

APPROACH FOR INTEGRATION OF THIRD PARTY NODES WITH PROPRIETARY OSS

1.1 ABSTRACT

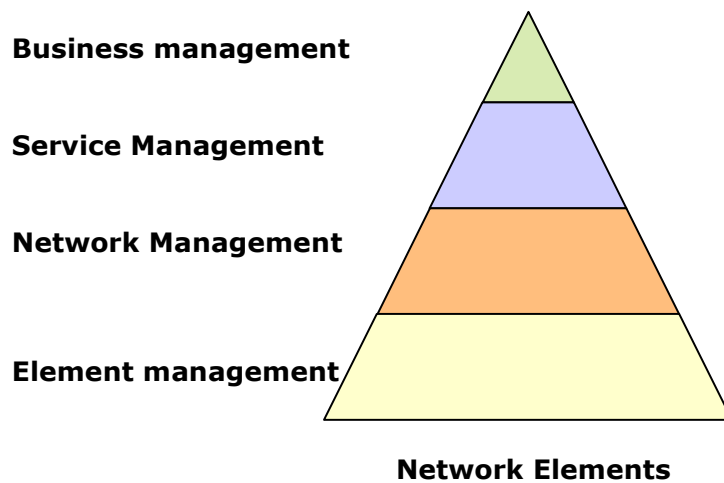
For efficient Operations, network elements of a Telecom Network (e.g. Cellular Network) are integrated with Operations support Systems (OSS). Typically the OSS comes from the vendor supplying the key Access and Core network nodes of the network as these nodes are well integrated with the proprietary OSS. Invariably the Network also includes Nodes from other vendors too to provide value added services to subscribers. To get a complete view of the Network for effective Fault and Performance management, Operators & Service providers mandate the integration of 3rd Party Nodes with the existing Proprietary OSS.

This Paper presents a brief introduction of Operations Support System (OSS) and discusses common approach for Fault Management (FM) integration of the 3rd party nodes with proprietary OSS in the customer environment.

CONTENTS	
<u>SECTION</u>	<u>PAGE</u>
1.1 ABSTRACT	1
1.2 OSS-TMN ARCHITECTURE.....	1
1.3 SNMP – TERMINOLOGY EXPLAINED	3
2.0 FAULT MANAGEMENT (FM) INTEGRATION OF 3RD PARTY NODES WITH PROPRIETARY OSS.....	3
3.0 HOW TO APPROACH AND WHAT TO LOOK FOR ..	4
4.0 CONFIGURATION STEPS	4
5.0 CASE STUDY WITH NETACT (NSN PRODUCT) AS OSS	5
6.0 CHALLENGES	5
7.0 REFERENCES	6

1.2 OSS-TMN Architecture

Operations Support System (OSS) is huge, and several attempts to slice it into understandable pieces exist. One, which has gained a general acceptance, is the TMN (Telecommunications Management Network) model M.3100 by ITU. They introduced the “TMN pyramid” where the management is divided into a number of layers as depicted below:



The meaning of the layers is described below. Please note that each layer is dependent on the services provided by the adjacent layer below:

Network Elements: Elemental or fundamental units of a network, such as a transmitter, amplifier, repeater, multiplexer, switch, router, copper or fiber optic transmission link, microwave antenna, or receiver constitute a network element.

Element Management: The management functionality that is required to operate an individual network element. The functionality is divided into five key areas: Fault, Configuration, Accounting, Performance and Security (FCAPS). This layer's northbound surface interfaces to Network Management System and it's southbound to the network elements (NE).

Network Management: When multiple network elements are interconnected they form a network. Network Management refers to the functionality required to monitor and administer a network. It involves a distributed database, auto polling of network devices, and high-end workstations generating real-time graphical views of network topology changes and traffic. In general, network management is a service that employs a variety of tools, applications, and devices to assist human network managers in monitoring and maintaining networks.

Service Management: The network provides services. A leased line subscription, an email account and a telephone subscription are a few examples. The Service management refers to the controlling of these services.

Business Management: The services are provided to subscribers (customers). The customer management and related issues like billing is referred to as Business Management.

The strength of the TMN model is that it provides the capability to reach a level of abstraction that is increased through the layers. Ideally there is no need for interference between layers that are not adjacent. Another contribution from the TMN model is the division of management functionality into a number of functional areas as mentioned before, the FCAPS model:

Fault Management: Handling of alarms.

Configuration Management: Installing and configuring the object in questions, be it a service or a physical port.

Accounting Management: The creation and mediation of resource usage data and the subsequent rating and billing of the service usage.

Performance Management: The creation, collection and aggregation of statistics related to resource usage. The creation and handling of reports related to the collected statistics.

Security Management: All aspects related to security of the management functionality. The area spans from authentication of the operators to access control, i.e. who is allowed to do what, when, from where.

1.3 SNMP – Terminology explained

Simple Network Management Protocol (SNMP) is a popular protocol for network management. It is used for collecting information from, and configuring, network devices, such as servers, printers, hubs, switches, and routers on an Internet Protocol (IP) network.

Basic SNMP components:

1- Managed devices. 2- SNMP agent. 3- SNMP Manager (NMS)

Each managed device runs a process called an *agent*. The agent is a server process that maintains the MIB database for the device. An SNMP manager is an application that generates requests for MIB information and processes the responses. The manager and agent communicate using the Simple Network Management Protocol.

SNMP agents (like `snmpd` daemon process) typically have predefined MIB objects that they can access. `SNMPD` is an SNMP agent which binds to a port (default port: 161) and awaits requests from SNMP manager. Upon receiving a request, it processes the request(s), collects the requested information and/or performs the requested operation(s) and returns the information to the sender.

An SNMP subagent is used to extend the number and type of MIB objects that an SNMP agent can support.

An SNMP manager can issue requests to an agent either to retrieve information from the agent's MIB (an SNMP Get request), or to change information in the agent's MIB (an SNMP Set request). An SNMP agent can also send unsolicited messages to the SNMP manager (SNMP traps).

The SNMP agent talks to both subagents and managers. The SNMP manager (which resides on one node in the network) sends requests to the agent (which resides on another). The agent sends responses and traps to the manager.

2.0 FAULT MANAGEMENT (FM) INTEGRATION OF 3RD PARTY NODES WITH PROPRIETARY OSS

From the customer's perspective, fault management forms one of the most important aspects to deal with in their day to day network management activities. It helps them to detect a faulty node in the network remotely and depending on the nature and severity of the alarm, actions are taken accordingly.

The fault management (FM) integration of a network node to an OSS typically means a successful integration resulting in receiving the traps from the network node (hosts SNMP agent) by an OSS (hosts SNMP manager) and displaying it on the alarm monitor with the meaningful information. The alarm monitor would be used by the network operator (FM engineer to be specific) and is responsible for the corrective actions on the faulty node thereafter.

The third party nodes may not be compliant with the proprietary OSS with respect to the format of the trap/PDU (protocol data unit). Hence, it becomes the responsibility of the OSS to convert the traps from the managed node (SNMP agent) into its own desirable format to display it on the alarm monitor. And for the SNMP get and set requests initiated by the SNMP manager, the OSS should be compliant with SNMP get and set PDUs formats specified under standard SNMP.

Some third party platforms may demand supervision of their SNMP agents from OSS. This feature is enabled from the SNMP manager and the agent should be capable enough to respond with appropriate information requested by the manager for supervision.

3.0 HOW TO APPROACH AND WHAT TO LOOK FOR

Fault management is an integral part of a Network Management System, and it entirely depends on the proprietary OSS as to how it would provide this feature.

Before you start the integration process, it is very important to understand the architecture of the third party node to be integrated and identify the crucial elements that can help us to appreciate the flow of the traps within their system. Help should be sought from the third party personnel in the following areas of their system:

- Architecture
- Flow of the traps within their system
- SNMP agents and subagents
- MIB files
- Other general system information based on the need

For integration personnel, the core information of the traps that would be sent by the managed device can be known from what are called as MIB files (Management Information Base). These MIB files are provided by the system where the SNMP agent and subagents are running.

These files provide most of the information about the various traps that the system is likely to send. Apart from this, it also has information about various objects and variables of the system, and the same can be queried and set by the SNMP manager through SNMP. All these traps, objects etc are identified by their OIDs (object Identifier).

These MIB files have to be analyzed in order to understand the format of the trap PDU.

4.0 CONFIGURATION STEPS

The scope of this document is limited to third party FM integration with an OSS.

As mentioned previously, third party nodes provide the needed information through MIB files. Once the information described in the previous section has been collected, OSS needs to be configured with the same. This configuration lies within the scope of OSS as to how it would facilitate it. For example, some OSSs provide the FM integration personnel with a toolkit which assists them in configuration. Meanwhile the third party should be doing the needed SNMP configuration from their end and they should be provided with OSS details like IP address, SNMP port number where OSS listens for the traps (default UDP port: 162).

The entire process can be broadly understood with the following steps:

- Configuration of the IP address of the managed node in OSS
- Adding the managed object (third party node) to the topology database maintained by the OSS.
- Configuration of the trap information in OSS
- IP connectivity between OSS and the managed node.
- Start receiving traps from the managed node.
-

The third party node is added to the topology database of OSS which enables the node to be managed from OSS's TLUI (Top Level User Interface) and to display its alarms on the alarm monitor.

Once the configuration is done, make sure that there is connectivity between OSS and the third party node over the customer's IP network. Basic connectivity can be confirmed by using ping command. There would be a separate team at the customer's place taking care of the network. Help can be taken from them. As mentioned above, SNMP manager listens on UDP port 162 for the traps which are being sent over the network. Hence, the managed node should send the traps to port 162.

This completes the integration task and a thorough testing should be done from both the ends. All types of alarms (critical, major and minor) should be tested to declare the integration as successful.

5.0 CASE STUDY WITH NETACT (NSN PRODUCT) AS OSS

This case study talks about third party. The third party nodes described in this section are the VAS (Value Added Service) nodes which have been employed by the customer in their network. Coming to the proprietary OSS, NetAct is a Network management System provided by NSN to manage a mobile network comprising network elements belonging to NSN as well as third party. NetAct provides a toolkit called “SNMP Integration Toolkit” to accomplish this task. This toolkit is one of the subsystems of NetAct.

Below configuration steps are followed:

- IP address of the managed node is configured on NetAct server by editing /etc/hosts file.
- The topology database of NetAct is updated with the new managed node.
- Once the information described about the third party node in the section 4 is collected, its time to use it in order to configure the SNMP integration toolkit. There are three important configuration files to be dealt with. They are “servicesmx.cf”, “traps.cf” and mapping.cf.
- The file servicesmx.cf is used to configure SNMP services to be provided for the third party node.
- The file traps.cf is written after analyzing the MIB files of the managed node. The MIB file is the source for writing the traps.cf file which eventually will have the entries of all the traps that the third party node is likely to send. The SNMP toolkit upon receiving a trap makes use of mapping.cf in order to resolve the content of the traps defined in traps.cf.
- Ensure that the connectivity exists between NetAct OSS and the managed node.
- Ensure from the third party personnel that the managed node has been configured appropriately to send the traps to NetAct.

All the traps coming from the managed node are seen on the alarm monitor of NetAct.

6.0 CHALLENGES

There are multiple challenges one faces while integrating the 3rd party nodes with OSS first time

- In few cases, the third party platforms do not send the trap in the expected format. The format is understood from the MIB file of the third party which defines the sequence of variables of the trap. For example, if the first field of the trap is defined in the MIB file to contain “alarm description” and when the OSS receives the trap with first field having “alarm number”, this becomes a case where the third party platform fails to be compliant with its own MIB file, and thus resulting in the unexpected format of the trap. As a consequence of this, the OSS either fails to process the trap or displays wrong information on the monitor. This is a clear error from the third party node. In such a scenario, the same has to be communicated to the third party team and seek a correction from them.
- Another challenge lies in testing of all the possible traps. Since the third party platform would not be sending all the traps without any reason, they will have to be generated manually by the third party team. This also becomes a challenge for them but not impossible. It is usually agreed by all the parties involved in the integration that this is an acceptable and preferred way of testing. When the alarm monitor displays all the information correctly, it indicates that the various fields of the trap have been processed successfully.

7.0 REFERENCES

- Configuring the SNMP integration toolkit – Authored by NSN

PROPRIETARY NOTICE

All rights reserved. This publication and its contents are proprietary to Hughes Systique Corporation. No part of this publication may be reproduced in any form or by any means without the written permission of Hughes Systique Corporation, 15245 Shady Grove Road, Suite 330, Rockville, MD 20850.

Copyright © 2006 Hughes Systique Coporation

CONTACT INFORMATION:

phone: +1.301.527.1629

fax: +1.301.527.1690

email: whitepaper@hsc.com

web: www.hsc.com

APPENDIX A ABOUT HUGHES SYSTIQUE CORPORATION

HUGHES Systique Corporation (HSC), part of the HUGHES group of companies, is a leading Consulting and Software company focused on Communications and Automotive Telematics. HSC is headquartered in Rockville, Maryland USA with its development centre in Gurgaon, India.

SERVICES OFFERED:

Technology Consulting & Architecture: Leverage extensive knowledge and experience of our domain experts to define product requirements, validate technology plans, and provide network level consulting services and deployment of several successful products from conceptualization to market delivery.

Development & Maintenance Services: We can help you design, develop and maintain software for diverse areas in the communication industry. We have a well-defined software development process, comprising of complete SDLC from requirement analysis to the deployment and post production support.

Testing : We have extensive experience in testing methodologies and processes and offer Performance testing (with bench marking metrics), Protocol testing, Conformance testing, Stress testing, White-box and black-box testing, Regression testing and Interoperability testing to our clients

System Integration : As system integrators of choice HSC works with global names to architect, integrate, deploy and manage their suite of OSS, BSS, VAS and IN in wireless (VoIP & IMS), wireline and hybrid networks.: NMS, Service Management & Provisioning .

DOMAIN EXPERTISE:

Terminals

- Terminal Platforms : iPhone, Android, Symbian, Windows CE/Mobile, BREW, PalmOS
- Middleware Experience & Applications : J2ME , IMS Client & OMA PoC,

Access

- Wired Access : PON & DSL, IP-DSLAM,
- Wireless Access : WLAN/WiMAX / LTE, UMTS, 2.5G, 2G ,Satellite Communication

Core Network

- IMS/3GPP , IPTV , SBC, Interworking , Switching solutions, VoIP

Applications

- Technologies : C, Java/J2ME, C++, Flash/lite,SIP, Presence, Location, AJAX/Mash
- Middleware: GlassFish, BEA, JBOSS, WebSphere, Tomcat, Apache etc.

Management & Back Office:

- Billing & OSS , Knowledge of COTS products , Mediation, CRM
- Network Management : NM Protocols, Java technologies,, Knowledge of COTS NM products, FCAPS, Security & Authentication

Platforms

- Embedded: Design, Development and Porting - RTOS, Device Drivers, Communications / Switching devices, Infrastructure components. Usage and Understanding of Debugging tools.
- FPGA & DSP : Design, System Prototyping. Re-engineering, System Verification, Testing

Automotive Telematics

- In Car unit (ECU) software design with CAN B & CAN C
- Telematics Network Design (CDMA, GSM, GPRS/UMTS)

BENEFITS:

- **Reduced Time to market :** Complement your existing skills, Experience in development-to-deployment in complex communication systems, with key resources available at all times
- **Stretch your R&D dollars :** Best Shore” strategy to outsourcing, World class processes, Insulate from resource fluctuations