

## **An Adapter based Network Management Layer Framework for provisioning Telecom Services**

**Anup D, Shantharam Nayak, Annamma Abraham**

<sup>1</sup>4<sup>th</sup> sem M.Tech(SE), Dept of ISE, R V College of Engg, Bangalore- 560059, Karnataka, India.

<sup>2</sup>Professor, Dept of ISE, R V College of Engg, Bangalore- 560059, Karnataka, India.

<sup>3</sup>Professor, BMS Institute of Technology, Bangalore – 560064, Karnataka, India.

---

**Abstract:** - The provisioning of telecom services requires the subscriber's and service related information to be provisioned onto the network elements. The network elements or element management systems store the information received and is used for activation of the services for the subscribers. These telecom services could be wire-line or wireless based services. All the telecom service providers are rapidly shifting to Next Generation Networks (NGN) which are IP based packet networks. NGN's are independent of the underlying transport technology, hence it does not matter whether the underlying hardware technology is wire-line or wireless based. This enables the service providers to focus on defining new telecom services which could be delivered to customers. Thus, there arises competition between service providers in creation and provision of services to customers as quickly as possible. Also, quick customization of services that are already provided to the customers based on their needs is a major requirement. Hence, the challenge of delivering services by provisioning them quickly on the network is to be addressed as the users scale in numbers. As a part of addressing this challenge, there is a need for effective communication with the network elements for provisioning. A framework that can integrate with multiple Service Order systems, transform their requests into network commands and techniques used to communicate efficiently is mentioned. The commands are generated suitably for the network element and communicated to it based on SOAP protocol. It can work with network elements that expose an interface using Web Service Description Language (WSDL).

**Keywords:** - Service Provisioning, SOAP request, SOAP response, Mapping Engine, Adapters.

---

### **I. INTRODUCTION**

Network operators are nowadays upgrading their access networks to cope with the increasing number of users and the increasing bandwidth required by services [1]. In the management of networks, it can be seen that each network element is associated with a well-defined service. The element uses the information associated with it while offering the service to the customer. It applies rules based on the information it possesses and decides whether to offer the service to the customer or not. Hence, the presence of information is crucial. Without the information, the network element won't be able to provide service. Therefore, even though the customer opts for the need of a service, the actual service can't be experienced by the customer until and unless it is provisioned onto the responsible network elements.

The networks are being moved quickly towards next generation networks built on horizontal "planes" moving away from traditional vertical structure for fixed voice and data communication networks [2]. Because of the greater capacity that these networks offer, the service provisioning systems need to handle thousands of transactions per second every day. This demands for a fast, scalable provisioning system which can cope with it. The numerous requests received need to be parsed and transformed into commands suitable for the network element. A framework which acts as a mediator between the upper stream systems (that contain the requests to be provisioned) and the network elements is proposed. This framework consists of modules that are responsible for efficient provisioning. The framework is intended to be independent of the upper stream systems. Hence, it is can be integrated with any system that can adhere to the interface the framework exposes. The modules comprise of adapters which can talk to network elements and respond back quickly. An adapter which is configurable and acts as a plug-in receives requests from provisioning system, transforms them into commands and communicates with network element. The adapter is developed in Java.

The use of Webservices has been accepted in the telecom domain and the usage offered by it continues to grow. The IP Multimedia Subsystem (IMS) and Web Services have been developed on the basis of different Service Delivery Platforms (SDP), opportunities for new services have emerged from converged access to IMS and Webservices [3]. Hence, the method of talking with network elements to support new services is very much possible with Webservices. Thus, the accepted Webservice protocol for communication which is SOAP is used in this proposal.

## II. SYSTEM ARCHITECTURE

The framework as shown in figure 1 consists of components for sending provisioning requests to element management systems and Network Elements. It comprises interfaces that can be plugged in, a mapping engine to invoke the adapters, and configuration driven adapters that actually interface with the elements. The Framework API allows direct invocation by upper stream systems.

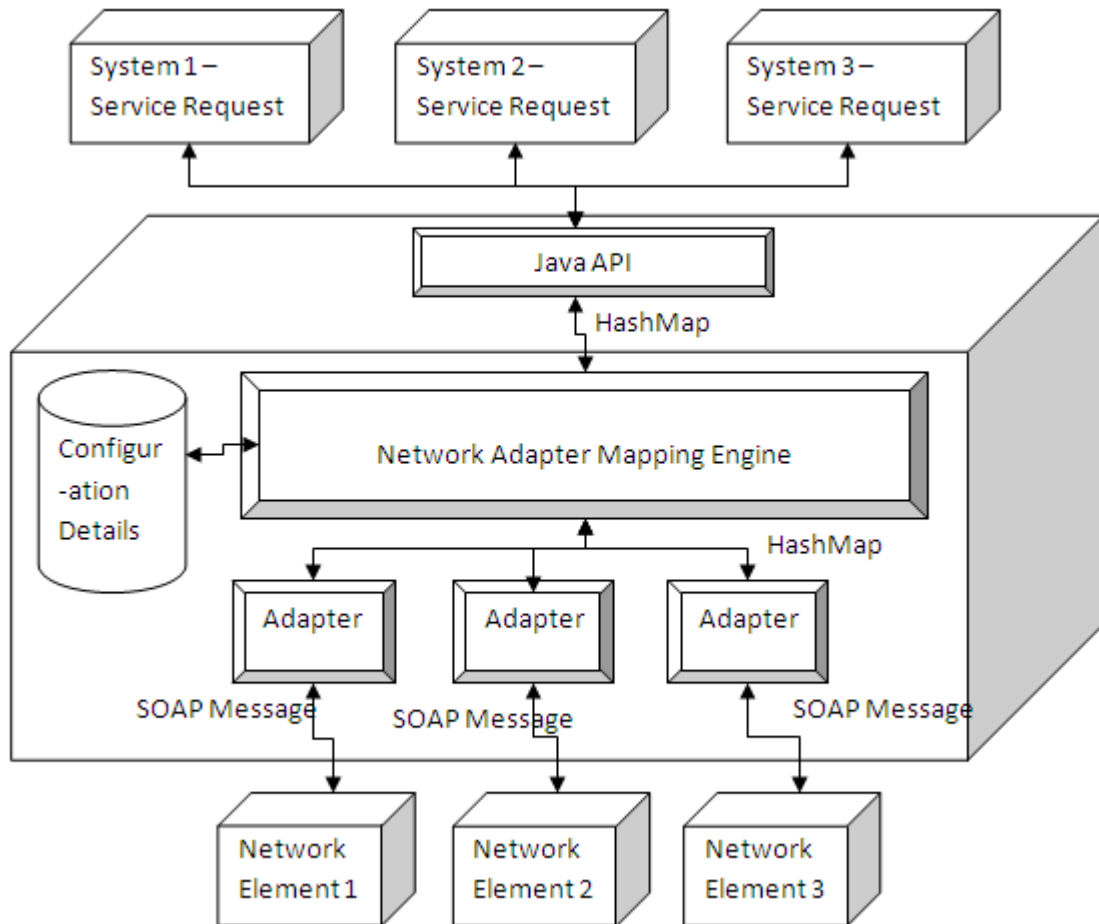


Figure 1: Architecture of the Framework

- **Java API**

The users of the system are the upper stream systems that send the information for provisioning. They talk to the framework via an API provided. So, it should be aware of the services provided by the API. They should send the message in a format the framework expects. In the message, the information would be present such as onto which network element the message is to be provisioned. The API component being an integrator of the framework with the rest of the upper stream systems converts incoming message into an internal java HashMap representation and calls the Mapping engine.

- **Mapping Engine**

The Mapping Engine is the main component that decodes the incoming HashMap structure based on defined configuration and calls the appropriate network element adapter implementation. All the Engine operations are controlled by a static configuration that needs to be pre-defined, before any provisioning requests are initiated for any of the supported network elements. The definition of new configuration rules and changes are automatically done during deployment of the framework.

- **Adapters**

The adapter implementation classes contain the necessary business logic to interface to the specific network element or the network elements' Element Management System (EMS). Any new network element support would require creating the adapter classes to use necessary business logic to convert the incoming HashMap (on the northbound side) to a structure that is understood by the southbound EMS. The adapters send

back the details/status of the response to the Engine also as a java HashMap. All the requests/responses to/from the adapter implementation are synchronous.

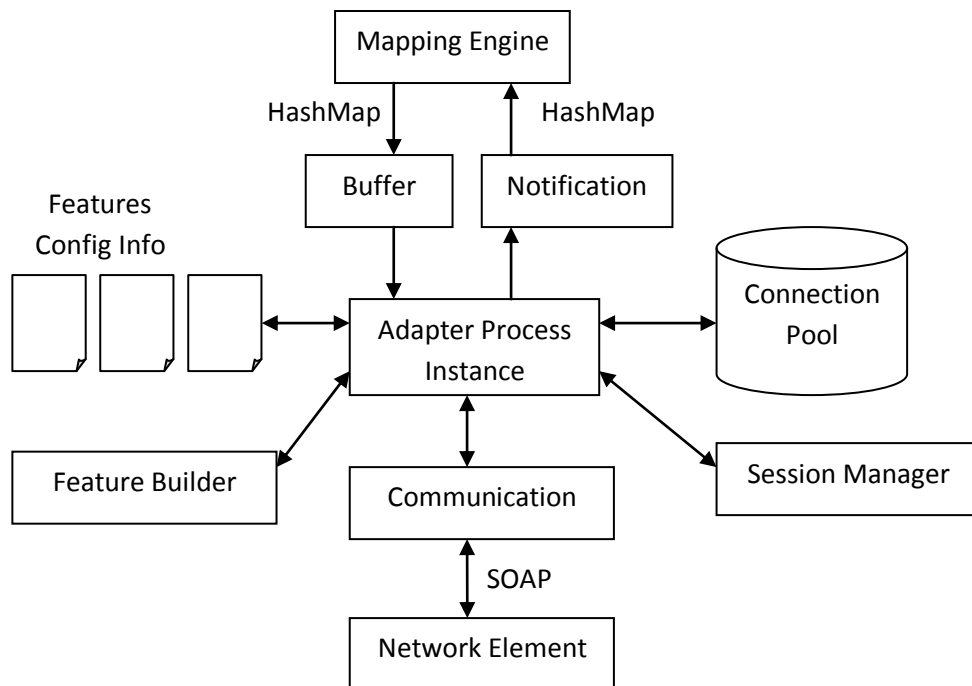
The adapters communicate with the Knees or EMS based on Web Service technology. Web Service is a software component invoked over the Web via an XML message that follows the Simple object Access Protocol (SOAP), which is a simple XML based protocol to let applications exchange information over HTTP and to transport the messages using open protocols standard. Web Services are based on distributed technology and provide standard means of interoperating between different software applications across and within organizational boundaries with the use of XML. Web services are provided with different Quality of Service (QOS) [4]. Hence, this allows for efficient communication.

### III. SYSTEM WORKFLOW

- The framework is responsible for communicating with both northbound upper stream systems as well as South bound EMS or network elements.
- The API layer receives the requests, processes them and transforms information into a HashMap and passes it to Mapping engine.
- The incoming requests to Mapping engine are accepted by a pool of threads. These are maintained upfront to enable efficient capturing of the requests. Then, these requests are formatted if necessary and kept ready to be accepted by the main process of the engine.
- The Mapping engine picks up the requests from the thread pool, decodes the information in the HashMap and looks up a configuration file which contains information on the adapter to be selected. It then tries to create an instance of adapter if it does not exist and passes the request to it.
- The adapter in this case is designed to send commands based on SOAP protocol. It performs a lot of pre-processing activity and transformation tasks on the message and finally communicates it to the network element.
- The response received from the network element is again a SOAP response which is parsed by the adapter and sends it to the mapping engine which communicates it to upper systems via API component.

### IV. ADAPTER DESIGN

The adapter as shown in figure 2 is designed to support all the features supported by the network element that can be provisioned. There will be numerous features which have to be supported. Following are the techniques used in the design of the adapter.



**Figure 2: Adapter Design**

One of the objectives of adapter is to fasten the transaction taking place. So, to reduce the time taken to send the request, connections are created in advance and used. The stubs are used to connect to the network element. These stubs are pooled well in advance and used when needed. Session management is crucial for reducing the transaction time on the network element servers. Hence, a session is created and maintained. The session is created when the first request is sent to the network element. This session info is stored locally and used for the rest of the transactions. Session recovery and clean up are also done to neatly utilize the resources.

The information about the features that need to be provisioned is stored in files. Hence, each feature is configurable in the way it is to be provisioned. Each request is tracked to make sure it gets executed properly. For this, timeout management, retry of requests are provided. The response received is also parsed to check for the status of the transaction. If the result of the transaction is in expected form, then a successful message is sent back. Otherwise, the result is parsed to check for the errors that would have occurred and a meaningful error response is constructed and informed. Command needed for the network element is constructed dynamically in the adapter. This means that the feature for which provisioning is being done is created based on CRUD operations. That is whether the request is to create, retrieve, update or delete mode of operation.

## V. CONCLUSION

Each network element in wired or wireless environment is used for a specific purpose. The framework tries to support provisioning of features of such network elements. It addresses many challenges during provisioning like fast and dynamic provisioning. To achieve these, it uses techniques like efficient connection management, session management and dynamically message construction of the features to be provisioned. The framework can integrate with any upper stream system that can adhere to the interface exposed by it. Likewise, it can communicate with network elements with few changes at its end. The expected provisioning has been achieved in this system.

## REFERENCES

- [1] Machuca, C.M.; Krauss, S.; Casier, K., Fault Management and service provisioning process model of next generation access networks, 7<sup>th</sup> International Conference on Network and Service Mgmt, 2011.
- [2] Sur, A.; Skidmore, D.; Chakravarthy, S., Webservices based SOA for Next Generation Telecom Networks, IEEE International Conference on Computing, 2006, Page(s): 520.
- [3] Islam, S.; Gregoire, J-C. Convergence of IMS and Web Services: A Review and a Novel Thin Client Based Architecture. Eighth Annual Communication Networks and Services Research Conference, 2010, Page(s) 221-228.
- [4] Dhore, S.R.; Army Inst. Of Technology, Pune, India; Gangwar, H.; Mishra, P.; Sharma, R; Systematic approach for composing Web Services using XML. Third International Conference on Computing Communication & Networking Technologies (ICCCNT), 2012