# A TINA Based Multimedia Service Delivery Platform for Parlay Service Deployment and Provisioning

T. Machethe, and H.E. Hanrahan

Centre for Telecommunications Access and Services[1]

University of the Witwatersrand, Johannesburg, South Africa

e:mail: {t.machethe,h.hanrahan@ee.wits.ac.za}

***Abstract* –Traditional Telecoms and IT related network technologies are converging. The present day Internet context does not meet the connectivity and service requirements of end-to-end provision of services in Telecoms networks, whereas current technology implemented in Telecoms networks does not allow for the rapid creation or delivery of feature rich services as found in the Internet. However, the convergence of Telecoms and IT related networks has led to the formation of the Next Generation Networks (NGN). The NGN is a multi-service network which inter-works with the Public Switched Telephone Network (PSTN), the voice network and the data network provided by Internet. Through network independent APIs such as OSA-Parlay, the NGN slowly migrates and converges these two networks, voice and Internet, into a common packet infrastructure. The OSA-Parlay group provides a set of open network independent APIs through which cross network applications can be accessed and developed. In accordance with the vision of easy and rapid service creation in the NGN, the TINA-Consortium (TINA-C) has developed a service architecture to support the development and deployment of services in the NGN. This architecture consists of reusable components which can be used in the implementation of NGN services. This project aims to investigate the design and implementation of a TINA based service delivery platform (SDP) capable of delivering multi-network multimedia services to end users through a user interface using the OSA-Parlay interface.**

## I. INTRODUCTION

In today's telecoms and IT infrastructures access to services is provided through a variety of access networks. For example, telephony is usually accessed through the PSTN, mobile telephony or messaging through a GSM network, and the Internet through an IP network. The range of applications is increased through steps already taken to standardize access to services through a single service delivery platform [1, 2]. In particular, the Parlay Group, an open, multi-vendor telecoms forum has been organized to create network independent APIs in order to enable cross network application development [1]. The APIs form the interface between the application layer or Service Network and the core network. Applications are logically positioned in the Service Network and can be deployed independent of the core network and access network that the end-user is using. This means that instead of the current approach where applications are tied to one specific network (PSTN, Mobile, IP) applications can be accessed and used from different types of networks or domains [3]. The end user domain and its interface to Parlay-based applications has however received little attention.

[1] The Centre is supported by Telkom SA Limited, Siemens Telecommunications, Sun Microsystems SA and the THRIP Programme of the Department of Trade and Industry

Along with network independent APIs, service architectures to support the rapid creation and delivery of cross network applications have been developed. One such is the TINA-C service architecture developed by the TINA-Consortium (TINA-C) [4]. The TINA-C architecture provides a set of concepts and principles to be applied in the specification, design, implementation, deployment, execution, and operation of software components in large scale distributed networks. The main goal of the TINA-C architecture is to ensure separation of service network and communication network concerns by hiding underlying transport network technologies from end-users, application developers and application service providers.

In the context of this research it is important to note that the TINA-C service architecture documentation provides a detailed description of the end user domain. This is done by describing a set of end user domain related components and their interactions with the service provider and retailer networks.

The convergence of the IT and telecoms infrastructures facilitated by Parlay and the TINA-C offers new opportunities to Telcos in terms of service delivery. Currently, the telco is able to provide national and international services by federating with other operators. The value-added services are primarily voice-centric, examples include freephone, voicemail, conference calling, operator assistance and customer calling. Simple data services such as fax and e-mail also operate over this network using circuit connections.[5]

Telcos will be able to deliver a range of services to end users through a single access network. For example, access to telephony, mobile telephony, and Internet may all be offered in a single service. This will imply the availability of more data-centric services such as video on demand (VoD) and multiparty video conferencing to end users.

Furthermore, new types of services may also be offered. One such is the *Context-driven service*, which allows the delivery of a service (service components, parameters and content) in differing ways depending on the context. For example, a service delivery context may be user preference (audio/video quality), terminal capability (display resolution, color depth, network card) or network capability (end to end QoS level) [6]. Another is the *Location-based service*, which allows the delivery of a service based on the geographical location of an end user. For example, a freephone service could be offered at a conference for anyone within the area where the conference is being held. The location based service could also be used to, for example, notify subscribers of a service when other subscribers are within the same geographical region [7].

In this project, we examine the design of a multi-service TINA-C based service platform for future integrated service provision scenarios using the Parlay gateway. We propose that in this implementation the user should be allowed to access cross-network services through a single customizable user interface. Using service components specified in the TINA-C service architecture the service platform should accommodate Next Generation network capabilities such as context driven and location based services.

This implementation must also allow the user greater involvement in service management, for example the user should be able to change service profiles within permitted limits, check service specific accounting information and to subscribe on line to new services.

To hide network heterogeneity, the OSA/Parlay APIs are proposed as a method of access. The system will be illustrated through a proof of concept case study.

## II. Background

### A. Introduction to the NGN

The NGN can be thought of as a packet-based network where the packet switching and transport network are logically and physically separated from the service/application network [8].

Most of the present NGN paradigms implement a softswitch architecture where the switching functionality is distributed across several nodes. All emerging NGN architectures share a common goal of rapid service creation and delivery. Softswitch architectures such as Open Services Access (OSA) or Parlay achieve this through the separation of service intelligence into service dependent and service independent logic across an open Application Programming Interface (API) [9]. The re-use of service-independent logic across the API by third party application service providers facilitates rapid service creation and delivery. TINA-C's service architecture, describes a set of feature-rich interfaces and operations useful in third party service provision. A subset of these interfaces will be implemented to provide the functionality required in this project.

In order to allow for independent service development and delivery, as well as for changes in the communications technology, it is necessary for the application/service network to be decoupled from the transport network [8].

This report documents the design and implementation of a hybrid TINA-C service architecture implementing an OSA-Parlay gateway in order to deliver multimedia multi-party telecommunications services to end users.

The main focus in this hybrid architecture will be the definition of end user domain components and interfaces for use in the access and usage of 3rd party multimedia services.

### B. The TINA-C Architecture

The TINA-C architecture is composed of four main sub-architectures, the service architecture, network resource architecture, computing architecture and management architecture. The service architecture is concerned primarily with problems of service management, user subscription management, charging and accounting for service usage. The network resource architecture is concerned with network connection setup for service requests in the service architecture. The software architecture ensures the development of interoperable and portable software components [5]. The management architecture provides concepts and principles for effectively managing services, networks and computing infrastructures.

Due to the nature of this research, this section emphasizes the TINA-C service architecture, which provides the basis for the realization of multimedia multi-party telecommunications services. The main goal of the TINA-C service architecture is to provide a set of concepts, principles, rules and guidelines for the construction, deployment and operation of NGN services. The TINA-C service architecture identifies components to be used in a Distributed Processing Environment (DPE) in order to build services as well as how these components interact and can be combined to support the instantiation, management, and use of NGN services [4].

The architecture accommodates NGN service requirements by hiding underlying transport network technologies from end-users, application developers and application service providers.

#### 1) The TINA Business Model

TINA also provides a set of business roles and responsibilities for each domain in the service architecture. [10] describes the scope of the service architecture with respect to the initial set of business roles and business relationships. Reference points are used to classify the relationships between the different business administrative domains. However only the relationship between the consumer and retailer business roles is defined by the TINA-C. Below is a brief summary of each of the defined business roles:

- The **Consumer** business role uses services provided in a TINA system. The broker and retailer business roles provide services for customers to use [4].
- The **Broker** business role provides stakeholders with information that enables them to find other stakeholders (business domains) and services in the TINA system.
- The **Retailer** business role provides stakeholders in the consumer role with access to services.
- The **Third party service provide**r aims to support retailers or other third party providers with services.
- The **Connectivity provider** manages the underlying transport network.

#### 2) Overview of Service Architecture Components

Due to the fact that this research requires the development of a service platform to support the delivery of multiparty multimedia telecommunications services, it is important that we examine the main generic service components defined by the TINA-C service architecture. It is possible to develop telecommunications services using these components or customized versions specific to a particular value added service. In this section, we take a look at the main service components that are proposed by the TINA-C Service Architecture [4].

*a) Access Related Components*

- **asUAP - Access Session User Application**

The User APplication (UAP) Service Component (SC) is defined to model the variety of applications and programs in the user domain.

- **PA - Provider Agent**

The Provider Agent (PA) is a service independent SC, defined as the user's end-point of an access session.

- **IA - Initial Agent**

An Initial Agent (IA) is a user and service independent SC that is the initial access point to a domain. The IA supports capabilities to authenticate the requesting domain and establishing access sessions.

- **User Agent (UA)**

A User Agent is a service independent SC that represents the user in the provider's domain and acts as a single contact point to control and manage (create/suspend/resume/delete) the life-cycle of service sessions and user service sessions. It is also used to manage the user's preferences (choices or constraints) on service access and service execution. This component will be used to support the Context Driven Service.

- **SUB - Subscription Management Component**

The Subscription Management Component (Sub) allows the management of subscribers, subscriptions and users for the whole set of services provided by a provider.

*b) Usage Related Components*

- **ssUAP - Service Session User Application**

The UAP is service specific and enables the user to make use of the capabilities of a service session (ie. suspend, resume, stop, ...a service session).

- **SF - Service Factory**

A Service Factory (SF) is a service-specific object, which manages the lifecycle of the service session COs for a service type. It creates the service session components for a service type and controls their life-cycle according to requests from the UAs. In this project, the SF will also have the responsibility of creating instances of 3rd party applications when requested by the user.

- **SSM - Service Session Manager**

An SSM supports service capabilities that are shared among users in a service session.

- **USM - User Service Session Manager**

The USM holds the context of a party, or resource in a service session.

The implementation of the USM and SSM are outside the scope of this research. Instead their functionality is delegated to the Application Logic and the Service Capability Features (SCF) in the Parlay Gateway.

It is important to note that only the most relevant service components to this research are mentioned. The list of components and supported capabilities presented here is by no means exhaustive. However, [10] presents each of the service components in greater detail.

### C. The Parlay Architecture

The OSA/Parlay architecture implements an application programming interface (API) that aids the rapid creation of telecommunications services. The OSA/Parlay APIs are technology independent and are designed to work with mobile, fixed and next-generation networks.

The API is composed of two major types of interfaces, Framework interfaces and Service interfaces which are grouped into Service Capability Features (SCFs) [5]. The current specification details 14 main SCFs:

- Call Control
  - Generic Call Control SCF
  - Multi-Party Call Control SCF
  - Multi-Media Call Control SCF
  - Conference Call Control SCF
- User Interaction
- Mobility
- Terminal capabilities
- Data session control
- Messaging
- Connectivity Management
- Account Management
- Charging
- Presence Availability Management
- Policy Management

These APIs expose much of the core functionality of the underlying telecommunications network to an application developer, enabling the creation of new applications that use a standardized set of APIs. In this project, the APIs will be used to support 3rd party services. However, the design and implementation of 3rd party services is outside the scope of this research and is left to the Parlay application designer.

It is important to note that the Parlay architecture pays little attention to the consumer (end user) domain. The Parlay specifications only detail the interaction between the 3rd party applications, the SCFs, and Enterprise Operators [11].

In the Parlay model, the enterprise operators act in the role of *subscriber/customer* of services (SCFs) and the 3rd party applications act in the role of *users or consumers* of *services*. The framework itself acts in the role of *retailer* of services.

However, the interfaces between an enterprise operator and the 3rd party applications in its domain are outside the scope of the Parlay API [11]. This gives the enterprise operators the capability to dynamically create, modify and delete the client applications and service contracts belonging to its domain.

Therefore, in the Parlay model, the SDP can be used by enterprise operators to perform the above functions as well as to deliver 3rd party applications to end users.

### D. The TINA based platform for Parlay service deployment and provisioning

Figure 3 puts together all the concepts that have been discussed in this section to present a model of the proposed hybrid TINA-C and Parlay service architecture. The first point of access to 3rd party multimedia services for an end user is the

user interface. The UI's functionality will be presented in section III.
As can be seen the TINA-C based SDP consists of TINA-C service architecture components which are used to access the 3rd party application server(s), as well as set up usage sessions. The asUAP, PA, and ssUAP are located in the consumer domain and are used to contact components in the service provider domain in order to setup access and usage sessions. The asUAP is the first point of contact when setting up the access session and the ssUAP is the first point of contact when setting up a service usage session. Note that access sessions only require that the service provider be contacted, however service usage sessions require the Parlay gateway to be contacted by 3rd party applications in order to perform the required call control setup and teardown.

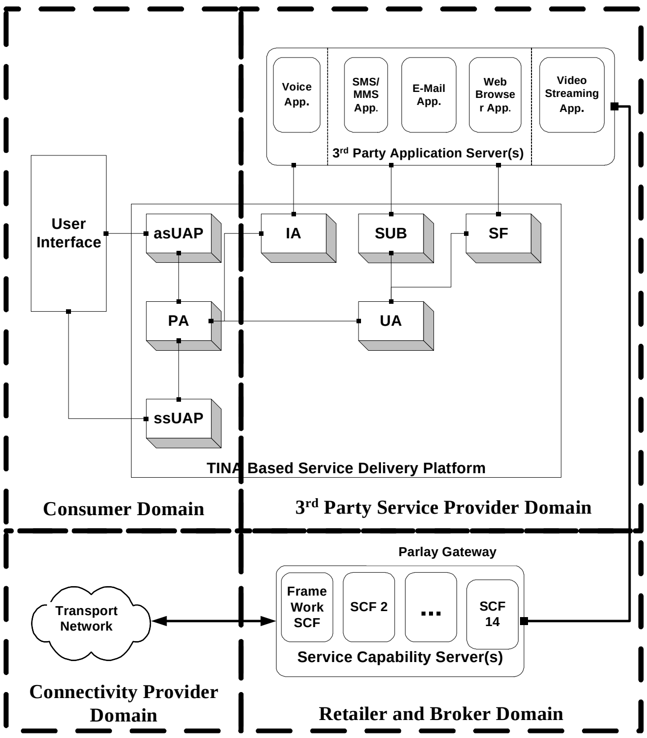


Figure 3. Overview of Proposed Service Architecture Component Interaction

The IA, UA, SUB, and SF are all located in the service provider domain. The IA is used to authenticate the user to a service provider before accessing and using 3rd party applications. The UA is mainly used for User and Service Context management, that is, to manage the user's preferences in reference to their own profile and their individual service profiles. The functionality of the SF, and the SUB is as mentioned in the section IIB.2. The SDP does not interact with the parlay gateway in any way as the functionality provided by the SCFs is not necessary for it to deliver 3rd party services to the end user.
3rd party applications use the Parlay gateway to implement their required functionality (ie. call control, mobility management, charging, …etc). However, applications first have to be registered with the gateway to use the SCFs.
The Parlay gateway acts as a retailer and a broker of services (SCFs) to applications in the service provider domain.

In figure 3, the IA, SUB, and SF interact with the application server(s) through some APIs, however it is important to emphasize that the APIs through which the SDP and application server interact are undefined. Since the application server offers services to end users in a similar way that the framework offers SCFs to applications, it is suggested that a similar approach be taken to standardize access to the application server. It is therefore suggested that a central point of access to the application server similar to the Parlay gateway's framework API be implemented. This will be discussed further in the following section.

## III. DESIGN

### A. Service Goals

Design of the user interface and service platform requires that the targeted services be identified. The proposed service should deliver a range of services through a single user interface. Basic existing services must be supported by the interface.

- Voice (PSTN)
- SMS and MMS (Mobile Network)
- Email (IP)
- Web Browsing (IP)
- Video Streaming (IP/DATA)

It is proposed that the following features be included in the service:

- User and Service Profile Management feature
- Service Discovery feature

  OSA-Parlay offers a service discovery interface through which a third party application can query which capabilities are available [8]. The end user should also be able to request a list of available 3rd party applications. This service should be accomodated for by the user interface and service delivery platform.
- Service Addition feature

  Once a user has discovered the availability of a service, they should be able to subscribe to it.

### B. UI and Service Delivery Platform Design Goals

#### 1) User Interface

The following guidelines will be used to design the User Interface:

- *Different front ends should be "adaptable" to the Service Delivery Platform.*

  This means that different front ends should be able to be "plugged into" the SDP. This is important for two reasons. First, the front end may undergo many changes during the lifetime of the project and these changes should not propagate to the SDP. Second, the ability to "adapt" the front end to a SDP may be important when evaluating the strengths and weaknesses of different UI designs.

- *UI should not be device dependent.*
  Due to the fact that the UI may be used across a range of terminal types (ie.pc, cellphone,…), it is important not to restrict its design to a particular terminal's specifications. For example, the UI should not be memory intensive as this may restrict its portability to mobile devices. However, it is important to note that portability is not a requirement.
- *UI should be predictable.*
  Users will not have any formal training in the use of the interface.

*2) Service Delivery Platform (SDP)*

As mentioned in section IID, no standardized set of Parlay interfaces have been defined to guide the interaction between the application server and end user applications seeking to access and use 3rd party applications. The addition of the SDP is meant to support these activities, however, an API to access the application server remains undefined. It was therefore suggested that a similar point of access to the application server as the Parlay gateway's framework API be implemented. In this section, we introduce the Service Provider Framework (SPF) and define its functionality as required by the SDP.

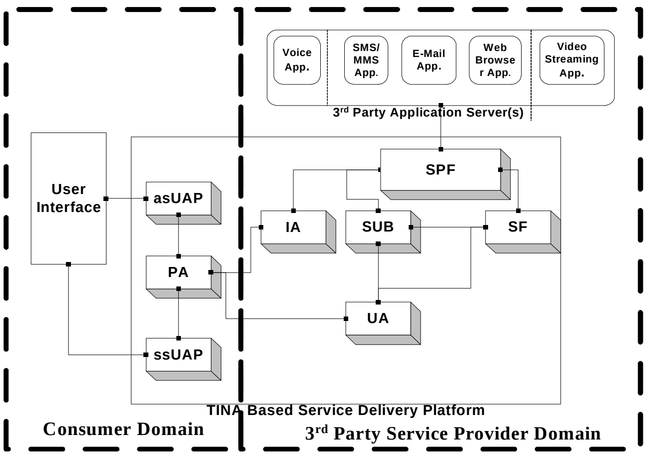


Figure 4. Introduction of the Service Provider Framework (SPF)

The Service Provider Framework is intended to support the access and usage of 3rd party application services by end users implementing the TINA-C based SDP. Its main functions are:

- authenticate end users to the service provider domain
- register new applications
- support 3rd party application discovery by end users
- support end user selection and subscription to 3rd party applications

Figure 4 illustrates how the SDF fits into the hybrid TINA-C Parlay service architecture.

The SDP will be designed to support the following access and usage related scenarios:

## Access related scenarios:

**1. Contact a provider**

*Description***:**

A user initiates the user interface on their end user device. Thereafter, (s)he attempts to contact a provider. On completion, the user receives a response from the provider.

*Minimum Required Components:*

asUAP, PA

**2. Login to a Provider**

*Description:*

A user logs into the provider domain by specifying an appropriate user name and password. On completion, the provider can correctly identify the user by a user name.

*Minimum Required Components:*

asUAP, PA, IA, UA, SDF

**3. Logout from a Provider**

*Description:*

A user logs out of the provider domain. All access and service sessions are terminated upon completion.

*Minimum Required Components:*

asUAP, PA, UA, SDF, Framework, Call Control SCF

**4. List Available Services**

*Description:*

A user requests to view a list of the available services. The subscription manager is contacted. On completion, the user can view a list of the available services.

*Minimum Required Components:*

asUAP, PA, UA, SUB, SDF

**5.Check Accounting Information**

*Description:*

The user requests to view their current billing status. The charging SCF is contacted. This SCF is used to pass billing information to the SUB which associates the information with a user's subscriber information. On completion, the user can view a list of billing records.

*Minimum Required Components:*

asUAP, PA, UA, SUB, Charging SCF

## Usage related scenarios:

**6. Start a Service Session**

*Description:*

A user starts a new service session. Depending on the type of service selected, a specific session model (ie. Basic call model, Multiparty call model, …etc) will also be selected. On completion, a service session is started and the user is ready to invite other users to join him/her in the session.

*Minimum Required Components:*

ssUAP, PA, UA, SF, SDF, Framework, Call Control SCF

**7. End a Service Session**

*Description:*

A user ends the service session that he/she created. On completion all participants of the service session are removed and the user is ready to start a new service session.

*Required Components:*
ssUAP, PA, UA, SF, SDF, Framework, Call Control SCF

**8. Invite a User to Join a Session**
*Description:*
A student participating in a service session invites another student participating in the same service session to direct communication. On completion, the invited user is left to decide whether to accept the invitation or not.
*Required Components:*
ssUAP, UA

**9. Join a Service Session with invitation**
*Description:*
A user participating in a service session accepts the invitation from another user participating in the same service session to engage in direct communication with him/her. The inviting user is informed of the decision of the invited user. On completion, the two students are ready to engage in direct communication with each other.
*Required Components:*
asUAP, PA, UA, Call Control SCF, SF, ssUAP

## IV. Conclusion

With the advent of multi-service transport networks, traditional telecoms and IT infrastructures are converging. This offers new opportunities for multi-network multimedia service provision. This convergence has led to the emergence of the NGN. Furthermore, several architectures have also emerged to support the extended capabilities of services offered in the NGN. Amongst these are the OSA-Parlay, and TINA-C architectures.

This report presented the design of a TINA-C based multi-party multimedia service delivery platform utilizing the OSA-Parlay gateway to present services to end users. Furthermore, the design of a user interface for this service was outlined.

## References


[1] The Parlay Group, Homepage at http://www.parlay.org.W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.
[2] 3GPP Technical Specification 22.127 v5.0.0, Update: Now at ver 6.6.0"Service Requirement for the Open Service Access (Release 5)", June 2001.
[3] Report No. 16, "3G Portal Study -- A Reference Handbook for Portal Operators, Developers and the Mobile Industry", UMTS Forum, November 2001.
[4] TINA-C , "*Service Architecture*," Deliverable version 5.0, June 1997.
[5] P. Nana, On a hybrid TINA-Parlay service architectecture for next generation networks, MSc(Eng) Project Report, University of the Witwatersrand, Johannesburg, 2003.
[6] A.Liotta, A. Yew, C. Bohoris, G. Pavlou, "Delivering service adaptation with 3G technology", Center for Communication Systems Research, Univ. of Surrey, http://www.ee.surrey.ac.uk/CCSR/Networks/
[7] M.Le Clerc, "Benefits and Business opportunities of building wireless applications using Parlay/OSA for developers", Ericsson, 2003
[8] "Next Generation Network Intelligence", http://www.mobilein.com/NGN-1.htm.
[9] R.K. Prasad, QoS provisioning to a SOHO telecommunications client using a DPE-enabled gateway, MSc(Eng) Project Report, University of the Witwatersrand, Johannesburg, 2004.
[10] TINA-C, "*Service Component Specification*," Deliverable 1.0 b, January 1998.
[11] " OSA API Part 3: Framework", August 2003, http://www.parlay.org/specs/archive.asp