ACCOUNTING FACILITIES IN MOBILE AND WIRELESS INTERNET

by

Hong Chen

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

University of Twente

Approved by Boudewijn R. Haverkort

Chairperson of Supervisory Committee

Aiko Pras

Dirk-Jaap Plas

Harold Teunissen

Date 4th July 2003

Place Enschede, the Netherlands

Acknowledgments

I still feel lucky that I have the opportunity to do my final project as a researcher at Bell Labs, which brought a perfect ending to my Master of Science study in Telematics. I can not achieve so much without the help of those people. Below I mention the names I am most indebted to.

First I would like to express my gratitude to the graduate committee: Professor Boudewijn R. Haverkort and Dr. Aiko Pras who gave me the high level guidance on the research direction, research approaches; Harold Teunissen and Dirk-Jaap Plas who gave me advice through the weekly meetings and daily coffee breaks. They spent a lot of time reviewing my thesis, offering suggestions and comments and correcting my grammar mistakes.

The open atmosphere at Bell Labs Twente enables me to think and work freely. The helpful discussions and conversations enable me to add more details to the coarse work. Many thanks to Jeroen Schot, Peter Leijdekkers, Dennis Bijwaard, Jacco Brok, Ko Lagerberg, Jeroen van Bemmel, Bastien Peelen, Ronald van Haalen, Maarten Wegdam, Arie de Heer, Richa Malhotra, Sue Atkins. Especially I would like to thank Miroslav Zivkovic, who listened to me patiently, pointed out the flaws in my design, gave me suggestions and encouragement.

Also, I would like to thank Róbert Párhonyi of UT and Sander Hille of Telematica Institute; they offered valuable references and suggestions in the beginning of my research.

Last but not the least important are the families and friends who will always be there when I run into trouble or dead end, give me encouragement and offer kind help when I am frustrated. Thank you Mom, I love you. And thank you Shuangqing Wei, Shawei Chen, Katarzyna Wac, BN Sapkota, Christian Tzolov, Xiaobo He, Dulce Pumareja, YanChun Huo, Alfraed Rosheuvel, Duong Do, Peng Xu and Weiqing Jiang.

Abstract

In this thesis, we introduce a system under 4GPLUS project where the service provisioning part to end users can be divided into three domains: access network domain, third party service provider domain and service platform domain. The Mobile and Wireless Internet enabled by the 4GPLUS project carries the characteristics that end users will be provided with transparent access across numerous heterogeneous networks to various services; there exist monetary relationships between end users and each of the service providers. However, it is desirable that the end user will receive only one and integrated bill from the service provisioning part.

After analyzing the requirements for the accounting facilities, we study the state-of-the-art charging models as well as current conventions, standards and existing researches of accounting systems. The main effort of the thesis is to present the development of an architecture of accounting facilities for the Mobile and Wireless Internet envisioned by the 4GPLUS project. A “One or Unified Bill” reference model is built up to represent the logical architecture of the accounting facilities. Our architecture allows service providers freely choose and change charging models without affecting others. The adjustment on charging model will affect the level of integration of records, which can be intra-domain integration or inter-domain integration. At the same time, our design supports real-time and non real-time accounting by using a time window mechanism. By introducing a money/credit reservation mechanism, the accounting facilities can support prepaid users as well as postpaid users. ParlayX web services are brought in our design to serve as interfaces between telecommunication networks and Internet. Several user cases are developed to provide details of the accounting facilities.

Finally, we set up a test environment, implement simplified accounting facilities and simulate the end user’s behaviors to prove the feasibility. The result shows by choosing certain charging models, intra-domain as well as inter-domain integration of charging records can be achieved. Our conclusion is that the end users will actually receive one integrated bill through our accounting facilities which include all the usage of different service providers in the access network domain as well as in the third party service provider domain.

Keywords: Accounting, Billing, Charging, Data Collecting, Metering, Mobile and Wireless Internet, Integration, ParlayX web services, 4GPLUS.

Table of contents

List of figures

List of tables

Acronyms

3PSP	3rd Party Service Provider
4G	4th Generation telecommunication networks
4GPLUS	4th Generation Platform Launching Ubiquitous Services
AAA	Authentication, Authorization and Accounting
ASM	Application Specific Module
AN	Access Network
BS	Billing System
CDR	Call Detailed Record
CN	Correspondent Node
COA	Care-of Address
EU	End-User
GSM	Global System for Mobile communication
GPRS	General Packet Radio Service
HA	Home Agent
IPDR	IP Detailed Record
IPsec	IP security
MIPv4	Mobile IP Version 4
MN	Mobile Node
SP	Service Platform
UMTS	Universal Mobile Telecommunication System
VPN	Virtual Private Network
WSDL	Web Service Definition Language

Any technology distinguishable from magic is insufficiently advanced.

-- Gregory Benford

1 Introduction

This chapter presents background, objectives, and approaches of the research as well as structure of the thesis.

This chapter is structured as follow: section 1.1 gives background information about the Freeband Knowledge Impulsive program, the 4GPLUS project which is affiliated to it and introduces the concept of Mobile and Wireless Internet. Section 1.2 states the objectives of this research; section 1.3 describes the approaches used in the research and section 1.4 outlines the structure of the thesis by presenting an overview of the chapters.

1.1 Background Information of the Research

1.1.1 The Freeband Knowledge Impulsive Program and the 4GPLUS Project

The Freeband Knowledge Impulse[1] (Freeband Kennisimpuls) is a joint initiative of the Dutch Government, research institutions and industry. The program focuses specifically on the enormous new challenges emerging in telecommunications and gives a lot of attention to the development of knowledge relating to “4G” or “fourth generation telecommunications”. 4G stands for one integrated, IP-based environment for all telecommunications requirements - including voice, video, broadcasting media and Internet - that utilizes both fixed and wireless networks.

4GPLUS stands for 4th Generation Platform Launching Ubiquitous Services[2]. It is one of the 12 projects within the Freeband framework focusing on the key aspect of the enabling software technology for next generation wireless and mobile services on top of heterogeneous networks, such as GSM, GPRS, UMTS and Wireless LAN. Interoperability issues for the 4G wireless network technology, such as multimedia session control and mobility management are addressed [PEEL02].

[1] http://www.freeband.nl/ENindex.html
[2] http://www.freeband.nl/projecten/4gplus/ENindex.html

The 4GPLUS project provides an architecture (Figure 1-1) for federated[3] service control across heterogeneous access networks and different administrative domains (data-access provider domain, UMTS service domain and enterprise domain), which will offer a uniform and open access to billing, user-information, location information and other service building blocks. The federation aspect implies that various service platform instances cooperate together to provide the overall functionality (the cooperation between the "service control" blocks in Figure 1-1). Dynamic auto-configuration features ensure that consistent and coherent service access is provided independent of the current network domain. The service platform ensures homogeneous service access across all federated networks, including session-mobility across network domains. The platform will provide transparencies to end-user and application providers with respect to both the domains and the underlying mobile and wireless network technologies, in particular 2.5G, 3G and Wireless LANs.

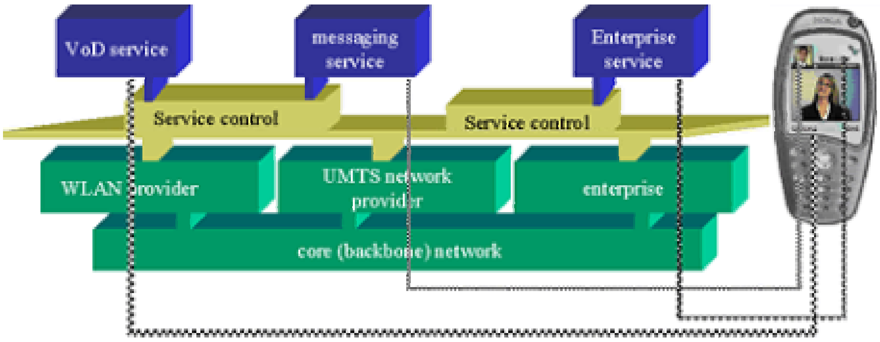


Figure 1-1: The 4GPLUS Platform

The Actors in 4GPLUS:

1. *End-User (EU)*
   A stakeholder role that represents a person that utilizes a telecommunications service provided by a service provider and subscribed by a subscriber. He does so by operating an application on a terminal.
2. *Access Networks (AN)*
   An access network can be a WLAN, GPRS or UMTS network. It is composed of all transport-related functions that enable a user to access services. It can be used to hide all access-specific peculiarities from the core network.
3. *Service Platform (SP)*
   A service platform consists of Service Platform components within one administrative domain to provide end-users with a transparent service access across heterogeneous networks and domains. These components are by definition distributed.
4. *Third Party Service Provider (3PSP)*
   A role represents a party that focuses on the production and operation of application and services for End-Users. There are three subtypes:
   a) Federated Third Party Service Provider: a third party service provider that offers services that in its turn relies on other services provided by other third party service providers.

[3] Federated: An organizational structure involving two or more autonomous stakeholders, in which the members have an agreement on how they will interwork with each other including the extent to which other members can share the resource of the one member.[PEEL02]

b) Content Provider: a service provider that offers raw content that might be used in other services.

c) Application Service Provider: an Application Service Provider is a company that offers individuals or enterprises access over the Internet to applications and related services that would otherwise have to be located in their own personal or enterprise computers.

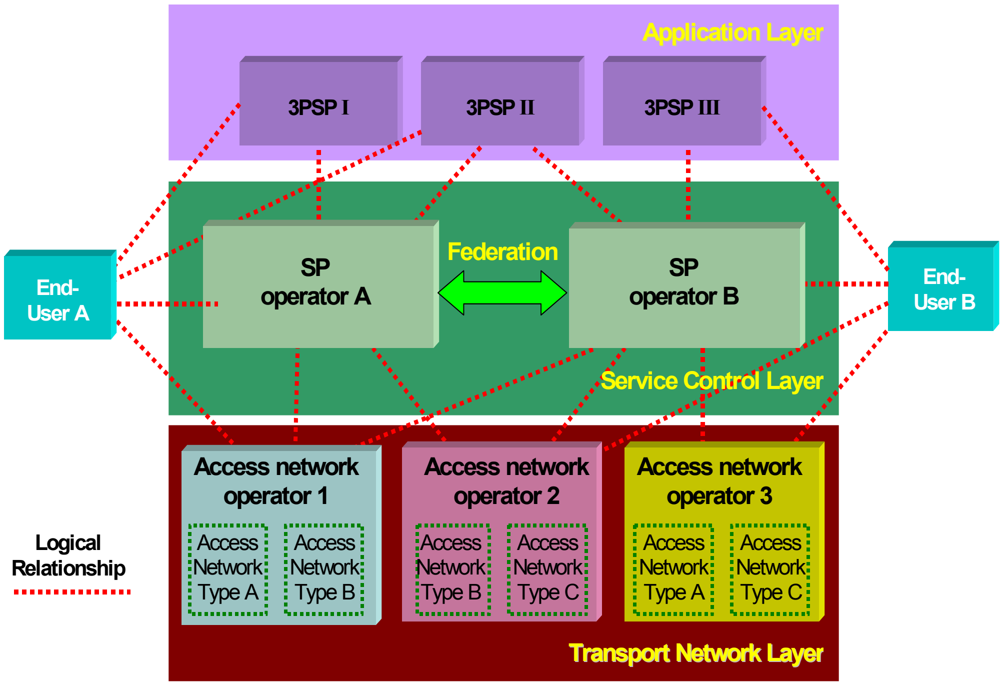


Figure 1-2: The SP's Basic Architecture and Its Context

Figure 1-2 shows that within a federation, different SPs operated by different SP operators cooperate for the support of end-user and 3PSP services and applications (Through federation, end-user A can receive services from 3PSP I while attaching to the network of Access Network Operator 3). Actually one function can be contained in multiple SPs and that not all SPs need to contain all functions. The added value of a SP is to serve the user with mobile network services, privacy and billing, but not necessarily the traditional Internet services.

1.1.2 The Mobile and Wireless Internet

Mobile and Wireless Internet is the merger of mobile communications and the Internet [KRAA02]. Mobile and Wireless Internet services not only aim to provide good performance over a wireless connection but also do so when user is mobile [ELSA02]. It can be seen as the extension of the Internet service, but there are many value-added features (e.g. applications using location service from the GRPS network to provide route guidance). Mobile and Wireless Internet is also the enabler of the anytime, anywhere, anything communication goal.

In Mobile and Wireless Internet, the access technologies/networks can be diverse, ranging from Satellite, Cellular networks (Mainly GSM, GPRS, CDMA and UMTS), Ultra-wideband, Wireless LAN to Bluetooth.

Mobile and Wireless Internet allows users great mobility, but as the Internet protocols were designed initially for fixed networks, there are also some problems when they are used in a wireless environment. These problems are now the hot research topics under fierce attack by the researchers.

The features of Mobile and Wireless Internet are summarized below; please note that these features are not orthogonal but more or less related to each other.

- User mobility
  User can move with a multi-network interfaces device which will dynamically associate with a variety of underlying access networks to various points of attachment, while still be connected to the applications he/she is using without interruption. While moving, the user may experience changing Quality of Service (QoS) of the applications due to the availability of underlying network resource.

- Dynamic topologies
  Result from node mobility, channel propagation effects, resource failure, power control, or antenna dynamics [MACK01]. Numerous routing protocols have been developed for operation in a highly dynamic networking environment (e.g. Mobile IP). To provide end-to-end reliability or flow control, the traditional transport-layer protocols (e.g. TCP) have been modified to adapt to differences in the characterizes of the networking environment and the services provided by lower-layer protocols

- Unreliable wireless links
  Unreliable here has two meanings: the error-prone information transportation and insecurity information transmission over wireless links.
  Because the signals are transported in an open environment, they are subjected to many interference such as fast multi-path fading, Doppler spreads, frequency dispersion, etc. The errors in lower layers will propagate to upper layers and cause further problems, such as drop of packets in network layer and re-transmission is transport layer.
  The open-air transmission exposes the users to possibilities of fraud, malicious hackers and malicious code much easier than the wired network.

- Heterogeneous underlying access technologies
  The technology includes all kinds of radio access technologies that support data transmission. The access networks range from e.g. low-bandwidth high delay satellite networks to high-bandwidth low error-rate pico-net formed by several portable devices.

- Self-configuration of device
  The portable device can decide when to hand over from one access network to another due the saved preference profile of the users, which can based on signal quality, bandwidth, price, QoS, etc.

- Adaptive and resilient applications and services.
  When the environment changes, the applications are able to detect the changing availability of the network resources of the users and make corresponding adjustment (e.g. an application of video streaming can provide different data rate to different users). Resilient means the applications still work normally when there are temporary connection interruption

- The broadcast or semi-broadcast nature of the wireless multiple-access media. The feature can be used to disseminate data intended from group communications.

1.2 Objective

The objective of this project is to develop an architecture of accounting facilities in support of the Mobile and Wireless Internet envisioned by the 4GPLUS platform. The accounting architecture will be quite complex due to the increasing number of access networks and third party service providers, but the complexity should be invisible to the end users. Although the end user use the services provided by various access networks and third party service providers, ultimately, they will receive one integrated bill from the SP.

The architecture should meet the requirements of the access network providers as well as the third party service providers; that is, the accounting architecture must support charging models currently in practice and also be extendable in the future. In addition, the architecture needs to be able to support the current practice in accounting: real-time and pre-paid.

The following requirements can be derived:

- The architecture should be able to combine accounting data from different entities and generate an integrated bill for the customers.

- The accounting architecture should be open so that new charging models can be added without much effort.

- The architecture should support real-time accounting as well as non real-time accounting.

- The architecture should support pre-paid users as well as post-paid users.

- Implementation of the accounting architecture as proof of feasibility.

1.3 Approach

1. Study the implication of 4GPLUS project and Mobile and Wireless Internet on accounting.

2. Exploit charging models and state-of-the-art technologies in accounting related area in PSTN, GSM, GPRS, UTMS and IP-based data communication networks. Determine terminology to be used in the project.

3. Follow a “reference mode” design methodology and come out with a design of the architecture of accounting facilities for the Mobile and Wireless Internet envisioned by the 4GPLUS project.

4. Implement the accounting facilities according to the design to prove feasibility.

1.4 Structure

The rest of the thesis is structured as follows:

Chapter 2: defines terminology that will be used in the thesis; offers a review of state-of-the-art charging models and accounting technologies in PSTN, GSM, GPRS, UTMS and IP-based data communication networks.

Chapter 3: presents the development of architecture of accounting facilities for the Mobile and Wireless Internet envisioned by the 4GPLUS project.

Chapter 4: describes a simplified implementation of the accounting facilities.

Chapter 5: brings out the conclusions and recommendations for future research.

Such exaltation of thought, while it let a drift the spirit, and gave it license in strange airs ...

- Seven Pillars of Wisdom -
--Thomas Edward Lawrence--

2 State-of-the-art in Accounting

This chapter first defines the terminology we will use through out the thesis and then presents a review of the available charging models and state-of-the-art in accounting technologies in telecommunication and Internet communications area.

This chapter is structured as follows: section 2.1 defines the terminology; section 2.2 introduces the charging models that are in use or have been proposed in PSTN, GSM, GPRS, and the Internet communications; section 2.3 focuses on the accounting architectures and corresponding technologies in telecommunication and Internet communications area. Accounting data representation and data transfer protocols are introduced in section 2.4. The "I-Mode™" will be analyzed and serves as a case study in section 2.5.

2.1 Terminology

There are many organizations which have defined terminology on accounting. Examples are the IETF (Internet Engineering Task Force), ISO (International Organization for Standardization), ITU (International Telecommunications Union - Telecommunication Standardization Sector) and ETSI (European Telecommunications Standards Institute). A detailed comparison on accounting related terminology that has been defined by these organizations can be found on the web site of Internet Next Generation (ING) work unit five[4].

The terminology defined by the above organizations is suited for a specific area, for example (ITU terminology is more suitable for the Telecommunication area, while IETF terminology is more suitable for the Internet communications area). The Mobile and Wireless Internet as proposed by the 4GPLUS project will be built on top of both the telecommunication networks (e.g. GPRS) and Internet networks (e.g. WLAN). In order to better reflect the characteristics of the Mobile and Wireless Internet, we define the terminology in the project by ourselves and will use it through out this thesis.

[4] http://ing.ctit.utwente.nl/WU5/D5.1/Definitions

Accounting: The collection of activities which consist of metering, data collecting charging, and billing.

Pricing: Pricing states a tariff to a unit of the resources consumed or services utilized.

Meter: The entity that collects information on consumption of resources and utilization of services. It can be software based as well as hardware based or a combination of both. Resources mean the bandwidth of networks in an AN, or the content provided by a 3PSP. Services can be the location information provided by an AN to other entities and also can be the routing guidance provided by a 3PSP.

Metering: The abstraction of activities that monitor the consumption of resource and/or utilization of services used by a served user, for the purpose of data collecting, charging, billing.

Metering policy: A set of rules used for configuring the metering layer, specifies when, where, who, what and how the consumption of resources and utilization of services will be recorded.

Usage record: A record that contains the information captured by a meter in a pre-defined format.

Data collecting: The abstraction of activities that collect the resource consumption and/or service utilization information provided by the metering entities; the creation of usage records; the transportation of usage records to the charging layer for the assignment of price. The usage record can be stored for a period of time (non real-time) or can be processed immediately (real-time).

Data collecting policy: Definition of where to collect the data, the type of data and the frequency in collecting them from the meters.

Charging: The process of calculating the resource consumption and service utilization using functions decided by the charging policies; the creation of charging records base on usage records coming from the same domain or different domains.

Charging policy: An instruction for calculating a charge. Usually, it is represented by a formula that consists of charging variables (e.g. volume, time) and charging coefficients (e.g. price per time unit). The charging variables are usually filled by information from usage records. Charging policies are one significant feature used by providers to distinguish themselves from competitors.

Charging record: A record that contains monetary information related to a consumption of resources and/or services.

Real-time accounting: In real-time accounting system, the periodically occurred accounting related activities take place within a defined time, which is generally short (e.g. every 15 seconds). This is usually associated with prepaid users.

Billing: The activity of generating invoices for the users based on the charging records coming from the same domain or different domains.

Charging model: The aggregation of charging policy, data collecting policy and metering policy. Generally, data collecting policy is based on charging policy, and metering policy is based on data collecting policy.

Prepaid services: Users pay in advance in order to get the services.

Postpaid services: Users have contract with 4GPLUS service platform and receive monthly bill upon resource consumption and service utilization.

2.2 Charging Models

Accounting is expensive: the cost of providing customer bills in a telecommunications network is approximately 40% to 50% of the annual revenue stream [GIBB99]; the cost in an ISP is approximately the same due to expensive cost of hardware and software of accounting facilities.

Two main motivations for accounting in mobile and wireless Internet are:

1. Recovery of financial investment for the provision of the infrastructure and generation of profits for shareholders, network operators and service providers;

2. Congestion control through differentiating service levels according to price. This may be achieved by placing price premiums on network bandwidth, quality of service (QoS) or utilization, which may then make the network self-regulating at a first level, or by reducing usage with additional charges that fewer subscribers may be prepared to pay for. [GIBB99]

Charging models are used by network operators or service providers to convert the consumption of resource into charging data and eventually into bills the customers receive.

We explore the charging models used by PSTN, GSM, GPRS and Internet in the following sections. Different terminologies are used in these systems to describe the same concept. In order to give a clear and consistent view of charging models, we will explain them in our terminology.

2.2.1 Charging Models in Public Switch Telephone Network

Odlyzko [ODLY00] describes the development of the telephony accounting from its start in the second half of the 19th century in the USA. At first, only flat rate billing was technically possible. According to the phone companies, a fixed price for any usage caused excessive usage and illegitimate borrowing of the telephones. On the other hand, the companies wanted to increase their revenues and limit their costs, especially the scaling costs caused by the unlimited usage. For a short time, this led to more complex pricing. First, every call cost a fixed sum (per-call billing) and then the calls had time-based tariffs. In addition, the phone companies later priced the calls by distance and by time of day. Especially the distance pricing was more complex than today. As the telephony billing advanced, the billing system became an essential part of the business, and sometimes the costs to make billing for a telephone company are more than the cost of the network. In the US, this has lead to local flat rates. It means that calling in the local network within one town or certainly local areas has a flat, usually monthly price; therefore considerable investment of the network operator on billing facility and the complexity of the system are reduced.

World-widely, the accounting of Public Switch Telephone Network (PSTN) is based on taxation pulses which determine a certain unit of time spent for a particular distance and time-of-day. Accounting is performed as recommended within the D-Series of International Telecommunications Union – Telecommunications Standardization Sector (ITU-T[5]) Recommendations [ITUD03].

The Table 2-1 lists common charging models of the telecommunication operators.

Charging model	Telephone service	Data service
Flat-fee	√	√
Distance	√	√
Spending caps	√	√
Duration	√	√
Reservation-based		√
Volume-based		√
Service Class-based		√
Bandwidth		√

Table 2-1: Charging Models Used by Telecomm. Operators

Note: Data service may include X.25, frame relay, ATM and Dial-in Internet service. Most of them are connection-oriented services.

In practice, telecommunication operators often combine several charging models. For example, the charge in telecommunication networks mainly consists of three parts [STIL98]:

1. Access-fee: collected which is usually a monthly charge for using an access link to the network.

2. A per-call or connection/reservation-setup-fee.

3. Usage-fee: be used to charge services on time, volume, or QoS-basis.

4. Content fee: Depending on the application content this fee may be omitted (e.g., telephony, fax, e-mail services where the “content” is provided by the customer herself), billed separately (e.g., CNN video service), or integrated into the telecommunication accounting system (e.g., 900 numbers).

A popular combination of charging models in PSTN is shown in Figure 2-1:

5 http://www.itu.int/ITU-T

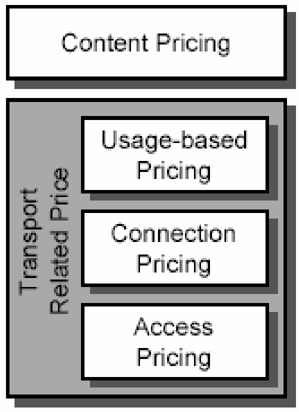


Figure 2-1: A Popular Combination of Charging Models in PSTN

2.2.2 *Charging Models in GSM, GPRS Networks*

In Global System for Mobile communication (GSM), accounting is performed in the same way as it is in the PSTN – by measuring time units (monetary unit per time unit), distance and there are also time-of-day and time-of-week based accounting and charging.

In General Packet Radio Services (GRPS) networks, the accounting has two parts: one is the accounting on traditional telephone service; the other is accounting on the data communication that GRPS provides, which is evolving toward all IP networks. Currently, the charging model in data communication part of GPRS is based on volume of traffic. Data volumes on both the up-link and down-link directions are counted separately. It is also reported that some operators use flat-fee charging model.

2.2.3 *Charging Models in Internet Communications*

Accounting in Internet communications is relatively young when compared to the history of accounting in the Telecommunication networks. It is quite interesting to look at the development of accounting in telecommunication area to predict the trend of accounting in Internet communication.

Currently, the popular Internet charging model is switching from the flat-fee model to usage-based models (ISPs advertise flat-fee in the Netherlands).

Below are the details of the representative charging models that are already in use or have been proposed:

Flat Fee
Network service provider sets a fixed charge (usually monthly) for the connection. The predictable fees reduce risks for users, who will have a known expectation for payments and can do planning/budgeting. Flat fee encourages usage (if the network is not congested), which increase customer satisfaction and does not hurt the provider. [DAVI95]
Advantage: Flat Fee can save considerable investment of the network operator on accounting facility and reduce the complexity of the system.
Disadvantage: Accounting will not serve as a method for congestion control. No added revenue for the service providers in times of above average.

Volume-based
It means measuring volume of traffic exchanged in a session. It requires the implementation of a traffic measuring system in the network and complex accounting systems that can process the usage records and charging records on a subscriber and customer basis.
Advantage: accurate usage of network and service can be measured, which enable the cross charging between networks as well as ISPs.
Disadvantage: the cost of measuring the traffic may be greater than their actual value, both from an infrastructure investment and additional network traffic viewpoint.

Edge Pricing
The model charges for the usage at the 'edge' of the network scope for the subscriber, rather than along the expected path of the source and destination of the calling session. In this charging model, the networks can cross-charge each other for the usage at the network 'edges'.
Advantage: all session data can be captured locally and does not involve exchanging billing data with other networks and partners for subscriber billing.
Disadvantage: The cost of collecting the edge usage information may be in excess of the value of the collected information.

Paris-Metro Charging
The Paris-Metro charging is based on the pricing scheme as used by the Paris metro system until about 15 years ago. In the Paris metro system, 1st and 2nd class would be the same in quality (same type and number of seats). The only difference between 1st and 2nd class was the price, a 1st class ticket would cost twice as much as a 2nd class ticket. The result was a less congested 1st class, since only people who valued getting a seat would pay the extra amount of money for a 1st class ticket. Applied to a network, the network would be partitioned into several logical channels each with a fixed capacity and all using best effort. This could be done using the same technique as DiffServ. The only difference between the channels would be their price. This pricing itself should result in different congestion levels in the logical channels and therefore different levels of QoS.
Advantage: Can give the subscriber flexibility to control the cost of his network traffic.
Disadvantage: Introduce complexity to the network. Difficult to realize when traffic is traversing too many congested bottlenecks.

Smart Market Mechanism
The smart market mechanism is based upon the principle that users of transport services should be charged for creation of congestion, because congestion creates 'social costs' by prohibiting others from using the network resources. MacKie-Mason and Varian proposed the mechanism in [MACK95c]. Packets should be prioritized based upon the value that users put on getting their packets through congested routers. To this end, packets are accompanied by a bid. At congested routers, packets are prioritized according to these bids. Users are not charged for the price they bid, but instead for the market-clearing price, which by construction will be lower than the bids of all admitted packets. See also [MACK95a] and [MACK95b].
Advantage: Provide user with the higher price the better services
Disadvantage: Would be worth only when the network is congested, difficult to implement.

Reverse Charging
Currently, Internet charging models only allow ISPs to get money from their own

customers. But it is desirable for ISPs to get money from receiving users who are connected via other ISPs, for the costs that would otherwise be paid by the sending user. More details can be found at [SPRE00].

2.3 Accounting Architectures and Corresponding Technologies

There are standardization organizations developing accounting architectures, in both the telecommunication and Internet communication area. The accounting architectures for circuit-switched voice service are matured, so in this section, we briefly introduce the accounting architectures in Telecommunications and mainly focus on the Internet accounting related works.

2.3.1 GSM networks

In Global System for Mobile communication (GSM) network, accounting related data for all calls made in the network are processed in the Mobile Switching Centers (MSC), usually based on subscriber IDs in the network. The collection of the metering data is normally via high-speed communication links using reliable data protocols such as File Transfer and Management (FTAM) and X.25. Once metering data is collected centrally it can be processed into charging records and later made into subscriber invoices/bills using dedicated billing systems and the mobile network's charging tariffs. The data can also be further processed by additional data-mining systems to detect subscriber's usage patterns, possible fraud detection and subscriber profile surveying.

2.3.2 GPRS networks

General Packet Radio Service (GPRS) is logically implemented on the GSM structure through the addition of two network nodes, the Serving GPRS Support Node and the Gateway GPRS Support Node [TS1202].

The technical specification of 3GPP[6] introduced: "Charging Gateway Functionality" (CGF), which provides a mechanism to transfer accounting information from the SGSN and GGSN nodes to the network operator's chosen Billing Systems (BS). A simplified GPRS accounting logical architecture is shown in Figure 2-2, the complete architecture can be found in [TS1202]. The Charging Gateway concept enables an operator to have just one logical interface between the CGF and the BS. The CGF may be supported in one of the following ways [TS1202]:

- As a centralized separate network element (Charging Gateway);
- As a distributed functionality resident in the SGSNs and GGSNs.

[6] http://www.3gpp.org

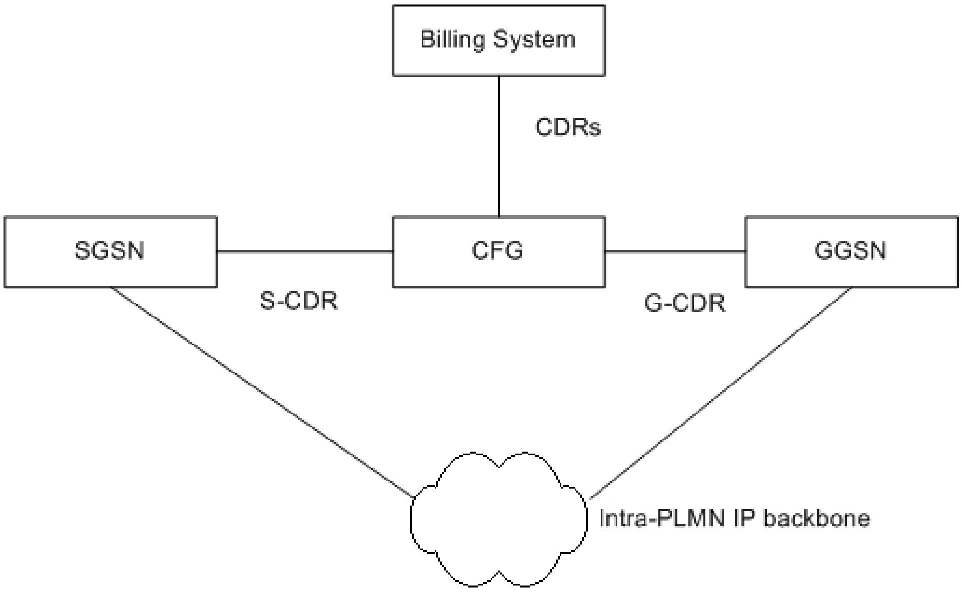


Figure 2-2: A Simplified GPRS Accounting Logical Architecture.

The main functions of the CGF are [TS1202]:

- Collection of GPRS Call Detailed Records (CDRs) from the GPRS nodes generating CDRs;
- Intermediate CDR storage buffering;
- The transfer of the CDR data to the billing systems.

2.3.3 UMTS Networks

Briefly, the 3rd generation accounting logical architecture can be subdivided into two transmission planes, the Circuit Switched (CS) domain and the Packet Switched (PS) domain. Figure 2-3 shows a simplified UMTS accounting logical architecture. The Call Detailed Records (CDRs) generated by the serving nodes: Serving GPRS Service Node (SGSN) and Gateway GPRS Service Node (GGSN) for the appropriate domain are forwarded via the Charging Gateway Function (CGF) entities to the Billing System for processing. The Service Control Function (SCF) may also transfer CDRs directly to the Billing System. However, the current specifications do not include any CDR descriptions for the SCF. (While not shown explicitly in this figure, the VLR may also generate CDRs.) CDRs for the Multimedia Messaging Service (MMS) are delivered by the MMS Relay/Server when receiving or delivering multimedia messages to the MMS User Agent or to another Multimedia Messaging Service Environment (MMSE). CDRs from the MMS Relay/Server are transferred directly to the Billing System. [TS3202]

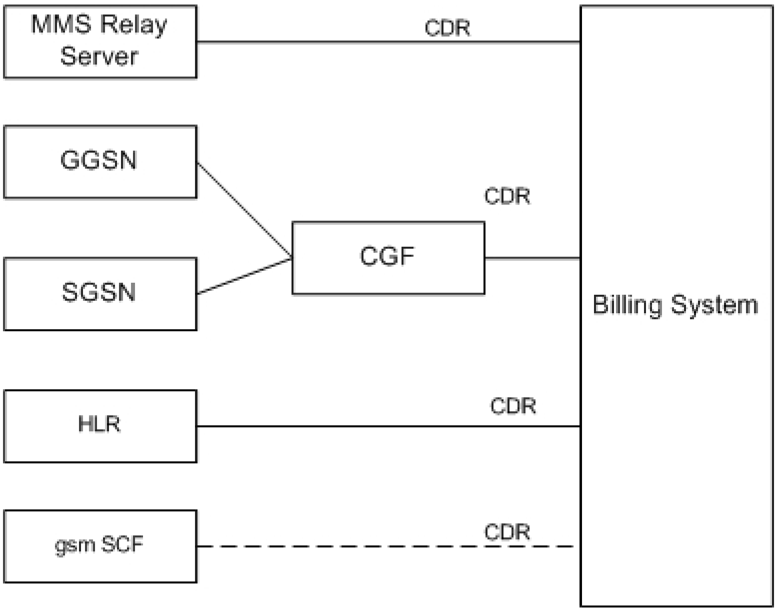


Figure 2-3: A Simplified UMTS Accounting Logical Architecture

2.3.4 Internet Accounting Architectures

2.3.4.1 The IRTF Generic AAA Architecture

The Internet Research Task Force is responsible for promoting research of importance to the evolution of the future Internet. Its research groups are working on topics related to Internet protocols, architecture and applications. The work in IRFT related to accounting is carried out in the Authentication, Authorization, Accounting and Architecture (AAAARCH) research group[7] [8], which will work to define a next generation AAA architecture in RFC2903 [LAAT00] that incorporates a set of interconnected:

- Generic AAA servers
- Application specific modules (ASMs), which would manage resources and configure the service equipment to provide the authorized service.
- Policy and event repositories. A database containing the available services and resources about which authorization decisions can be made.

[7] http://www.aaaarch.org/
[8] Here we use the terminology of IETF defined by AAAARCH working group in order to keep the consistence of the original work.

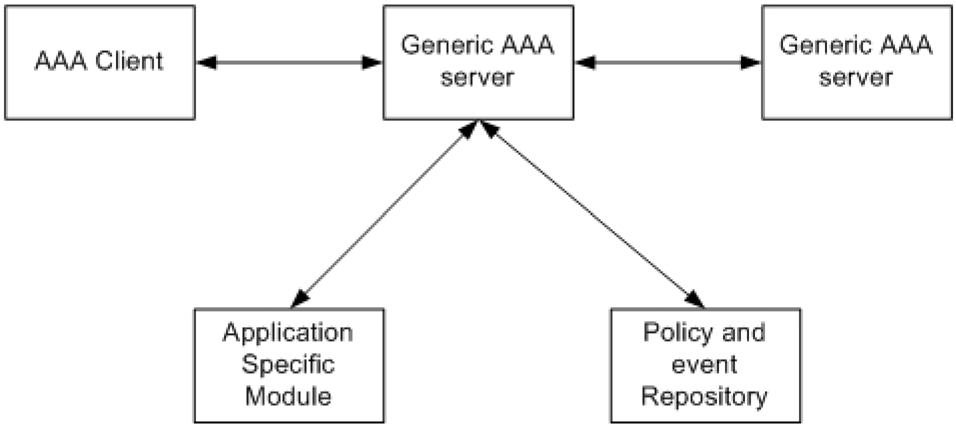


Figure 2-4: The Generic AAA Architecture of AAAARCH Research Group

In this architecture (Figure 2-4), user or another AAA server contacts the AAA server to get authorization. The AAA servers provide an interface for the ASMs to perform application specific AAA functions, it has no knowledge about the application-specific services: the application-specific information it forwards is opaque to it. The AAA server will consult Policy and event Repository to make the authorization decision. The communication protocols between these entities are not defined yet.

The generic AAA server modules can contact each other to evaluate parts of requests (which can be authentication, authorization or accounting requests), thus allowing for inter-domain AAA services and improve the scalability.

Accounting is defined as a function of AAA server and does not include metering and data collection, which serve as the input of the accounting function.

The recent work had been focus on policy-based accounting [ZSEB02]. It will enable service providers to apply flexible and configurable charging scheme and accounting service to user. Work has also been emphasized on inter-domain accounting.

2.3.4.2 The IETF RTFM Architecture

The real-time flow measurement (RTFM) working group[9] of Internet Engineering Task Force (IETF) focuses on developing a standardized solution for metering Internet traffic flows and collecting metered data for accounting purposes. The RTFM architecture[10] is based on the interaction between four entities [BROW99b]:

- *Meter:* Meters are placed at measurement points determined by Network Operations personnel. Each meter selectively records network activity as directed by its configuration settings. It can also aggregate, transform and further process the recorded activity before the data is stored. The processed and stored results are called the 'usage data'.

- *Meter Reader:* A meter reader transports usage data from meters so that it is available to analysis applications.

[9] http://www.ietf.org/html.charters/OLD/rtfm-charter.html
[10] Here we use the terminology defined by RTFM working group of IRTF in order to keep the consistence of the original work.

- *Manager:* A traffic measurement manager is an application, which configures "meter" entities and controls "meter reader" entities. It sends configuration commands to the meters, and supervises the proper operation of each meter and meter reader. It may well be convenient to combine the functions of meter reader and manager within a single network entity.

- *Analysis Application:* An analysis application processes the usage data so as to provide information and reports which are useful for network engineering and management purposes

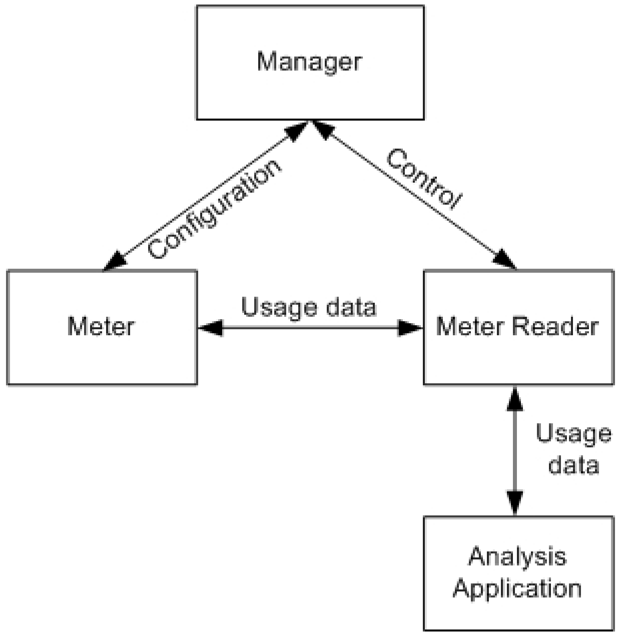


Figure 2-5: The RTFM Traffic Measurement Architecture.

The interactions between these entities as presented in Figure 2-5 are:

Meter – Meter Reader:
A meter holds usage data in an array of flow data records known as the FLOW TABLE. A meter reader may collect the data in any suitable manner. (E.g. collect entire or portion of the data in the table, at any time).

Manager – Meter:
A manager is responsible for configuring and controlling one or more meters. Each meter's configuration includes information such as: Flow specifications, Meter control parameters, sampling behavior. One Manager can control multiple meters. One meter may have rules set by several managers, but there is a Master Manager that controls the key parameters.

Manager – Meter Reader:
A manager is responsible for configuring and controlling one or more meter readers. A meter reader may only be controlled by a single manager. A reader must know the following information to read from a meter: A meter reader needs to know at least the following from every meter it is collecting usage data from:
the meter's unique identity, data collecting frequency, which flow records are to be collected, etc.

Meter Reader – Analysis Application
Once a collection of usage data has been assembled by a meter reader it can be

processed by an analysis application

There may be interactions between a Manager and an Analysis Application and Manager to Manager interactions.

Meters may report to one or more meter readers, and redundancy of meters or meter readers can be used to increase the reliability.

The RTFM work group also developed Meter Management Information Base (Meter MIB) for use in controlling an RTFM Traffic Meter, in particular for specifying the flows to be measured. It uses SNMP to retrieve data from the meter. The Meter MIB supports multiple protocol stacks on which traffic flows can be based [BROW99a]. The protocol entities supported are:

- Adjacent Address: Ethernet (MAC address), FDDI, Token Ring.
- Peer Address: IP, CLNS, IPx (Novell), AppleTalk, DECnet.
- Transport Address: UDP or TCP port number.

Additional attributes that are supported relate to transport protocol and application protocol. So it is possible to monitor e.g. HTTP flows.

NeTraMet is an implementation of the RTFM architecture. The NeTraMet package consists of several tools. NeMaC is a combined manager and meter reader, As an addition for easy starting there is nifty, a program, which graphically displays data like “Packet rate vs. Flow lifetime”. Another main component of the NeTraMet distribution is a SRL compiler. SRL stands for Simple Rule Set language, which was defined for easy creation of new filters. From the rules defined in SRL the compiler will generate filter rules for the traffic meter. NeTraMet uses a Packet Matching Engine (PME) which classifies the packets according to the rules.

2.3.4.3 Policy-based Architecture from the SUSIE Project

SUSIE[11] is a research project co-funded by the European Union under the ACTS program, it concerns accounting for Internet services with Quality of Service. The project has two primary goals:

- To establish the basis for convergence accounting based on an ATM network with a usage based charging model delivering Premium IP[12] with ATM related QoS and charge prioritized ATM streams selected via a QoS/Price trader.
- To integrate and validate TINA services (including management) with an IP/ATM network.

[11] http://www.teltec.dcu.ie/susie/welcome.html

[12] Premium IP enables the vast range of existing IP-based applications to be used, but over an underlying network that will allow QoS guarantees, in terms of delay, delay variation and error rates, to be given to the user. [OLEA99]

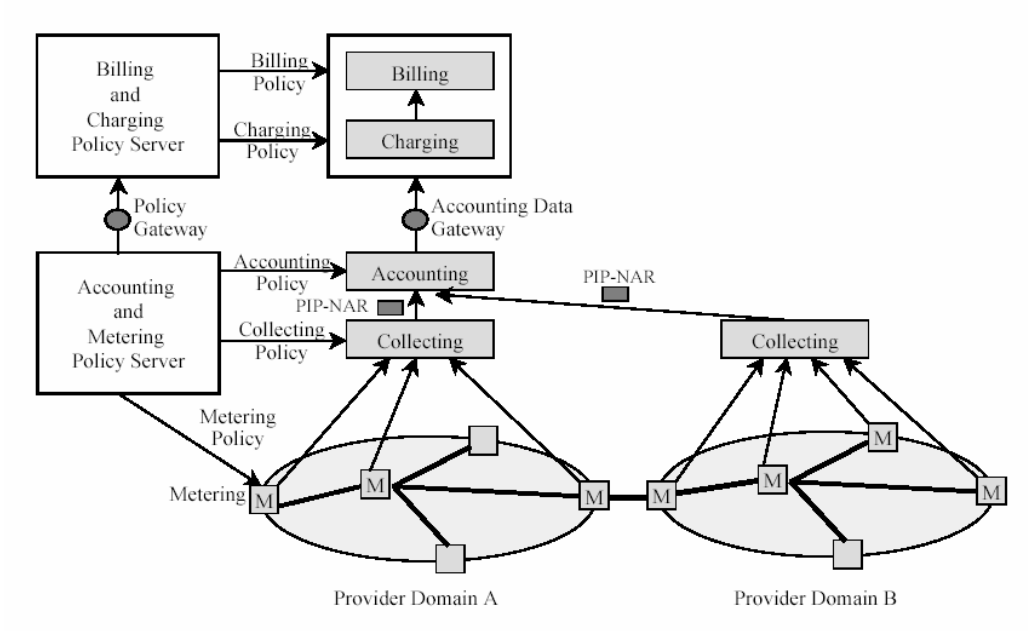


Figure 2-6: Accounting Architecture from SUSIE Project

In the reference model of its architecture (Figure 2-6), the accounting system is divided into five layers: metering layer, meter reader layer, accounting processing layer, charging layer and billing layer. Each layer has a corresponding policy. [OLEA99]

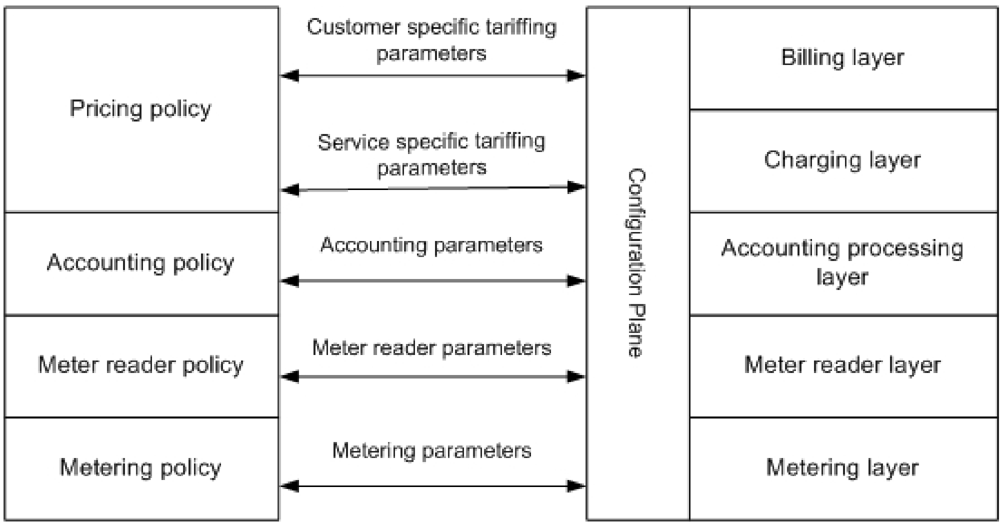


Figure 2-7: Charging and Accounting Reference Model of SUSIE

The architecture support multicast accounting; metering may take place in the edge routers as well as at the multicast routers. It uses metering that conforms to the IETF Real Time Flow Measurement Architecture (RTFM). For configuring a meter and for collecting usage data, SNMP and a meter MIB are used. The collecting entities at each domain collect the data from the meters and fill the Premium IP Network Usage record (PIP-NAR) data structure, which is used for transporting usage information within one provider domain and also for exchanging usage information in a multi-provider scenario. [HART99]

The policy-based architecture supports flat rate, duration-based and volume-based charging it also supports static and dynamic pricing schemes.

2.3.4.4 The AAAC of Moby Dick

The Mobility and Differentiated Services in a Future IP Network[13] (Moby Dick) is a project under the Information Society Technologies[14] (IST). One of its main objectives is to propose architecture for wireless Internet access by developing new mechanisms for seamless hand-over, QoS support after and during hand-over, AAA, and charging[15].

Authentication, Authorization, Accounting, and Charging (AAAC) from Moby Dick project enhances the AAA Architecture proposed by the IETF and IRTF with charging and auditing functionality and targets this generalization at the QoS-enabled Mobile Ipv6 (MIPv6).

The AAAC architecture (Figure 2-8) uses to Application-Service Modules (ASMs) to support a Mobile IPv6-centered Attendant as well as a QoS Broker. The main functionality of the metering module is to meters and aggregates the resource/service consumption and sends this information to Accounting and Charging Module. The metering module can also meter QoS information (e.g. DSCP) if the QoS broker does not provide these to the AAAC system. [MOBY02]

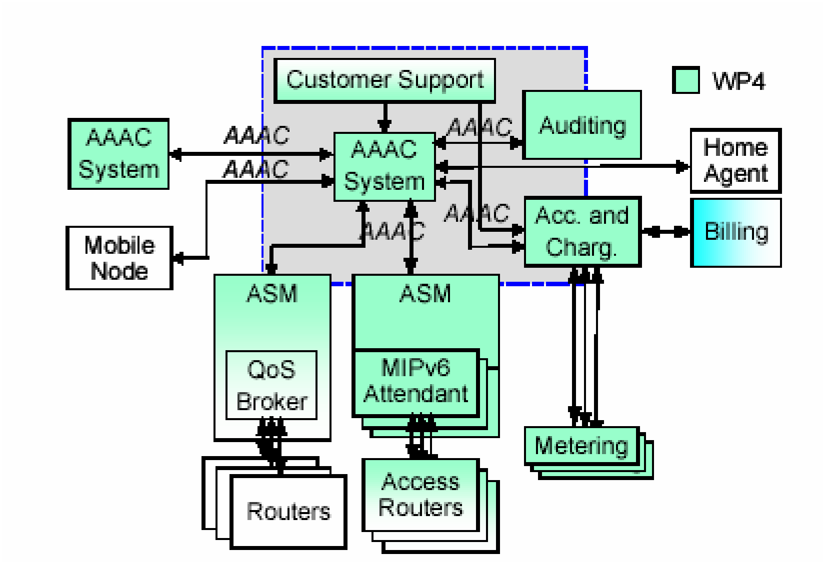


Figure 2-8: AAAC Architecture for QoS-enabled Mobile IPv6 Environment

2.3.4.5 The GigaABP Accounting Architecture

Giga Accounting, Billing and Payment[16 17] (GigaABP) is a project carried out at the Telematics Institute[18], the Netherlands. The goal of the project is to develop a

[13] http://www.ist-mobydick.org
[14] http://www.cordis.lu/ist/home.html
[15] Here we stick to the original terminology used by the project to keep the consistency.
[16] http://www.telin.nl/Middleware/GIGAABP/ENindex.htm

[17] Here we follow the original terminology of the GigaABP project to keep the consistency

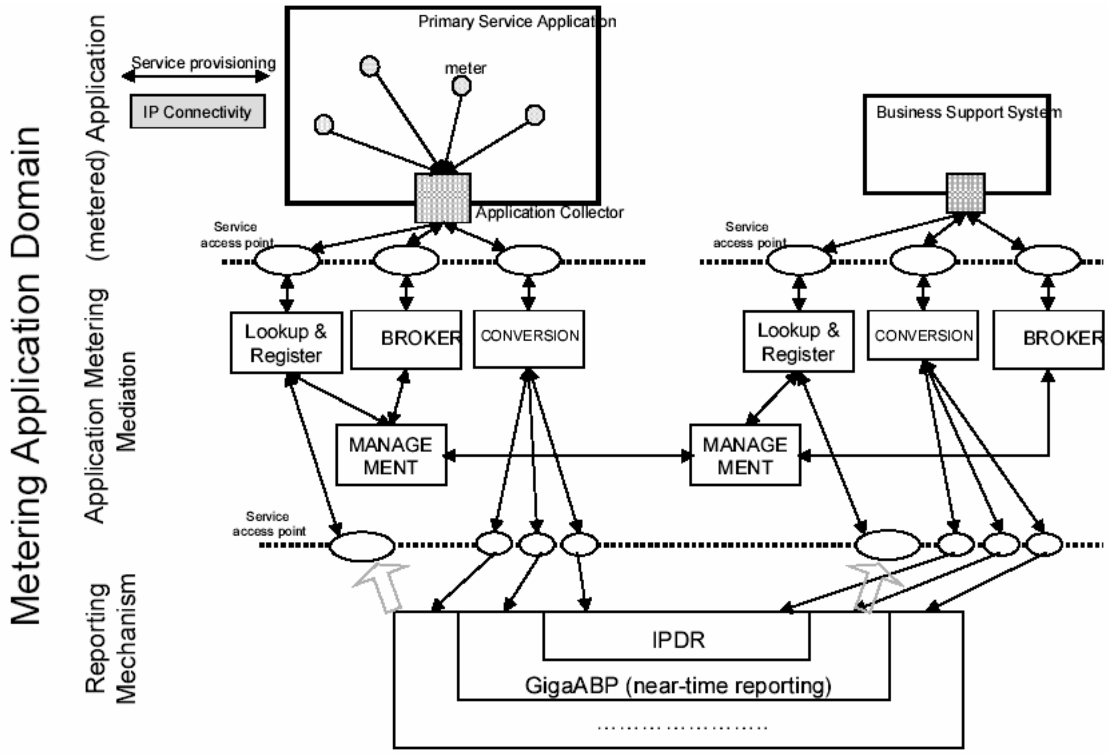


Figure 2-9: Functional Architecture of GigaABP - Application Metering Domain

flexible architecture for accounting, billing and payment systems, which can integrate the accounting at the network level, at the application level and at the product level. Customers will have a large freedom of choice in quality versus price, at different levels of services.

The GigaABP project developed an application metering infrastructure that can offer transparency of metering services to applications. Figure 2-9 shows the functional architecture of application metering domain of the GigaABP project The Metering application layer contains the applications for generating and processing metering data. The Application Metering Mediation layer contains functionality to enable applications to be transparent for particular metering components. The Reporting Mechanism layer identifies different reporting mechanisms, e.g. the IPDR mechanism or the real-time reporting mechanism.

The project also implemented an accounting and billing system that presents service for streaming Video-Over-IP. Detailed description of the experiment can be found at [TOKM00]

[18] www.telin.nl

2.3.4.6 The Reverse Charging Architecture of ING

Internet Next Generation[19] (ING) is a collaborative project between Center for Telematics and Information Technology[20] (CTIT) of the University of Twente (UT), Ericsson EuroLab Netherlands BV[21], the Telematica Instituut and KPN Research. The Work Unit 5 (WU5) of the project focuses on Internet accounting.

The Reverse Charging[22] (RC) architecture shown in Figure 2-10 enables Internet Service Providers (ISPs) to charge customers of other ISPs for data that is transferred to these customers. The architecture introduces a Trust Third Party (TTP) that has bilateral trust relationships with the both the client ISP and the server ISP. Trust is conveyed by means of trust certificates. The certificates use cryptographic techniques to ensure authenticity and non-repudiation of authorizations. The tasks of the TTP are:

- To periodically issue a trust certificate to each ISP;
- To periodically settle the financial balance between all participating ISPs.

The user scenario of reverse charging goes like this: the customer sends a request to the server asking for content. The server notices that the requested information should be reverse charged and sends out authorization request to the customer to make sure the customer will pay for the transmission and maybe for the content. The RC agent on the client system will present the request to the customer and forward it to the authorization device within its ISP's domain. The authorization devices store the authorization request for billing purpose and configure the access router of the customer to accept and forward PC traffic coming from the server. At the same time, it forwards the authorization information to the authorization device in the server ISP domain, where the information will be stored and the access router of the server will be configured to be able to forward the RC traffic coming from the server. The server will then be notified that authorization from the customer has arrived and it can start sending RC content to the customer. [SPRE00]

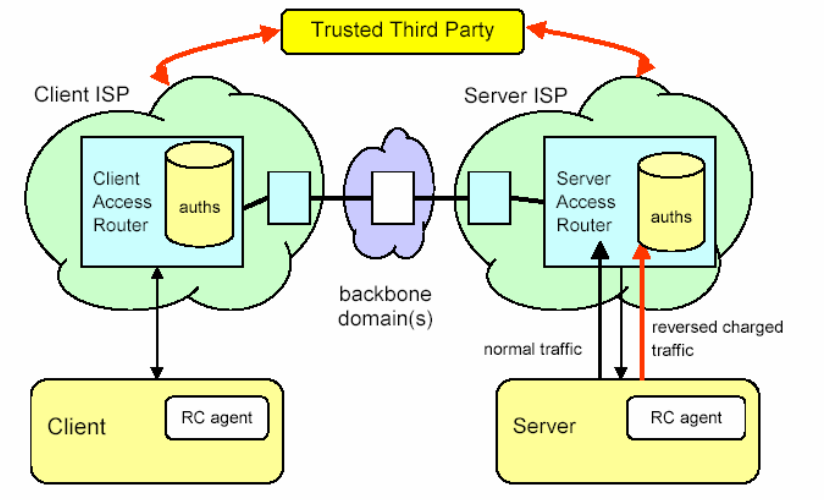


Figure 2-10: Reversed Charging Architecture

[19] http://ing.ctit.utwente.nl/
[20] http://www.ctit.utwente.nl/
[21] http://www.ericsson.com/se/
[22] Here we follow the original terminology used by RC architecture to keep the consistency.

The Provider Based Accounting architecture (PBAarch) is the evolution of the Reversed Charging (RC) Architecture. The purpose of the PBA architecture is to allow customers to pay their own ISPs for transport and RC content delivered to them from various content providers, while the content providers will receive their payments from their own ISPs. Detailed information about PBA arch can be found at [PARH01]

2.3.4.7 *The Charging and Accounting Architecture for Secure VPN*

The charging and accounting architecture for secure VPN is part of the IST project INTERNODE[23] [24] . Its goals are to design, specify and implement a platform to be used to create multi-domain S-VPN services for mobile users. [TPGS02]

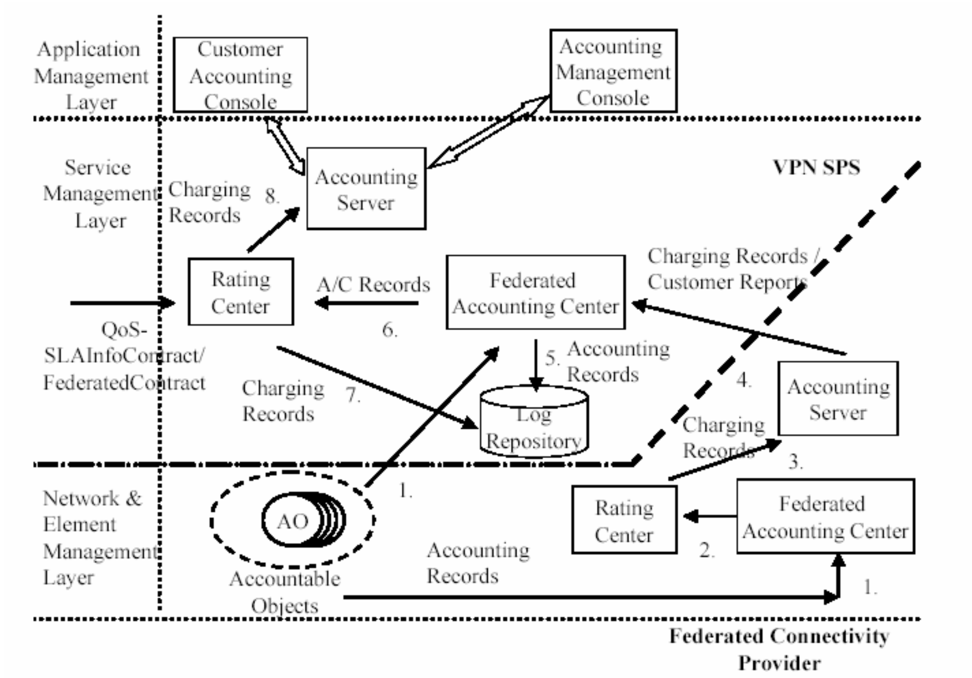


Figure 2-11: The Charging and Accounting Architecture of Secure VPN.

The main building blocks of the architecture as shown in Figure 2-11 are:

Accountable Objects (AO): For each new S-VPN creation, AOs are associated with the corresponding Mobility Agents (MAs) and the Security Gateways (SGs) for this S-VPN, while an other AO is associated with the S-VPN service itself. The AOs associated with MAs and SGs collect the accounting information associated with the VPN and forward this information to the AO of the corresponding VPN service. the AO associated to the S-VPN service forwards the accounting events to the Federated Accounting Centers both of the federated Connectivity Provider (CP) and of the VPN Service Provisioning and Support Platform (SPS).

Federated Accounting Center: receives all accounting information collected by the AOs associated with the VPN services provided. Also, it categorizes usage and charging information on a per customer basis, produces Usage records, and forwards them to the Rating Center. Furthermore, it configures AOs according to

[23] http://www.internode.org/
[24] Here we follow the original terminology used by RC architecture to keep the consistency

the accounting policy it also store the accounting information in a database called "Log Repository for future use.

Rating Center: takes accounting information from the Federated Accounting Center and produces Charging Records taking also into account both the SLA/QoS violations and the charging scheme for the offered SLA.

Accounting Server: serves as the reference of the accounting architecture to external software components. Accounting Server either forwards the charging information to the Federated Accounting Center of the VPN SPS according to the federation agreements (in case that it belongs to the federated CP); or it produces the bill and forwards it to the customer of the VPN SPS . [THAN02]

2.3.4.8 Charging Platform Architecture from Lancaster University

The charging platform architecture[25] [CUSH01] proposed by Distributed Multimedia Research Group of Lancaster University UK. The purpose of this platform is to facilitate the development of charging and billing system by automating the process which presents the captured data.

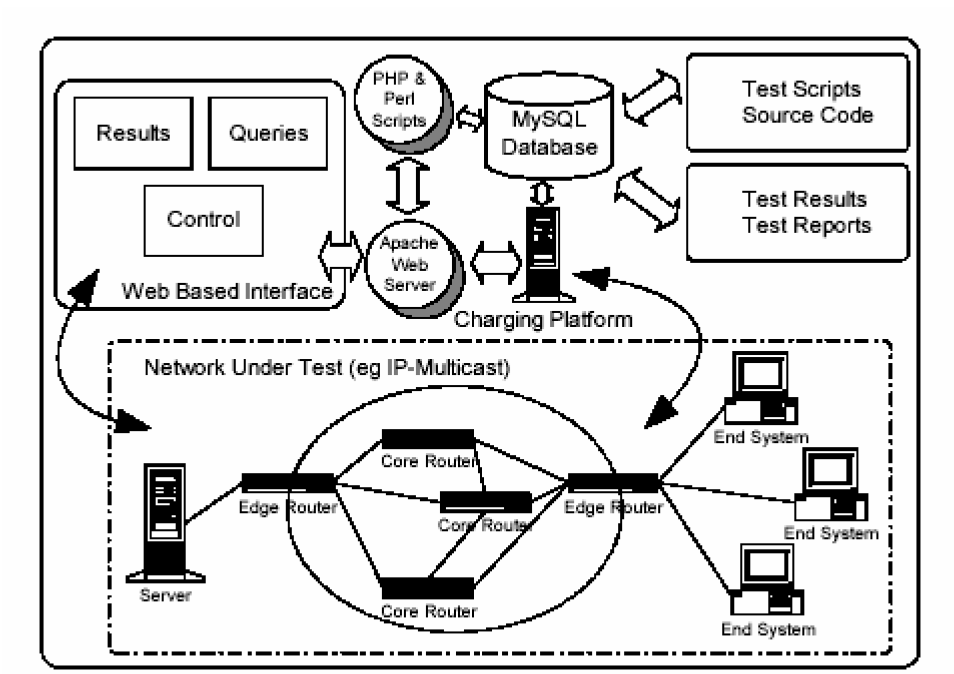


Figure 2-12: Charging Platform Architecture

The charging platform (Figure 2-12) includes an SQL database back-end, together with a web server front end. The web server is used for visualization and processing of the database contents. Scripting languages such as Perl and php are used as the main interface to the databases. These enable the rapid prototyping of experiment configuration and result presentation.

2.3.4.9 Others Architectures

There are also many architectures developed by different organizations for different projects, among them are:

[25] Here we follow the original terminology used by RC architecture to keep the consistency.

- The Arrow architecture [FANK99]
  Arrow is a proposed architecture for an accounting infrastructure under IPv6 environment, the main goal of it is to serve as a platform for experimenting with different pricing schemes.

- Cisco NetFlow[26]
  Cisco NetFlow has a four-tier structure, which consists of Data Export, FlowCollectors, Server and End-user applications. The services NetFlow can provide include detailed data collection and access lists processing for packet filtering and security services. NetFlow provides customer applications such as Accounting, Network Planning and Analysis, Network Monitoring, etc.

2.4 Data Representation and Data Transfer

In this section, we introduce standards and protocols which are used in accounting related data representation and data transfer.

2.4.1 Data Representation

Accounting related data are represented in certain formats, an extensive examination of existing accounting related data formats can be found at RFC 2924 [BROW00]. Currently, the IP Detailed Record (IPDR) is gaining a lot acceptance. IPDR can be seen as the corresponding part of CDR in the Internet accounting area. Released by IPDR[27] consortium, IPDR is aimed to be the future standard for exchanging accounting related data between different providers and different accounting systems. One of the key goals is to define a common usage record format and exchange protocol to facilitate the flow of usage information from IP network elements managers to support systems.

IPDR record format is released in the framework Network Data Management - Usage (NDM-U) for IP-Based Services. The latest version of NDM-U is v3.1.1. The IPDR NDM-U high-level reference model as shown in Figure 2-13 defined three layers [NDMU02]:

- Network and service element layer (NSE):
  The NSE layer consists of all the network and service elements required to provide an IP-based service to a given customer. In addition to physical devices, the systems that configure and manage such devices are considered part of the NSE layer

- Mediation layer:
  A mediation system provides a single interface to BSS systems that provides all network usage data and often a single interface for service elements provisioning.

- Business support systems (BSS) layer:
  The BSS layer consists of the systems deployed by a Service Provider or provider to support IP business operations.

[26] http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.htm
[27] http://www.ipdr.org

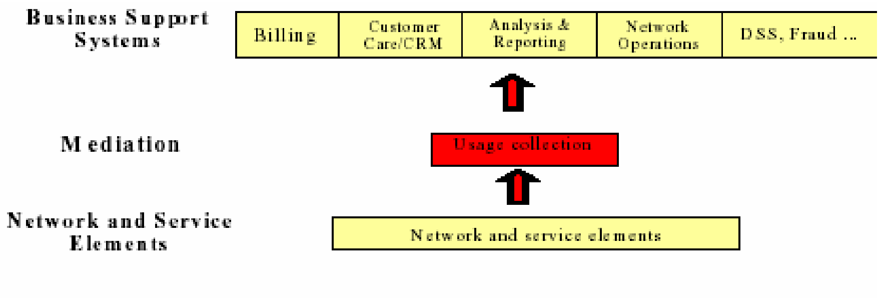


Figure 2-13: IPDR NDM-U High-level Reference Model

A typical IPDR record consists of five attributes: who, what, where, when and why. Table 2-2 provides detailed descriptions of these attributes.

who	Who is responsible for the usage (user ID)
when	End Time or Event Time
what	Service Usage measures / quantities Ex: Bytes, packets, flows, hits, transactions, time duration... QoS measures State information Event code (logon, logoff, threshold exceeded)
where	Traceability / Context Source Identifier Destination Identifier Service Element identifier (originator)
why	Event trigger type – (i.e., why is the network and service element reporting this data?)

Table 2-2: Attributes of a Common IPDR Record

Note: In addition to the "5Ws" defined above, each record may include reference pointers to other IPDR records that either capture related usage information, or contain usage information that was used to create the given record.

The NDM-U reference model (Figure 2-14) defines a set of interfaces for exchanging IPDRs between NDM-U-enabled devices or systems.

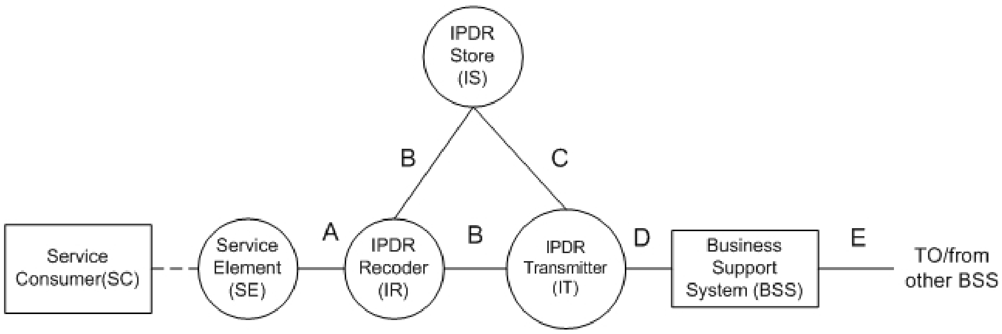


Figure 2-14: NDM-U Reference Model

- *Service Consumer (SC):* the human or machine initiating requests for services from the Service Element (SE).

- *Service Element (SE):* this is the set of equipment and software that provides a valuable service to a Service Consumer. The Service Element provides access to services and a requested resource, authenticates Service Consumers, authorizes access, performs accounting measurement for resources provided, provides services requested by Service Consumer, and performs accounting measurement for services provided. The IPDR reference model applies to any type of Service Element that is capable of generating accountable usage records

- *IPDR Recorder (IR):* Mediates proprietary protocols and data transactions from a Service Element and produces IPDRs resulting from the transformation of that proprietary data. It is through this entity that IPDR hide the heterogeneity date format of the underlying service element.

- *IPDR Store (IS):* this entity provides persistence to the IPDRs recorded by an IPDR Recorder.

- *IPDR Transmitter (IT):* delivers IPDR documents to Business Support Systems.

- *Business Support System (BSS):* receives information contained in IPDR Docs from an IPDR Transmitter, processes the information contained in the contained IPDRs for use in the commercial activities of a Service Provider, and presents information for transmittal to other Business Support Systems.

IPDR provides two schemes for encoding IP usage records. One is a text-based XML encoding and the other is a compact encoding based on XDR. A one-to-one mapping exists between each encoding scheme. In both schemes, a single Master IPDR Schema Document declares elements common to all IP-based Services; a service-specific schema document is then used to define the usage attributes which describe usage events for a particular type of service. The "Service Specification" now includes: e-mail, streaming media, voice over IP, etc and can be found at the web site[28] of IPDR consortium.

2.4.2 Data Transfer Protocols

TAP3 -Transferred Account Procedure 3
Developed by GSM association[29], the standard provides telecommunication operators with a capability to bill new value-added services such as Multimedia Message Services (MMS) and mobile Internet access (including GPRS). It allows the transfer of Call Detailed Records (CDRs) during roaming therefore enables the operators to exchange billing information for roaming services

The GTP' protocol
The GTP' protocol designed for GPRS accounting related data collection has been derived from the GTP (GPRS Tunnel Protocol) protocol (defined in GSM 09.60) which is used for packet data tunneling in the GPRS backbone network.
The GTP' protocol has been designed to deliver GPRS CDRs to the CGF(s) from those network elements or functional entities generating charging records.

---

[28] http://www.ipdr.org/service_specs/index.html
[29] http://www.gsmworld.com/index.shtml

Note: The recommended protocols between Charging Gateway and Billing System are FTAM protocol on X.25 or TCP/IP, and FTP over TCP/IP.

SNMP - Simple Network Management Protocol
The Simple Network Management Protocol (SNMP) was designed as a general network management protocol rather than a specialized accounting protocol. It is widely used for inter-domain accounting purpose due to accounting-related MIB modules. SNMP Support for notifications makes it possible to implement the event-driven, event-driven polling and event-driven batching models. This makes it possible to notify domains of available data rather than requiring them to poll for it, which is critical in shared use networks and roaming. SNMPv3 provides enhanced secure environment for the management of systems and networks. However, SNMP-based accounting has limitations in terms of efficiency and latency that may make it inappropriate for use in situations requiring low processing delay or low overhead. This includes usage sensitive billing applications where fraud detection may be required. [ABOB00].

2.5 Case Study: I-Mode™

“I” stands for Information. I-Mode[30] is the world’s first full-color, always-on, packet-switched, Internet service for cellular phones offered by NTT DoCoMo[31] (Nippon Telephone and Telegraph DoCoMo - *doco mo* means "anyplace you go" in Japanese and the acronym stands for "Do Communication Over the Mobile Network."). I-Mode has been a great success, with a meteoric 36,773,000[32] users since it was launched in February 1999 - adding 50,000 - 60,000 users/day. Now I-Mode is available around some European countries such as the Netherlands, Germany, Italy and Belgium. We look into the charging model of I-Mode and analysis

I-Mode uses Compact HTML (cHTML). Many websites can be easily and quickly reformatted. It is the ease of reformatting websites that has played a key role in driving the development of content and services.

The primary services can be split into four categories:

- Database (phone directory search, restaurant guide, dictionary service);
- Information (weather update, stock quotes, sports news, transport information);
- Entertainment (network games, fortune telling, downloading cartoon characters, TV guide);
- Transaction (banking, security trading, ticket reservation).

Many services are free (dictionaries services); some services are flat-fee: the subscribers pay a monthly subscription ($0.90-2.70) and can use it any time; the other services, such as making a purchase on stock market are charged on per transaction basis. For I-mode emails, users are charged on the amount of data they send or receive.

[30] http://www.nttdocomo.co.jp/english/p_s/imode
[31] http://www.nttdocomo.co.jp/english/index.shtml
[32] Data obtained from I-mode official web site on Feb 16th, 2003. http://www.nttdocomo.co.jp/english/imode/news/num.html

The charging model of I-mode is shown in Figure 2-15. Information providers charge users for certain services, either as a subscription charge or a transaction charge. NTT DoCoMo can collect these charges on provider's behalf (Step 1, 2 in Figure 2-15); along with it own packet communication rates. Data transmission charges for viewing sites are based on the amount of data transmission. Packet transmission charges are applied even if data could not be received properly due to transmission conditions.

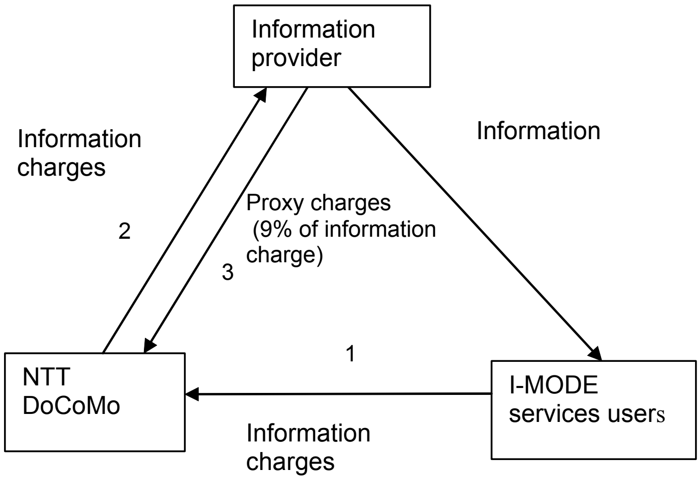


Figure 2-15: The Charging Mode of I-Mode

NTT Docomo earns considerable amount of money in proxy fee. This comes from fees collected from information providers who pay a 9% commission fee on revenue earned from service charges to customer over I-mode (Step 3 in Figure 2-15). NTT Docomo operates the billing procedure.

The lessons of I-MODE for other countries [EMCD00]:

- Ease of reformatting websites;
- Open standards;
- Lowering cost of entry;
- Pricing of content and services;
- Diversity of content and services;
- Services tailored to locality.

However, we should notice that, the success of I-Mode is partly due to the unique situation in Japan. The high concentration of population in urban areas, long commute time, consumer comfort with small electronic devices, and the lack of a ubiquitous fixed-line Internet infrastructure. There are only few telecommunication operators in Japan, and in this monopoly environment, application or content providers have few choices but to provide services through NTT DoCoMo. Therefore, this is unlikely to happen in Europe where there are lots telecommunication operators facing fierce competition.

Imagination is the beginning of creation. You imagine what you desire, you will what you imagine and at last you create what you will.

-- George Bernard Shaw

3 The Architectural Design of the Accounting facilities

This chapter describes how we design the accounting facilities for the Mobile and Wireless Internet Environment. By following a specific design methodology we come out with an architecture of the accounting facilities represented in the form of a "reference model", which will serve as guidance for later implementation.

This chapter is organized following the steps designing a "reference model": Section 3.1 articulates the design methodology we will use in this project. Section 3.2 outlines the scope and gives assumptions of our design. Section 3.3 explains the mechanisms in our design for a real-time system and for prepaid users. Section 3.4 discusses the actors and roles in the system. User scenarios are developed in section 3.5. The discovery of service and the reference model is in section 3.6. Section 3.7 defines the interfaces.

3.1 The "Reference model" Design Methodology

By architecture we mean the specification of the overall structure, logical components and logical relationship of the accounting system. The architecture will offer a blueprint for the concrete implementation of the system. In this section, we design our accounting architecture by following the "Reference model" design methodology that has been introduced in Biemans' book "Reference Models for Networked Applications" [BIEM02]. This methodology identifies the following main steps:

- Define the scope of the system – Define the boundary of the system, what will be included and what will not.
- Define user scenarios – Textual description of how people use the system; description of roles, actions on the system and responses from the system.

- Define the related roles – Description of relevant roles based on the scenario.
- Discover the services – Discover the main services of the system, based on the scenarios.
- Define the building blocks – The building blocks provide services, and have responsibilities
- Define the interfaces – The definition of interfaces is based on relevant standards and existing building blocks.

The design process, however, is not strict. The sequence of the steps can be changed under different conditions. For example, the definition of actors can precede the definition of scenarios. It depends on which parts are the clearly defined at the beginning, and then we can follow our own steps to discover the unknown part. We will adjust this design methodology according to the conditions in our project. The change will be mentioned later in this chapter.

3.2 The Design Scope and Assumptions

3.2.1 The Scope of the Design

Our design will focus on the generation and integration of multi-domain accounting information in a Mobile and Wireless Internet environment as it has been envisioned by the 4GPLUS project.

In this project, we are mainly concerned with the following issues:

- Accounting data generation and collection

This is how we meter the resource consumption/service utilization and collect accounting data (usage records and charging records).

- Data representation and exchange

This is mainly about how to represent accounting information, the format of the usage records and charging records, and the accounting interfaces between different functions/domains.

- Data integration

The accounting information may be generated across different domains but related to one service that the user consumed: data integration merges the accounting information from a same domain or different domains.

- Accounting for prepaid and postpaid users in real-time and non real-time.

Pre-paid users are closely linked with real-time accounting. The accounting for prepaid and postpaid has the trend to integrate; we will design a system that supports real-time accounting and non real-time accounting on both prepaid and postpaid users.

An accounting system involves wide areas and has many aspects. Due to the limited time and human resources, we will mainly focus on the unique topics that are related to next generation Mobile and Wireless Internet and leave the common topics in accounting system out of the scope. In this project the following aspects are not addressed:

- Database related aspects;
- Invoice generation and payment;
- The portable device related issues: such as user profile, operating system;
- Privacy and legal issues regarding the location and other personal information.

3.2.2 Assumptions

To enable a clear and common understanding on reasoning in various cases for or against certain solution proposals, a set of unique assumptions for the accounting architecture is necessary. The assumptions are valid with respect to technology, protocols, and user scenarios. The work on the accounting architecture requires the definition of the underlying technology to be considered with respect to their technical characteristics.

- In order for the user to maintain connectivity while roaming between different networks (WLAN, UMTS, and GPRS), the Mobile and Wireless Internet need to use Mobile IP. We choose Mobile IPv4. This implementation decision is based on the fact that MIPv4 is widely accepted as IETF RFCs and there are a lot of packages available. Furthermore, the implementation of 4GPLUS project using Mobile IPv4 enforces us the decision.

- Authentication, Authorization and Accounting (AAA) are closely related to each other. Here we assume that the different types of access networks adopt different authentication methods. For example, WLANs use certificates, IEEE 802.1X (EAP-TLS) plus RADIUS, which offers an effective framework for authenticating and controlling user traffic to a protected network, as well as dynamically varying encryption keys. GPRS networks use SIM-card based authentication.
  It is desirable that the user signs on just once to the SP to use the services provided by ANs and 3PSPs. The organization Liberty Alliance Project[33] has been formed to address this problem. The Liberty Alliance Project specification supports all kinds of identifiers provided by Identity provider (e.g. SP). It is doesn't matter how we choose the identity of the user as long as the identifier is unique in the system. We choose to use the format Username@SP-domain. (For example, when the user name is Bill and SP-domain is Gate.com, the user is identified as Bill@gate.com). This is a typical identification in the Internet communication area. In telecommunication systems, users are identified by their Mobile Station International ISDN Number[34] (MSISDN). A translation function between the two user identification systems is needed.

[33] http://www.projectliberty.org/
[34] The Mobile Station International ISDN Number is the standard international telephone number used to identify a given subscriber. The number is based on the ITU-T (International Telecommunications Union-Telecommunication Standardization Sector) E.164 standard.

- We assume that user identifications can be used to discriminate users. The accounting system of an enterprise network will not charge its internal user, but may charge the visiting external user when he/she uses the access network.

- An end-user can select ANs according to the information provided by the access networks, such as signal strength, cost, delay and available bandwidth. The selection of AN can also be automatically done by the mobile device according to the profile set by the user.

- Use IPDR to record the accounting information. The advantages of IDPR are explained in section 2.4.1.

- The 3PSP is able to support different payment system: macro-payment, small-payment and micro-payment. A micro-payment refers to a payment of approximately €1 or less per transaction/request, and in the Mobile and Wireless Internet environment this will often be for content, such as video downloads or gaming. Small-payment refers to a payment with an amount between micro-payment and marco-payment; usually between €1 and €10. Macro-payments refer to larger value payments such as online shopping or proximity-based payments. The distinction between the three types of payment is important since the security required for each will be different. For example, authentication for every macro-payment transaction through a trusted financial entity is extremely important, whereas network authentication, such as SIM, may be sufficient for micro-payments or small-payment that only uses the operator's infrastructure. [MPFO02]. We assume the 3PSP is able to select a payment system according to the amount of money, or provides both payment method and leave the decision to the users.

3.3 The Actors/Roles in the System

As it had been described in the Section 1.1.1, there are mainly four actors/roles involved in the next generation Mobile and Wireless Interne envisioned by the 4GPLUS project:

1. The End-User (EU);
2. The Service Platform (SP);
3. The Access Network (AN);
4. The Third Party Service Provider (3PSP).

As the WLAN technology (e.g. IEEE 802.11™) becomes mature and access points are being deployed widely, it can be foreseen that in the future, there will be hundreds of thousands WLANs owned by individuals or small wireless network operators or wireless Internet service providers (WISP) that are willing to contribute their networks, together with other types of access networks to form a seamless network environment.

It is impractical that the SP will have contract with each of the WLAN owned by individuals because of the cost. Therefore, it is necessary to introduce the role of Access Network broker to ensure the scalability of the access networks. The broker will establish trust and legal relationship with a large number of access networks. The service platform operator then establishes relationships with the access network brokers [JERO02]. From the perspective of access network, the broker acts as service platform; while from the perspective of service platform, the broker acts as a large access network. But the introduction of the role of broker does not introduce new functionality. The functionality of a broker can be realized by moving some of the functionality of SP and ANs into the broker's domain. So, in the design, we do not consider the actor/role of an access network broker.

Similarly, the broker role can be introduced between 3PSP and SP to alleviate the burden of SP when the number of 3PSP grows exponentially. Since the 3PSP broker's role can be realized by relocate some of the functionality of 3PSPs and SP, in order to simplify the design, we do not consider the role of 3PSP broker.

Further, the 3PSP can be divided into content provider, value-added applications provider and portal supplier. These roles will be treated as one role in our project because they share one common attribute: provide services to the end-users by utilizing the services the SP provides.

3.4 Real-time system and prepaid users

The accounting facilities should support real-time accounting. The distinction between real-time and non real-time lies in the frequency of the periodically occurring accounting activities. In a real-time accounting system, the generation of usage records, charging records happens more frequently. Correspondingly, the end users are charged more frequently (e.g. every 30 seconds). In a non real-time accounting system, these activities occur less frequently (e.g. every 30 minutes). We introduce a time window mechanism which will serve as guidance in the implementation of a real-time accounting system.

In this time window mechanism, there exists a time window and all the accounting related activities (e.g. charging requests) are carried out within the time window. For real-time accounting, the size of the window is very small; while for the non-real time accounting; the time interval can be set to large. The size of the window can be decided by the service provisioning entities (AN/3PSP/SPdynamically. This approach can significantly simplify the implementation of a real-time system.

Prepaid users are closely linked with real-time accounting. The service providers need to know the actual balance in real-time to prevent the user running out of it while still using the services.

In 4GPLUS platform, the SP knows if a user is a prepaid or a postpaid user, since it keeps a user database, but the AN/3PSP can not distinguish the user type from the user identifier. If an AN/3PSP treat a prepaid user as a postpaid user, it will not receive the compensation from the services it provides. One solution is the SP will tell the AN/3PSP about the user type, then they will adapt to the corresponding charging method (e.g. adapt real-time accounting for prepaid users). In this solution, the AN/3PSP need to keep separate database for prepaid and postpaid users. This will add burden to the AN/3PSP.

Nevertheless, there is trend to converge the prepaid and postpaid users to simplify the management and reduce the cost. Here we bring in another solution, which introduces a money/credit reservation mechanism. The AN/3PSP reserves certain amount of money/credit from the SP, and charge on the reserved money. When the reserved amount/credit has been used up, it will reserve again. However, when the reservation is excessive (the user does not use up the reserved amount before he roams to another network), AN/3PSP will release the extra amount. For a postpaid user, the AN/3PSP can always make a reservation because the SP will pay for him/her. But for a prepaid user, the user needs to pay the SP certain amount of money in advance. If a prepaid user runs out of his balance paid to a SP, the reservation cannot be confirmed; correspondingly he will not access to the services. Using this mechanism, there will be no distinction of a user type in the AN/3PSP domain. The difference only lies in the SP domain: a postpaid user has relatively more freedom to use the services while the prepaid user has to make sure that he/she has enough credits. This solution also reduces the frequency the charging requests are sent. However, it is up to the AN/3PSP whether to reserve money/credits in advance or not. We adapt the second solution in our design.

3.5 User Scenarios

In this section, we provide three scenarios which enable us to look into the details of the accounting facilities.

3.5.1 Video on Demand

Suppose a man is waiting at the boarding gate of the airport. He has to wait for an hour before the boarding and he has nothing to do. Feeling bored, he takes out his mobile device and accesses a 3PSP, which provides jokes and funny videos. The text jokes are free of charge but soon he gets tired of reading and he begins searching for the video clips. There is one video clip named "Gone Nutty" that catches his interests, so he selects the video. The following information appears on his mobile device:

- The video clip will cost you €0.30
- The traffic will cost you €0.05
- The total amount €0.35 will be paid by your SP, do you accept it?

When the user agrees upon the payment, the video will be streamed to the user's mobile device. The funny video clips make the man laugh, and the one-hour of waiting is killed by laughers.

What lie beneath the user's simple clicks are:

- When the user enters the airport, his mobile device automatically switches from the GRPS network to the WLANs the airport providers. (This seamless roaming from GPRS network to WLANs is enabled by the Mobile and Wireless Internet environment, the selection of the AN is automatically done).
- The pricing information of the access network will be shown at the corner of the screen of the mobile device. (e.g. €0.01/MB) (pricing info)
- The user begins surfing on the Internet and finds the 3PSP. When he clicks on the video clip, the "Gone Nutty", a "video request" will be sent to 3PSP. ①
- The 3PSP will send a request to the SP. This request includes: what kind of access network the user is attached to, QoS information of the AN (e.g. bandwidth, delay), the pricing of the AN (e.g. €/MB). ②

- The SP will forward the request to the WLAN the user is attached to. ③ The WLAN will reply to SP (e.g. AN type: WLAN – Wi-Fi, bandwidth available: 100k/s; delay 0.01s, price €0.01/MB). ④

- The SP forwards the reply to the 3PSP. ⑤ .The 3PSP then figures out the sending rate of the streaming video from the information (e.g. if it has different quality of video: 30kb/s, 100kb/s, 300kb/s etc and the price for different streaming rate differ in several cents). There is also a computation done on how much it will cost for the transportation of the content (e.g. the video clip is 5MB in 100kb/s streaming video, so it will cost €0.05). The calculation could be quite complex due to the different charging models of the AN. The price of the video clip is €0.30.

- Because the small amount of money (less than €10), the 3PSP decides the money will be paid using micro payment system. The 3PSP will then send the previously mentioned message to the end-user. ⑥

- The end user sends a confirmation that he wants to pay. ⑦

- When the 3PSP gets confirmation from the user, it reserves certain amount of money/credits from the SP (e.g. €0.35, partly from itself, partly for the AN).

- The SP sends a reservation confirmation to 3PSP and also inform the AN that there is a reservation for it. ⑨

- When the 3PSP receives confirmation from the SP, it will begin the streaming of the video.

- The WLAN gathers the information about the network usage and charges the SP using the reservation ID it acquired from the SP. (Besides the description of the network usage, the information will contain the user ID and other information for the SP to integrate records coming from different domains.) In this case, the full length of the information is sent, the AN will charge for €0.05.

- The 3PSP will have some monitoring mechanism to meter the content it sends. After having finished sending the video clip, it will charge the SP from the reserved amount using the reservation ID. (The accounting information will also include the user ID, description of the content and other information for the SP to integrate records coming from different domains). In this case, the full length of the video is sent, and 3PSP charges for full price: €0.30.

- The SP pays for the user to the AN and 3PSP. Then it generates an integrated record for the user for the video clip "Gone Nutty". The amount will be €0.35. It will be an item in the bill to the user.

Similar application can be thought for a voice over IP scenario (VoIP).

The actually signaling flow and data flow is shown in Figure 3-1 and Figure 3-2.

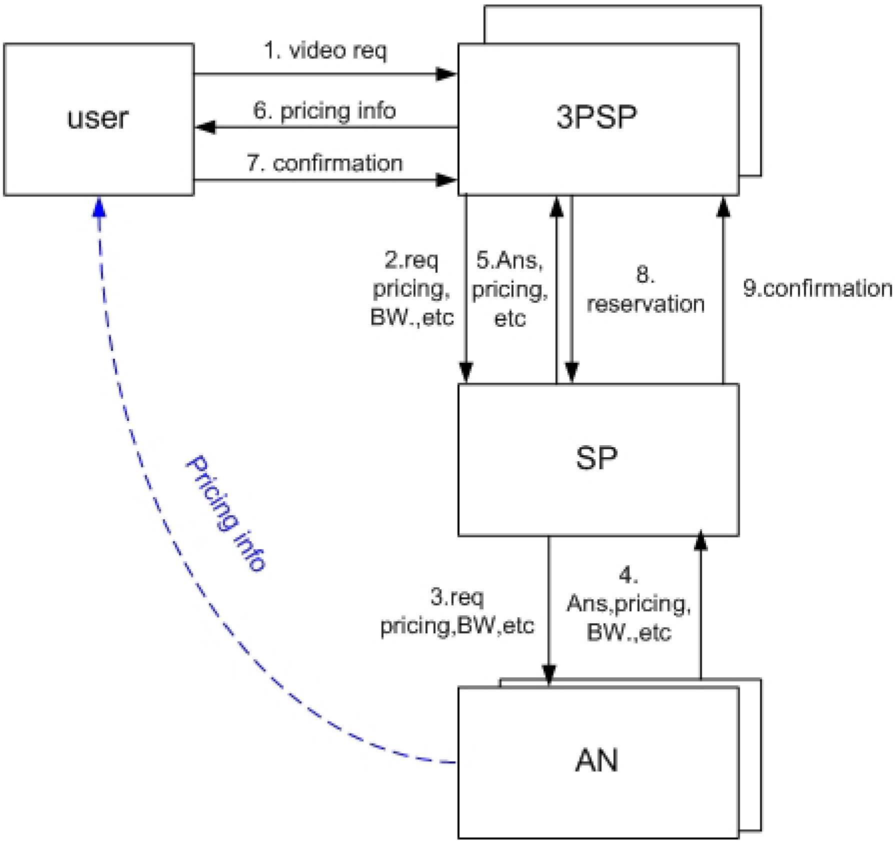


Figure 3-1: Message Flow of the Video on Demand Service.

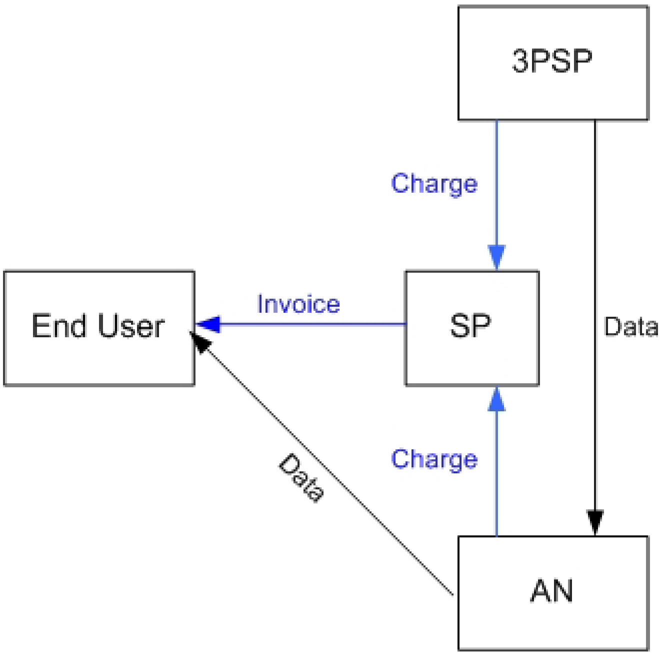


Figure 3-2: Data and Charging Information Flow of Video on Demand Service

3.5.2 M-health

Let's suppose that Luis is an old man who has brain and heart problems. He sometimes falls unconscious, a life-threatening situation. He is living alone and he has some friends in nearby cities. Knowing his own limit but not willing to stay at hospital, Luis wears a Body Area Network[35] (BAN) and is subscribed to an M-health service[36] under the requirement of his insurance company. (This needs him to better to be a postpaid user). He also has a multi-interfaces mobile phone.

On a sunny day, Luis set out to visit a friend in another city. While walking from the bus stop to his friend's house - a quiet path without many people walking around -, Luis suddenly falls unconscious before he can dial the emergency number. The BAN Luis wearing detects abnormal behaviors of the blood pressure and heart rate and triggers the M-health alarm service in his mobile device. In 5 minutes, the ambulance arrives with healthcare practitioners. Luis's life is saved.

The mechanism behind this successful rescue is:

- When Luis walks from the bus stop to his friend's house, his mobile device handovers from the bus stop's WLAN to UMTS network, which is facilitated by the Mobile and Wireless Internet environment.

- When the BAN triggers the M-health services in Luis's mobile device, it starts a session and begins sending alarms to the M-health service provided by a 3PSP carrying Luis' identity. ①.

- The 3PSP receives alarms; It sends request to SP asks for Luis' location information ②; the SP directs the request to the AN that Luis' mobile is attached to ③.

- The AN gives the location information of Luis. ④ And charges the SP for the location service it provides (e.g. €0.01/request). The SP then forwards the information to the 3PSP. ⑤

- When the 3PSP receives location information of Luis, it will send a confirmation back to the mobile device to inform the M-health service that the alarm is being processed.⑥. Otherwise the mobile device will keep sending the alarms until it gets some confirmation from the 3PSP.

- When the M-health service receives confirmation, it will trigger the BAN to sends vital signals like heart rate, blood pressure, et cetera to the 3PSP through Luis' mobile device continuously as long as needed.

- Upon receiving the emergency alarm, medical specialties from the nearest hospital will be sent to the location. Routing information will be given to the mobile device of these medical stuffs: the hospital will be the starting point; the destination will be the location of Luis. Their mobile device will be added to the session that Luis' device initiated for accounting purpose.

- The 3PSP keeps asking the Location information of Luis and the ambulance every 20 seconds. New routing information is provided according to Luis'

[35] A collection of (inter) communicating devices which are worn on the body; providing an integrated set of personalized services to the user.

[36] M-health service is based on a mobile telematics system that operates as an ambulatory health monitoring tool and/or ambulatory health therapy tool and delivers a professional healthcare service and/or personal health service to its users.

location and the ambulance's new location. The location of Luis is used to predict when the ambulance will arrive at the ER. In case when someone finds him before the ambulance arrives, the 3PSP can keep track of Luis' movement.

- Once the medics reach Luis, they can terminate the M-health service.

- The AN charges the SP for the each Location service and traffic it sends based on the session ID (can also includes the service used by the medical stuffs). The 3PSP charges the SP for the service based on the session ID and the SP generates the integrated bill based on the session ID for the user.

The signaling flow and charging information flow are shown in Figure 3-3 and Figure 3-4.

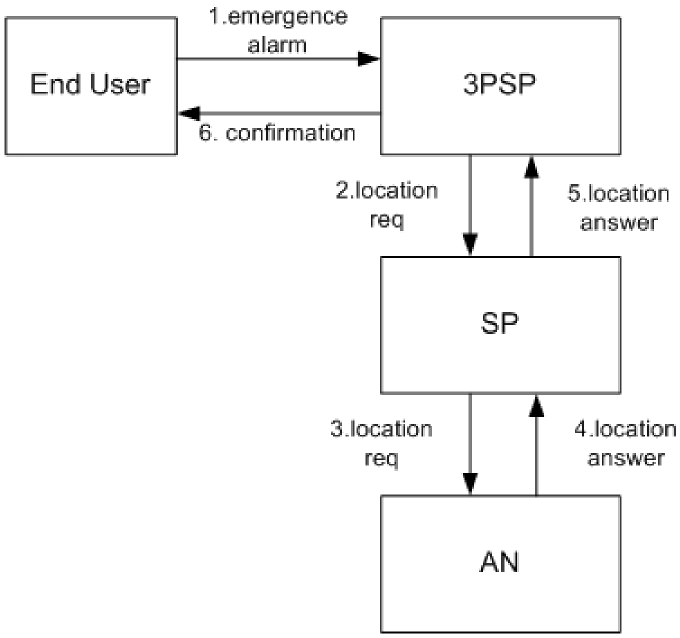


Figure 3-3: Message Flow of the M-health Service

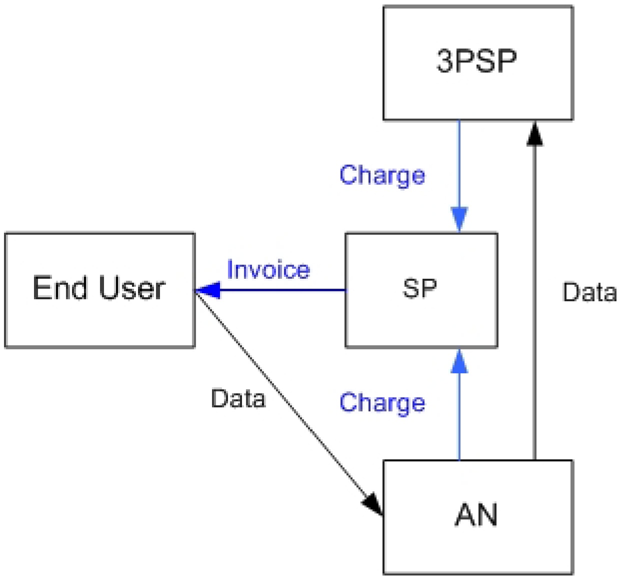


Figure 3-4: Data and Charging Information Flow of M-health Service

In this scenario, location information is one of the key aspects to save Luis' life. There are also a lot applications/services that are based on the user location information, among them are parking lot/restaurant hunting, city guidance, mobile dating service, routing guidance for drivers, etc.

3.5.3 *Subscription Based Services*

Suppose everyday Jill needs to drive 30 kilometers to her office. One morning, when she gets up, she receives a message on her mobile device that it will snow heavily. Jill decides to leave home earlier in case she is caught in the heavy snow. Just as she pulls the car out of the garage, she receives a message that there is long queue to the high way. So she decides to change her route. On the way to office, she receives an alert that there is a kind of e-mail worm spreading on the Internet. When Jill is in the office, she updates her anti-virus software before she opens her email box. There are two emails that already have been infected by the virus, but the virus is safely removed by the anti-virus software. During lunchtime, Jill gets a message that one of her stocks has reached the ideal price she has set before and she sells it. In the afternoon, Jill receives some beauty tips in her mobile device and she decides to buy some cream after work. At the shopping center, Jill receives a message that Laura Pausini is going to give concert in her city next month and she immediately books the ticket and gets a nice seat. What a day!

The mechanisms for this type of service as shown in Figure 3-5 are:

- Jill subscribes the services from the different 3PSPs and pays a monthly fee. For example: €1/month for to a 3PSP. The services can be weather alert, news alert, stock alert, traffic alert, music information or jokes.

- When there is information to be sent, the 3PSP pushes the information to the SP, SP will send the information to the AN to which the subscriber is attached to. The AN can be home a WLAN, public WLAN, UMTS or GPRS networks. Because this kind of information is generally short, the AN will charge SP for a fixed amount of money, (e.g. €0.001/message). This will be added to the user's accounting information later.

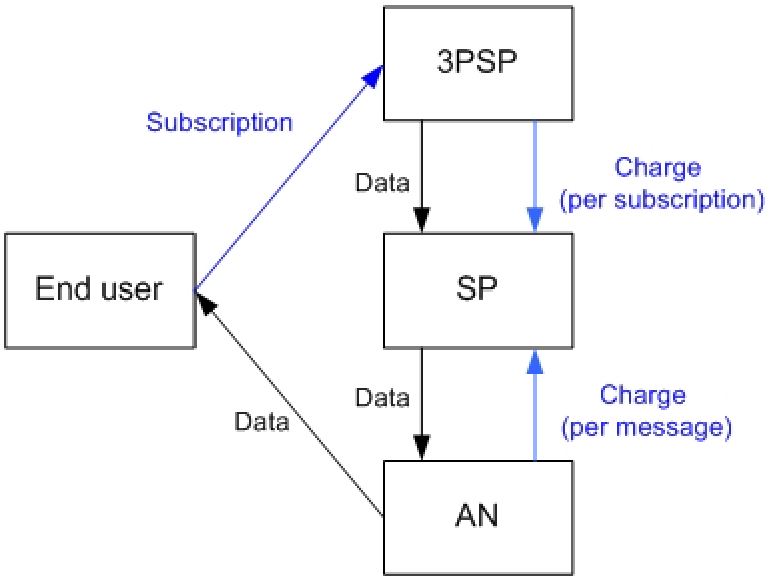


Figure 3-5: Subscription Based Services

3.6 Discovery of Services and of Accounting Reference Models

3.6.1 The "One Bill" Service and the Reference Models

While there are many ways to look at the accounting system, the user's point of view is essential. From the end-user's perception, the ANs, SP and the 3PSPs are an integrated part that provides services to him/her. Apparently, the user does not want to get bills from each access network his device has been attached to, neither does he want to get bills from each 3PSP he receives services. It is the goal of the accounting facilities that to provide the end-user with "One (Unified) Bill", in other words, the accounting facilities provide the end-user "One Bill" service. From the user's perception, the "One Bill service" is comparable with the I-Mode service (section 2.5) where the user receives one bill from KPN on usage of the network, the I-Mode services and subscription. But in the Mobile and Wireless Internet envisioned by 4GPLUS platform, he may use the services of numerous access networks and third party service providers and still get "One Bill". From the SP's perception, it has to integrate the accounting facilities information generated in different administrative domains and based on the different charging models.

In a traditional accounting system, metering, data collecting, charging and billing functions are in the same domain. Therefore, reference model[37] [PARH02] of traditional accounting system is straightforward. Even in a roaming situation, when the user roams between two different administrative domains, the accounting relationship only exists at charging/billing level (e.g. the charging of a roaming user between two GSM operators). This is illustrated in Figure 3-6.

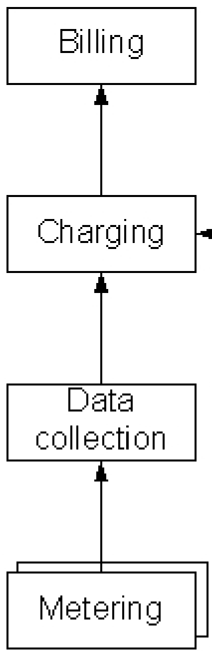


Figure 3-6: Traditional Accounting Reference Model

In the Mobile and Wireless Internet environment, the traditional "one" service-provisioning domain is divided into three domains. Although to provide the "One

[37] Reference model is a model used as a point of reference, e.g. in designing concrete applications. It describes how systems can be composed by putting together "building blocks".

Bill" service, functional building blocks like metering, data collecting, charging and billing are still needed, but the distribution of these building blocks is different. The 3PSP domain needs to charge the user for the services he/she uses; therefore, the metering function, data collecting function and charging function in 3PSP domain are required. The same goes for the AN domain. The information generated by different functional building blocks needs to be integrated in the SP domain. Ultimately, the user will receive one bill from the SP.

In this thesis, we will not define the interactions between building blocks in different domains as services. We tend to treat them as distributed building blocks that together provide the "One Bill" service to end-user. We try to distribute the building blocks into different domains and come out with a new reference mode for the service provisioning part. The ender user domain has the functionality of invoking and utilizing the services. However, we do not consider end-user domain in this model and leave the design of these functions open, while with more emphasis put on how to generate one bill in the service provisioning part.

Our reference model (Figure 3-7) will mainly identify three domains: the AN domain, the SP domain and the 3PSP domain.

By distributing metering, data collecting and charging functions into AN and 3PSP domains, the AN and the 3PSP can use any charging models they need and change them without affecting other domains. The dash lines between charging models and functional building blocks in Figure 3-7 indicate charging models only affect the behaviors of functional building blocks within a same domain.

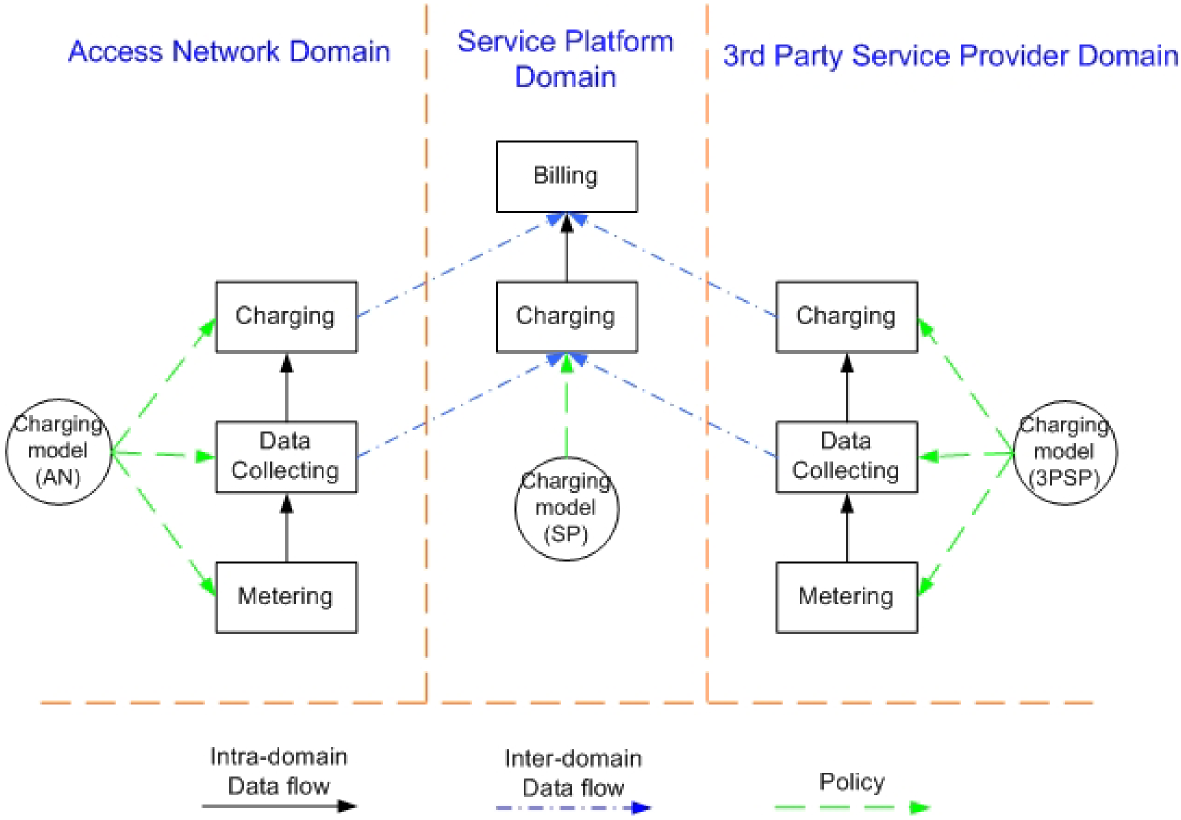


Figure 3-7: Reference Model of the Accounting Facilities

The Access Network Domain:
The functions in the Access Network domain are mainly focused on metering and

data collecting of the services it provides (we will treat data transportation as service). For small AN operators (e.g. WLAN hotspots), charging functions are expensive and impractical to be placed in their own domain. For large AN operators (e.g. traditional network operators), they already have their own charging functions when they join the service platform and want to keep it (e.g. a GPRS operator). The other reason for ANs to have their own charging functions is that they can apply their own charging policies and keeps them private and flexible. Because the goal is to generate one integrated bill for the user in the SP domain, the AN domain will not contain a billing function.

Metering: The metering function in AN records the consumption of resources and the utilization of services of a user. The metering policy is based on data collecting policy. The data collecting function will periodically collect the metering data. We considered enabling the SP to access to the metering data, but this turned out to be impractical. It is obvious that each AN wants to keep the metering private because of security reasons. AN operators also tend to keep the data collecting and charging functions to themselves when possible. In the model we propose, the AN can have full control on the charging model it prefers and it can keep the metering data private.

Data collecting: This function provides usage records to the charging function within the same domain or charge the SP based on the usage records - in case of the AN does not have the charging function itself. This function will collect the metered services (e.g. location service, data transportation) the AN provides to the end users and the SP. The data collecting policy will be based on charging policy. The data collecting function creates the usage records and is responsible for storage of the records if necessary.
The line between Data Collecting (AN domain) and Charging (SP domain) in Figure 3-7 represents a charging relationship in the situation that the AN does not have its own charging function. The AN domain charges the SP domain based on the usage records. The records contain raw data, which maybe represents the volume of traffic the user generated, or the amount of time the user has stayed in the AN.
Correspondingly, the line between Data collecting (AN domain) and charging (AN domain) in Figure 3-7 represents the situation that the AN has its own charging function and can generate the charging records itself.

Charging: To charge the SP based on the charging records. The charging function in the AN domain is optional because it can also be placed in the SP domain. The charging function in AN mainly calculates the resource consumption and service utilization using functions determined by the charging policy. It will try to integrate charging records if the records share some common characteristics (e.g. bear same session ID of a user). The charging function sends charging requests to the SP. This is represented by the line between Charging (AN domain) and Billing (SP domain) in Figure 3-7.

The Third Party Service Provider Domain:
We can see from the reference model that the functions in the 3PSP domain are symmetric to those in the AN domain. It also has metering, data collecting and charging functions. But these functions are services/content/application oriented. For the same reason as in the AN, that some 3PSPs outsource the charging function to the SP: to save the operational cost.

Metering: The metering function in 3PSP records the services/contents/applications used by a served user. The metering policy is based on the data collecting policy.

The data collecting function will periodically (decides by the data collecting policy) collects the metering data. We also considered that the SP to collect the metering data but later we found it impractical. The 3PSP wants to keep the metering function private out of security reasons. 3PSP will also keep the data collecting and charging function to itself when possible.

Data collecting: Provides usage records to the charging function within the same domain or charge the SP based on the usage record - in case of the 3PSP does not have the charging function itself. This function will collect the metered data about services/contents/applications. The data collecting policy is based on charging policy. The data collecting function creates the usage records and is responsible for the store of records if necessary.
The line between Data collecting (3PSP domain) and Charging (SP domain) in Figure 3-7 represents a charging relationship in situation that the 3PSP does not have its own charging function. The 3PSP domain charges the SP domain based on the usage records. The records contain raw data, which maybe represents the amount of content the user required, or the services the user used. The 3PSP and SP will have mutual agreement about how to process the charging requests. Correspondingly, the line between Data collecting (3PSP domain) and Charging (3PSP domain) in Figure 3-7 represents the situation that the 3PSP has its own charging function.

Charging: To charge the SP based on the charging records. The charging function in 3PSP domain is optional because it can be also placed in the SP domain. The charging function in 3PSP mainly calculates the services/content/applications usage using functions decided by the charging policy. The charging function will try to combine the charging records if they share some common characteristics (e.g. same session ID of a user). The charging function sends charging requests to the SP. This is represented by the line between Charging (3PSP domain) and Billing (SP domain) in Figure 3-7.

The Service Platform Domain:
The functional building blocks that are distributed in SP domain are charging and billing. But the charging function in SP domain is slightly different from the charging functions in AN/3PSP domain. We will explain in next paragraph.

Charging: Provides charging records to the Billing function. It handles the charging request from ANs and 3PSPs, when they do not have charging functions of their own. The charging policy is based on the mutual agreement between the SP and ANs, SP and 3PSPs. The charging function merges the generated charging records if they contain certain identical information (e.g. same session ID of a user). The mergence can happen among records originated from the same domain, or from different domains.

Billing: The Billing function will generate the invoice for the end-users. It handles the charging requests that come from ANs and 3PSPs. Then it tries to merge the charging records from AN domain, 3PSP and its own domain if they contain certain identical information (e.g. same session ID of a user).

In Figure 3-7, there is also a charging model for the SP. This charging model will decide how the SP will charge the end-users for the “One Bill” service and other services it provides. But since in this reference model, there are no metering, data collecting functions to accurately measure services the SP provides, the SP will charge the end-user based on flat-fee, or based on the percentage of the “One Bill”. (E.g. if the amount of “One Bill” is €30, the service provisioning fee is 2%, then the SP charges the end-user €0.6 (30* 2%)). It is also a solution to reduce the operational cost of the SP. Discussion of the charging model of the SP is out of the scope of this thesis.

Note the reference models mentioned above are mainly focused on the end-user's aspect: *how to generate one integrated bill in the SP domain for an end-user*. The end-users pay to the SP. SP will pay ANs and 3PSPs from the money it receives from the end-users (Figure 3-8).

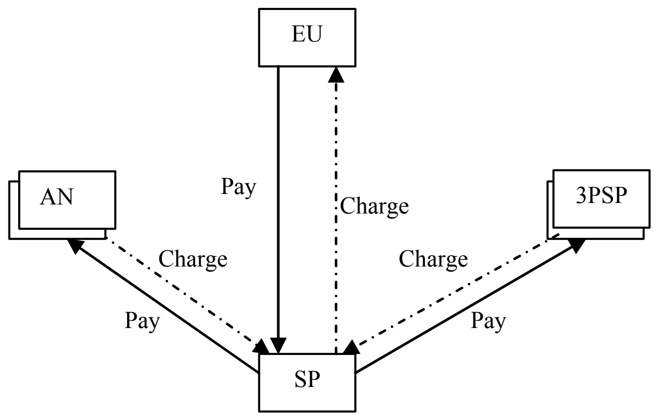


Figure 3-8: How ANs/3PSPs/SP Generate Revenue

The reference model describes how ANs/3PSPs/SP can get money from the users, but the model does not specify how SP could make revenue from the 3PSPs. The picture will be different if we look at the accounting facilities from the SP's aspect, where the SP may want to charge for the services that it provides to the 3PSPs. Then the SP will have its own metering, data collecting, charging and billing functions which are dedicated to charge the services it provides to the 3PSPs. The reasons why we only design the system from the end-user's aspect are:

- If the SP charges 3PSPs for the services it provides, there will be extra cost for the SP to implement the metering and data collecting functions. The 3PSP will also increase its cost. Ultimately, the cost will be added to the end-users. SP can directly charge the end-users for the services it provides.

- It is not clear now what kinds of value-added services that the SP can provide and charge for them from the 3PSPs.

It is the same if the SP wants to accurately meter the services it provides to the end-users: it needs the metering function, data collecting functions that are dedicated to this specific purpose.

The Federated Accounting Facilities:
One of the important characteristics of Mobile and Wireless Internet envisioned by 4GPLUS is the concept of federation. Federation provides user wider range of service coverage. It also gives service providers (ANs/3PSPs) in different administrative domains more opportunities by increasing the potential number of users. The exchange of accounting related information of federated systems takes place at charging level; please refer to Figure 3-9:

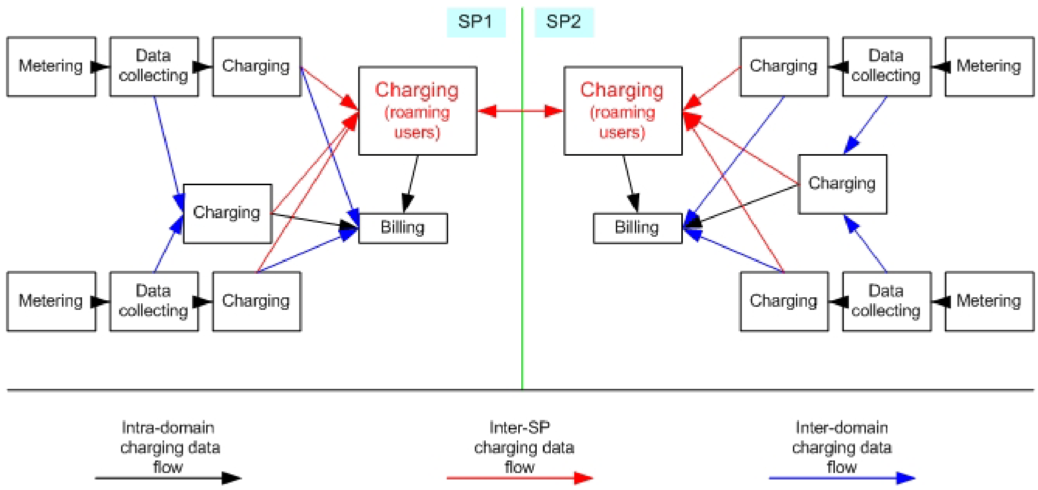


Figure 3-9: The Federated Accounting Facilities

In this model, we assume the roaming user will use the services of ANs and 3PSPs from the foreign SP (an SP that the user does not have direct contractual relationship); there is a dedicated functional block that handles the integration of charging records for roaming users. Please refer to the blocks "Charging function for roaming users". The foreign SP then charges the home SP of a roaming user based on the integrated charging records. The billing function of a home SP will put together charging records from its own domain as well as a foreign SP. The user still gets only one bill even he is roaming and uses the services provided by a foreign SP.

The advantage of this model is that the user may use the service provided by the foreign AN/SP/3PSP as a local user. The user does not need to worry about the expensive roaming cost.

There could be other reference models for the federated accounting facilities. It depends on how we define the behaviors of federated systems. It is more flexible when the user can use the services of home 3PSPs through the foreign ANs. But currently it is still under discussion by the 4GPLUS research group how the federated system works. Therefore, we will focus on accounting facilities of single SP domain in this thesis.

3.6.2 The Correlation of Charging Information

When the accounting information sent by different domains reaches SP, it is the responsibility of SP to integrate related information.

The integration can occur at two levels: one is the simple add-up of all related records coming from one domain. The other is to associate the records coming from different domains but have close relationship (e.g. belong to the same session of an application) into one record. Which level of integration will to be used depends on the charging models and the corresponding accounting information sent by the different domains.

When the AN/3PSP do not/can not distinguish each application of the user, the first level integration will occur. For example, if specific AN and 3PSP both use a charging model that will charge user based on the time of connection. When the user uses the mobile device to see an online movie, both the AN and the 3PSP will generate charging request. However, there is no tight relation between the two charging requests since the user may at the same time downloading large files through the AN. It is not correct to correlate the charging records from the 3PSP with those from AN and tell the user that watching the movie generates those charging requests from the AN. In this situation, the integration will be the simply add-up of the records coming from the same domain. More specific, in this situation if the user utilizes access network A and it generates 5 charging requests in half an hour, the SP will integrate these 5 charging records into one.

If the AN and the 3PSP can distinguish each application of the user, then it is possible to integrate the records coming from different domains but related to one application into one record. For example, if the AN uses a charging model based on volume of traffic, from which the AN can use some mechanisms to identify each application of the user. The 3PSP uses a pay per-view charging model and it also use the same mechanism as the AN to identify the application of the user. If the user watches an online movie and downloads files through the AN at the same time, the AN can send charging requests that distinguish between these two applications. The charging requests will consist of application specific information such as time, source and destination IP address, port numbers and transportation protocol that can uniquely identify an application. At the same time, the charging requests from the 3PSP also consist of similar information. The SP then can utilize these information to correlate the charging requests from the AN and the 3PSP. Detailed explanation and concrete implementation will be provided in chapter 4.

3.6.3 The Supplementary Services

In addition to the basic service the accounting facilities provide, there are also some services that are related to accounting, which will be the desirable features of the 4GPLUS platform.

Pricing information: It can be foreseen in the near future numerous access networks will compete by using different charging models and provide different pricing for the services they offer. The pricing information can be used by the end-users as a key parameter to select the access networks. Also, pricing information can also be required by the 3PSP to figure out the total price of a particular service. Please refer to the Video-On-Demand scenario in section 3.5.1.

Network QoS information: The QoS attributes of the AN such as bandwidth, delay, jitter, packet loss rate, etc. are important parameters for the users to choose the access networks. While the SP hides the heterogeneity of the underlying ANs from the 3PSPs, sometime the 3PSP must know the attributes of the underlying AN in order to provide corresponding services with different quality.

3.7 The Definition of Interfaces

In this section, we define the interfaces between different service provisioning domains. Because the 4GPLUS platform is centralized in Service Platform and also because our goal is to generate one integrated bill for the users, we mainly consider the interfaces between AN-SP, SP- 3PSP. The interface between 3PSP and End-user remains open.

3.7.1 Accounting Related Message Exchange Between Domains

We derive the message exchange between different domains through the UML diagrams from our user scenarios in section 3.5

From the Video-On-Demand service scenario (Section 3.5.1), there should be (Figure 3-10):

- Pricing inquire/answer message between AN and SP
- Pricing inquire/answer message between 3PSP and SP
- Pricing provision message from 3PSP to EU
- Money/credits reservation/confirmation message between 3PSP and SP
- Charging message from AN to SP, based on the services AN provides, from the reserved money/credits.
- Charging message from 3PSP to SP, based on the services/content/application 3PSP provides, from the reserved money/credits.

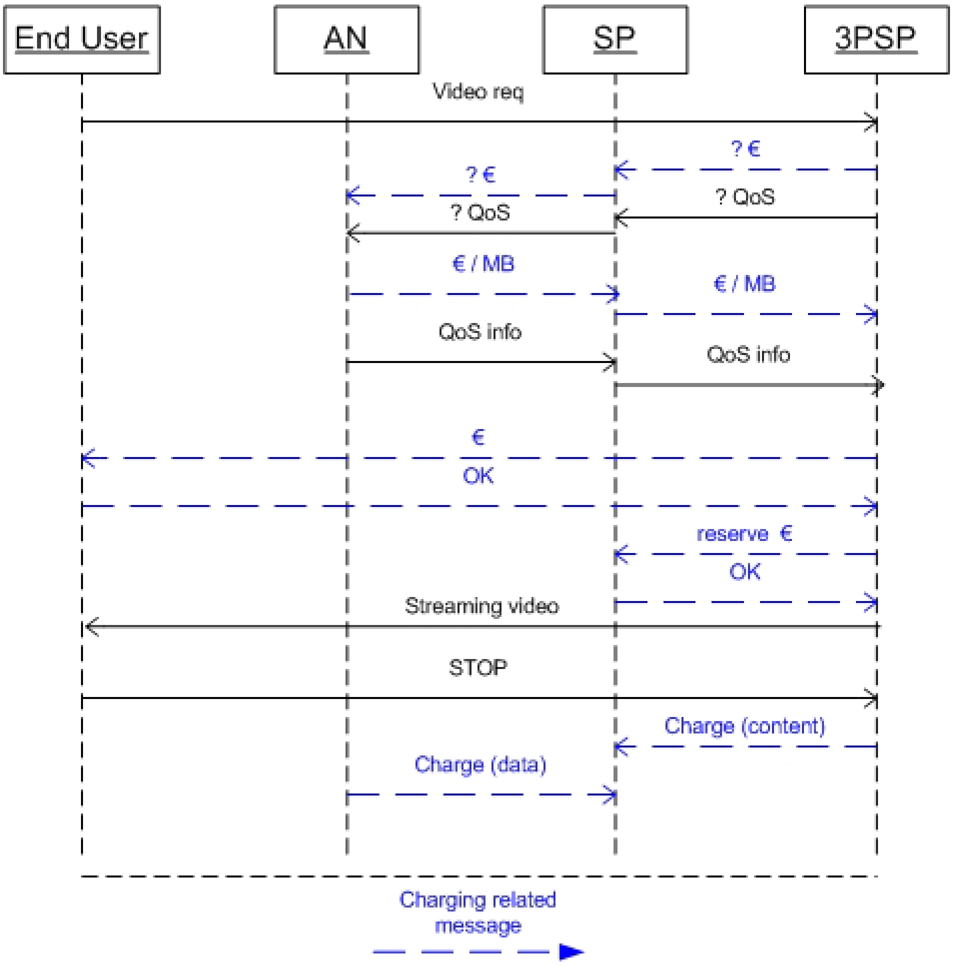


Figure 3-10: Message Sequence Diagram of the Video-On-Demand Service

From the M-health service scenario (Section 3.5.2), there should be (Figure 3-11):

- Charging message from AN to SP, for the Location service AN provides.
- Charging message from 3PSP to SP, for the services/content/application 3PSP provides
- Charging message from AN to SP, for the volume of traffic AN transports.

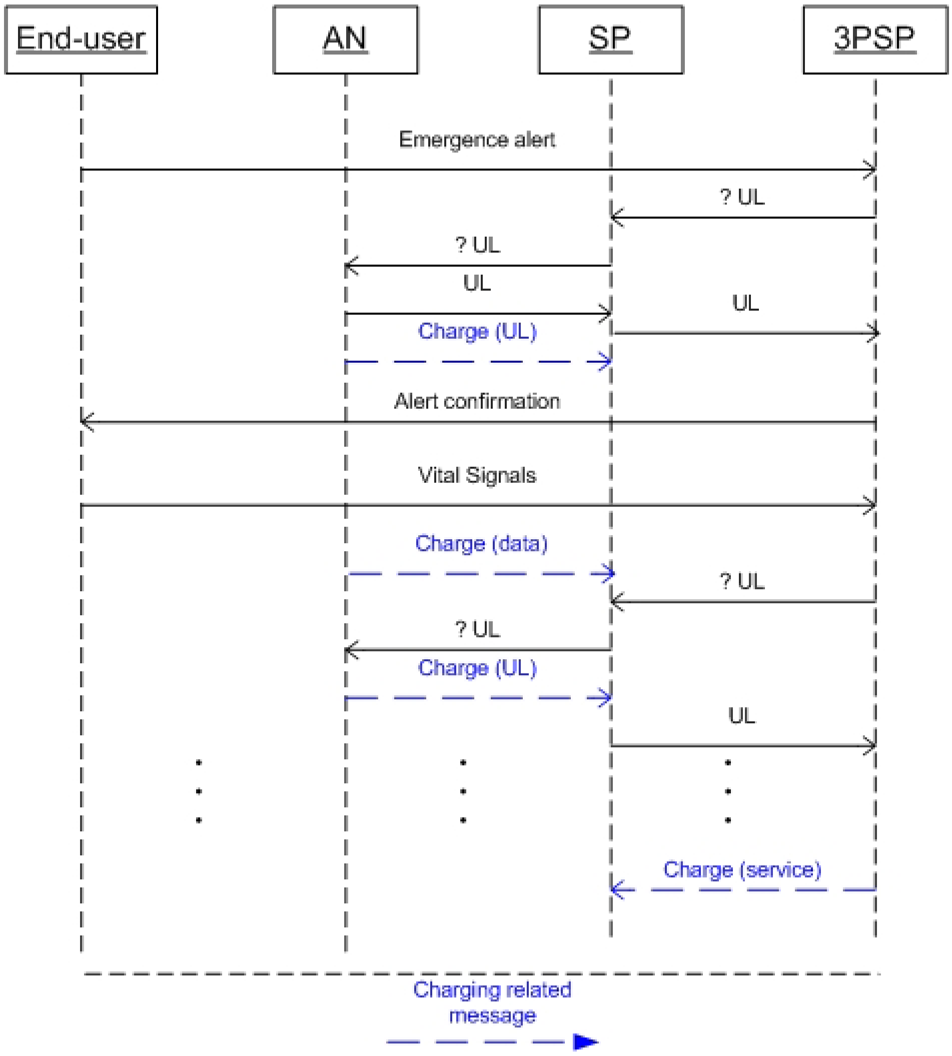


Figure 3-11: Message Sequence Diagram of M-health Services

From the subscription-based services (Section 3.5.3), there should be (Figure 3-12):

- Charging message from AN to SP, for the services AN provides
- Charging message from 3PSP to SP, for the services 3PSP provides

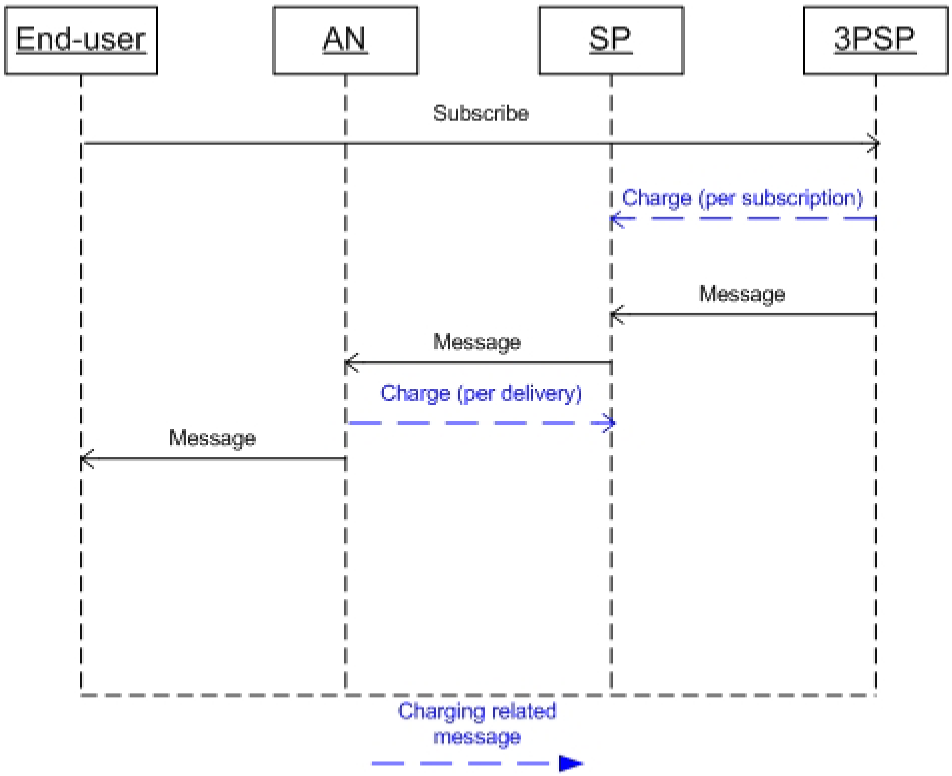


Figure 3-12: Message Sequence Diagram of Subscription-based Services

3.7.2 OSA/Parlay and Parlay X

In the Mobile and Wireless Internet envisioned by the 4GPLUS project, the access networks can be GPRS networks, UMTS networks or WLANs. Because the SP lies in the Internet domain, there must be interfaces with the networks in the telecommunication domain. Within the telecommunication networks, there are standard interfaces which allow communication between different domains, but the only specifications exist that provides standard interfaces of telecommunication networks to other type of network (Internet) are OSA/Parlay[38] standards.

The OSA/Parlay specifications define an architecture that enables Service Developers and Application Developers to access of network capabilities through an open standardized interface, i.e. the OSA/ Parlay APIs (Figure 3-13). The OSA/Parlay APIs are designed to enable creation of telephony applications as well as to "telecomm-enabled" IT applications. [PARL02a].

However, the OSA/Parlay API are too complex to use, the Parlay group introduced ParlayX web services. It is an abstraction and simplification of Parlay APIs. While the Parlay standards emphasis more on the how to open up the telecommunication network, web services are more Internet oriented and emphasis on how to make the

[38] The terms 'Parlay', 'OSA', and 'Open Service Access' exist for historical reasons. 'Parlay' refers to the work product generated by The Parlay Group, and 'OSA' refers to the work product generated by 3GPP. The two specifications and APIs have been harmonized, and are jointly published by The Parlay Group, ETSI and 3GPP. Therefore it is common to see the API set referenced as 'Parlay', 'OSA' or 'Parlay/OSA'. Parlay Group (http://www.parlay.org/).

services available through the web. ParlayX web services stimulate the development of next generation network applications by developers in the IT community who are not necessarily experts in telephony or telecommunications [PARL02b]. Most of the Parlay X web services will be implemented by invoking functionality on a Parlay Gateway. A Parlay X application can be written in any language as long as it can make proper web service invocations.

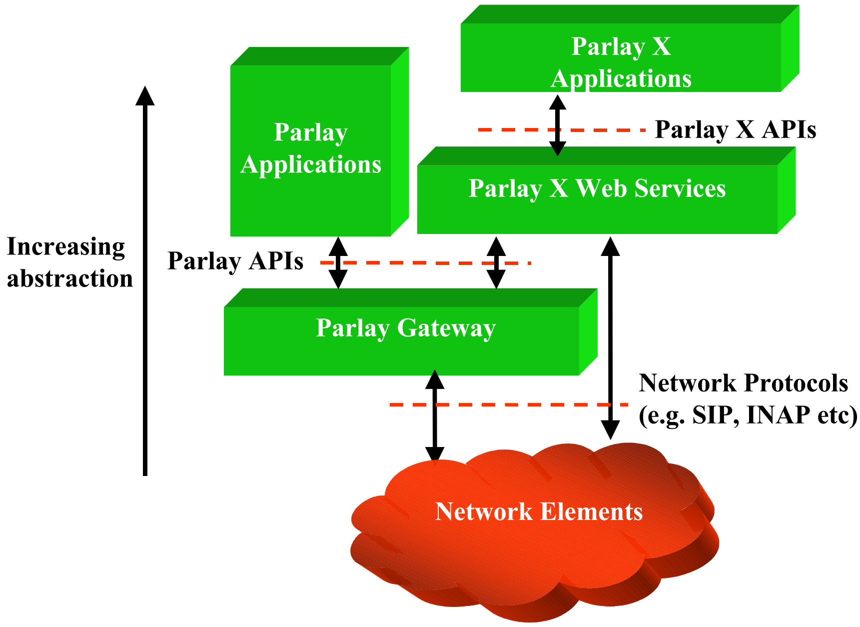


Figure 3-13: Parlay X Web Services, Parlay X APIs and Parlay APIs

We make investigation of ParlayX web services and find out there are ParlayX payment web services in the specification [PARL02b], which supports payment reservation, prepaid payments, and postpaid payments. It supports charging of both volume and currency amounts.

Further investigation of ParlayX payment web services shows that they support most the message exchange we describe in section 3.7.1. Because OSA/Parlay standards are the only standards to provide interfaces of telecommunication networks, and the ParlayX payment web services can provides simplified Parlay interfaces which meet our requirements, we decide to adapt ParlayX payment web services in our accounting facilities.

ParlayX payment web services provide amount charging API and volume charging API. In the amount charging API, there are message-based invocations with chargeable information represented in the amount of currency. If an AN/3PSP has its own charging function, it will apply its own charging policy upon the usage records and generate charging records, which are represented in the amount of currency. The amount charging API can be used as the interfaces between the Charging functions in AN/3PSP domain and Billing function in SP domain.

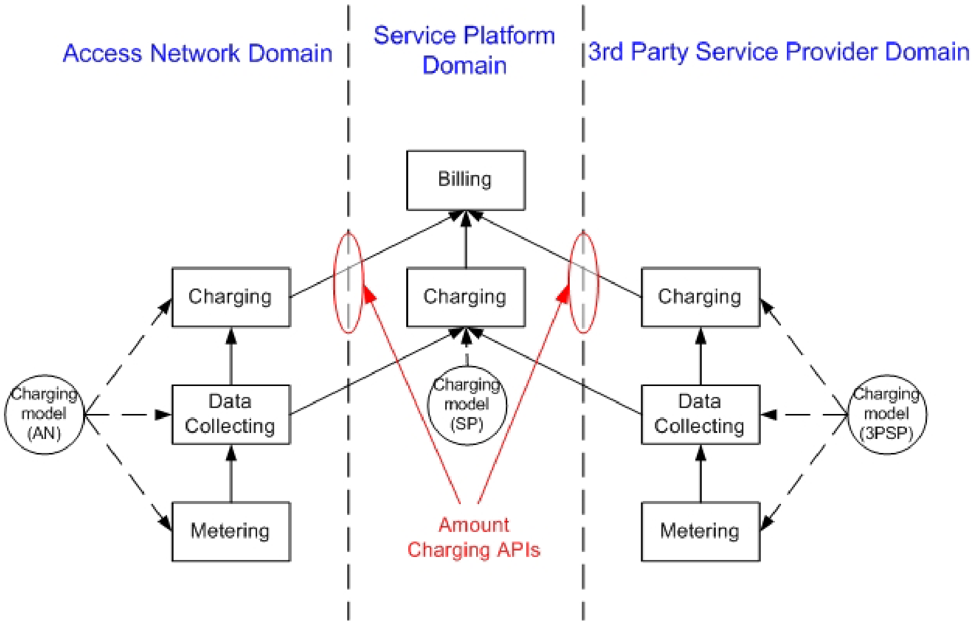


Figure 3-14: The Location of Amount Charging APIs

While in the volume charging API, the chargeable information is represented in the volume to be charged, that is, in bits and bytes. When the AN/3PSP doesn't have its own charging function to generate the charging records, AN/3PSP will send the usage records to the SP for further processing. In another words, AN/3PSP will charge the SP based on the usage records. The volume charging APIs can be used as the interfaces between the Data Collecting functions in AN/3PSP domains and the Charging function SP domain.

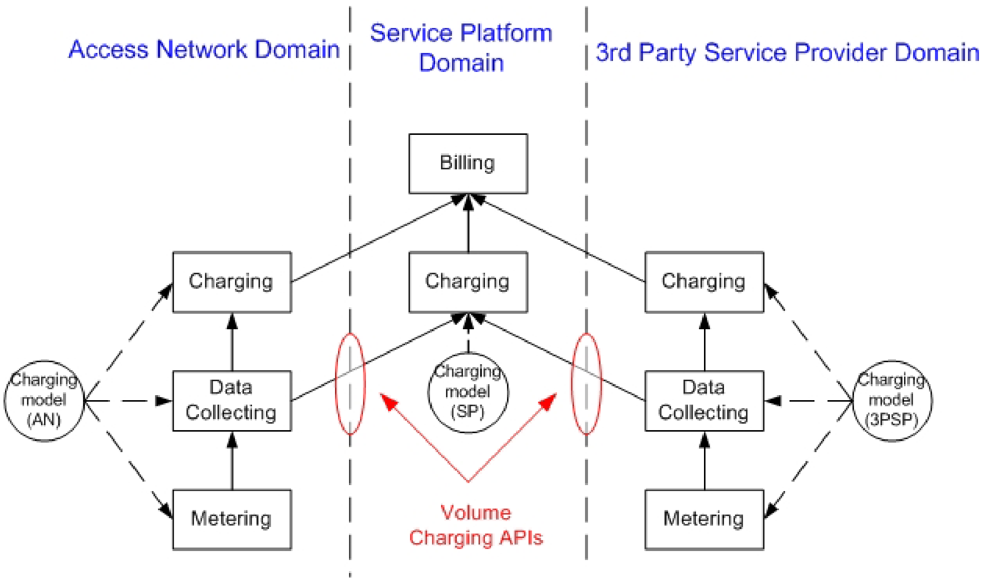


Figure 3-15: The Location of Volume Charging APIs

The Parlay X web services also support money/volume reservation, which can be used to support prepaid and postpaid user type. There are also getAmount() invocations in the Payment web service of Parlay X, which can be used to inform the end-user about the estimation of cost of a particular service. We also need this feature in our project.

Note that in Parlay X, the service provisioning part has only two domains: the Operator domain and the Service Provider domain. While in our project, there are three domains. But when we treat the Access Networks as service providers that provides data transport as well as other services; and treat the SP domain as the operator domain, then the Parlay X model can be fitted well into our project.

There are four charging APIs in the documents of Parlay X web services. The message-based invocations of each API are list below:

Amount Charging API	➢ chargeAmount() ➢ refundAmount()
Volume Charging API	➢ chargeVolume ➢ getAmount() ➢ refundAmount()
Reserved Amount Charging API	➢ reserveAmount() ➢ reserveAdditonalAmount() ➢ chargeReservation() ➢ releaseReservation()
Reserved Volume Charging API	➢ getAmount() ➢ reserveVolume() ➢ reserveAdditionalVolume() ➢ chargeReservation() ➢ releaseReservation()

Table 3-1: The Payment APIs and Their Corresponding Methods in Parlay X Web Services

The methods in the draft document [PARL02b] of Parlay X web services are limited. Certain parameters are negotiated offline, for example, the currency, volume type, default reservation enforcement time, et cetera. It is desirable that these parameters can be negotiated dynamically inside the methods, for example, the message itself carries whether the charge is in US Dollar or in Euro. In the Mobile and Wireless Internet environment, the AN can have more charging models, not only based on volume of traffic, it can also be based on time of connection. It is more reasonable to introduce the concept Price/Unit instead of Price/Volume. Parlay X web services are still in their infancy, but they are poised to make great inroad in the development of next generation network applications

Everything should be made as simple as possible,
but not one bit simpler.

-- Albert Einstein

4 Implementation

This chapter describes our implementation of the accounting facilities, which serves as a proof of feasibility of the design.

This chapter is organized as follows: section 4.1introduces the implementation environment; section 4.2 describes in details the VOD scenario we have implemented. The implementations of AN, 3PSP and SP are presented in section 4.3, 4.4, and 4.5 respectively.

4.1 The Implementation Environment

The accounting facilities are implemented as an extension to an existing service development and deployment environment for beyond 3G services at the premises of Bell Labs Advanced Technologies in Enschede, the Netherlands. The hardware topology of the implementation is shown in Figure 4-1.

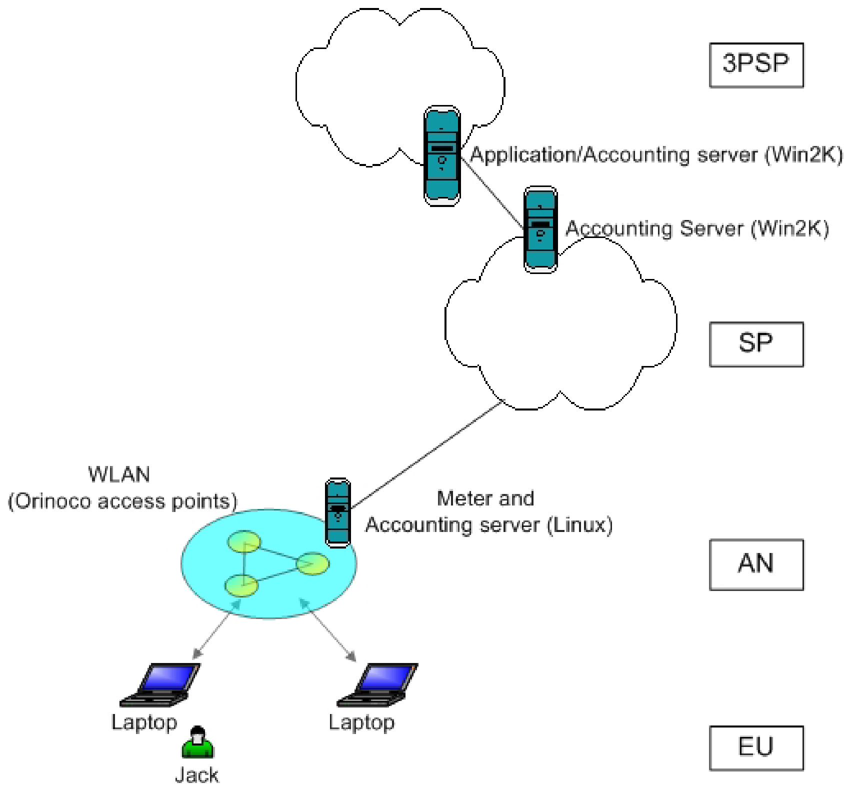


Figure 4-1: Components of the Accounting Facilities

Because of academic research purposes, most of the software components are free of charge to download or open source so that we can modify the source codes according to our requirements. The names and purposes of these components are listed in **Error! Reference source not found.**. Detailed description of the software will be given in the places where it is used.

Components	Purpose
Windows (XP or win2k)	OS
Linux (Red Hat v)	OS
Tcpdump	Meter http://www.tcpdump.org/
JBoss	Application server http://www.jboss.org
Tomcat Web server	Web Server http://jakarta.apache.org/tomcat/index.html
Apache Axis, WSDL4J	SOAP http://ws.apache.org/axis
HUT Mobile IP Home Agent (IPv4)	MIP Home Agent http://www.cs.hut.fi/Research/Dynamics
SECGO	MIP Client http://www.secgo.com
HSQL Database	Data Storage http://hsqldb.sourceforge.net/

Table 4-1: Software Components List

4.2 A Video-On-Demand Scenario

For the evaluation of the accounting facilities, we have selected the video-on-demand (VOD) scenario, which is described in section 3.5.1. We are in favor of this scenario because of the following reasons:

1. Compared with the voice service of the GSM system or the MMS service of the GPRS system, video streaming is envisioned as a key feature of the next generation service enabled by the all IP Mobile and Wireless Internet.

2. Video streaming can generate large volume of traffic in a short period. It is easier to observe and manipulate the generated data. For example, if the charging model in the AN is based on volume of traffic and the price is 5MB/euro. A video streaming will generate quite some traffic in a couple of minutes, which may cost 0.5 euro or 1 euro. The traffic generated by the randomly browsing the Internet will be much less and the price may be 0.0000001 euro, which is not convenient to read difficult to generate.

The charging model in the AN domain will be based on the volume of traffic, the user will be charged for the traffic he/she generates; to simplify the implementation, our metering policy is only to capture payload of UDP packets of the streaming videos. In the 3PSP domain, we use a pay per-view charging model (i.e. amount based) as it is a popular charging model and not complex to implement. The user will be charged on each video he/she watches.

The charging models serve as guidance of the implementation of functional building blocks. We choose to implement Metering and Data Collecting functions in AN domain. The meter will capture IP packets, filter out UDP packets and add-up the UDP payloads. The Data Collecting function will generate usage records which consist of information of the UDP packets as well as the volume of traffic. In the 3PSP domain, all the three functions will be implemented: Metering, Data Collecting and Charging. The pay per-view charging model will require the Metering and Data Collecting function generate one usage record for one video request. In the SP domain, we implement the Charging function and part of the Billing function. Generation of bill for user is out of the scope of this project.

Figure 4-2 shows the functional blocks in our implementation. In our design (Figure 3-7), the charging functional blocks in AN domain and 3PSP domain are symmetric, which can cover all the charging models. Our implementation of the VOD scenario is an asymmetric approach, where the AN domain has two functional blocks and the 3PSP domains has three functional blocks. In this way, we can examine the integration capability of the SP in minimum efforts when the charge volume requests come from the AN domain and the charge amount requests come from 3PSP domain.

Compared to the description in section 3.5.1, the user scenario is simplified. There is no pricing information, neither network QoS information provided by the AN.

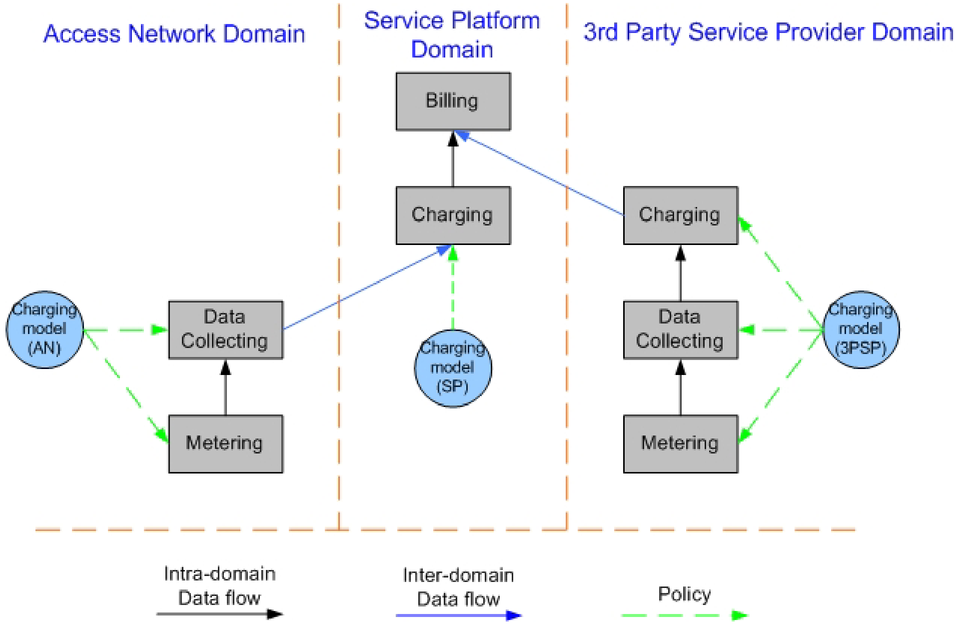


Figure 4-2: Implementation of the Reference Model

4.3 Implementation of the Access Network

This section describes the implementation of the metering and charging function of the access network.

4.3.1 Implementation of the Metering Function

Most of the access points provide simple metering function. For example, the Orinoco™ access point AP-500 provides Management Information Base (MIB) that can bind MAC address (Media Access Control address) of a user with the amount of traffic he/she generates (in bytes). However, this is not sufficient if we want to distinguish the different applications that the user invokes.

We assume each application of the user will setup a TCP connection or use UDP to send data. Let's examine the TCP/IP protocol stack from bottom to top. We can not get much information from the header of physical layer packets about the application. It is the same for the data link layer. (For example, if we use Ethernet, we get only get MAC addresses of the sending host and destination host in the fragment of Ethernet). If we look into the network layer, from the IP packet, we can retrieve source IP, destination IP and transport protocol used in the data portion of the Internet datagram. In the transport layer, most of the packets are either TCP packets or UDP packets. Looking into TCP or UDP packet, we will find information such as source port number and destination port number. The data filed of transport layer packet contains application layer data.

Source IP, destination IP and transport protocol together can not uniquely identify an application; two TCP connections may have the same source IP, destination IP and transport protocol but using different ports. To uniquely identify the application, we need to gather the following information (5-tuple):

- Source IP address

- Destination IP address
- Source port number
- Destination port number
- Transport protocol

While it is relatively easy for the AN to capture the above information, it is difficult to capture it in the 3PSP side, where we need to hook into the application (streaming) software and it is a lot effort. In order to keep the implementation simple, we capture only 3 pieces of information (3-tuple) which can be captured from the IP packets: source IP address, destination IP address and name of transport protocol. The correlation of the information from different domains and the limitations of the 3-tuple solution will be analyzed in section 4.5.2.

All information we need to capture besides the 3-tuple is the time and pay load of the UDP packet. By connecting the access points to a Linux PC (which also serves as a router), we use Tcpdump[39] to inspect every IP packets.

As it has been explained in section 3.2.2, in order for the user to maintain connectivity while roaming between different networks, we choose to use MIPv4. The data format of MIPv4 in RFC2003 describes an IP-in-IP encapsulation (Figure 4-3). There are four IP addresses in a MIPv4 packet: the IP address of the Home Agent (HA), the Care-of address (COA), the IP address of Correspondent Node (CN) and the IP address of mobile node (MN). But in practice, we need only two of the IP addresses: the IP address of CN, which is in our case, the IP address of the server of the 3PSP; the IP address of MN, which is in our case, the fixed IP address of the mobile terminal of the end-user.

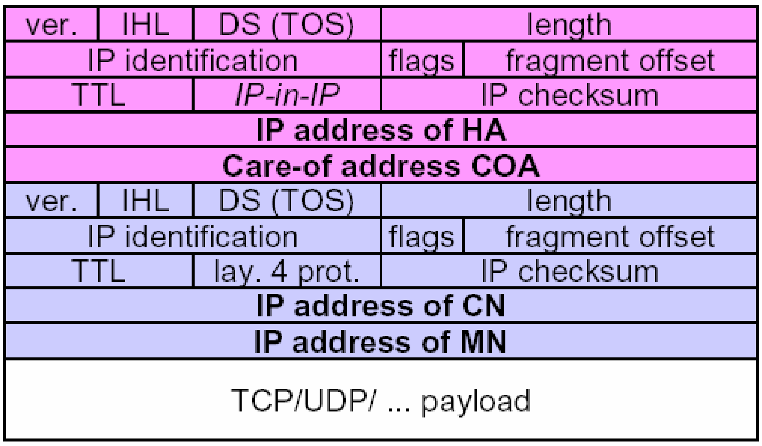


Figure 4-3: The IP-in-IP Encapsulation

The Tcpdump running in default mode capture too much information and the format of the information does not meet our requirement. We add parameters to the Tcpdump command as well as using some string manipulation commands in Linux to obtain exactly the information we need. The command in Table 4-2 shows how we use Tcpdump to inspect and filter the traffic information. The records in Table

[39] Tcpdump is a tool for network monitoring, protocol debugging and data acquisition. The manual of it can be found at the official website: http://www.tcpdump.org.

4-3 show the content of the information we actually acquire; each record contains time stamp, payload of the packet and the 3-tuple we mentioned above.

```
# tcpdump -I eth0 -tt -q -nn -f  ip and not ip proto \\icmp and not
ip proto \\igmp and not ip proto \\tcp | grep "(ipip)" | cut -d' ' -
f1,5,7,8,9,10 |grep -v frag |grep "udp"| ./metering

-tt   Print an unformatted timestamp on each dump line

-q    Quick output. Print less protocol information so output
      lines are shorter.

-nn   Don't convert protocol and port numbers etc. to names either.

-f    Print 'foreign' Internet addresses numerically rather than
      symbolically.
```

Table 4-2: The Commands to Filter the IP packets Using Tcpdump

```
...
1054741362154 217.114.111.135 217.114.111.140 udp 1288
1054741362253 217.114.111.135 217.114.111.140 udp 1288
1054741362352 217.114.111.135 217.114.111.140 udp 1288
1054741362452 217.114.111.135 217.114.111.140 udp 1288
1054741362551 217.114.111.135 217.114.111.140 udp 1288
1054741362649 217.114.111.135 217.114.111.140 udp 1288
1054741362758 217.114.111.135 217.114.111.140 udp 1288
1054741362856 217.114.111.135 217.114.111.140 udp 1288
1054741362954 217.114.111.135 217.114.111.140 udp 1288
1054741363053 217.114.111.135 217.114.111.140 udp 1288
...
```

Table 4-3: The Filtered Traffic Information of an Access Network

The filtered traffic information can not be used to generate usage records yet. The information is not compressed and it is too much to be put into a database. To compress the "raw" material, every piece of filtered traffic information will serve as the input of a Perl[40] script, which will merge records that carry the same 3-tuples. For example, in Table 4-4, if the original traffic generates four records, they will be merged into one record, which carries the time stamp of the first record and the volume of traffic is the sum of the four records. However, if the Perl script finds a unique record that can not be merged with others, it will keep it untouched and later write this record to the database.

```
Original traffic information:
1054741362649 217.114.111.135 217.114.111.140 udp 1288
1054741362758 217.114.111.135 217.114.111.140 udp 1288
1054741362856 217.114.111.135 217.114.111.140 udp 1288
1054741362954 217.114.111.135 217.114.111.140 udp 1288
1054741363053 217.114.111.135 217.114.111.140 udp 1288

Compressed traffic information:
1054741362649 217.114.111.135 217.114.111.140 udp 6440
```

Table 4-4: The Merging of the Records

[40] Perl is a script programming language that is similar in syntax to the C language and that includes a number of popular UNIX facilities such as SED, awk, and tr.

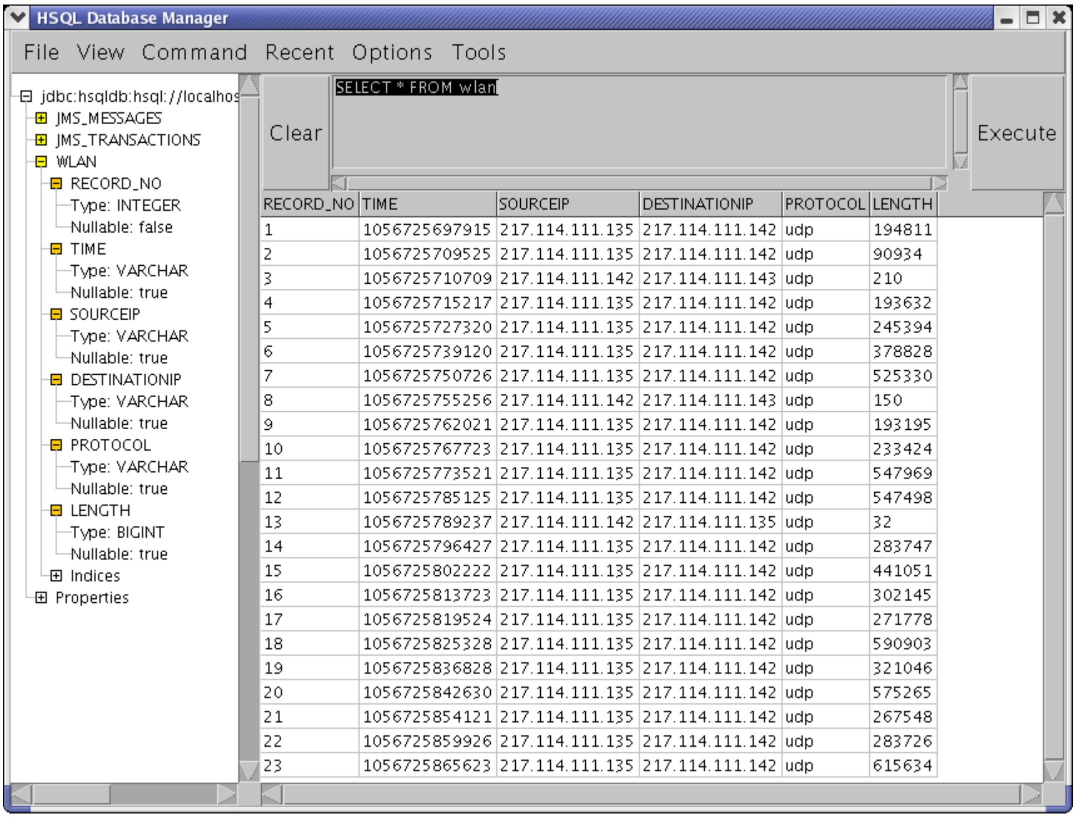
Tcpdump continuously examines the traffic and feed information to the Perl script. The Perl script keeps merging records for a period of time (e.g. 30 seconds, the time interval can be changed through the source codes, and the shorter the time interval, the closer the system is a real-time system).

Then the compressed records will be written to a Hypersonic database as usage records directly. Since the IPDR standard for WLAN traffic is not available yet, we will use the record format defined by ourselves Table 4-5.

Filed name	Format
RECORD_NO	INTEGER (NOT NULL PRIMARY KEY)
TIME	VARCHAR
SOURCEIP	VARCHAR
DESTINATIONIP	VARCHAR
PROTOCOL	VARCHAR
LENGTH	BIGINT (long)

Table 4-5: Usage Record Format in AN

Figure 4-4 shows the copy screen of the database in the AN side.



RECORD_NO	TIME	SOURCEIP	DESTINATIONIP	PROTOCOL	LENGTH
1	1056725697915	217.114.111.135	217.114.111.142	udp	194811
2	1056725709525	217.114.111.135	217.114.111.142	udp	90934
3	1056725710709	217.114.111.142	217.114.111.143	udp	210
4	1056725715217	217.114.111.135	217.114.111.142	udp	193632
5	1056725727320	217.114.111.135	217.114.111.142	udp	245394
6	1056725739120	217.114.111.135	217.114.111.142	udp	378828
7	1056725750726	217.114.111.135	217.114.111.142	udp	525330
8	1056725755256	217.114.111.142	217.114.111.143	udp	150
9	1056725762021	217.114.111.135	217.114.111.142	udp	193195
10	1056725767723	217.114.111.135	217.114.111.142	udp	233424
11	1056725773521	217.114.111.135	217.114.111.142	udp	547969
12	1056725785125	217.114.111.135	217.114.111.142	udp	547498
13	1056725789237	217.114.111.142	217.114.111.135	udp	32
14	1056725796427	217.114.111.135	217.114.111.142	udp	283747
15	1056725802222	217.114.111.135	217.114.111.142	udp	441051
16	1056725813723	217.114.111.135	217.114.111.142	udp	302145
17	1056725819524	217.114.111.135	217.114.111.142	udp	271778
18	1056725825328	217.114.111.135	217.114.111.142	udp	590903
19	1056725836828	217.114.111.135	217.114.111.142	udp	321046
20	1056725842630	217.114.111.135	217.114.111.142	udp	575265
21	1056725854121	217.114.111.135	217.114.111.142	udp	267548
22	1056725859926	217.114.111.135	217.114.111.142	udp	283726
23	1056725865623	217.114.111.135	217.114.111.142	udp	615634

Figure 4-4: The Screenshot of Usage Records in a Database

4.3.2 *Implementation of the Charging Function*

In our implementation, the AN doesn't have its own charging function, it will send charging requests (volume) to the SP. And SP will calculate the price from the volume information carried in the charging requests. We assume there are Service Level Agreements between AN and SP about the way to calculate the price. In our implementation, 10 mega bytes of traffic will be charge for one euro.

We have used ParlayX standard in our implementation, but due to the incompatibility between the WSDL definition and the development environment, we had to simplify the WSDL definition and write our own WSDL files. In theory, the results of this project are comparable using both WSDL definitions.

Two ParlayX web services have been implemented: the charge volume web service and the charge amount web service. The WSDL files describing the services can be found in appendix A. AN sends the charge volume request to the SP through charge volume web service.

Table 4-6 shows the format of the charge volume request:

Name	Type	Description
endUserIdentifier	String	The identifier that uniquely identifies the end user
Volume	long	The volume of traffic generated by the user
billingText	String	3-tuple: source IP address destination IP address transfer protocol
referenceCode	String	Time, it serves as information to uniquely identify the request, e.g. in case of disputes

Table 4-6: The Format of the ChargeVolume Request

Note there will be a mapping between the IP address of an end user's device and the endUserIdentifier. Here we already have the mapping in our source codes as we have already known the mapping between end user and the IP address of his/her mobile device. But in real situation, the mapping needs to be obtained from the authentication function.

There are two steps in the implementation of the Data Collecting function:

1. Generate java codes from the WSDL file that describes the charge volume web service using software tool.

2. Write java program which retrieves usage records from the database and sends charge volume requests through web service.

In step 1, we use the WSDL2JAVA tool from Apache AXIS to generate java classes from the WSDL file. There are client side and server side in a web service; the AN will act as the client side of the web service. The description of how to generate the client side java classes can be found in the CDROM that comes with the thesis. For detailed explanations of the generated files, please refer to the User's Guide of APACHE AXIS[41]. The generated client side java classes of the charge volume web service are put into a jar file; it will be later invoked by other class later.

In step 2, the charging function is actually implemented through an MBean (Managed bean, Java class implementing a management interface and representing a resource to be managed or monitored). The advantage using MBean in the implementation is that we can manage the charge volume service through the console provided by the application server. Through the console, we can start and stop the charge volume service and we can also set the time interval between each task. By manipulating the time interval, we can change the frequency of charging requests which are sent and therefore realize the real-time and non real-time charging.

There are four java classes in our implementation, the description of them are shown in Table 4-7, the source codes can be found in the CDROM that comes with the thesis.

Class name	**Description**
ChargeVolumeMBean.java	Define our own MBean interface
ChargeVolume.java	Implementation of the MBean interface
DataCollect.java	A class that has the function of accessing usage records in the database and instantiate ChargeVolumeClient to send chargevolume requests. The startService() method of the MBean will instantiate this class
ChargeVolumeClient.java	A class that uses charge volume web service to generate charge volume requests.

Table 4-7: File Descriptions of the AN_chargeVolume.sar

The relationship between these java classes are shown in the class diagram Figure 4-5. Package com.jimmie.www contains the classes of charge volume web service.

[41] http://ws.apache.org/axis/

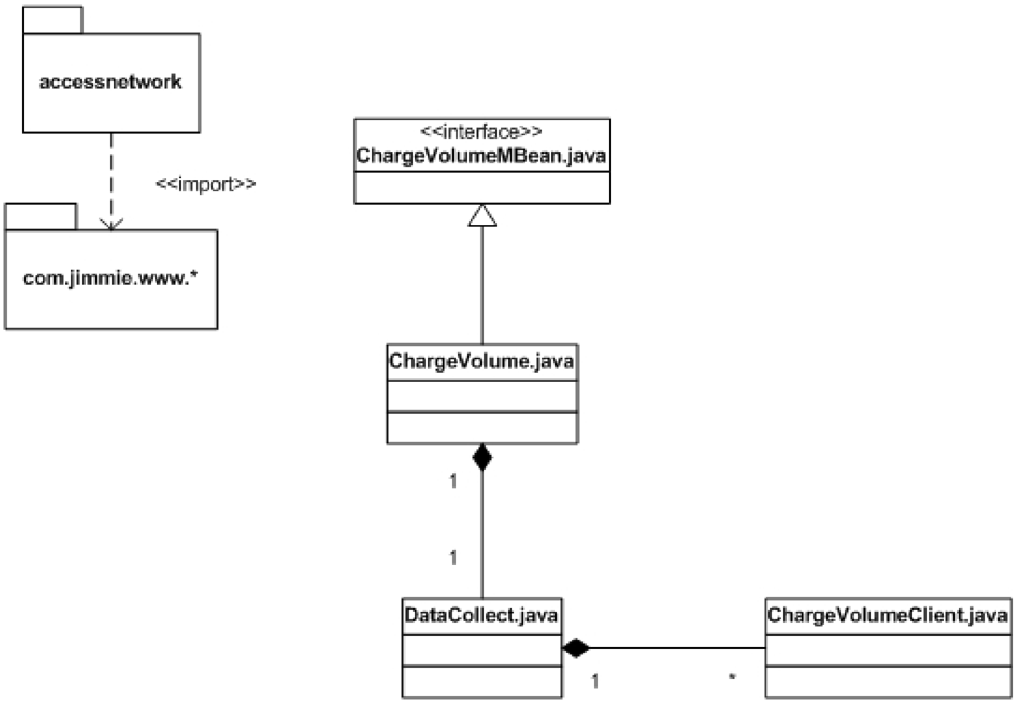


Figure 4-5: Class Diagram of AN Implementation

After compilation, the java classes are put into a sar (service archive) file and deployed under the JBoss[42] application server. Details about how to generate and deploy a sar file can be found at the CDROM comes with the thesis. Details about JBoss server can be found at its website.

Periodically (e.g. every minute, the time interval can be change through a console), the charging function will retrieve the usage records in the database and send charge volume requests. We can set the time interval to make the charging system real-time, near real-time or non real-time.

4.4 Implementation of the 3PSP

Implementation of the 3PSP consists of two parts:

1. The front-end of the 3PSP, which displays content of the services it provides and process the requests from the users;

2. The back-end of the 3PSP, which generates charging requests.

In our front-end implementation, the 3PSP offers a web page with a list of name of the video clips. Figure 4-6 shows the content of the web page. It is hosted in a web server of the 3PSP.

[42] JBoss is an open source J2EE compatible application server. http://www.jboss.org

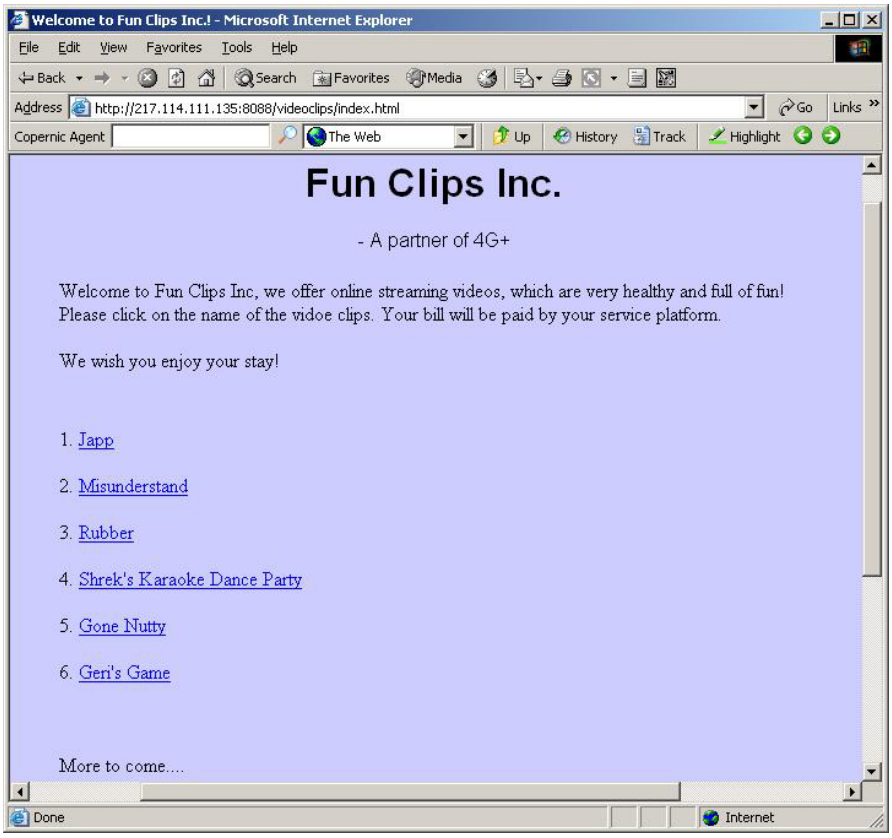


Figure 4-6: The Web Page of a 3PSP

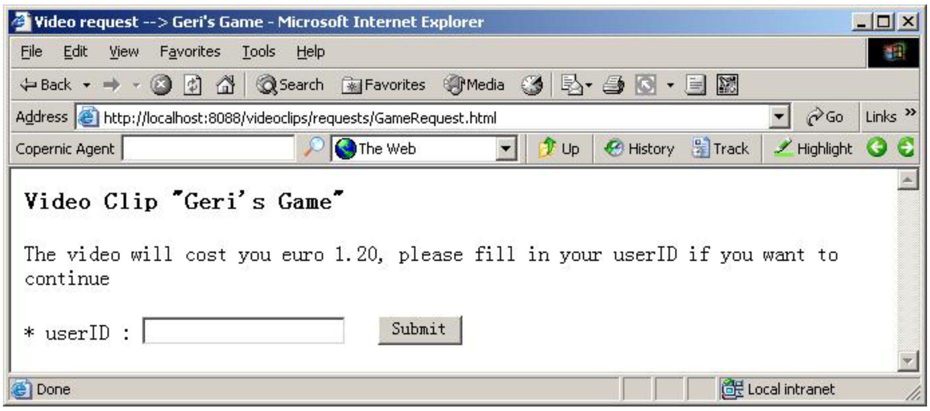


Figure 4-7: Pricing Information of a Video Clip

When the user clicks on a specific video name, he will be directed to another webpage (Figure 4-7) which shows the price of the video clip and asks the user to fill in his user ID. Since authentication was not a main goal of the project, it has not been implemented although the architecture has been designed to be extended with security features. We obtain the user ID from the user's input instead through authentication process.

As the user clicks the submit button:

1. A request will be sent to the 3PSP. The request contains the user ID and the name of the video the user wants to see.

2. A link will be sent to the user that redirects him to a video streaming server, which is a Microsoft MMS server in our case. The user can start watching the streaming video he requests.

The implementation of the back-end of 3PSP is made up of two steps:

1. Generate java codes from the WSDL file that describes the charge amount web service using software tool.

2. Write java programs which have Metering, Data Collecting and Charging functions.

The procedure to generate client side java classes of charge amount web service is similar to that of charge volume web service. Documentation of it can be found at the CDROM that comes with the thesis. The generated java classes are put into a jar file and can be invoked by other classes. The format of charge amount request is shown in Table 4-8.

Name	Type	Description
endUserIdentifier	String	The identifier that uniquely identifies the end user
amount	double	The amount of money in euro that the sender need to charge from the user
billingText	String	The 3-tuples: source IP address destination IP address transfer protocol
referenceCode	String	Time, it serves as information to uniquely identify the request, e.g. in case of disputes; the description of the charging request.

Table 4-8: The Format of the Charge Amount Request

In order to correlate the charging information from the 3PSP domain with AN domain, the 3PSP side needs to send charging requests which are consistent to the format of charge amount request. In addition to endUserIdentifier, time, IP address of the client, IP address of the streaming server, transportation protocol of the streaming, the 3PSP needs to send user friendly information which tells the user why he is being charged (e.g. watch video clip “Geri’s game”). This information is provided through the referenceCode part of the charge amount request. It is straightforward to offer content related description in the 3PSP side instead of in the AN domain.

The Metering, Data Collecting and Charging functions are actually implemented in two java classes; the relationship between the two classes is shown in Figure 4-8. Package com.jimmie.www.* consists of classes which are necessary to invoke the charge amount service.

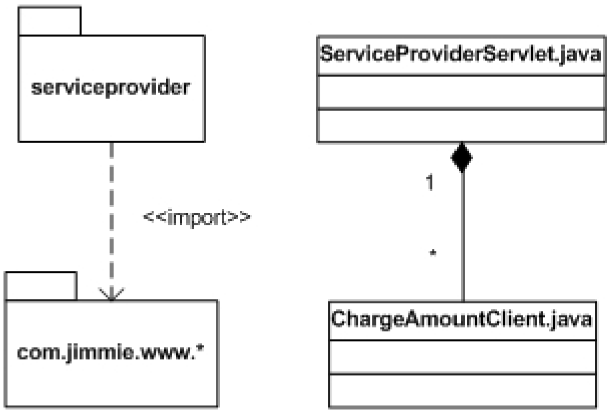


Figure 4-8: Class Diagram of 3PSP Implementation

ServiceProviderServlet has already known the IP address of the streaming server and the transfer protocol it uses (an implementation decision). It will obtain user ID, name of the video clip and user IP address from the user request. It also generates a time stamp for this request. As we tend to simplify the implementation and do not use a database in the 3PSP domain, after ServiceProviderServlet obtains all the parameters, it will process some of them: according to the name of the video clip, finds out the price and generate description text of the video clip. Finally, ServiceProviderServlet will instantiate ChargeAmountClient to send charge amount request using charge amount web service.

Deployment of the front-end and back-end of the 3PSP to an application server can be found in the CDROM that comes with the thesis.

4.5 Implementation of the SP and Correlation of the Charging Information

The implementation of the SP consists of three parts:

1. The server side implementation of charge volume web service

2. The server side implementation of charge amount web service

3. The integration of charging records

4.5.1 The Server Side Implementation of the Web Services

There are differences between java codes of the server side web service[43] and those of the client side. While we can directly use the generated client side classes, we need to add our own implementation to the service side in order to handle the requests. The web service only describes how the client side can use the service, the concrete implementation of the service at server side are left open.

For the charge volume web service, when the client side sends charge volume request, the accounting data will be finally passed to an instance of class ChargeVolume_BindingImpl at the server side. Therefore, we add our own implementation to this class to process the accounting data.

[43] For the detailed explanations of the generated files for server side, please refer to the User's Guide of APACHE AXIS.

Since in our implementation, the AN domain does not have its own charging function, when the SP receive requests from the AN, it will first process the data and generate charging records. As it has been mentioned in section 4.3.2, we assume the Service Level Agreement between the AN and SP is 10 mega bytes of traffic costs one euro (10MB/€). The volume of traffic will be converted into amount of money. Then the charging records are put into database and wait for further processing.

The charge amount request from the 3PSP already consists of monetary information; the data need not to be further processed. We only add a method to class ChargeAmount_BindingImpl, which enables it to write the charging records to the database.

The charging records from AN and 3PSP are put into a same database but under different tables.

Figure 4-9 shows the table that contains charging records from the AN:

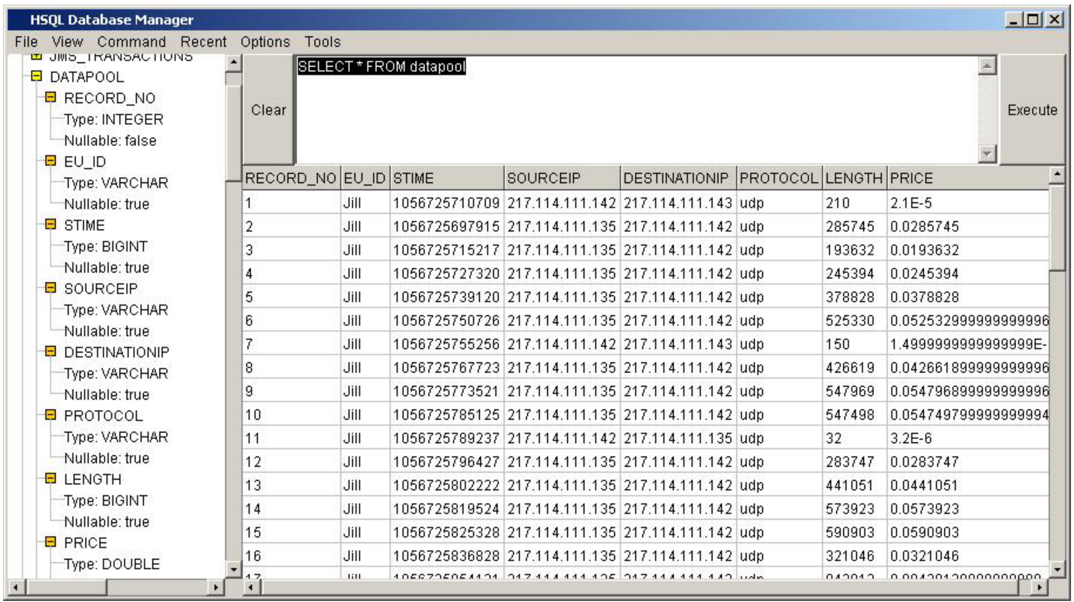


Figure 4-9: Charging Records of AN in the Database of SP

Figure 4-10 shows the table that consists of charging records from the 3PSP. The column "DATAFEE" is used indicate the data transportation fee of this application. When the SP first receives charge amount request from the 3PSP and fills the database, this column is left empty. It will be filled with data when the charging function in the SP try to integrate records from the AN domain with those from the 3PSP domain (i.e. the cost to transfer data).

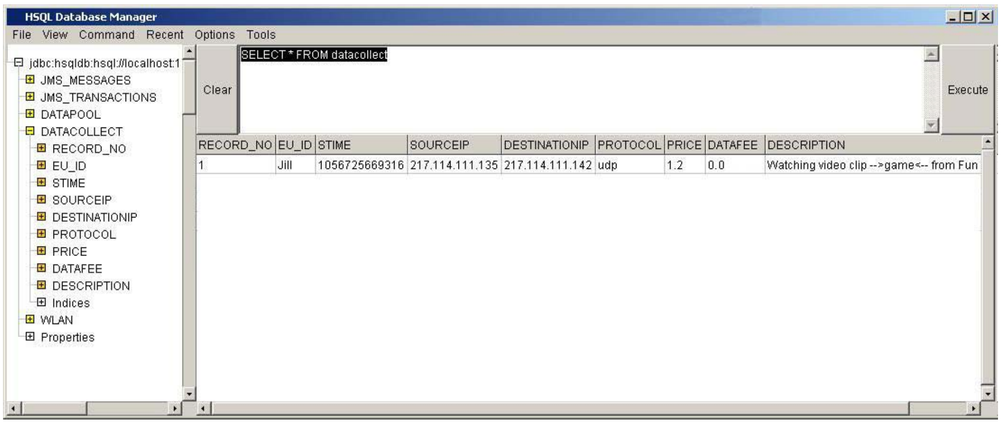


Figure 4-10: Charging Records of 3PSP in the Database of SP

4.5.2 Correlation of the Charging Records

Before describing the concrete implementation of the integration, we will explain the mechanism to correlate and merge the records. As it has been introduced in section 3.6.2, there are two levels of integration. Which level of integration will be used depends on the charging models adapted by the AN and the 3PSP. In our implementation, the charging models used by the AN and the 3PSP enable inter-domain integration. We give details in this section.

Suppose the charging request sent by the AN to the SP contains the following information (the referenceCode represents time difference, measured in milliseconds, between the record generation time and midnight, January 1, 1970 UTC), shown in Table 4-9:

endUserIdentifier	Jack
Volume	5,000,000 bytes
billingText	Source IP: 130.89.1.1 Destination IP: 217.114.1.1 Transfer protocol: UDP
referenceCode	1000000300000

Table 4-9: Information in the AN Charging Request

The charging function of the SP calculates the actual price of this volume of traffic according to a preset formula. It will generate a charging record that contains the following information, shown in Table 4-10:

endUserIdentifier	Jack
Amount	0.5 euro
billingText	Source IP: 130.89.1.1 Destination IP: 217.114.1.1 Transfer protocol: UDP

referenceCode	1000000300000

Table 4-10: AN charging Request Processed by the Charging Function of SP

And the charging request sent by the 3PSP contains the following information in Table 4-11:

endUserIdentifier	Jack
Amount	1 euro
billingText	Source IP: 130.89.1.1 Destination IP: 217.114.1.1 Transfer protocol: UDP Description: Streaming of "Geri's Game " from Fun Clips Inc.
referenceCode	1000000000000

Table 4-11: The Information in the 3PSP Charging Request

There is difference between the times of the two charging records. It is because the 3PSP initiate the streaming and it takes time for the packets to reach the AN (When the time is measured in milliseconds). The time synchronization between AN/SP/3PSP is very important in the correlation mechanism. We can use the Network Time Protocol[44] (NTP) for precise synchronization. However, in the implementation we have, the computers are synchronized with the accuracy of one second.

The integration function will merge the records containing the same Source IP, Destination IP and transfer protocol. It will generate one charging record (Table 4-12); with the description derived from the billingText from the 3PSP.

User ID	Subtotal	Service Fee	Data Transfer Fee	Time	Description
Jack	€1.5	€1.0	€0.5	16:53 July 07 2003	Streaming of "Geri's Game " from Fun Clips Inc

Table 4-12: The Merged Charging Information

If a video clip A lasts 20 minutes, and the AN generates charging requests every 5 minutes. The SP will receive one charging request from the 3PSP because of the 3PSP's pay per-view charging model A. It will receive at least 4 charging requests from the AN that are related to video clip A. The integration function will first retrieve a charging record from the AN and integrate it with the record from the 3PSP. The amount of money charged by the 3PSP is described as Service Fee, the amount of money charged by the AN is described as Data Transfer Fee. If SP finds

[44] http://www.ntp.org/

more charging records from the AN that are related to the integrated record, the integration function will add the amount of money to the Data Transfer Fee part. By this way, the integration function can integrate charging records coming from the same domain as well as from different domains.

The 3-tuple correlation mechanism can identify application which setup only one TCP connection or use only one UDP port to send data with one 3PSPs. While it is acceptable to be used in a simplified implementation, it can not be used in a general condition. For example, if the user watches two video clips at the same time from the same streaming server, then the AN will generate identical 3-tuple for these two clips and could not distinguish traffic of these two different applications.

As we mentioned in the pervious section, a 5-tuple correlation mechanism which includes the source port number and destination port number in addition to the 3-tuple will solve the problem. The 5-tuple mechanism can uniquely identify an application.

Generally, it will be much more complicated if we use other mechanisms to correlate the charging records from different domains. For example, to use the Session ID which is shared by the AN and the 3PSP when they are talking Session Initiation Protocol (SIP). This mechanism is upper layer protocol related. When the 3PSP runs different protocols, (e.g. RTSP, MMS), the AN should also be able to adjust to corresponding correlation mechanism. Due to the diversity of the application layer protocols, the metering function in the AN will be very complicated. On the contrary, the use of the 5-tuple information from IP packets provides a general and neat solution.

The disadvantage of the 5-tuple mechanism is that it does not work well when there are Network Address Translation (NAT) proxies lie in between the AN and 3PSP and change the source and destination IP addresses of a packet. But in today's Internet, most of the proxies are placed at the edge of the network. The backbone of the Internet will not change the source and destination addresses of an IP packet. In case there is a NAT proxy in an access network, we can put the metering function together with the NAT proxy. Because the NAT proxy knows the mapping of the IP addresses between the inside network and outside network, the accounting server will bind the end-user with the IP address and port number with the outside network. In this way, it is still possible to merge the charging records from AN and 3PSP base on the 5-tuple information.

Another disadvantage of the 5-tuple mechanism is that it can not support Virtual Private Network (VPN). Currently, the most widely used technology in VPN is IP security (IPsec), which offers authentication, integrity and confidentiality [KENT98]. On providing the confidentiality, data payload in an IP packet will be encrypted (even certain sensitive IP address). So, the 5-tuple information can not be acquired from the VPN traffic.

4.5.3 The Integration of Charging Records

The integration function is implemented through an MBean. The advantage of MBean is mentioned in section 4.3.2. There are three java classes in the implementation; the source codes can be found in the CDROM that comes with the thesis; the relationships between these classes are shown in Figure 4-11:

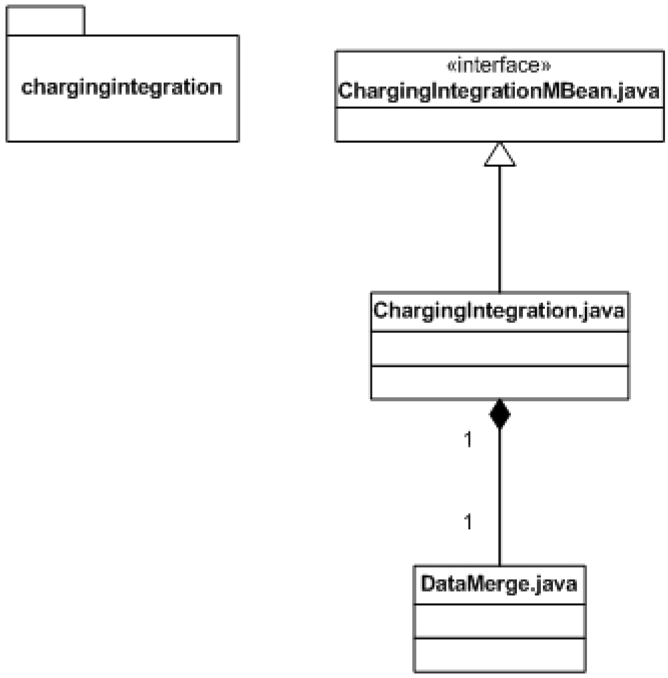


Figure 4-11: Class Diagram of SP Implementation

Class ChargingIntegrationMbean and class ChargingIntegration are necessary implementation according to the MBean design pattern [JSTE02]. Our main effort is in class DataMerge; this class retrieves charging records from the database and integrates them according to the 3-tuple correlation principle (section 4.5.2).

Figure 4-12 shows the integrated records. Notice when compared with Figure 4-10, the "DATAFEE" column is filled with data which indicate the transportation fee of this application.

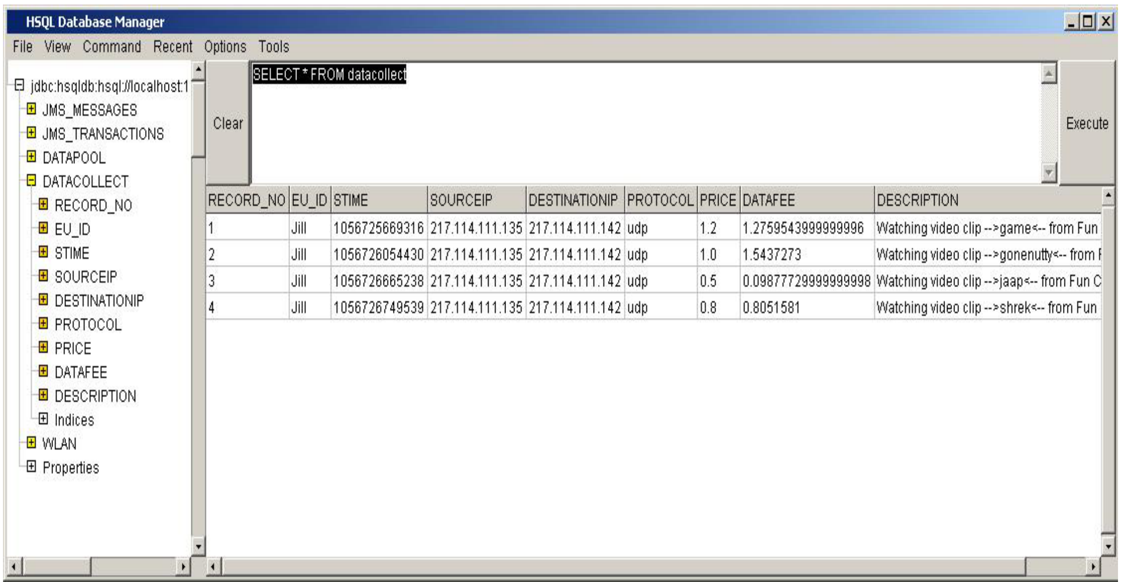
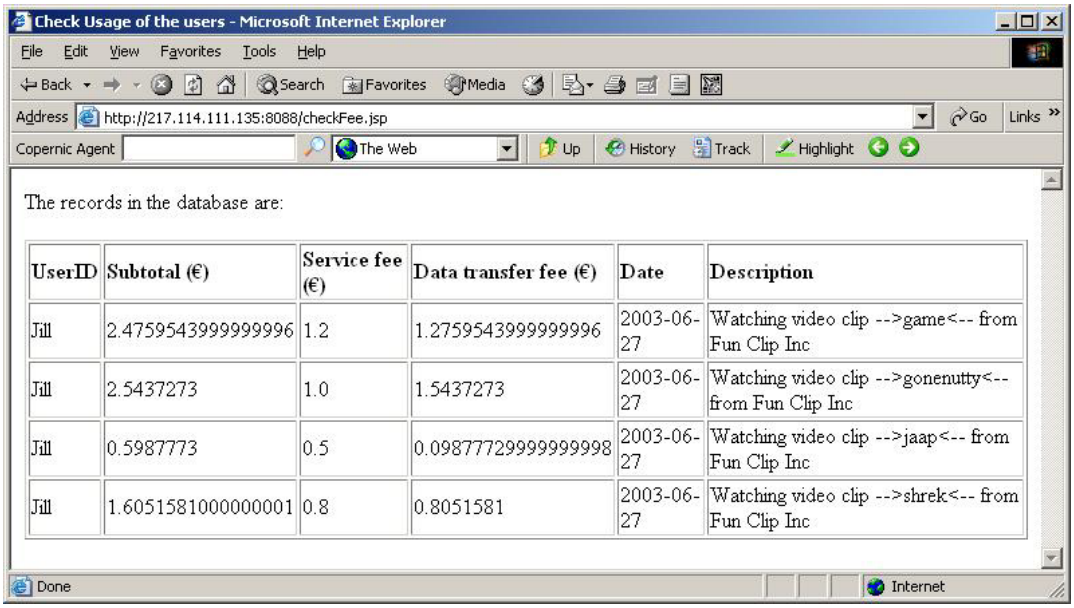


Figure 4-12: Screenshot of Integrated Records in SP

In addition to the basic implementation, we also implemented JSP code which enables the user to query the usage and see their deficit online through webpage. Figure 4-13 shows the GUI.



UserID	Subtotal (€)	Service fee (€)	Data transfer fee (€)	Date	Description
Jill	2.4759543999999996	1.2	1.2759543999999996	2003-06-27	Watching video clip -->game<-- from Fun Clip Inc
Jill	2.5437273	1.0	1.5437273	2003-06-27	Watching video clip -->gonenutty<-- from Fun Clip Inc
Jill	0.5987773	0.5	0.09877729999999998	2003-06-27	Watching video clip -->jaap<-- from Fun Clip Inc
Jill	1.6051581000000001	0.8	0.8051581	2003-06-27	Watching video clip -->shrek<-- from Fun Clip Inc

Figure 4-13: Screenshot of the GUI Where the Users Check Usage

The best way to predict the future is to invent it.

-- Alan Kay

5 Conclusions and Future Work

The chapter offers conclusions of the research presented in the thesis and suggests directions for future research.

Section 5.1 presents the conclusions; section 5.2 recommends direction for future research.

5.1 Conclusions

The objective of the project (section 1.2) is to develop an architecture for accounting facilities for the Mobile and Wireless Internet environment, which has been envisioned by the 4GPLUS project. The basic requirement is that the accounting facilities should provide integrated accounting information to the end users. At the same time, the accounting facilities need to:

- Be open, which means the accounting facilities allow ANs and 3PSPs to freely use and change charging models;
- Support real-time and non real-time accounting;
- Support prepaid and postpaid users.

5.1.1 Conclusions of the Design

From the design perspectives, the accounting facilities meet the requirements. It will be illustrated in the next paragraphs.

Openness:
The accounting facilities allow the end user to receive one integrated bill from the SP. As it has been explained in section 3.6.2, the integration of the charging

information can happen at two levels: intra-domain and inter-domain. Which level of integration will happen depends on the charging models used by the ANs and 3PSPs (section 3.6.2).

In our design, the access network service providers and the third party service providers can choose different charging models and can change them freely (e.g. AN can choose charging models based on time, volume, connection and change them without much effort). The changing of charging models will only affect a single domain or in other words, one service provider (i.e. AN has to adjust part of its accounting facilities to adapt to the new charging model).

Support of real-time and non real-time accounting:
The accounting facilities support both real-time and non real-time accounting . We use a time window which enables the AN/3PSP/SP to set the frequency of the accounting related activities (e.g. metering, charging). There is no clear distinction between real-time and non real-time accounting; we can shorten the size of time window in our accounting facilities to make them a real-time system.

Support of prepaid and postpaid users:
A money/credit reservation mechanism has been introduced in our design to support prepaid users. By applying this mechanism also to the postpaid users, the accounting facilities can support both type of users using one mechanism. Using this mechanism, the AN and the 3PSP do not need to distinguish between these two types of users. Only the SP who has actually monetary/contract relation with the end users distinguishes them. This reduced the complexity of the accounting facilities.

5.1.2 Conclusions of the Implementation

The implementation confirms us that the users can receive one integrated bill from the SP. Because the specific charging models we choose for the AN and the 3PSP, the integration of the records can happen both intra-domain (the charging records come from one domain but related to one application can be integrated) and inter-domain (the charging records come from different domains but related to one application can be integrated). The usage of the AN and 3PSP in a specific application is presented in one record to the end user. This means to the user that he/she will get a clear and short bill upon the usage, while getting rid of detailed description of each AN and 3PSP he/she accesses.

We can also change the frequency of the charging requests sent by the AN to realize real-time or near real-time charging. The JBoss implementation environment we choose enables us to manipulate the time interval through a console easily.

However, due to the limited time, we did not implement the money/credit reservation mechanism. According to our experience using web service technology, the money/credit reservation mechanism can be added to the system without much effort.

Briefly, the design of the accounting facilities satisfies the objective of the project and it is partly proved by our concrete implementation.

5.2 Recommendations for Future Research

Security issues are of high importance for accounting facilities. In addition to authentication and authorization, which are closely related to accounting, there are also other quality aspects in security area such as: confidentiality, integrity, non-repudiation and availability. Analysis and evaluation of the security of the system definitely requires attention. Moreover, how to integrate the accounting facilities with authentication and authorization requires further investigation.

The accounting facilities were designed mainly for the situation where there is only one SP. However, the significance of 4GPLUS project lies in the federation of SPs. In our design, a "Charging" functional block for roaming users was proposed (Figure 3-9); further examination of this solution is suggested. There could be other solutions as the federation mechanism between service platforms evolves.

We kept scalability in mind during our design, however the scalability of the accounting facilities need to be further studied. Possible direction could be the scalability of the correlation mechanism and the scalability of the PaylayX payment web services.

We introduced the role of brokers in section 3.3, it is interesting to study how to distribute functionalities of AN, SP and 3PSP to various brokers and how the brokers improve the scalability of the system.

We also introduced some supplemental services in addition to the "one bill" service in section 3.6.3 (such as the provision of pricing information), and we considered them as valuable features of the accounting facilities in the future. Implementation and integration of them into our accounting facilities are highly recommended.

We suggested using ParlayX as the interfaces between the accounting facilities in different domains. However, the current ParlayX standards are limited as we have analyzed in section 3.7.2. It is recommended to enhance the ParlayX standard by designing and building up web services according to our suggestions.

The AN part of our implementation mainly focused on WLAN; it is worthy to expand the range of access networks to include GPRS and UMTS networks.

References

[ABOB00] B. Aboba, J. Arkko, D. Harrington, "Introduction to Accounting Management", RFC 2975, October 2000.

[BIEM02] F.P.M Biemans, et. al., "Reference models for networked applications", ISBN 90-75176-31-7, Telematica Instituut, 2002.

[BROW99a] N. Brownlee, "Traffic Flow Measurement: Meter MIB", RFC2720, October 1999.

[BROW99b] N. Brownlee, C. Mills, G. Ruth, "Traffic Flow Measurement: Architecture", RFC 2722, October 1999.

[BROW00] N. Brownlee, A. Blount, "Accounting Attributes and Record Formats", RFC2924, September 2000.

[CUSH01] John Cushnie, David Hutchison, Huw Oliver, "Internet Services: Infrastructure for Charging and Billing", Proc. Infomatik 2001, Vienna, Austria, Sept 2001, OCG (Austrian Computer Society) (ISBN 3-85403-157-2), pp104-111.

[DAVI95] David D. Clark, "A Model for Cost Allocation and Pricing in the Internet", http://www.press.umich.edu/jep/works/ClarkModel.html

[ELSA02] El-Sayed, M., Jaffe, J. "A view of telecommunications network evolution", IEEE Communications Magazine, December 2002, pp74-81.

[EMCD00] EMC World Cellular Database, "I-MODE, The Convergence of Content and Mobility in Japan a cellular packet data case study", http://www.emc-database.com/website.nsf/index/i-mode June 2000.

[FANK99] Fankhauser, G., B. Stiller and B. Plattner, "Arrow: A flexible architecture for an accounting and charging infrastructure in the Next Generation Internet", Netonomics, vol. 1, 1999, pp. 201-223.

[GIBB99] Gibbens, R., Kelly, F., "Resource pricing and the evolution of congestion control", Automatica 35 1999, pp1969-1985.

[HART99] Felix Hartanto, Georg Carle, "Policy-based billing architecture for Internet differentiated service", Proceedings of IFIP Fifth International Conference on Broadband Communications (BC'99), November 1999.

[ITUD03] "General tariff principles (ITU-T Recommendations, D-series)", http://www.itu.int/rec/recommendation.asp?type=products&lang=e&parent=T-REC-D, Feb 2003.

[JERO02] Jeroen van Vugt, et. al., "Operational aspect of the Federated Service Platform ", https://doc.telin.nl/dscgi/ds.py/view/Collection-5292, December 2002.

[JSTE02] J. Steven Perry, "Java Management Extensions", ISBN 0-596-00245-9, 2002, chapter two.

[KENT98] Kent, Stephen., Atkinson, Randall. "Security Architecture for the Internet Protocol," RFC 2401, November 1998.

[KRAA02] "Research challenges in Wireless Internet." http://www.cs.helsinki.fi/u/kraatika/Courses/EVC-WINsummerschool/ResearchChallengesinWirelessInternet.pdf

[LAAT00] C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, D. Spence, "Generic AAA Architecture" RFC2903, August 2000.

[MACK95a] MacKie-Mason, J.K., and H.R. Varian. "Pricing the Internet", In Public Access to the Internet , B. Kahin and J. Keller ,eds., MIT Press, 1995., pp. 269-314

[MACK95b] MacKie-Mason, J.K., and H.R. Varian. "Some FAQs about usage-based pricing". http://www.sims.berkeley.edu/~hal/people/hal/papers.html

[MACK95c] MacKie-Mason, J.K., and H.R. Varian. "Some economics of the Internet". In W. Sichel (Ed.), Networks, Infrastructure and the New Task for Regulation, University of Michigan Press, Ann Arbor, 1995.

[MACK01] Macker, J.P.; Park, V.D.; Corson, M.S. "Mobile and wireless Internet services: putting the pieces together". IEEE Communications Magazine, June 2001 pp148-155.

[MOBY02] Moby Dick WP4, "Design and Specification of an AAAC Architecture Draft on administrative, heterogeneous, multi-provider, and mobile IPv6 sub-networks", http://www-int.berkom.de/~mobydick/, January 2002.

[MPFO02] "Mobile Payment Forum White Paper" http://www.mobilepaymentforum.org/pdfs/mpf_whitepaper.pdf, December 2002

[NDMU02] "Network Data Management – Usage (NDM-U) For IP-Based Services V3.1.1", http://ipdr.org/download-docs/index.html, October 2002.

[ODLY00] Andrew Odlyzko, "The history of communications and its implications for the Internet", http://www.research.att.com/~amo/doc/history.communications0.pdf.

[OLEA99] Kevin O'Leary, "Charging for Premium IP Services", http://www.teltec.dcu.ie/susie/D4.PDF, April 1999.

[PARH01] R.Parhonyi, "Provider Based Accounting Architecture", Deliverable 5.7 of the ING project, April 2001.

[PARH02] Robert Parhonyi, " Status of the provider based accounting architecture (PBA arch)", http://ing.ctit.utwente.nl/WU5/D5.15/parhonyi-pba.pdf, January 2002.

[PARL02a] Parlay Group, "Parlay API 4.0: Parlay X Web Services White Paper", http://www.parlay.org/specs/library/ParlayX-WhitePaper-1.0.pdf December 2002.

[PARL02b] Parlay Group "Parlay 4.0: Parlay X Web Services Specification v.091", December 2002.

[PEEL02] B. Peelen, DJ. Plas, J. Vugt, F. Vervuurt, M. Vooren, M. Bargh "Functional specification of the federated platform architecture", December 2002.

[SPRE00] R. A. M. Sprenkels, R. Párhonyi, A. Pras, B. J. van Beijnum, B. L. de Goede "An Architecture for Reverse Charging in the Internet", Proceedings of the IEEE workshop IPOM 2000, Cracow, September 2000.

[STIL98] Burkhard Stiller, George Fankhauser, Bernhard Plattner, Nathalie Weiler "Customer Care, Charging, Accounting, Billing, and Pricing", http://www.tik.ee.ethz.ch/~stiller/projects-and-research.html#ChargingandAccounting , Febuarary 1998.

[TOKM00] Tokmakoff, A. (Ed.), & Wibbels, M. "Description of Experimental Results", https://doc.telin.nl/dscgi/ds.py/Get/File-11895/Description_of_Experimental_Results.pdf/, November 2000.

[TS1202] 3GPP TS 12.15 V7.7.0 "GPRS charging", http://www.3gpp.org , December 2002.

[THAN02] Thanasis G. Papaioannou and George D. Stamoulis, " Design and Implementation of a Charging and Accounting Architecture for Secure VPN Services to Mobile Users", http://www.internode.org/workshop_presentations/notes/Stamoulis_notes.pdf, January 2002.

[TS3202] 3GPP TS 32.200 V5.2.0 "Telecommunication management, Charging management, Charging principles", December 2002.

[ZSEB02] T. Zseby, S. Zander, G. Carle "Policy based accounting", RFC3334, October, 20002.

Appendix A - WSDL files

The WSDL file of charge volume service

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="ChargeVolumeService"

targetNamespace="http://www.jimmie.com/wsdl/ChargeVolumeService.wsdl"
   xmlns="http://schemas.xmlsoap.org/wsdl/"
   xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
   xmlns:tns="http://www.jimmie.com/wsdl/ChargeVolumeService.wsdl"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">

   <message name="VolumeChargingRequest">
    <part name="endUserIdentifier" type="xsd:string"/>
    <part name="volume" type="xsd:long"/>
    <part name="billingText" type="xsd:string"/>
    <part name="referenceCode" type="xsd:string"/>
   </message>

   <message name="VolumeChargingResponse">
     <part name="ok" type="xsd:string"/>
   </message>

   <portType name="ChargeVolume_PortType">
      <operation name="chargeVolume">
         <input message="tns:VolumeChargingRequest"/>
         <output message="tns:VolumeChargingResponse"/>
      </operation>
   </portType>

   <binding name="ChargeVolume_Binding"
type="tns:ChargeVolume_PortType">
      <soap:binding style="rpc"
         transport="http://schemas.xmlsoap.org/soap/http"/>
      <operation name="chargeVolume">
         <soap:operation soapAction="chargeVolume"/>
         <input>
            <soap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
               namespace="urn:prototype:volumechargingservice"
               use="encoded"/>
         </input>
         <output>
            <soap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
               namespace="urn:prototype:volumechargingservice"
               use="encoded"/>
         </output>
      </operation>
   </binding>

   <service name="ChargeVolume_Service">
      <documentation> WSDL File for ChargeVolume_Service
</documentation>
      <port binding="tns:ChargeVolume_Binding"
name="ChargeVolume_Port">
         <soap:address
            location="http://217.114.111.135:8080/jboss-
net/services/ChargeVolume_Port"/>
      </port>
   </service>
</definitions>
```

The WSDL file of charge amount service

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="ChargeAmountService"

targetNamespace="http://www.jimmie.com/wsdl/ChargeAmountService.wsdl"
   xmlns="http://schemas.xmlsoap.org/wsdl/"
   xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
   xmlns:tns="http://www.jimmie.com/wsdl/ChargeAmountService.wsdl"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">

   <message name="AmountChargingRequest">
    <part name="endUserIdentifier" type="xsd:string"/>
    <part name="amount" type="xsd:double"/>
    <part name="billingText" type="xsd:string"/>
    <part name="referenceCode" type="xsd:string"/>
   </message>

   <message name="AmountChargingResponse">
     <part name="ok" type="xsd:string"/>
   </message>

   <portType name="ChargeAmount_PortType">
      <operation name="chargeAmount">
         <input message="tns:AmountChargingRequest"/>
         <output message="tns:AmountChargingResponse"/>
      </operation>
   </portType>

   <binding name="ChargeAmount_Binding"
type="tns:ChargeAmount_PortType">
      <soap:binding style="rpc"
         transport="http://schemas.xmlsoap.org/soap/http"/>
      <operation name="chargeAmount">
         <soap:operation soapAction="chargeAmount"/>
         <input>
            <soap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
               namespace="urn:prototype:amountchargingservice"
               use="encoded"/>
         </input>
         <output>
            <soap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
               namespace="urn:prototype:amountchargingservice"
               use="encoded"/>
         </output>
      </operation>
   </binding>

   <service name="ChargeAmount_Service">
      <documentation> WSDL File for ChargeAmount_Service
</documentation>
      <port binding="tns:ChargeAmount_Binding"
name="ChargeAmount_Port">
         <soap:address
            location="http://localhost:8070/jboss-
net/services/ChargeAmount_Port"/>
      </port>
   </service>
</definitions>
```