Billing Service for TINA Business Model

Chien-Yu Chen, Russell Achterberg, Setumo Mohapi, Hu Hanrahan
Centre for Telecommunications Access and Services[1]
Department of Electrical Engineering
University of the Witwatersrand, Johannesburg
Private Bag 3, 2050, South Africa
{j.chen; r.achterberg; s.mohapi; h.hanrahan}@ee.wits.ac.za

Abstract: In the past decade the telecommunication and information service are converging. New services are introduced rapidly to meet customers' demand and a convenient service billing mechanism is required. A TINA-based billing model is presented in this paper to provide telecommunications service users with a centralised billing service to support the dynamic service environment.

I. INTRODUCTION

The telecommunication industry has changed rapidly in the past decade due to deregulations and advancement in technology. It has changed from a relatively static service environment that introduced changes very slowly to one that is able to rapidly offer new services responding to the customers needs, under competitive pressures. However, the billing services offered by typical telco's billing systems are proprietary, rigid and non-flexible. In the dynamic telecommunications service delivery environment, a versatile telecommunications service charging system is necessary to provide billing for the multi-user services offered over different network technologies and domains.

TINA provides a framework which offers flexible billing control and management for dynamic service provision platforms. A Billing Service Provider (BSP), built within the TINA-Retailer domain, is proposed to provide a centralised billing service, which allows for customers needing to only interact with only trusted party, when handling sensitive financial information. In this way, instead of the consumer presenting their financial details directly to various (unknown) parties when placing an order or purchasing services/products, a trusted relationship can be established between the BSP and the customers.

The proposed BSP will operate in a Distributed Processing Environment (DPE). The DPE model is an abstract distribution environment that serves many different requirements expressed by the applications from the different stakeholders specified in the TINA business model [1]. The TINA DPE is implemented based on the Common Object Request Broker Architecture (CORBA), specified by the Object Management Group (OMG) [2], and it brings benefits such as programming language independency, access transparency, design flexibility, and software component reusability. Under this object-oriented programming environment, the billing service applications will be written as a set of interacting objects that may, for operational reasons, be located on different nodes from each other [3]. The data used in the billing service will also be stored as objects in a database to allow for ease of management, as well as for presentations to the customer.

The purpose of this paper reports work in progress on demonstrating the billing service implementations on the SATINA platform. In section II, two accounting management issues namely the usage measurement and billing of services, as well as a proposed TINA BSP model are presented. In section III, the proposed BSP is analysed in more detail using the TINA computational and service models. Section IV presents the implementation technologies and section V discusses further possible research areas. Finally, a conclusion follows at the end of this paper.

II. BACKGROUND

Two accounting management problems were

[1] This work is supported by Telskom SA Limited, Siemens Telecommunications and the THRIP Programme of the Department of Trade and Industry.

identified in telecommunications networks by TINA:

- *Usage measurement*: Collection of resource usage statistics on a service basis; i.e. duration and amount of resources used in supporting of a service.
- *Billing services*: Computation of charging ("bills") to be paid by each subscriber and the billing policies for the services; delivery of billing notices and tracking payment by subscribers [1].

The first problem falls under the scope of accounting management in the TINA Network Resource Architecture (NRA) and the second issue can be treated as a service (or a collection of services) that may be designed much like any other TINA service.

A. TINA business model

In the TINA business model, five basic roles are defined namely the Consumer/Customer, the Retailer, the Broker, the Connectivity Provider and the 3rd Party Service Provider. In general, the responsibility of the Retailer is to serve the stakeholders in the consumer business domain. The number of retailers that can be engaged in a global system can be anything from a few to several thousand and a Retailer can be anything from a large corporation to a small "garage" company [4]. The Retailer will be the focal point of the cash flow since it is through the Retailer that 3rd Party Service Provider serves/products are offered to and accessed by the Consumer. Hence a standardised billing/accounting mechanism will be required to provide a mechanism where service generated customer billing information can be processed.

One of the specified function of the Retailer is to collect accounting information for the purpose of billing. In general, this is required for each invoked service as well as for the network resources and connectivity used. However, due to the distributed nature of the TINA network architecture, the retailer business role can also carry out one or more of the other business roles defined in the business model [5].

Hence, the proposed BSP is an integrated component of the Retailer itself as shown in Figure 1. The BSP can be regarded as a subsystem of the Retailer, which will translate all the collected accounting information from various parties into billing information. The billing information can be displayed to the user during the on-going service session, and must be stored in a centralised billing database. The customers will be able to access the centralised billing information through the registered Retailer, which is responsible for Consumer profile management.

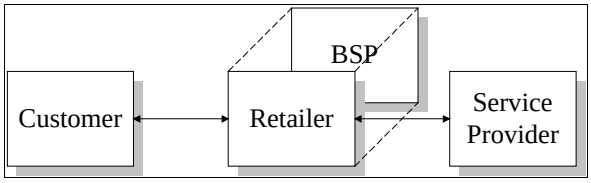

Figure 1 TINA business model

The problems that are required to be addressed in implementing such a service are discussed in the following section.

B. Problem Domain

The billing information is generated from the collection of accounting information that was produced by accounting object entities distributed across the network in a set of complex activities. It is important that the following issues are addressed when the TINA accounting architecture is adopted for billing purposes:

- **Distributed Accounting**: the accounting entities in TINA such as an accounting policy manager, metering manager, etc. are distributed across the network, as well as the objects to be accounted. In this distributed management context, the question is how these accounting management objects can be linked to communicate with each other and how the accounting event information collected can be turned into consistent billing information.
- **Dynamic Aspects in Accounting**: this refers to the situation where the tariffing structure is subject to changes due to dynamic market needs, the deployment of new services and withdrawal of old services. It is also sometimes possible for that even the metering cycle to be dependent on a specific service and its corresponding tariffing structure**.** For example, a flat rate tariffing scheme in a certain service category may require that only at times that a session component is being accessed to be reported as metering information. Therefore, it is necessary for the

charging mechanism used for services can be dynamically added, removed and modified easily, without impacting on the performance of the entire accounting service system.

- **Flexible Role Assignment in Accounting**: refers to the situation that the traditional fixed distinction between the user and the service (network) provider needs to be re-examined and re-defined, in relation to the distributed client-server concept in TINA. For example, in the layered network concept of the TINA connection management architecture, a service (network) provider can be a user (client) of another service (network) provider. Hence, metering/charging of service activities can no longer be attributed to the user-id, which is used to represent the identification of the user party that is using the service. Instead it requires that the task of accounting be properly divided between the client and the server (or among the user, the agents, and the resource provider) [6].

In summary, this paper will investigate a method of providing proper information flow between the different accounting management objects. The resulting billing system has to be extremely flexible; new features and services must be able to be added to the system and billed without impacting negatively on the billing flow for other existing services.

III. BILLING IN THE TINA SERVICE ARCHITECTURE

The proposed implementation concept for the TINA-based BSP is described here using the TINA Service Architecture (SA).

A. Computational model

Figure 2 illustrates the computational objects in an accounting management domain. Within an accounting domain, there are three types of computational objects [1]:

- *Accountable objects (AO)*: an accountable object represents a resource that is subject to accounting. A computational object representing a network resource is an accountable object if it supports an interface that provides accounting control operations. When an object representing a resource is created, it retrieves the accounting policy applicable to that resource from the accounting policy manager. If the accounting parameters of a domain change, the accounting policy manager notifies all accountable objects in the domain through their individual accounting control interfaces of the new parameter values.
- *Accounting Policy Manager (APM)*: this object maintains the accounting policy for services provided within a domain. The policy is represented as a set of tagged value pairs, where each tag denotes a policy parameter (or rule). The object provides operations for querying and modifying the charging policy parameters by having the *Accounting Policy Query* Interface the *Accounting Policy Control Interface*. Thus the accounting policy can be modified by an authorised party and applied to appropriate accounting objects.
- *Metering manager (MM)*: this object receives accounting event reports from all accountable objects in the domain and logs them. An object in one domain can subscribe to event reports from another domain by requesting the metering manager of the latter domain.

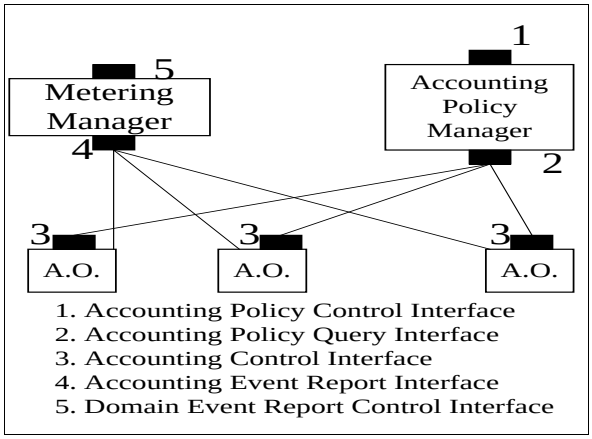

Figure 2. Accounting Management Computational Objects

The problems mentioned in section II.B are addressed by using these computational components. The APM provides a unified way of maintaining the charging policies applicable to different services. The introduction of any new service charging policies and modifications of existing service charging policies can be implemented using this object and then be made accessible to all the accountable objects. In this way,

the distributed accounting service components are under the control of the same accounting manager; hence, the consistencies of the billing information can be maintained. This also allows for a simple change in charging mechanism/policy within one object, in response to the dynamic nature of the services. By creating the service session using the service components, the involved parties are identified by the service objects created when the service was invoked. This would overcome the problem of the metering/charging attributes being limited to the use of the user-id.

B. Accounting Management Component Ladder (AmcLadder)

The AmcLadder is a generic object for accounting management, from which other components such as a SSM (Service Session Manager) and USM (User Service Session Manager) can be derived; amcLadder is made up of service components such as the SSM, USM, etc. (see Figure 3). When an AmcLadder is created (SSM, CSM etc.), its notification destination is set to another AmcLadder, which is to be positioned above in the ladder. For example, when the CSM is created its notification destination is set to the corresponding SSM; the SSM sets its notification destination to the USM, etc. so that an accounting event path is formed among the session components, from the bottom up (also see section C.2). In other words, the AmcLadder does not exist as a stand alone object, it is made up of other service components [7].

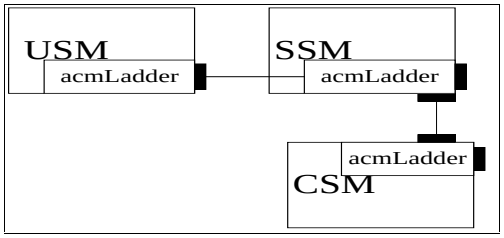

Figure 3 AmcLadder components

However, although the interfaces between these accounting objects are specified, no Interface Definition Language (IDL) has been defined [8]. Thus further development for the actual implementations are still necessary.

C. Service Component model

Two types of accountable events can occur for interactions between a 3rd party service provider and a customer:

- Billing for products/services purchased during the session. For example, a physical product purchased or requesting for a video item from a video on demand (VoD) service provider.
- Billing for the use of connectivity/network resources for the delivery of the service purchased. E.g. the network resource reserved in order to deliver the VoD service or to support other types of data stream flow.

In the following sections, the two charging scenarios are discussed. The discussions are based upon the following assumptions:

- Service session creation: service components such as the UA, USM, SSM, etc. are already created and are in place.
- Network resource components set-up: network resource components such as the CC, LNC, TCM, etc. are already created and are in place.
- Stream binding set-up: necessary Network Flow End Point (NFEPs) and Stream Flow End Point (SFEPs) are already provisioned and bound together in stream bindings.

Although all of these above steps are part of the accounting management, they do not directly correlate with usage accounting; they should rather be considered as provisions for usage accounting.

1. Charging model for Service/product purchase

In this service session, the customer purchases a service/product from the provider therefore no connectivity considerations needed to be concerned. The customer is billed for what is asked for and the cost of the item(s) will be recorded.

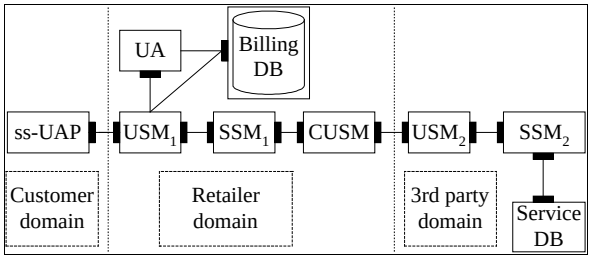

Figure 4 Service accounting model

In Figure 4 the user performs the purchase operation to obtain the desirable product/service. The USM and the SSM exists for the management purpose of this

service session [8]. Since the service is provided by another service party, a compUSM is required to support the composition of the service session. The SSM_2 comprises the service-specific logic offering the service/product and it has the costs of the services/products offered stored in a separate database (DB).

When the customer makes a purchase, the billing information gets pushed all the way back to the customer USM and is then stored in a billing DB to give the record permanency. This information can be pushed to the User Agent (UA) and then be displayed via the corresponding Provider Agent (PA), which is not shown here, to the user when on-line billing is used [7]. Also, the customer is able to gain access to their billing information, using the PA, from the UA and the billing DB [8].

2. Charging model for connectivity usage

In this situation, the bare transport level traffic is measured, which corresponds to the measurement of the Network Flow Connections (NFC) [7]. The usage information of the NFC is collected by the Connection Coordinator (CC) and then passed on to CSM. Then the filtered accounting events, which may cause a change in the Customer's billing status, are passed to the SSM, which in turn passes the events to the corresponding USM or to UA (see Figure 5). The quality of service (QoS) obtained by the Customer can be considered as an accounting event indication. Different levels of QoS delivered to the customer will be charged differently.

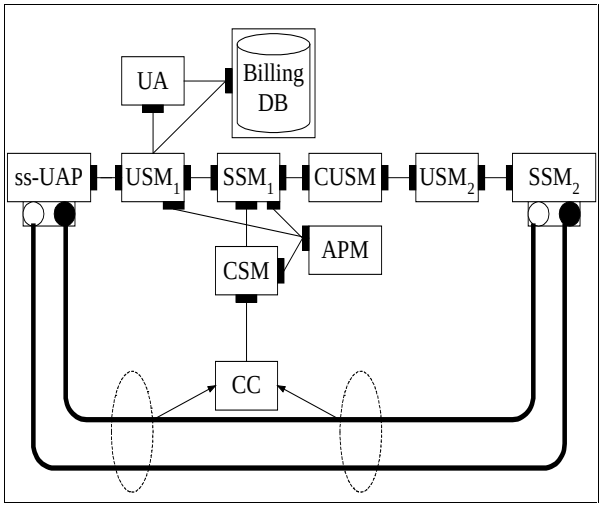

Figure 5. Connection accounting model

An APM exists in the retailer's domain to provide management for the charging mechanism. The billing information is stored in DB and is accessible through UA in the same manner as described in the previous section.

IV. IMPLEMENTATION TECHNOLOGY

In this section, the choice of technology/software to implement the proposed BSP is presented.

XML is proposed for the purpose of information flow between the computational objects. The power of XML lies in its ability to represent the data itself and to define its structure and meaning. This is useful in terms of providing a meaningful way to transport the billing information between the objects. It can also be used in conjunction with JAVA-platform programs that send and retrieve information to client/server applications, where the XML-coded data is transmitted between the client and the server, and an interaction between the customer and the BSP takes place. The other advantage of using XML is that the stored information can be organised and managed simply as the data is clearly defined.

In order to utilize the full benefit of the object-oriented programming environment, the billing service will be developed using the JAVA language, and operated over the distributed processing infrastructure, using CORBA. All the constituent object in the CORBA based architecture have IDL-defined interfaces which allows the customer to deal with a single business object component, even though it may be distributed over several objects running in different server machines.

The Retailer is able to serve the customer using a JAVA applet. The applet is downloaded from the server (Retailer) to the JAVA enabled browser and run through the JAVA run-time security gauntlet and then loaded into memory [9]. The JAVA applet includes the IDL-generated client stubs, which then invoke objects on the ORB server. The billing information will be created and stored as CORBA objects. In this way, CORBA allows the user to interact with the server by clicking on any of the components embedded from the browser and dynamically makes requests on service applications from the server.

V. FURTHER WORK

The proposed billing service model aims to provide

customers with easy access to their billing information while giving the service party complete control over their billing data.

This BSP fit in with the TINA business model as part of the Retailer domain. This falls short of an universal billing service as the billing information collected is limited to a specific Retailer that the customer obtains services from; a customer can receive more than one bill as more than one Retailer may be engaged. Further work is required to make this BSP an independent 3rd party service provider that can be utilized by different "Retailer" stakeholders. In this way a true centralised billing system containing billing information from different Retailers can be presented to the customer.

VI. Conclusion

A TINA based billing service model has been presented. The accounting management of this service is composed of three computational objects namely the accountable object, which represents a subject for accounting, the accounting policy manager, which manages the accounting strategy, and the metering manager, which is responsible for the measurement of resources used. These objects form an accounting management component ladder, which is distributed amongst various service components. When services are used, the accounting information is generated by these accounting objects and is then translated into billing information, which is stored in a database. This allows the customer to access the centralised billing information through the service session provisioned by the Retailer. This work in progress aims demonstrate the flexibilities of this model in comparison to a proprietary billing system.

VII. References

[1] Y. Inoue, M. Lapierre, C. Mossotto, "*The TINA book: A co-operative solution for a competitive world*", Prentice Hall Europe, 1999.

[2] OMG, http://www.omg.org/gettingstarted/corbafaq.htm

[3] TINA-C, "*Overall Concepts and Principles of TINA*", Version 1.0, February 1995.

[4] M. Schenk, "*OMG white paper: Open Service Marketplace*", Version 1.1, June 2000.

[5] TINA-C, "*Business Model and Reference Points*", Version 4.0, May 1997.

[6] TINA-C, "*Accounting Management Architecture*", Version 1.3, February 1996.

[7] TINA-C, "*Network Components Specification*", Version 2.2, December 1997.

[8] TINA-C, "*Service Component Specification*", Version 1.0b, January 1998.

[9] R. Orfali, D. Harkety, "*Client/Server Programming with JAVA and CORBA*", John Wiley & Sons, INC. 2nd Edition, 1998.