An Integrated Accounting and Charging Architecture for Mobile Grids

Cristian Morariu[1], Martin Waldburger[1], Burkhard Stiller[1,2]

[1] University of Zürich, Department of Informatics IFI, Communication Systems Group CSG
Binzmühlestrasse 14, CH—8050 Zürich, Switzerland
[2] Swiss Federal Institute of Technology ETH Zürich, TIK, Gloriastrasse 35, CH—8092 Zürich, Switzerland
[morariu|waldburger|stiller]@ifi.unizh.ch

The adoption of the Internet Protocol (IP) by a number of non-IP network operators, such as telecom or cable TV operators, opens the path toward new business models. IP will allow operators to provide a unified wired as well as wireless access to a wide range of services to their users. Additionally, using the same communication protocols and standard interfaces, enables different providers to coordinate any type of resources in Virtual Organizations (VO) and supports the composition of services aggregated across multiple domains.

On one hand, such an open environment requires new business models to be adopted by the involved parties. On the other hand, Grid middleware infrastructure supporting integrated accounting, charging, pricing, and billing across multiple domains has to be in place to facilitate service provisioning in multiple VOs. Based on the relevant set of requirements derived, a new and extended A4C Architecture (Authentication, Authorization, Accounting, Auditing, Charging) has been developed, implemented, and evaluated for mobile Grids providing pervasive access to knowledge.

Keywords: ***Mobile Grid, Accounting, Charging, A4C, Business Grid, IP-based Grid Services, Virtual Organization***

I. Introduction and Problem Statement

The rapid growth of Internet penetration in the mobile communication market as well as the capabilities improvement of today's mobile devices offer new opportunities for existing technologies designed for traditional wired networks to be deployed on modern mobile communications platforms. Most of the current research efforts on future mobile communication technologies adopt the Internet Protocol (IP) as the underlying communication protocol for video, voice, and data. The usage of IP will ease the process of integration of different communication infrastructures and will trigger the development of middleware services, such as AAA (Authentication, Authorization, Accounting), which offer their functionality to a broad range of applications to be deployed.

Integration of different access technologies as well as video, voice, and data services in an all-IP environment also paves the way towards offering of more complex services being composed by basic services. This, however, requires an appropriate service delivery platform to be in place, providing mechanisms for service delivery and service composition. Driven by the main concept of resource coordination across administrative domains and a strong service-orientation [11], grid systems qualify here as a well-suited solution. In the same way as IP serves as an integration enabler for underlying protocols and access technologies, grids allow for an integration of various stand-alone services, offered by different providers, to be composed in applications. This demands for accounting and charging mechanisms, since various service providers need to charge service consumption if services are offered in a competitive environment.

Several grid-related projects have designed and implemented accounting mechanisms for grid-services, such as APEL [8], DGAS [20], GASA [3], [12], GRASP [13], GSAX [4], Nimrod/G [2], and SGAS [22]. However, the key drawback with those approaches is that these solutions did propose and implement proprietary accounting and charging mechanisms only. Additionally, any interconnection of such accounting tasks with charging functions is not part of those solutions either. Thus, the accounting and charging architecture proposed within this paper extends existing work by far.

The term *Grid* was traditionally used for defining a distributed high-performance computing (HPC) architecture. Computational and storage grids are the two examples everyone thinks about when dealing with this domain. However, more recently, grid principles are not only applied to HPC, but are used for service virtualization [10] so that Virtual Organizations (VO) can be created by aggregating resources and services from different domains and different providers, irrespective of the underlying infrastructure and protocols used in those different domains. Taking the VO concept a step further, by granting mobile or nomadic users pervasive access to knowledge, sets the key focus for the work performed here and mobile grids in general.

Such mobile grids need to integrate and harmonize various views of all actors involved in an operational VO. These views are expressed typically by business roles determining a player's specific behavior and requirements. In order to outline the full range of organizational arrangements in providing mobile grid services in VOs, a comprehensive understanding of the underlying role model marks a prerequisite for the development of the according accounting and charging architecture for mobile grids. Thus, the respective role model for mobile grids has been developed in a first step, being described in this paper, to provide the basis for the charging and accounting technology developed in a second step, being described afterwards.

Over the last few years multiple research efforts have been performed on accounting, charging, and billing models for telecommunication operators and ISPs (Internet Service Provider). [16] compares existing accounting, charging, and billing mechanisms used by ISPs and mobile network operators, while key challenges are shown, which will be faced as the two worlds converge. An overview of existing pricing schemes used in broadband IP networks is given in [9]. Those schemes are evaluated in terms of technical and economic efficiency, coming to the conclusion that there are still multiple aspects to be addressed, such as multicast or existing of virtual operators.

Thus, an approach to design and implement a commercial grid solution for service providers and telecom operators in support of mobile users has been undertaken in Akogrimo [1], one of the two only projects in the world so far addressing mobile grid support explicitly. In accordance with the definition given by Foster for grid systems [11], Akogrimo defines mobile grids as follows: "A Mobile Grid consists out of resources that are not subject to centralized control, while it supports all kind of mobility (such as device, user, and session) and communicates using standard, open, general-purpose protocols and interfaces in order to deliver non-trivial and optimized Quality-of-Service (QoS) depending on the current context of the resource or the user. Furthermore, a Mobile Grid is communicating with the underlying network in both directions enabling cross-layer cooperation between Grid middleware and the network." [26]

The architecture developed provides relevant mechanisms to deploy grid applications in a mobile environment. Accordingly, the main problem in a mobile business grid is found in determining and implementing accounting and charging mechanisms of mobile grid services in a multi-provider setting. The solution proposed here, goes beyond current state of the art and embraces on one hand the identification of relevant key requirements on an accounting and charging architecture for mobile grid services by means of a business-driven role model and a mobile grid scenario. On the other hand, the extended architecture design is new, and presents in a detailed manner selected implementation aspects, which include as well the evaluation of the architecture against the scenario drawn for detailed motivation purposes.

The remainder of this paper is structured as follows. Section II extracts as new work the relevant set of requirements for the accounting and charging architecture, which is newly developed and presented in Section III. Section IV delivers major implementation details, followed by an extensive functional evaluation of the architecture against a mobile grid scenario. Finally, conclusions are drawn and ideas for future extensions are discussed in Section V.

II. Accounting and Charging in Mobile Grids

Accounting and charging mechanisms have to be able to capture the specifics of each of the different entities involved. This assures seamless interoperability between different business partners, not only for service provisioning, but also for generating and sharing revenue. Towards this goal, the relevant accounting- and charging-related terminology in use has to be determined. This provides the basis for an investigation of the specific organizational characteristics prevailing in mobile grids. Accordingly, the role model for mobile grid service provision in a VO is developed, and a sample mobile grid scenario is depicted. Based on both, the scenario, key requirements on the integrated A4C architecture (Authentication, Authorization, Accounting, Auditing, and Charging) are derived.

A. Accounting and Charging Terminology

The two worlds of mobile telecommunication and IP-based networks use different terminologies with regard to charging and its associated processes. Based on [24] and [15], a layered model and definition is used, where charging is positioned logically on top of metering and accounting, but below billing, as Table 1 describes in detail. This terminology defines the understanding of accounting and charging as it is used in the context of commercial electronic service provisioning. Thus, those terms are also applied to the domain of mobile grids.

B. Role Model for Mobile Dynamic VOs

From an organizational point of view, the grid systems' main idea — namely resource sharing and resource coordination across administrative borders [11] — is reflected by Virtual Organizations (VO). Taking mobility aspects in grid systems into account suggests an inspection of the VO concept with respect to potential extensions or even fundamental changes required.

1) Mobile Dynamic VO Nature

Several definitions for VOs exist, each focusing on different, partly overlapping aspects [23], [6]. By following the definition of [27], VOs for grid systems are perceived as to "allow for information and communications technology-supported, accountable and chargeable resource coordination across administrative domains, incorporating mechanisms for parameterizable secure authentication and authorization".

Mobile grid systems show important extensions over fixed grid systems in the area of mobile grid resources and users as well as with respect to dynamics, whereas the latter is provoked by mobility support. Mobility in terms of user, device, and service mobility requires a grid system to dynamically adapt to changing context, which has to be supported by adaptive workflows and an accounting system that envisages context-based charging. Since mobility and dynamics traditionally are not reflected by VOs, Mobile Dynamic Virtual Organizations (MDVO) are seen as extensions of VOs: "MDVOs are virtual organizations whose members are able to change locations while provided or consumed services remain available even after temporary loss of reachability, and while running or yet to be initiated workflows adapt to changed conditions, so that MDVOs are characterized by a strong dynamic element with respect to their organizational composition and their business processes" [27].

2) VO Role Model for Electronic Service Provisioning

In order to determine the full picture of service provisioning in MDVOs, a role model is needed that is on one hand generic enough to reflect all stake holders incentives, and to cover all composing elements for commercially offered electronic services, while this model on the other hand has to integrate the specific characteristics of grid systems and VOs. In other words, the role model has to consider in the

Table 1: Accounting and Charging Terminology [24], [15]

Terminus	Understanding
Charging	Charging calculates the charge for a given service consumption based on accounting records and the tariffs defined in the SLA. Charging acts as an umbrella term for charging options and charging mechanisms. This orthogonal separation helps to emphasize either on the more technical or economic aspects of charging. Charging mechanisms are used to implement and realize charging options.
Prepaid/post-paid Charging Option	With the prepaid charging option, the customer has to have a certain amount of credits prior to the service usage. Depending on the type service, periodical credit checks during service usage are performed. Prepaid charging influences the delivery of services to the customer, e.g., service usage may be denied. With the postpaid charging option, service charges are aggregated on the user's account after service usage and the user is invoiced after a predefined period.
Online/offline Charging Mechanism	With the online charging mechanism, the charging (i.e. charge calculation) has to be performed in real-time. Note that online charging implies that accounting and metering have to be done in real-time as well. For the offline charging mechanism, no strict time constraints concerning the processing time of charging (i.e. charge calculation) are defined.
Billing	Billing is the process of consolidating charging information on a per customer basis and delivering a certain aggregate of it to a customer. Tariff or tariff function, takes a set of accounting parameters as its input and outputs the charge to be paid for the particular value of those input parameters.
Accounting	Accounting defines summarized information (accounting records) in relation to a customer's service utilization. It is expressed in metered resource consumption, e.g., for applications, calls, or any type of connections.
Charge Calculation	Charge Calculation covers the complete calculation of a price for a given accounting record and its consolidation into a charging record, while mapping technical values into monetary units. Therefore, charge calculation applies a given tariff to the data accounted for.
Mediation	Mediation is intended to filter, aggregate, and correlate raw technical data which in most cases has been collected by metering. Mediation transforms these data into a form which can be used for storing and further processing.
Metering	Metering determines the particular usage of resources within end-systems (hosts) or intermediate systems (routers) on a technical level, including Quality-of-Service (QoS), management, and networking parameters.
Pricing	Pricing covers the specification and setting of prices for goods, specifically networking resources and services in an open market situation. This process may combine technical considerations, e.g., resource consumption, and economical ones, e.g., applying tariffing theory or marketing methods. Prices may be calculated on a cost/profit base or on the current market situation.

most general form an as wide as possible range of grid services that in turn aim at high-level resource (i.e. knowledge) coordination. Coordinating and sharing knowledge requires the aggregation of basic services into more complex, composed services to be in place. Since these services are considered to be commercial in terms of that they have to be charged to an entity, the respective role set dealing with financial flows has to be included in the role model. Incorporating those requirements results in the basic role model for electronic service provision within VOs as it is shown in the form of an Entity Relationship diagram in Figure 1. The role model does not explicitly reflect mobility or dynamics aspects. Even though those aspects are specific extensions of MDVOs over VOs, the same basic role model is valid in both cases. This is caused by its generic nature: While mobility and dynamics show implications on MDVO business flows, they do not alter the role of, *e.g.*, a service user. The understanding of this role remains unchanged, independent of whether a service user is mobile or not.

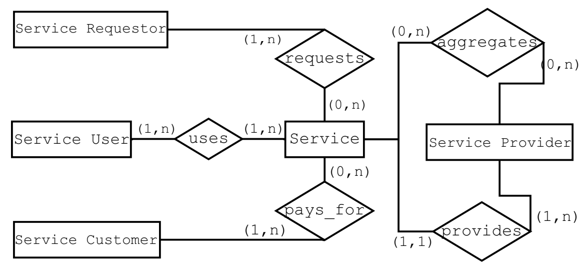

Figure 1: Basic Role Model for Electronic Service Provision in Virtual Organizations

The proposed role model has been drawn from an organizational viewpoint, which has implications on the understanding of the terminology in use. The term *Service* accordingly is not understood in a technical sense as a well-defined functionality that — in a layered approach — is offered through a service access point to a higher layer [7]. Services are rather perceived as non-tangible goods with an assigned utility level (macroeconomic and accounting view) or as electronic products (marketing view), while in fact only aggregated electronic services, bundled in an application (potentially including additional benefits like warranty), are assumed to satisfy the characteristics of a full-fledged electronic product. Nevertheless, the present gap between the respective service notions from a technical and business-driven view has to be bridged in order to integrate both understandings what is implemented by means of the grid middleware, whereof A4C is one important part with regard to commercial service provision. Since the role model needs to support a wide range of commercial services and organizational arrangements, only an at most generalized minimal role set has been expressed in Figure 1, consisting of entities that

- initiate service provisioning by explicitly expressing a demand for it (service requestor),
- consume the service results in terms of an electronic product (service user),
- are charged for service consumption (service customers), and finally
- offer and potentially also aggregate services (service provider).

The role of a service provider will find in specific scenarios more concrete realizations, such as communications provider, grid service provider, billing provider, or content and application service provider.

A real-word entity in terms of an actor can embody one or multiple roles. For instance, the roles of a service requestor, user and customer in basic organizational alignments probably are taken by one organizational entity, while in more complex arrangements different actors play these roles. For instance, for a pull-type service, an authorized agent might trigger upon a certain threshold or event reached that updated stock market analyses are prepared by the service provider and sent to a second entity, the user, while this service in the end will be charged to a third entity, *e.g.*, a company that subscribed to this service with the service provider so that the company's employees would be informed on important changes and news.

In commercial as well as in non-free service provision, one service thus has exactly one service provider assigned. The service itself is requested by none (push service type) or one or multiple (pull service type) requestors, whereas either one or multiple service users exist. One user only represents a unicast service, the case of multiple users can be sub-divided in multicast services with more than one, but specified number of users, while for broadcast services the full range of potential users in reach are supplied. With respect to compensation for service consumption, at least one service customer must be present, exactly one in case a full charging, and more than one in case a cost splitting scheme is applied.

C. Mobile Grid Scenario

At this stage a realistic mobile-grid scenario has been developed to enable the reader an easy to understanding of interactions among different players as depicted in Figure 2 and of players involved in service provisioning and in revenue distribution.

InsurerComp (IC) company just released its new service, a wearable heart monitoring device. This device continuously monitors human heart activity and periodically sends data to a monitoring facility that — based on a patient's historic records and other patterns in its database — can detect when a heart condition is imminent. For delivering this service, IC concluded a contract with a Mobile Network Operator (MNO) to get access to a large number of potential customers. For the necessary resources to perform health monitoring services, IC collaborates with two service providers, SP1 that provides database storage for patient records, and SP2 that provides the necessary computation power for analyzing the real-time data received from different patients.

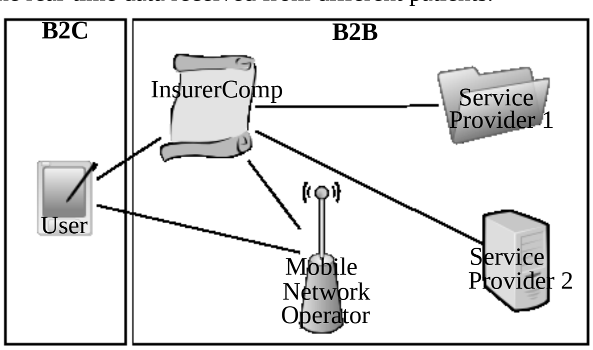

Figure 2: Actors and Contractual Relationships in the Mobile Grid Scenario

In the presented scenario, the user is assumed to maintain contractual relationships both with MNO for communication services and IC for the e-health application. Based on a B2B (Business-to-Business) relationship with IC, MNO handles systems integration and runs the technical infrastructure, while IC is concerned with direct customer relations and marketing activities of the e-health application. From a services viewpoint, thus applying the proposed role model (cf. Section II.B), MNO takes the roles of a *communications provider* (for voice and data services) and of a *billing provider,* whereas IC acts as a *content* and *application service provider.* SP1 and SP2 accordingly focus on their specific competencies in the role of *grid service providers*. This division of tasks allows a user - in this scenario an individual, taking the roles of a *service requestor, user,* and *customer* - to have access to multiple services and still receiving a single aggregated bill for all the services used during a given period of time.

D. Requirements

Driven by the investigated specific scenario and the underlying general role model, the set of key requirements has been identified. They address the middleware infrastructure related to A4C architecture an protocols as the main links between business and technology view on commercial service provisioning as follows:

- Robustness of the architecture— The A4C architecture should provide mechanisms to overcome faulty behavior of its components. Unavailability of any component should have minimal impact on the user's possibility to access services or provider's ability to charge for service consumption.
- Single Sign-On (SSO) Functionality — A single action of authentication and authorization from the user should give access to all the services she or he is allowed to use. A SSO approach allows users to authenticate once and then use services across different domains. It also allows for the mapping of all service sessions to an initial authentication event.
- Security — *Confidentiality*: As the A4C components transfer sensitive data, such as user credentials, service usage records and charging records, communication between these components needs to be secured. *Anonymity:* When using services in domains other than the home domain, a user's real identity should be protected. No service provider should have access to a users's real identity without the user's authorization. It should be the user who decides which level of anonymity she or he prefers: *No anonymity* — the real identity of the user can be used in foreign domains, *pseudo anonymity* — a separate identity (virtual identity) for each domain needs to be created and used whenever a service is requested from that domain, *full anonymity* — a new virtual identity is generated whenever a user requests a service from a foreign domain. The A4C Server in the home domain of the user should be the only network component capable to map between virtual and real user identities.
- Multi-service and multi-domain accounting and charging— Whereas in traditional networks accounting consists in counting the number of octets, packets or flows, an integrated accounting infrastructure for a mobile grid is required to understand and manage a larger set of parameters and multiple accounting record formats. This requires to have meaningful accountable units and QoS parameters

at hand, that hold for all relevant levels, particularly network, grid service as well as content level. Applications offered in a MDVO will most of the time aggregate services from multiple service providers. The accounting and charging components and protocols need to provide the mechanisms to aggregate accounting sessions from different administration domains and allow the aggregation of different service charges into a single bill.

- Roaming and Mobility — Typical users of a mobile grid are customers of mobile network operators. For having success a mobile grid should allow its services to be accessible from anywhere, using any type of access network.
- Deployment — For integrating an accounting and charging architecture for mobile grids in an existing infrastructure of a mobile operator compatible technologies need to be used.

The solution presented in this paper will be later evaluated in Section IV.C against these requirements.

Besides those specific requirements derived, a mobile grid accounting and charging architecture has to be flexible enough to support the full role model set, and it should be based on existing adopted or upcoming standards in order to be accepted widely by MNOs. Based on IETF (Internet Engineering Task Force) AAA standards, key requirements and behaviors of AAA architectures as well as protocols are already standardized by the IETF and applied by network operators.

Starting from the the identified requirements, an A4C architecture for mobile grids was designed and prototypically implemented on foundation of existing AAA standards.

III. A4C Architecture

In response to those requirements listed above, an A4C architecture for mobile grids was designed and prototypically implemented. This process was performed partially in the context of the Akogrimo [1] project Therefore, this section describes the main components of the A4C architecture, its functionality, and protocols used for communication. A discussion about how inter-domain service composition is supported follows, and the section ends with an overview on how different QoS parameters can influence accounting mechanisms and charging policies.

A. A4C Architecture Design

The A4C architecture presented in this paper is based on the generic AAA architecture [17] as defined by the IETF. All components of the A4C architecture are outlined in Figure 3. A more detailed view on the interactions between different components can be seen inFigure 4. The two most important components are the *A4C Server* and the *A4C Client*. Besides these, the SSO functionality is supported by the integration of a *SAML (Security Assertion Markup Language) Authority* component.

The A4C Server is the central component of the architecture. Its main tasks cover authentication of users, access control to services, service usage accounting and charging. Additional tasks are the auditing of service consumption for QoS compliance and storing of user and service specific profiles. All nominated tasks are services offered to components located in the same domain as the A4C Server in question.

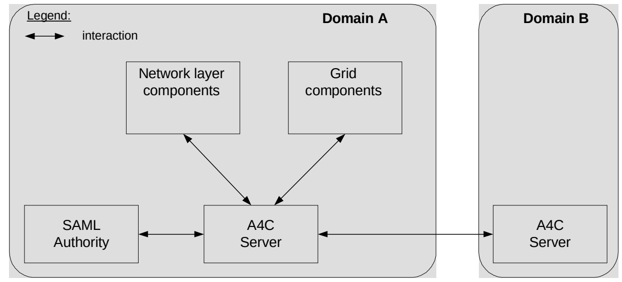

Figure 3: A4C Architecture Components

Besides these, A4C Servers also have to manage inter-domain related tasks such as authentication and authorization of roaming users, or accounting and charging for service sessions spanning across multiple domains.

Thus, the A4C Server is one of the key components in a service provider‘s domain as it provides those mechanisms for user identification, access control to services, and collection of data required for service charging. As a clear must, the A4C Server keeps all internal data integer and consistent. For achieving this, the architecture uses a logically centralized A4C Server in every domain. The central approach has to be seen only in the context of the architecture design. The physical deployment of the A4C Server might include several physical nodes acting as A4C Servers, i.e. for load-balancing purposes or for distributing A4C tasks to specialized nodes.

The A4C Client is the counterpart of the A4C Server on the client side. Its main task is to give network or grid components access to A4C services. Each service component that requires one of the functionality the A4C Server provides requires to integrate an A4C Client.

All communication between A4C Server and A4C Client is based on the Diameter protocol [5]. Diameter provides support for delivery of AVPs (Attribute Value Pairs), capabilities negotiation, error notification, extensibility through addition of new commands and AVPs and services necessary for applications, such as session handling or accounting. For the communication between the SAML Authority and the A4C Server the SOAP protocol is used, transported over HTTP.

SAML [21] is used to send security information in the form of authentication and attribute assertions to the mobile grid components. Furthermore, SAML provides an additional security block concerning high confidential information (such as authentication and attribute information of a user) in the A4C architecture. SAML is a secure interoperable language used to share users‘ information from the A4C Server to other components in order to provide SSO capability to the user and to offer attribute sharing of the user to other components. In order to provide support for SAML messaging, a SAML Authority is needed. Its task is to generate XML messages based on the SAML standard for sending authentication and attribute information in a secure manner. The SAML Authority has been designed as an internal subcomponent of the A4C Server. It aims at supplying IDTokens and SAML assertions to the A4C Server. The A4C Server contacts the SAML Authority when it requires to generate IDTokens and to verify such tokens presented by different components.

Support for an SSO and anonymity is achieved by using IDTokens. An IDToken is information that can be linked to a previous authentication event. Each time an authentication is successful, an IDToken is generated and handed to the user who requested an authentication. The IDToken can be used further by the user when requesting for services as a proof of authentication. The IDToken does not reveal a user's real identity, but only who can prove that the user is authenticated. The IDToken is always generated in the home domain of the user. If the user connects to the access network from a foreign domain the authentication request is forwarded to the home domain of the user, which will generate an IDToken after authentication and send it via the A4C Server in the visited domain. Whenever a service component receives an IDToken from a user, it needs to contact an A4C Server for checking the validity of this IDToken. If it proves to be valid, the A4C Server can return the real identity of the user or a virtual identity, based on previously defined policies.

B. A4C Session Model

For accounting and charging purposes, as well as for auditing the SLA compliance, service sessions between service delivery components and users need corresponding A4C sessions between service management and monitoring components and the A4C Server. An A4C session can be an authentication, authorization, or accounting session for a running service session. Whenever services are automatically instantiated and aggregated by *service composition* entities, session hierarchies are created. Session hierarchies have the purpose of keeping track of how multiple services interacted in order to assure the delivery of a more complex application. The A4C components keep track of the session hierarchies by using two techniques: *uniquely identification of each A4C session* and *tracking the parent session of each session.* The unique identifier is based on ongoing work performed in [25] and it is globally unique for each session, so that service hierarchies across multiple domains can be formed. Tracking of parent sessions assure that every service that was executed by a service provider can be linked to a session requested by a user, and then charged accordingly. The use of parent sessions requires a close coupling of the service component to the A4C Client in the sense that services are required to be aware of the session ID of the attached A4C accounting session ID.

C. Multi-layer QoS Definition

Both, requirements for multi-service accounting and flexible charging mechanisms, have to be addressed by the A4C architecture in a multi-domain service provisioning environment (cf. Section II.D), since different customers will probably have different expectations on the service quality and they are ready to pay accordingly. Besides these individual expectations, QoS (Quality-of-Service) parameters for mobile grids will influence the accounting and charging mechanism with respect to context-based charging, which embraces device and user context elements to be considered. In mobile grids, the set of QoS parameters for context-based charging needs to include the network, grid services, and content view, since guarantees given on one of these three levels influence the cost drivers of assigned accountable units. For instance, if on grid level a considerable amount of main memory is reserved for one job request, thus preventing other larger jobs from being accepted, shows impacts on the costliness of service provision.

Only on the networking level QoS parameters are widely understood to consist of the named parameters as they are presented in detail in the first section of Table 2 For the grid and the content level, those parameters, are not standardized or commonly used so far. The respective sections in Table 2, thus, propose the complete set of parameters for the purpose of this work. This paper does not address the mechanisms needed to signal, measure, and enforce the QoS at different layers, but argues that flexible SLAs and QoS parameters are required for an inter-layer, integrated architecture. Thus, having a broader understanding of QoS than traditional throughput, delay, jitter and loss will determine the basis for mobile grid architectures. .

IV. IMPLEMENTATION AND EVALUATION

For the prototypical implementation performed, the OpenDiameter library was chosen for the core of the A4C architecture. The A4C Server and A4C Client are implemented in C++. The SAML Authority is implemented in Java. To enable the integration of the A4C Client in grid components — mainly developed in C++ and Java — A4C Client interfaces are provided both, as C++ and as Java interfaces.

For better understanding, Figure 4 shows in more detail how the accounting process is realized for a service started in a foreign domain in our proposed architecture. For exemplification the network access service in a foreign domain has been chosen. As soon as a user accessed the network (in the foreign domain) and was authenticated, the metering service starts uses the C++/Java interface provided by the A4C Client to start an accounting session for that user. The A4C Client sends an Accounting-Start-Request to the A4C Server of its own domain sing the Diameter protocol. If required, the Charging component of the A4C Server in domain A can make a request to the A4C Server in the home domain of the user in order to check whether the user is authorized/able to pay for the service. This request is also encapsulated in a Diameter message. If everything is OK an accounting session ID is sent back to the metering component. During the service provisioning phase, the metering component periodically sends accounting records for user's session using the C++/Java interface provided by the A4C Client. These accounting records are encapsulated in Diameter messages and sent to the A4C Server in domain A. Once the session is terminated, the A4C Server is informed and a session record containing a summary of consumed resources and corresponding charges is created and sent to the A4C Server in the home domain of the user. The creation and transmission of this session record is the task of the charging component. The session record is sent using Diameter protocol. At this point, Domain B has enough information to charge the user for the consumed service in the foreign domain, while Domain A has the information required to charge Domain B B for the service session.

The operating system used for development and testing of the A4C implementation is Linux. The remainder of the section will emphasize major implementation decisions and

Table 2: Multi-layer Quality-of-Service Parameters

Layer	Parameter	Definition
Network QoS Parameters	Jitter	The variance of the expected difference between the minimal and maximal datagram delay
	Delay/ Latency	The time between an event and the expected effect
	Connection Availability	The ratio of the time a system is functional and the summarized time intervals a system is functional and not functional
	Packet Loss	The ratio of packets that arrived at the destination and the total amount of sent packets
	Throughput/ Data Rate	The amount of information units transmitted per given time interval
Grid QoS Parameters	Response Time	The time between a grid service request and the arrival time of the expected response
	CPU Type	The relevant set of CPU architecture architecture characteristics, *e.g.* the instruction set architecture in use
	CPU Topology	The characteristic pattern of interconnections between single CPUs
	Reserved Throughput	Guaranteed throughput, available exclusively for grid-related communication
	Memory	The available amount of main memory available for a process on a given grid resource for a given time period
	Storage	The available amount of background storage available for a process on a given grid resource for a given time period or for permanent storage
Content QoS Parameters	Confidentiality	The guarantee that only authorized entities are provided access to a given piece of information
	Integrity	The guarantee that data or messages are not altered in any way
	Anonymity	The guarantee that data or messages are not linkable with an identifying element of an entity
	Authenticity	The proof of the data or message origin
	Privacy	The possibility to restrict personal information about an individual or group of individuals from being distributed to unauthorized entities
	Liability	Availability of an entity that assumes legal responsibility for the delivered content
	Encryption	The guarantee that content can be inspected only in the presence of the corresponding decrypting key
	Context-specific Guarantees	Parameters that modify the quality degree of a service based on specific context information (*e.g.* presence, location, device type, connection type, mood)
	Application-specific Guarantees	Guarantees that are specific for an application or a group of applications. (*e.g.,* frames per second, resolution, color depth, encoding)

details of the A4C Server and the A4C Client.

A. *A4C Server Implementation*

The A4C Server implements a set of applications (*e.g.,* authentication, accounting, or charging) on top of the Diameter protocol. Diameter messages are pre-processed by the Diameter protocol handler as depicted in Figure 5 and delivered to the A4C Server module required to process the request. For the mobile-grid A4C implementation proposed

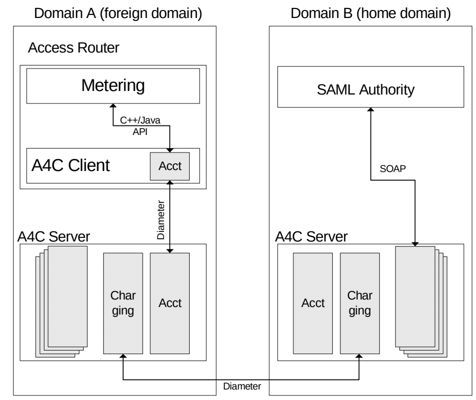

Figure 4: Accounting Implementation Architecture

in this paper a new authentication/authorization Diameter application was created as well as a new accounting application. Based on the command code included in the Diameter packet the message is forwarded to the proper module. The A4C Client is responsible to set the right command code in the Diameter request sent to the A4C Server. The A4C Server controller defines a set of interfaces between different A4C Server modules for internal communication. It also handles inter-domain-related tasks, such as authentication in a foreign domain, or control of the inter-domain charging messages exchange. The A4C Server is implemented in C++ and acts as a stand-alone application. The A4C Server includes two external components: a database which stores all information related to users, services, and service sessions (such as user profiles, service profiles, tariff schemes, accounting and auditing records, and authentication and authorization logs) and a SAML Authority. The database is used by the A4C Server. The implementation performed uses a MySQL database [18], but any other database can be used as long as it implements the interface defined.

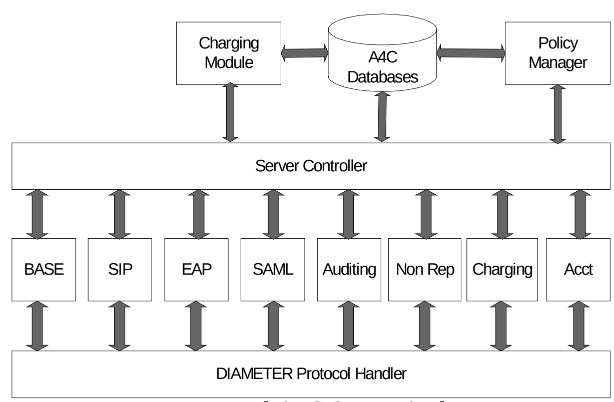

Figure 5: Internal A4C Server Architecture

The SAML Authority is implemented in Java [14] and the SOAP protocol is used for communication to the A4C Server. SOAP was used simply because of the current OASIS specifications for SAML, but the possibility of using the Diameter protocol for the communication between the A4C Server and the SAML Authority is investigated.

B. *A4C Client*

The A4C Client implements the client end of the respective applications in the A4C Server. The internal architecture of the A4C Client (cf. Figure 6) is quite similar to the internal architecture of the A4C Server. It shows the major difference that the control of different modules is not internally centralized, but it is outsourced to external applications through C++ and Java interfaces.

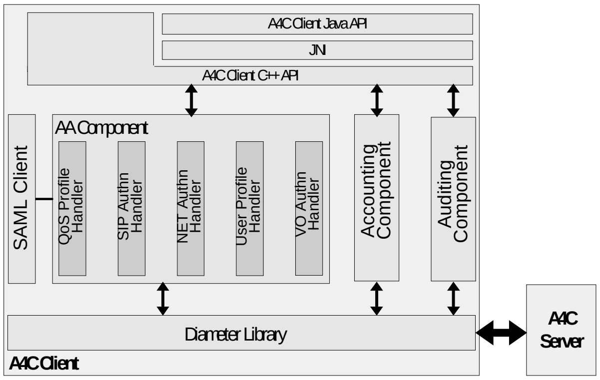

Figure 6: Internal A4C Client Architecture

As seen in Figure 6, all communication between the A4C Client and the A4C Server is handled by the OpenDiameter library. On top of this, three major modules for handling messages are implemented:

- *Auditing module* — It handles auditing related tasks as non-repudiation of messages or event notifications.
- *Accounting module* — It creates accounting sessions and handles the accounting messages related to service sessions.
- *Authentication and Authorization (AA) module* — It is involved in the processes of initial authentication, IDToken validation, and service access authorization.

The functionality of internal modules is made available to external applications through a set of interfaces. Any change in the internal A4C mechanisms can be handled locally in the module they belong to, without any impact on external applications or other modules.

The A4C Client is implemented in C++ and deployed as a Linux shared library. As grid services are typically implemented in Java, this A4C Client implementation uses the Java Native Interface (JNI) concept and offers interfaces to the A4C Client library in both, C++ and Java.

C. Evaluation

This section runs a qualitative evaluation of the architecture proposed against the scenario described in Section II.C and all major identified requirements of the A4C.

1) Scenario Mapping

Figure 7 depicts the network architecture used in the presented scenario. Each domain operates an A4C Server that performs access control, accounting and charging for service sessions that are executed inside the domain. Every service component integrates an A4C Client that connects to the A4C Server of the respective service provider domain.

Whenever a user connects to an access network of an MNO he is first authenticated and a network accounting session is started. The EH (E-Health) monitoring service is started by connecting to the EH Service component in the IC domain and presenting the IDToken for proofing the authenticity of the user. Based on the IDToken, the IC can obtain from the MNO the session ID generated for the authentication process. Based on this information, any service request in the IC domain can be related to an authentication process. After the service request is received, the EH Service component using its A4C Client starts an accounting session and requests a new database access service instance from SP1 and a new computation service instance from SP2. All requests for the two subservices contain the accounting session ID of the EH Service. SP1 and SP2 account separately for their services based on their own policies and relate their accounting sessions to the session ID received from IC. As A4C session IDs are globally unique, the two services from SP1 and SP2 can be related to the EH service, even the latest is executed in a different administrative domain. When the service session ends, SP1 and SP2 apply their own charging schemes on the accounted for data and generate charging records for each of the two services separately. Charging records are generated by the A4C Server of SP1 and SP2, respectively. Once charging records have been generated, they are sent to IC. The charging records contain information related to a service session (*e.g.*, service session duration, summary of accounting data, price) and are required for later billing. Based on those charging records received, the IC can, on one hand, include the costs of the two subservices in the price of the EH Service, and, on the other hand, the IC can distribute to SP1 and SP2 revenue received for the EH Service.

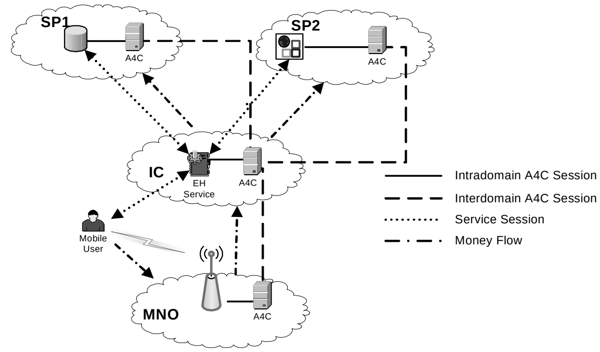

Figure 7: Heart Monitoring Scenario Network Architecture

2) Robustness of the Architecture

Each service component maintains through its A4C Client permanent connectivity with several A4C Servers at the same time. In case the server who is serving a current client becomes unavailable, the next messages will be forwarded to one of the other servers from the same domain. This way a single point of failure is avoided and the basic robustness of the A4C server infrastructure is achieved. Moreover multiple parallel sessions to different A4C Servers allows to load-balance the tasks of A4C infrastructure.

3) Security

Security determines the important topic for all tasks performed by the A4C infrastructure. Thus, communication between A4C Servers and Clients can be secured by using dedicated IPSec and Transport Layer Security (TLS) channels between these components. The use of TLS and IPSec assures the confidentiality and integrity of the control data transmitted. The use of the Diameter protocol is fully in-line with those measures.

Additionally, the use the SAML approach and its respective IDTokens for authentication of users enables a certain degree of anonymity by the hiding of user's real identity from the provider.

4) Single Sign-on Functionality

The A4C architecture developed provides a key functionality for commercial applications: Single Sign-on. This is achieved by using IDTokens for authentication and having trust between different administrative domains. IDTokens also provide support for anonymity. For each service an accounting record format is defined in the A4C Server database. The A4C Server retrieves from this database the format of the accounting record expected whenever an accounting session is started.

5) Multi-Service and Multi-Domain Support

The support of accounting and charging across multiple domains is provided by the use of globally unique session IDs for all accounting sessions and by mapping them to service hierarchies. In addition, the A4C servers may inter operate between two different domains by applying standard Diameter protocol messages and exchanges. Furthermore, every domain may offer independent services or mobility support functions on their own.

6) Roaming and Mobility

The use of IDTokens and the trust relationship between different service providers allow users to authenticate, while being in a foreign access network, or accessing services provided by third parties, while being charged by their home operator. The Diameter protocol provides support for creating roaming-aware A4C applications.

7) Deployment

The A4C architecture described is fully based on the Diameter protocol, which is widely accepted by mobile operators, thus making the integration in existing network infrastructures easier. Extensions of the base protocol are specified and can be added on top of existing implementations. Thus, only A4C Servers need to be deployed in every service provider domain and each service is required to integrate an A4C Client. Finally, for security purposes SAML Authorities are required in each administrative domain, which operates a customer database and which need to perform authentication.

V. Conclusions and Future Work

Driven by the key set of newly developed requirements on accounting and charging of electronic service provisioning in Mobile Dynamic Virtual Organizations — forming the concrete instantiation of a mobile grid —, this paper presented a new and extended A4C architecture for accounting and charging of such mobile grid services. The solution proposed, targeted at multi-service service provisioning platforms and spanning across several administrative domains, addresses on one hand those requirements of single sing-on, anonymity, multi-service accounting, and flexible charging technology-wise by an integrative effort of the various mechanisms discussed both for the A4C Server and Client components. On the other hand, the A4C architecture proposed was implemented in key parts and successfully evaluated against a mobile grid scenario as well as major functional requirements. Thus, the solution developed shows the first integrated approach on these subjects in a homogeneous architecture.

Further work will be focused on the integration of a charging settlemtent entity, which will enable grid service providers to bill for their service in an integrated manner, on the integration with the architecture proposed into existing grid metering systems, such as HPC clusters and computers, and on the investigation of detailed grid accounting and charging policies, which will be based in user as well as customer preferences and provider application markets.

Acknowledgements

This work has been performed partially in the framework of the EU IST project Akogrimo *"Access to Knowledge through the Grid in a Mobile World"* (FP6-2003-IST-2-004293). Discussions with Peter Racz, David Hausheer (University of Zürich), and Ruth del Campo (University of Stuttgart) in particular with respect to multi-domain service provisioning and SAML issues are acknowledged kindly.

References

[1] Akogrimo: *Access to Knowledge through Grids in a Mobile World*, EU Research Project FP6-2003-IST-2-004293, http://www.mobilegrids.org, March 2006.

[2] A. Barmouta*: Authorization and Accounting Services for the World Wide Grid*; Master thesis, University of Western Australia, June 2004, pp 1-154.

[3] A. Barmouta, R. Buyya: GridBank: *A Grid Accounting Services Architecture (GASA) for Distributed Systems Sharing and Integration;* Int. Parallel and Distributed Processing Symposium 2003 (IPDPS 2003), Nice, France, April 2003.

[4] A. Beardsmore, K. Hartley, S. Hawkins, S. Laws, J. Magowan, A. Twigg: *GSAX Grid Service Accounting Extensions*; September 2002, pp 1-30. http://www.doc.ic.ac.uk/~sjn5/GGF/ggf-rus-gsax-01.pdf.

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

[6] L. M. Camarinha-Matos: Infrastructures for Virtual Organizations - Where We Are; IEEE Conference Emerging Technologies and Factory Automation 2003, September 2003

[7] J.D. Day, H. Zimmermann: *The OSI Reference Model*; Proceedings of the IEEE, Vol. 71, No. 12, December 1983, pp 1334-1340.

[8] J. Gordon: *Accounting, 'the last A'*; Workshop Presentation, May 2005, pp 1-18. http://www.e-irg.org/meetings/2005-NL/johngordon-accounting.pdf.

[9] M. Falkner, M. Devetsikiotis, I. Lambadaris: *An overview of pricing concepts for broadband ip networks*; IEEE Communications Review, Vol. 3, No. 2, 2000.

[10] I. Foster: *The Anatomy of the Grid: Enabling Scalable Virtual Organizations;* 1st IEEE/ACM International Symposium on Cluster Computing and the Grid, Brisbane, Australia, May 2001, pp 6-7.

[11] I. Foster: *What is the Grid? A Three Point Checklist;* GridToday, Vol. 1, No. 6, June 2002, August 2005.

[12] C. Frogner, T. Mandt, S. Wethal: *Cluster and Grid Computing: Accounting and Banking Systems*; Study Project Report, May 2004, pp 1-59. http://hovedprosjekter.hig.no/v2004/data/gruppe05/files/cgc2004_report.pdf.

[13] The Grid-based Application Service Provision (GRASP) Project: *GRASP Tutorial, First Presentation: SLA Document, Manageability and Accounting Subsystem*; February 2005. http://eu-grasp.net/english/SalernoMeeting/GRASP%20Tutorial%20Final%20-%20Verdino.ppt.

[14] Java Website: *http://java.sun.com/*; March 2006.

[15] M. Karsten, J. Schmitt, B. Stiller, L. Wolf: *Charging for Packet-switched Network Communications—Motivation and Overview*. Computer Communication, Vol. 23, No. 3, February 2000, pp 290-302.

[16] M. Koutsopoulou, A. Kaloxylos, A. Alonistioti: *Charging, Accounting and Billing as a Sophisticated and Reconfigurable Discrete Service for next Generation Mobile Networks*; IEEE Semiannual Vehicular Technology Conference, (VTC 2002), Vancouver, Canada, September 2002.

[17] C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, D. Spence: *Generic AAA Architecture.* IETF, RFC 2903, August 2000.

[18] MySQL Website: *http://www.mysql.org/*; March 2006.

[19] OpenDiameter Project Website: *http://www.opendiame.ter.org/;* March 2006.

[20] R. M. Piro, A. Guarise, A. Werbrouck: *An Economy-based Accounting Infrastructure for the DataGrid*; 4th International Workshop on Grid Computing 2003, Phoenix, Arizona, USA, November 2003, pp 202-204.

[21] SAML Homepage: *http://www.oasis-open.org/committees/security/*; March 2006

[22] T. Sandholm, P. Gardfjäll, E. Elmroth, L. Johnsson, O. Mullmo: *An OGSA-based accounting system for allocation enforcement across HPC centers*; 2nd International Conference on Service Oriented Computing (ICSOC 2004), New York City, New York, U.S.A., November 2004, pp 279-288.

[23] Y. P. Shao, M. K. O. Lee, S. Y. Liao: Virtual Organizations: The Key Dimensions; Academia/Industry Working Conference on Research Challenges 2000 (AIWoRC'00), April 2000, pp 3-8.

[24] B. Stiller, J. Gerke, P. Reichl, P. Flury: *Management of Differentiated Services Usage by the Cumulus Pricing Scheme and a Generic Internet Charging System*; IEEE/IFIP Integrated Network Management Symposiums (IM 2001), Seattle, Washington, U.S.A., May 14-16, 2001.

[25] J. Vollbrecht, G. Carle, S. Zander, T. Zseby: *Session ID.* Internet Draft, http://www.aaaarch.org/drafts/irtf-aaaarch-session-id-00.txt, Work in Progress.

[26] M. Waldburger, C. Morariu, P. Racz, J. Jähnert, S. Wesner, B. Stiller: *Grids in a Mobile World: Akogrimo's Network and Business Views*; IFI Technical Report 2006.05, April 2006, pp 1-14, ftp://ftp.ifi.unizh.ch/pub/techreports/TR-2006/ifi-2006.05.pdf.

[27] M. Waldburger, B. Stiller: *Toward the Mobile Grid: Service Provisioning in a Mobile Dynamic Virtual Organization*; 4th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-06), Dubai/Sharjah, UAE, March 2006, pp 579-583.