A Proposed Structure for Application Server in NGN

H.Yeganeh, A.H.Darvishan, M.Dindoost, H.Sabaei
IRAN Telecommunication Research center
{Yeganeh, Darvish, Dindoost, Hsabaei}@itrc.ac.ir

Abstract

In this paper a structure for providing and offering service in NGN network is proposed. This structure, which is named Application Server, is a specified unit and it is not a tool. Actually, an Application Server is a set of different parameters providing connection between various parts, which are dealing to network problems, such as third parties, developers and etc. This structure is designed modular and it has many advantages in NGN network. Service providing is the most important task which this structure has. This structure decrease the expense used for adding new services. Also in this structure adding new services is independent of deeper layer protocols. The implementation method for this unit will be proposed in this paper. In this method, adding new service to Application Server is easy and it doesn't need to any particular technical knowledge.

Key words
NGN (Next Generation Network), Application Server, Service Provider, Third-party, Application Developer, Service Delivery, Service Creation

1. Introduction

Next Generation Network (NGN) is an expression, explains the development path in networks architecture which will prevail over the other architectures in next few years [1]. Thus, using this kind of networks will be an essential need in the coming future. Next Generation Networks for attracting new customers, keeping old customers and providing their telecommunication needs must present new advanced services. Because of quick development in NGN network, network operators are looking forward to a new environment for providing NGN services with more efficiency and less expense and intricacy.

Providing new services in nowadays telecommunication networks, which are based on special hardware and software, is a time consuming and expensive process needs technical knowledge about network structures. Each network has its own infrastructure and this limitation does not allow the network operators to provide new advanced services. For example mobile network is allowed to use only its own infrastructure. Generally operators need to high flexibility services with having high capability for performing in a proper way, re implementing standards, developing of operational program for unification and separation of services

Traditional method used in service provisioning, are being changed with new structures and methods in service delivery such as IMS (IP Multimedia Subsystem). The infrastructure used in NGN networks conceals the network intricate from Third-party's, developer's and Application's point of views, so they are involved with their own challenges. Some organizations such as ITU are studding on NGN standards in order to create an environment which allow operators to provide application programs to the end users. Also the operators are able to create high transparency between fixed networks and cellular network for transferring voice, data and video. Standardizing allows service providers to migrate to packet-based structure, also the vendors will being able to present interoperability equipment. Generally, an application server must have these abilities for providing and presenting their services:

- Quick implementation, development and management of the value added services
- Provide services which are independent of network elements and specifications
- Provide connections among different network elements
- Provide flat access to capabilities for applications
- Have open access to service abilities for service providers and Enterprise applications

According to above specifications, it is necessary to have an individual element in NGN to create and deliver services. This element is named Application Server.
2. Application server

Application Server is a unit for providing services in NGN network and it must be implemented independently of the other elements. The position of this unit in NGN is shown in fig 1.

In this paper the structure of an Application Server and the way in which an Application Server can connect to different customers and the way for adding new services with minimum expense are introduced.

The solutions for manufacturing this kind of Application Server must be based on OSA and use API standard. The use of API standard allows Applications to be used in different infrastructures, so the integrating applications, defining new services by third parties and developing applications by developers will be done easily. This solution is not sufficient by itself and it can not provide enough assurance and high efficiency for providing services especially real time services.

The Application Server can be made of XML (Extensible Mark-up Language) or J2SE (Java 2 Standard Edition) Scripts or can be created in a standard environment such as JAIN SLEE (Java API for Integrated Networks Service Logic Execution). Using SLEE reveals the closed and proprietary solutions and by this method the Applications can be developed rapidly. Without standardizing, each SLEE is an individual unit which doesn't have any transactions with other strategies. Some advantages of standardizing of SLEE are: Service Portability, accelerating of development process with using redefined parameters, moving toward 3G and integrating new services with using traditional network equipment.

3. Services to be delivered by the application server

NGN must offer the same services, which are presented in existing networks, at least with the same quality for having popularity. These services, which are named value-added services, are implemented by Intelligent Networks in traditional networks, but in NGN networks it is better to implement these services applying SIP 0 using Application Servers in an open architecture, distributed and loosely-coupled with high scalability. In fact, IN network uses a group of client-services and other network resources to control call setup and provide media services such as Announcements and voice mail. So in the similar way, NGN networks need to proper client-services for offering these kind of services. In IN networks, these services are defined according to Q.1210 [6] and Q.1221 0 standards and some implementation strategies are defined in SIP 0. Applications requirements are changing dramatically and delivering IN services cannot cover that. In actuality, NGN services are mixture of IN services and web services.

4. Designing the application server

The first step in designing is plan top-down modulating; it means the services are distributed into several arrays according to specific parameters, which are explained in part 5, so for each array there is a particular module. Designing without modulating is not appropriate for vendors and service providers, because this designing is faced with many problems when new services are added or when the system is upgraded. In fact vendors and service providers are looking forward to unbundled and modular system for using the advantages of different products of different companies. For example each manufacture designs and implements one or several modules then the service providers can purchase the best module with more capabilities. Also the vendors can apply the each other's indexes in their proposed strategies.

After modular designing, different APIs are being considered, so these APIs can be used by third parties to add new services and by developers to develop applications.

Next step in designing, is applying SIP Servlet API, which are open and based on standard, with using SIP Servlet API the developing capability of Application Server is guaranteed.

If a service provider is able to use only its APIs, the third parties and developers have to know all APIs from different vendors for adding new services and developing the applications or the vendors must do these tasks by themselves, which is possible. For this reason these plans can not provide integration between telecommunication services and web services. The first proposed plan for Application Server is shown in fig 2.

5. SIP-based design

One of the reasons that SIP protocol is used in designing is that the resources of the network often use SIP and the application server should also use SIP in order to use these resources. This is not accidental and the many capabilities and advantages of SIP have caused its being current. The first characteristic of SIP protocol is that it's not a Master/Slave protocol, it's Peer-to-Peer. Because a Master/Slave protocol has controlling essence and the heavy load of controlling traffic causes complexity in Call Flows. Another alternative that can be considered for SIP in implementation of value-added services is IN. As mentioned before, this implementation is used in PSTN networks. The main problems that can be mentioned for this implementation are the very low speed, high
cost of design and implementation, technical problems and the very low flexibility. The main cause is that in implementation with IN if a service should be used by various service providers, this service should be designed and implemented separately for each service provider [10]. Also IN is specially for providing telecommunications services, not web services.

Another reason in using SIP is that by the essential flexibility of SIP, making different interfaces between SIP networks and other environments of service delivery is possible by which the capabilities and advantages of those environments can be added to the SIP network.

6. The proposed architecture

The proposed structure has an open distributed architecture like internet and is not dependant to APIs or any special operating systems because it only operates based on the SIP messages and HTTP. This open architecture is a good choice for service providers too. The proposed application server has some components. The goal of these components is to make the connection between Application Server with different sections that somehow have relation with the service issue such as developers, Third-parties, service providers and etc. Therefore, the service providers can provide a basic set of their services, developers can improve the applications by standard APIs and Third parties can add the services they like to the set of application server by standard APIs and also use the implemented services and present them to their users too. By this means, a lot of effort and knowledge needed for making new applications and implementing new services is saved because all the parts that have interaction with Application Server directly can use the applications and services that have been implemented by the others and there is no need to implement them again. It is suggested that these building components be in a SDK (Software Development Kit) form with the ability of working with J2EE’s IDE (Integrated Development Environment) [10].

As mentioned in section 4, a specific criterion should be used for designing the Application Server. This specific criterion is considered in services that can be provided and based on it, the building components will be defined; so a building component includes one or more servers. Each server can provide one or more various services. Also one service can be provided by several servers. These servers can be in one building component or be in various components. The considered Application Server can be like figure (3) based on the criterion that has been mentioned.

The building component that has been shown is figure is:

- **AAA Server (Authentication, Authorization, Accounting)** Doing AAA operations and etc.
- **Controller** to make synchronization between different servers
- **Portal Server** that provides IVR (Interactive Voice Response) service and also can be used as a Voice Portal
- **Web Server** for making connection between users and network for requesting service, receiving answer and things like that
- **Presence Server** for recognizing Status of the user including being Online, Offline and etc.
- **User Servers** that user interact with and provides services needed by the user like Text-to-Speech Server, Voice Recognition Server, Universal Messaging Server and etc.
- **Portfolio Servers** that collect portfolios and provide them to the user. As a sample Servers related to Application Service Provider, Credit Card Verification and Transaction can be named. For example software like Office can be presented to the user using this kind of servers.
- **Session Control**: Provides the ability of establishing, disconnecting and changing the existing connection sessions for all kind of media.
- **Call processing language (CPL)**: Provides the ability of saving and running CPL that help the developers to develop user services such as Call Forwarding, Find-me/Follow-me and etc.
- **Call Forwarding**: Provides a set Call Forwarding attributes for developers that want to have a higher level of abstraction of CPL for call center.
- **Application Event Logging**: Collecting Application-context-sensitive information for Accounting, Fault and/or Performance management
- **Presence Management**: Enables applications to subscribe for the notification considering the other status of registered SIP entities.
- **Instant Messaging**: Provides the ability of managing Instant Messages based on SIMPLE standard (SIP for Instant Messaging and Presence Leveraging Extensions) [11].
- **Conferencing**: Enables Conference Servers to connect to Application server in order to hand over a complete controlling ability of conferences to Application Developers.
- **IVR Control**: Enables developers to reach the IVR abilities inside the applications.
- **SIP Registration**: Provides SIP Registrar capabilities with the related information of user’s location for devices and SIP applications.

The servers shown in figure are connected in a loosely-coupled way to each other to make more flexibility. When a function from a component's
function is called by a call flow, the details of the operation are given over to the component itself so that this function has no effect on other components.

This structure can integrate NGN services by keeping the dependence of the operation of each building components and also enables the developers to add new characteristics of SIP by standard mechanisms like JAN SIP Servlet API, so that by increasing characteristic of SIP, the capabilities of Application Server can be increased.

7. Advantages of the design

In contrast, with the design and implementation of IN in which the programmer and creator of the service should have detailed technical information of all of the network details, by using Application Servers based on SIP, the programmers can create their services in a short time with a low cost without having knowledge about SIP and the structure of deeper layers in telecommunications network.

Also, all sections can share their defined and designed services and applications so that others can use them and there would be no need to re-design the applications and services.

Modularity is another characteristic of this architecture that helps the service provider to benefit from the advantages of other developing companies' and third parties' products in the best possible way.

8. Conclusion

In this article, a structure for creating and delivering services in next generation networks as one of the key goals of these networks has been introduced. In this article, a unique model has been considered as Application Server in which every section do its task independent from others. A group develops applications, other create new services and provides existing services to the users and etc. This model has lots of advantages including it's modularity consisting of several components which decreases cost of adding new services and developing applications by making the possibility of reusability ability of defined applications and existing services. Also, for implementing the platform a method has been proposed in which adding new services is easy and needs no technical knowledge. Future efforts in this field can be introducing new building blocks and considering requirements of different groups as third-parties, developers, IMS customers, etc in the proposed structure.

9. References

[9] 3GPP TS22.228, Service requirements for the Internet Protocol (IP) multimedia core network subsystem (IMS); Stage 1 2-3GPP TS23.228, IP Multimedia Subsystem (IMS); Stage 2
Figure 1. The Application Server's position in NGN networks [2]

Figure 2. A basic plan of Application Server [10]

Figure 3. The basic sample of the proposed plan for Application Server