On the Service Management Framework for Service Delivery Paltform on Top of IP Multimedia Subsystem

Muhammad Shoaib Siddiqui[*], Choong Seon Hong[*†], Won-Kyu Hong[¤], Sung Bong Moon[¤]

Department of Computer Engineering, Kyung Hee University[*]

Network Technology Lab, KT[¤]

{siddiqui, cshong}@khu.ac.kr[*], {wkhong, sbmoon}@kt.co.kr[¤]

Abstract

In this paper we identify the needs and critical issues of management and monitoring in IP Multimedia Subsystem (IMS) architecture for providing multimedia services in next-generation networks using Service Delivery Platform (SDP). IMS is standardized by third generation partnership project (3GPP) and 3GPP2 as next generation convergence network. As an IMS provides a layered architecture converging heterogeneous networks; it requires network management as well as service management and session management. We provide a framework for managing an IMS system by utilizing network management server, managed object (MOs) and service managed objects (SMOs). SMOs are similar to managed objects (MO) except they are used for managing the provided services, error detection and fault recovery in services. These objects are also used to manage media sessions within the system. We also provide a case study for the management using a scenario of a triple play service.[†]

1. Introduction

IP multimedia subsystem (IMS) is the future for all-IP next-generation converged networks with potential of enabling service providers to create and provide value added services to users on heterogeneous networks. IMS was defined by 3GPP [1] as an standard architecture which provides a horizontal, cross-functional layer of intelligence on top of IP, enabling the creation, control and execution of new and rich user-to-user services (video streaming), user-to-server offerings (IPTV) and multi-user media services (game-playing on the move and at home via PC).

To enable this, IMS architecture must be made compatible with existing service delivery environment such as Service Delivery Platform (SDP) [2]. A service delivery platform helps to standardize all the service interfaces for a provider, creating a horizontal platform from which they can provision, control and bill for all the value-added services they provide, whether the services are created by third-party application developers or by the service providers themselves. By ensuring a consistent, highly automated and reusable service environment, a service delivery platform can dramatically accelerate a positive return on investment. A typical SDP+IMS solution is depicted in Figure 1.

As IMS is described to be the solution for the future services delivery, it must provide a cost effective solution to the companies. Therefore, the services provide by the IMS environment should be delivered according to Service Level Agreements and ensure Quality of Service. Furthermore, IMS nodes should provide best performance and error free environment to ensure the best delivery of the services. This gives rise to the issue of managing the IMS environment so that it provides cost effective solution and provide increasing revenue. Management of IMS system to provide the Service Delivery platform should be designed to provide non-disruptive services with maximum quality [3]. Performance of the network nodes should be monitored to ensure there is no fault in the system. Error and faults in the system should be identified and reported and efficient error recovery mechanism should be applied to ensure the services are delivered up-to the demand of the customers.

In this article, we have provided a study of Management and Control for IP multimedia Subsystem Architecture for providing Service Delivery Platform. In section 2, we discuss some related work to management of

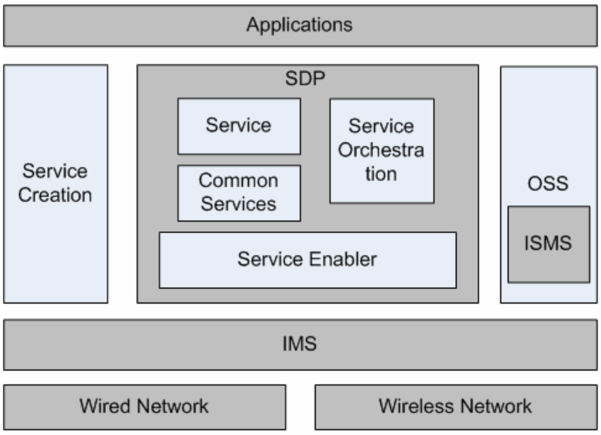

Figure 1. SDP on top of IMS architecture.

[†] Prof. CS Hong is the corresponding author.

IMS and SDP. In section 3, we discuss the management of IMS and SDP. Section 4 discusses a new concept for the management of IMS and SDP, which we have named Service Managed Objects (SMO). Section 5 gives a case study of a Triple Play service provided by IMS and SDP and discusses how this scenario can be managed according to our discussed management solution. Finally, we conclude our work.

2. Related Work

Standards organizations such as ETSI/TISPAN, 3GPP, 3GPP2, OMA, Parlay Group, Java Community Process (JCP) and Internet Engineering Task Force (IETF) are all cooperating to deliver mature open, industry-standard architectural specifications, which spell out what protocols and APIs are needed by CSPs and application developers to facilitate the deployment of IMS and SDP systems. Many companies have introduced their solution of SDP provisioning through IMS, which include IBM, Ericsson, BEA, HP, and Motorola etc.

BEA WebLogic Communications Platform consists of the IMS SIP application server, BEA WebLogic SIP Server, and the powerful policy enforcement, 3rd party partner management, and Telecom Web Services platform™, BEA WebLogic Network Gatekeeper™. By combining a high performance SIP application server with a platform for policy enforcement, partner management and Telecom Web Services, BEA WebLogic Communications Platform offers CSPs a unique IMS service creation, delivery and control platform.

The IP Multimedia Subsystem solution from IBM defines to monitor and manage composite service execution to maintain service-level agreements and user satisfaction and scale and adapt infrastructure quickly and efficiently using a modular and open standards based blade server platform.

Ericsson delivers IMS end-to-end management from the core, including the CSCF and HSS, to the application server. Ericsson Mobile Platforms includes IMS Client architecture in its platform releases, which are licensed openly to the industry. This makes the Ericsson IMS offering a true end-to-end offering right through to the user.

The Fraunhofer FOKUS Institute opened in July 2004, the "Open IMS Playground." The FOKUS Open IMS Playground [4] is an open test environment, all major IMS components (especially the FOKUS Open IMS Core). The components come from our own development and from leading industry partners. Open IMS has the IMS Management architecture, which provides means for monitoring and controlling all vital IMS core network components. Open IMS use Active and passive traffic generation to control and monitor the IMS environment.

All these IMS solutions use conventional traffic analysis and performance parameters to manage the IMS and SDP environment. We believe that more emphasis should be given to the management and control issues of IMS and SDP to provide fault free services with good performance

3. Management in SDP/IMS

Management functions in IMS/SDP can be separated for:

- IMS (Supervision & Control of Devices, processes and traffic of Control plane such as CSCF, HSS)
- Components of Service Delivery Platform (such as Applications, Third party Servers and Service Enablers)

Management Server (ISMS) is a fundamental part of OSS as shown in figure 1. It supervises the vital states of IMS/SDP components. IMS/SDP process and traffic are constantly monitored, both actively and passively. The captured traffic is collected, correlated and analyzed and performance information of session control layer and quality of delivered services is diagnosed. Management server also gathers different performance, security, fault and configuration parameters from all over the network to maintain a healthy state of the network. For this it uses Agents at each managed nodes to deliver it the required information.

3.1. Management Architecture

The main conceptual architecture of managing the system is given in figure 2. Each managed node or IMS/SDP component is a Network Element (NE). These network elements are managed by different Element Managers (EM). One Element Manager may manage more than one Network Element. The management server manages all the Element Managers in the system. Each element manager is running one or more agents on itself which gather the information about this NE and deliver it to Element Managers and/or Network Manager.

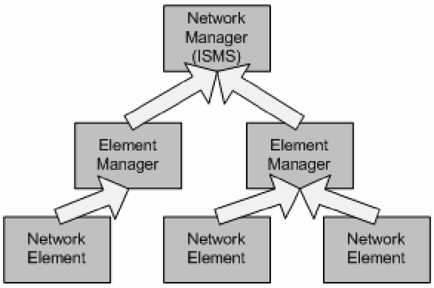

Figure 2. Conceptual architecture for Network Management

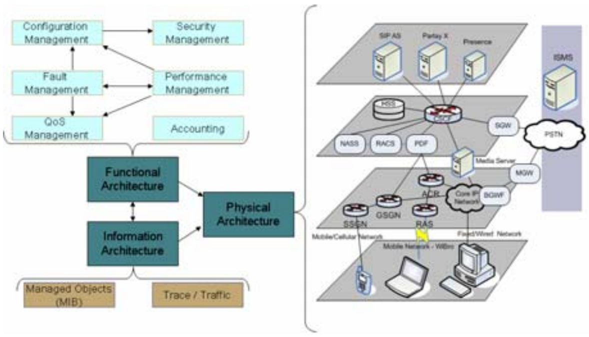

Figure 3. Management architecture for SDP/IMS

Figure 3 gives the architecture of the Management of IMS/SDP. There are three architectural blocks defined in the system:

- Functional Architecture
- Information Architecture and
- Physical Architecture

3.2. Physical Architecture

The physical architecture is consisted of IMS and SDP nodes. It is the actual network of the nodes which are managed by the management server. These nodes are called the managed nodes. The Information architecture defines how the managed nodes are managed by the centralized management Server.

3.3. Information Architecture

This architecture shows the types of information exist in the system to maintain the overall state of the system. This information is used to access and assess the network. There are basically two components of the information architecture.

- Managed Objects
- Trace / Traffic within the system

3.3.1. SNMP Agents

An agent is a network-management software module that resides in a managed device. This agent has local knowledge of management information and translates that information into a form compatible with SNMP. If Message calls are used to update the state of nodes (such application Servers) an XML to SNMP gateway is required for communication. Some capabilities of the agent are:

- Gathering information from managed objects
- Configuring parameters of the managed objects
- Responding to managers' requests
- Generating alarms or traps

3.3.2. Managed Objects

In a network, a Managed Object is an abstract representation of network resources that are managed. A managed object may represent a physical entity, a network service, or an abstraction of a resource that exists independently of its use in management. In IMS/SDP each component that can exist on its own can be represented as a Managed Object. Similarly, a single entity can be consisted of many managed objects. Managed Objects provide information about the managed node entities so that they can be efficiently managed by the centralized management server.

MIB (Management Information Base) are sent to and/or gathered from the managed nodes to manage the overall system. The information is periodically updated, and provided on demand to the Centralized Management Server. Whenever there is a problem, these MIB are sent to the Network manager. These MIBs define the following properties of the node:

- Configuration parameters
- Fault Detection
- Accounting parameters
- Security parameters
- Performance parameters
- Quality of Service parameters

3.3.2.1. Managed Objects for IMS Components

Managed Objects for the components of IMS are basically divided into seven modules. These modules further define the managed objects for each component. Managed Objects for similar components are classified together for the simplicity. These seven modules are:

- **SDPIMS-COMMON-MIB** contains common MIB objects used in all the IMS entities
- **SDPIMS-HSS-MIB** contains objects specific to HSS and SLF
- **SDPIMS-CALL-MIB** contains objects specific to Serving, Proxy and Interrogating CSCFs.
- **SDPIMS-APPSERVER-MIB** contains objects specific to Application Servers and Media Server
- **SDPIMS-GW-MIB** contains objects specific to Breakout Gateway, PSTN Gateway
- **SDPIMS-POLICY-MIB** contains objects specific to Policy Management at RACS, NASS and PDF.
- **SDPIMS-CHARGE-MIB** contains objects specific to Charging Collector Function

3.3.2.2. Managed Objects for SDP Components

Managed Objects for the components of SDP are not as exactly classified as those of IMS as the components of SDP are not consistent or standardized. We have divided them into five modules. These modules further define the managed objects for each component. Managed Objects for similar components are classified together for the simplicity. These five modules are:

- **SDPIMS-SERVICE-DISCOVERY-MIB:** consists of MIBs of configuration and performance statistics parameter related to Service discovery.
- **SDPIMS-SERVICE-CREATION-MIB:** consists of MIBs of configuration and performance statistics parameter related to Service Creation
- **SDPIMS-SERVICE-EXECUTION-MIB:** consists of MIBs of configuration and performance statistics parameter related to Service Execution
- **SDPIMS-SERVICE-ENABLERS-MIB:** consists of MIBs of configuration and performance statistics parameter related to Service Enablers
- **SDPIMS-SERVICE-ORCHESTRATE-MIB:** consists of MIBs of configuration and performance statistics parameter related to Service Orchestration

3.3.3.4. Trace / Traffic Record

Apart from supervising the vital states of IMS/NGN devices, IMS processes and traffic are constantly being monitored. This is achieved by passively monitoring traffic (with the help of distributed network taps) as well as by active performance tests (by means of an UA/AS emulator). Captured traffic information is being collected, correlated and analyzed centrally so that diagnostics constantly deliver information about the performance of the session control layer as well as about the quality of delivered services. Local Agents, deployed on IMS-Core devices as well as on components of the service enabling layer not only supervise processes and system states locally and enable the control of local processes centrally, but also conduct self-healing mechanisms autonomously.

Testing for a sound user-to-NGN connection implicates several probing mechanisms. By periodically emulating user registration procedures, not only connectivity parameters, but also performance parameters, can be observed. Logging connectivity errors over time provides insight about the overall robustness of the NGN. Tracking the registration delay, i.e. the time needed to successfully register at the IMS, over time allows for network performance monitoring and overload detection and prediction.

Passively capturing traffic at several, meaningful interfaces in the NGN allows not only for Service Level Agreement (SLA) validation, but also for early fault detection as well as performance evaluation. The big advantage of passive measurement strategies is the fact that the network performance is not being influenced, since no extra load is put on the network. However, correlating the captured traffic samples and analyzing them, is a complex task. On the network taps, adaptive filtering mechanisms allow for capturing of specific traffic profiles and streams. Bringing this logic into the network taps decreases the load on the central correlation and diagnostic entity. In relation to active measurement strategies, the passive strategy delivers more information about the eventually problematic points of failure in the NGN.

3.4. Functional Architecture

This architecture describes the six basic functional components of network management. These are defined as follows:

3.4.1. Performance Management

The goal of performance management is to measure and make available various aspects of network performance so that internetwork performance can be maintained at an acceptable level.

Performance management involves three main steps. First, performance data is gathered on variables of interest to network administrators. Second, the data is analyzed to determine normal (baseline) levels. Finally, appropriate performance thresholds are determined for each important variable so that exceeding these thresholds indicates a network problem worthy of attention.

Management entities continually monitor performance variables. When a performance threshold is exceeded, an alert is generated and sent to the network management system.

3.4.2. Fault Management

The goal of fault management is to detect, log, notify users of, and (to the extent possible) automatically fix network problems to keep the network running effectively. Because faults can cause downtime or unacceptable network degradation, fault management is perhaps the most widely implemented of the ISO network management elements.

Fault management involves first determining symptoms and isolating the problem. Then the problem is fixed and the solution is tested on all-important subsystems. Finally, the detection and resolution of the problem is recorded. Figure 4 gives an idea of over all fault reporting and error rectification (if possible).

3.4.2.1. Alarm & Notification

Whenever a fault is detected in the network an appropriate alarms is generated by the faulty network entity. These notifications of alarm contain all the information provided by the fault detection process. The information is used to identify the fault and in its localization.

3.4.3. Configuration Management

The goal of configuration management is to monitor network and system configuration information so that the effects on network operation of various versions of hardware and software elements can be tracked and managed.

Each network device has a variety of version information associated with it. Configuration management subsystems store this information in a database for easy access. When a problem occurs, this database can be searched for clues that may help solve the problem.

3.4.4. Accounting Management

The goal of accounting management is to measure network utilization parameters so that individual or group uses on the network can be regulated appropriately. As with performance management, the first step toward appropriate accounting management is to measure utilization of all important network resources. Analysis of the results provides insight into current usage patterns, and usage quotas can be set at this point. Some correction, of course, will be required to reach optimal access practices. From this point, ongoing measurement of resource use can yield billing information as well as information used to assess continued fair and optimal resource utilization.

3.4.5. Security Management

The goal of security management is to control access to network resources according to local guidelines so that the network cannot be sabotaged (intentionally or

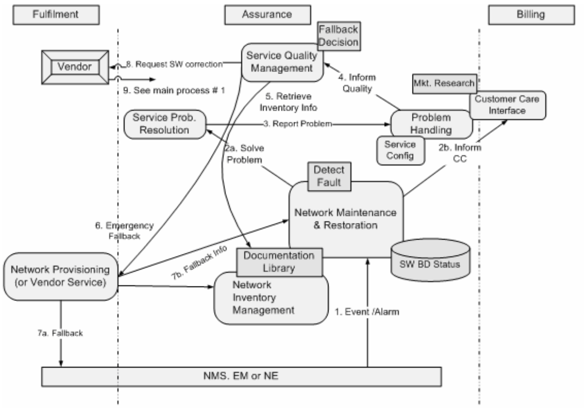

Figure 4. Notification and Fault recovery after fault detection

unintentionally) and sensitive information cannot be accessed by those without appropriate authorization. A security management subsystem, for example, can monitor users logging on to a network resource and can refuse access to those who enter inappropriate access codes.

Security management subsystems work by partitioning network resources into authorized and unauthorized areas.

3.4.6. Quality of Service Management

Quality of Service comprises all the aspects of a connection, such as time to provide service, voice quality, echo, loss, reliability and so on. The term Quality of Service is sometimes used as a quality measure, with many alternative definitions, rather than referring to the ability to reserve resources. Quality of Service sometimes refers to the level of Quality of service, i.e. the guaranteed service quality. High QoS is often confused with a high level of performance or achieved service quality, for example high bit rate, low latency and low bit error probability.

3.4.7. Test Management

Testing provides capabilities that can be used in different phases of the Fault Management (FM) and performance management and Provides end-to-end Fault Monitoring. For example:

- when alarm report is not sufficient to localize the faulty resource, tests can be executed to better localize the fault
- during normal operation of the NE, tests can be executed for the purpose of detecting faults
- once a faulty resource has been repaired or replaced, before it is restored to service, tests can be executed on that resource to be sure that it is fault free

Table 1. Alarm Notification attributes

Attribute Name	Description
AlarmId	The ID of the Alarm
notificationId	The ID of the Notification
alarmRaisedTime	Time at which alarm was raised
alarmClearedTime	Time at which alarm is cleared (if cleared)
alarmChangedTime	Time at which alarm was changed (if changed)
eventType	Type of Alarm event
probableCause	The cause of alarm
perceivedSeverity	Severity level of the fault
specificProblem	Problem which cause the fault
monitoredAttributes	The parameters which were monitored to detect the fault
proposedRepairActions	The proposed action to recover the fault
additionalText	Some additional messages
additionalInformation	Some additional messages
ackTime	Time the alarm is acknowledged

3.4.7.1. Network Element Testing

Following test can be performed to check the performance of the system and detecting faults:

- User-Network Interaction
- Registration
- Deregistration
- User to user session
- Call initiation
- User to Application Session
- Service Execution
- Service Cancellation
- System level check ups
- Heavy Load
- High Traffic

3.4.7.2. Logging

The alarm history information may be stored in the subordinate entities. The NM is able to create logs for alarm reports and to define the criteria for storage of alarm information. The alarm history information should be returned by files when UMP Agent finished collecting all the alarm history information that NM requested

4. Services Managed Objects

Service Level Managed Object is a new concept to define the service parameters which identify the performance, configuration, faults, accounting and security parameters. As the service is provided by the service provider with help of different node within the system, service parameters are gathered throughout the network. A SMO may consists of managed object from the Network Element level parameters gathered from IMS or SDP nodes or an SMO may be consisted of different SMOs. Following pseudo code shows an example of a high level SMO, which is composed of low level SMOs and final at the lowest level there are some Managed Objects.

High level SMOs are made of generic terminologies which are understandable to service provider, operator and as well as the customer. As the SMOs of lower level are defined, they become more and more technology specific. Given is an example of an SMO defined for a session disruption in IPTV service. The higher level SMO is defined and then more detailed SMOs are defined (one for each level).

```
IPTV_Service_Session_Disruption
{
attribute   IPTV_Service_Video_Visibility           videoVisibility;
attribute   IPTV_Service_Video_Flickering           videoFlicker;
attribute   IPTV_Service_Video_Distortion           videoDistortion;
attribute   IPTV_Service_Audio_Availability         audioAvalaibility;
attribute   IPTV_Service_Audio_Noise                audioNoise;
attribute   IPTV_Service_AV_NonSync                 avNonSync;
attribute   IPTV_Service_Channel_NonAvailability channelavalaibility;
attribute   IPTV_Service_Subscription               subscription;
}
IPTV_Service_Video_Visibility
{
attribute   IPTV_Service_Video_Quality              videoQuality;
attribute   IPTV_Service_Video_Codec                videoCodec;
attribute   IPTV_Service_Video_FrameRate            videoFrameRate;
attribute   IPTV_Service_Network_Quality            networkQuality;
}
IPTV_Service_Video_Quality
{
attribute   IPTV_Service_Video_Stream_Metric        videoStreamMetric;
            // Quality Metrics
attribute   IPTV_Service_Video_QM                   videoQM;
           // Transmission Quality
attribute    IPTV_Service_Video_TQ                    videoTQ;

attribute    IPTV_Service_Video_Transport_Metric
videoTransportMetric;
}
IPTV_Service_Video_Stream_Metric
{
attribute    IPTV_Service_Video_I_Frame_Count   iFrameCount;
attribute    IPTV_Service_Video_B_Frame_Count   bFrameCount;
attribute    IPTV_Service_Video_P_Frame_Count   pFrameCount;
}
IPTV_Service_Video_I_Frame_Count
{
attribute    unsigned_int              bFrameGoodCount;
attribute    unsigned_int              pFrameImpairedCount;
}
```

4.1. Managed Objects

Managed Objects are simple Management Information Base (MIB) objects, which are gathered from the network elements. Some of the managed objects for IMS and SDP components are given below.

The following paragraphs list the managed objects that may exist in the management model for triple play service using IMS and SDP. The managed objects that are directly derived from requirements are complemented by objects that their purpose is either administrative (e.g., table row index) or producing an efficient management model. Each managed object is described with the following attributes:

Attribute Name: Unique identifier of the managed object.

Type: Type of the Attributes

Description: Describes the role of the attribute in managed object.

4.1.1. Audio and Video Quality and Descriptive MOs

These Managed objects are gathered from the service enabler and content provider. They are shown in tables 2 – 5.

4.1.2. Transport Layer MOs.

These Managed objects are gathered through CSCF nodes, which are using the SIP protocol to maintain the call session at the transport layer. They are shown in table 6.

Table 2. Audio_Stream_MO

Attribute Name	Type	Description
audio_stream_id	Int	ID to uniquely identify the video stream.
audio_codec_type	Codec	Instance of Codec MO; describes type of codec (e.g. Speex,G.711)
audio_stream_type	Int	Type of audio stream e.g. Mono, Stereo, 5.1 surround,

Table 3. Video_Stream_MO

Attribute Name	Type	Description
video_stream_id	Int	ID to uniquely identify the video stream.
video_codec_type	Codec	Instance of Codec MO; describes type of codec (e.g. MPEG4)
gop	GoP	Group of Pictures e.g. IBBP
length	Int	No. of frames per GoP
resolution	Resolution	Image size in pixels (X x Y)
fps	Int	Number of Frames per second
scan_type	String	Interlaced/Progressive scan

Table 4. Video_Resolution_MO

Attribute Name	Type	Description
x_value	Int	Number of pixels on horizontal X-Axis
y_value	Int	Number of pixels on vertical Y-Axis

Table 5. Codec_MO

Attribute Name	Type	Description
codec_id	Int	Unique ID to identify the Codec
codec_type	Int	0=Audio, 1=Video
codec_name	String	Name of the codec.
average_bit_rate	Int	Average bit rate per second
sampling_rate	Int	Number of samples per second (Hz)

4.1.3. Multicasting MOs.

These MOs are gathered from the network layer. The components, which provide these MOs are Multicast Service enabler, RACS and CSCF. These MOs are shown in tables 7 and 8.

Table 6 Packet_Loss_MO

Attribute Name	Type	Description
session_id	Int	Transport layer Session ID.
total_packets_transmitted	Int	Number of packets transmitted
packets_lost_uncorrected	Int	Number of packets lost without FEC (Forward Error Correction).
packets_lost_corrected	Int	Number of packets lost with FEC.
total_packets_received	Int	Number of packets received.
packets_discarded	Int	Number of packets discarded due to late arrival.
packets_out_of_sequence	Int	Number of packets arriving out of sequence.
packet_found_duplicate	Int	Number of duplicate packets.
packets_burst_lost	Int	Number of packets lost within burst.
average_burst_length	Float	Average burst length.
packets_gap_lost	Int	Number of packets lost within gap.
gap_length	Float	Average gap length

Table 7. Multicast_Session_MO

Attribute Name	Type	Description
audio_stream_id	Int	ID to uniquely identify the video stream.
video_stream_id	Int	ID to uniquely identify the video stream.
ip_mcast_group_id	Int	ID of the multicast group.
ip_mcast_network_itface_list	List	List of interfaces subscribed to the servive. This is a list of ip_mcast_route_list object (described below)
ip_mcast_protocol	String	Multicast protocol running on the stream.
ip_mcast_interface_ratelimit	Int	The rate-limit, in kilobits per second, of forwarded multicast stream. A rate-limit of 0 indicates that no rate limiting is done.
ip_mcast _packets_transferred	Int	Number of packets transmitted
ip_mcast	Int	Number of packets received

_packets_received		

Table 8. IP_MCast_Interface_MO

Attribute Name	Type	Description
ip_mcast_iface_index	Int	ID of Interface
ip_mcast_iface_ratelimit	Int	The rate-limit, in kilobits per second, of forwarded multicast stream at specific interface. A rate-limit of 0 indicates that no rate limiting is done.
ip_mcast_iface_transferred	Int	Number of packets transmitted at given interface
ip_mcast_iface_received	Int	Number of packets received at given interface.

4.1.4. Service Subscription MOs

The MOs are gathered from the NASS and also from HSS nodes are shown in tables 9 & 10.

Table 9. NASS_Configuration_MO

Attribute Name	Type	Description
nass_server_index	Int	A unique identifier of a server address when multiple addresses are configured. If one address isn't reachable, then another can be tried
nass_server_address_type	String	0 for IPv4 1 for IPv6
nass_server_address	Int	Number of packets transmitted at given interface
nass_server_role	List	List of strings. Depicts the role of NASS like accounting, user profiling, and policy and so on.
nass_server_authentication_method	String	Authentication method used by the NASS server.

5. Management in SDP/IMS: A Case Study (Triple Play Service)

The "Triple Play" [5] is the new buzzword describing the convergence of the three terms: "voice (telephony), internet and TV as commercial notation for driving market rather than a new technology. This is challenging the traditional telecoms world dramatically. Multimedia Internet services have paved the way for the emergence of content based services and new business models.

Table 10. NASS_Stats_MO

Attribute Name	Type	Description
nass_server_index	Int	A unique identifier of a server address when multiple addresses are configured. If one address isn't reachable, then another can be tried
nass_server_request_failures	Int	0 for IPv4 1 for IPv6
nass_server_up_time	Time	Depicts server up time
nass_server_number_of_disconnects	Int	Number of times an NASS server goes down.
nass_server_max_user_expiry	Time	This object reflects the maximum expiry that may be requested by a User Agent for a particular contact.
nass_server_ registered users	List	List of users object. This object reflects list of users currently registered.
nass_server_registereduser_count	Int	This object reflects number of users currently registered.

Triple play provides an access interface (an example shown in figure 5), which can be a TV monitor or a web page with a large menu organized in several categories: Movies, Music, Video games, and radio. This interface also involves some options concerning the control (i.e. Start, Skip, and Stop) of the whole program .The user will be able to,

- Start the program (that he has chosen) right away or at a specific time
- Modify the program and do another choice before starting watching it
- Skip the media currently playing and go to the next one
- Stop the program. Let us also mention that the user can stop the program without using the function "Stop" of the interface

The integration of IPTV services in the IMS architecture has been identified and will be one of the key features of IMS/SDP in the near future. The easiest way for integrating these services consists in defining a dedicated IPTV subsystem sharing already available interfaces like the Resource and Admission Control Subsystem (RACS) and Network Attachment Subsystem (NASS) with other subsystems in combination with the core IMS network. Driven by a strong market demand for more sophisticated integration levels of IPTV and (mobile) communication services this strategy will provide a new multimedia experience to the end-user.

5.2. Triple Play Management & Control

Service providers around the world are moving swiftly toward triple-play service delivery as the core strategy to increase competitive differentiation, expand revenue and gain new market share. One such goal is to provide innovative and user-centric IPTV and streaming multimedia services combined with VoIP and Internet access as a convenient one-stop shopping experience for end users.

To introduce new services efficiently - such as voice over IP (VoIP), video on demand (VoD) and IP television (IPTV) - providers must maximize investments already made in existing infrastructures. They also require dynamic service management tools to support sophisticated new services. But to achieve long-term success, triple play service providers that offer voice, video, data and other Internet Protocol (IP)-based services must place a new emphasis on content, customer satisfaction, price and

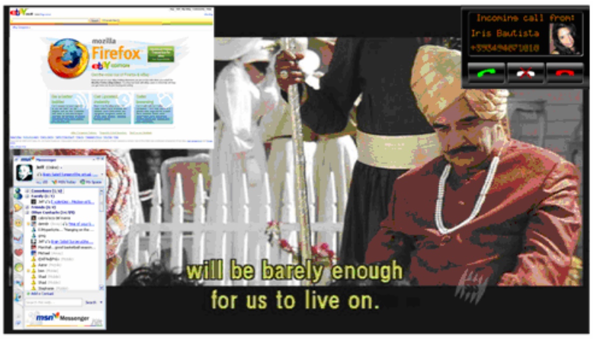

Figure 5. A view of a Triple play service

quality of service (QoS).

The key performance and quality indicators for IPTV and VoIP Services in real time are:

For VoIP Quality

- Mean opinion score (MOS)
- Grade of Service (GoS)---percentage of calls blocked by the network/platform
- Drop call rates (normal termination / abnormal)
- Post Dial Delay (PDD)
- Traffic Load

For IPTV Quality

- Mean opinion score (MOS)
- Peak Signal to Noise Ratio
- Mean Squared Error
- Channel Surfing Time
- Channel Surfing Errors
- Video Stream Utilization
- Synchronization Errors
- Blockiness/Blurriness/Jerkiness

These parameters are measured at the managed nodes in the Network along with the Set-Top-Boxes and Access network nodes and centrally monitored by a management server. SNMP agents at each managed node are deployed which maintain up-to-date values for these parameters and can be requested by the manager. For measuring the performance and QoS of the service, we have defined Service Manage Objects (SMOs). These SMOs uses Managed Objects at different SDP and IMS components and helps in management of the provisioned services.

5.3. Triple Play Service Management

Service management in the system is handled by the ISMS server. Figure 6 shows the modular diagram of ISMS. Along with the conventional modules of any management server, there are five more modules for service management. These modules are SMO Creator, SMO Composer, Threshold Comparer, Error & Fault Detecter and Reporter & Notifier

A typical service disruption observed by an end user, it reporting and management can be envisioned as follows:

- Service is disrupted while the client is watching IPTV
- Customer calls the Help Desk using the VoIP service
- From the Help desk the message of Service disruption is sent to ISMS. IPTV_Session_Service_Disruption SMO is created at the ISMS by the SMO Creator module.
- ISMS sends on-demand request of Managed Objects to the Managed Nodes and Managed Objects (MOs) are sent back by the different IMS, SDP nodes and other network elements. SMO composer builds up SMO from these gathered MOs
- Using the SMOs the system can identify the problems in the system by comparing the measured values with the threshold values
- Fault detection & localization is performed by the fault detection module and alarm notification is sent to the configuration management module for reconfigurations. Similarly, maintenance parameters are sent to the Network Managers and maintenance personnel
- Once the problem is identified, it can also be reported to the concerned nodes (Network Elements). The notification is made to the service providers and QoS and SLA ensuring nodes
- Business processing modules may be notified about the problems in the system
- The configuration parameters of those nodes are changed accordingly
- Testing can be performed to verify the system after the problem is solved

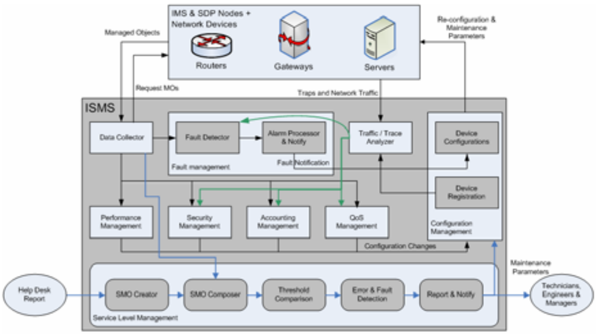

Figure 6. Modular diagram of IMS/SDP management server (ISMS) shows the working of the management scenario.

6. Conclusion

In this paper, we provided a anagement server for IMS/SDP solution, which utilizes the new concept of service managed objects. To be a competitor in the field of service provision an IMS/SDP solution needs to be properly managed to provide competitive services at high quality with efficient performance. This freamework promises to provide a solution to this critical problem.

7. Reference

[1] TS 23.228 IP Multimedia Subsystem (IMS), Stage 2/3GPP2 X.S0013-002-0 v1.0, *www.3gpp.org*

[2] Service Delivery Platform - Efficient Deployment Of Services, Whitepaper, Ericsson.

[3] Miikka Poikselka, Aki Niemi, Hisham Khartabil, Georg Mayer *"The IMS: IP Multimedia Concepts and Services"* (John Wiley & Sons, 2006, ISBN 0-470-01906-9)

[4] The IMS Playground @ FOKUS – www.open-ims.org

[5] Triple play (telecommunications) *http://en.wikipedia.org/wiki/Triple_play_(telecommunications)*