty communications among different users. Other reason was that SIP easily enables the integration with others protocols such as: SDP, SMTP, RTSP, thus makes easy to program complex streaming mobile services, multimedia session control, and so on.

4.4.2. Real-Time transport Protocol (RTP)/Real-time Transport Control Protocol (RTCP)

The Real-Time transport Protocol (RTP) is the de facto standard media transport protocol in the Internet

- RTP does not: guarantee QoS for real-time services, address resource reservation, perform signalling (negotiate the media format) and doesn't guarantee packet delivery.
- RTCP is a companion control protocol to RTP which is in charge of end-to-end monitoring and data delivery and QoS.
- Both SIP and H.323 run over RTP/RTCP, that is, both of them use RTP/RTCP as media transport protocol.

4.4.3. Real Time Streaming Protocol (RTSP)

This protocol provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video.

- ***RTSP/HTTP***:

  The Real Time Streaming Protocol RTSP [63] has some overlap in functionality with HTTP. However, RTSP differs fundamentally from HTTP in that data delivery takes place out-of-band in a different protocol. It usually works on conjunction with RTP protocol to deliver streaming video, audio and text content, although RTSP does not depend on the transport mechanism used to carry continuous media.

  HTTP is an asymmetric protocol where the client issues requests and the server responds. In RTSP, both the media client and media server can issue requests. RTSP requests are also not stateless; they may set parameters and continue to control a media stream long after the request has been acknowledged.

  Re-using HTTP functionality has advantages in at least two areas, namely security and proxies. The requirements are very similar, so having the ability to adopt HTTP work on caches, proxies and authentication is valuable.

  Other differences between both protocols are as follows:

  - RTSP maintains a server state during transmission unlike HTTP.
  - Avoids shortfalls/limitations in HTTP.
  - RTSP provides synchronization of events.
  - Enhancement of HTTP functions

- ***RTSP/SIP***: both are very similar.

  Similarities:

  - Both SIP and RTSP are used to initiate multimedia sessions where the actual user data is carried "out-of-band", usually using RTP.
  - Both use SDP to describe sessions.
  - They have similar syntax, derived from HTTP or email.
  - RTSP and SIP both support a form of aggregate control, i.e., the ability to control multiple streams from different locations with one control session.
  - Both support redirection.
  - Both use DNS to resolve URIs.

  Differences:

  - **RTSP is used only for streaming media**, e.g., video-on-demand, **while SIP supports any session** type (usually used for multimedia sessions, although SIP supports others session types).
  - RTSP URIs are closer to HTTP URIs, identifying a server and file, while SIP URIs identify users.
  - SIP supports proxies for routing requests; that's not really supported in RTSP.

4.4.4. Assessment summary

Considering the comparison of H.323 and SIP (section 4.4.1), the overall assessment for Akogrimo purposes favours SIP.

Related to RTP and RTCP, both are companion protocols which work together. This means, RTP is used as de facto media transport protocol, and RTCP is in charge of end-to-end monitoring and data delivery and QoS.

Finally comparing RTSP and SIP, RTSP is used only for streaming media, e.g., video-on-demand, while, although usually used for multimedia sessions, SIP supports any session type.

4.5. Security

For ensuring a healthy infrastructure for Grid Services every entity that it is using it must obey a set of rules. One of the main tasks of the Network Middleware layer is to define such rules, strictly monitor if they are respected and take necessary measures when abuse is detected. Defining the good and bad behaviour and its monitoring and accountability is the task of the AAA subsystem.

4.5.1. Functionality - Challenges & Requirements

Security requirements within the Grid environment are driven by the need to support scalable, dynamic, distributed virtual organizations. The main challenges for a mobile grid architecture can be grouped in the following categories:

- *Heterogeneous distributed environment* - in a heterogeneous environment of different types of devices running different software, participation in the grid necessitates management of transparent access.
- *Multiple security mechanisms* – a platform that spreads across several physical and logical network domains having different access technologies should provide the necessary mechanisms to facilitate the interoperability of the different security architectures
- *End-to-end security* - cooperating systems with different security policies and protocols will have to negotiate trust arrangements in order to provide end-to-end security (identification, authentication and authorization)
- *Dynamic creation of services* – users must be able to create new services and resources dynamically without the intervention of an administrator. These services must be coordinated and must interact securely with other services.

In order to provide quality security services at the Network Middleware Layer, the following requirements should be addressed:

Authentication – for enabling interoperability the platform should provide plug points for multiple authentication mechanisms

Authorization –access *to* grid services must be controlled based on authorization policies attached to each service. It should accommodate various access control models and implementations

Delegation – establishment of dynamic trust domains requires facilities to allow for delegation of access rights from requestors to services.

Single sign-on – participants in a grid environment often need to coordinate multiple resources to accomplish a single task. Security mechanisms have to ensure that once a successfully authentication is performed no need for re-authentication is required.

Secure logging – facilities for time stamping and mechanisms for securely logging any kind of operational information or event should be provided. The term *securely* in this context means reliably and accurately so that this information cannot be altered by inappropriate agents.

Privacy – both service requester and service provider must be allowed to define and enforce privacy policies.

Manageability – security management in Grids is needed, such as: identity management, policy management, key management.

Security considerations shall be taken in all of the four layers in the Akogrimo platform. The Network Middleware shall be responsible for secure authentication of users, offering access to different services based on user credentials and keeping track of all the events occurred, for accounting and billing as well as for security purposes.

4.5.2. Standards - AAA Infrastructure

Diameter [37] is a new AAA protocol that aims to replacing the RADIUS protocol. Diameter closely follows the AAA protocol requirements as specified in [33] by IETF. Although RADIUS is a widely used AAA protocol today, Diameter addresses some of the key requirements of AAA protocols that RADIUS does not implement:

- *Failover* – RADIUS does not define failover mechanisms. Any Radius implementation has different failover behaviour.
- *Transmission-level security* – RADIUS does not support packet-level confidentiality. [36] defines the use of IPSec with RADIUS, but support for IPSec is not required. Diameter makes the use of IPSec mandatory and provides optional support for TLS connectivity.
- *Reliable transport* – While RADIUS uses UDP for message delivery, Diameter makes use of TCP and STCP which results in a reliable transport of AAA messages.
- *Agent support* – Diameter defines explicitly agents behaviour, such as Relays, Proxies or Redirects. In RADIUS, these agents differ between different implementations.
- *Server-initiated messages* – Support for server-initiated communication is only optional in RADIUS, but mandatory in Diameter. Server-initiated messages are important when the AAA server needs to request reauthentication/reauthorization or to abort an ongoing session.
- *Capability negotiation* - RADIUS does not support error messages, capability negotiation, or a mandatory/non-mandatory flag for attributes. Since RADIUS clients and servers are not aware of each other's capabilities, they may not be able to successfully negotiate a mutually acceptable service, or in some cases, even be aware of what service has been implemented. Diameter includes support for error handling, capability negotiation, and mandatory/non-mandatory attribute-value pairs (AVPs).
- *Peer discovery and configuration* - Diameter enables dynamic discovery of peers, which does not exist in RADIUS.

- *Roaming support* – One important thing missing in RADIUS and provided by Diameter is the support for roaming operations [34] [35] which is very important in the context of a mobile grid.

In the state of the art deliverable a recommendation for using Diameter as the protocol to be used for A4C communication. The main risks of using Diameter are analyzed below:

- *Adoption by the mobile communication community* - Diameter is already adopted by 3GPP as the protocol to carry AAA information in the next generation mobile networks. The risk of this protocol being rejected by this community is very low.
- *Adoption by the grid community*- Grid frameworks currently have no standardized protocol for carrying AAA information. Moreover, the AAA infrastructures built in different projects don't fully comply with the AAA protocol requirements specified in [33]. As grid services are based on Web-Services technologies, the use of a new communication protocol for AAA issues will regarded with scepticism by the grid community. As a mobile grid platform will deliver services on top of next generation mobile networks, which already standardized Diameter as the AAA protocol to be used, grid service providers will be required to integrate Diameter in their technologies to make sure interoperability with other service and network providers.
- *Technical integration in architecture components* - There are only a couple of implementations of Diameter protocol, and they are under development. There are two important risks in the integration of Diameter framework in existing components:
    - *Lack of functionality* – as an open source framework (which is also under development) is to be used, the framework might not provide the full capabilities specified by the protocol. Since Akogrimo will not be a final commercial product, but a proof of concept, and also due to the fact that Diameter developer community is very active, this issue does not have to influence the overall acceptance of Diameter protocol.
    - *Technology interoperability* - There is no Java-based framework for Diameter today. As most Web Services are written in Java, integration of a C++ component might be a risk. As shown in the first Akogrimo demonstration, the integration of a C++ based A4C client in a Java component is feasible from a technical point of view.

4.5.3. A4C Implementation Risks

One aspect of A4C implementation is the need to interface with Web Services – this is discussed in section 5.4.

There is a technical risk related to the implementation of A4C functions mainly due to their dependence on other software libraries, e.g. OpenDiameter. Three main risks can be identified with respect to this dependency:

Portability

If a specific software library is not portable or not available for a certain platform, this affects also the portability of software which depends on that library. The non-portability of software can have a big impact on the deployment, as it limits the scope of platforms on which a certain software can be deployed. This is especially dangerous if other components with certain deployment requirements rely on that software.

Supported platforms of OpenDiameter currently include Linux, FreeBSD, and Windows 2000/XP. It is unclear if and when support for other platforms will be added. This means that the A4C client, which is implemented on top of OpenDiameter, would be limited to those three platforms. However, currently it supports only Linux.

Availability of Features

The set of features provided by a certain software library often increases with every new release. Early releases of a library typically contain only the most important features, while less important features will be added later on. Often, a roadmap exists which outlines the set of features planned to be added in future releases. Sometimes a library implements a certain standard which typically defines a specific set of features that must exist. However, the library might only implement a part of those standardized features, i.e. not be fully compliant with the standard.

There is a risk, if an implementation relies on certain features of a particular library based on its roadmap or the standard it implements. If those features will not become available in time, the implementation might fail to provide its functionality.

OpenDiameter is an implementation of the Diameter Base Protocol defined in RFC3588 and certain other IETF standards. Features planned to be added in future releases include, e.g. an agent translating between Diameter and RADIUS.

Currently, all features that A4C relies on are provided by OpenDiameter.

Malfunction of Software (Bugs)

Intermittent – and therefore hard to detect – errors associated with releases of software libraries, especially early ones, can cause a security risk with software such as A4C. For example they could allow an attacker to exploit the error by sending a malformed function call to the library.

4.5.4. Assessment summary

AAA in Akogrimo should be based on the DIAMETER protocol, since it has a strong support for mobility and it is very easily scalable for any other future requirements for the AAA implementation. Several gateways should be provided for interaction with legacy AAA systems like RADIUS and TACACS which are largely used by network operators. The overall AAA architecture shall be layer-independent with extension modules capable of offering layer-specific functions in each of the four layers.

A4C client will probably not be made available for other platforms such as Windows, as the resources for this are not available within the project. Thus, components that rely on A4C client will have to be deployed on Linux.

With respect to features and bugs of libraries on which A4C depends, it is recommended to closely follow the new releases of those libraries, namely OpenDiameter, and port A4C to these new releases whenever an important new feature becomes available or a critical bug is fixed.

4.6. References for Mobile Network Middleware Layer

[32] C. Mascolo, L. Capra and W. Emmerich. "Middleware for Mobile Computing (A Survey)", In Advanced Lectures on Networking - Networking 2002 Tutorials, Pisa, Italy. volume 2497 of LNCS, pages 20-58, Springer Verlag. May 2002.

[33] Aboba, B., Calhoun, P., Glass, S., Hiller, T., McCann, P., Shiino, H., Zorn, G., Dommety, G., Perkins, C., Patil, B., Mitton, D., Manning, S., Beadles, M., Walsh, P., Chen, X., Sivalingham, S., Hameed, A., Munson, M., Jacobs, S., Lim, B., Hirschman, B., Hsu, R., Xu, Y., Campbell, E., Baba, S. and E. Jaques, "Criteria for Evaluating AAA Protocols for Network Access", RFC 2989, November 2000.

[34] Aboba, B., Lu, J., Alsop, J., Ding, J. and W. Wang, "Review of Roaming Implementations", RFC 2194, September 1997.

[35] Aboba, B. and G. Zorn, "Criteria for Evaluating Roaming Protocols", RFC 2477, January 1999.

[36] Aboba B., Zorn G., Mitton D., "RADIUS and IPv6", RFC 3162, August 2001.

[37] Calhoun P., Loughney J., Guttman E., Zorn G., Arkko J.. "Diameter Base Protocol", IETF RFC 3588, September 2003.

[38] Computer Weekly, 21 April 2004, http://www.computerweekly.com/Article130068.htm

[39] Object Management Group (OMG), "Introduction to OMG's specifications, CORBA", http://www.omg.org/gettingstarted/specintro.htm#CORB

[40] Object Management Group (OMG), OMG Specification, "Wireless Access and Terminal Mobility in CORBA", 2004, http://www.omg.org/docs/formal/04-04-02.pdf

[41] OMG web site, "CORBA basics", http://www.omg.org/gettingstarted/corbafaq.htm

[42] Microsoft, "COM: Component Object Model Technologies", http://www.microsoft.com/com/tech/DCOM.asp

[43] L. F. Cabrera, C. Kurt, D. Box, Microsoft, "An Introduction to the Web Services Architecture and Its Specifications, version 2.0", 2004, http://msdn.microsoft.com/webservices/webservices/understanding/advancedwebservices/default.aspx?pull=/library/en-us/dnwebsrv/html/introwsa.asp

[44] A. Skonnard, Understanding SOAP, March 2003, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnsoap/html/understandsoap.asp

[45] E.O'Tuathail, M.Rose, "Using the Simple Object Access Protocol (SOAP) in Blocks Extensible Exchange Protocol (BEEP)", RFC 3288, http://www.ietf.org/rfc/rfc3288.txt

[46] Microsoft .NET Framework Developer Center, http://msdn.microsoft.com/netframework/

[47] Sun Microsystems, "Java Remote Method Invocation (Java RMI)", http://java.sun.com/products/jdk/rmi/

[48] Sun Developer Network, "Jini Network Technology", http://java.sun.com/developer/products/jini/index.jsp

[49] JXTA™ technology web site, http://www.jxta.org/

[50] IBM WebSphere software, http://www-306.ibm.com/software/websphere/

[51] Microsoft, " Microsoft Message Queuing", http://www.microsoft.com/windows2000/technologies/communications/msmq/default.asp

[52] BEA MessageQ™, http://www.bea.com/framework.jsp?CNT=index.htm&FP=/content/products/more/messageq

[53] Joram Web site, "JORAM: Java (TM) Open Reliable Asynchronous Messaging", http://joram.objectweb.org/

[54] Object Management Group (OMG) , "CORBA® Success Stories", http://www.corba.org/success.htm

[55] A. Puder, "CORBA Product Profiles", http://www.puder.org/corba/matrix/

[56] Wikipedia, "Web Service", http://en.wikipedia.org/wiki/Web_service

[57] W3C, "Web Services Architecture", W3C Working Group Note, 11 February 2004, http://www.w3.org/TR/ws-arch/

[58] Wikipedia, "Java Platform", http://en.wikipedia.org/wiki/Java_platform

[59] Wikipedia, "Java EE", http://en.wikipedia.org/wiki/Java_Platform%2C_Enterprise_Edition

[60] http://java.sun.com/javaee/5/docs/tutorial/doc/Security-J2EE.html

[61] Iptel.org website, SIP versus H.323, **http://www.iptel.org/info/trends/sip.html**

[62] Packetizer, website, H.323 versus SIP: A Comparison, **http://www.packetizer.com/voip/h323_vs_sip/**

[63] "Real Time Streaming Protocol (RTSP)" , H.Schulzrinne, A.Rao, R,Lanphier, RFC2326, http://www.ietf.org/rfc/rfc2326.txt

[64] Java Sun website, Java Technology and XML-Part 3: Performance Improvement Tips, **http://java.sun.com/developer/technicalArticles/xml/JavaTechandXML_part3/**

[65] S. E. Spero, Analysis of HTTP Performance problems, http://www.ibiblio.org/mdma-release/http-prob.html

[66] J. Wedvik et al., "Overall Network Middleware Requirements Report", Akogrimo Deliverable D4.2.1, 2005, http://www.akogrimo.org/modules.php?name=UpDownload&req=getit&lid=39

[67] IETF working group, SIP for Instant Messaging and Presence Leveraging Extensions (simple), http://www.ietf.org/html.charters/simple-charter.html

[68] J.Rosenberg, "A Presence Event Package for the Session Initiation Protocol (SIP)", 2004, RFC3856, http://www.ietf.org/rfc/rfc3856.txt

[69] C. Loos et.al., "Testbed Description", Akogrimo Deliverable 2.3.1, 2005, http://bscw.hlrs.de/bscw/bscw.cgi/d57381/Akogrimo_D2.3.1_Final.pdf

[70] Open Source Message Queue, http://www.osmq.org/

[71] XmlBlaster web site, "What is xmlBlaster?", http://www.xmlblaster.org/

[72] "Intelligent Mobile Agents", https://www.cs.tcd.ie/research_groups/aig/iag/pubreview/chap5/chap5.html, chapter 5 of "Software Agents : A Review", S.Green, L.Hurst, B.Nangle, P.Cunningham, F.Somers, R.Evans, 1997, https://www.cs.tcd.ie/research_groups/aig/iag/pubreview/chap1/chap1.html

[73] O. Risnes et.al., "Project P910: Middleware for telecommunications. Facilitating the open services market", EURESCOM project P910 on "Technology assessment of middleware for telecommunications", 2001.

[74] M. Humphrey et.al., "State and Events for Web Services: A comparison of five WS-Resource Framework and WS-Notification implementations", 14th IEEE International Symposium on High Performance Distributed Computing, July 2005.

[75] Wikipedia, "Web Services Resource Framework", http://en.wikipedia.org/wiki/Web_Services_Resource_Framework

[76] M. Baldauf, S. Dustdar, F. Rosenberg, "A survey on context-aware systems", International Journal of Ad Hoc and Ubiquitous Computing, 2006

[77] H. Melnikov, Open Solutions for Location Based Services in WLAN Environment, MSC thesis, Tampere University of Technology, 2004

[78] Wikipedia, "Global Positioning System", http://en.wikipedia.org/wiki/Global_Positioning_System

[79] Wikipedia, "Java remote method invocation", http://en.wikipedia.org/wiki/Java_remote_method_invocation

[80] Y. Huang, "JISGA: A Jini-Based Web Service-Oriented Grid Architecture", http://www.wesc.ac.uk/resources/publications/pdf/JISGA.pdf#search=%22JISGA%22

[81] J. Newmarch, "Jan Newmarch's Guide to Jini Technologies", http://jan.netcomp.monash.edu.au/java/jini/tutorial/Jini.xml

[82] jini.org Web site, Community Resource for Jini Technology, http://www.jini.org

[83] jini.org Web site, "How is Jini any different from RMI/CORBA/...?", http://www.jini.org/wiki/How_is_Jini_any_different_from_RMI/CORBA/...%3F

[84] P. Saint-Andre et.al., "Basic Messaging and Presence Interoperability between the Extensible Messaging and Presence Protocol (XMPP) and Session Initiation Protocol (SIP) for Instant Messaging and Presence Leveraging Extensions (SIMPLE)", Internet-Draft , august 2006, http://www.xmpp.org/drafts/draft-saintandre-xmpp-simple-08.html

[85] J. Ferreira et.al, "Overview of the IM and Mobile IM Trial", presentation of Eurescom P1402, 2005, http://www.eurescom.de/~pub/deliverables/documents/P1400-series/P1402/D5/P1402-D5.pdf

[86] C. Moore, "XMPP: Google Talk backs open IM", InfoWorld, 2005, http://www.techworld.com/applications/features/index.cfm?featureid=1745&pagtype=samecatsamechan

[87] W3C, "CC/PP Information page", http://www.w3.org/Mobile/CCPP/

[88] UPnP Forum, UPnP Device Architecture 1.0, Dec. 2003, http://www.upnp.org/resources/documents/CleanUPnPDA101-20031202s.pdf

[89] Wireless Application Forum, WAP-248-UAPROF-20011020-a, Version 20 October 2001, http://www.openmobilealliance.org/tech/affiliates/wap/wap-248-uaprof-20011020-a.pdf

[90] Open Mobile Alliance, "OMA DS Device Information », Approved Version 1.2 – 10, Jul 2006, http://www.openmobilealliance.org/release_program/docs/DS/V1_2-20060710-A/OMA-TS-DS_DevInf-V1_2-20060710-A.pdf

[91] M. H. Butler, "Current Technologies for Device Independence", Publishing Systems and Solutions Laboratory, HP Laboratories Bristol, HPL-2001-83, 2001, http://www.hpl.hp.com/techreports/2001/HPL-2001-83.pdf

[92] J. Barnes, C. Rizos, J. Wang, D. Small, G. Voigt, N. Gambale, "High precision indoor and outdoor positioning using LocataNet", Journal of Global Positioning Systems, Vol.2, No.2, 2003, http://www.gmat.unsw.edu.au/snap/publications/barnes_etal2003f.pdf

5. Mobile Grid Infrastructure Services Layer

In this chapter, we discuss a number of issues that relate to OGSA, including the foundation of Grid infrastructure (messaging, state and resource provision and notification of events), Web Service Management, access from Web Services of to middleware components such as AAA, and Execution Management Services (EMS), which is the essential.

5.1. Convergence plans

The enhancement of Web Services to support an OGSA Grid infrastructure has been the subject of two strands of specification work, which may be referred to as the WSRF strand and the WS-Transfer strand. These cover the areas of resources, events and management. There is potentially a risk of relying on what may turn out to be the strand that doesn't eventually become adopted. The WSRF strand is backed by OASIS and implemented in two widely used implementations and the WS-Transfer strand is backed by highly influential companies.

Since the publication of a joint white paper [94] from Hewlett Packard, IBM, Intel and Microsoft in March 2006 a convergence of Web Services specifications can be seen in these areas. The main specifications that are the subject of this convergence are WSRF, WS-Notification, WSDM on the one side and WS-Transfer, WS-Enumeration, WS-Eventing, WS-Management on the other side.

All involved parties have stated a commitment to make the migration path as smooth as possible. Support for existing implementations is planned for both strands. Features are to extended where missing and enhanced where they exist. A major step forward will be made with an upcoming common management specification for Web-services.

The correspondences are noted in the appropriate sections below (0, 5.2.3 and 5.3).

The Akogrimo prototypes use the WSRF strand of specifications.

5.2. Grid Foundation

5.2.1. Messaging

The status of the relevant messaging specifications to support OGSA has been fairly stable since the Akogrimo State of the Art document [106]. Implementations continue to mature where they exist.

Specification	Maturity	Implementations	Relevance
WS-Addressing	Standard (Aug, 2004)	Several, including WSRF.net, GT4 and Axis-Pubscribe (the latter is poorly documented and not very active)	WS-Addressing is an essential part of the Akogrimo messaging infrastructure. Many other specifications depend on it.
WS-ReliableMessaging	Converging	Several, including WSE and Axis	Not used yet

Specification	Maturity	Implementations	Relevance
and WS-RM Policy		WSE and Axis	

Table 3 - Web Service messaging specifications

Messaging using Web Services are also discussed in section 4.1.4.

5.2.2. State & Resource Provision

The following table summarizes the current state of the specifications and the relevance for the project.

Specification	Maturity	Implementations	Relevance
WS-ResourceProperties	Standard (April 2006)	Several, including WSRF.net and GT4	Essential specification in order to provide statefulness.
WS-ResourceLifetime	Standard (April 2006)	Several, including WSRF.net and GT4	Used
WS-ServiceGroup	Standard (April 2006)	Limited support	Not used yet
WS-BaseFault	Standard (April 2006)	Limited support	Not used yet
WS-Transfer	Update Submitted to W3C (September 2006)	Experimental (e.g. by Roman Kiss [100])	Not used
WS-ResourceTransfer	Public (August 2006)	Experimental	Not used
WS-Enumeration	Submitted to W3C (March 2006)	Experimental (e.g. Globus Toolkit 4.2 [95], The Wiseman Project [101])	Not used

Table 4 - Web Service resource and state specifications

WS-ResourceProperties has successfully been used in a prototype implementations in Akogrimo. Interoperability between GT4 [95] and WSRF.net [96] has been achieved for the required functionality.

The convergence plans for Resource components as outlined in the convergence white paper [94] are in the table below.

Resource	
WS-ResourceFramework: • WS-Resource • WS-ResourceProperties • WS-ResourceLifetime • WS-ServiceGroup • WS-BaseFaults	WS-Transfer (updated) WS-ResourceTransfer (new) WS-MetaDataExchange 1.1 (new) WS-Enumeration

Table 5 - Converging Web Service resource and state specifications

5.2.3. Notification of events

The status of the relevant notification specifications has been fairly stable since the last version of the Akogrimo State of the Art document [106]. Implementations continue to mature where they exist, but are still missing important parts, e.g. notification topics are not supported yet and the same is true for brokered notifications. With WS-BaseNotification the vital notification mechanism can be implemented, but large scale implementation would profit from the functionality and semantic richness of the aforementioned items.

Specification	Maturity	Implementations	Relevance
WS-BaseNotification	Standard (Oct, 2006)	Several, including WSRF.net, GT4 and Axis-Pubscribe	Essential
WS-BrokeredNotification	Standard (Oct, 2006)	Limited Support	Not used yet
WS-Topics	Standard (Oct, 2006)	Limited Support	Not used yet
WS-Eventing	Submitted to W3C (March 2006)	Existing (e.g. by Plumbwork Orange [97], Roman Kiss [99] and the OMII [98])	Not used

Table 6 - Web Service specifications for notification of events

The convergence plans for notification/events components as outlined in the convergence white paper [94] are summarised in the table below.

Events	
WS-Notification: • WS-BaseNotification • WS-BrokeredNotification • WS-Topics	WS-Eventing • WS -EventNotification (new) This makes use of WS-ResourceTransfer (new)

Table 7 - Converging Web Service specifications for notification of events

Note that the white paper uses WS-EventNotification and WS-EventingNotification interchangably. From an investigation of other online material, it would appear that the term WS-EventNotification is intended.

5.2.4. Assessment summary

Section 5.1 on Convergence plans discusses the convergence plans which hold the prospect of bringing together the 2 major strands and hence reducing the risk of eventual non-acceptance. In addition, the WSRF strand is available in widely used implementations which means that lessons learned elsewhere can be made use of here.

Ideally interoperability problems are reduced if only one implementation is used. However some involved partners have existing experience of GT4 and others WSRF.NET. Interoperability problems have been identified (for instance, differing interpretations of WSDL) which requires attention in the development of services that make use of these implementations. Further interoperability complications are eliminated by restricting to these two implementations.

5.3. Manageability

In the State of the Art document [106], WS-DistributedManagement (WSDM) was chosen as the support technology for manageability. The main characteristics that were evaluated in the decision process are:

- Maturity: WSDM is a standard and is usually a guarantee of less risk.
- Existing implementations: With Apache Muse there was only an existing implementation for WSDM.
- Technical position: No interoperability problems because WSDM is using the same WS specifications like GT4.

At the moment the status of the relevant manageability specifications is still the same. The only difference is that with the Wiseman project [101] there is a WS-Management implementation available now. The Wiseman project is an implementation of the WS-Management specification for the Java SE platform. The project scope includes the WS-Management specification and also its dependent specifications WS-Addressing, WS-Enumeration, WS-Eventing and WS-Transfer. Nevertheless, there are still interoperability problems with GT4 e.g. GT4 is using WS-BaseNotification and not WS-Eventing. The announcement of the upcoming management specification for Web-services by Hewlett Packard, IBM, Intel and Microsoft ([94]) is no threat because of the dedicated support for existing specification. It seems likely that there will be support for an eventual migration path.

Specification	Maturity	Implementations	Relevance
WS-Distributed-Management	Standard (March 2005)	Apache MUSE	Basic features used.
WS-Management	Submission to DMTF (April 2006)	Experimental, Wiseman project	Not used

Table 8 - Web Service Management specifications

The convergence plans for notification/events components as outlined in the convergence white paper [94] are summarised in the table below.

Management	
WSDM • MUWS • MOWS	WS-Management This makes use of: • WS-EventNotification (new) • WS-ResourceTransfer (new)

Table 9 - Converging Web Service Management specifications

5.4. AAA in a Web-Services Environment

Diameter, which was assessed in section 4.5.2, provides AAA functionalities in a networking environment and it was designed mainly with the focus of network access services but with the possibility of flexible extensibility. Web-Services – and, through WSRF or WS-Transfer, OGSA Grid Services - enable flexible service provisioning in a common way and require AAA functionalities in a commercial environment in order to authenticate users, authorize service access and enable charging based on accounting information related to resource and service usage. Therefore, the provisioning of AAA functionalities based on Diameter and its integration in a Web-Services environment seems straightforward but also needs some further considerations, which can be summarized as the following:

- Provisioning of AAA functionalities based on Diameter
- Provisioning of AAA functionalities as a Web-Service supported by Diameter in the back-end
- Provisioning of AAA functionalities based on other technologies like the eXtensible Access Control Markup Language (XACML) without Diameter
- Performance
- Security

According to that list, there are three general ways to integrate AAA functionalities into a Web-Services environment. First, AAA can be fully based on Diameter. In this case, Diameter has to be integrated into each component requiring AAA functions and the communication between network components is fully based on Diameter. This is opposite to the concept of Web-Services where each service is provided via SOAP as a Web-Service. However, AAA is considered as a network operational function and not a generic service that needs to be accessed from any entity

and component, but only from a predefined set of service provisioning components in the network. Additionally, this solution provides high performance, as Diameter was designed and optimized also regarding performance.

Second, AAA functions can be provided without the use of Diameter. However, this would mean additional risks and require further considerations to integrate new technologies, as discussed in section 4.5.2. Additionally, component included in a Grid infrastructure and traditionally having a Diameter based interface (e.g. network layer components) would require also new interfaces.

Third, AAA functions might be provided also as a Web-Service that can be accessed by different components via a common Web-Service interface based on SOAP. In this case, a mixed solution is required where AAA is based on Diameter but provided also via SOAP to Web-Services components and via Diameter to other components. This would require a protocol translator Diameter-SOAP Gateway that converts between Diameter and SOAP and enables Web-Services components to communicate with the A4C server. The integration of a Diameter-SOAP Gateway requires further considerations and means additional risks. The smooth integration of the two protocols requires a careful analysis to keep protocol state consistent in the gateway. Using a Diameter-SOAP Gateway means also performance degradation and additional overhead because of the performance issues of SOAP and the permanent translation between the two protocols. Finally, while Diameter provides means for inter-domain communication and security, the Diameter-SOAP integration requires additional analysis in these topics as well.

5.5. Execution Management Services

Functionality: The Akogrimo Execution Management Service (EMS) based on the OGSA and WSRF specifications comprises the central controller of the business service execution. The importance of EMS stems from the dynamic computing environments, since Grids are expected to be used in a great number of settings, with both the available resources and the load the latter are faced with being extremely variable. In such environments monitoring of the application execution is required so that the fulfillment of the service level agreements is achieved and unexpected failures are dealt with through resource reallocation and restart of service execution. The main functionalities of Akogrimo EMS focus on finding execution candidate locations, making advance reservation on selected resources, preparing, initiating and managing/monitoring the execution of a business service.

Implementation: The GT4, the platform upon which the Akogrimo EMS was developed, includes many high-level services, developed to take advantage of the potentials that the toolkit offers to the fullest. Among the advantages of this approach is the one of advanced functionality and flexibility as well as the encouragement of service reuse. Depending on the explicit nature of the EMS, we have decided to leverage the GT4's *WS-GRAM* and *MDS4* functionalities into it, using Java WS Core, and in this way create a much more powerful and flexible version of the EMS.

WS-GRAM is a suite of Web services provided by the GT4 used for submission, monitoring and cancelling jobs on local or remote computing resources. GT4 provides both a WS-GRAM command line client and Java, C and Python client APIs. Through the usage of the WS-GRAM Client Java API and Java WS Core, the WS-GRAM functionality has been successfully integrated into the EMS.

MDS4 is a WSRF implementation of information services released with GT4. MDS4 builds on query, subscription, and notification protocols and interfaces defined by the WS Resource

Framework (WSRF) and WS-Notification families of specifications and implemented by the GT4 Web Services Core. Building on this base, we have successfully integrated the functionality of the Index service, one of the two high-level services that the MDS4 provides, into the EMS and use it for discovering purposes.

The integration of high-level GT4 services into EMS has made it much more powerful but at the same time it has introduced limitations and restrictions on the overall Akogrimo environment:

- In order for EMS to be fully functional, some basic setup, mostly related to WS-GRAM and MDS4 is needed. A full GT4 installation is required on each machine hosting the EMS so that the additional features of the EMS related to the WS-GRAM and MDS4 will be enabled and fully functional.
- A Java core-only installation will not be enough for the machines that host the business services as it does not include the GT4 Index service necessary for the discovery process.
- In order to execute the business service through WS-GRAM, a full GT4 installation is strictly required on the machines that are hosting them. Each of these machines must be configured so as to trust the CA that issued the certificate that the GT4 installation on the machine hosting EMS is configured to work with.
- EMS includes a recovery mechanism that comes in use in case the execution of a business service fails. In order for this mechanism to be functional, each business service that is offered to the clients through Akogrimo, apart from its basic functionality, it should be developed in such a way as to provide EMS with the necessary information.

EMS still functions even when the above criteria are not met but much of its high-level functionality is disabled.

5.5.1. Assessment of Risks

Fault tolerance/Recovery mechanism

EMS includes many interactions with various services, developed on different platforms. This high level of dependency on other services, apart from the obvious overhead in the execution time of the EMS itself, increases the possibility of failure. The recovery mechanism of the EMS should be extended so as to be able to deal not only with failures that are directly related to the execution of the actual business services but with the whole EMS supporting subsystem.

GT4 bugs

Globus Toolkit is an experimental platform under development. Unavoidably, some of its tools/features have bugs that prevent them from functioning correctly or have poor performance that affects the performance of EMS. New versions of the GT4 toolkit that guarantee backwards compatibility are released on a regular basis. EMS has already migrated to the latest stable version of GT4 successfully.

Security vs performance/Interoperability issues

Since EMS acts as the central controller of the Grid layer services, it is very critical that only the authorized users actually access the resources provided by the Grid. On the other hand, security enforcement affects dramatically the performance at runtime. Since each business service execution is controlled by EMS, applying security will slow it down and consequently will affect the overall performance of the system. The challenge is to protect the EMS system from poten-

tial exploits while at the same time optimize the system performance. For this reason, appropriate security mechanisms must be applied only when and where it is critical.

EMS interacts with services developed both on GT4 and WSRF.NET platforms. EMS will make use of GT4's Grid Security Infrastructure (GSI), a set of tools, libraries and protocols for enforcing security. It is critical to test interoperability between the different security mechanisms offered by the toolkits. Interoperability issues may lead to a degradation of the overall security in terms of performance.

5.6. References for Mobile Grid Infrastructure Services Layer

[93] J. Treadwell, Open Grid Services Architecture Glossary of Terms, GFD-I.044, http://www.gridforum.org/documents/GFD.44.pdf

[94] *Toward Converging Web Service Standards for Resources, Events, and Management*, A Joint White Paper from Hewlett Packard Corporation, IBM Corporation, Intel Corporation and Microsoft Corporation, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwebsrv/html/convergence.asp

[95] The Globus Toolkit, http://www.globus.org/toolkit/

[96] WSRF.net, University of Virginia Grid Computing Group, http://www.cs.virginia.edu/~gsw2c/wsrf.net.html

[97] Plumbwork Orange, http://sourceforge.net/projects/plumbworkorange/

[98] WS-Eventing Implementation, open middleware infrastructure institute http://www.omii.ac.uk/downloads/omii-distributions.jsp

[99] Roman Kiss, WS-Eventing Implementation, http://www.codeproject.com/useritems/WSEventing.asp

[100] Roman Kiss, WS-Transfer Implementation, http://www.codeproject.com/soap/WSTransfer.asp

[101] The Wiseman Project, https://wiseman.dev.java.net/

6. Mobile Grid Application Support Services Layer

6.1. Introduction

Today, grid application support frameworks have the aim of providing a layer of abstraction on top of the underlying grid infrastructure and of providing an easy, comprehensible programming model with minimal possible extensions. They should make it possible to manage easy tasks such as file transfers, finding, and picking up the right resources on the basis of QoS parameters and job execution.

In Akogrimo, the Grid application support services layer is implemented strictly within the vision of the Web Services Architecture (according with the last evolution in the frame of Grid research that led to the introduction of WSRF). Such an application framework is built on Web Services related specifications, can coexist with other Web Services specifications, and the aim is to leverage on existing tools for Web Service (and WS-Resource) development.

Interoperability at the underlying message level is a general risk for the Mobile Grid Application Support Services Layer and, in particular, it is related to the SOAP message exchange between different services implemented on different platform. These interoperability issues are covered in sections 4.1.4 (on Web Services) and 5.2.

Having chosen this as a foundation, the following sections mainly will describe technologies and specifications coming from the Web Service world, in particular, focussing on aspects related to SLA definition, service composition and security.

6.2. Service composition and workflow management

Often, Web Services need to be put together using pre-defined scripts or orchestration scripts, containing messages, branching logic and invocation sequences. Web Services orchestration is about providing an open, standards-based approach for connecting Web Services together to create higher-level business processes. Standards such as BPEL4WS and BPML are designed to reduce the complexity required to orchestrate Web Services, thereby reducing time-to-market and costs, and increasing the overall efficiency and accuracy of business processes. Without a common set of standards, each organization is left to build their own set of proprietary business protocols, leaving little flexibility for true Web Services collaboration.

The Business Process Execution Language (BPEL) is a powerful protocol that allows to design business processes and execute them by using existing Web Services. It makes use of the underlying technologies (XML and TCP/IP) to provide the capability to:

1. Have standardized communication between different applications
2. Avoid the problems with the underlying operating system.

Furthermore security can be integrated in business process using protocols related to the security of communication between Web Services (WS-Security). Then the use of BPEL has many advantages as has been explained in the Akogrimo State of the Art [106]. Here we focus only on some disadvantages which represent risks to be identified and evaluated:

- **Complex business processes**:

In an extremely complicated business process, BPEL would be not enough to ensure a business process successful management. In fact, BPEL just coordinates the flow of Web Service invocation.

In Akogrimo, business processes are multiparty with a lot of critical interactions that could require specific state actions and with the possibility of failed work flow branches. In order to solve this issue BPEL could be complemented with Web Service Choreography Description Language (WS-CDL) that has been designed to address, among others, this issue and is passing through the W3C approval process. Alternatively, BPEL could be complemented with the introduction of external services (external to the BPEL engine) that have a control on the workflow execution and can control some critical situations that otherwise would be managed inside the BPEL engine or worse still would not be managed at all.

We can summarize that using BPEL in Akogrimo can be a risk due to the potential complexity of Akogrimo related business process. However this risk is low because, even if WS-CDL is used, BPEL would also be required as a complementary technology.

- **BPEL orchestrates only Web Services**:

BPEL is an XML based language to describe a business process that are based on orchestration of Web Services, and it is excellent to address this goal, but BPEL is not a programming language. In some cases, use of code (e.g. Java or .NET) in the business process description could be useful (e.g. variable initialization, loop condition and so on). Currently some existing BPEL engines (e.g. BizTalk,…) provide this capability but the resulting business process description is not standard. In this case, the risk is that BPEL could not be expressive enough to describe Akogrimo related business processes. The risk is low because extensions (supposed to be approved in future) have been proposed to solve this problem (e.g. BPELJ)

- **Legacy application**:

The use of BPEL in Akogrimo will require integrating legacy application converting them in a design that is compliant with BPEL requirements (mainly this means to migrate the legacy application towards a SOA based on Web Services). The risk can be high if the cost of migration is high. However this risk is also associated with other business process solutions based on Web Services and is therefore a risk that needs to be accepted when moving to this technology

- **Security**:

BPEL implies using WS-Security otherwise the infrastructure is under risk of security attacks. This means that Akogrimo platform has to use WS-Security protocols and it has to be integrated with network related security mechanism. The risk is related to the difficulty of integrating different security approaches/solutions.

6.2.1. Assessment summary

Although BPEL carries certain risks, it is a key technology for expressing business processes in a Web Services environment and, as it can be used in conjunction with more advanced specifications that may be used in future, there is a migration path.

6.3. Security: Web Services approach to Authentication & Authorization

6.3.1. Security WS-* specifications (WS-Security, WS-Trust, WS-SecureConversation)

WS-* specifications identify a set of protocols, based on XML, which allow enhancing Web Services features. From a Web Service security viewpoint, WS-* includes:

- **WS-Security**, which describes enhancements to SOAP messaging to provide quality of protection through message integrity, message confidentiality, and message authentication. WS-Security is designed to support multiple security token formats and to provide a general-purpose mechanism for associating security tokens with messages.
- **WS-Trust**, which defines a Web security model to establish a trust relation between client and server; this relation is important because the two parties must exchange security credentials (either directly or indirectly) and each party needs to determine if they can "trust" the asserted credentials of the other party.
- **WS-SecureConversation**, which defines mechanisms for establishing and sharing security contexts, and deriving keys from security contexts, to enable a secure conversation.

These specifications allow defining trust relations among Web Services, creating secure communication channels to protect exchanged SOAP message and identifying a message sender. However, each of them does not provide a complete solution for Web Services, but is a building block that can be used in conjunction with other specifications to accommodate a wide variety of security solutions.

We can use WS-* specifications in order to address Akogrimo security requirements at message level but we have to pay attention to the following issues:

- **message confidentiality:** WS security specifications allow encrypting only the *content* of the SOAP message, while the rest of the SOAP message is left unencrypted. If we need to encrypt all the information exchanged between the client and the server, then we need to evaluate the possibility to apply security at transport level.
- **poor message processing performance:** Message-level security in Web Services offers more features than transport-level security (e.g. a better integration with Web Services standards), but its performance still leaves a bit to be desired. In fact, a SOAP engine adds a lot of time overhead to the elaboration of SOAP messages because sign validation and encryption/decryption are time consuming tasks. The performance consideration is often a topic of concern and prevents their wider adoption to protect SOAP messages. The overall effect on the performance of a total system can be reduced by appropriate choice of messages, which are to be passed using WS-Services and SOAP; any messages which are not required at a Web Services level can be secured using transport-level security.
- **use of .NET framework and Java Runtime Environment:** In Akogrimo some components are coded in C#, while others are coded in JAVA and from the security viewpoint we need a SOAP engine in C# and another one in JAVA. In order to guarantee interoperability, the SOAP engines have to implement the same security specification versions, otherwise they will not able to elaborate correctly security headers. To solve this issue, we need to agree on a specific version of a SOAP engine for .NET and on another one for JAVA. Then the selected version becomes a mandatory software requirement for each machine involved in Akogrimo environment.

6.3.2. X.509 Certificate and VO implications

The X.509 certificate, described in RFC 3280 [102], is based on a PKI infrastructure. It specifies a binding between a public key and a set of attributes (e.g. subject name, issuer name, serial number, validity interval). The X.509 v3 certificate format includes some extensions, which provide a method for associating additional attributes with users or public keys. In particular, it is possible to define private extensions to carry information valid inside a community.

In Akogrimo, each member could prove their identity using a X.509 certificate, which can be issued by a commercial CA, by the Akogrimo CA or can be self-signed by the member.

It is important to choose the best method for certification generation and technical problems should be considered:

- **commercial CA:** It is the most convenient choice because it delegates certificate generation to a commercial entity, but an Akogrimo member has to pay to obtain this certificate.
- **Akogrimo CA:** In this case we have to face with all technical issues in order to make it available inside the Akogrimo environment (installation and configuration).
- **Self-signed:** Akogrimo member has to use a public tool to generate his certificate.

In Akogrimo we can take advantages of certificate format extensibility; in fact it is a convenient way to include in the certificate custom information suitable for BaseVO and OpVO (such as an identifier of a BaseVO or OpVO), but caution ought to be exercised in adopting any critical extensions in certificates, which might prevent use in a general context.

6.4. Service Level Agreement (SLA)

WS-Agreement (WS-A) specifies an XML-based language for creating contracts, agreement and guarantees from offers between a service provider and a customer. The strength of WS-Agreement lies in a well-defined template for specifying agreements. The template or part of the template, such as the service description terms and the guarantee terms, can be used in the content of exchanged messages. Moreover, generally speaking, this template is suitable in cases where interactions are concerned with reaching agreements and drawing up contracts.

Anyway it presents some shortcomings and some of them could be relevant in Akogrimo environment. In particular WS-A only supports "offer and agree" messages without an interaction protocol.

The WS-Agreement specification is only used at the last stage in a contractual conversation where the parties are closing their interaction with a contract specified as a WS-Agreement. It doesn't support a real negotiation because there is only a two steps conversation: offer the SLA followed by agreeing the proposed SLA. Of course this approach is not appropriate for modeling Akogrimo negotiations that should support an interaction between the parties (customer and provider). In order to address this weakness, the use of a negotiation process is required and initial consideration of WS-AgreementNegotiation has taken place, in order to eventually follow the approach proposed in the frame of Grid Resource Allocation Agreement Protocol Working Group (GRAAP-WG) of the Open Grid Forum. This should reduce the risk of deprecation related to the adoption of specification/standards that have a slow evolution and that are still at very initial stages of development, which applies to contract and SLA standards for web services, because currently the focus is on consolidation of the lower levels of the web services standards stack.

6.5. WSE Versions 2.0 vs 3.0

Web Services Enhancements for Microsoft .NET (WSE) provides developers with the latest WS-* specification implementation in order to allow keeping pace with the evolution of Web Services protocol specifications.

In Akogrimo WSE can be used to:

- protect SOAP messages, from unauthorized users, using digital signatures and encryption
- establish trust relations
- create secure communication channel between two components
- route messages through intermediaries

WSE is provided as an add-on to Microsoft Visual Studio .NET and to Microsoft .NET Framework. Currently, WSE 2.0[4] and WSE 3.0 versions are commonly used, but WSE 2.0 is not wire-level compatible with WSE 3.0 due to changes in a number of specifications, such as WS-Addressing, WS-SecureConversation, and WS-Security.

The main concerns related to the use of WSE 2.0 or WSE 3.0 are:

- **Interoperability**

This is probably one of the main concerns for people involved in the development of web services using WSE. Unfortunately, WSE 3.0 was designed from the beginning to be compatible at messages level with Indigo and therefore it doesn't interoperate well with WSE 2.0.

- **WS-Security specification**

At this moment, there are two available versions of this specification, 1.0 and 1.1 (also called WS-Security extensions). WSE 2.0 only implements the first version whereas WSE 3.0 uses features of both versions (such as signature confirmation and key derivation).

- **WS-Secure conversation specification**

Using this specification the client and server can negotiate a session token to protect the communication for a specific period of time. This feature decreases the response time because the token negotiation happens once compared to other turn-key scenarios where the negotiation is done for each message. The SecureContext token used in WSE 3.0 is not compatible with WSE 2.0.

- **WS-Addressing specification**

WSE 3.0 uses a newer version of this specification (the same as Indigo) and therefore the messages produced by both versions are not compatible.

- **Algorithm suite**

They use different algorithms for signing and encryption. WSE 3.0 uses AES256 for symmetric encryption and RSA-OAEP for key wrap. WSE 2.0 uses AES128 and RSA-15.

- **SOAP messages**

[4] WSE 2.0 SP3 is supported on both .NET Framework 1.1 and on .NET Framework 2.0.

The SOAP messages generated using WSE 3.0 will be compliant with the latest WS-* specification. On the Java side, a SOAP engine that implements the same specifications version has to be used, otherwise the SOAP engine will not able to understand some SOAP message security headers, which will be skipped.

6.5.1. Assessment summary

From Akogrimo implementation viewpoint, to choose the WSE version we have to take into account:

- The risk of the migration phase for all Akogrimo components already developed using .NET framework (in term of effort cost) and
- The interoperability problems with other components implemented using Globus toolkit core. Indeed interoperability tests have been already performed between WSRF.NET 2.x (based on WSE 2.0) and Globus toolkit core frameworks and solution have been already found.

Currently, Akogrimo services developed on Microsoft platform are designed on the top of WSRF.NET 2.1.0, which uses .NET framework 1.1 and WSE 2.0. To use WSE 3.0, which works on .NET framework 2.0, we have to migrate those Akogrimo components on WSRF.NET 3.0 framework, which does not guarantee backward compatibility.

This has to be balanced against the more advanced security specifications in WSE3.0 and it will need to be decided how necessary these are for the 2nd cycle.

6.6. References for Grid Application Support Services Layer

[102] "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC3280, 2002, http://www.ietf.org/rfc/rfc3280.txt

[103] Towards Open Grid Services Architecture (OGSA), GGF, http://www.globus.org/ogsa/

[104] The Open Grid Services Architecture, Version 1.0, January 2005, http://www.gridforum.org/documents/GWD-I-E/GFD-I.030.pdf

[105] I.Foster, C.Kesselman, J.M.Nick, S.Tuecke, The physiology of the grid: An open grid services architecture for distributed systems integration, http://www.globus.org/research/papers/ogsa.pdf

7. Conclusion

This report provides information and assessment of technological alternatives for the many specific aspects of the Akogrimo project at a range of network layers. In all cases it is possible to identify an approach where the technological risks can be identified and understood.

8. Abbreviations and terms

8.1. Abbreviations

2G	Second-generation wireless telephone technology.
3G	Third-generation mobile telephone technology
AAA	Authentication, Authorisation and Accounting
Akogrimo	Access To Knowledge through the Grid in a Mobile World
CIM	Common Information Model
CN	Correspondent Node
CoA	Care-of Address
COPS	Common Open Policy Service
DCF	Distributed Control Function
DMTF	Distributed Management Task Force
EAP	Extensible Authentication Protocol
EDCF	Enhanced Distributed Control Function
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HA	Home Agent
HoA	Home Address
HCF	Hybrid Control Function
HIP	Host Identity Protocol
IPSec	IP Security
LAN	Local Area Network
MAN	Metropolitan Area Network
MDVO	Mobile Dynamic Virtual Organisation
MIPv6	Mobile IP version 6
MN	Mobile Node

nrtPS	Non-real-time Polling Service
OASIS	Organization for the Advancement of Structured Information Standards
OGSA	Open Grid Service Architecture
PAN	Personal Area Network
PANA	Protocol for carrying Authentication for Network Access
PBNM	Policy-based Network Management
PCF	Point Control Function
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PPTP	Point-to-point tunnelling Protocol
RFC	(Internet IETF) Request For Comment
rtPS	Real-time Polling Service
SDS	Service Discovery Service
SIP	Session Initiation Protocol
SNMP	Simple Network Management Protocol
SSL	Secure Socket Layer
TLS	Transport Layer Security
UGS	Unsolicited Grant Service
UMTS	Universal Mobile Telecommunications System
VoIP	Voice over IP
VPN	Virtual Private Network
WEP	Wired Equivalent Privacy
WiFi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WPA	WiFi Protected Access
WSDL	Web Service Description Language

8.2. Terms

This section provides definitions of major terms as used in this document.

Sources used for definitions of some terms include:

- Webopedia: Online Computer Dictionary for Computer and Internet Terms and Definitions [107]
- W3C Web Services Glossary [109]
- Terms and Definitions Database Interactive (TEDDI). [110]
- Yourdictionary.com [111]
- Ask Oxford [112]
- RFCs, which (using the example RFC3334) are accessed using a URL of the form http://www.faqs.org/rfcs/rfc3334.html or http://www.ietf.org/rfc/rfc3334.txt

If consulting this document online, this letter list can be used to navigate alphabetically: [A] [B] [C] [D] [E] [F] [G] [H] [I] [J] [K] [L] [M] [N] [O] [P] [Q] [R] [S] [T] [U] [V] [W]

[A]

Term:	Accounting
Definition:	Accounting describes the collection of data about resource consumption. This includes the control of data gathering (via metering), transport and storage of such data. Comment: This definition is used in a networking context, which need not preclude its use in other contexts.
Source:	RFC 3334

Term:	Authentication, Authorisation and Accounting (AAA)
Definition:	Mechanism for identifying users, authorising or denying their access to specific resources and accounting for their usage.
Source:	RFC2903

[B]

Term:	Billing

Definition:	Billing translates costs calculated by a charging scheme into monetary units and generates a final bill for the customer. Comment: This definition is used in a networking context, which need not preclude its use in other contexts.
Source:	RFC 3334

Term:	Business Process Execution Language for Web Services (BPEL4WS)	
Definition:	"BPEL4WS explicitly allows the use of nondeterministic data values to make it possible to capture the essence of public behaviour while hiding private aspects. […] It is also possible to use BPEL4WS to define an executable business process. The logic and state of the process determine the nature and sequence of the Web Service interactions conducted at each business partner, and thus the interaction protocols.". Comment: This definition is used by the official development project consortium of BPEL4WS consisting of BEA, SAP, Microsoft, IBM, Siebel;	
Source:	http://www.oasis-open.org/committees/download.php/4413/wsbpel-specification-draft-Nov2303.htm	

[C]

Term:	Charging
Definition:	Charging derives non-monetary costs for accounting data sets based on service and customer specific tariff parameters. A charging scheme is an instruction for calculating a charge and is usually represented by a formula that consists of charging variables (e.g. volume, time, reserved peak rate) and charging coefficients (e.g. price per time unit). The charging variables are usually filled by information from accounting data. Charging policies define the tariffs and parameters which are applied. Comment: This definition is used in a networking context, which need not preclude its use in other contexts.
Source:	RFC 3334

Term:	Common Information Model (CIM)
Definition:	The CIM is a conceptual information model for describing managed entities, their composition and their relationships.
Source:	[9]

[D]

Term:	Distributed Management Task Force (DMTF)

Definition:	The Distributed Management Task Force, Inc. (DMTF) is the industry organization leading the development of management standards and integration technology for enterprise and Internet environments. DMTF standards provide common management infrastructure components for instrumentation, control and communication in a platform-independent and technology neutral way. DMTF technologies include information models (CIM), communication/control protocols (WBEM), and core management services/utilities.
Source:	http://www.dmtf.org/about

Term:	Dynamic Virtual Organisation (DVO)
Definition:	The "dynamic nature" implies that the entire set up of a virtual organization may change in response to the market place. In this sense, virtual organizations of this type are temporary as to their ability to react quickly as regards the membership, the structure, the objectives etc. Its fluid boundaries and opportunism, as well as equity of partners and shared leadership mainly characterize a dynamic virtual organization
Source:	Literature http://www.vtt.fi/

[E]

[F]

[G]

Term:	Grid Service
Definition:	A Grid Service is a Web Service that follows specific conventions to do with address discovery, dynamic service creation, lifetime management, notification and manageability. In the OGSA Glossary, "In its more general use, a Grid service is a *Web service* that is designed to operate in a *Grid* environment, and meets the requirements of the Grid(s) in which it participates."
Source:	OGSA Glossary of Terms [93]

[H]

[I]

Term:	Identity
Definition:	"The collective aspect of the set of characteristics by which a thing is definitively recognizable or known"
Source:	Yourdictionary

Term:	IPv6
Definition:	"IP version 6 is the internet protocol which is intended to replace the previous standard IPv4. The major changes involved are the following: expanded addressing capabilities, header format simplification, improved support for extensions and options, flow labelling capability, and authentication and privacy capabilities."
Source:	Based on RFC 2460

Term:	Inter-domain Mobility
Definition:	"The ability to transparently use resources of different administration domains while holding a single contract with only one of them."
Source:	Akogrimo State of the Art [106]

[J]

[K]

[L]

Term:	Local Area Network (LAN)
Definition:	"A network that spans a relatively small area."
Source:	Webopedia [107]

[M]

Term:	Meter, metering
Definition:	In a network, a meter is responsible for measuring traffic; it observes packets as they pass by a single point on their way through a network and classifies them into certain groups, this process being referred to as metering. Comment: This definition is used in a networking context, which need not preclude its use in other contexts.
Source:	Based on RFC3334 and RFC 2722.

Term:	Metropolitan Area Network (MAN)
Definition:	"A large network usually spanning a campus or a city."
Source:	Based on www.wikipedia.com

Term:	MIPv6

Definition:	"Is a protocol which allows nodes to remain reachable while moving around in the IPv6 Internet."
Source:	RFC 3775

Term:	Mobile, Mobility
Definition:	In common usage, mobile or mobility refers to the ability to move. A mobile phone, by contrast with a domestic phone, can be used wherever its bearer is located, subject to restrictions of technology and social obligation. In Akogrimo, the idea of mobility is extended to any device that enables a user to stay connected while on the move, to services on a Grid, and to the Grid notion of Virtual Organisation (VO). It also includes the idea that a user can be mobile by moving from one device to another, while retaining the continuity of engagement with ongoing services.
Source:	Akogrimo State of the Art [106]

Term:	Mobile Dynamic Virtual Organisation (MDVO)
Definition:	"A Dynamic Virtual Organisation with at least one essential entity (in Akogrimo's case typically an Application User) that is not bound to a location but can move so that mobility aspects like contextuality and personalization become important". In addition it is a goal of Akogrimo that mobility will extend to the services not only the users.
Source:	Akogrimo State of the Art [106]

[N]

Term:	Nomadic
Definition:	A more limited form of mobility where the user or service may disengage from the network for a period and re-engage when a suitable point of access is available, possibly at a different location. Continuity of service is generally required. Another definition, "Nomadic computing is a technology allowing anyone to leave their office and still have seamless access to the same set of network services as they had at their office, wherever they go with whatever device they're carrying, regardless of the environment they enter."
Source:	[114]

[O]

Term:	Open Grid Services Architecture (OGSA)

Definition:	Historically, the notion of a networked Grid began with an idea which was initially defined in terms of a specific implementation, but matured into a publicly defined architecture, the Open Grid Services Architecture (OGSA). In the OGSA document is the following definition (rephrased): the Open Grid Services Architecture (OGSA) addresses the need for standardization by defining a set of core capabilities and behaviours that address key concerns in Grid systems.
Source:	A summary of the OGSA activity in the GGF [103] and the OGSA document itself [104]

Term:	Open Grid Services Infrastructure (OGSI)
Definition:	OGSI refers to the first base infrastructure on which OGSA has been built, now being replaced by WSRF. It defines the standard interfaces and behaviours of a Grid service, building on a Web Services base.
Source:	A summary of the OGSA activity in the GGF, this also provides a link to the current OGSI document itself [103]

Term:	Operator
Definition:	Within a discussion concerned with mobile access to a computer network, an Operator is in generally the provider of a mobile network, mobile network operator, and there is a responsibility and motivation for mobile networks to interwork.
Source:	Wikipedia [108]

Term:	Organisation for the Advancement of Structured Information Standards (OASIS)
Definition:	It is a not-for-profit, international consortium that drives the development, convergence, and adoption of e-business standards.
Source:	The OASIS "about" web page http://www.oasis-open.org/who/

[P]

Term:	Policy Based Network Management (PBNM)
Definition:	"The management of complex networks by means of a set of rules which are followed by the devices controlling the network configuration."
Source:	Akogrimo State of the Art [106]

[Q]

Term:	Quality-of-Service (QoS)
Definition:	"Quality of Service is defined in CCITT Recommendation E.800 as follows: "The collective effect of service performances which determine the degree of satisfaction of a user of the service." For a given service, QoS is a statement of the performance of the service as offered or specified to the customer. It is defined and measured in terms of parameters which are stated in user understandable language appropriate to the particular service concerned, and which are user verifiable. These parameters will depend upon the service definition, and upon the point at which the service is accessed by the user"
Source:	Literature TEDDI

[R]

[S]

Term:	Service
Definition:	(1) In the context of a network layer, this is the provision of a specific function to a customer connected to the network. (2) In a Grid context, it is specifically a Grid Service or a Web Service.
Source:	For (2), sources are given under Grid Service and Web Service.

Term:	Service Level Agreement (SLA)
Definition:	An SLA is an agreement between the provider and consumer of a service. In the Web and Grid Service world this may result from a negotiation between a provider and consumer and subsequently needs to be monitored for enforcement and accounting purposes. From the OGSA Glossary, "A contract between a provider and a user that specifies the level of service that is expected during the term of the contract."
Source:	OGSA Glossary of Terms [93]

Term:	Session Initiation Protocol (SIP)
Definition:	Session Initiation Protocol is being standardized by the IETF and is a protocol oriented to establish multimedia communication services over IP networks. SIP allows users to call each other independently of their location.
Source:	[18]

[T]

Term:	Terminal Mobility
Definition:	"The ability of a terminal to freely change its location while maintaining alive the communications already established with other entities." Or "In commercial wireless networks, the ability of a terminal, while in motion, to access telecommunication services from different locations, and the capability of the network to identify and locate that terminal"
Source:	US Federal Standard Glossary of Telecommunication Terms [113]

[U]

[V]

Term:	Virtual Organisation (VO)
Definition:	« A network of organisations and/or individuals, with a commonality of purpose or interest, which collectively make up an identifiable, coherent, entity »
Source:	Now commonly used within Grids, an early use of this is in [105].

Term:	Virtual Private Network (VPN)
Definition:	"A VPN is a private network which communicates over a public network. Secure VPNs use cryptographic tunnelling protocols to provide confidentiality, sender authentication and message integrity."
Source:	Based on Wikipedia [108]

[W]

Term:	Web Service
Definition:	The World Wide Web is more and more used for communication between applications. The programmatic interfaces made available are referred to as *Web services*. A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards. Comment: Note that the preceding the definition in the Web Services Architecture document, it says: For the purpose of this Working Group and this architecture, and without prejudice toward other definitions, we will use the following definition.
Source:	Web Services Architecture document at W3C http://www.w3.org/TR/ws-arch/ .

Term:	Web Service Description Language (WSDL)
Definition:	WSDL is a language for describing Web Services. Comment: Each Web Service has a description associated with it using the WSDL XML-based language. WSDL is machine-processable and human-readable. One of the aspects of a Web Service described in WSDL is the set of message types accepted and produced by the Web Service
Source:	Web Services Architecture document at W3C http://www.w3.org/TR/ws-arch/

Term:	Web Service Resource Framework (WSRF)
Definition:	The WSRF is a set of Web Service specifications that define an approach to modelling and managing state in a Web Service context. It treats the persistent state as a resource. This is a long term successor to OGSI as an supporting infrastructure for the Open Grid Services Architecture (OGSA). A set of proposed specifications dealing with the association of Web services with stateful resources.
Source:	OGSA Glossary of Terms [93]

9. Generic references

Most references are provided in individual sections. A few apply to all sections and are shown here.

[106] Akogrimo Project, "State of the Art in Grids and Mobility", Akogrimo deliverable D2.2.4 (has also been labelled D2.2.1 Volume 2), http://www.mobilegrids.org/modules.php?op=modload&name=News&file=article&sid=45&mode=thread&order=0&thold=0 , May 2005

[107] Webopedia: Online Computer Dictionary for Computer and Internet Terms and Definitions, http://www.webopedia.com/

[108] Wikipedia, http://www.wikipedia.com

[109] W3C Web Services Glossary, http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/

[110] Terms and Definitions Database Interactive (TEDDI). http://webapp.etsi.org/Teddi/ , access: 2004-09-06

[111] Yourdictionary, www.yourdictionary.com

[112] AskOxford: Online Dictionary Resource from Oxford University Press, http://www.askoxford.com/

[113] Telecommunications: Glossary of Telecommunication Terms, US Federal Standard 1037C, http://www.its.bldrdoc.gov/fs-1037/fs-1037c.htm

[114] Kleinrock on nomadic computing, interview with Leonard Kleinrock in Ubiquity, an ACM journal, http://www.acm.org/ubiquity/interviews/v6i25_kleinrock.html