



SGSN-MME 2010 System Administration

1 *System and Architecture*

OBJECTIVES

Upon completion of this course, the student will be able to:

- › Describe basic procedures in GSM and WCDMA handled by the SGSN-MME such as GPRS Attach, PDP Context Activation and Inter-SGSN Routing Area Update.
- › Explore the SGSN-MME software and hardware architecture.
- › Discover IP address usage in the SGSN-MME and how different components are identified.

Figure 1-1. Objectives

INTRODUCTION

THE GPRS NETWORK

The advent of mobile telecommunications has brought about many changes and evolution to the network infrastructure supporting the services provided by operators to the end users. The mobile networks that we see today comprises of the core network and the radio network.

Earlier networks providing circuit switched or voice call services worked well with the GSM radio network technology. However, with the introduction of 3G mobile networks, the range of services available expanded to high speed data services as well as video call capabilities among others. Thus, the mobile core network was improved to provide support for both GSM and WCDMA radio access and more importantly data services.

The packet switching component of the GSM/WCDMA core network is called General Packet Radio Services (GPRS). The main switching nodes that add packet switching functionality to the GSM/WCDMA network are called the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN). The GPRS Support Node detailed in this book is based on Ericsson's Wireless Packet Platform which will be explained later in this chapter. The evolution of the GPRS network will also be explained i.e. 'Evolved Packet Core' (EPC).

The SGSN which is now called 'SGSN-Mobility Management Entity (SGSN-MME) provides packet routing to and from the geographical service area referred to as the Routing Area. The Routing Area is a subset of the traditional Location Area used for Circuit Switching. While the GGSN provides the interface between the internal GPRS network and the external Internet Protocol (IP) networks, like a Corporate LAN, ISP, etc.

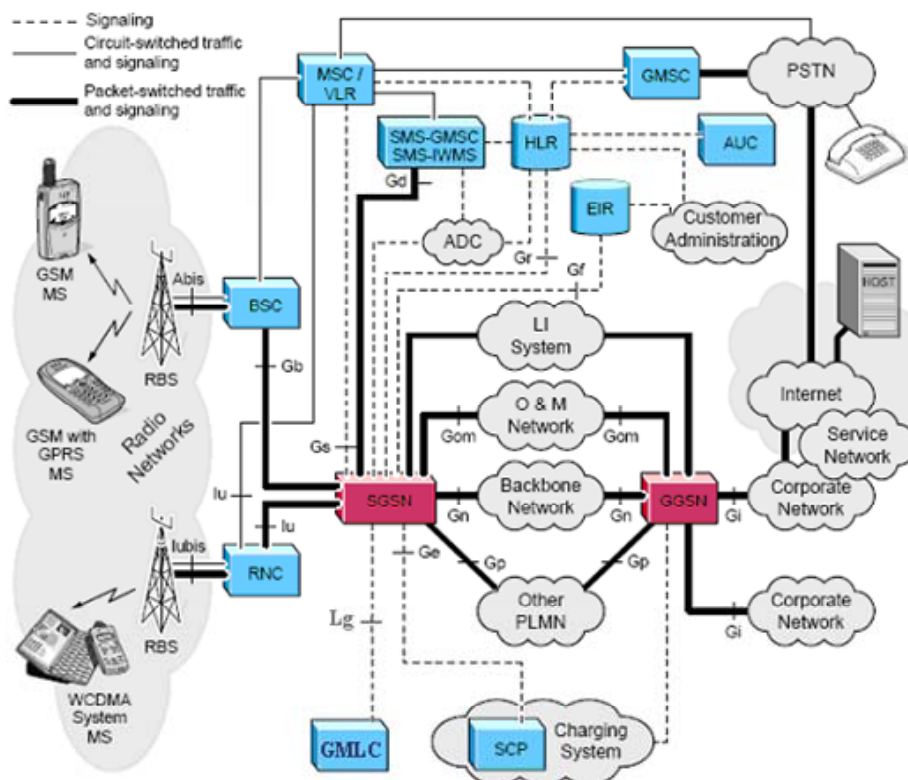


Figure 1-2. GPRS network – logical view

Ericsson's implementation of the SGSN-MME and GGSN is the physical separation between the traditional Circuit Switched parts of the generic GSM/WCDMA architectures. The implementation complies with the generic GPRS architecture specified by 3rd Generation Partnership Project (3GPP) Technical Specification. All interfaces and reference points to the SGSN-MME and GGSN either complies with the 3GPP specification or is based on other open standards. The GSN nodes also comply with the 3GPP standards detailed for both Circuit Switched and Packet Switched architectures. Functions of the Packet Switching nodes are discussed below.

SGSN-MME

The SGSN-MME Node is a primary component in the GSM/WCDMA GPRS architecture. The SGSN-MME's primary task is to forward incoming and outgoing IP packets addressed to/from a mobile station that is attached within the SGSN-MME service area. In particular the SGSN-MME provides:

- Packet routing, roaming and transfer to and from the SGSN-MME service area. It serves all GPRS subscribers that are physically located within the geographical SGSN-MME service area, much the same way that the MSC/VLR serves MS's

currently roaming in the Circuit Switched environment. A GPRS subscriber may be served by any SGSN-MME in the network, all depending on mobile station (MS) location. The traffic is routed from the SGSN-MME; to the Base Station Controller (BSC) via the Base Transceiver Station (BTS) in GSM Systems, or via the Radio Network Controller (RNC) via the Node B (Node B) in WCDMA Systems toward the MS/UE.

- Security over radio access is by means of ciphering and authentication.
- Session and mobility management procedures.
- Logical link management towards the MS.
- Connection via physical interfaces to Home Location Register (HLR- Gr interface), Mobile Switching Centre / Visitor Location Register (MSC/VLR- Gs interface, only GSM), Base Station Controller (BSC- Gb interface, only GSM), Short Message Service-Gateway MSC (SMS-GMSC- Gd interface), SMS-Inter Working MSC (SMS-IWMSC- Gd interface), Service Control Function (SCF- Ge interface), Equipment Identity Register (EIR- Gf interface), Gateway Mobile Location Centre (GMLC- Lg interface), Other Public Land Mobile Network (PLMN- Gp interface) and the Gateway GPRS Support Node (GGSN- Gn interface). In WCDMA Systems the only difference is the connecting interface toward the Radio Network Controller (RNC) this interface is the Iu interface, it is logically separated into the Iu-U and the Iu-C (User and Control planes).
- Output of charging data, in the form of Call Data Records
- (CDR's). The SGSN-MME collects CDR's for each MS related to the radio network usage and events. SGSN-MME supports prepaid services via the Customized Application for Mobile Enhanced Logic (CAMEL) application.

GATEWAY GPRS SUPPORT NODE - GGSN

Like the SGSN, the Gateway GPRS Support Node (GGSN) is a primary component in the GSM/WCDMA GPRS architecture. In particular the GGSN provides:

- The interface towards the external IP packet networks (ISP's, Corporate LAN's, etc). The GGSN, therefore, contains access functionality such as Remote Access Dial-In User Service (RADIUS) and a Dynamic Host Configuration Protocol (DHCP) Client, which is used for security and IP address allocation purposes.

- Gateway function to external IP networks for both GSM/WCDMA/GPRS networks.
- GPRS session management, communication set-up towards external IP networks.
- Connection via physical interfaces to the Serving GPRS Support Node (SGSN-MME- Gn interface), the external IP packet networks (ISP's, Corporate LAN's, etc-Gi interface) and to Other Public Land Mobile Network (PLMN- Gp interface).
- Output of charging data in form of CDR's. The GGSN collects charging information for each MS/UE related to the external data network usage. The GGSN collects charging information on usage of the GPRS network resources.
- A Border Gateway (BG) functions for secure interconnection with other packet data networks.

TRAFFIC CASES

GPRS ATTACH PROCEDURE

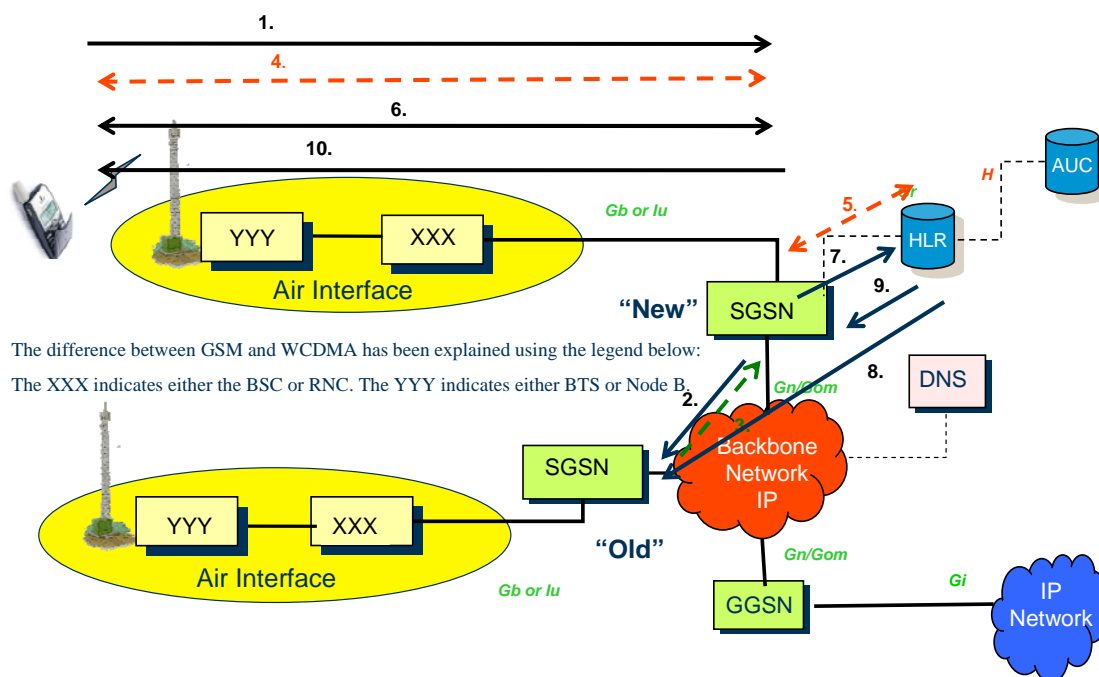


Figure 1-3. GPRS attach procedure

The below mentioned GPRS attach procedure is both relevant for GSM and WCDMA systems. The only thing to remember is that the radio access for GSM will be controlled by the BSC, whilst the radio access for WCDMA will be controlled by the RNC. The diagram below will be talking in generic terms. Signaling sequences are almost identical for both GSM and WCDMA systems.

- 1 The MS/UE sends a **“GPRS attach request”** to the NEW SGSN-MME. This message contains the old Routing Area Identity (old RAI) as well as the old Packet – temporary IMSI (P-TMSI). The NEW SGSN-MME will consult his Co-operating SGSN table (this ties together the old SGSN and old RAI).
- 2 The NEW SGSN-MME will ask the OLD SGSN-MME for information like about the MS/UE (the P-TMSI is presented; therefore the IMSI can be identified. See above).
- 3 If the MS/UE is known then the OLD SGSN-MME will send the Authentication triplets (GSM) or quintuplets (WCDMA) and IMSI to the NEW SGSN-MME. Or it will respond with MS/UE unknown.
- 4 If the OLD SGSN-MME does not know about the MS/UE then the NEW SGSN-MME will ask the MS/UE for its IMSI Number (only).
- 5 The NEW SGSN-MME fetches the Authentication triplets (GSM) or quintuplets (WCDMA) for the corresponding IMSI from the HLR.
- 6 The NEW SGSN-MME will authenticate the MS/UE, using one of the authentication triplets.
- 7 The NEW SGSN-MME then sends a MAP **“Update Location”** message (containing the IMSI and NEW SGSN-MME Address) to the HLR, to update the location address of the IMSI. The SGSN-MME converts the IMSI into Mobile Global Title (MGT); the conversion information is stored in the IMSI series analysis table. MGT is the routing information used to route the location update message to the correct HLR on the SS7 network.
- 8 The HLR will send a MAP **“Cancel Location”** message to the OLD SGSN. The OLD SGSN will purge the MS/UE information from its VLR.

- 9 The HLR also complies with the Subscribers data to send back to the NEW SGSN-MME. The common information for GPRS is the subscribed Access Point Names (APN) and their respective Quality of Service parameters (QoS), along with any other relevant parameters.
- 10 The NEW SGSN-MME will then inform the MS/UE that they are “Location updated” with the HLR. The information passed through to be temporarily stored on the SIM of the MS/UE, contains among things the new LAI/RAI.

GPRS PDP CONTEXT ACTIVATION PROCEDURE

Listed below are the “PDP context activate” procedures for both GSM GPRS and WCDMA GPRS. This has been included to detail the slight differences between the ways GPRS is implemented, for both systems. Another important note to make here is the procedure is affected in different ways depending on the configuration of MS/UE APN selection method and the IP address allocation method. These traffic cases are assuming that the IP address allocation method is “External IP network” allocated, also that the MS/UE can select from a list of subscribed APN’s.

GSM GPRS PDP Context Activation

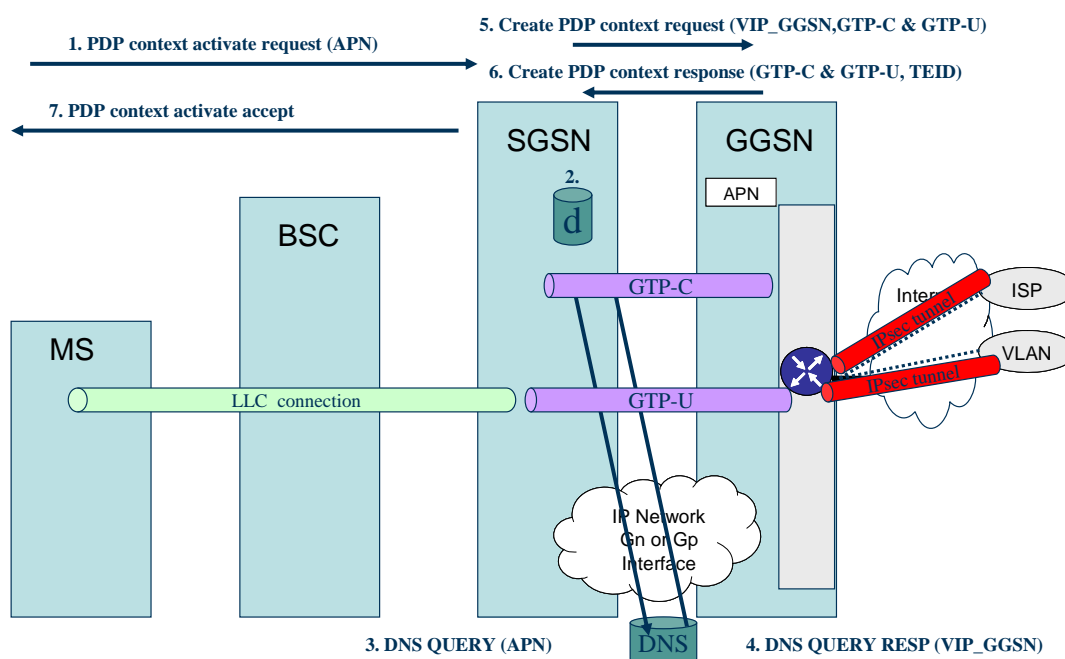


Figure 1-4. GSM GPRS PDP Context activation

The below mentioned procedure can only be initiated after the MS has performed a GPRS attach.

- 1 The MS begins by sending a “**PDP context activate request**” message to the SGSN-MME currently serving the MS, the subscriber selects the particular APN (external IP network). Given the list of APN’s that was returned and stored in the VLR of the SGSN-MME, the SGSN-MME will perform the Authentication and security functions, then check the subscribers subscriptions with it’s QoS parameters. The SGSN-MME also modifies the APN by adding the “Default APN Operator ID” to the back.
- 2 The SGSN-MME then checks the APN against the “caching only DNS” on the NCB for a result. If there is a result there then we move to step 5. If not then go to step 3. At this point the APN looks something like this: *www.isp.com.mncXXX.mccXXX.gprs*. The default APN Operator ID is retrieved from the IMSI series analysis table defined in the SGSN-MME.
- 3 The SGSN-MME then constructs a “DNS query” message containing the APN to be resolved. It is important to note that the APN at this stage is the APN plus the Default APN Operator ID. This message is sent across the internal IP network of the operator, to a predefined Domain Name Server (DNS).
- 4 The DNS will look up the APN in its table. The result should equate to a particular VIP_GGSN IP address. The VIP_GGSN is returned to the SGSN-MME in a DNS query result. This result is stored in the caching only DNS of the NCB for the next time an MS uses the same APN.
- 5 The SGSN-MME then constructs a “**Create PDP context activate request**” message. This is sent out toward the GGSN. This message contains information used to establish a GTP-C and GTP-U tunnel on the SGSN-MME. There is also a Tunnel End Identity (TEID), the APN, an NSAPI, MSISDN and QoS profile.
- 6 Once the GGSN has received and processed the “Create PDP context activate request” message, it will respond with a “**Create PDP context activate response**” message. This message contains information used to establish a GTP-C and GTP-U tunnel on the GGSN. There is also a Tunnel End Identity (TEID and QoS profile. Once this is sent back to the SGSN-MME the GTP-C and GTP-U tunnels are established.

- 7 The final step in the procedure is to reply to the original request toward the MS with a “**PDP context activate accept**”. At this point the MS is considered to be Point-to-point with the external IP network.

WCDMA GPRS PDP Context Activate

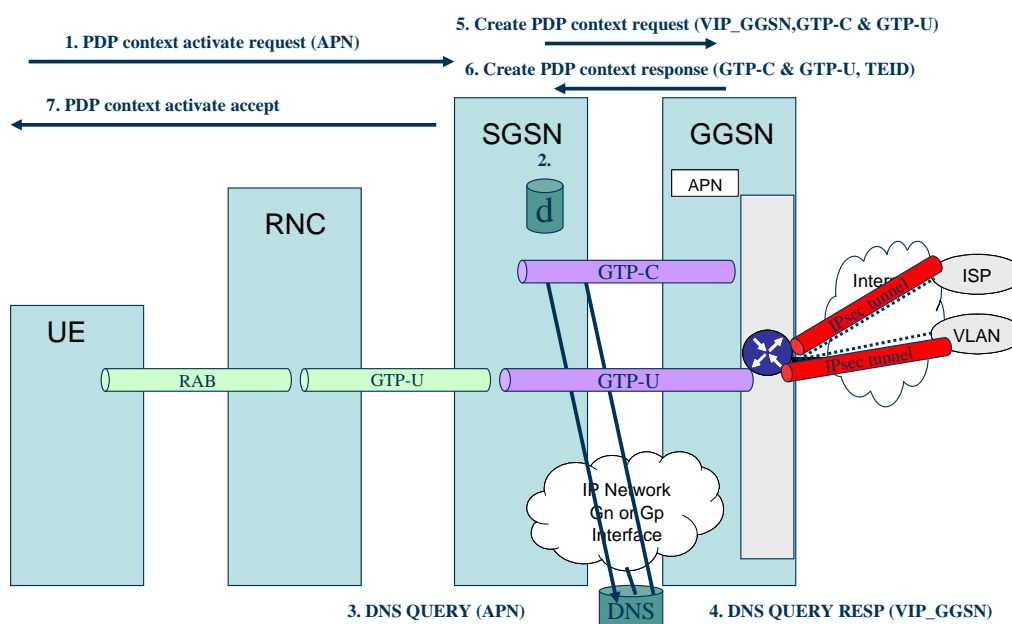


Figure 1-5. WCDMA GPRS PDP Context activation

The below mentioned procedure can only be initiated after the UE has performed a GPRS attach.

- 1 The MS begins by sending a “**PDP context activate request**” message to the SGSN-MME currently serving the MS, the subscriber selects the particular APN (external IP network). Given the list of APN’s that was returned and stored in the VLR of the SGSN-MME, the SGSN-MME will perform the Authentication and security functions, then check the subscribers subscriptions with it’s QoS parameters. The SGSN-MME also modifies the APN by adding the “Default APN Operator ID” to the back.
- 2 The SGSN-MME then checks the APN against the “caching only DNS” on the NCB for a result. If there is a result there then we move to step 5. If not then go to step 3. At this point the APN looks something like this: *www.isp.com.mncXXX.mccXXX.gprs*. The default APN Operator ID is retrieved from the IMSI series analysis table defined in the SGSN-MME.

- 3 The SGSN-MME then constructs a “DNS query” message containing the APN to be resolved. It is important to note that the APN at this stage is the APN plus the Default APN Operator ID. This message is sent across the internal IP network of the operator, to a predefined Domain Name Server (DNS).
- 4 The DNS will look up the APN in its table. The result should equate to a particular VIP_GGSN IP address. The VIP_GGSN is returned to the SGSN in a DNS query result. This result is stored in the caching only DNS of the NCB for the next time an MS uses the same APN.
- 5 The SGSN-MME then constructs a “**Create PDP context activate request**” message. This is sent out toward the GGSN. This message contains information used to establish a GTP-C and GTP-U tunnel on the SGSN-MME. There is also a Tunnel End Identity (TEID), the APN, an NSAPI, MSISDN and QoS profile.
- 6 Once the GGSN has received and processed the “Create PDP context activate request” message, it will respond with a “**Create PDP context activate response**” message. This message contains information used to establish a GTP-C and GTP-U tunnel on the GGSN. There is also a Tunnel End Identity (TEID) and QoS profile. Once this is sent back to the SGSN-MME the GTP-C and GTP-U tunnels are established.
- 7 The final step in the procedure is to reply to the original request toward the UE with a “**PDP context activate accept**”. At this point the MS is considered to be Point-to-point with the external IP network.

Note: For WCDMA GPRS PDP context activation, a GTP-U tunnel between an RNC and an SGSN-MME realizes a Radio Access Bearer (RAB) over the Iu interface.

The RAB carries end-user information between the MS and the SGSN-MME. There can be as many RABs between the MS and the SGSN-MME as there are active PDP contexts for the MS. A RAB is identified by an RAB ID.

RABs over the Iu interface are established, released, and managed by means of the control protocol Radio Access Network Application Protocol (RANAP). The IP address and the TEID are transported as RANAP parameters.

INTER-SGSN ROUTING AREA UPDATE

To enable the MS/UE to upload/download data, GPRS Attach and PDP Context Activation have to be performed. While the MS/UE is in active upload/download session, if the MS/UE moves from one SGSN-MME area to another, the session needs to be served by the SGSN-MME in the new area. This procedure is called inter SGSN-MME routing area update (ISRAU).

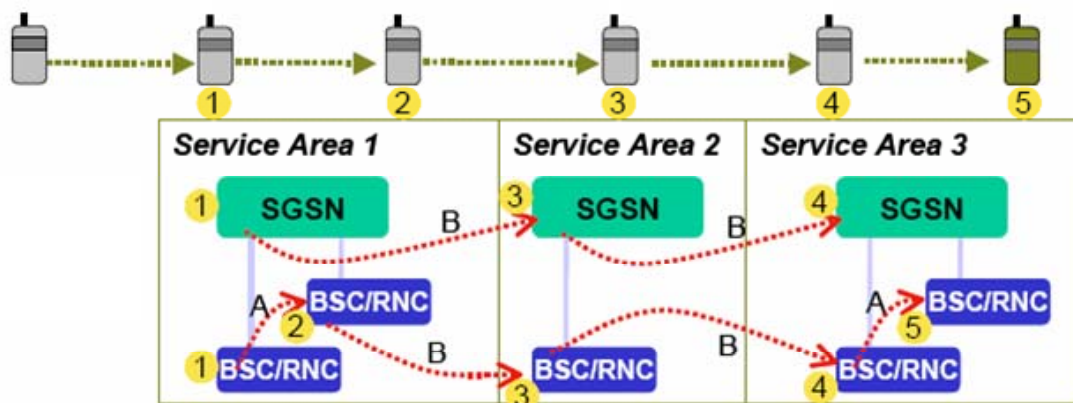


Figure 1-6. Inter SGSN Routing Area Update

- 1 When the MS moves to Service Area 1 (while in active session), the procedure below is performed:
 - a. MS sends the message RA_Update_Request (old RAI, old P_TMSI) to the new SGSN-MME.
 - b. The new SGSN-MME identifies the old SGSN-MME using the old RAI, and asks the old SGSN-MME for the IMSI number.
 - c. The old SGSN-MME sends the contexts (Mobility and possibly PDP
 - d. The new SGSN-MME authenticates the MS.
 - e. The new SGSN-MME sends an Update Location (IMSI, new SGSN-MME address) to the HLR.
 - f. The HLR sends a Cancel_Location to the old SGSN-MME in order to purge information on this MS.
 - g. The HLR sends suBSCriber data to the new SGSN-MME.
 - h. The SGSN-MME informs the MS about some new temporary identities (P_TMSI, TLLI).

2. The MS/UE moves to a new BSC/RNC area that is connected to the same SGSN-MME (arrow A), no ISRAU taking place.
3. Now the MS/UE moves to a new BSC/RNC area (arrow B) that is connected to another SGSN-MME, a new ISRAU have to be performed. The step is the same as step 1.

EPS ARCHITECTURE

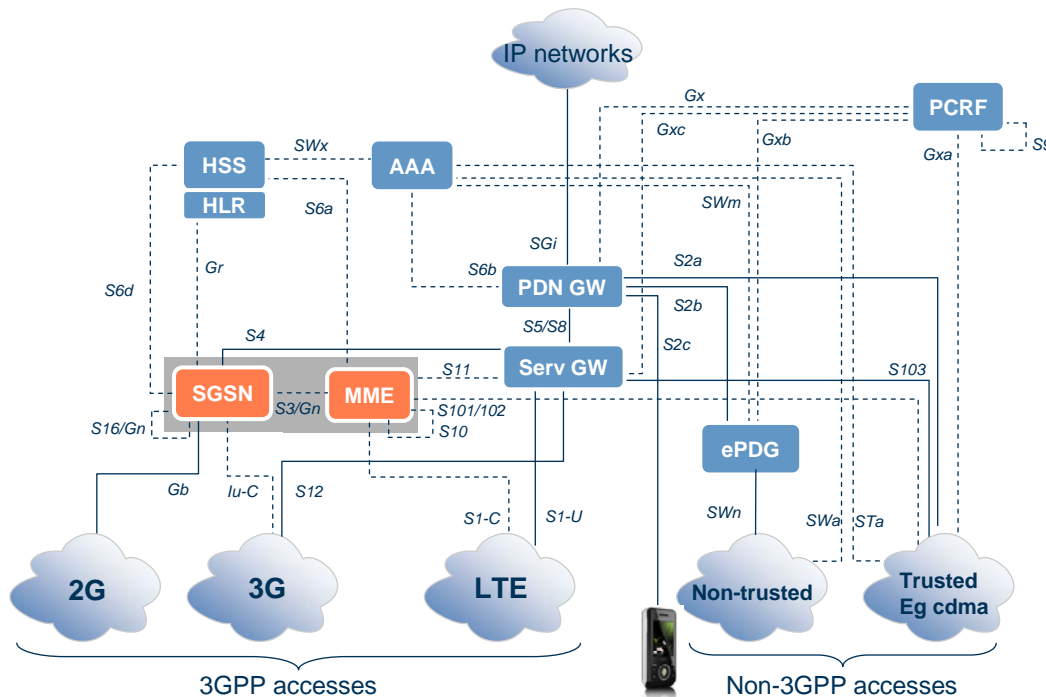


Figure 1-7 LTE/EPC architecture overview.

The Evolved Packet System (EPS) architecture is packet based with no circuit switched part. The voice service logic is being standardized through IP Multimedia System (IMS) and the MultiMedia Telephony (MMTel) services. As bridges to full MMTel services, other network solutions to provide voice services are available in 3GPP.

Core Network Domain

The EPC consists of following network elements:

The Home Subscriber Server (HSS)

The HSS is the master database for an operator. It is the entity containing the subscription-related information to support the network entities actually handling calls/sessions.

A Home Network may contain one or several HSSs: it depends on the number of mobile subscribers, on the capacity of the equipment and on the organisation of the network.

As an example, the HSS provides support to the call control servers in order to complete the routing/roaming procedures by solving authentication, authorisation, naming/addressing resolution, location dependencies, etc.

The HSS is responsible for holding the following user related information:

- User identification, numbering and addressing information
- User security information:
Network access control information for authentication and authorization
- User Location information at inter-system level: the HSS supports the user registration, and stores inter-system location information, etc.
- User profile information.

The HSS also generates User Security information for mutual authentication, communication integrity check and ciphering.

Based on this information, the HSS also is responsible to support the call control and session management entities of the different Domains and Subsystems of the operator.

Mobility Management Entity MME

MME is the control plane entity within EPS, Ericsson solution is the SGSN-MME and will be explained later. It supports the following functions:

Mobility Management:

- Non-Access Stratum (NAS) signalling and security
- Inter CN node signalling for mobility between 3GPP access networks
- Tracking Area list management
- PDN GW and Serving GW selection
- SGSN selection for handovers to 2G or 3G 3GPP access networks
- Roaming
- Authentication

- Bearer management functions including dedicated bearer establishment
- Lawful interception of signalling traffic
- Initiating IMSI detach at EPS detach
- Initiating paging procedure towards eNodeB when MSC pages the UE for CS services
- Supporting SMS procedures for CS fallback.
- Support CS fallback interface and related functions for CDMA access.

Serving GW

The Serving GW is the gateway which terminates the interface towards E-UTRAN.

For each UE associated with the EPS, at a given point of time, there is a single Serving GW.

Some functions of the Serving GW are listed:

- the local Mobility Anchor point for inter-eNodeB handover
- Mobility anchoring for inter-3GPP mobility
- ECM-IDLE mode downlink packet buffering and initiation of network triggered service request procedure
- Lawful interception
- Packet routing and forwarding
- Transport level packet marking in the uplink and the downlink
- Accounting on user and QCI granularity for inter-operator charging
- A local non-3GPP anchor for the case of roaming when the non-3GPP IP accesses connected to the VPLMN
- Event reporting (change of RAT, etc.) to the PCRF
- Uplink and downlink bearer binding towards 3GPP accesses
- Uplink bearer binding verification with packet dropping of "misbehaving UL traffic"

Packet Data Network Gateway PDN GW

The PDN GW is the gateway which terminates the SGi interface towards the PDN.

The P GW provides PDN connectivity to both GERAN/UTRAN only UEs and E UTRAN capable UEs using any of E UTRAN, GERAN or UTRAN. The P GW provides PDN connectivity to E UTRAN capable UEs using E UTRAN only over the S5/S8 interface.

Some of PDN GW functions include:

- Per-user based packet filtering (by e.g. deep packet inspection)
- Lawful interception
- UE IP address allocation
- Transport level packet marking in the uplink and downlink, e.g. setting the DiffServ Code Point, based on the QCI of the associated EPS bearer
- UL and DL service level charging, gating control, rate enforcement
- UL and DL rate enforcement based on APN-AMBR
- DL rate enforcement based on the accumulated MBRs of the aggregate of SDFs with the same GBR QCI (e.g. by rate policing/shaping)
- DHCPv4 (server and client) and DHCPv6 (client and server) functions
- Additionally the PDN GW includes the following functions for the GTP-based S5/S8
- UL and DL bearer binding
- UL bearer binding verification

The PDN GW functions also includes user plane anchor for mobility between 3GPP access and non-3GPP access.

Radio Access Domain

Radio Access Domain is referred as Evolved UTRAN and consists only of a number of eNodeBs (eNB) that are interconnected via X2 interface through an IP network. The eNBs are connected by means of the S1 interface to the EPC, more specifically to the MME by means of the S1-MME and to the S-GW by means of the S1-U interface. The S1 interface supports a many-to-many relation between MMEs / Serving Gateways and eNBs.

E-UTRAN Node B - eNB

An eNB is a logical network component which serves one or more E-UTRAN Cells. It is responsible for radio transmission and reception from/to UE. Some of eNodeB functionalities are:

- Cell Control and MME pool support:
- Control and User Plane Security
- Segmentation/Concatenation
- Error Correction
- Shared Channel Handling
- Scheduling
- Physical Layer functions such as channel coding, modulation, filtering
- Handling of Measurement Control/Reporting
- Mobility Control

An eNB can support FDD mode, TDD mode or dual mode operation

USER EQUIPMENT (UE)

The User Equipment allows a user access to network services. For the purpose of 3GPP specifications the interface between the UE and the network is the radio interface.

A User Equipment can be subdivided into a number of domains, the domains being separated by reference points. Currently the User Equipment is subdivided into the Universal Integrated Circuit Card (UICC) domain and the Mobile Equipment (ME) Domain. The ME Domain can further be subdivided into one or more Mobile Termination (MT) and Terminal Equipment (TE) components showing the connectivity between multiple functional groups

Ericsson EPC Nodes

Mobility Management Entity

Ericsson introduces the Mobility Management Entity (MME) within the Ericsson SAE/EPC R1 Solution (2009) to be evolved 2010 and beyond. The MME is the logical node serving users attaching over LTE access with functionality in the areas of session and bearer management, mobility management and security including authentication.

It is referred to as the SGSN-MME product since it is a further evolution of the Ericsson SGSN, in service worldwide with a considerable market footprint (in service in more than 250 operator networks). This means that the MME implementation reuses hardware and software architecture and all common software functionality, while adding the MME specific functionality. The first commercial version including MME functionality is named SGSN-MME 2009B, where “2009” indicates the targeted release year and where “B” means the second half-year (accordingly, the “A” suffix means first half-year).

The SGSN-MME 2010B will be followed by SGSN-MME 2011 A/B and so on.

MME Interfaces

The diagram below shows the MME interfaces towards the EPC core network and LTE nodes. The interworking interfaces to GSM/WCDMA and CDMA 2000 is also shown.

Inter-face	Network elements	Main Purpose	3GPP 2G/3G analogy	3GPP TS
S1-MME (S1-C)	E-UTRAN - MME	<ul style="list-style-type: none"> Control plane protocol E-UTRAN - MME Control Plane protocol UE – MME including NAS^{*)} protocol for mobility management, user plane bearer activation, modification, deactivation, ciphering and integrity protection. 	lu-C, Gb	23.401
S3	SGSN – MME	Inter CN node signaling for mobility between 3GPP access networks (3GPP Rel-8)	-	23.401
S4	SGSN - Serving GW	Serving Gateway user plane access to the 3GPP WCDMA/GSM access via SGSN	-	23.401
Gn/Gp	SGSN - MME and SGSN – S/PDN GW	Inter CN node signaling for mobility between 3GPP access networks (3GPP pre-Rel-8 SGSN)	-	23.401
S6a	MME - HSS.	Transfer of subscription and authentication data for authenticating/authorizing user access using the AAA system. Evolved DIAMETER protocol.	Gr	23.401
S10	MME - MME	MME relocation (move of context between MMEs) and MME to MME information transfer	Gn/Gp	23.401
S11	MME - Serving GW	Control plane protocol MME – Serving GW	Gn/Gp	23.401
S101	MME - HRPD AN	Pre-Registration, Session Maintenance and Active handovers between E-UTRAN and HRPD (CDMA2000) networks.	-	23.402

^{*)}NAS = Non Access Stratum

Figure 1-8 Prime MME related interfaces.

The SGSN-MME 2010B supports the following MME interfaces:

- S1-MME Interface

The S1-MME interface connects the MME to eNodeBs and UE. The S1-MME interface is based on SCTP. S1 Application Protocol (S1-AP) messages are transferred between the MME and eNodeB, and NAS messages are transferred between the MME and UE.

- S11 Interface

The S11 interface connects the MME to the Serving Gateway. The S11 interface is an IP-based interface used for EPC signaling between the MME and the Serving Gateway.

- S6a Interface

The S6a interface connects the MME to the HSS. It enables transfer of subscription and authentication data for authenticating and authorizing user access. The S6a interface is based on the Diameter protocol.

- Gom Interface

The Gom interface connects the SGSN-MME to Operation & Maintenance (O&M) equipment in the O&M network, making it possible for an operator to communicate with the SGSN-MME.

- S3 Interface

The S3 interface connects the SGSN-MME to the 3GPP SGSN (2G and 3G). This is to allow interworking between LTE and 2G/3G.

- S4 Interface

S4 interface connects the SGSN to the SGW. It enables the mobility between LTE and 2G/3G.

- Gn/Gp Interface

The Gn/GP interface allows Inter CN node signaling for mobility between 3GPP access networks (3GPP pre-Rel-8 SGSN)

- S101 Interface

The S101 interface connects the MME to the HRPD to enable interworking between LTE and CDMA2000 networks.

ARCHITECTURE

SGSN-MME SYSTEM SOFTWARE ARCHITECTURE

The software structure of the SGSN-MME node is divided into two main areas, the Wireless Packet Platform (WPP) and the GPRS application areas respectively. The GPRS application areas are divided into the SGSN application components i.e. GSM, WCDMA and LTE and also the Common application component.

The WPP consists of Ericsson's MkIV, MkV, MkVI and MKVI+ hardware, including backplane switch, interfaces and processors. It also includes software components such as operating systems (Solaris Linux and VxWorks), high-availability support middleware, and Operation and Maintenance support.

As shown in the figure below, the WPP is made up of Distributed Processing Environment Software (DPE), middleware and standard industry processing hardware.

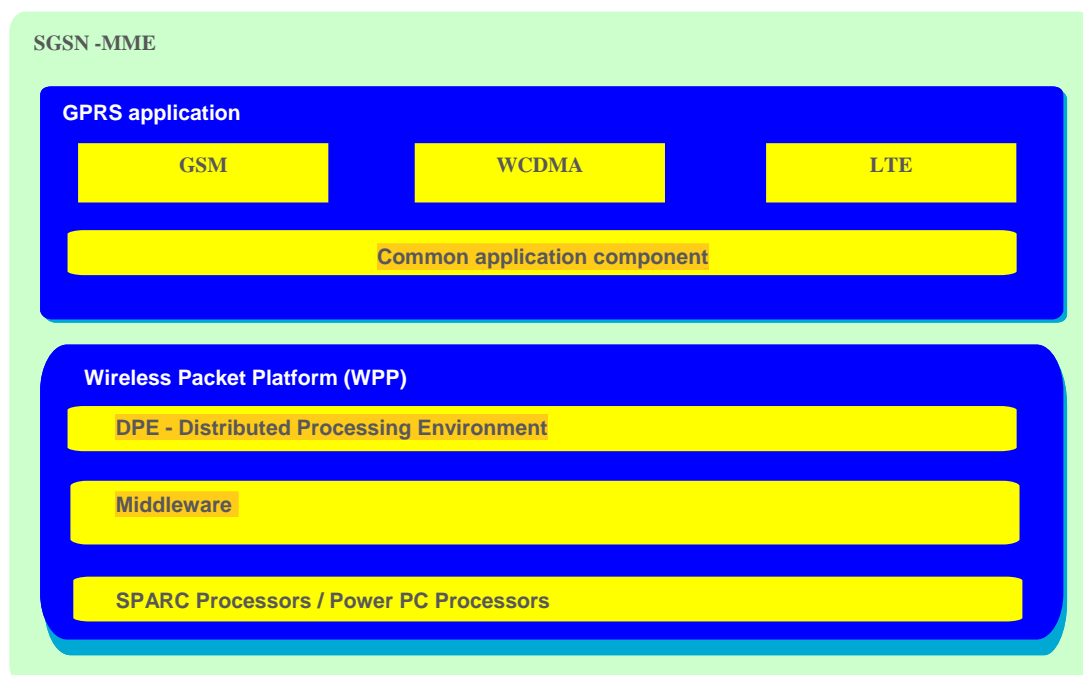


Figure 1-9 SGSN-MME System Software Structure.

There are several types of software running on a SGSN-MME, of which the following are examples:

- Operating System (OS) – This is the software that provides services and functionality on a Plug In Unit (PIU). There are different OSs depending on which role a PIU has.
 - Solaris runs on the Application Processors (APs) and Node Controller Boards (NCBs) in MkIV hardware.
 - Linux runs on APs FSBs, NCBs, Power Ethernet Boards (PEBs) and IBEN used in MkV, MkVI and MKVI+ hardware.
 - VxWorks runs on Device Processors (DPs) and PIUs with SS7 Front Ends (FEs), SS7 Back Ends (BEs), or routers. Also, the PEBv3 used in MkIV hardware.
- Distributed Processing Environment (DPE) – This is software that distributes the execution of the SGSN-MME application software to the different PIUs, as well as the coordination among them. The SGSN-MME is a multi-computer system so it needs a special environment to co-ordinate GPRS (for both GSM, WCDMA and LTE systems) activities across this system. Each application executes locally on a local processor and its operating system. The processors are connected and function as loosely coupled processors. To create one system, all those processors are held together by means of the DPE software. The DPE supports redundancy and the distribution of functions, detects application failure, and can activate redundant applications in different ways.
- SGSN-MME application software – This is the software that manages traffic in an SGSN-MME.

SGSN-MME GPRS Application Software

The GPRS application software is software that is specific for GPRS functionality. It includes GPRS protocols, session and mobility management as well as node operation and maintenance functionality.

Examples of common GPRS SGSN-MME Application Components are: Charging, Configuration Management, Network Element Control and Distribution, Session Management and payload forwarding of GPRS specific protocols e.g.; GTP tunneling.

Examples of SGSN-MME Application components are: Visitor Location Register and Mobility Management.

SGSN-MME Middleware

The Ericsson middleware is a generic system for fault tolerant real time applications and provides a platform and a set of tools for easy and accurate generation of datacom/telecom applications.

The middleware is built on standard industry components using open standards such as UNIX, C/C++, Erlang, Java, Common Object Request Broker Architecture (CORBA) interface, Hyper Text Transfer Protocol (HTTP) and Simple Network Management Protocol (SNMP).

The C programming language is a structured, procedural programming language that is widely used both for operating systems and applications. Many versions of UNIX-based operating systems are written in C. C is standardized as part of the Portable Operating System Interface (POSIX).

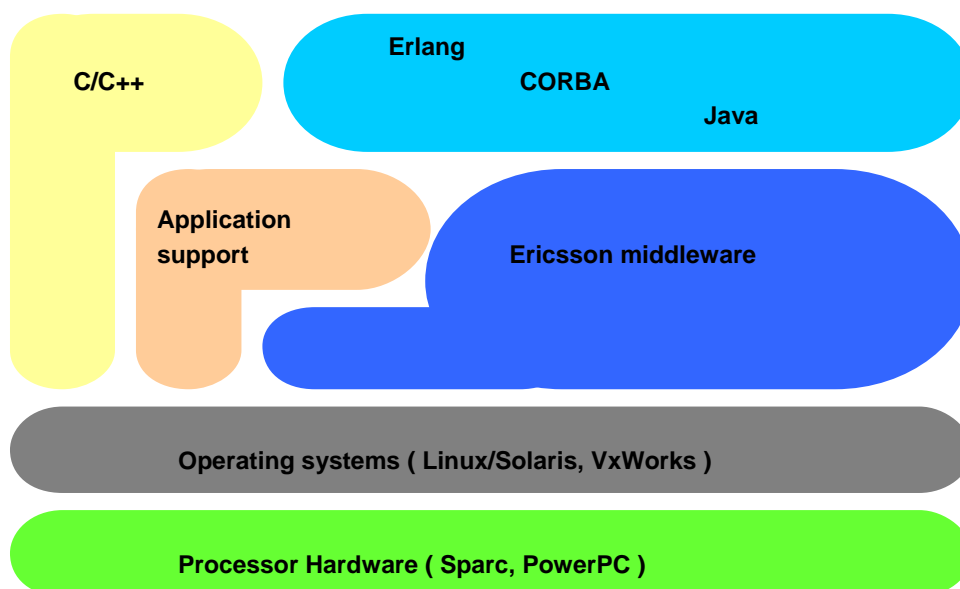


Figure 1-10 SGSN-MME Middleware architecture.

The C++ programming language is an object-oriented programming language, also widely used both for operating systems and applications. C++ is supported on both VxWorks and Solaris.

The Erlang programming language is designed for development of robust systems of programs that can be distributed among different computers in a network. It supports development of applications with fault tolerance, distributed processing, a large amount of concurrent activity, real time response times in milliseconds, and non-stop operation. The language was developed by the Ericsson Computer Sciences Lab to build software for telecommunication products.

The middleware provides a distributed real time database as well as a number of standard operations support, configuration management, and fault handling functions. These support functions includes features such as:

- Process replication
- Process take-over
- Process supervision
- Load sharing
- Events and alarms
- CORBA IIOP (Internet Inter -ORB Protocol)
- Node element and processor restart and stop (different Logs levels)
- SNMP
- HTTP server
- Equipment load and start
- Equipment detection and supervision
- Real time database
- In-service backup
- Software management
- Performance measurement
- Security, IPSec (IP security), GRE (Generic Routing Encapsulation), DHCP Client and RADIUS Client.
- Routing – Open Shortest Path First (OSPFv2), Routing Information Protocol (RIPv2) and Border Gateway Protocol (BGP-4)
- Caching-only DNS

SGSN-MME LOGICAL ARCHITECTURE FOR GSM AND WCDMA

The SGSN-MME logical architecture for GSM and WCDMA systems makes a distinct separation between transmission and control processing. The control system and transmission system are interconnected through the backplane as illustrated in the figure below. The applications consist of software components, or more specifically, devices and device support functions where a device is an abstraction of a software package executing on a DP. The devices and device support functions are mainly implemented for the transmission system, while the control system primarily handles signaling traffic and high-level protocols. Users payload is processed in the transmission system. Both signaling traffic (Control System) and payload traffic (Transmission System) will use the router boards (Transmission System) to transfer and route their traffic to other nodes in the network.

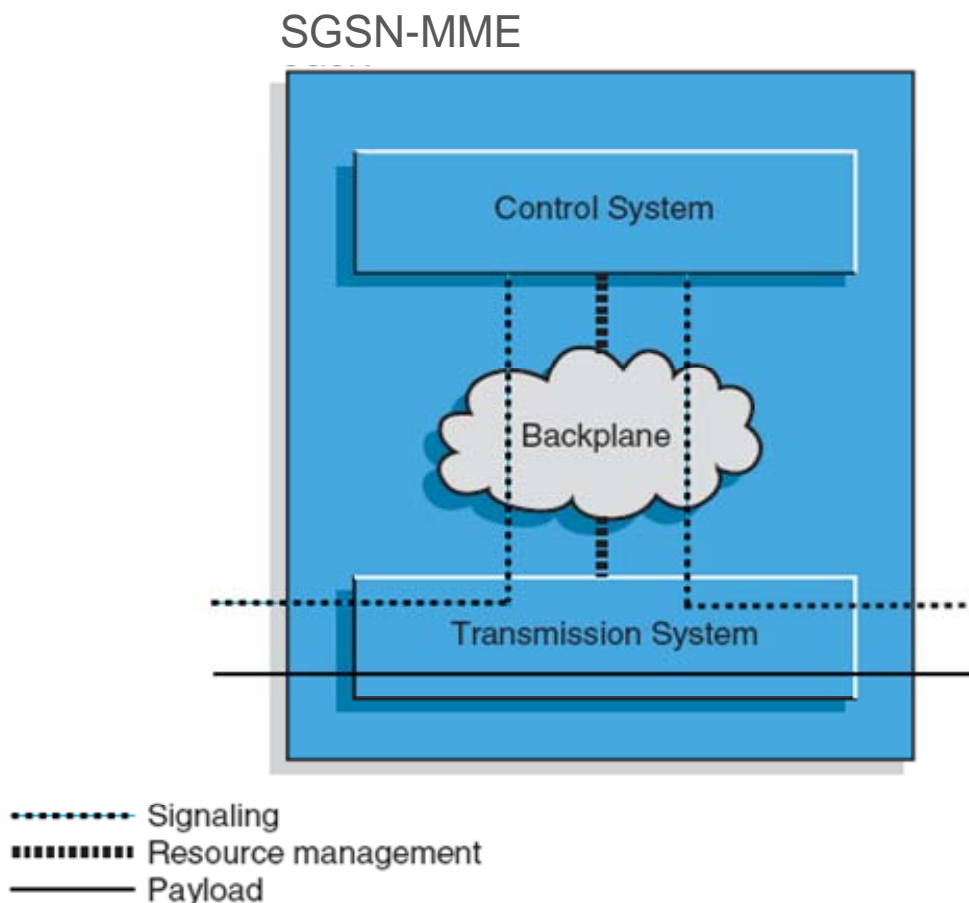


Figure 1-11: SGSN-MME logical architecture

Control System Functions

The control system is designed to process high-level protocols and to control payload routing in the transmission system. It handles traffic control activities, such as mobility and session management and high-level protocol processing; system-internal control activities, such as recovery, distribution, and O&M; and adaptation activities, such as transmission system drivers.

The control system consists of a number of Application Processors (APs) interconnected through the backplane.

The central AP (AP/C) handles the central functions of the control system and runs on the PIU that serves as active NCB. The remaining AP PIUs host one local AP each and form a generic processing pool in order to provide shared load handling.

SGSN-MME HARDWARE

HARDWARE CHARACTERISTICS

This section describes the hardware in the SGSN-MME for GSM, WCDMA and LTE Systems.

The SGSN-MME can operate on the MkIV, MkV, MkVI and MkVI+ hardware configurations, whereby the SGSN-MME for LTE access will only be supported on MkVI and MkVI+. The SGSN-MME hardware incorporates redundancy, hot swap capabilities, two-step high-ohmic distribution of power, and worldwide telecom approvals (such as electromagnetic compatibility, earthquake protection, and safety features). Ericsson-standard building practices are applied for easy installation and maintenance.

HARDWARE LAYOUT

The SGSN-MME hardware consists of a cabinet housing internal cabling, Power Distribution Units (PDUs), fan units, and two or three magazines containing various Plug-In Units (PIUs).

SGSN-MME Cabinet – BYB501 Mechanics

The physical enclosure or cabinet is as according to standard Ericsson BYB 501 building practice, with two or three magazines, including internal cabling, power distribution, fans and cable shelves with air inlets. The layout of the SGSN-MME cabinet is illustrated in the figure below.

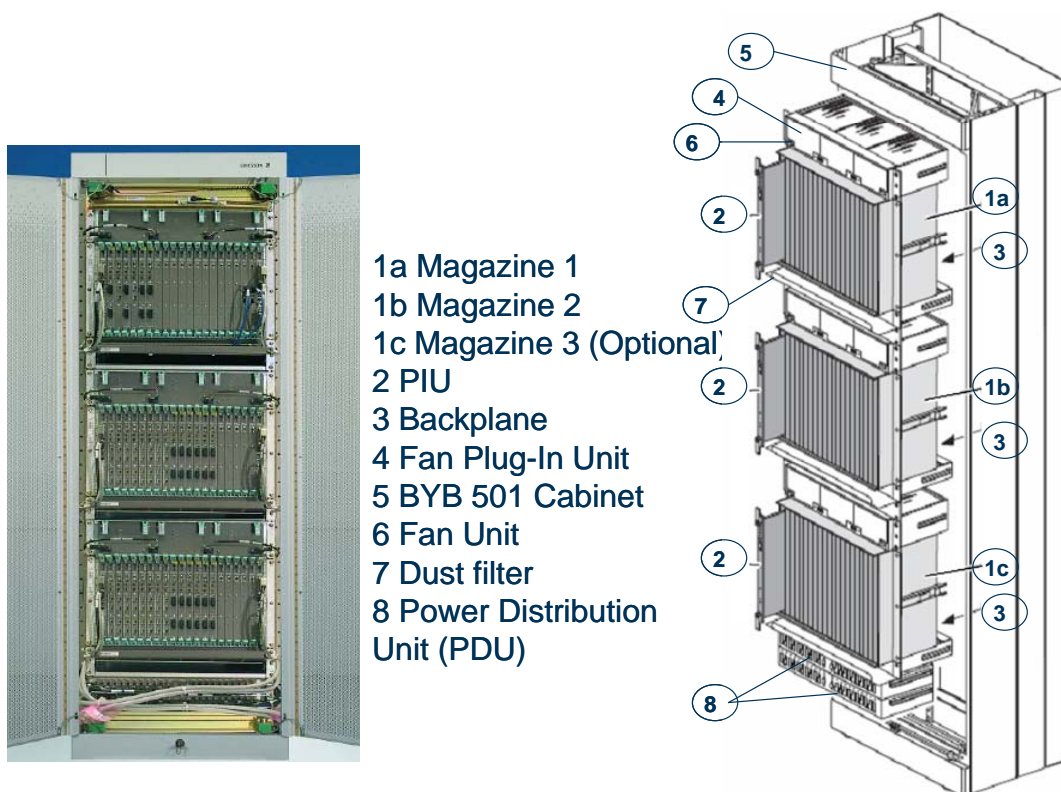


Figure 1-12. SGSN-MME BYB501 cabinet

Note: Two types of Power Distribution Units (PDU) can be installed:

- Power Distribution Unit (PDU) 20x15A (BMG 663 014/31)
- Power Distribution Unit (PDU) 5x10A (BMG 663 011/11)

SGSN-MME Magazines

The SGSN-MME cabinet houses two or, optionally, three magazines interconnected via 1 Gbit Ethernet connection when using PEBv3 or 10 Gbit Ethernet when using PEBv4/v5 (10GBASE-CX4). The magazine frame has guides for easy insertion and withdrawal of PIUs. This is further facilitated by latches on the PIU upper front edges. Each magazine can hold 21 (19+2) slots for PIUs, with a slot size of 20mm.

The PIUs are interconnected by a dual 1 Gbit Ethernet interconnection on the backplane. The magazine backplane also handles dual redundant power distribution.

Backplane

The backplane of a magazine provides dual redundant power distribution as well as dual redundant Ethernet connections to all slots in the magazine.

The internal communication between PIUs over the backplane is depicted in the two figures below.

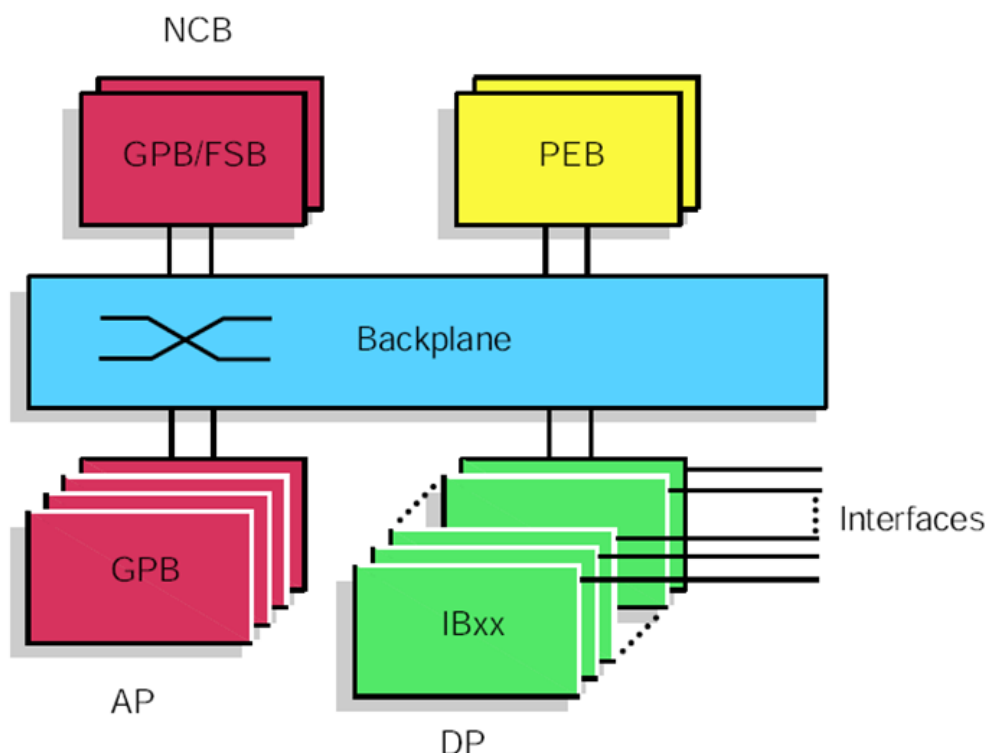


Figure 1-13. Hardware architecture, MkIV

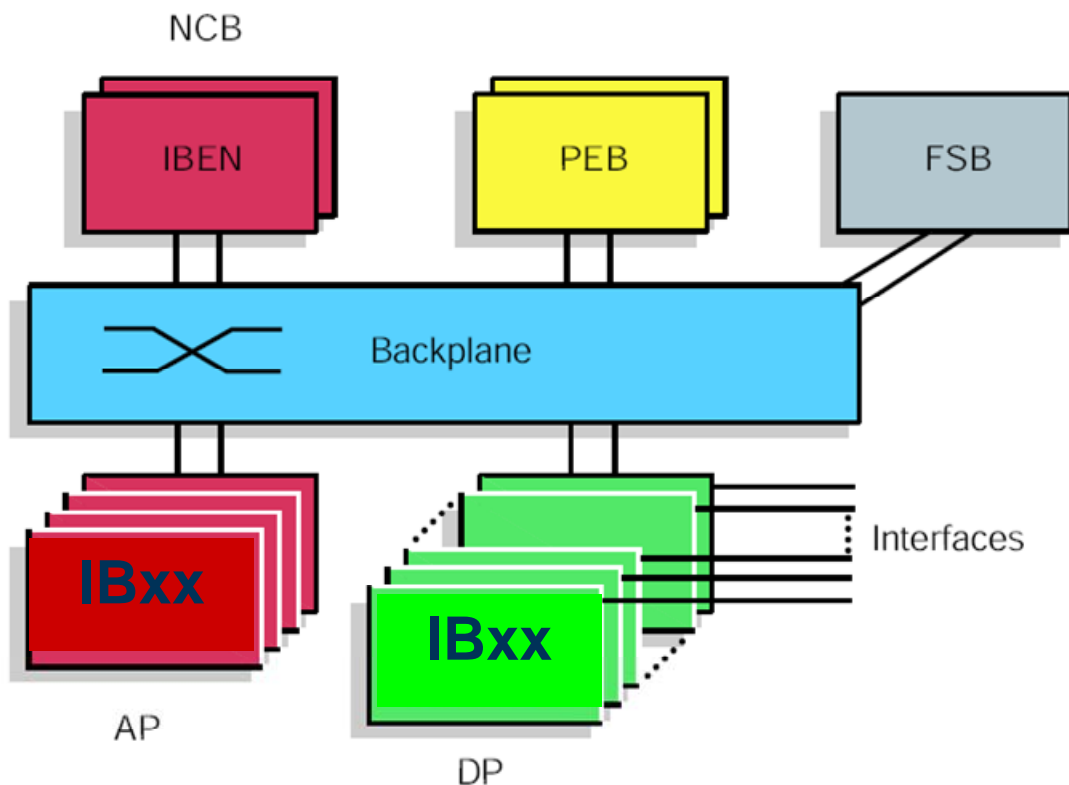


Figure 1-14. Hardware architecture, MkV, MkVI and MkVI+

Roles

Some PIUs can assume different roles depending on position and configuration. This section describes the different roles the PIUs can have.

Using TMO, available on MkVI hardware and newer, the roles of Application Processors (APs) and Device Processors (DPs) can be redeployed to optimize the SGSN-MME capacity for some of the PIUs. For example, in order to increase the signaling capacity a TMO step is performed. A predefined number of PIUs is then redeployed from DPs to APs.

- Application Processor (AP) – The AP runs the local packet data application. AP PIUs are used for node management, processing, and signaling.
- Device Processor (DP) – The DP handles payload processing and Signaling System No. 7 (SS7) signaling.
- IP Router – The Internet Protocol (IP) routers determine the optimal routing paths between MSs and GGSNs and handle transport of packets through the networks, Operation and Maintenance (O&M) traffic, and SS7 over IP.

- SS7 Front end (SS7 FE) – The SS7 front end represents the low-level protocols of the SS7 stack, distributing incoming traffic to the SS7 back ends or Network Management Module (NMM). SS7 traffic over narrowband and broadband is handled by the narrowband SS7 front end and the broadband SS7 front end, respectively. SS7 over IP is handled by the SCTP front end.
- Node Controller Board (NCB) – The active Node Controller Board (NCB) provides central support and functions, such as O&M, Hardware and Software monitoring and distribution, Packet Data (PD) application and SGSN supervision. It is referred to AP/C also. In the SGSN, one PIU serves as the active NCB. A second PIU serves as a passive NCB replicating the node control. If the active NCB fails or needs replacement, the passive NCB takes over operation.
- File and Boot Server (FSB) – The File and Boot Server provides disk storage and boot services in the SGSN, MkV and MkVI.

- Application Processor (AP)
 - used for node management, processing, and signaling.
- Device Processor (DP)
 - handles payload processing and SS7 signaling.
- IP Router
 - determines the optimal routing paths between MSs and GGSNs and handle transport of packets through the networks, O&M traffic, and SS7 over IP.
- SS7 Front end (SS7 FE)
 - represents the low-level protocols of the SS7 stack, distributing incoming traffic to the SS7 back ends or Network Management Module (NMM).
 - narrowband SS7 FE, broadband SS7 FE and SCTP FE.

Figure 1-15. PIU roles

- › Node Controller Board (NCB)
 - provides central support and functions, such as O&M, Hardware and Software monitoring and distribution, Packet Data application and SGSN supervision.
 - Active NCB & passive NCB
- › File and Boot Server (FSB)
 - provides disk storage and boot services in the SGSN, from MkV to MkVI+.
 - Primary FSB & Secondary FSB

Figure 1-16. PIU roles

Plug-in Units – PIUs

A Plug-in Unit can also be referred to as a SGSN-MME board. A PIU can carry different processor modules and interfaces. A standard board size is 265x225 mm.

The SGSN-MME contains several types of PIUs, see the figure below. The magazine identification is set on the two Power and Ethernet Boards (PEBs) in the magazine supervised by all PIUs. A mismatch between the settings in the two PEBs in the magazine generates an alarm. In addition, each PIU handles hardware alarms, which for example can be raised for excessive temperatures or voltages, or for a power drop-out of one of the redundant branches.

There are 4 main types of PIU in the SGSN-MME nodes:

- Power and Ethernet Board (**PEB**) – The PEB supplies the magazine with power, which is distributed from power connectors on the PEB front panel, through the backplane, to all slots. The PEB supplies the backplane communication with a dual star Ethernet network. The front panel has external ports that are used for magazine interconnection. PEBv5 in MKVI+ can now be used as router boards.
- General Processing Board (**GPB**) – The General Processing Board (GPB) acts as an AP or an NCB, and is used in MkIV hardware.
- File Server Board (**FSB**) – The File Server Board (FSB) acts as a File and Boot Server. In the SGSN-MME, one FSB serves as the master FSB. A second FSB serves as a standby FSB mirroring all stored data. If the master FSB fails or needs replacement, the spare FSB takes over operation. **Note:** The FSBv3 can sometimes be used as NCBs, but this co-located

functionality is only used in some previous SGSN-MME releases (in MkIV).

- **Interface Boards (IBxx)** – The Interface Board version 3 (IBxxv3) PIUs can act as DPs handling payload processing, or handle IP routing and SS7 signaling. The IBxxv4 PIUs can, apart from the IBxxv3 roles, also act as APs. These boards have physical ports. They are listed below:
 - **IBAS** for ATM Single-mode fiber or Ethernet (Copper)
 - **IBAC** for ATM or Ethernet (Optical). New in MkVI+.
 - **IBTE** for E1/T1
 - **IBS7** for narrow-band SS7
 - **IBEN** for Ethernet

PIU	MkVI+	MkVI	MkV	MkIV
Power and Ethernet Board – PEB	v5	v4	v3	v3
General Processing Board – GPB	-	-	-	v3
File Server Board – FSB	v4	v4	v4	v3
Interface board Ethernet - IBEN	v4	v4	v4	-
Interface board E1/T1 - IBTE	v4	v4	v3	v3
Interface board Narrowband SS7 - IBS7	v4	v4	v3	v3
Interface board ATM single Mode Fibre-optic – IBAS	-	v4	v3	v3
Interface Board for ATM with Ethernet Media Converter – IBAC	v4	-	-	-

Figure 1-17. SGSN Plug-in Units (PIU)

PEBv4 Overview

PEBv4 has the following functions:

- Power distribution (4 branches of 250 W each).
- Supervision circuitry for primary 48 V voltage levels.
- 22 port Gigabit Ethernet switch.
- Two 1 Gbps Ethernet interfaces for maintenance usage only.
- Three 10 Gbps Ethernet copper interfaces (10GBASE-CX4) for connections between magazines.

- 19 port 1 Gbps Ethernet interfaces (10/100/1000 Base-T) for connections between PIUs.
- Configuration switch for Mag ID selection.
- Alarm handling for fans (fan unit powered directly from Power Distribution Unit (PDU)).
- Temperature sensor indicating if the temperature is too high or too low. The sensor cut-off the local power supply if the temperature is too high.
- Alarm indicating if the supply voltage is too high or too low.

PEBv5 Overview

PEBv5 has same functions as PEBv4 but higher throughput. The only difference is:

- One optical interface (10Gbit SFP+) for connection to external interfaces. Whereby the PEBv4 uses (10GBASE-CX4).
- With functionality, the PEBv5 can now also act as an IP router board.

See below architecture diagram of the PEBv5. The only difference is the 10Gbit SFP interface.

GPBv3 Overview

The GPBv3 handles the administrative tasks in the GSN MK IV only. The GPBv3 is also used for data and software storage.

The GPBv3 handles services such as attach/detach, PDP context, SMS and subscriber mobility. It's the signaling board.

The GPBv3 is equipped with a 650 MHz Ultra SPARC processor, a 40 GB EIDE hard disk and 1 GB RAM.

FSBv4 Overview

The FSBv4 is the only board within the system that contains a hard disk in MKV/VI/VI+. It handles boot of other boards and shared disk input-output (IO) services for other boards. The disk storage within an FSBv4 provides a secure redundant storage if the FSBv4 is in pair with another FSB.

The FSBv4 can have the following roles:

- Primary FSB. The primary FSB is the master central processing unit (CPU) that receives and executes shared disk IO operations within the SGSN-MME. All data stored on the primary FSB is mirrored to the secondary FSB directly when receiving IO operations.
- Secondary FSB. This is a hot standby to the primary FSB. The secondary FSB will take over if the primary FSB fails.

The FSBv4 is equipped with an 1131 MHz PowerPC processor, 73 GB SAS hard disk and 1 GB RAM.

IBENv4 Overview

The IBENv4 can have different functionality depending on the Traffic Mix Optimization (TMO) configuration and Serving GPRS Support Node (SGSN) deployment. The IBENv4 can either have the functionality of an Application Processor (AP) or a Device Processor (DP).

IBTEv4 Overview

The IBTEv4 provides eight E1/T1 interfaces and supports Frame Relay. Using Frame Relay it can also function as a DP. If Frame Relay is not used it can function as either an AP or a DP depending on the TMO configuration.

IBS7v4 Overview

The IBS7v4 provides interfaces for Signaling System no. 7 (SS7) traffic over E1 or T1. It supports narrowband SS7 protocol on 64 timeslots.

IBACv4 Overview

The IBACv4 provides one short-haul fiber interface for ATM. It supports traffic processing and also SS7 over ATM.

The IBACv4 has the same functionality as the IBASv4 (see after IBAC diagram), with the difference that the IBACv4 board has additional functionality in terms of a media converter. The converter module has two 1Gbps Ethernet electrical ports that convert to optical ports, 1000BASE-SX or 1000BASE-LX.

The IBAS is also used for Ethernet connections and IP routing functionality. Therefore, as well as having two ports for ATM connections it also has two ports for Ethernet connections.

The computing boards are based on Crane Boards. A Crane Board denotes a board that can carry different processor modules and interfaces by 'piggybacking'. The different computing modules and interfaces on the board are interconnected by means of an internal bus called the Peripheral Component Interconnection (PCI) bus.

Host Naming Convention

The figure below illustrates a SGSN-MME magazine configuration, where there are 3 magazines, each magazine consisting of 21 slots. This figure also details the host naming convention used so the active NCB can address certain processors on each of the boards. The host file is found on the active NCB under */etc/hosts*. The main processor on the active NCB board can be addressed with 'eqm01s14p2'; that is eqm01 for the first magazine, s14 is the hexadecimal encoding for slot 20, and p2 is the main processor of this board. In the example below, the main processor on the specified GPBv3 can be addressed with 'eqm01s0bp2'.

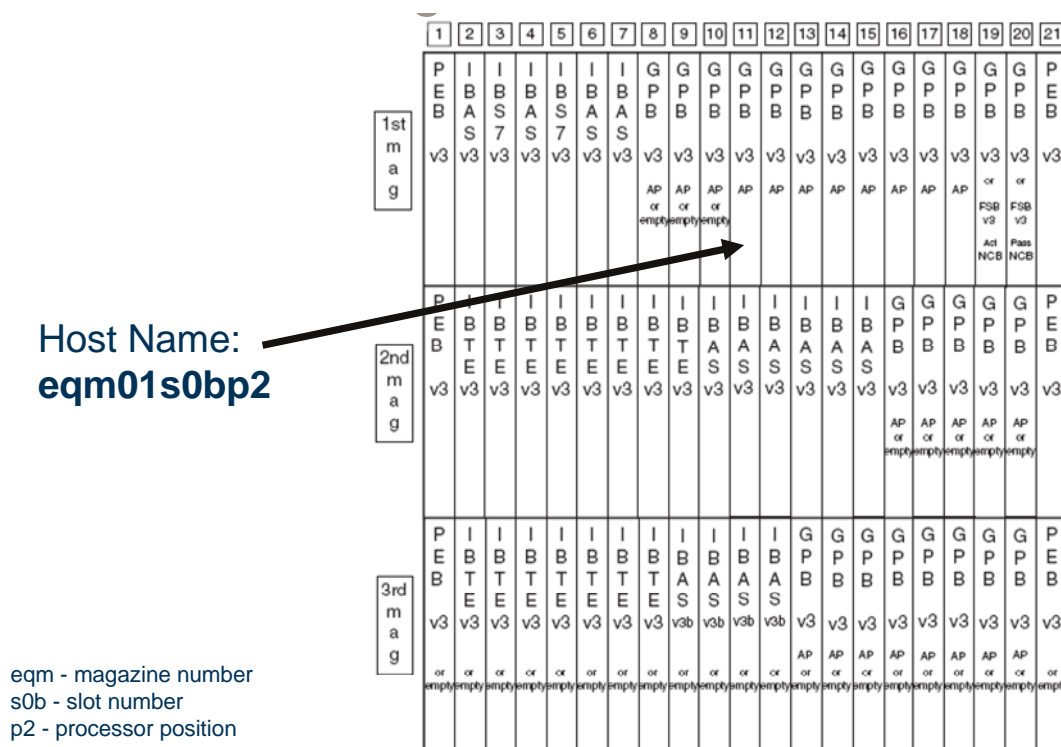


Figure 1-18. PIU host naming convention

Each board can read its hardware address on the back plane. It can also read the magazine ID from backplane. Magazine ID is set in each PEB via a rotary switch, and is distributed individually to each board. The PEB monitors the magazine ID of the other PEB's, and if a mismatch occurs, the 'red' LED on the PEB front panel is lit. This generates a MagID mismatch fault.

CONNECTIVITY

SGSN-MME IP CONNECTIVITY

The traffic is separated into Virtual Private Networks (VPNs) to improve security, capacity, and Quality of Service (QoS), and to facilitate operation. Each VPN corresponds to a logical IP network. VPNs do not require the physical separation of router PIUs; thus, these VPNs can share one or more router PIUs and the corresponding physical interfaces.

The SS7-over-IP services run on the Stream Control Transmission Protocol (SCTP), which contains a faster redundancy mechanism than the Open Shortest Path First (OSPF) protocol, but it does not contain any loadsharing. To attain redundancy, two VPNs are required for SS7-over-IP services over the core network, and two VPNs are required for SS7-over-IP services over Iu-C.

Each VPN consists of an external network, a set of routers with external interfaces, a set of application PIUs executing the services connected to the VPN, and separate routing tables. Several VPNs share the physical interfaces on a router PIU by using Virtual Local Area Networks (VLANs).

Two examples of the distribution of VPNs over the router PIUs in a Dual Access SGSN-MME WG and in a SGSN-MME L (LTE) are shown below.

SGSN-MME WG with all IP connectivity:

1. Gb/IP (Gb-VPN)
2. Iu-C/IP (IuC-VPN)
3. Iu-U/IP (IuU-VPN)
4. CN/IP (CN-VPN)
5. Gn/IP (Gn-VPN)
6. Gom/IP (Gom-VPN)

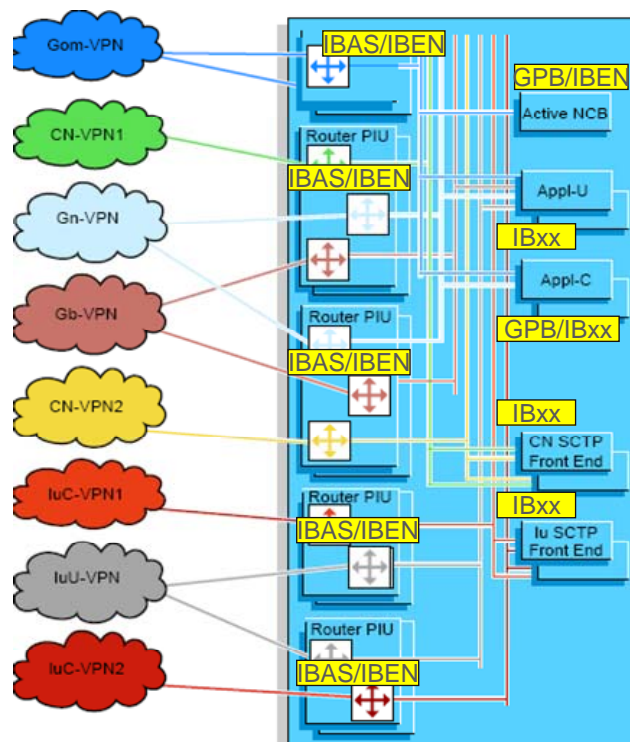


Figure 1-19. SGSN-MME WG PIUs and VPNs

SGSN-MME IP ADDRESSES

As previously described, the SGSN-MME is a multi computer system. This means that several processes on different PIUs can communicate to each other over the internal backplane provided by the PEBs. Each process with its own specific function.

The SGSN-MME requires three types of IP addresses:

- **IP address for node internal communication**
- **IP addresses for node external addresses, these are referred to as node edge interface addresses or transport IP address**
- **Service IP addresses**

Figure 1-20. SGSN-MME IP addresses

We saw in one of the previous figures the connection and communication flow between the internal SGSN-MME computing entities or PIU's. As shown, these PIUs are connected over the node internal (Ethernet) switch. Since IP is the communication protocol, IP addressing must be assigned to all the computing entities.

The Node internal IP addresses are used to distribute internal traffic around the SGSN-MME node, internal operations. This type of traffic is never seen outside of the SGSN-MME node.

Internal IP addresses use : 169.254.0.0/17

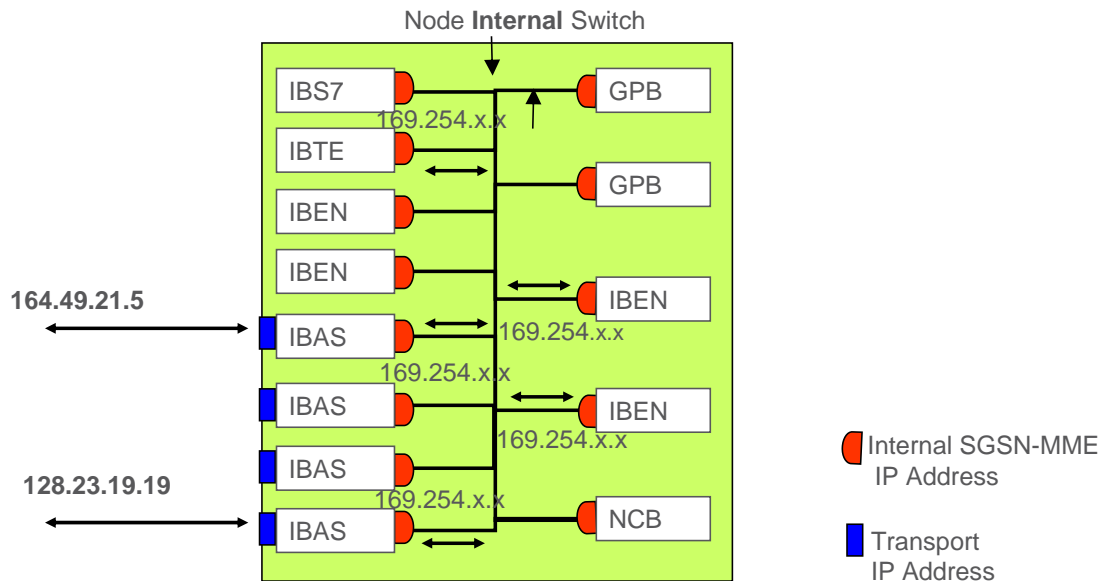


Figure 1-21. Internal and Transport IP addresses

The Transport IP addresses are usually used as Gateway IP addresses to direct traffic through specific interfaces. Explained in detail on page 58.

Internal SGSN-MME IP Addresses

The internal IP network connects all the PIUs in the cabinet with dual Ethernet networks for redundancy. The internal IP network address is 169.254.0.0 (with netmask 255.255.128.0). Each PIU is assigned an internal IP address in the range 169.254.0.1 - 169.254.127.254 (or 169.254.0.0/17 in classless notation). This address range is reused in all SGSN-MMEs and must not be used on any external network connected to the SGSN-MME. The IP address range 169.254.0.1 through 169.254.255.254 (or 169.254.0.0/16 in classless notation) is reserved according to IETF RFC3330.

The 169.254.224.0/19 network is reserved for GIS access. All SGSN-MMEs use a 169.254.X.0/24 network in communication with the GSS, where X is an SGSN-unique network identifier between 224 and 255. The GSS supports 20 such networks.

This new internal IP network 169.254.0.0/17 is not optional, all SGSN-MMEs are given the same internal network.

Address Calculation

As stated previously, the node internal (private) IP addresses for the SGSN-MME system are allocated from the range 169.254.x.x classless private IP addresses. The addresses for each board are allocated during initial configuration, using a specific algorithm to assign host addresses (node internal addresses).

The host part of the IP address is based on a specific algorithm. It is derived as follows:

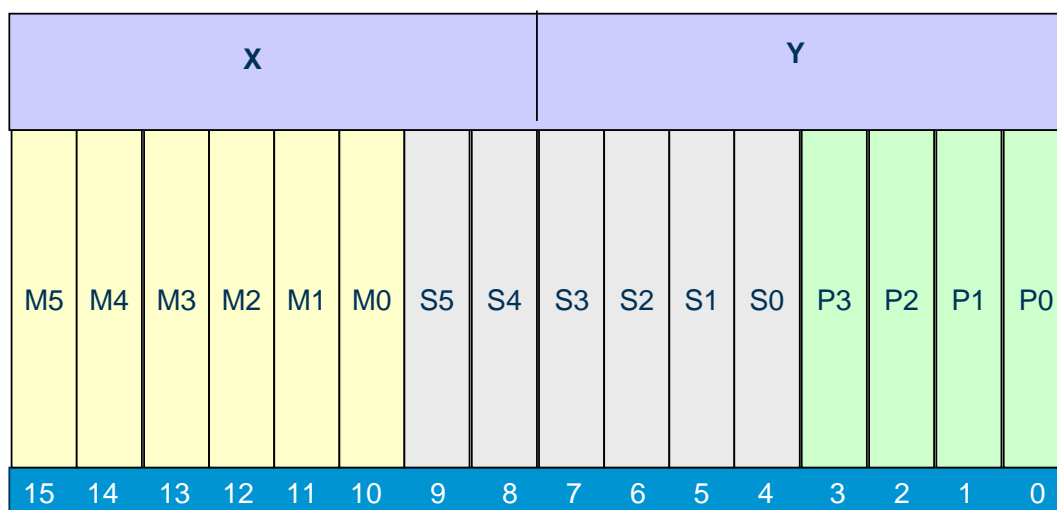
Magazine - 6bits (M0 - M5)

Slot - 6 bits (S0-S5)

Position (processor)- 4 bits (P0-P3). Value 0000 is considered illegal

The decimal values for the Magazine, Slot and Port are translated into binary format and then entered in the tables below to calculate the internal IP address.

For this example, the default network address used is **169.254.X.Y**, where X and Y are given by the following figure. The combination X=Y=255 is reserved as broadcast address.



A node internal Backplane IP Address consist of:

1. 16-20 bits Network Identity (i.e. M_5 - M_2 may also be part of NW when number of magazine ≤ 3)
2. 2-6 bits Magazine Id
3. 6 bits Slot Id
4. 4 bits Position Id

$N_{15}N_{14}N_{13}N_{12}N_{11}N_{10}N_9N_8 \mid N_7N_6N_5N_4N_3N_2N_1N_0 \mid M_5M_4M_3M_2M_1M_0 \mid S_5S_4 \mid S_3S_2S_1S_0 \mid P_3P_2P_1P_0$

Figure 1-22. Internal Private IP address algorithm

For example, let's calculate the IP address of the active NCB board which is found in **eqm01s14p2** (remember that **s14** is actually decimal 20). Enter the values in the table above:

Mag – decimal 1 binary 000001 (6bits)

Slot – decimal 20 binary 010100 (6bits)

Position – decimal 2 binary 0010 (4bits)

Therefore the internal IP address is **169.254.5.66**

Transport IP Addresses

Interface boards are usually assigned an IP address, so traffic can be routed toward specific interfaces for different reasons. For example O&M traffic will be specifically directed toward the Gom interface cards.

Transport IP addresses can be either public or private, depending on the interface and the IP network structure. The external node IP address and subnet mask has to be set by the operator on the interface port.

To use public IP address ranges on the external interfaces, means that the Internet Authority must assign them. The use of private IP address ranges means that the interface/s should not be seen on the public network.

A transport IP address is used as an external IP interface address for a router PIU. One external IP interface address is used for each Ethernet port, or each Ethernet VLAN interface.

SGSN-MME Service IP Addresses

A Service IP address is used for addressing an IP service. These services are specific functions such as GTP-C tunnel endpoint IP address.

The SGSN-MME provides a set of IP-based services for communication with the nodes on the VPN. In this section these services are listed below. Some of the services are fundamental for the operation of the SGSN-MME, while others are optional or only available for users with the appropriate license.

- 1 CDR-FTP
- 2 CDR-GTP-P

- 3 CN-SS7-1
- 4 CN-SS7-2
- 5 DNS
- 6 Filterlog
- 7 GbIP
- 8 GbIP-C
- 9 Gn-GTPC
- 10 Gn-GTP-U
- 11 Iu-GTP-U
- 12 Iu-SS7-1
- 13 Iu-SS7-2
- 14 LI-C
- 15 LI-U
- 16 NTP
- 17 OAM

Normally, only one Service IP address is used per VPN and its associated services (that is, all services for a given VPN share the same Service IP address). The same IP address may be used for services that run in the same VPN. The Service IP addresses cannot be reused in different VPNs.

OAM Service

The OAM service provides the transfer of operation and maintenance traffic between the SGSN and the node management terminal over the Gom interface.

As mentioned previously the active NCB is addressed by using the OAM VPN IP address it also has its node internal IP address. If the active NCB should develop a hardware fault the redundant NCB will take over as active, the use of the OAM VPN IP would route to the active NCB (this was the redundant NCB and has a different node internal IP address) this type of changeover would not be possible if we were using the node internal IP address of the original active NCB.

Gn-GTP-C Service

The Gn-GTP-C service is used for initial control traffic over the Gn and Gp interfaces and for inter-SGSN-MME communication such as Routing Area Updates (RAU). The Gn-GTP-C service also provides the transfer of GPRS Tunneling Protocol (GTP) control plane traffic between the SGSN-MME and the GGSN over the Gn and Gp interfaces.

Gn-GTP-C executes on the Appl-C (AP) PIUs and uses one Service IP address. The Service IP address used for Gn-GTP-C can be added to the DNS server in the GPRS backbone network for SGSN-MME lookups.

Gn-GTP-U Service

The Gn-GTP-U service provides the transfer of GTP user plane traffic between the SGSN-MME and the GGSN over the Gn and Gp interfaces for GSM and WCDMA traffic.

Gn-GTP-U executes on the Appl-U (DP) PIUs and uses the same Service IP address as the Gn-GTP-C as they are in the same VPN.

Multiple services on the same VPN can share the same address. For example, Gn-GTP-C and Gn-GTP-U can share the same IP address because SGSN-MME doesn't use ranges of IP addresses for any non SS7-services.

- › SGSN-MME 2010B doesn't use ranges of IP addresses for any non SS7-services (e.g., Gn-GTP-C).
- › Same IP address is used on multiple AP/DP boards.
- › Multiple services on same VPN can use the same address.
- › The "Reduction of IP" feature is a prerequisite for the 2010B session resilience feature.
- › The operator lease cost of IP addresses is reduced.
- › The planning of IP networks is considerably simplified.

Figure 1-23. Reduction of VPN IP addresses and benefits

SGSN-MME FEATURE HIGHLIGHTS

MkVI+ Capacity

The MkVI+ hardware have introduced two new boards, PEBv5 and IBACv4. With this introduction of new hardware the throughput for WCDMA traffic has increased to 15 Gbps or from 200 to 300 percent increase on the Iu/IP interface. The throughput for GSM traffic has increased to 600 Mbps for Gb/IP or between 30 to 40 percent increase.

High Density Pooling

Expect to enlarge SGSN-MME pool for next generation migration. The SGSN-MME now supports 8 bits for NRI instead of 5 bits in 2010A. This has the result of a SGSN-MME pool size of 256 SGSN-MMEs in Pool.

Pool: Session Continuity at move of WCDMA subscribers

The subscribers are moved within SGSN-MME Pool without loss of PDP contexts and with an insignificant suspension of payload. Any subscriber can be moved regardless it being in PMM-IDLE or PMM-CONNECTED state and also regardless of it having a PDP context or not.

Roaming Restrictions (MME)

Access restriction control is performed when a UE is accessing the SGSN-MME. It is possible to define IMSI Number Series restricting a UE accessing the LTE network. The SGSN-MME analyzes the IMSI number of the UE to check if this is the case. If the UE is restricted from accessing the LTE network, the SGSN-MME sends an appropriate reject message to the UE with a configured cause code. This can be configured on per tracking area basis.

Multiple Time Zones (MME)

SGSN-MME supports multiple time zones. This is useful for operators who have SGSN-MME controlling Tracking areas in different time zones. It enables precise charging of subscribers and their applications. It also enables the UE to have the correct time.

Shared Networks (MOCN) (MME)

Just like the Multiple Operator Core Network (MOCN) feature in the SGSN-MME (GW). This also allows two or more operators to share a radio network while maintaining separate core networks.

Dynamic 3GDT activation based on terminal (SGSN)

As well as the existing methods for selecting 3GDT for subscribers which are HLR and APN based, this allows 3GDT to be selected based on subscriber terminal. This is achieved by configuring 3GDT activation based on user terminal type, namely Type Allocation Code (TAC) consisting of the first eight digits of International Mobile Equipment Identity Software Version (IMEISV).

QoS: Enhanced allocation & retention priority (ARP) parameter for SGSN W/L

In order to ensure that the ‘high priority subscribers’ can get the prioritized access in case of congestion, the SGSN-MME shall support the enhanced ARP over Gn/Gp, Gr and Iu interface. With the enhanced ARP, the packet switch can support a much better granularity for the service priority and can add the operator control and application influence on ARP.

Network Request Secondary PDP Context Activation (NRSPCA) SGSN GW

The NRSPCA procedure is a new procedure for the GPRS IP-CAN. It allows the GGSN to initiate the Secondary PDP Context Activation Procedure. It is applicable for both GSM and WCDMA. It's a new optional feature: Network Requested Secondary PDP Context (FAJ 121 0477). It aims to improve the utilization of network resources since there are more and more possibilities to request bearer establishment with the best suitable QoS from network side for multi-service deployments (Internet, Voice and TV etc..).

CS Fallback to 1xRTT (MME)

The Circuit-Switched (CS) Fallback to 1xRTT (CDMA 1x) feature is used for dual Rx terminals and enables voice and SMS services for LTE subscribers in the CDMA network. The CS fallback is performed using the legacy CS access (CDMA), without need for IP Multimedia Subsystem (IMS) support.

SMS Over SGs (MME)

The SMS over SGs feature enables SMS handling to and from the UE through the MME and the Mobile services Switching Center (MSC)/Visitor Location Register (VLR), using the SGs interface between the MME and the MSC/VLR.

HSS Reset (MME)

At HSS reset all related subscribers will be marked with “Subscriber to be restored in HSS”. Location management procedure will be used to update the MME identity stored in the HSS.

- › MkVI+ capacity
- › High Density Pooling
- › Pool: Session Continuity at move of WCDMA subscribers
- › Roaming Restrictions (MME)
- › Multiple Time Zones (MME)
- › Shared Networks (MOCN) (MME)
- › QoS: Enhanced allocation & retention priority (ARP) parameter for SGSN W/L
- › Network Request Secondary PDP Context Activation (SGSN)
- › CS Fallback to 1xRTT (MME)
- › SMS Over SGs (MME)
- › HSS Reset (MME)

Figure 1-24: SGSN-MME Feature Highlights

2 Features and Functions

OBJECTIVES

Upon completion of this chapter, the student will be able to:

- › Explain the concept of log management and list the available logs in SGSN-MME.
- › Discuss the license management functionality on SGSN-MME.
- › Identify optional features such as SGSN-MME Pool for GSM/WCDMA/LTE and Over Load Protection (OLP)

Figure 2-1: Objectives

.

LOGGING SUPPORT

INTRODUCTION

The logging support makes it possible to collect data in log files and allows secure and efficient logging to files. The logs can be configured and viewed.

To inform the operator of a new published log file, an event is sent. The operator uses FTP to retrieve and delete log files in the directory for ready log files. The logs are stored on two separate partitions on the hard drive, one partition for charging logs and one partition for other logs. All logs, except for performance logs, are predefined at system startup. That is, the operator cannot add or remove logs. All logs are so called multiple logs, which mean that each log consists of a predefined amount of log files. A log file is recognized by the name of the log followed by an index: *Logname.index*. The data is written to the log files in a circular manner, which means that when the first file in the log has reached its maximum file size, it is given the index 1 and a new file is opened. Each log file is identified with a sequential number. When the last file in the chain has reached its maximum size, the first file is overwritten.

- **Logging Support is the Wireless Packet Platform (WPP) function which is responsible for:**
 - creation of logs
 - writing data in logs
 - redundant, secure and efficient storage of data
 - provision of Management Interface (MI) to handle and maintain logs

Figure 2-2. Introduction

As long as measurement data is collected, the log file is kept in a temporary directory. When the log file is closed, it is moved to a target directory (if it has been specified).

- Different parts of the system log information for providing a history of events – these are ‘built-in’ logs (FM, MM or SM)
- Charging uses logging support to write CDRs to the filesystem
 - It uses two ‘built-in’ logs to implement standard and near-real-time charging
- Performance Monitoring (PM) uses logs to write statistical data to files
 - Each PM Job relates to exactly one log - which will have the job's name
 - A maximum of 50 PM logs may exist at the same time
- Logs are distinguished using log names
- Each log has its own log configuration which defines how data is logged

Figure 2-3. SGSN-MME Logs

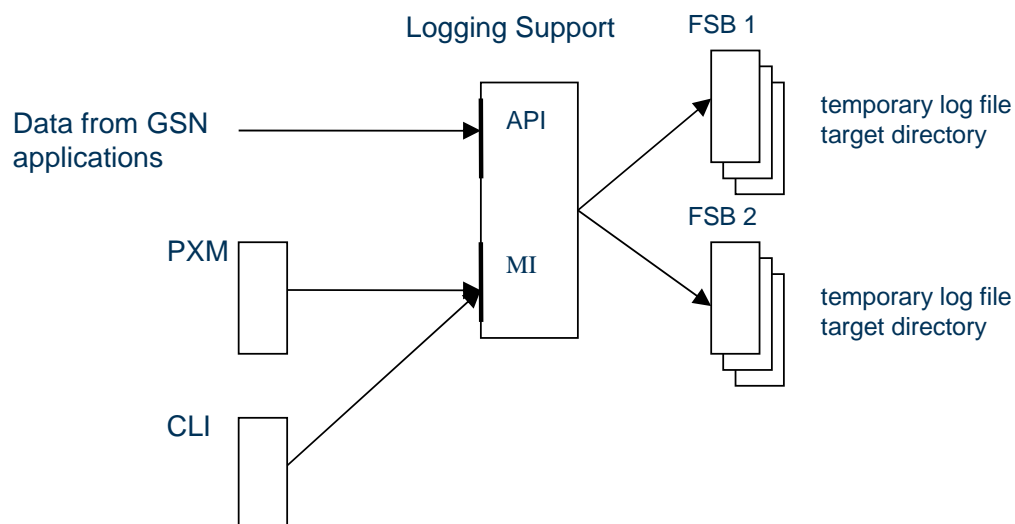


Figure 2-4. Logging Support

Data is forwarded from the different applications to the Logging Support, which stores this data in a temporary log file on the File Server Board (FSB MkV or MkVI) or Node Control Board (NCB MkIV). When the maximum file size of the temporary log file is reached, the log file is closed and stored in the target directory. The Logging Support creates a new temporary log file, which is then used for logging.

Besides this, the system administrator has to take care of storing log files to an external media, in case a persistent storage of data is needed.

After the creation of a log, the configuration data of the log is stored in the Mnesia database. The system administrator can adapt the log configuration data to the given requirements. For the adaptation of log configuration data only one Management Interfaces (MI) is available:

Command Line Interface (CLI)

Changes to the log configuration data can be stored in a persistent way with the checkpoint function of Software Configuration Management.

The GUI used in previous releases to retrieve information stored in the log files has been removed from PXM since SGSN R7. CLI and UNIX commands can be used instead.

- The 16 built-in logs are configured with default parameters.
- The log parameters can only be configured with Command Line Interface (CLI).
- To view the log files only CLI or UNIX commands can be used.

Figure 2-5. Log Configuration

BUILT-IN LOG FILES

Events for both GSM and WCDMA Systems are recorded in combined logs to ease fault management. This is also applicable for inter-system changes between GSM and WCDMA Systems.

The following logs are stored in the `/tmp/OMS_LOGS/` directory, except from the `chsLog` and `chsGtpPrimeLog` logs, which are stored in the `/tmp/OMS_CHARGING` directory. All the logs store information for both GSM and WCDMA Systems with the exception of the `ADC` and `Gs_interface_log` logs, which store information for GSM only.

During the initial start up, the built-in logs of a SGSN-MME are automatically created. The SGSN-MME has 16 logs to store all important data and actions:

- **ADC** Data is stored in the Automatic Device Configuration (ADC) log each time an unknown Home Public Land Mobile Network (HPLMN) subscriber attaches the GSN, or the International Mobile Station Equipment Identity Software

Version (IMEISV) is modified for a known HPLMN subscriber, or an Access Point Name (APN) in Home Location Register (HLR) data is new or modified for a known HPLMN subscriber.

- **AdmissionControlUsage** Events related to features and capacity licenses, such as changed feature or capacity licenses, exceeded licensed capacities, invalid license-key files, or emergency unlocks, are stored in the AdmissionControlUsage log
- **au_data_log** Failed MS authentications are stored in au_data_log. Failed MS authentication procedure can be caused due to, for example, SRES failure, MAC failure or synchronization failure. The GSN is able to store the au_data_log for a minimum of 72 hours. Only users belonging to the security group can access this log.
- **chsLog** In postpaid charging, the CDRs are collected in chsLog. Only users belonging to the charging group can access this log.
- **chsGtpPrimeLog** In near-real-time charging, the CDRs are grouped into GTP' PDUs. If the connection for transfer of GTP' PDUs to the external billing systems fail or the billing systems do not respond, the CDRs are after a while stored in chsGtpPrimLog. Only users belonging to the charging group can access this log.
- **ebs** The events collected by the Event-Based Statistics (EBS-S) for OSS-RC feature are stored in the ebs. EBS-S logs successful and unsuccessful events, formatted according to logging criteria.
- **er_data_log** Traffic events are stored in er_data_log. Traffic event recording is a tool for finding problems in the network and is used for tracking subscribers who complain about problems with their GPRS service.
- **fm_alarm** All occurred alarms and alarm clearings are listed in the fm_alarm log.
- **fm_event** All occurred events are stored in the fm_event log.
- **Gf_IMEIcheck_log** All IMEI_CHECK failures are stored in the Gf_IMEIcheck_log.

- **Gs_interface_log** Mobile status messages sent over the Gs interface, for indicating errors, are stored in the Gs_interface_log.
- **list_subscribers_result** Subscribers registered in the GSN can be listed with the list subscribers CLI command and are stored in the list_subscribers_result log.
- **mmi_log** All activities on the machine-to-machine interface are stored in the mmi_log.
- **mobility_event_log** All Attach Reject messages due to network failure are stored in the mobility_event_log.
- **OMS_SM_Log** Each action performed by the operator is stored in the OMS_SM_Log. Only users belonging to the security group can access this log.
- **Performance monitoring logs** The measurement data collected by a measurement job is stored in a log, whose name is identical to the name of the measurement job. A maximum of 50 performance monitoring logs can exist at the same time.

- ADC – Automatic Device Configuration (ADC) log
- AdmissionControlUsage – Events related to features and capacity licenses
- au_data_log – Failed MS authentications
- chsLog – CDRs are collected in chsLog
- chsGtpPrimeLog – near-real-time charging CDRs
- ebm – Events collected by the Event-Based Monitoring feature
- er_data_log – Traffic event recording
- fm_alarm – All occurred alarms and alarm clearings
- fm_event – All occurred events

Figure 2-6. SGSN-MME Default Logs (1/2)

- › Gf_IMEIcheck_log – All IMEI_CHECK failures
- › Gs_interface_log – Mobile status messages sent over the Gs interface, for indicating errors
- › list_subscribers_result – Subscribers registered in the GSN can be listed with the list_subscribers CLI command
- › mmi_log – All activities on the machine-to-machine interface
- › mobility_event_log – All Attach Reject messages due to network failure
- › OMS_SM_Log – Each Cli action performed by the operator
- › Performance monitoring logs
- › UE Tracer Log – Information on signaling messages for UE

Figure 2-7: SGSN-MME Default Logs (2/2)

LOG CONFIGURATION

As discussed earlier, there are 16 default log files in the system. All log files are configured with a set of default parameters. If required, the parameter sets can be modified by using the Command Line Interface (CLI).

The command syntax of the CLI commands is similar to the standard Unix commands, and CLI and Unix commands can be combined to Shell scripts for the automated execution of commands. The CLI provides the following commands for the log configuration:

set_log_config for the configuration of a log

get_log_config for the printout of the log configuration

list_logs for displaying a list of all the available logs.

- The following CLI commands are available to configure a log:

- `list_logs` list all the available logs
- `set_log_config` for the configuration of a log
- `get_log_config` to printout the log configuration

Printout example of `get_log_config` :

```
=== root@eqm01s14p2 ANCB ~ # gsh get_log_config fm_event
Log Name                : fm_event
Status of Log           : open
Description              : The fm_event log is used for
logging of sent events.
Maximum File Size (Bytes) : 1000000
Number of Files in Log   : 255
Date in Target File      : false
Auto. Delete Target Files : false
Auto. Delete After (Days) : 5
Min. Wrap Interval (Hrs) : 2
```

Figure 2-8. Log Configuration using CLI

Log Parameters

- The following parameters are available:
 - **Log name**
Label to uniquely identify this log. Cannot be changed for build-in logs
 - **Status of Log**
indicated by the options open or closed
 - **Description**
A explanation text about the log's purpose - cannot be changed once the log is created (0-255 characters)
 - **Maximum File Size**
The maximum file size in bytes - between 512 bytes and 10Mbytes
 - **Number of Files**
The number of files created until the first file will be overwritten minimum 1 file, maximum 64999 files

Figure 2-9. Log Configuration Parameters (1/2)

- The following parameter is available:
 - **Date in Target File**
optionally a date string may be included in the file's name may be "true" or "false".
 - **Auto delete**
The log system can optionally delete old target files after a defined duration may be "true" or "false".
 - **Auto delete after**
The duration is defined here in days - min. 1 day - max. 180 days
 - **Min. Wrap Interval**
The wrap interval defines the smallest time duration in hours for overwriting an existing log file. If underrun the event *logWrappedTooEarly* is raised.
Minimum 1 hour, maximum 168 hours

Figure 2-10. Log Configuration Parameters (2/2)

Log file parameters:

- **Log name** – Label to uniquely identify this log. Cannot be changed for built-in logs
- **Status of Log** – indicated by the options open or closed. Cannot be changed for built-in logs
- **Description** – A explanation text about the log's purpose - cannot be changed once the log is created
- **Number of Files** – The number of files created until the first file will be overwritten. The minimum value is 1 file; the maximum value is 64999 files.
- **Maximum File Size** – The maximum file size in bytes. Values can be set between 512 bytes and 10Mbytes.

Each log keeps a temporary file having an index number as file name extension, which will take the most recent data. Temporary files are closed at two events:

- When maximum file size parameter is reached. Or
- When a node reload occurs.

In both cases the temporary file is closed and shifted into the permanent area ("ready" directory) and a new temporary file with a shifted index number is created and opened for taking further log information.

If the Number of Files parameter is crossed by stepping the index number of the file, the temporary file will get the value 1 again. As soon as this temporary file is filled or a reload occurs the original file in the “ready” directory will be overwritten with the closed temporary file. This action is considered to be a **wrap**.

- **Date in Target Files** – The variable specifies if the ready date shall be included in the log files contained in the target directory.

The ready date consists of year and the day of the year (gregorian date). For example, 2001055, is the 55th day of year 2001, which is the 25th of February 2001. This variable is set for each application that generates logs.

- **Auto delete** – The log system can optionally delete old target files after a defined duration. This function does not depend on the file wrapping. The Parameter may be “true” or “false”.
- **Auto delete after** – The duration for deletion of target files is defined here. The parameter unit is in days. Minimum is 1 day; Maximum is 180 days.
- **Min. Wrap Interval** – The wrap interval defines the smallest expected time duration in hours for overwriting an existing target log file with a closed temporary log file. If under run the event *logWrappedTooEarly* is raised.

Minimum value is 1 hour; Maximum value is 168 hours.

- Logs are kept in files on the filesystem of the NCB/FSB boards
- Logs except charging logs are stored under /tmp/OMS_LOGS
- Charging logs are kept in a separate partition below /tmp/OMS_CHARGING
- Each log owns one directory below /tmp/OMS_LOGS with the log's name
- The directory contains a 'ready' and 'tmp' directory
- The finished files are stored in the 'ready' directory

Figure 2-11. Logging targets

- Each log uses multiple files to hold the logged data
- A file is filled when the defined '*maximum file size*' is reached or the GSN is restarted
- Then the logging service creates a new file to store later data
- This is done until the '*number of files*' files are created then, the logging service starts overwriting the first log file - a wrap has occurred
- If the wrap happened faster than defined in 'Min Wrap interval' the event *logWrappedTooEarly* is raised

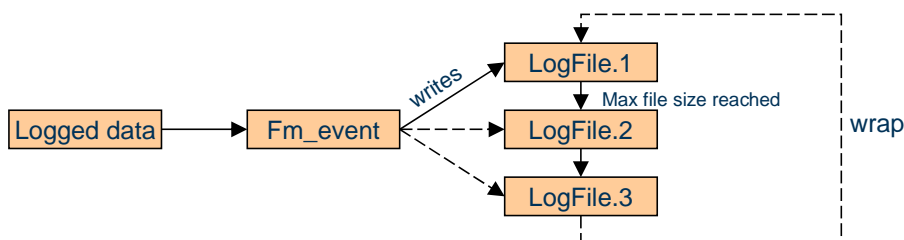


Figure 2-12. Wrapping and Multiple Logfiles

ALARMS AND EVENTS

- The log containing all system event information is implemented by the 'fm_event' log
- fm_event log is located in /tmp/OMS_LOGS/fm_event/
- Two directories are below fm_event:
 - tmp/ contains the currently open log file
 - ready/ contains all closed and ready for transfer log files
- By default the ready directory will contain up to 255 files of a maximum size of 1 Mbyte each
- The files in ready directory are named *fm_event.<index>* where *<index>* is an integer number - e.g. fm_event.4

Figure 2-13. Example: fm_event log

- The following alarms and events are related to log files:
 - logDiskFailure
The logging function could not access its disk partitions. Data can not be written to the log or log files can not be wrapped (or both). Reasons can be e.g. too many open files, too large log file.
 - logWrappedTooEarly (event)
This event is generated when a log wraps faster than expected

Figure 2-14. Alarms and Events related to Logging

LogDiskFailure

The logging function could not access its disk partitions. Data can not be written to the log or log files can not be wrapped (or both).

Cause of the Alarm

This alarm is generated when:

- There is no free space left on the device while writing an ordinary file or creating a directory entry.
- The user's quota of disk blocks or inodes has been exceeded.
- The system file table is full.
- Too many files are open.
- The log file is too large.
- An attempt to modify a file or directory was made on a device mounted read-only.
- An attempt was made to access a file in a way forbidden by the protection system.

Consequences

Data can not be written to the log or log files can not be wrapped (or both).

Severity

Critical

Category

Processing

Actions to Be Taken

Check the sum of the maximum log size for all logs and delete unnecessary files from the disk partition of logging.

LogWrappedTooEarly

This event is generated for logs when a log file is overwritten too soon by a new version.

Each file in a log has a sequence number, which is appended to the filename specified in the File Path field in the Log Configuration PXM form as described earlier.

Wrapping means that one file has been filled, and the application starts writing log data to the log file with the next sequence number, overwriting a possible previous version of the file. When the file with the highest number has been filled, the application opens the file with sequence number 1 again.

Cause of the Event

The following are the most likely reasons for this event:

- The log's Min. Wrap Interval value is too high.
- The log's Max. File Size value is too low.
- The log's Number of Files value is too low.

Consequences

For every log wrap, there is a risk of data loss. If these wraps occur too often, considerable amounts of logging information can be lost. Using the Maximum Wrap Loss setting, you can specify how great a portion of the maximum log size can be lost before an alarm is issued. An alarm is issued if two log wraps have occurred too close in time and the size of the file being overwritten is equal to or greater than the following value: $\text{Maximum Wrap Loss} \times \text{Maximum File Size} \times \text{Number of Files} \times 0.01$

Severity

Critical

Category

Processing

Actions to Be Taken

For the log, increase at least one of the following values by using the *set_log_config* CLI command:

- Max. File Size
- Number of Files

For the log, decrease the following value by using *set_log_config* the CLI command:

- Min. Wrap Interval

SESSION RESILIENCE

The Session Resilience feature, for both Application Processor (AP) and payload Device Processor (DP), is always active for all supported HW and SW configurations.

Session resilience maintains true end-user session or payload services during situations when board of the node is out of service for maintenance, or board failure reasons. Session continuity is of particular importance when running streaming or conversational services. Examples of those kinds of services are video streaming, mobile TV, or Voice over IP (VoIP).

Through session continuity, the end-users do not experience session disturbances when parts of the node are out of service. The operator can manage the node without worrying for outages, and the end users are more satisfied and more likely to be loyal to the operator.

- › VoIP and streaming sessions demands higher serveability.
- › Ericsson SGSN-MME has proven ISP with 99,9995 % system availability or better measured on commercial nodes. Introduction of session continuity is a step in the direction to keep Ericsson on a competitive edge.
- › Seamless board replacement is needed in relation to capacity scale-up and HW maintenance. It will also be possible to move subscribers between boards for maintenance reasons with TMO functionality.
- › SGSN-MME Pool does not cover short and partial downtime like single board events.

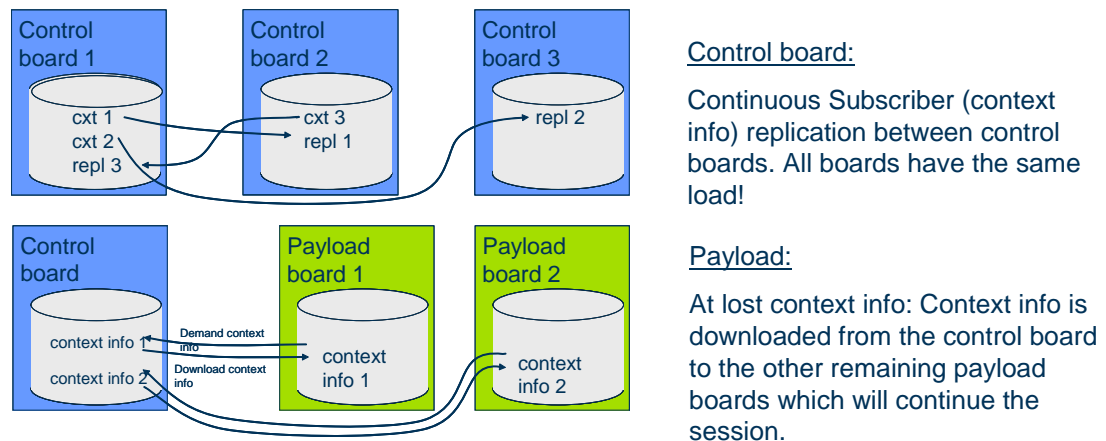
Figure 2-15: Why need Session Resilience?

The SGSN-MME has proven excellent In-Service Performance (ISP) characteristics. Session resilience is a further step towards session continuity by securing user redundancy at single board outages and also at planned maintenance.

In general, session resilience could be accomplished in several different ways. The SGSN-MME uses true M:N redundancy mechanism, meaning that all boards are backup for each other, as well as load-sharing is performed between available boards. Session resilience requires no extra HW or extra configuration to accomplish this.

Moreover, there are two Session Resilience mechanisms included in SGSN-MME 2010B:

1. Subscriber context preservation, pertinent to the AP.
2. Payload continuity, pertinent to the payload DPs.



- No or little session interruption if boards are blocked or taken out of service
- No or little SAU loss if boards are blocked or taken out of service
- Automatic recovery and load balancing after de-block

New
RevenuesUser
Experience

Figure 2-16. Session Resilience Mechanisms

Without the function preserving subscriber context information, the subscriber data is lost and must be retrieved for that PIU, meaning the subscriber must reattach. When preserving subscriber context information, if a PIU, containing this information, is taken out of service, the information is not lost. The same information has been replicated to other PIUs having the same role.

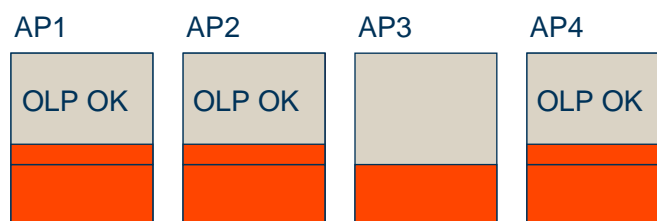
Payload continuation means that if a payload DP is taken out of service, all other payload boards will automatically take over and share the load from the lost board and serving of subscriber payload. The payload users are not disturbed by the event. The takeover is done after the PIU retrieved the context information. Without this feature, the streaming sessions must be restarted before the service can continue. Session resilience in SGSN-MME results in very short session outages, which are not even noticed by the end users in a majority of the cases. When the PIU is de-blocked, after being blocked, the load is balanced between the PIUs.

OVERLOAD PROTECTION (OLP)

OLP is designed to protect the node from overload. In case a processor is loaded over the level for a defined duration new user requests will be rejected, if the overload continues also existing subscribers will be removed from the node. CPU load on APs can increase up to 95% and available memory is used up to remaining max 30MB before OLP is rejecting subscribers (Different HW is equipped with different amount of memory). OLP is improved in SGSN-MME 2010B with higher CPU level and memory measurement used in OLP mechanism. OLP mechanism is used in normal operation but also to reject AP and DP take over in case it will result in overload.

Application Processor OLP

The Application Processor OLP protects the control system on the APs from being overloaded by discarding signaling messages that would start new transactions. Existing subscribers are prioritized since the limits for new subscribers are lower than for existing subscribers.



- OLP will constantly measure and keep track of CPU resources (load and memory) on APs. At AP intervention the information is used when taking decision to accept or reject the AP takeover. This is to avoid overload of the remaining APs in the node.

Figure 2-17. OLP handling for AP

LICENSE MANAGEMENT

The license management provides support to manage feature and capacity licenses, according to the license agreement with Ericsson. The licenses are defined in a License Key File (LKF) that specifies what features are possible to configure and use, and what capacity the SGSN-MME allows.

Ericsson License Manager (ELIM) is used for SGSN-MME same as CPP products (WRAN-RBS, RNC, M-MGw).

Each license is identified with Software License Target ID (SWLT ID), which SGSN-MME uses IPB number as. And the IPB number is the identity of the SGSN-MME that Ericsson assigns to the SGSN-MME when it is delivered, to keep track of all SGSN-MMEs.

- › Uses Ericsson License Manager (ELIM)
- › Each SGSN-MME is identified with Software License Target ID (SWLT ID = IPB number)
- › Each license file binds to a specific SGSN-MME by a unique fingerprint
- › The complete set of licenses are delivered in an XML based text file called License Key File (LKF)
- › Two types of licenses
 - Feature licenses enables functionality
 - Capacity licenses trigger alarms when limits are exceeded

Figure 2-18. License Management

LICENSE KEY FILE

The LKF is supplied by the local Ericsson Support. The LKF is bound to a specific SGSN-MME by a uniquely defined fingerprint, which is created by the software the first time the SGSN-MME 2010B software is installed.

The fingerprint will be changed every time an initial installation is performed. A new LKF, with the correct fingerprint specified, needs to be requested from the local Ericsson support.

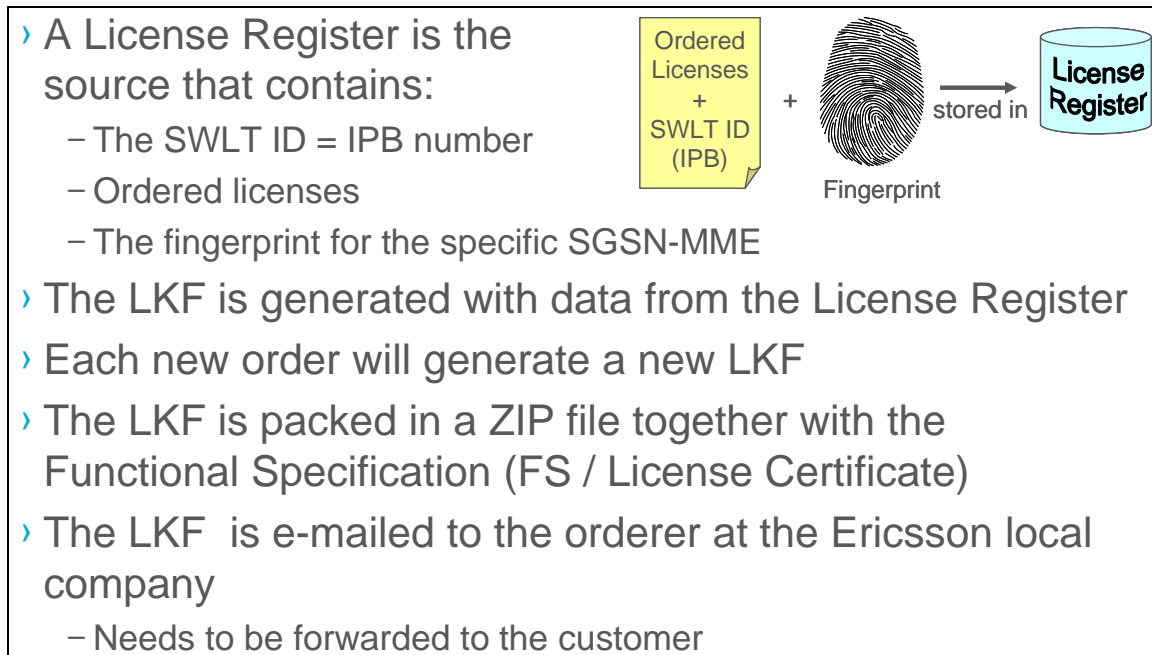


Figure 2-19: License Register & LKF Distribution

The LKF also contains information such as a sequence number that is incremented every time a new LKF for the specific SGSN-MME is requested. Together with other SGSN-MME and license-specific information such as Software License Target ID (SWLTID) number and an encrypted signature.



Figure 2-20. Example of LKF

After an initial installation all feature licenses are granted and the capacity licenses are set to the highest sellable value, for a period of maximum 30 days or until the first LKF is installed. The license management functions can temporarily be bypassed with the emergency unlock function when an LKF has been installed.

The emergency unlock function should only be used when license management are causing major service interruptions for features included in the license agreement. A new LKF needs to be ordered by the local Ericsson support, to reset the emergency reset counter.

The license state in the SGSN-MME can be one of the following:

1. Sticky. After an initial installation, the license state is set to sticky. The sticky duration is 30 days or until the first LKF is installed. Observe that if no LKF has been installed before the 30 days expires, all optional features licenses will be set to invalid, the capacity licenses will be set to the lowest sellable value, and all enabled features will be set to feature state off.
2. Normal. After a new LKF is loaded, the license state is set to normal. After the sticky period has expired, the license state is also set to normal.

3. Emergency 1. After performing a first emergency unlock, to enable all feature licenses and set the capacity licenses to the highest sellable values, the license state is set to Emergency 1. The Emergency 1 expires after 7 days.
4. Emergency 2. After performing a second emergency unlock, to enable all feature licenses and set the capacity licenses to the highest sellable values, the license state is set to Emergency 2. The Emergency 2 expires after 7 days.

When the Emergency 1 or Emergency 2 period ends, the license settings are set according to the last loaded LKF in the SGSN-MME, or to minimum licenses if the initial license state was set to sticky. If the license state becomes invalid the feature state is automatically set to off.

- › Gives the operator a grace period of 30 days after an initial installation or an upgrade.
- › No restrictions. Allows the use of all features and maximum capacity of the SGSN-MME (HW dependent).
- › An alarm is generated when in sticky mode.
- › The first installed and valid LKF forces the node to require LKFs thereafter.
- › A LKF should be installed within 30 days. Otherwise are the feature licenses removed and the capacity licenses are set to the lowest sellable value.
- › The sticky mode alarm is cleared when the 30 days has expired or when a LKF has been installed.

Figure 2-21. Sticky Mode

INSTALLING LICENSES

This section describes how to install the feature and capacity licenses in the SGSN-MME.

Transferring the LKF to the SGSN-MME

The LKF is delivered from the local Ericsson support as a separate file or in a compressed archive (zip file) together with the Functional Specification (FS). If the LKF is delivered in a compressed archive, it must be extracted before it is transferred to the SGSN-MME.

The LKF is named according to the following format:
`<fingerprint>_<date>_<time>.xml`.

Transfer the LKF to the SGSN-MME, using the Secure File Transfer Protocol (SFTP). Store the LKF in the following directory on the SGSN-MME: `/tmp/DPE_ROOT/DeliveryPackages`.

Verifying the LKF

Verify the LKF and check that all information is set correctly. If the information of the LKF is incorrect, contact the local Ericsson support.

If there already exists an LKF in the SGSN-MME, compare it to the transferred LKF.

Display the transferred LKF, by using the *action_ne_print_license_file* CLI command.

Compare the displayed fingerprint with the fingerprint generated by the *get_ne* CLI command. The fingerprints should be identical.

Check if there already exists an LKF in the SGSN-MME, by using the *action_ne_print_license_file* CLI command.

Compare the sequence number in the head of the displayed LKF with the currently loaded LKF. The sequence number is always incremented by 1.

Loading the LKF

When the LKF is loaded the license settings in the SGSN-MME is changed according to the LKF. The feature licenses that are not included in the LKF are set to invalid. Features with invalid licenses are turned off by the SGSN-MME. It is possible to load a LKF in any possible license state.

Load the LKF, by using the *action_ne_load_license* CLI command.

Verifying the Loaded LKF

A verification of the loaded LKF is to be performed.

If the output of the LKF does not include the correct licenses it is possible to set the SGSN-MME in an emergency state for seven days, where all the feature licenses are granted and the capacity licenses are set to the maximum value defined for the SGSN-MME, by using the *action_ne_emergency_unlock* CLI command. If the emergency unlock is triggered, a new LKF including an emergency reset key must be ordered from the local Ericsson support.

Check the license state, by using the *get_ne* CLI command. The license state should be normal.

Verify that all new or changed licenses are set as ordered, by using the *list_feature* CLI command.

Verify that the *admNoLicenseKeyFileInstalled* alarm is cleared, by using the *list_alarms* CLI command.

Verify that all features are granted, by using the *list_feature* CLI command.

Verify that the capacity values are set as ordered, by using the *list_capacity* CLI command.

Checkpointing the SC

To persistently store the Software Configuration (SC) in the SGSN-MME, it is recommended to perform a checkpoint.

Perform a checkpoint, by using the *checkpoint* CLI command.

- › The LKF file is installed into the Current Software Configuration (SC) by a CLI command.
 - *action_ne_load_license* loads a license-key file into the SGSN-MME and sets the licenses according to the license keys in the license-key file.
 - The SC needs to be checkpointed.
 - If the SGSN-MME falls back to another SC, then the licenses will not be included. The LKF therefore needs to be re-installed. This also applies if the SGSN-MME is restarted on another SC, that has been created on an SC that did not have the targeted LKF installed.
- › The LKF and the status of the licenses/features can be viewed by CLI commands.
 - *action_ne_print_license_file* displays the content of a license-key file.
 - *list_feature* / *list_capacity*
 - *get_feature* / *get_capacity*

Figure 2-22. License Key File on SGSN-MME

The following figure shows how to install the license based on a new SGSN-MME node.

- › **New SGSN-MME order:**
- A new IPB number will be created
 - The fingerprint is generated at the factory
- Steps:
1. The SGSN-MME is installed in the factory, and the fingerprint is created, entering Sticky Mode
 2. The fingerprint is sent to Customer License Center.
 3. The SGSN-MME is shipped to site
 4. The LKF is emailed to site
 5. FTP the LKF to a predefined path on SGSN-MME
 6. Install the LKF
 7. Enable features
 8. Checkpoint the SC

Figure 2-23: New SGSN-MME node

SGSN-MME POOL IN GSM/WCDMA

While having an active PDP session, an MS/UE has to perform ISRAU each time the MS/UE moves to a new SGSN-MME area. This incurs a lot of signaling to the HLR, 'new' SGSN-MME and 'old' SGSN-MME. Signaling can result in delay and cost.

To minimize signaling an ISRAU SGSN-MME can now be pooled. Regardless to where the MS/UE moves in the pool area, the serving SGSN-MME for that MS/UE will not change.

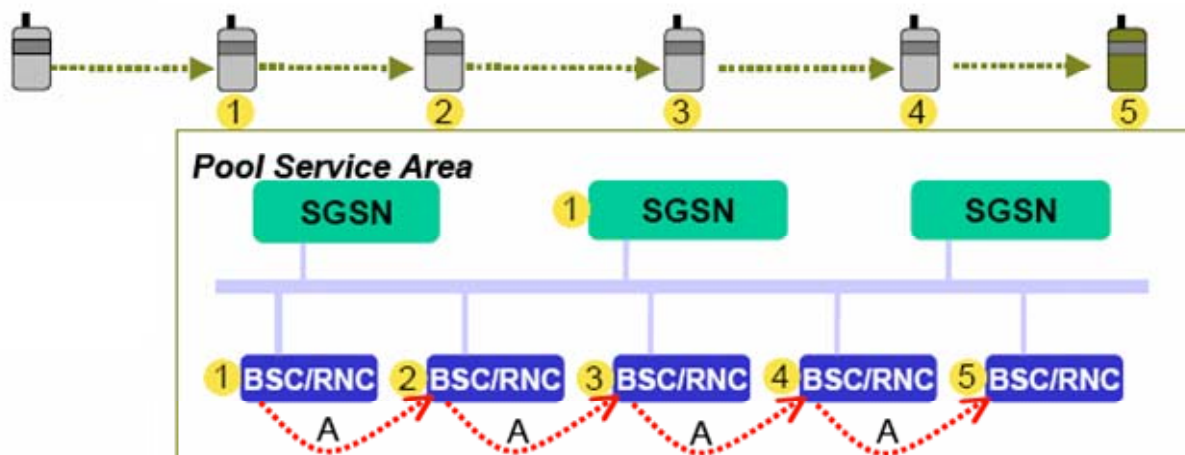


Figure 2-24. SGSN-MME Pool

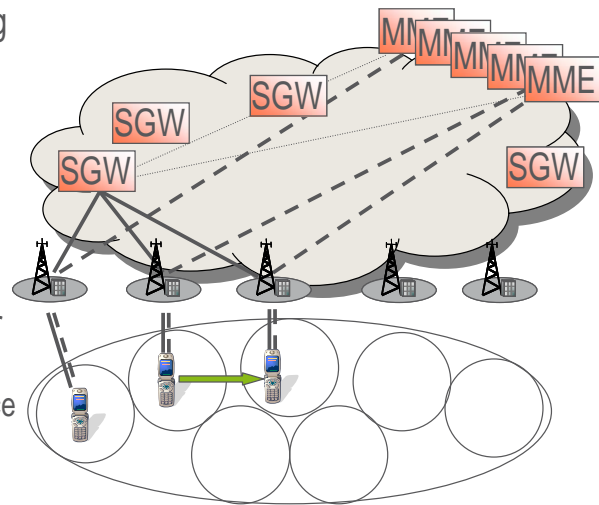
- 1 When an MS connects to an SGSN-MME in pool by attach or Inter-SGSN-MME Routing Area Update (ISRAU) procedures, it is allocated a Packet Temporary Mobile Subscriber Identity (P-TMSI) containing the Network Resource Identity (NRI) of the SGSN-MME. Then MS copies the allocated NRI to its generated Temporary Logical Link Identity (TLLI), allowing the RAN to route the MS traffic to its assigned SGSN-MME. The SGSN-MMEs acting in a pool eliminates inter-SGSN RA update as long as an MS stays within the SGSN-MME pool area.
- 2 When the MS/UE moves to a different BSC/RNC area (arrow A), while it is in the same pool, it will be served by the same SGSN-MME (1). The SGSN-MME in the pool is identified by the NRI.

SGSN-MME POOL FOR LTE

A SGSN-MME pool for LTE (also referred to as S1 Flex in the LTE radio context) is a collection of SGSN-MMEs, configured to serve any common part of an eNodeB network in LTE.

- › Pooled SGSN-MMEs in LTE, serving in parallel the mobiles in the SGSN-MME Pool Area, deployed as a centralized resource

- Simplifies capacity planning and enables load balancing
- Provide network level redundancy for the control functionality
- Simplified operation and maintainance



- › Full mesh IP connectivity needed

Figure 2-25: SGSN-MME Pooling in LTE

This common part is referred to as the SGSN-MME Pool Area and defined as the geographical area in which a UE may move without the need to change SGSN-MME. Hence, even though the UE moves between eNodeBs, it remains anchored to the same SGSN-MME as long as covered by the same Pool Area. This means significant signaling reduction compared to a non-pooled network.

The SGSN-MMEs may be (physically) located independently from the eNodeBs. While the eNodeBs are located in relation to cells and tracking areas from a geographic perspective (cell planning), the SGSN-MMEs may be located freely to separate and even distant sites. This is an important characteristic of the SGSN-MME Pool concept, justified by the fact that the UE remains anchored to a specific UE even during mobility inside the SGSN-MME Pool Area.

A single group of SGSN-MMEs as above may also serve multiple, geographically split SGSN-MME Pool Areas. Then a country (PLMN) may for instance be divided in regional, smaller areas, served by only one or two collections of SGSN-MMEs. By this, a collection of SGSN-MMEs could be dimensioned for the aggregate total shared capacity, which will efficiently absorb temporary peak loads in one part of the network.

The solution also includes load sharing between the SGSN-MMEs pool members under the control of the eNodeBs. In this, the SGSN-MMEs can be defined a size by a weight factor in relation to the other pool members. This weight factor – related to the installed capacity of the SGSN-MME in proportion to the other pool members – is configurable by the operator and used in the eNBs load distribution algorithms.

Furthermore, a SGSN-MME can be taken out service while continuing network traffic. This is possible thanks to the Ericsson unique move functionality in the SGSN-MME, whereby subscribers are gracefully moved from the related SGSN-MME to the other in the Pool.

The move command may be used to empty a SGSN-MME fully or partly (selected UEs) and is also a very efficient method of re-population of re-installed SGSN-MMEs (which is a very time-consuming process without this unique move functionality). The methodology to move subscribers will gradually be evolved in the Ericsson SGSN-MME.

The SGSN-MME Pool/S1 Flex solution is similar to the SGSN-MME Pool/RNC/Iu-flex and the SGSN-MME Pool/BSC/Gb-flex functionality as previously discussed, which simplifies mobility handling between LTE and 3GPP 2G/3G access networks.

In a dual or triple access SGSN-MME, multiple access pools can be served.

All in all, SGSN-MME Pool provides the following benefits:

- Signaling capacity savings;
- Full redundancy ensuring almost 100% network ISP;
- Optimized capacity dimensioning due to common capacity sharing between SGSN-MMEs;
- Support for multiple, geographically split, SGSN-MME Pools enables high flexibility in network topology design;
- Support for dual/triple access (LTE/WCDMA/GSM) SGSN-MME Pools by the same node ensures simultaneous, shared node resources at all times.

3 Node Management

OBJECTIVES

Upon completion of this chapter, the student will be able to:

- › Identify the O&M network supporting the GPRS network
- › Understand the different management domains in the SGSN-MME.

Figure 3-1. Objectives

OPERATION & MAINTENANCE

INTRODUCTION

The O&M networks consist of node management terminals, Operational Support Systems (OSS), billing system, and Network Time Protocol (NTP) servers.

The user interfaces, the Command Line Interface (CLI) and the Packet Exchange Manager (PXM), are used to carry out O&M functionality on the SGSN-MME.

The SGSN-MME is connected to the O&M networks over the Gom interface.

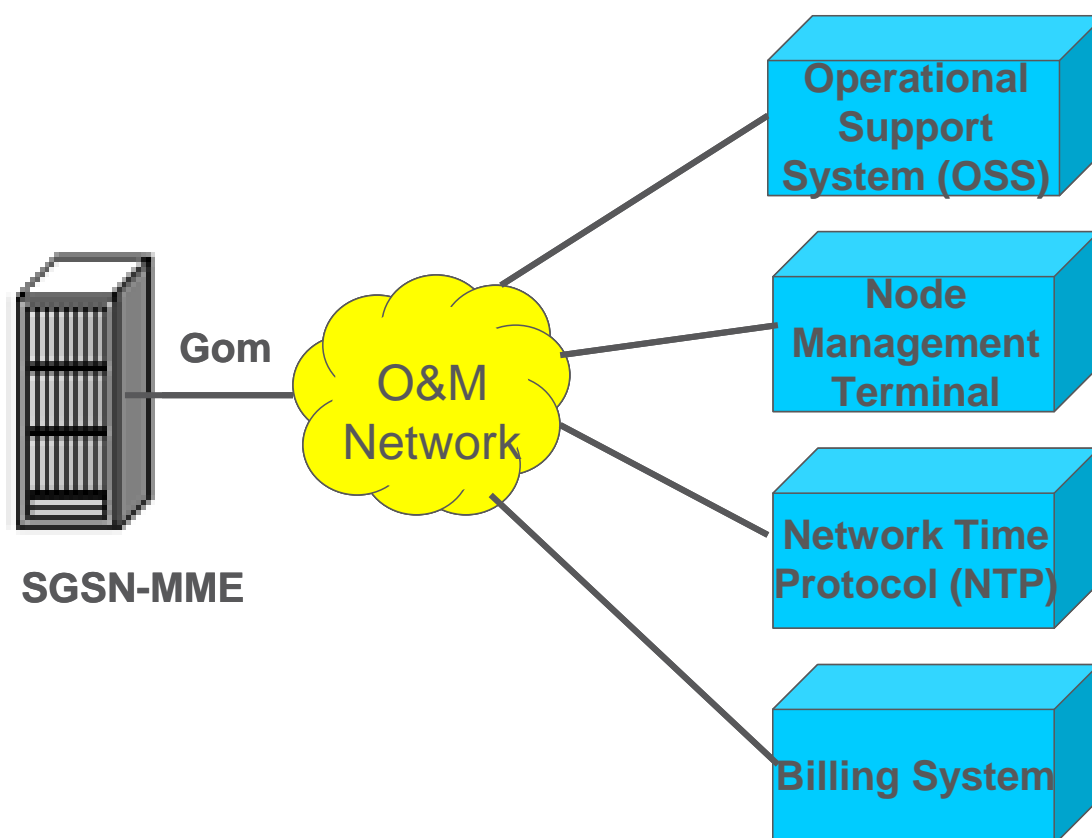


Figure 3-2. O&M Networks

The SGSN-MME WPP platform is based on industry standards and is easily managed using a Web-based management system (PXM) or a Command Line Interface (CLI). The management system is highly robust, with advanced functions for capturing software faults, isolating hardware faults, and protecting against overloads.

Operation Support System (OSS)

The Operational Support Systems (OSS) handles sub-network management for several nodes. The O&M protocols used for communication and transfer of data are IIOP, File Transfer Protocol (FTP) and Simple Network Management Protocol (SNMP). For transport of configuration related commands a protocol based on the NETCONF (RFC 4741, RFC4742) standard is used. NETCONF is a network management protocol where operations and data are sent as XML strings over, for example, telnet or SSH. SSH also provides support for SFTP and SCP, enables automatic encryption, authentication, and compression of transmitted data.

Billing System

The billing system handles charging output. The O&M protocols used for transfer of data are FTP and GPRS Tunneling Protocol Prime (GTP'). Remote Procedure Call (RPC) is used for communication.

Network Time Protocol (NTP)

The SGSN-MME requires an accurate local time for tariff change and time-stamping of alarms, events, and Charging Data Records (CDRs). This is facilitated by the NTP servers that the SGSN-MME is connected to. The SGSN-MME fetches the time from one or several NTP servers. Then, the accurate time is calculated using the NTP-server time values and standard NTP algorithms.

NTP provides the mechanisms to synchronize time and coordinate time distribution in large, diverse global networks. A distributed subnet of time servers operating in a self-organizing, hierarchical master-slave configuration synchronizes local clocks within the subnet to national time standards through wire or radio. The servers can also redistribute reference time.

To assure the identity of the external NTP servers the SGSN-MME uses cryptographic authentication of the Network Time Protocol (NTP) Version 3 protocol. Message Digest algorithm no. 5 (MD5) has been added, as specified in RFC 1305.

SGSN-MME is computing a key dependent cryptographic check sum over the packet so that spoofs, replays etc can be detected and discarded.

MANAGEMENT DOMAINS

- Configuration Management (CM)
- Fault Management (FM)
- Security Management
- Software Management (SwM)
- Performance Management (PM)
- Accounting Management

Figure 3-3. Operation and Maintenance (O&M) functions supported

The O&M tasks can be performed from the embedded element manager that is accessed over two interfaces: the Command Line Interface (CLI) or the Packet eXchange Manager (PXM). An external sub-network management system, such as Ericsson's OSS-RC, can also be used.

The CLI is the main interface for CM thanks to its support for batch handling.

The PXM is the main interface for monitoring, thanks to the easy visibility of its GUI. The PXM software is implemented by use of a client-server architecture, and the client software can be uploaded from the SGSN-MME to a standard computer (PC) and then launched as a standard web GUI based on Java. The Java applets communicate with the node over the Internet Inter-Orb Protocol (IIOP). This ensures that the GUI is consistent with the node traffic handling software, and different versions of the SGSN-MME nodes may coexist in the same network.

The O&M activities can also be performed from a sub-network management system, typically the OSS-RC system. The sub-network management system is connected through the Simple Network Management Protocol (SNMP) for alarm and event handling and through File Transfer Protocol (FTP) for PM. It should be noted that Ericsson's sub-network management system, OSS-RC, provides support for all areas, and both PXM and CLI interfaces can be accessed from OSS-RC.

The SGSN-MME can be managed across an IP backbone network using TCP/IP, and thus remote access is fully supported. The O&M communication can be secured by using SSH2 and IPSec to prevent unauthorized access. IPSec for Secure Network Traffic (FAJ 122 742) is an optional feature.

The SGSN-MME supports O&M related functions within the areas of operation described below.

FAULT MANAGEMENT

The purpose of fault management is to report detected failures, and to limit the failure effect on the network performance. The SGSN-MME automatically creates and sends alarms and events to the node management terminal or to external fault management systems via SNMP, which is used to control and manage IP gateways and other network functions. Each alarm or event is described in its own Alarm and Event Description, which can be found in the CPI library.

Integrated Traffic Capture (ITC) simplifies troubleshooting of traffic-related faults by capturing signaling traffic on multiple interfaces and storing output in log files.

The external fault management systems enable the operator to manage a large number of SGSN-MMEs with a minimum of staff. Therefore, the operator only needs to connect to a specific SGSN-MME when it has sent an alarm or event, instead of having to check it regularly for faults.

Fault management in the SGSN-MME includes the following functions:

- Alarm and event list
- Alarm and event log
- Alarm and event subscribing
- Alarm and event filtering
- Clear alarm
- Force clear alarm
- ITC log files

CONFIGURATION MANAGEMENT

The configuration management functions enable the operator to set, modify and examine configuration parameters at any given time when the SGSN-MME is up and running. This is done by using CLI commands and in some cases through PXM. The OSS application GPRS CM can also be used to configure the SGSN-MME. The configuration data is contained in files and databases in the SGSN-MME. All database tables and files are part of the active Software Configuration (SC) meaning that they will be saved to disk when the active software configuration is check pointed.

There are two different types of Configuration Management (CM) frameworks in the SGSN-MME:

- A CM framework that uses a distributed database for the applications. Configuration changes handled by the framework are distributed and applied instantly. This CM framework handles all configurations in the SGSN-MME, except SS7, IP and Feature Management Configuration.
- A CM framework, Object Manager (OBM), which consists of a central database which stores configuration data for all applications handled by the framework. The OBM handles the SS7, IP and Feature Management configuration.

The relation between the frameworks, protocols and interacting systems is shown in the figure below.

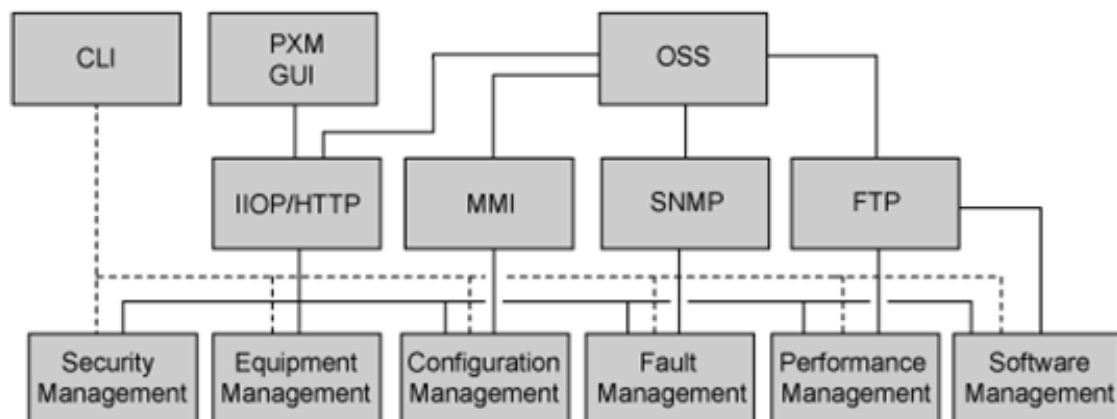


Figure 3-4. Management Systems Overview

Object Manager

The OBM works with two configuration areas, of which one is active and one is planned. The active area holds the currently running configuration of the node and the planned area holds the modified configuration before it is activated.

The OBM gives the operator the possibility to modify the configuration in the planned area without changing the running configuration in the active area. Because of this, configuration changes do not have to be performed in any specific order. Before the new configuration is activated (propagated to the application and moved from planned to active area), a consistency check of the new configuration must be completed. If the consistency check reports errors the operator can either undo all pending changes in the planned area, or correct the errors by using applicable CLI commands. A new CLI, introduced in SGSN-MME 2008B, that forces an activation shall only be used when authorized by Ericsson, as it could impact the stability of the node.

The active configuration data can be exported to a file in CLI format. It can for example be used to configure another node, after editing of parameters unique to the node.

The list CLI commands handled by the OBM support different ways of filtering the output compared to those handled by the regular CM framework. The usage texts for OBM CLI commands are generally more detailed than the texts for regular CM CLI commands.

Note: The PXM application can not monitor configuration handled by the OBM framework (that is SS7, IP and Feature Management).

Separate access privilege is necessary for the different configuration areas in OBM. For example, SS7 configuration area needs the action sets such as `ss7_passive`, `ss7_active` and `ss7_action_active`. IP area needs the action sets such as `ip_passive` and `ip_active`. This course doesn't cover the detail about action set.

Figure 3-5. OBM Coverage

NODE MANAGEMENT TERMINAL

The node management terminal can be any desktop or stationary computer on which the CLI, the PXM, and the Customer Product Information (CPI) can be accessed either locally or remotely. A client-server architecture is used where the node management terminal is the client part and the server part executes on the SGSN-MME. The CLI, the PXM and the CPI are a part of the SGSN-MME software. Any software update or upgrade that introduces changes to the functionality will include the corresponding CLI, PXM, and CPI. A web browser, Java plug-in, and a telnet client must be installed on the node management terminal.

The CLI can be accessed through telnet and rlogin, either unencrypted or through Secure Shell (SSH). SSH traffic is encrypted and can enhance the security when remotely logging on to the SGSN-MME. Hence, SSH enables the use of CLI and File Transfer Protocol (FTP), Secure FTP (SFTP), and Secure CoPy (SCP) without sending passwords, and other traffic, in clear text over the IP network. SSH is only used for providing security for terminal login, such as CLI, FTP, SFTP, and SCP.

SGSN-MME supports SSH protocol version 2 (SSHv2) only. With SSH, the SGSN-MME can be accessed using password authentication or public key authentication. When using public key authentication, additional configuration is required on the SGSN-MME.

The PXM software consists of Hypertext Markup Language (HTML) pages, Java applets, and pictures in gif and pdf format. The HTML pages are downloaded through Hypertext Transfer Protocol (HTTP). The Java applets are executed inside the web browser with the help of the installed Java plug-in. When the Java applets are started, they connect to the SGSN-MME through Internet Inter-ORB Protocol (IIOP). IIOP is an object-oriented transport protocol, which makes it possible for distributed programs in different programming languages to communicate. For protecting PXM traffic, IPSec can be used. For charging data it is recommended to use IPSec due to the large dataload.

CPI provides the customer with all the information necessary to operate and maintain the SGSN-MME. It consists of HTML pages, which are downloaded through HTTP, and illustrations in gif and pdf formats. The CPI is stored on a SGSN-MME in a CPI library and can be viewed and printed online with a web browser. The CPI library is accessed through the PXM and the user can navigate in the CPI library with the Active Library Explorer (ALEX).

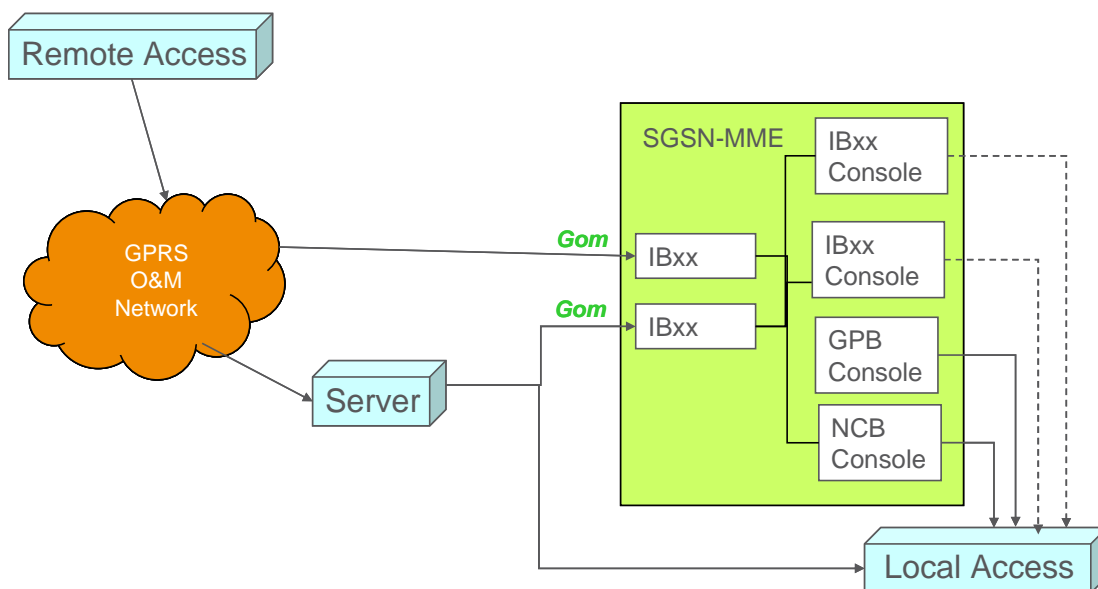


Figure 3-6. SGSN-MME Access

The operator can use the NMT when:

- Configuring hardware and software on a SGSN-MME, this includes timers, interfaces and parameters
- Supervising and monitoring performance in the GPRS system
- Detecting and correcting errors in the GPRS system
- Reading the customer documentation stored on the nodes

An NMT can also address components such as routers and switches in the GPRS internal backbone network. However, the components must have software installed to allow remote access over the IP network.

Equipment

This section describes the equipment required for setting up the PXM and accessing the CLI and ALEX environment.

PXM, CLI and ALEX can be accessed either from a Solaris or a MS Windows based system. The hardware and software requirements for each are specified in the following two sections:

Solaris Requirement

PXM can be set up on any Solaris-based system (with a web browser) that meets the following minimum requirements:

- Processor: SUN Microsystems SPARC 5 running at least 170 MHz
- 128 MB Random Access Memory (RAM)
- 50 MB free space on the Hard Disc Drive (HDD)
- Network adapter, Ethernet 10/100 Mbps
- Transmission Control Protocol/Internet Protocol (TCP/IP) connection (100Mbps connection recommended for PXM)
- The latest patch cluster for Solaris installed
- Common Desktop Environment (CDE) is the recommended windowing system on Solaris
- Web browser: Mozilla 1.7.0 or later
- Required Java Runtime Environment (JRE) version 1.5.0 or later. The JRE can be downloaded from the Sun website.

MS Windows Requirements

PXM can be set up on any Microsoft Windows-based system (with a web browser) that meets the following minimum requirements:

- Processor: Intel Pentium running at least 200 MHz
- 128 MB Random Access Memory (RAM)
- 50 MB free space on the Hard Disc Drive (HDD)
- Network adapter, Ethernet 10/100 Mbps
- TCP/IP connection (100Mbps connection recommended for PXM)
- Web browser: Internet Explorer 6.0 or later
- Required version 1.5.0 or later. The JRE can be downloaded from the Sun website.

Tools and Applications

The following tools and applications rely on several services provided by the GSN:

- File Transfer Protocol (FTP)
- Hypertext Transfer Protocol (HTTP)
- Internet Inter-ORB Protocol (IIOP)

- Simple Network Management Protocol (SNMP)
- Telnet

NMT Connection

A stationary computer configured as a node management terminal is located onsite or at a supervisory center.

Please note that Node Management function can co-exist on a GSN Support System (GSS) server, used for installation and backup/restore of a GSN node.

A computer configured as a node management terminal enables direct, onsite access to individual SGSN-MME.

Connected directly to the SGSN-MME, the node management terminal can only access the internal IP network. There is no access to the Operation and Maintenance (O&M) network. This is often referred to as local access.

It is possible to connect the NMT to the console port on any Interface Board in the SGSN-MME, all boards FSB's, GPB's or IBXX's have a 9 pin Micro D-sub connector using RS232 protocol to communicate with the boards.

Connected to the O&M network, the NMT can access the SGSN-MMEs through their OAM IP Address using the Gom network. It is, however, not possible to access the internal IP network. This is often referred to as remote access. The choices for the Gom interfaces are: ATM or Ethernet via IBxx PIUs.

ELEMENT MANAGEMENT

Management Architecture

The Packet Switching Core Network Management system solution has a clear separation between Element Management, Sub-Network and Network Management.

The proposed management solutions are implemented as logical applications that can be accessed, both locally and remotely over an IP-based Gom interface, by any desktop computer that has a Web browser.

The proposed Management applications for the Packet Switching Core Network consist of:

- Network Management mediation

- Sub-network Manager
- Embedded Element Manager, EM

Network Management Mediation

Integration towards a Network Management system is mainly concerned with Alarm handling. This is supported by protocols like SNMP, FTP or via Command Line Interface (CLI).

The Sub-Network Manager, OSS-RC

Operation Systems Support (OSS-RC) is the sub-network manager for the core and radio network, managing both circuit and packet switched parts. OSS-RC is used for coordinating activities across several network elements such as setting configuration data for tasks that need to be repeated in several different GSN nodes.

OSS-RC gives a comprehensive view and understanding of the situation in all nodes in the core and radio network. OSS-RC is used to collect and store both alarms and performance data for the packet switching sub-network. It acts as a mediation device for network management systems. There are more details on OSS-RC are later in the chapter.

The Embedded Element Manager

The element manager is focused on performing operations on a per network element basis and should be used when the operator wants to perform a task on a GSN node such as configuring parameter(s) or viewing the alarm log or counters for one particular node. When a problem has arisen in a SGSN-MME node and troubleshooting should be done, a field technician uses the element manager locally or remotely to help to locate the fault and carry out corrective actions. PXM is the SGSN-MME's embedded element manager.

The embedded Element Manager is one of the fundamental principles in the Packet Switching Core Network O&M system. This implies that all required software for performing management tasks is contained in the SGSN-MME nodes. The Element Management solution is implemented using a Client / Server architecture.

An element manager can be connected locally or remotely from different hosts. The actual connection to the SGSN-MME node is transparent to the user.

Packet Exchange Manager (PXM)

The Packet eXchange Manager (PXM) is the element manager based on the thin client concept, meaning that all GUI software, such as HTML pages and Java applets, are stored on the node and that a thin presentation layer (Java) is downloaded to and run on the client.

The thin client concept makes sure that the GUI always is consistent with the node traffic handling software. Several different versions of the Ericsson SGSN-MME nodes may exist simultaneously in the network without creating problems with the central management of the nodes.

Management of the SGSN-MME nodes is done via the internal IP backbone network using TCP/IP and thus remote access is fully supported. Local access is also possible via TCP/IP or RS232. This strategy allows many users to manage the same node independently of each other, as well as for the single user to manage many nodes from one centrally placed location like the OSS-RC server.

Optionally, the operation and maintenance traffic can be run through IPSec tunneling to ensure that non-authorized persons can use no information.

The SGSN-MME nodes support several different protocols for PXM:

- HTTP, for downloading html pages such as all the on-line customer documentation and the Java applets and classes for the PXM.

Corba/IIOP, for communication between the PXM and the SGSN-MME.

› **PXM at the Client /NMT**

- Minimum HW/OS required
 - › Sun Solaris or
 - › Microsoft Windows
- IP connection
- WWW browser
 - › Mozilla 1.7 or later
 - › Internet Explorer 6.0 or later
- Java virtual machine
 - › JRE 1.5 or later
- O&M client software is downloaded on demand

› **PXM (packet exchange manager) Server:**

- Integrated within SGSN nodes
- Accessed via active NCB OAM IP address and Port Number 8888

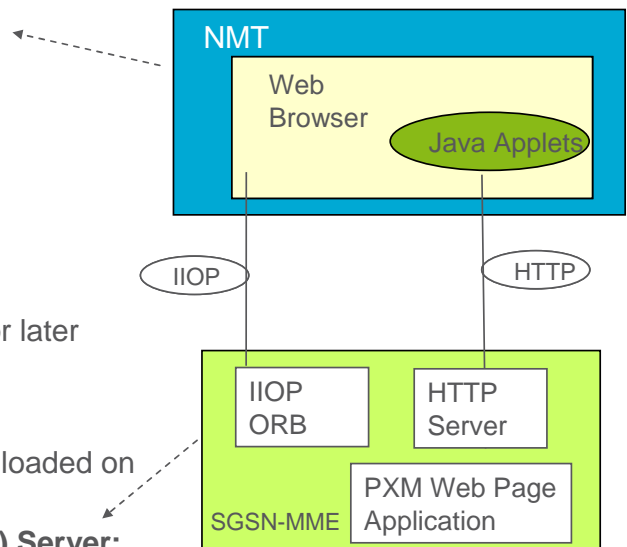


Figure 3-7: Element Management PXM GUI

The PXM has a Graphical User Interface (GUI) which allows many users to manage the same SGSN-MME independently of each other. Conflicting concurrent operations in PXM on the same object are detected and rejected. When the user connects to a SGSN-MME using port number 8888, the PXM is displayed. When opening the PXM, the Login dialog box appears which allows the user to log on to the GSN.

In the packet switching nodes, different operator categories or operator roles can be defined, making it possible to differentiate users depending on assigned tasks and experience. To make sure that no unauthorized access is made a user name and password is required for logging into the Element manager.

Below is an example of PXM with the Login Screen to the current SGSN-MME.

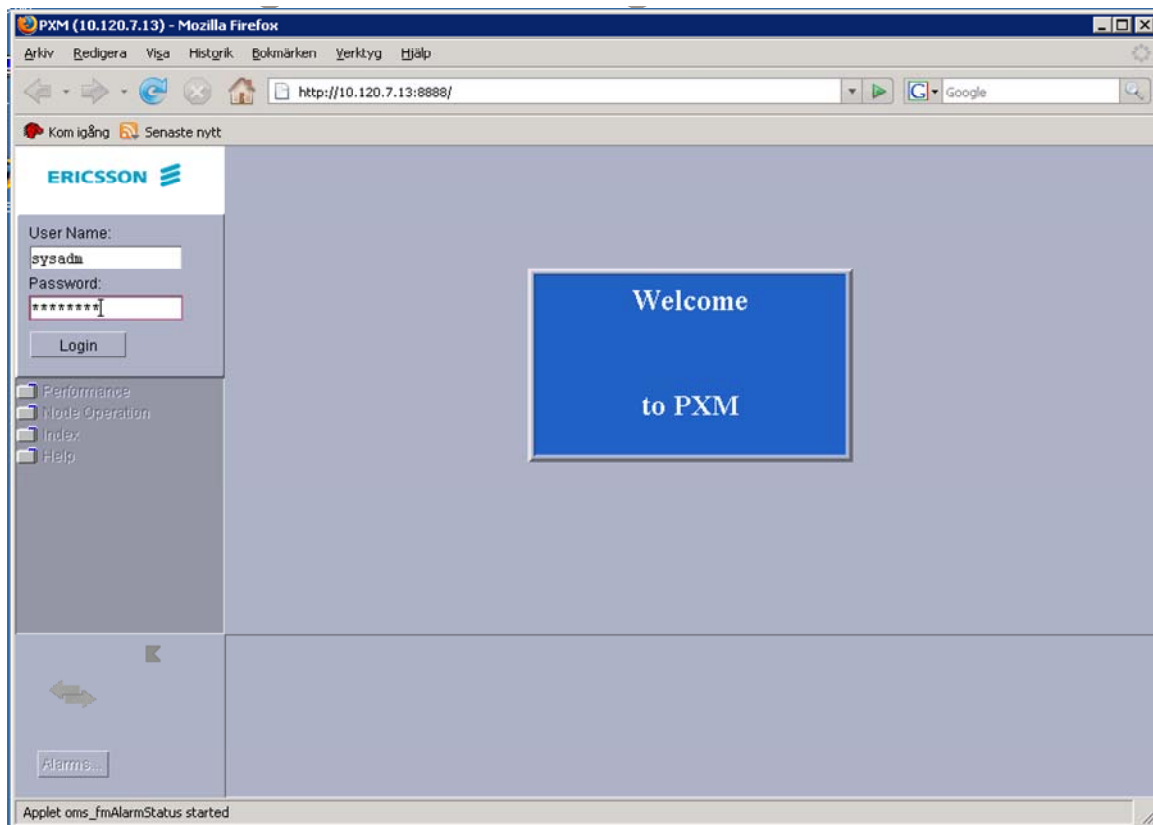


Figure 3-8. SGSN-MME login view using PXM

The **Login** dialog box consists of the following elements:

- The **User Name** box specifies the user name.
- The **Password** box specifies the password.
- The **Login** button approves the logging on process if the user name and password are correct.

When logging in the user name and password will dictate the role assigned at login. This means the type of accesses the operator is allowed to do on the node. For example, a SGSN-MME node would be configured for an O&M person, a NMC person, a network planner. In each definition, the O&M person would require an access to modify interface configuration where as a Network planner would require print access only. It would be decided what type of access each of the groups will need.

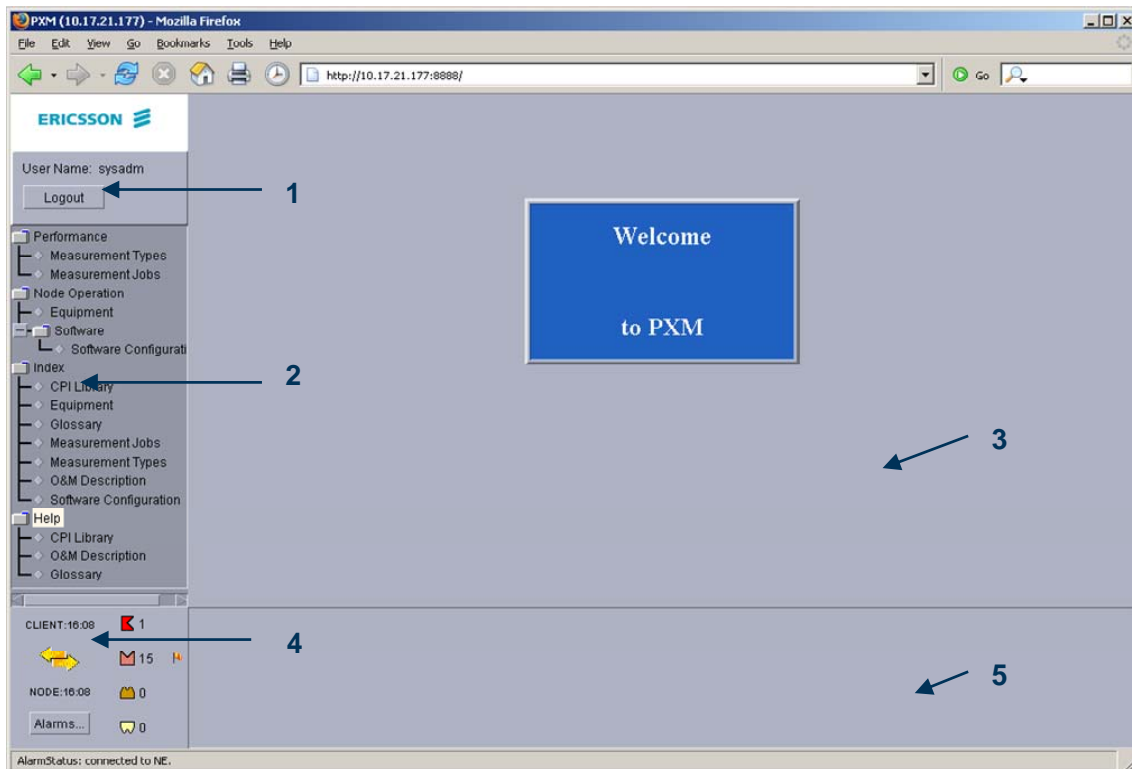


Figure 3-9. Packet Exchange Manager

A typical view of PXM is shown in the figure above. It contains the following:

- 1 User Status Panel
- 2 Root menu
- 3 PXM form
- 4 Alarm status panel
- 5 Result viewer

User Status Panel

The **User Status** panel consists of the following elements:

- The **User Name** box specifies the user name.
- The **Password** box specifies the password.
- The **Login** button approves the logging on process if the user name and password are correct.

When logged in to PXM, the user name will be displayed in the **User Status** panel and the **Logout** button will appear.

The **Logout** button in the **User Status** panel consists of the following elements:

- The **OK** button confirms the logging off.
- The **Cancel** button cancels the logging off.

If the user has logged off manually, the **User Status** panel appears and the root menu and the alarm status panel will be locked. If logging on again, the user clicks the **Login** button in the **User Status** panel. After logging on again, the user can keep on working in the displayed PXM form.

Root Menu

The root menu consists of functionality folders, the **Index** folder, and the **Help** folder. The folders contain subfolders and links to PXM forms and to the CPI library. Expand or collapse the root menu by double-clicking on the folder, or by clicking on the plus (+) or minus (-) symbols, respectively.

Functionality Folders

Each functionality folder represents an area of Operation and Maintenance (O&M) functionality. Clicking on a link loads the PXM form for that functionality into the PXM main window.

Each functionality folder represents an area of Operation and Maintenance (O&M) functionality. Clicking on a link loads the PXM form for that functionality into the PXM main window.

The PXM does not cover all operations that are managed using the CLI. Existing PXM forms are listed below.

- Alarms and Events
- Equipment Management
- Measurement Jobs
- Measurement Types
- Software Configuration Management

- › All configurations cannot be performed with PXM GUIs.

 - › The following PXM GUIs can be used in the SGSN-MME:
 - Alarms and Events
 - Equipment Management
 - Measurement Jobs
 - Measurement Types
 - Software Configuration Management

Figure 3-10. Tasks to be performed with PXM

Index Folder

All PXM forms can be found in alphabetical order in the **Index** folder.

Help Folder

The **Help** folder contains a link to the CPI library, a link to this O&M description, and a link to Glossary of Terms.

PXM Forms

Each PXM form, from which the user operates on the GSN, is described in its own GUI User Guide. All GUI User Guides can be found in the CPI library.

Alarm Status Panel

The alarm status panel, which indicates the alarm status of the connected GSN, is viewed in later.

Results Viewer

The result viewer displays the results of each user operation.

There are four types of results:

- Immediate operation result, which indicates that an operation has started and been completed
- Ordered operation result, which indicates that an operation has started but not yet been completed

- Deferred operation result, which indicates that a previously ordered operation has been completed
- Operation error, which indicates that an operation has not been executed due to some sort of error condition

Command Line Interface (CLI)

The Command Line Interface (CLI) is the main user interface for configuration purposes and allows the operator to manage the GSN using commands of similar syntax to UNIX commands and provides for batch processing and mass configuration. The CLI commands are divided into groups depending on which area they are executed in and each group is described in its own CLI Description. All CLI Descriptions can be found in the CPI library. The **list_cmds** CLI command lists all available CLI commands. Also, they can be sorted by function with the **list_sort_cmds** CLI command.

CLI commands specific for GSM are only available on a SGSN-MME (G) and CLI commands specific for WCDMA Systems are only available on a SGSN-MME (W) and same applies for LTE.

Following a start, restart, or activation of a Software Configuration (SC), no CLI commands are available until the SGSN-MME has reached the node fully started state. If a user tries to run CLI commands before the SGSN-MME has reached the node fully started state, the system responds: CLI is currently not available.

Some CLI commands are not available right after the SGSN-MME has reached the node fully started state, in order to reduce traffic downtime. All CLI commands are available within 10 minutes after the SGSN-MME has reached the node fully started state. The time depends on the amount of configuration data. If a user tries to run CLI commands before they are available, the system responds: Command not found. Try later, restart is still in progress.

The GPRS shell (**gsh**) is the generic framework which provides support for the CLI. The shell is opened by typing **gsh**. The **gsh** command is applicable only in a UNIX shell.

CLI can be run in three modes:

- Interactive mode
- Single command mode
- Batch command mode

For CLI, the connection is made via TCP/IP and telnet/Secure Shell (SSH) to a SGSN-MME node. In the case of a Unix workstation there is also the ability to use rlogin

Usually you would access the active NCB board, executing the GPRS Shell. GSH is especially designed as the CLI interface to the SGSN-MME node itself.

- › Can perform operation and maintenance
- › Can be run from any OS that supports TELNET or RLOGIN
- › Log-onto Active NCB in SGSN-MME via OAM VIP
- › At prompt on Active NCB start GPRS shell with GSH command
- › CLI command “list_cmds” will list all the commands available to you through CLI.

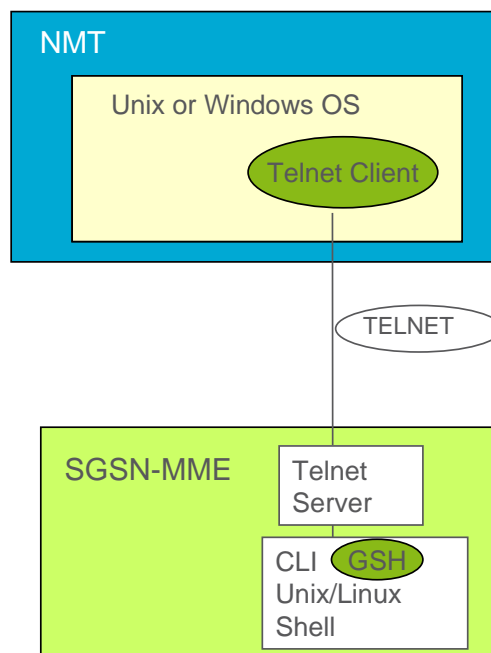
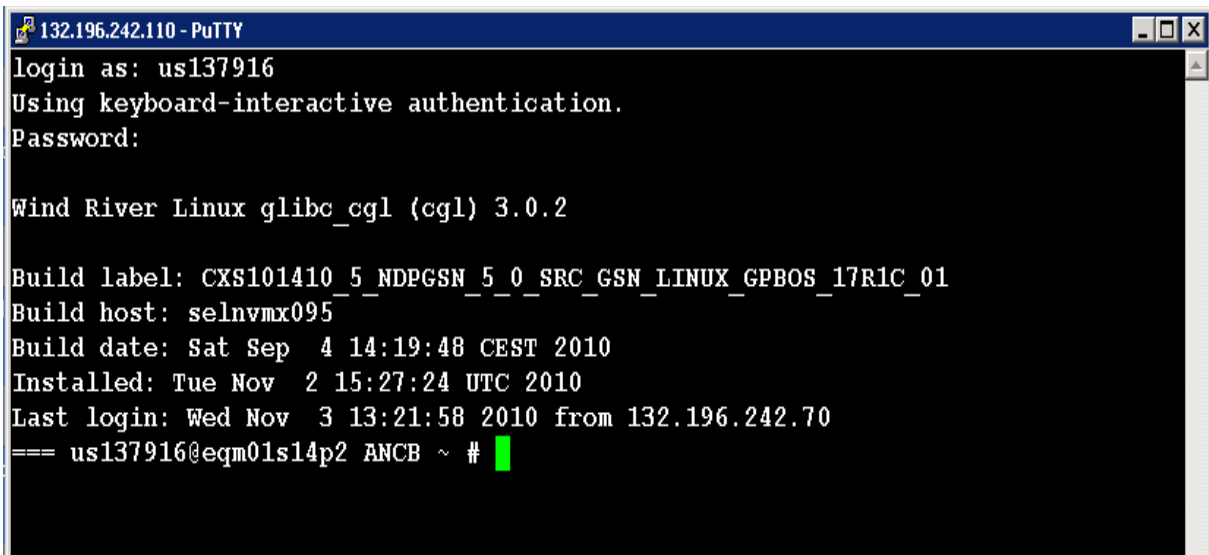


Figure 3-11: Command line interface (CLI)

- gsh list_imsins
- cli > list_imsins ← In gprs shell, no 'gsh' needed

- list_* ← Provides listing
- get_* ← Get detailed information
- create_* ← Create item
- delete_* ← Delete/Erase item
- modify_* ← Modify/Change item
- action_* ← Perform action

Figure 3-12. CLI Format



```
132.196.242.110 - PuTTY
login as: us137916
Using keyboard-interactive authentication.
Password:

Wind River Linux glibc_cgl (cgl) 3.0.2

Build label: CXS101410_5_NDPGSN_5_0_SRC_GSN_LINUX_GPBOS_17R1C_01
Build host: selnvmx095
Build date: Sat Sep 4 14:19:48 CEST 2010
Installed: Tue Nov 2 15:27:24 UTC 2010
Last login: Wed Nov 3 13:21:58 2010 from 132.196.242.70
=== us137916@eqm01s14p2 ANCB ~ #
```

Figure 3-13: Telnet example to the Active NCB

Interactive mode

Interactive mode allows execution of one CLI command at a time with no possibility of output manipulation. Enter the interactive mode by typing **gsh** and pressing **Enter**. To exit the interactive mode, press **Ctrl + D** or **Ctrl+C**.

Example of Interactive Mode:

```
# gsh  
cli> list_logs
```

Single Command Mode

The single command mode allows execution of one command at a time with the possibility of output manipulation. Use the single command mode by preceding each command with **gsh**. The advantage of the single command mode is that it fully supports command and argument expansion, as normally provided in a UNIX shell. There are no restrictions on command processing in single command mode in a UNIX shell.

Example of Single Command Mode:

```
# gsh list_logs
```

Example of Command Expansion with UNIX provided **grep** Command:

```
# gsh list_logs | grep chs
```

Batch Command Mode

The batch command mode allows execution of CLI scripts suitable for mass configuration. The CLI script is an executable text file. The file is made executable by the **chmod** UNIX command. Run the script by typing **SearchPath/FileName** and pressing **Enter**. The commands are used according to the same syntax as in single command mode. The commands will be executed one at a time. The execution proceeds with the next command, even if one command fails. An error message will appear for each failing command. UNIX environment variables are supported.

The batch command mode also supports the execution of CLI commands listed in a text file. There should not be more than one CLI command per line. The text file must end with a line break. Run the CLI commands by typing `gsh < FileName` and pressing Enter. The CLI commands will be executed one at a time. The execution proceeds with the next command, even if one command fails. An error message will appear for each failing command. Lines starting with % are comments and are disregarded. UNIX environment commands are not supported.

Example of a Script:

```
gsh list_logs | grep chs

gsh set_log_config chsLog -nf 124
```

Example of a Text File:

```
% list all log files

list_logs

% update the configuration of the charging log

set_log_config chsLog -nf 124
```

Unix prompt> `gsh < BatchFileName`

Or

Unix prompt> `./cli_script.cfg`

Each command in a batch file is executed one at a time even if one or more commands fail.

In the case of batch mode the operator must understand the syntax used within the batch file.

Figure 3-14. Batch file example

In the example shown above, the first line specifies that as each line is executed print the cli command to the screen.

The `GSH=/usr/bin/gsh` line specifies where the GPRS Shell executable is found. The following lines are assigning specific values to variable names.

The `$GSH create_plmn $PLM1 -mcc $MCC1 -mnc $MNC1 -fnn $FNN1 -snn $SNN1 -ci` is the actual cli command to be executed. This command is using the values assigned previously in the script file.

Command Syntax Description

The CLI command syntax is structured like the UNIX command syntax, but with enhancements specific to SGSN-MME to support the structure of the arguments. If the syntax of a command is unknown, type the command followed by any invalid operand, for example **-test**, and press **Enter**. An error message will appear, displaying the correct syntax for the command.

The structure of a CLI command consists of the command itself and is, in applicable cases, followed by operands and variables.

The following symbols may appear in syntax for CLI commands:

[] Includes optional items. The user does not type the brackets when executing the command.

{ } Contains items of a specific structure. These braces have to be typed in the command string.

| Mutually exclusive items. The user types one of the choices and not the symbol.

... The structure or item preceding these ellipsis points can be repeated one or more times. The user does not type the ellipsis points when executing the command.

System Response

An execution of active CLI commands, that is, CLI commands that create, delete, or change attributes, is accepted if no output response appears. Otherwise, an error message will appear.

An execution of passive CLI commands, that is, CLI commands that display attributes, is successful when it results in an output response. Otherwise, an error message will appear. The output response displays the current values of the parameters displayed by the passive CLI command. The displayed parameters can include parameters specified by the operands and variables included in the active CLI command syntaxes and internal non-configurable parameters in the SGSN-MME.

Naming Convention

The CLI command names follow a naming convention. There are two parts of the command, separated by an underscore. The first part of the CLI command shows the action of the command. The last part shows on which function or part the CLI command operates.

Examples of CLI Commands

The list below shows commands that create, delete, get, list, or set cooperating routing areas.

create_cra

delete_cra

get_cra

list_cras

set_cra

Syntax Error Messages

In the event of errors during the execution of a command, error messages are generated to the standard error (stderr). All internal errors (for example, database faults) will be logged. Illegal command syntax will cause error messages to appear on the screen.

The following error messages are used:

Illegal option -- xOption

Usage:

An unknown option has been used.

Unexpected argument – xArgument

Usage:

Some extra data was entered.

xCommand: Command not found

Usage:

The xCommand is not supported by GPRS CLI.

Command name is missing

Usage:

Missing command name

Insufficient arguments

Usage:

Some mandatory options are missing.

Unmatched xQuote

Command line contains unmatched quotes: double or single.

Syntax error

Usage:

General syntax error

Wrong argument type for option xOption

Usage:

The argument following xOption is of wrong type.

Wrong argument type

Usage:

An argument without an option (normally only one per command) is of wrong type.

GPRS CLI is not available

Node might be down. All error messages are internationalized.

4 Security Management

OBJECTIVES

Upon completion of this chapter, the student will be able to:

- › Explain the concepts of security management
- › Manage users in terms of creating, modifying and deleting users account
- › Assign the tailored roles for different users

Figure 4-1. Objectives

SECURITY MANAGEMENT

INTRODUCTION

The SGSN-MME provides a set of functions securing the GPRS O&M communication.

For performing O&M work on the SGSN-MME, the operator must log on with user name and password.

As a part of the O&M security the SGSN-MME also supports a role-based access control of the configurable resources. The access control is linked to the operator's user name and password, and can therefore be configured on individual basis. It is possible for a user to change the password through CLI.

The O&M security related functions that are implemented in the SGSN-MME are as follows:

- User administration

It is important to make sure that only the operator's staff can control and access to the SGSN-MME node. Also the operator might want to level access to grant access according to staff's skills or job position preventing errors and downtime by human mistake.

The SGSN-MME node implements a User authentication function. For any O&M access (e.g. PXM, ftp, telnet or ssh) to the network element a valid user name and password must be specified. The information about the login and login attempt is stored on the network element for later reference. Then authorization is also applied to the users logging in. A role based authorization model is used to level access and to limit the possibilities of configuration access.

- Service request log

Each command performed by the operator on the SGSN-MME is logged in the OMS_SM_Log for tracking purposes.

- Secure shell

Secure Shell (SSH) can be used to enhance security when remotely logging on to the SGSN-MME. SSH enables the use of CLI and File Transfer Protocol (FTP), Secure FTP (SFTP), and Secure Copy (SCP) without sending passwords, and other traffic, in clear text over the IP network, since all SSH traffic is encrypted.

SSH is used only for providing security for terminal logging on, such as CLI, FTP, SFTP, and SCP. SGSN-MME 2010B supports

SSH version 2 (SSHv2) only of the SSH protocol. To connect to the SGSN-MME through SSH, an SSHv2-capable client, such as OpenSSH, is required.

- Virtual Private Network

To protect O&M traffic, a separate Virtual Private Network (VPN) for O&M traffic can be used.

- IPSec

For improved protection of the O&M traffic, IPSec can be used over the VPN. The IPSec tunnel can be configured to terminate in a security gateway that separates the O&M transport network from the O&M management network.

- Firewall

The Object Request Broker (ORB) source port range is limited to 9900-9999. This makes it easier to identify IIOP traffic in the packet filters and firewalls. However, IIOP is an insecure protocol, and an intruder system can pass the firewall by knowing or guessing the names of the services or objects that may pass the firewall.

- Sandbox

For maximum protection, the PXM application should be configured to run in a sandbox environment, a small area within a computer where the PXM application can run unhindered, but where it is prohibited from running anywhere else. The sandbox can be placed within the Demilitarized Zone of a firewall. An IPSec tunnel can be configured between the SGSN-MME and the sandbox, and an SSH tunnel can be configured between the sandbox and the O&M workstation.

This chapter covers the basic Security Management (SM), user management in the SGSN-MME. Security Management is the Wireless Packet Platform (WPP) function, which is responsible for

- User authentication.
- Operations authorization in the SGSN-MME.
- Logging of all Operation & Maintenance (O&M) service requests.

- › “Security Management” is the Wireless Packet Platform (WPP) function which is responsible for:
 - user authentication
 - operation authorization
 - logging of all Operation & Maintenance (O&M) service requests

Figure 4-2. Introduction

The SGSN-MME platform is designed for packet data applications, for GSM, WCDMA and LTE, and it provides the foundation for the applications to run on.

The execution environment allows execution of SGSN-MME software on Power PC and UltraSPARC processors, using the industry-standard operating systems VxWorks, Linux, and Solaris. Designed for high availability, the execution environment is built on fault-tolerant components and allows change of software and hardware with limited impact on node performance. High availability is also promoted by using redundant hardware, and the platform allows load sharing over the available SGSN-MME resources. This has already been covered in detail in chapter 1 including different hardware configurations.

OPERATOR ACCESS

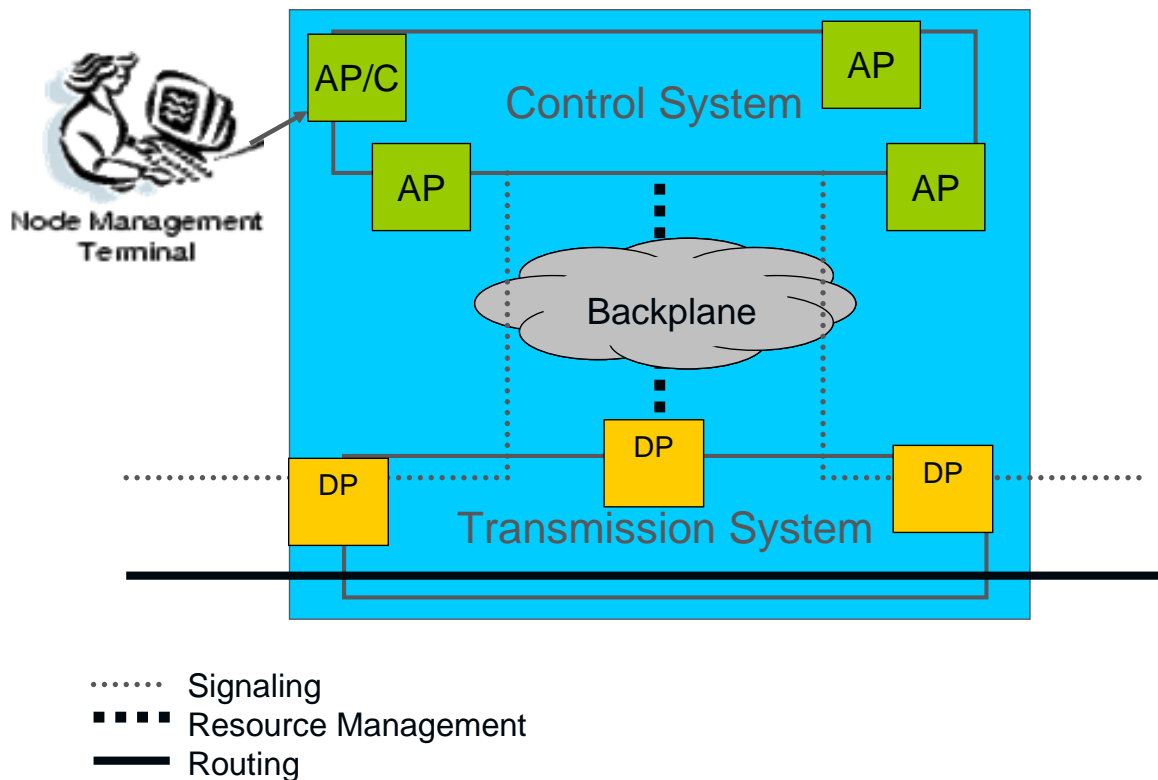


Figure 4-3. Operator Access

There are many ways to access the SGSN-MME. They are:

- Access through the GSS. This external server has a direct connection to the FSBs and from there you can access the rest of the node. This is normally used prior to O+M VIP being configured and sometimes for backup purposes.
- Access through the console port. Again like the access through the FSBs ethernet interface above, this is normally used when the node itself is not configured. It is normally used after the initial install or for troubleshooting purposes when access through the everyday connection is down or node is reloading. Only one user at a time can access the node through the console port.

- Access using OAM IP address. This is the interface that must be configured. It's the telnet address visible over the GOM network. It's the address that OSS will use to communicate to the SGSN-MME and the address that will be used by everyday operators of the SGSN-MME. It will give you direct access to the ANCB from where you perform all of the O+M activities.

USER AUTHENTICATION

The user authentication controls the access to the SGSN. The user authentication is done with an account (user ID) and password check. For this the Unix/Linux build in function is used. User authentication is performed for:

- UNIX/Linux and Vxworks environment logon.
- File Transfer Protocol (FTP) access and Secure FTP.
- Remote login access using telnet, rlogin, rsh and ssh.
- Hyper Text Transfer Protocol (HTTP)/IIOP access to Packet Exchange Manager.

- › User Authentication is used to control access to the SGSN-MME
 - Every user has a unique user name and a secure password to get access to the system
- › The Unix/Linux authentication function is reused for WPP
 - /etc/passwd contains user accounts
 - /etc/shadow contains encrypted user passwords
- › User Authentication is performed for
 - UNIX/Linux and Vxworks environment logon
 - FTP access and SecureFTP access
 - Remote Login access using telnet, rlogin, rsh and ssh
 - HTTP/IIOP access to Packet Exchange Manager

Figure 4-4. User authentication

OPERATION AUTHORIZATION

User authorization and access control are based on user roles and group membership. The group membership limits the access rights within the UNIX/Linux environment, whereas the user roles limit the access rights for PXM and CLI operations.

After a successful logon to the SGSN it is necessary to perform an authorization check on all O&M service requests to control the access to specific functions in the GSN.

An O&M service request can be an action initiated by the operator using Packet Exchange Manager (PXM) or a Command Line Interface (CLI) command.

Before any service request is executed, an authorization check is performed. The check is done in order to prevent unauthorized service requests towards the SGSN that were initiated accidentally or maliciously.

The authorization restriction applies to all Management Interfaces (MI) like PXM and CLI. For the user authorization a role-based control is used. This allows defining and assigning different roles to different types of users.

In Unix/Linux environment authorization is applied on file access permissions only as standard. The user's role is checked only when a CLI commands or PXM is used.

O&M SERVICE REQUEST LOG

All O&M service requests are logged in a log file. This ensures that all actions can be used for error tracking later on. The name of the log file is OMS_SM_LOG. The Log Viewer in PXM cannot be used anymore from SGSN R7 to retrieve information from the log file because the related form has been removed in this release. To view the file the normal UNIX/Linux commands can be used. The following information is recorded in human readable format:

- Service request
- Date and time of the service request
- User name
- User role

Please note that the result of the service request is not logged. The execution of Unix programs or scripts is not logged.

Security Management provides a Management Interface (MI) which is used for:

- User administration.
- Role administration.

The following sections present how the MI is used.

- › Authorisation is based on O&M service requests
 - A O&M service request is any action initiated with PXM or CLI
 - To each opened PXM form or issued CLI command authorization is applied according to the user's settings
 - All O&M service requests are logged in O&M Service Request Log (OMS_SM_Log)
- › The Authorization log OMS_SM_Log includes :
 - services requested
 - date and time of the requested service
 - user name and role originated the request
 - Successful and unsuccessful login attempts to PXM
- › The underlying Unix/Linux environment authorizes only on file permission and group membership basis
- › Unix/Linux Login attempts can be found in Unix/Linux log files only – Unix/Linux command execution and file modification is not logged

Figure 4-5. User authorization

USER ADMINISTRATION

Security Management offers a MI for the user administration in the SGSN. User administration can be done only by means of CLI.

There are two types of users on a SGSN:

- Plain Unix/Linux user

The plain Unix users that have access to the operating system and Unix services like FTP. An example, coreUser is the account used for loading software patches onto the node.

The plain Unix accounts neither have access to PXM nor can run CLI commands. Though the Unix command “gsh” can be executed any send CLI command will fail, raising an authorization fault.

These users can be created for machine accounts that fetch off files from the network element but don't need any configuration access. The plain Unix account is created using the Unix tools like "useradd" or Admintool.

- Full featured PXM user

The second class of user accounts is a "full featured" PXM account. These PXM users are also created as Unix accounts on the system. So they can also login to the Unix environment and do the same as a simple Unix user. But additionally they have rights to issue CLI commands or access PXM forms according to their authorization settings in WPP security management.

These types of users are created using CLI command. The administrator doesn't have to worry about the Unix account. The necessary Unix definitions are automatically done.

› There are two possible types of users on a SGSN-MME:

- A plain Unix/Linux user who
 - › may access via all Unix/Linux services - e.g. ftp, telnet or ssh
 - › may logon to the Unix/Linux operating system and run local Unix/Linux commands
 - › can't connect to PXM nor run a CLI command
 - › This type of user is created using the Unix/Linux command "useradd" see man page for closer information
- A full featured PXM user who
 - › may use all Unix/Linux services as the plain Unix User
 - › can access to PXM and CLI commands according to the account's authorization level
 - › This type of user is created using CLI commands

Figure 4-6. User types in SGSN

At initial start of the SGSN, the system is only configured with the default **om_admin** user and the **pdg_user**. The om_admin user has the user administration role, which only gives the authority to add and change user accounts. The pdg_user can only execute the Performance Data Collection (PDC) script in the toolbox.

The user **root** has been removed from PXM users since SGSN R7. This user still exists on the UNIX/Linux environment of the node.

User names and passwords are not part of the Software Configurations (SCs). This prevents the SGSN from reverting to an old user name and password at roll back to an earlier SC. However, the user names and passwords are part of the backup procedure.

- › Passwords on node level in UNIX/Linux.
- › Users and passwords are not part of software configurations.
- › The root user is removed from the type of PXM user.
- › Authority for user om_admin changed.
- › Create user with ConfigRole as first step after initial installation or upgrade.

Figure 4-7. User types

- › User Administration on the SGSN-MME can be done only with CLI.
- › The concept of group membership has been added in SGSN-MME for PXM users.
 - The group membership limits access rights within Unix/Linux environment.
 - The users can be designated a membership in one or more of the following groups.
- › After an initial installation “om_admin” and “pdc_user” PXM accounts are available on a SGSN-MME.
 - The om_admin user can only define new users.
 - The pdc_user user can only execute the Performance Data Collection (PDC) script in the toolbox.

Figure 4-8. User administration

Create a new user

When a new user is created, the system administrator has to set a password and a role. The password has to be 6 or more characters long, but only the first eight are considered and should contain special characters and digits. The system administrator can select a role from the list of existing roles. The data is stored in the Mensia database and the Unix configuration files. A home directory is created. The default location is */Core/home*. After the first logon the user can change the password.

The default location of all home directories in the Node Control Board (NCB) is */Core/home*. Also all needed entries in the Unix/Linux configuration files (*/etc/passwd*, */etc/shadow*) are made.

Delete a user

When a user is deleted, all user data in the Mnesia database and in the Unix configuration files are deleted. Also the home directory, all sub directories and all files in the subdirectories are deleted.

Assign a user to a group

The users can be dedicated a membership in one or more of the following groups, where the gsnuser is the default group. The group membership limits the access rights within the UNIX environment.

There are 3 groups supported in the SGSN:

- Charging group allows read and write access to the charging files.
- Security group allows read and write access to the user administration files.
- Gsnuser group allows read and write access to all files, except for the charging and user administration files.

Users must be designated and assigned passwords. New users are automatically assigned the gsnuser group. In order to access security- or charging-related log files, users need to be assigned the corresponding groups.

Change the role for a user

Security Management uses a role model. This means that every user is assigned a role. The role defines what kind of actions the

user can perform on a SGSN. The system administrator can change the role for a user all the time.

In order to perform user administration, a user needs to be assigned a UserAdmRole. For more information about the different roles see the section Role Administration.

Retrieve information about the last logon of the user

It is possible to retrieve information about every user. The user name and the related role are given.

The figure below shows the CLI commands for user administration.

- › Define user and password
 - Log in as user om_admin
 - add_sm_user -ui UserId -psw password
 - set_sm_roleToUser -ui UserId -role role
 - Log out
- › Optionally, assign a user to a group other than default group
 - list_sm_groups
 - set_sm_group -ui UserId -g UserGroup...
- › Optionally, define new role
 - list_sm_role [-role Role]
 - list_sm_actions
 - add_sm_role -role role [-a actions...]
 - add_sm_action -role Role -a Action...
 - rm_sm_action -role Role -a Action...
 - set_sm_roleToUser -ui UserId -role role

Figure 4-9. User administration: CLIs

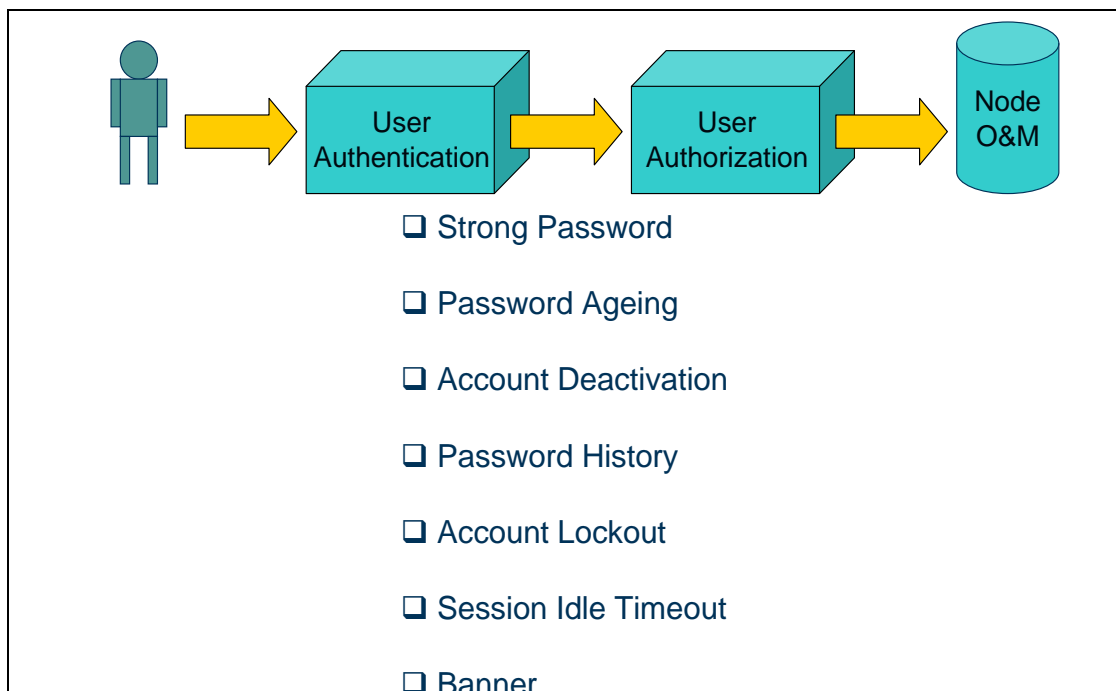


Figure 4-10. Security Management

- Global configuration settings and settings for an individual user can be retrieved with the command ***get_sec_conf***
- | user | pa | pat | ad | adt | ph | phr | phc | al | ala | ald | sit | sity |
|----------|-----|-----|-----|-----|----|-----|-----|-----|-----|-----|-----|------|
| (global) | on | 60 | on | 365 | on | 10 | 1 | on | 5 | 60 | on | 60 |
| elsa | off | 365 | off | 100 | on | 10 | 1 | off | 10 | 100 | on | 60 |
| maria | on | 1 | on | 0 | on | 10 | 1 | on | 1 | 0 | on | 60 |
- Individual user configuration settings can be retrieved with the commands ***get_user_sec_conf -ui Username***
 - There will also be a note in case the user is locked out by Account Lockout or Account Deactivation.
 - An account can be reactivated from Account Lockout or Account Deactivation by the command ***unlock_user -ui Username***
 - Configuration settings for an individual user overrides the global settings for applicable functions in Security Enhancement (i.e Password Aging, Account Deactivation and Account Lockout).
 - ***modify_sec_conf*** (not gsh command) to configure security settings

Figure 4-11. General CLI properties

ROLE ADMINISTRATION

User authentication is based on user roles. The user roles limit the access rights for PXM and CLI operations.

Only authorized users, that is, users with the user administration role, can change a user authority.

The user can manually log off from the system. An inactive SSH user is automatically logged off after a certain period.

Authorized users can also log off other users that appear to threaten the system. This is applicable for users logged on through PXM or as root in UNIX.

An authority change, logging on or off, and account deletion or creation is all registered in the security log.

Role Based Access Control

In the SGSN nodes, different operator categories or operator roles are defined, making it possible to differentiate users depending on tasks assigned and experience.

The Role Based Access Control (RBAC) is used for controlling the execution of operations on the Network Element. With RBAC each user is assigned one role, and each role is assigned one or more action sets that permit certain actions to do. One role may implicitly include operations that are associated with another role. Every action set belongs to one operation in the MI, for example, the definition of a cooperating SGSN.

A predefined system administrator function is provided in the user interface, CLI, in order to define operator roles and action sets to configure the roles thus defining who has read and write access to all action sets. Only one role per user will be allowed.

- › A Role Based Access Control (RBAC) model is used on the SGSN-MME.
- › Every user has a role with one or several “action sets”.
- › An action set is related to the related CLI commands on the SGSN-MME, e. g. adding a new IMSI number series.
- › This gives the system administrator the possibility to create tailor made roles for the different types of users
- › The roles limit the access rights for PXM and CLI operations.

Figure 4-12. Role Based Access Control - Authorization

Each role is defined by a collection of action sets, representing the authority to view or modify information by use of CLI commands or PXM forms.

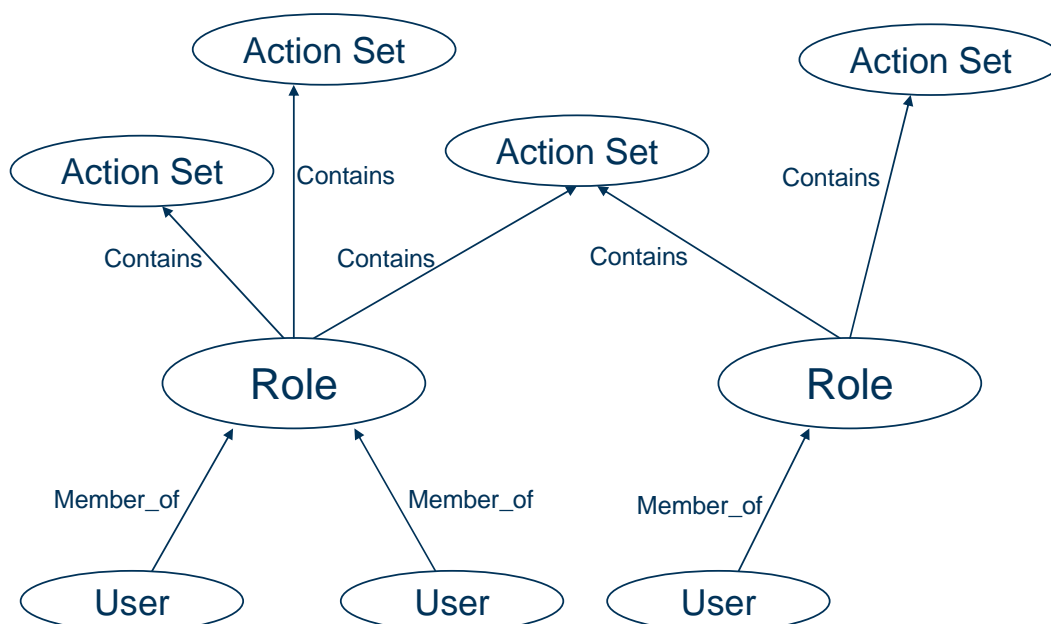


Figure 4-13. Role administration

There are two different types of action sets for each area of CLI commands, one passive and one active.

The passive action set is used by CLI commands or PXM form that retrieves data, while the active action set is used by CLI and PXM that creates and modifies data. When a role is assigned an active action set, the corresponding passive action set shall also be assigned to the role. If neither the passive nor the active action set is assigned to a role, none of the CLI commands for that specific area of CLI commands is accessible. Each CLI description lists the action set that is needed to run a specific CLI.

There is one current exception made for SS7 configuration:

There three action sets exist:

- `ss7_active` to modify SS7 configurations.
- `ss7_passive` to view SS7 configurations.
- `ss7_action_active` to initiate actions like blocking and deblocking of SS7 links

The complete list of action sets is given at the end of this chapter.

The action set **oms_sm_user_admin_passive** should always be included in every role. Without this role the users are unable to change his/her password.

As mentioned before RBAC allows the system administrator to create tailor made roles for the different types of user.

The users are also assigned a role that defines the users' authority. Six default roles are defined by the following system:

- The system administrator role **SysAdmRole**, which allows read and write access to the operations in all action sets.
- The system read only role **ReadOnlyRole**, which allows access to the operations in all passive action sets.
- The charging role **ChargingRole**, to handle just the charging configuration data.
- The configuration role **ConfigRole**, to handle general configuration data.
- The user administration role **UserAdmRole**, to handle the definition of new users.
- The performance data collection role **PdcRole**, to handle the performance data collection.

- › Usually two types of action sets exist for every operation on the SGSN-MME:
 - an active action set which gives the permission to configure the specific node data
 - a passive action set which gives the permission to view the specific node data
- › The following roles are available on the SGSN-MME:
 - The system administrator role *SysAdmRole*
 - The system read only role *ReadOnlyRole*
 - The charging role *ChargingRole*
 - The configuration role *ConfigRole*
 - The user administration role *UserAdmRole*
 - The performance data collection role *PdcRole*

Figure 4-14. Action set and user roles

- › Each role can be modified at any time with CLI commands.
- › A role name may contain any alpha numeric character but no special characters.
- › Note: The action set *oms_sm_user_admin_passive* should always be included in every custom role, otherwise users connected to that role can't change her/his password nor login via PXM.

Figure 4-15. Role administration

Example of a User Role

Let's assume that the system administrator has to define a role for a group of users. These users should be able to

- Define a new Iu interface on the GSN
- Modify and delete an existing Iu interface on the GSN

- Define a new Gb over Frame Relay (FR) interface on the GSN
- Modify and delete an existing FR Gb interface on the GSN
- View alarms and events in the alarm list
- Search for old alarms in the alarm log
- View subscriber data on the GSN
- Perform event recordings on the GSN
- Change her/his own password

- Let's assume the system administrator has to define a role for a group of users. These users should be able to:
 - Define a new lu Interface on the GSN
 - Modify and delete an existing lu interface on the GSN
 - Define a new Gb over Frame Relay (FR) interface on the GSN
 - Modify and delete an existing FR Gb interface on the GSN
 - View alarms and events in the alarm list
 - Search for old alarms in the alarm log
 - View subscriber data on the GSN
 - Perform event recordings on the GSN
 - Change her/his own password

Figure 4-16. Example of an user role (1/4)

- The new role has to contain the following action sets (1/3):
 - ip_passive, ip_active
To view and modify ip configurations (lu-U and lu-C)
 - ss7_active, ss7_passive, ss7_action_active
To allow access to SS7 interfaces (lu-C)
 - nwc_rnc_passive, nwc_rnc_active
To view/set RNC parameters
 - nwc_local_ra_w_passive, nwc_local_ra_w_active:
To set/change local routing area settings

Figure 4-17. Example of an user role (2/4)

- The new role has to contain the following action sets (2/3):
 - linkFrSLO_passive, linkFrSLO_active
View and configure the Frame Relay Fraction parameters
 - linkTsSLO_passive, linkTsSLO_active
View and configure the E1/T1 parameters
 - mts_Nse_passive, mts_Nse_active
View and configure NS entity
 - mts_Nsvc_passive, mts_Nsvc_active
View and configure NS virtual connection
 - nwc_ptpbvc_passive, nwc_ptpbvc_active
view and configure PTP BSSGP virtual connection
 - nwc_bsc_passive, nwc_bsc_active
View and configure the BSCs parameters

Figure 4-18. Example of an user role (3/4)

- The new role has to contain the following action sets (3/3):
 - oms_fm_admin_passive
To allow user to see alarms and events on the network element
 - vlr_subscriber_data_passive
To allow to read subscriber data
 - ncs_EventRec_active, ncs_EventRec_passive
To allow to read and change/start event recording
 - oms_sm_user_admin_passive
To allow user to change his/her password

Figure 4-19. Example of an user role (4/4)

Note that no specific action set is required for retrieving the alarm log files.

CLI COMMANDS

- add_sm_action: Setting Actions to a Role
- add_sm_role: Creating a New Role
- add_sm_user: Creating a User
- delete_sm_role: Deleting a Role
- delete_sm_user: Deleting a User
- forced_sm_unlock: Unlocking the Node
- list_sm_actions: list security management actions
- list_sm_groups: list security management groups
- list_sm_role: list security management roles
- list_sm_users: list security management users
- rm_sm_action: Removing Actions from a Role
- set_sm_group: Assigning Groups to a User
- set_sm_passwd: Change Password
- set_sm_roleToUser: Assigning the ReadOnlyRole to a User
- set_sm_specified_user_passwd: Change Pwd for Any User

Figure 4-20. CLI commands

List of Action Sets

Action sets specify the authority to manage a functional area with CLI commands and PXM forms. The CLI commands for a functional area are described in one or several CLI Description documents. In the CLI descriptions the action set for each command is specified. The passive action set allows retrieval of data and the active action set allows the modification of attributes. For a complete list of the action sets available, refer to the SGSN 2010B CPI.

Intentionally Blank

5 System Administration

OBJECTIVES

Upon completion of this chapter, the student will be able to:

- › Identify the severity of a fault in the SGSN-MME and act according to the escalation procedure.
- › Handle the Fault Management with PXM and CLI.
- › Identify different software management on SGSN-MME and the various methods of creating a Software Configuration (SC).
- › Detail and perform Checkpoint operation.

Figure 5-1. Objectives

FAULT MANAGEMENT

INTRODUCTION

The fault management functions in SGSN-MME handle the notification of alarms and events to the operator. This includes the sending of alarms and events and the clearing of alarms.

Alarms and events are defined differently. An event is defined as an informative notification from the SGSN-MME and requires no operator manual intervention and thus no corrective action. An alarm is an indication that a fault has occurred. The alarm normally requires a manual intervention where the operator resolves the fault. A resolved alarm normally clears automatically, but it is also possible for an operator to force clearance of an alarm.

The SGSN-MME keeps a list of all currently active alarms and one of the most recent events. These lists are cleared at a node restart.

The alarms and events can be subscribed to from both a Node Management Terminal and from an Operational Support System (OSS).

The Packet Exchange Manager (PXM) which runs on a Node Management Terminal allows the operator to define filters for the alarm and event list. The filters define which alarms and events will be shown to the operator in PXM. The Command Line Interface (CLI) allows the operator to define filters for the alarms and events that are sent to an external fault management system, for example, OSS or NMS (Network Management System).

Alarms and events are logged separately.

The purpose of the Fault Management function is to minimize the effects of such failures in terms of quality of service experienced by the network user. Therefore the Fault Management function should:

- Detect failures in the network as soon as they occur and alert the operating personnel as fast as possible.
- Isolate the failures (autonomously or through operator intervention). That means switch off faulty units and, if applicable, minimize the effect of the failure by reconfiguration of the faulty unit.
- If possible, determine the cause of the failure using diagnosis and test routines.

- Repair/eliminate failures in due time using maintenance procedures.

OPERATION AND MAINTENANCE ACCESS

An alarm indicates the existence of a fault in the SGSN-MME. The alarm exists as long as the fault persists. When the fault is resolved, the alarm is automatically cleared. SGSN-MME notifies the operator when a fault has been detected and when it has been removed. The function also informs the operator about events.

Alarm and event subscription is possible over the Internet Inter-ORB Protocol (IIOP) interface, to a Node Management Terminal, and through Simple Network Management Protocol (SNMP), to OSS see the figure below. The Gom interface is defined for operation and maintenance traffic. When an alarm or event is triggered or when an alarm is cleared, the SGSN-MME immediately sends a notification to all operators that subscribe for this service. It also offers the possibility retrieve the current active alarm list.

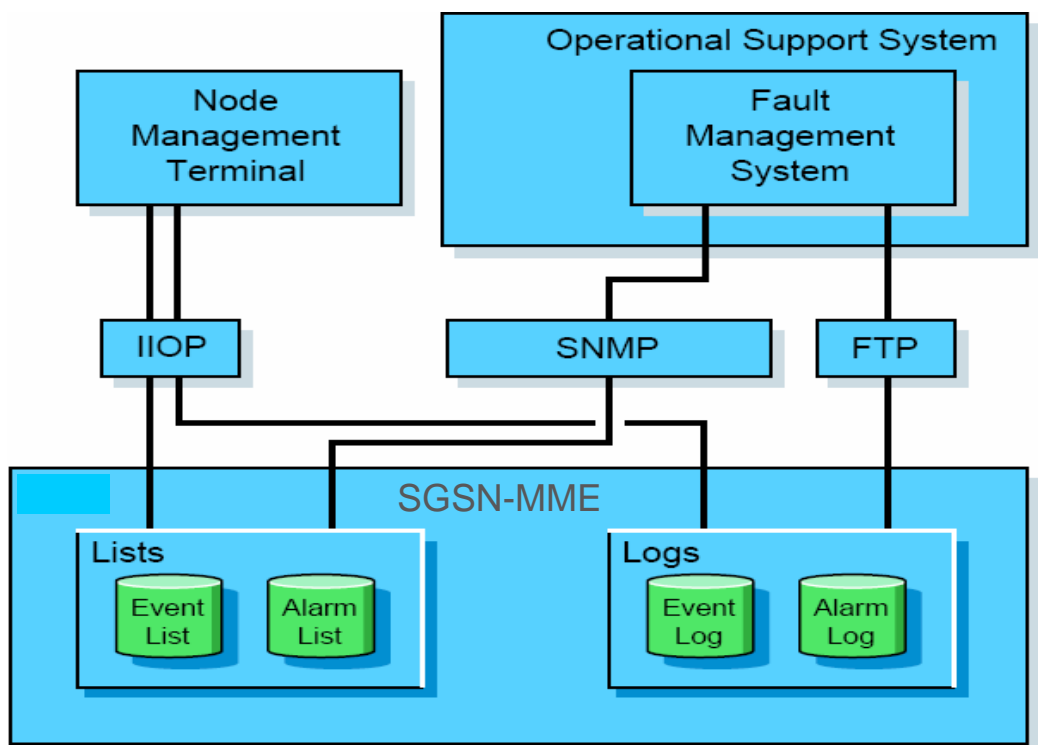


Figure 5-2. Fault Management in SGSN-MME

The fault management functions provide the following data, stored in the SGSN-MME:

- **Subscriber data** :Address, protocols, and filter data for each operator

- **Alarm list** : Currently active alarms in the GSN
- **Event list** : Latest events
- **Alarm log** : Logs in the file system
- **Event log** : Logs in the file system

FAULT MANAGEMENT LIMITATIONS

For greater usability and efficiency, fault management introduces limits on the size of its data structures.

When fault management has reached the limit of 500 active alarms, the 500th alarm indicates that the alarm list is full and no more alarms are accepted. New alarms are then logged but otherwise discarded until the number of active alarms has decreased to 450.

Fault management maintains a circular list of the 500 most recent events. This means that no regular overflow in the event list is possible. However, event flooding will cause a short history because of the circular nature of the event list.

ALARM AND EVENTS

The SGSN-MME informs subscribed clients that a fault has occurred and that something important has taken place by forwarding alarm and event notifications.

The send alarm and the send event functions take the following actions:

- Adds the alarm or event locally to the active alarm list or event list.
- Checks with the subscription list and the filter list, which subscribers that shall have the notification and then sends the notification to the correct destination.
- Logs the alarm or event locally in the alarm or event log.

Alarm and Event Lists

The GSN keeps a list of all currently active alarms and a list of the last occurred events.

A new alarm is added to the active alarm list and cleared when the fault is resolved. All active alarms are automatically cleared from the alarm list when the content of the database is lost during a node restart. GSN adds the most recent events to the list of events. The oldest event in the event list will be overwritten when the list of events reaches the maximum number of allowed events.

The alarms and events can be viewed through the **list_alarms** CLI command, the **list_events** CLI command, or the Alarm and Events PXM form.

Synchronization of Alarm and Event Lists

The operator can demand a synchronization of the active alarm list and the event list in order to make sure that the information displayed is correct. The content in PXM and OSS is updated. This can for instance be suitable when the link between the GSN and the fault management system has been out of order.

Heartbeat Notification

Heartbeats are broadcasted from the GSN for link supervision between the GSN and the fault management clients.

Every minute, the GSN sends the current local time as a heartbeat. If the link between the GSN and the fault management client has been lost, the operator can synchronize the active alarm list and the event list when the link is up and running again.

Alarm and event logs

GSN contains one log for alarms and one for events. The alarm log (fm_alarm) persistently stores alarm notifications and alarm clear notifications. The event log (fm_event) persistently logs all event notifications. The logs give a complete picture of the alarm and event history.

The log files are so-called multiple logs, which means that they are written in a circular manner. When the logs wrap around the oldest entries are overwritten first. Both logs can be configured independently, through the logging support.

All alarms and events are logged independently of any filters.

The alarm log is updated in the following situations:

- Detection of a new fault situation and an alarm is raised.
- Clearance of an alarm.

- Forced clearance of an alarm, which means the operator clears the alarm.

The event log is updated when GSN detects a new event and an event notification is sent.

The alarm and event logs can be listed through the **list_logs** CLI command. The log files can be fetched through the File Transfer Protocol (FTP). The log files are plain ASCII files with a format that can easily be post processed.

OPERATOR NOTIFICATIONS

The GSN informs subscribed clients (see below) of alarms and events by forwarding alarm and event notifications. They include the following information:

Name: The name of the alarm or event

Category: An alarm or event classification, see the Alarm and Event Categories below

Time: The time the alarm or event occurred

Probable Cause: The probable cause characterizes alarm. The information on probable cause is only available in the SNMP trap sent to OSS. Probable cause is only applicable for alarms.

Severity: The severity level, as proposed in the ITU X.733 standard.

Source ID: The sender of the alarm or event

Fault ID: A unique alarm identifier, not visible in the PXM but in log files and in the CLIs

Display String: A short textual explanation of the event or alarm

Alarm and Event Categories

The categories are defined into several classes, all according to the standard Alarm reporting function.

Communications: Associated with the procedures and processes required to convey information from one point to another

Quality of service: Influence on the quality of a service

Processing: Associated with the software area or data processing

Equipment: Associated with hardware

Environmental: Associated with a condition related to an enclosure in which the equipment resides.

Alarm and Event Severity

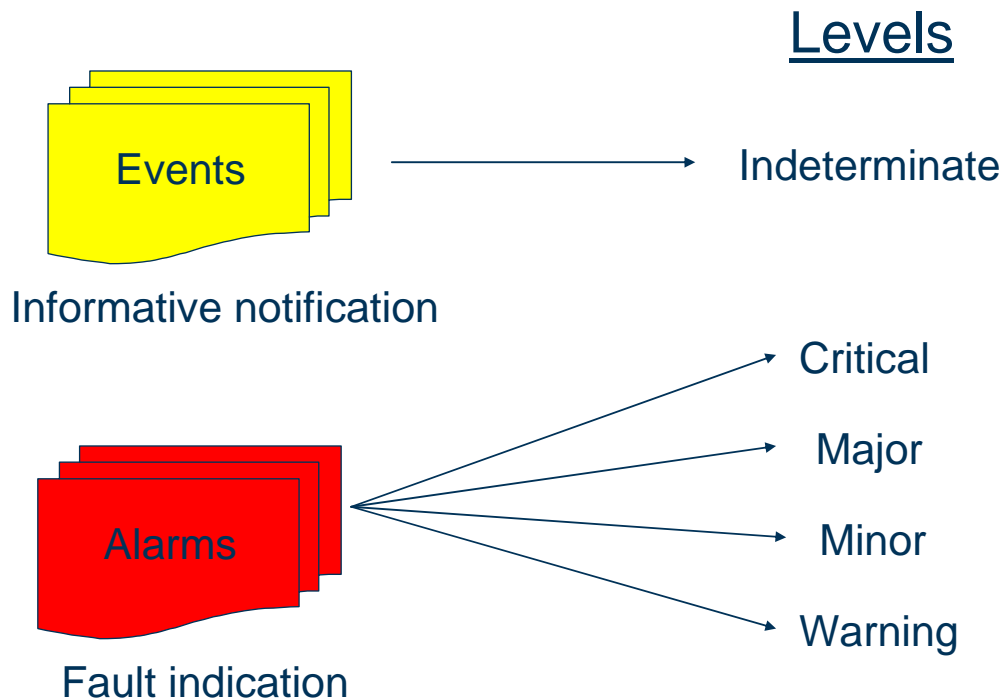


Figure 5-3. Alarms and Events

Alarms and events have grades of severity. By definition events are purely informative and do not have to indicate any fault situation. Consequently, the events may also have the severity level Indeterminate in addition to the severity levels, which are valid for alarms.

In descending order of severity, alarms are graded as follows:

Critical: The severity level critical indicates a condition that affects service and requires an immediate corrective action. An object that goes totally out of service is deemed critical.

Major: The severity level major indicates a condition that affects service and requires an urgent corrective action. An object whose capacity degrades significantly is deemed major.

Minor: The severity level minor indicates a condition that does not affect service. An alarm of this severity requires a corrective action to avert a more serious fault that affects service. An object whose capacity does not degrade is deemed minor.

Warning: The severity level warning indicates the detection of a potential or impending service affecting condition, before any significant effects detected

Indeterminate: Indeterminate means no severity level indication. This level can only be used for events.

Probable Cause

The probable cause characterizes alarm and provides further information than alarm Category. Valid values are defined according to the standard Alarm reporting function, ITU X.733.

Alarm Examples

The following is an example of a generic alarm message:

```
"The filename log file has reached its  
maximum size"
```

Alarm types and event types can also be defined exclusively to the different applications.

The following is an example of a specific alarm message:

```
"The /usr/logs/pm_log3 log file has reached  
its maximum size"
```

However, there can be several alarms or events using the text in the generic alarm message, because new versions of the file are created continuously. What makes a specific alarm or event unique is the identity, which is not included in the text part of the message.

Each alarm is described in a document entitled as the alarm. Refer to these documents for further information about the alarms.

fmUnknown (Alarm)

This alarm is generated when no definitions for a certain alarm can be found in the fm_alarm_def database.

The characteristics of the alarm are:

Name: fmUnknown

Severity: Minor

Category: Processing

Probable Cause: The following are the most likely reasons for this alarm:

- There may be a design error.
- The databases containing the alarm definitions are not loaded correctly.
- The databases containing the alarm definitions were not properly generated

Display String: Unable to look up notification definition in database. The original notification data was: alarm.

Actions to Be taken: Write a Customer Service Request (CSR).

fmAlarmListOverFlow

There is a large amount of alarms in the system and these must be dealt with. This alarm would not be raised under normal conditions.

The characteristics of the alarm are:

Name: fmAlarmListOverFlowUnknown

Severity: Critical

Category: Processing

Probable Cause: The following are the most likely reasons for this alarm:

- This alarm is generated when there is a severe technical problem in the network element.
- The fault management system will no longer try to display all alarms as this consumes processing power, which the system now needs.

Display String: There are more than 499 active alarms. No further alarms will be added to the list.

Actions to Be taken: Take measures so that at least 50 of the alarms are cleared automatically. (When there are less than 450 alarms in the list, this alarm message is cleared.). Otherwise the active alarm list is no longer updated.

Alarms are cleared when the fault situation no longer exists. Normally, the alarm software *automatically* clears the alarm when the fault has been solved. You may clear alarms with the Clear Selected Alarms function if the problem is not relevant, but this should rather be an exception than the rule.

fmUnknown (Event)

This event is generated when no definitions for a certain event can be found in the fm_event_def database.

The characteristics of the alarm are:

Name: fmUnknown

Severity: Indeterminate

Category: Processing

Probable Cause: The following are the most likely reasons for this alarm:

- There may be a design error.
- The databases containing the event definitions are not loaded correctly.
- The databases containing the event definitions were not properly generated

Display String: Event is not defined in the database. This is an installation error.

Actions to Be taken: Write a Customer Service Request (CSR).

ALARM AND EVENT SUBSCRIPTIONS

The alarms and events can be subscribed from both the Node Management Terminal and from the OSS. When an alarm is raised or cleared and when an event is sent, an immediate notification is sent from the GSN to all subscribing clients.

Technical Specification

The clients subscribes to alarm and event information over one of the following protocols:

- **SNMP:** The Simple Network Management Protocol (SNMP) interface is intended for external management clients. It cannot be used to configure the GSN. It offers only the possibility to get notifications when the active alarm list or the event list is updated with new alarms or events or when an alarm is cleared, see Alex documentation for SNMP Configuration.
- **IIOP:** Internet Inter-ORB Protocol (IIOP) is a communication protocol for PXM. At startup, a PXM client is automatically added to the list of alarm and event subscribers, and the subscriber automatically gets notifications when the active alarm list and the event list is updated with new alarms or events or when an alarm is cleared.

ALARM AND EVENT FILTERINGS

The operator can through filters define which alarms or events to include in a notification subscription. Filtering is done to reduce the load on the operation and maintenance network.

The operator chooses which alarm types and event types to view through the filter function. Included alarms will pass the filter. Excluded alarms will be filtered out.

Forwarding Alarms and Events

The configuration of which alarms or events to forward to an external fault management system, for example, OSS or NMS is done through the CLI commands.

Displaying Alarms and Events

The configuration of which alarms or events to display in the PXM alarm viewer is done in the alarm filter window.

The alarm filter window in the PXM includes four tabs: Alarms, Alarm Filter, Events, and Event filter. The Alarm Filter and the Event Filter tabs list all possible alarms and events.

Filtering could be done by the following criteria:

Name: Allows the operator to include or exclude alarms and events by name.

Category: Allows the operator to include or exclude all alarms or events for a specific category.

Severity: Allows the operator to include or exclude all alarms or events for a specific severity.

All Entries: Allows the operator to include or exclude all alarms or events

Sort by is an additional editing function, which allows the operator to sort alarm or event types based on name, category, severity, or selection. The sorting and presentation is done locally in each filter tab in the PXM form.

Note: No alarms or events are removed by sorting; they are just more suitably presented in the tabs.

The list of active alarms is automatically updated as soon as a new alarm arrives or an alarm is cleared. For this list a filter function can be applied to display only a selected number of alarms in the PXM. There are no CLI commands for filtering the output from `list_alarms` and `list_events`.

Note: By default, all alarms and events are always added to the alarm and events lists. It is not possible to switch off the sending of alarms. The operator chooses which alarms and events to view through the filter function.

The filter function can optionally:

- Filter out alarms or events raised before a certain timestamp
- Include or exclude individual alarms and events
- Include or exclude alarms and events by category or severity

Personal filter settings, for displayed alarms, and for each operator connected to the GSN are only applicable through PXM, see the Alarms and Events PXM form later. Operators connecting through SNMP share one filter.

Fault management logs all events and alarms to files independent of any filters. These log files can also be viewed and accessed through the File Transfer Protocol (FTP).

Help On Selected Items

To obtain more information, select an alarm or event in the Alarm and Event PXM form, and click Help in the menu to get the applicable alarm or event description from the ALEX library. The description will give the following information: explanation, background, proposed solution, and consequences of the displayed alarm or event. The help function is only applicable through PXM.

ALARM CLEARING

The alarm clearings are logged in the alarm log.

Automatic Clearing of Alarms

The GSN informs the operator that a fault situation is cleared and that the function is back in normal operation. When the cause of an alarm has been resolved, the GSN automatically removes the alarm from the active alarm list, and informs all subscribers that the alarm has been cleared.

Note: The active alarm list and the event list will be cleared at a node reload. After the restart, new alarms will be created indicating remaining faults.

The clear alarm function takes the following actions:

- Removes the alarm locally from the active alarm list.
- Checks with the subscription list and the filter list, which subscribers that shall have the notification and then sends the notification to the correct destination.
- Logs the alarm clearing in the alarm log.

Forced-Clearing of Alarms

Occasionally, the operator can force-clear an alarm that is not cleared automatically. This is however considered to be an abnormal case.

The function removes the alarm from the active alarm list. The forced clearance of the alarm is logged in the alarm log. A clear message is sent to all subscribing clients. Clearing an alarm is possible through **clear_alarms** CLI command or the **Alarms and Events PXM** form.

ALARMS HANDLING WITH PXM AND CLI

ALARM STATUS PANEL

The alarm status panel, which indicates the alarm status of the connected SGSN-MME, is viewed below:

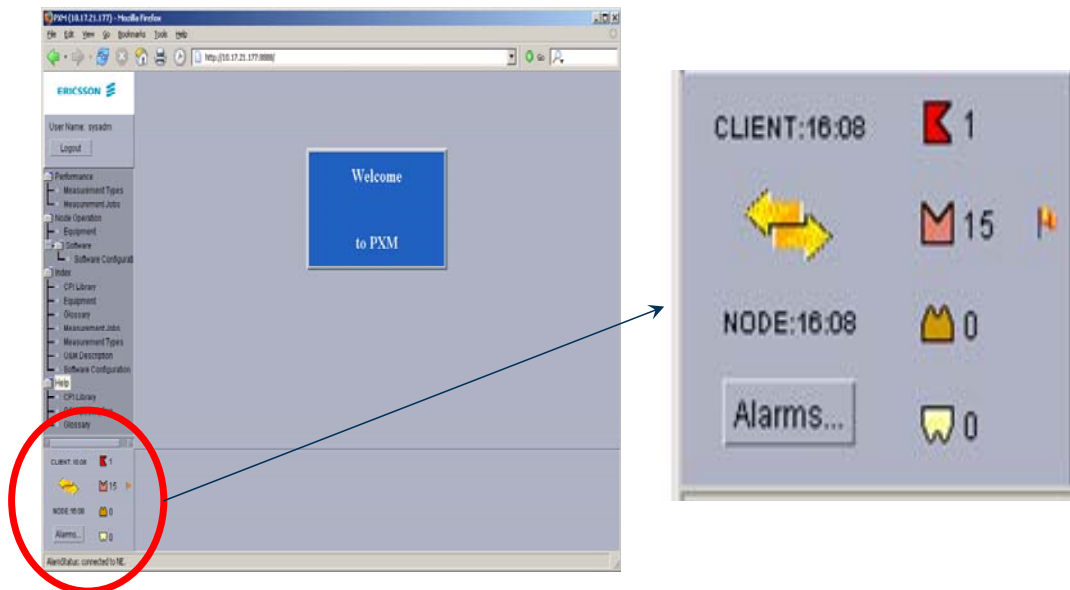


Figure 5-4. PXM Alarm window

The alarm status window contains the following information:

- 1 The **CLIENT** time indicator indicates the local time on the node management terminal.
- 2 The two-way arrows indicate whether heartbeat supervision is successful. When heartbeat supervision fails, the arrows have a black cross through them.
- 3 The **NODE** time indicator indicates when a heartbeat was last received. The heartbeat lets the PXM know when the GSN is functioning by sending a notification from the GSN to the PXM every minute. The indicated time comes from the NTP server. If no time has been received from the GSN, the text --:-- is displayed. This should only be displayed during the first minute after logging on to PXM.

- 4 The **Alarms** button opens the **Alarms and Events PXM** form in a new browser window. In this PXM form, the active alarms and recent events can be viewed and alarms can be cleared. Configurable filters determine the types of events and alarms to be viewed.
- 5 There are four alarm severities represented by standard symbols. From top to bottom they are critical, major, minor, and warning, followed by the number of active alarms for each category. The displayed number is affected by the user's current filter, which is possible to configure in the Alarms and Events PXM form.
- 6 Flags appearing to the right of any category indicate that new alarms for the respective category have arrived since the **Alarms and Events PXM** form was last opened.

The Alarm and Event form can be displayed by choosing the **“Alarms...” button** in the Alarm and Events status window on the bottom left in the PXM menu.

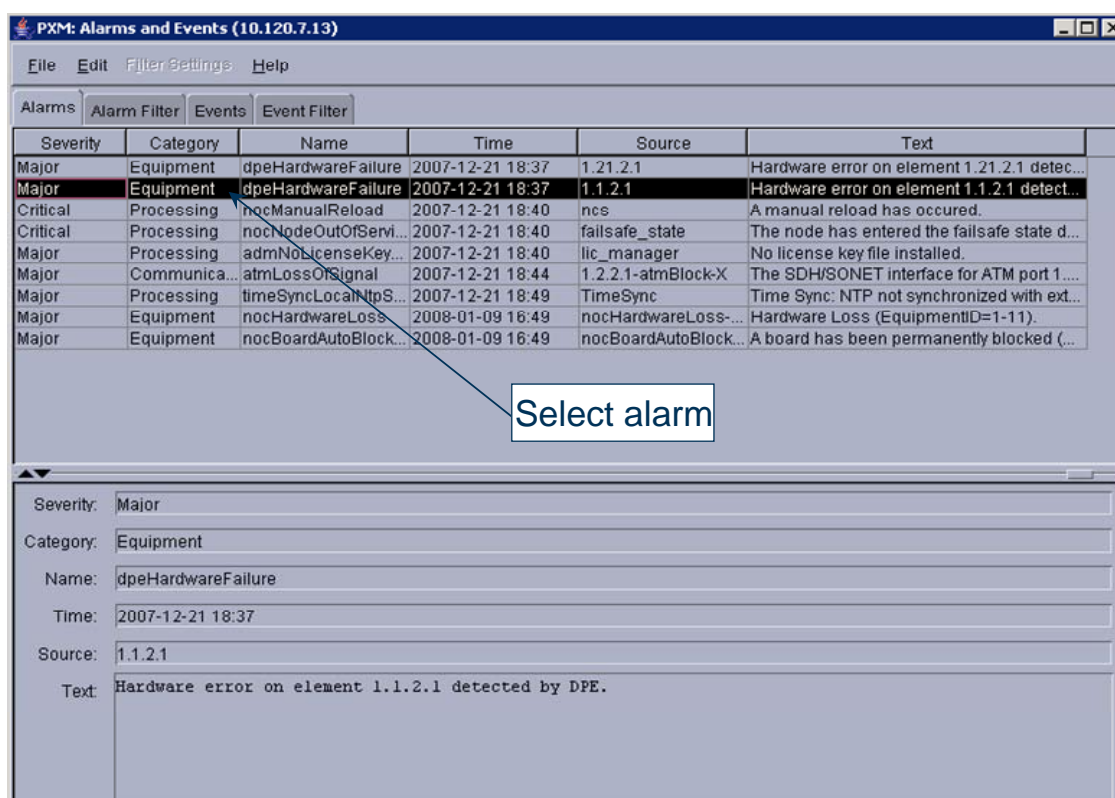


Figure 5-5. PXM Alarms and Events window

In this window the following tasks can be performed:

- View active alarms or recent events

- Determine what event types to include in the list of recent events
- Determine what alarm types to include in the list of active alarms
- Clear alarms
- Perform correct action on an alarm
- Filter the list of active alarms and events

In the PXM menu two different notifications can be present, *alarms* and *events*.

The data is displayed in column format in a table containing the following columns:

- **Severity** – this column displays the severity: critical, major, minor or warning
- **Category** – this column displays the following categories:
 - Communication.
 - Quality of service
 - Processing.
 - Equipment.
 - Environmental.
- **Name** – this column displays the alarm or event name.
- **Time** – this column displays the time the alarm or event was received.
- **Source** – this column displays the alarm or event sender source-id.
- **Text** – this column displays the text or comments issued by the alarm or event sender.

RESOLVING THE ALARM

Some alarms are automatically resolved by the SGSN-MME, but most alarms require operator interaction.

Determine the cause of the alarm, by consulting the appropriate Alarm and Event Description (AED) and follow the instructions in the section Actions to Be Taken.

CLEARING THE ALARM

After the cause of an alarm is rectified, the SGSN-MME clears the alarm from the alarm list and logs the clearing in the alarm log. If this fails, the alarm should be cleared manually.

Clear the alarm, by using the **clear_alarms** CLI command or the **Alarms and Events PXM** form.

SOFTWARE MANAGEMENT INTRODUCTION

Software management concerns the management of Software Configurations (SCs), Node Delivery Packages (NDPs), and Correction Packages (CPs).

Software management provides the following different functions:

- Software Installation
- Software upgrade
- Software update and patch
- Hardware upgrade
- Dual Access Reconfiguration
- SS7 Bearer Reconfiguration
- SAU Capacity Expansion
- Traffic Mix Optimization
- Software configuration management
- Interface Reconfiguration
- Backup and Restore

- The software management concerns the management of Software Configurations (SCs), Node Delivery Packages (NDPs), and Correction Packages (CPs).
- The software management function includes the following operations:

Term	What	How	Handled by
SW Upgrade	Change to a new release	Installing a new NDP	Ericsson personal
SW Update and Patch	Correction	Install a correction package (CP)	Explained detailed here
HW Upgrade	Change to a new HW deployment	A new NDP and HW upgrade kit	Ericsson personal
Dual Access Reconfiguration (DAR)	Change Access Type	Redeploy the running SC	Ericsson personal
SS7 Bearer Reconfiguration	Change SS7 Bearer	Redeploy the running SC	Ericsson personal
SAU Capacity Expansion (SAUCE)	Add more PIUs for higher capacity	Expansion kit + cli modify_sau	Ericsson personal
Traffic Mix Optimization (TMO)	Change the role of some PIUs	CLI modify_tmo	Explained detailed here
Software Configuration Management	Operate the SC	CLI checkpoint, etc	Explained detailed here
Interface Reconfiguration	Higher capacity and go to all_ip	All interfaces via IP routers	Ericsson personal

Figure 5-6. Software Management Introduction

SOFTWARE INSTALLATION

The software is installed with the GSN Support System (GSS), which is an external server and thus not part of the SGSN-MME product. The GSS server acts as a boot and installation server for initial installation and for software upgrades that include major changes of operating system. There are no specific requirements for the GSS, other than that it requires support for File Transfer Protocol (FTP), Trivial FTP (TFTP), Dynamic Host Configuration Protocol (DHCP), Time Port Service, and Network File System (NFS) services. The GSS also requires a Unix-based OS.

The software is delivered as a Node Delivery Package (NDP). During installation the NDP is configured for the specific hardware configuration, node type being GSM, WCDMA, or both, TMO deployment, SS7 signaling standard, and SS7 bearer type.

Regarding OSs, Solaris is used for MkIV configurations, while Linux is used for MkV and MkVI configurations. The software delivery package for SGSN 2008B and onwards is common for both Solaris and Linux OSs, rendering the possibility to install the delivery package on any OS.

SOFTWARE CHANGE

Ericsson employs the concepts SW upgrade, SW update, and SW patching when discussing software change methods for the SGSN-MME.

SW upgrade

SW upgrade is defined as a functional change of the software between two product releases, for example, SGSN R7 to SGSN 2008B. Hence, the SW upgrade involves a major software change with significant additions to the functionality.

A software upgrade includes installing a new NDP and sometimes updating the OS. Reconfiguration of the node may be necessary.

Ericsson engineers execute the software upgrade process usually so this is not subject of this course.

SW update

SW update is defined as a correction package of the software within a product release. A typical example of a SW update is the maintenance package delivered at regular intervals to correct software faults and introduce minor adaptations of the software. SW updates are characterized by fast deployment and minimal downtime.

The SW update is delivered as a SuperCP, and can be activated with limited disturbance to the node.

SW patch

SW patch is defined as a correction of a specific fault, that is, an emergency correction. The SW patch is delivered as a SuperCP, and can be activated with limited disturbance to the node.

A Correction Package (CP) contains updated software. A CP is distributed in a deliverable software package called a SuperCP. A SuperCP can contain one or more CPs. A SuperCP is installed to correct errors in the software running on the SGSN-MME, or to bring the software on the SGSN-MME to the latest software level.

SuperCPs are referred to as software patches, and the process of applying a SuperCP to an SC is called software patching.

Both SW updates and patches can be loaded from the CLI, PXM or OSS-RC (SMO), over the Gom interface, and no support is needed from the GSS. The software can be downloaded to the SGSN-MME over a secure IP network.

The SMO provides additional support for scheduling of SW updates and SW patching.

HARDWARE UPGRADE

A hardware upgrade adds or improves functionality from one hardware product release to a later one, for example, MkIV to MkV. It also corrects known hardware-related errors.

A hardware upgrade includes replacing old low capacity hardware with improved hardware, or installing hardware with new functionality. Conversion of the node configuration is necessary. A hardware upgrade may require a previous software upgrade to the corresponding product release.

Hardware upgrades are usually executed as part of a software upgrade. These tasks are executed by Ericsson personal so this is not subject of this course.

DUAL ACCESS RECONFIGURATION

Dual Access reconfiguration is defined as change of access mode, for example, going from a GSM-configured node to a Dual Access-configured node. Configuration data is reused where applicable, and new configuration connected to the added radio access type is needed.

SS7 BEARER RECONFIGURATION

SS7 Bearer Reconfiguration is performed on SGSN-MME 2010B, in which an SS7 bearer type is either added or removed so that SS7 bearer type changes between Narrowband (NB) and Broadband (BB), ss7 standard also changes.

SAU CAPACITY EXPANSION

A two-magazine SGSN-MME not fully equipped with General Processing Board (GPB) PIUs at purchase, can at a later point be accommodated with additional GPBs in order to expand the capacity regarding Simultaneously Attached Users (SAU), Packet Data Protocol (PDP) contexts, and throughput. For the same purpose, an SGSN-MME with two fully equipped magazines can be accommodated with a third magazine containing GPBs, Interface Board (IBxx) PIUs, and Power and Ethernet Board (PEB) PIUs. The procedure of installing and activating additional PIUs into a less than fully equipped SGSN-MME, is referred to as a capacity expansion.

TRAFFIC MIX OPTIMIZATION (TMO)

Using TMO, the roles of some of the Plug-In Units (PIUs) can be reassigned, being Application Processors (APs) or Device Processors (DPs). Reassigning of roles makes it possible to optimize the SGSN-MME signaling and payload capacity ratio for different traffic needs, typically balance the load between GSM and WCDMA traffic. TMO is the software management operation to change software deployment.

SOFTWARE CONFIGURATION (SC) MANAGEMENT

Check Pointing of Configuration

The purpose of check pointing is to save the current configuration data to hard disk. The check point is initiated by the operations staff by command, but it is also possible to schedule periodic check pointing.

The saved configuration data is used to prevent loss of configuration changes after a node reload. For example, the operator may check point after configuration of a new BSC. If a node restart occurs, the node will be loaded with the latest software configuration, which includes the recent BSC configuration change.

Software Configuration Fallback

If a critical fault occurs during node startup, an automatic fallback mechanism restarts the node with a previously valid software configuration. In case automatic fallback fails, it is possible to restore the system manually.

Management of Software

An SGSN-MME can store several software configurations, each check pointed with different configuration data.

It is possible to list the software that has been installed on the SGSN-MME. The list includes information on the software versions and patches that have been loaded on the node. The list can be accessed from the CLI, PXM, or a sub-network management system such as the OSS-RC, over an IIOP (CORBA) interface.

INTERFACE RECONFIGURATION

The procedure to go to SAU-level 2000 in SGSN-MME 2010B MKVI is called Interface Reconfiguration. All interfaces in the SGSN-MME are reconfigured to IP, including SS7 over IP and/or the Gb over IP and/or Iu over IP on the Router PIUs.

Redeployment of the running SC to SAU 2000 is also performed.

BACKUP AND RESTORE

The GSN Server Application (GSA) is an application in the SGSN-MME, which offers functions to backup and restore the SGSN-MME. All backup and restore operations are initiated through a user interface in the GSA. When a backup is taken, all software, all configuration data, and most files, the exception being charging data and log files, are copied from the SGSN-MME to an external storage server, running the GSS.

Restore is a fallback procedure that uses the backup to bring the node configuration back to the same state as when the dump was saved.

SOFTWARE CONFIGURATION MANAGEMENT

An SC is a combination of software (executable applications and scripts) and configuration data. To change an SC, a new SC is created with the desired modifications. Modifications can be made either to the software or the configuration data.

An SC is a file tree containing the software and configuration data needed to run the GSN application. The SC also contains the OSs for the different PIUs, except for MkIV where the GPBOS (Solaris) does not exist within the SC. There can be several SCs on an SGSN-MME, but only one is active at any given point in time.

The SC is the software and configuration stored persistently on disk. Configuration changes that are made after the activation of an SC are not stored persistently until a checkpoint is performed,

SC CREATION AND ACTIVATION

The following ways exist to create an SC:

- Installing an NDP
- Installing a patch onto an existing SC
- Performing a checkpoint
- Redeploying an existing SC

- › An SC is a combination of software (executable applications and scripts) and configuration data.
- › An SC is a file tree containing the software and configuration data needed to run the GSN application. The SC also contains the OSs for the different PIUs, except for MkIV where the GPBOS (Solaris) does not exist within the SC.
- › The SC is stored persistently on disk.
- › The following ways exist to create an SC:
 - Installing an NDP (not covered in the course)
 - Installing a patch onto an existing SC
 - Performing a checkpoint
 - Redeploying an existing SC

Figure 5-7. Software Configuration (SC)

To install and run new software on an SGSN-MME, an SC must first be created by installing an NDP or a patch. Then the SC must be activated to initiate a restart (if needed) and to copy or convert configuration into the new SC.

A checkpoint is used to persistently store configuration data. Unless a manual fallback is desired, a check pointed SC does not require activation, since it does not introduce new or changed software in the SGSN-MME.

Checkpoint

Performing a checkpoint stores the currently running configuration data to disk, creating a new SC. An SC that is created by a checkpoint operation does not need to be activated, unlike the other types of SCs. It is possible, however, to activate a previously checkpointed SC.

Note: Perform checkpoint on consistent configuration data only.

Creation of a Checkpointed SC

When the checkpoint is finished, the new SC is immediately made the active SC.

- › Running SC can be redeployed using the CLI.
- › Redeploying an SC need not reinstall the original NDP – Faster procedure.
- › Redeploying an SC changes the deployment parameters of the SC.
 - Dual Access Reconfiguration, which changes the access mode of the SGSN-MME.
 - SS7 Bearer Reconfiguration, which changes the SS7 standard and bearer used by the SGSN-MME.
 - Interface Reconfiguration to All-IP on MkVI/VI+.
 - SAU capacity expansion.
 - Traffic Mix Optimization Deployment (TMOD).

Figure 5-8. Redeployment of a running SC

The SC in the figure above is of type Checkpointed, and assumes the role Active SC. This type of SC is derived from performing a checkpoint. Checkpointing saves changes to the configuration persistently. Active SC means the most recently activated SC, and thus contains the currently running software and the last checkpoint of the configuration data. In SC listings and in the file system, this is referred to as LastActivated.

Activation of a Checkpointed SC

A checkpointed SC can also be activated. For example, to perform a manual fallback to a previous software level, an SC containing the desired software level can be activated. Care must be taken when activating a checkpointed SC, to avoid unnecessary restarts of the SGSN-MME.

It is possible to undo configuration changes by activating a checkpointed SC created before the configuration was changed. Most configuration changes can, however, be undone in runtime. Configuration changes that are undone in runtime only require a new checkpoint to store the corrected configuration persistently, and not activation of a previously checkpointed SC.

Redeployment

Redeploying an SC changes the deployment parameters of the SC.

The deployment of an SC consists of a number of parameters, which define the SGSN-MME on which the SC runs.

The deployment parameters are:

- Node type
- Hardware configuration
- SAU capacity level
- Signaling System No. 7 (SS7) standard and bearer
- Traffic Mix Optimization Deployment (TMOD)

Deployment is added when an NDP is to be installed (creating an SC). Deployment can also be modified in an existing SC, when the need arises to change any of the parameters of the SGSN-MME. The deployment parameters are modified by means of CLI commands.

The following are examples of redeployments:

- Dual Access Reconfiguration, which changes the access mode of the SGSN-MME.

SS7 Bearer Reconfiguration, which changes the SS7 standard and bearer used by the SGSN-MME.

SOFTWARE CHECKPOINTING

A software checkpoint snapshots the current configuration and saves it for permanent storage. The snapshot created can be used at any reload as a reference at which level of configuration the node should be started. Note also: If configuration changes are made without a software checkpointing after next reload the changes made are gone. So after any bigger change in configuration one should take care to create a new software configuration.

A software configuration consists of parameters but also different parts of software. Though the major software parts are kept in the “release” a software configuration takes about 100-150 Mbytes on hard disk space. An administrator should check the node’s software configurations from time to time and delete old, no longer used versions to free space on the hard drive.

The software configuration names are limited in terms of characters used. Alphanumeric characters are the only allowed, no special characters like “,-_” etc can be used.

SC can be assigned any of the following roles.

- Active SC

This is the most recently activated SC, and thus contains the currently running software and the last checkpoint of the configuration data. In SC listings and in the file system, this is referred to as **LastActivated**.

- Next SC

This is the SC designated to be the next one that the GSN starts on. This is normally handled automatically by the SGSN-MME, but can also be manually set. In SC listings and in the file system, this is referred to as **Next**.

- Default SC

This role is manually assigned. It is the first SC that the GSN will attempt to start on if it fails to start on the designated next SC. Therefore, only an SC that can be considered stable and trusted should be set to default SC. In SC listings and in the file system, this is referred to as **Permanent**.

- Last Booted SC

This is the SC that was used the last time the GSN successfully booted after restarting with any of the activation methods **RebootNode** or **RestartDPE**. In SC listings and in the file system, this is referred to as **LastBooted**.

If the SGSN-MME fails to start on the designated SC, the SGSN-MME attempts to start on other SCs in a predefined sequence. This mechanism is intended to avoid an SGSN-MME that reboots cyclically due to inconsistent software or configuration data.

Automatic Fallback Sequence complies with the following. If the SGSN-MME fails to start on the designated SC (the SC set as **Next**), the SGSN-MME first attempts to start on the **Default** SC. If this also fails, the SGSN-MME attempts to start on the **LastBooted** SC. Should this also fail, the SGSN-MME starts in maintenance mode.

- › When the SGSN-MME “checkpoints” its current configuration into the new SC, a new Software Configuration (SC) is created. The new SC will get the “active” software configuration.
- › The “next” SC is the SC taken at next reload restoring all configuration settings saved at checkpoint.
- › The “default” SC is loaded when the start of “next” SC failed.
- › If the load of “default” SC also fails, the SGSN-MME attempts to start on the “LastBooted” SC. Should this also fail, the SGSN-MME starts in maintenance mode.
 - This is to prevent cyclic reloads on the network element

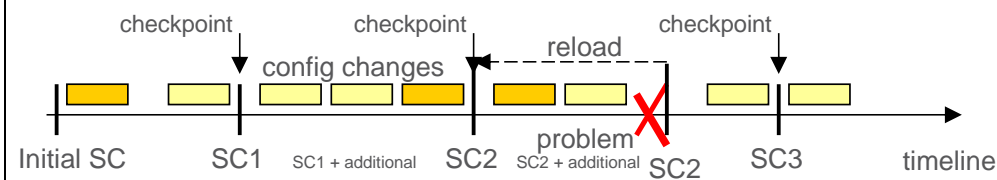


Figure 5-9. Software Configuration Management

SOFTWARE MANAGEMENT LOCAL TOOLS

This chapter describes the menus and elements within the Software Configuration Management Packet Exchange Manager (PXM) form and Software Management using CLI.

The PXM Form

- The Software Configuration Management Form in PXM can be used to manage Software configurations
- The Form is found below “Node Operation” -> “Software”
- The Form contains information about:
 - The last booted SC
 - The last activated SC
 - The SC used for next reload
 - The SC used as fall-back if next reload SC fails
 - All available Releases and Software Configurations on the node

Figure 5-10. Software Configuration Management PXM form (1/3)

- Actions which can be performed using the Form:
 - Create new SCs - software checkpointing
 - Change the SC used for next reload
 - Activate an other SC - reloading the node (!)
 - Delete a SC from the node
 - Define a periodic checkpoint (automatic checkpoint)
 - Change the SC used as fall-back SC - default SC

Figure 5-11. Software Configuration Management PXM form (2/3)

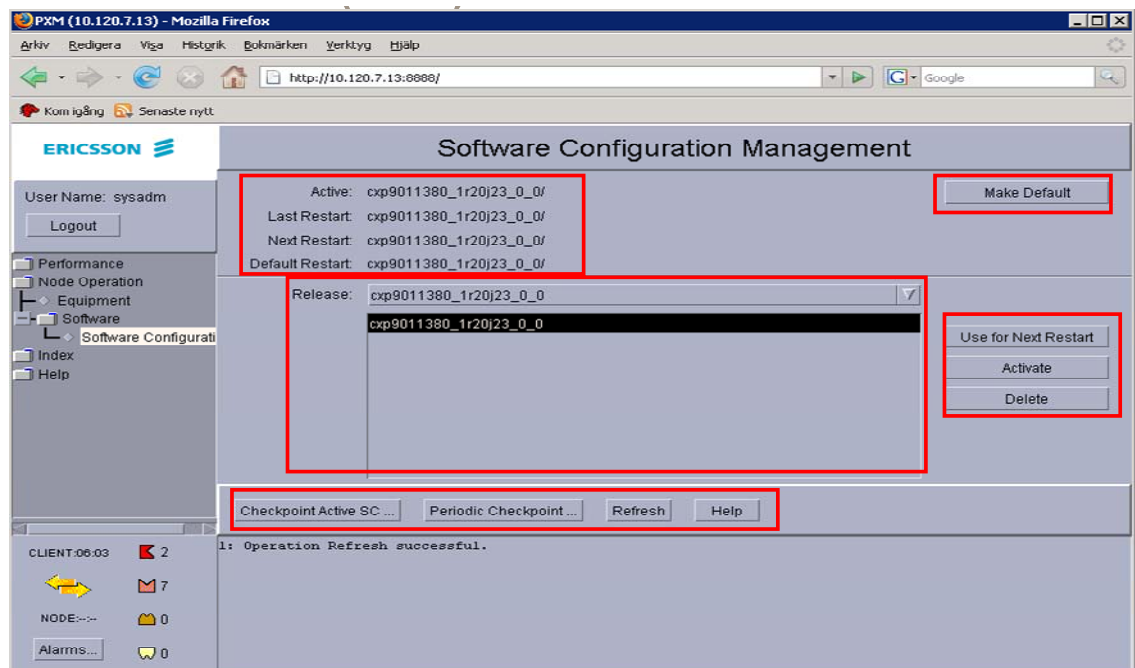


Figure 5-12. Software Configuration Management PXM form (3/3)

Checkpointing an Active SC

- 1 Perform a Checkpoint
- 2 Click Checkpoint Active SC.
- 3 Enter a Logical Name
- 4 In the Logical Name dialog box that appears, enter a logical name for the new checkpoint of the SC, and click OK.
- 5 An Information dialog box appears, informing that the task is queued for execution in the background. Click OK to accept the information, and the logical name is added to the list of checkpointed SCs below the name of the active release.

Checkpointing using the PXM form

- A checkpoint can be done either using the PXM GUI or by CLI commands
- The PXM form is found in “Node Operation” -> “Software” -> “Software Configuration”
- The Checkpoint function is implemented by the button “Checkpoint active SC...”
- A dialog Window will popup asking for a label, which uniquely identifies this SC - No special characters are allowed.
- The SC is created in background - a message window appears when finished

Figure 5-13. Software Checkpointing

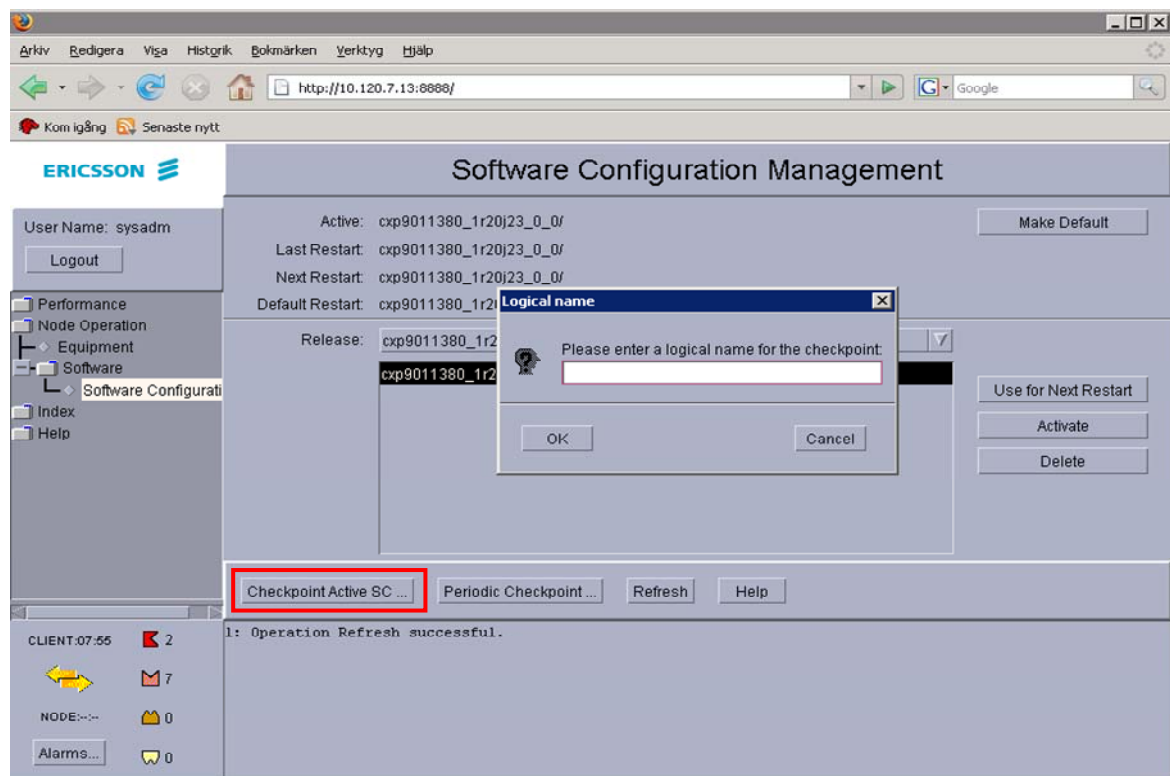


Figure 5-14. Software Checkpointing with PXM

Checkpointing using CLI

The command line interface can be used to achieve the same task. The CLI command to create a new software configuration is named “checkpoint”. Checkpoint takes two options, the release name and the name of the new software configuration.

The CLI command will respond with a printout containing the release name and software configuration name to acknowledge.

- Use the CLI command “checkpoint” to create a new SC
- Syntax:
`checkpoint {[-rel ReleaseName] -cpn CheckpointName}`
- *ReleaseName* must match the current software release found either in PXM Software Configuration or using the following CLI command: `get_active_sc`.
- *CheckpointName* must not contain any special characters and uniquely identifies the new SC

Figure 5-15. Software Checkpointing with CLI

The checkpoint CLI command checkpoints the currently active Software Configuration (SC). The resulting SC is named `ReleaseName_CheckpointName`.

Activating an SC

Activating an SC means to reload the network element with the selected software configuration.

Select Release and SC

Select the release from the Release list and the original or the checkpoint of the SC from the list below the release. This implies the persistently storage of all currently open CDRs and a node downtime for traffic for 5-10 minutes.

Activating an SC using PXM

From the list below the active release, select the SC to be used for activation.

Click Activate to activate the SC. In the Confirmation dialog box that appears, click OK to confirm the activation of the SC. The

Active text at the top of the Software Configuration Management PXM form is updated with the name of the specified SC.

Activating an SC using CLI

The CLI commands for managing the SC activation parameter are *get_active_sc* and *set_active_sc*. The *get_active_sc* command takes no options and prints out the current setting. The *set_active_sc* is used to reload the node using the selected SC. The command takes two options, the release name and the software configuration name to make active.

It's also possible to query which SC was booted at last reload using CLI commands. The command "*get_booted_sc*" is used to query this printout. The result will contain release and software configuration name.

- The following CLI commands can be used:
 - *get_active_sc* prints out the currently active SC
 - *get_booted_sc* prints out the last booted SC
 - *listSCs (toolbox)* retrieves the software status information of the SGSN.
- Example:

```
=== sysadm@eqm01s14p2 ANCB ~ # gsh get_active_sc
Release Name : eabln15652_ndpgsn_1_0
CheckpointName: eabln15652_ndpgsn_1_0_WithSS7andIPsec
=== sysadm@eqm01s14p2 ANCB ~ # gsh get_booted_sc
Release Name : eabln15652_ndpgsn_1_0
CheckpointName: eabln15652_ndpgsn_1_0_WithSS7
```

Figure 5-16. Reading active/booted SC and other related information (CLI)

- The following CLI commands can be used:
 - **rm_sc** **removes a SC or Release**
 - **set_active_sc** **activates an SC - including a restart of DPE applications or a reload (!)**
 - **set_next_sc** **defines the SC loaded at next reload**
- Syntax:
 - rm_sc {-rel ReleaseName [-cpn ScName] }**
 - set_active_sc {-rel ReleaseName [-cpn ScName]}**
 - set_next_sc {-rel ReleaseName [-cpn ScName]}**
 - ReleaseName is the release name the SC belongs to
 - ScName is the software configuration to delete, activate or use for next reload

Figure 5-17. Activating/Deleting/Using a SC (CLI)

Deleting a Checkpointed SC

Deleting an SC is important to gain free disk space again. To avoid confusion and to keep the disk usage low there should be only a few software configurations on the node. The old and no longer used software configurations should be deleted. The procedure will remove the software configuration from the network element. An active or permanent software configuration can't be deleted directly. One has first to activate another SC and/or to choose another default SC to be able to delete the old versions.

For security reasons also software configurations defined for next restart are saved from deletion. One has to select another SC for next restart to remove the original.

Deleting an SC using PXM

From the list below the Release list, select the original or the checkpoint of the SC.

Click Delete to delete the SC. In the Confirmation dialog box that appears, click OK to confirm the deletion of the SC.

An Information dialog box appears, informing that the task is queued for execution in the background. Click OK to accept the information, and the selected SC is deleted from the list of active releases.

Deleting an SC using CLI

The CLI command `rm_sc` is used to remove a software configuration from the network element. The command takes two options, the release name and the software configuration name to remove.

Creating the Default Restart SC

A default software configuration is used on a special case. When the system reloads and the selected software configuration to start is not starting properly a second reload will occur. If the same software configuration is loaded again the node will end up in cyclic reloads without getting operational again. To avoid this situation the default software configuration is loaded if the first reload attempt fails. One should take care to always select a proper software configuration to be default SC. Therefore, only an SC that can be considered stable and trusted should be set to default SC.

The system helps in defining a proper SC automatically by limiting the way in which default SCs may be defined. Only the last restarted software configuration, which has proved to be working because it is currently running, can be set as default SC.

Creating a default SC using PXM

Click Make Default to specify the last restarted SC to be the default restart SC.

In the Confirmation dialog box that appears, click OK to confirm the default restart SC. The Default Restart text and Next Restart text at the top of the Software Configuration Management PXM form is updated with the name of the specified SC.

- Defining a default SC is special
- The currently active SC proves to be operational
- By selecting the “make default” button the active SC label is “copied” into the default
- There is no undo function
 - One has to reload the “old” SC and select “Make Default” again to undo(!)

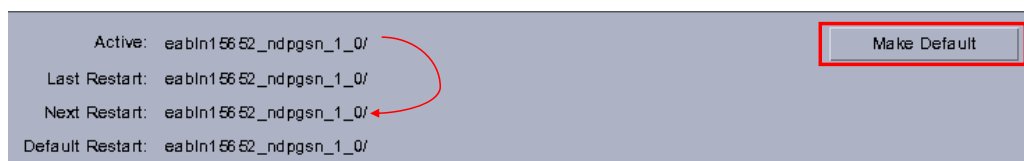


Figure 5-18. Defining a default SC using PXM

Creating a default SC using CLI

The CLI command “*set_default_sc*” can be used to define a new default SC. The command takes no options. The last restarted software configuration will get the default. The “*get_default_sc*” command can be used to query which SC is used as default SC.

The following CLI commands can be used:

- **get_default_sc** prints out the fall-back SC
 - Example:

```
=== sysadm@eqm01s14p2 ANCB ~ # gsh get_default_sc
Release Name : eabln15652_ndpgsn_1_0
CheckpointName: eabln15652_ndpgsn_1_0_WithSS7andIPsec
=== sysadm@eqm01s14p2 ANCB ~ #
```
- **set_default_sc** defines the currently active SC to be the fall-back
 - Syntax:
`set_default_sc`
 - The command takes no options - the currently active SC will be used as default SC

Figure 5-19. Defining a default SC using CLI

Getting Node Info

Getting Node Info using CLI

- `get_ne`

The following CLI commands can be used:

- › `get_ne`
 - displays information about the SGSN-MME, such as node type, SAU level, hardware configuration, and SS7 stack standard.
- › `get_node_info -info InfoAttribute [{ -rel ReleaseName [-cpn CheckpointName] }]`
 - returns the requested node information such as hardware information, node type, network type, SS7 stack standard, software level, software level, NDP build, CP level, TMO or node size (SAU). The current software configuration is used if no software configuration is specified.

Example:

```
=== sysadm@eqm01s14p2 ANCB Core/log # gsh get_node_info -info Type
sgsnwg
=== sysadm@eqm01s14p2 ANCB Core/log # gsh get_node_info -info Hw
mkiv
```

Figure 5-20: Getting Node Information

The `get_ne` CLI command displays information about the SGSN-MME, such as node type, SAU level, hardware configuration, and SS7 stack standard.

- `get_node_info`

The CLI command “`get_node_info`” returns the requested node information such as hardware information, node type, network type, SS7 stack standard, software level, software level, Node Delivery Package (NDP) build, Correction Package (CP) level, Traffic Mix Optimization (TMO) or node size (SAU). The current software configuration is used if no software configuration is specified.

The `InfoAttribute` variable specifies the type of node information requested. It contains node attributes such as the node and network type, hardware family, SS7 stack standard, software revision, or node size. Depending on the need the variable “`InfoAttribute`” can be one of the following: `Type`, `HW`, `SAULevel`, `TMOD`, `SS7 SWLevel`, `NDPbuild`, `CPName`.

SOFTWARE CONFIGURATIONS BEHIND THE SCENES

- Software configurations kept in the directory
/Core/DPE_root/SoftwareConfigurations
- Each new SC creates a new directory
- The directory is named like the SC:
 - *ReleaseName_CheckpointName*
 - E.g. ECF_cxp9010273_r2j01_0_11_initialPlusPM
ReleaseName = ECF_cxp9010273_r2j01_0_11
CheckpointName = initialPlusPM
- The directory home four additional ASCII files:
 - LastActivatedSoftwareConfiguration
 - LastBootedSoftwareConfiguration
 - NextSoftwareConfiguration
 - PermanentSoftwareConfiguration
- Do not modify the ASCII files - always use CLI or PXM commands

Figure 5-21. SCs behind the scenes

Intentionally Blank

6 Support Systems

OBJECTIVES

- › Administer system software management with the GSS/GSA functionality including listing installed software, verifying and
- › Performing backup and restore as well as system checkpoint procedures.

Figure 6-1. Objectives

GSN SUPPORT SYSTEM

INTRODUCTION

In today's complex computing environment, keeping a working backup of the system is synonymous to reliability and robustness. The same will apply to the SGSN-MME, which handles a far more complicated content in a real time processing basis.

The storage of a system backup outside the node is necessary. In the event of major faults like software or hardware failures, fire, faulty updates/upgrades and so on, the existence of a good backup system will lessen the impact of the failure and facilitate the recovery process.

The backup and restore procedures are executed in the GSN Support Application (GSA). The GSA runs on the node and backups are transferred and stored on the GSN Support System (GSS) server. Consequently, the number of backups that can be stored depends on their size and the storage capacity of the GSS server. The GSS server runs on a UNIX-based system such as a Solaris Operating System (OS) or a Linux OS.

The GSA consists of scripts for creating, administering and restoring backups. When a backup is taken, all software, all configurations, and all files, except charging data and log files, are backed up.

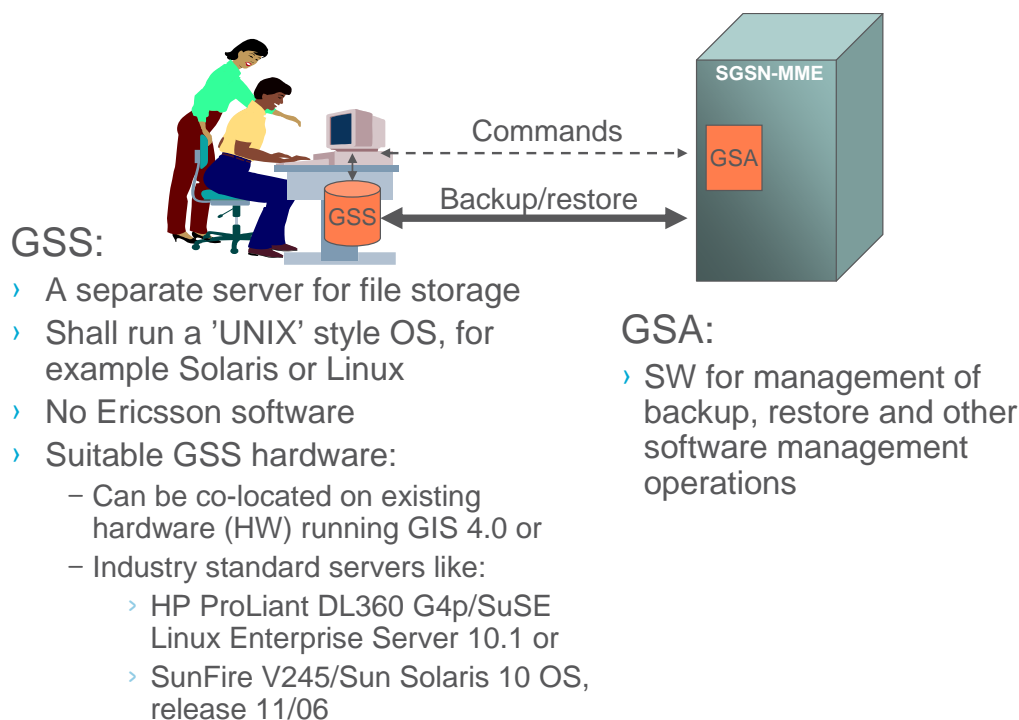


Figure 6-2. GSS/GSA Backup/Restore Solution

GSS/GSA SYSTEM DESCRIPTION

The Ericsson GSS requires a properly configured server.

The GSS server supports the SGSN-MME 2010B in the following operations:

- Initial installation of SGSN-MME software
- Backup and restore
- Software upgrade

The GSS server stores backups taken with the GSA.

The GSS server is a UNIX server, configured to communicate with the SGSN-MME. No Ericsson software needs to be installed on the GSS server. The GSS server also needs to be configured differently depending on the SGSN-MME node. MkIV runs the Solaris Operating System (OS) and is connected to the GSS server through the Power and Ethernet Board (PEB) PIU, while the MkV/VI/VI+ run the Linux OS and are connected to the File Server Board (FSB) PIU through a network.

- › GSS Server is a server for file storage, which is owned and managed by the customer, but setup according to the GSS service requirements from Ericsson.
- › GSS Server runs with a Unix-like OS.
- › GSS Server supports DHCP, FTP, TFTP(only MKIV) and Time services.
- › GSS Server supports the following operations:
 - Initial installation of SGSN-MME software.
 - File storage for backup and restore.
 - Software upgrade.

Figure 6-3. GSN Support System (GSS)

The GSA tool runs on the active Node Controller Board (NCB) on MkIV and on the standalone File Server Board (FSB) on MkV/VI/VI+.

The GSA contains functionality for backup, restore and other software management operations in SGSN-MME 2010B. The GSA user interface contains a series of commands described in this chapter.

- › GSA is SW with backup, restore and other software management functions in SGSN-MME.
- › It resides on the Active NCB (MkIV) or the FSB (MkV/MkVI/MkVI+).
- › The GSA user interface contains a series of commands.
- › There are two main ways to access the GSA user interface:
 - By using the console port of the PIU where the GSA is running
 - By using the O&M network
- › The GSA commands are quite similar to CLI commands but are not classic CLI commands under the gsh shell.

Figure 6-4. GSN Support Application (GSA) in 2010B

Due to different hardware and software configurations of the SGSN-MMEs the instructions for back up and restore may differ between the different hardware versions. See the figure below as overview of the hardware and connections.

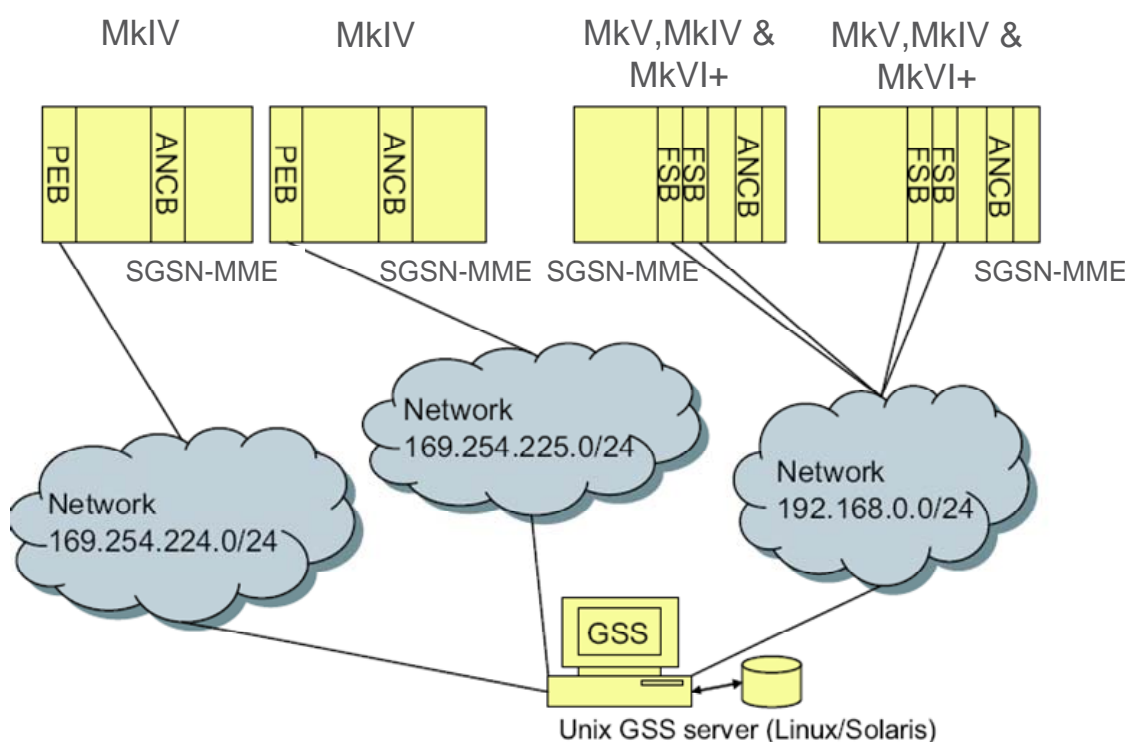


Figure 6-5. Network Topology of GSN Support System (GSS)

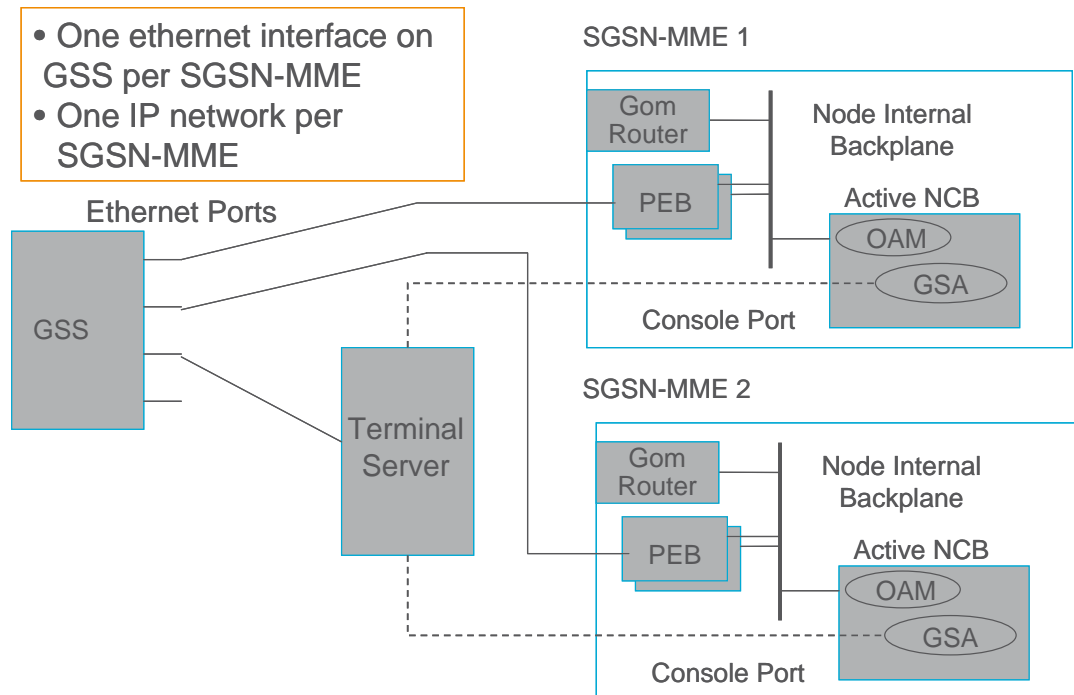


Figure 6-6. GSS - Multiple SGSN-MME MkIV connection

When SGSN-MME is equipped with MkIV hardware, the connections between GSS and SGSN-MME comply with the figure below.

- Backup data traffic:

The GSS is connected to the all GPBs and NCB via PEB. This connection is used for backup data. The connection is the same as that of earlier SGSNs.

- GSA interface access:

Cable is connected to the console port on the NCB.

An Administrator can login to the GSA interface from a terminal server via the TCP/IP network and then GSA commands (Backup/restore etc) can be issued.

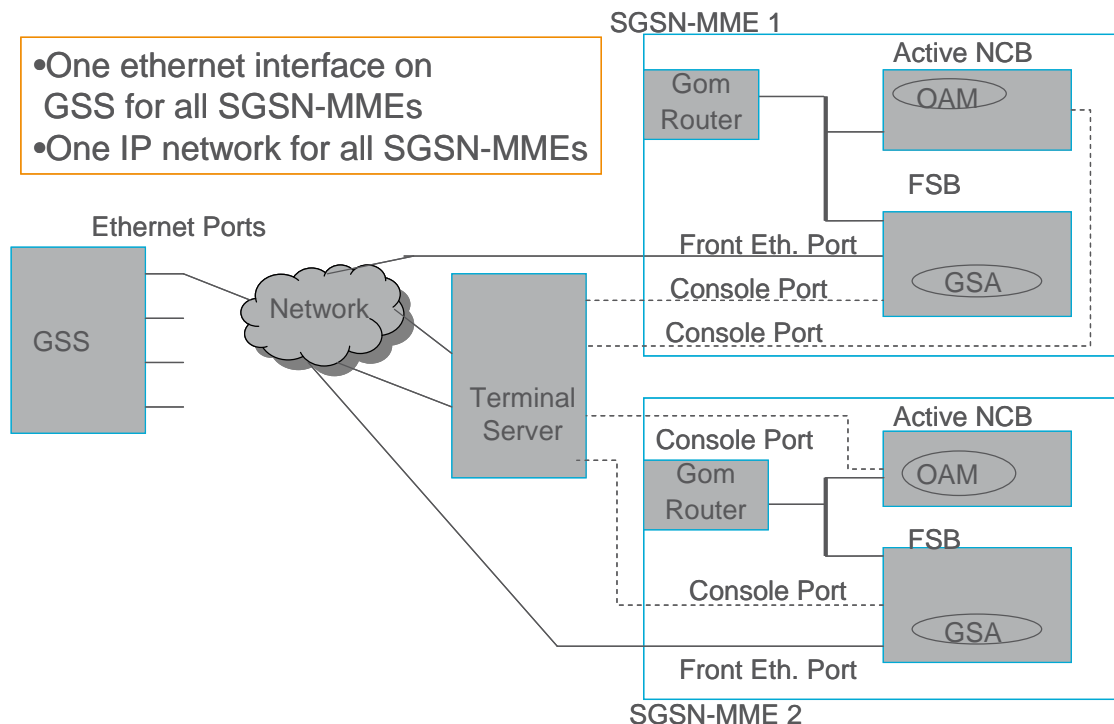


Figure 6-7: GSS-Multiple SGSN-MME MkV/MkVI & MkVI+ connections

When SGSN-MME is equipped with MKV, MKVI OR MkVI+ hardware, the connections between GSS and SGSN-MME comply with the figure above.

- Backup data traffic:

The GSS is connected to the FSB via the front Ethernet port. This connection is used for installation, backup, or restore of the SGSN-MME.

- GSA interface access:

Cable is connected to the console port on the FSB. RS232 port is for permanent connection to a console or terminal server. The connection can be used to reach the GSA user interface to perform backup, restore, or installation.

SGSN-MME BACKPLANE ADDRESSING

Backplane uses **169.254.0.0/17**:

- Addresses are "Link-local" and not routable according to IETF (most routers will not route them)
- Addresses from the 169.254.0.0/16 network can not be used for external links (has never been supported)
- All customers must only use the network above. They must not use any other network or network mask!

Even if the addresses used are public, an RFC states that this particular class is not routable and then it can be considered as private.

The large number of hosts IP addresses available, 32766, is due to the fact that new generation GSS can handle up to 20 nodes.

› Backplane uses **169.254.0.0/17**:

- Addresses are "Link-local" and not routable according to IETF (most routers will not route them)
- Addresses from the 169.254.0.0/16 network can not be used for external links (has never been supported)
- All customers must only use the network above. They must not use any other network or network mask!
- The addressing principles for the bits from mag/slot/pos are:
 - › Main board on p=2
 - › Slot hops for IP address is 16

A node internal Backplane IP Address consist of:

1. 16-20 bit Network Identity (i.e. M_5 - M_2 may also be part of NW when number of magazine ≤ 3)
2. 2-6 bit Magazine Id
3. 6 bit Slot Id
4. 4 bit Position Id

$N_{15}N_{14}N_{13}N_{12}N_{11}N_{10}N_9N_8 \mid N_7N_6N_5N_4N_3N_2N_1N_0 \mid M_5M_4M_3M_2M_1M_0 \mid S_5S_4 \mid S_3S_2S_1S_0 \mid P_3P_2P_1P_0$

Figure 6-8: Backplane Addressing

GSS ADDRESSING FOR SGSN-MMES MKIV

The whole **169.254.224.0/19** network is reserved for GSS.

- Each node is assigned a GSS network - **169.254.X.0/24**, where X depends on the node (224-255).
- Addresses are "Link-local" and cannot be routed
- IP addresses corresponding to slot positions 22-31 are reserved for OS installation
- The addressing principles are different from the backplane:
- No position concept (no P in IP address) - Slot hops for IP address is 1

Only GPBs has an interface to this "second backplane" towards GSS, although the algorithm reserves an address for each slot within the node.

- The whole **169.254.224.0/19** network is reserved for GSS.
 - Each node is assigned a GSS network - **169.254.X.0/24**, where X depends on the node (224-255).
 - Addresses are "Link-local" and cannot be routed.
 - IP addresses corresponding to slot positions 22-31 are reserved for OS installation.
 - The addressing principles are different from the backplane:
 - No position concept (no P in IP address) - Slot hops for IP address is 1.
 - Only GPB (MKIV) has an interface to this "second backplane" towards GSS, although the algorithm reserves an address for each slot within the node.

A GSS/Node IP Address consist of:

1. 24 bit Network Identity
2. 3 bit Magazine Id
3. 5 bit Slot Id
4. No Position Id

$N_{23}N_{22}N_{21}N_{20}N_{19}N_{18}N_{17}N_{16} | N_{15}N_{14}N_{13}N_{12}N_{11}N_{10}N_9N_8 | N_7N_6N_5N_4N_3N_2N_1N_0 | M_2M_1M_0S_4S_3S_2S_1S_0$

Figure 6-9. GSS Addressing for SGSN-MMEs MkIV

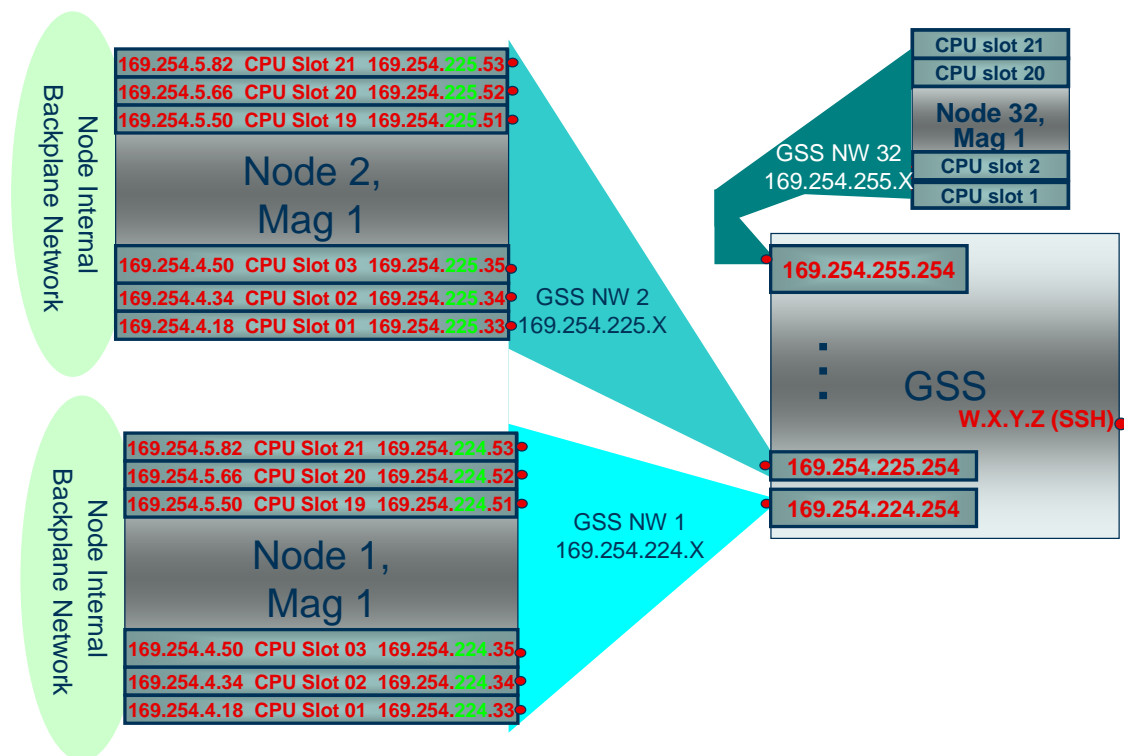


Figure 6-10. GSS and Back-Plane Network for MkIV

GSS SYSTEM REQUIREMENTS

The GSS is a server for file storage, which is selected, owned and managed by the customer, but setup according to the GSS service requirements from Ericsson. No Ericsson software will reside on the GSS server.

Unlike the GIS 3.0 or GIS 4.0 server in SGSN R7 and earlier, it is not required that the GSS server is run on specified hardware. This gives more freedom in the choice of hardware and also the possibility to collocate the GSS function in hardware used for other applications. Some basic requirements exist though:

The GSS shall run a 'UNIX' style OS, for example Solaris or Linux, and have built-in support for the following standard services:

- Network File System (NFS)
- DHCP
- FTP
- Trivial File Transfer Protocol (TFTP) (MkIV only)
- Time Service

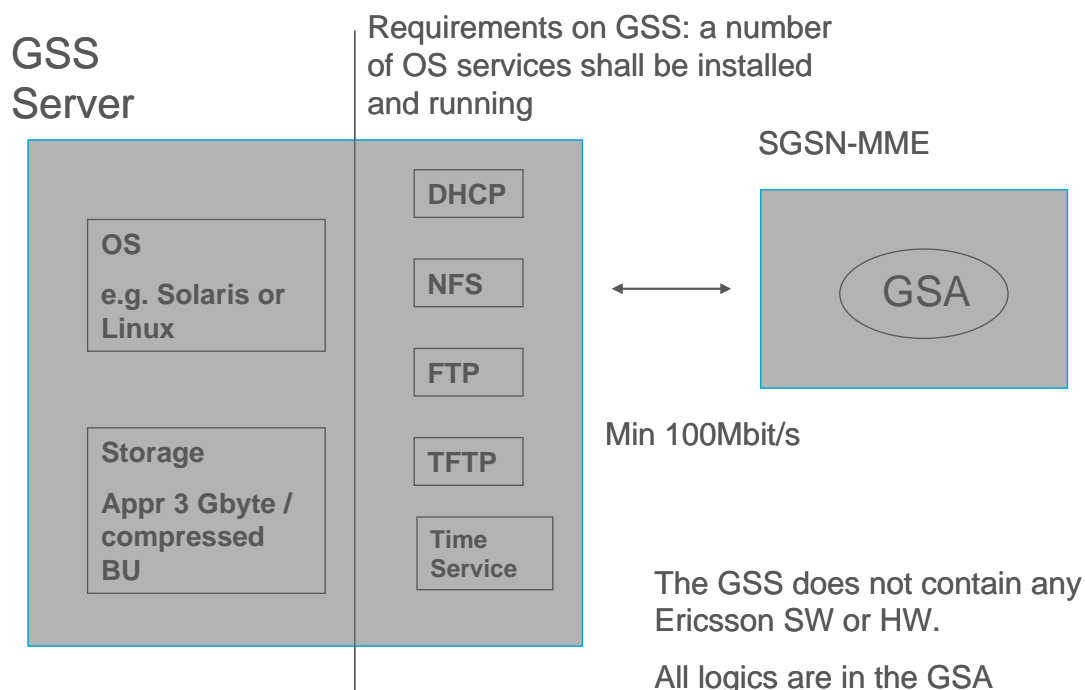


Figure 6-11: GSS Requirements

Hardware Requirements

Storage Requirements

The GSS server stores backup dumps and SGSN-MME software used for installation of the SGSN-MME. Choose hard disk size depending on the OS requirements and the number of backups to be stored on the GSS server.

The examples below give an idea of the required disk space:

- Approximately 5 GB disk space is needed to store the SGSN-MME software.
- Approximately 5 GB disk space is needed for the UNIX based OS.
- Approximately 1-4 GB is needed for each SGSN-MME MkIV backup. An SGSN-MME MkV/VI/VI+ needs approximately 20 GB for the backup.

Bandwidth requirement

The bandwidth between the GSS server and the SGSN-MME should be at least 100 Mbps.

Verified hardware for GSS based on Solaris

SunFire V245

- Sun Solaris 10 Operating System (OS), release 11/06 or later
- 1.5 GHz UltraSPARC IIIi Processor
- DVD-ROM Drive
- Two 73 Gb internal SCSI disks or larger. The disks must be equally sized for mirrored setup.
- 1024 Mb RAM
- One PCI, PCI-X or PCI Express 4-port Gigabit Ethernet card, also called quad card

Figure 6-12. Verified HW for GSS based on Solaris

Verified hardware for GSS based on Linux

HP ProLiant DL360 G4p

- SuSE Linux Enterprise Server 10.1
- 3.8 GHz Xeon Processor
- DVD-ROM Drive
- Two 73 Gb internal SCSI disks or larger. The disks must be equally sized for mirrored setup.
- 2048 Mb RAM
- One PCI, PCI-X or PCI Express 4-port Gigabit Ethernet card, also called quad card

Figure 6-13. Verified HW for GSS based on Linux

To store GSN backups on tape, an external tape drive can be connected to the GSS server.

GSS FILE AND BACKUP STRUCTURE

In GSS server a partition is created on the hard disk called */gsn* to store the backup dumps and the SGSN-MME software deliverables.

Each SGSN-MME connected to the GSS server has a unique root path where backups are stored. It is important that the owner of the root path directories is *coreUser* to give the FTP client on the GSA access to the directories.

A root path for each SGSN-MME is created to connect to the GSS server. The directories will store backup files for the corresponding SGSN-MME. Use the following naming conventions:

/gsn/nodes/<nodename>

The directory */gsn/sw* is created for the software deliverables.

The network booting of the MkIV SGSN-MME requires a directory */tftpboot* where the network booting files are stored.

Improved file structure:

- › One sub directory for each SGSN-MME controlled by the GSS
- › One sub-sub directory for each distribution initial installation or backup taken
- › Naming of backups done by GSA on the SGSN-MME. Standardized behaviour.

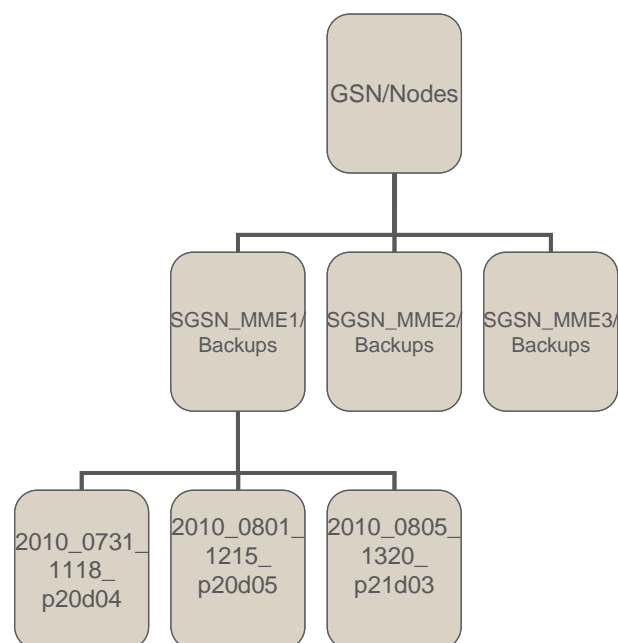


Figure 6-14. GSS File and Backup Structure

The benefits of GSS/GSA are the following:

Less trouble with different versions of scripts for Backup server/node, as all logics now on SGSN-MME (GSA).

Separate HW for DHCP Server and Storage is possible.

Some restrictions regarding co-location: No DNS (server or client) allowed on GSS servers for MkIV.

- › Freedom of HW selection, reuse/co-location with other customer applications.
- › Improved file/backup structure.
 - Timestamps, versions etc.
- › Support for handling multiple SGSN-MMEs in parallel.
 - Backups of several nodes can be done at the same time.
 - Restore several nodes at a time.
 - Automatic backup functionality.
- › Support for running GSS on a central site (MkV/MkVI/MkVI+)
 - Requires bandwidth between central site and node.
- › All Ericsson SW for BU/restore in one package (the NDP). This improves present and future quality.
- › All operations are managed from the SGSN-MME.

Figure 6-15. Benefits of GSS/GSA

It is possible to use the same HW to obtain GSS and GIS 3.0 (Solaris 8) or GIS 4.0 (Solaris 10) server functionality.

After upgrading to SGSN-MME 2010B, a back up of the SGSN-MME 2010B can be initiated on the GSA interface. A restore of SGSN-MME 2010B requires that the DHCP definitions must be changed to point to the 2010B NIB package first.

GSA COMMAND DESCRIPTION

The GSA commands are quite similar to Command Line Interface (CLI) commands but are not classic CLI commands under the gsh shell.

There are two main ways to access the GSA user interface:

- By using the console port of the PIU where the GSA is running
- By using the O&M network

- backup [-t *tag-name*]

The backup command creates and saves a compressed backup. The compressed backup is stored on the GSN Support System (GSS) server on the format: YYYY_MMDD_hhmm_<tag-name>.

YYYY_MMDD_hhmm indicates the start date and time of the backup.

-t *tag-name*

The variable tag specifies an optional name tag to label a backup. This is a convenient way to remember the reason for the backup.

Type: string

- restore *backup-name*

The restore command gets and installs a compressed backup. The node is restored to a previously stored backup.

- restore_fsb

The restore_fsb command restores an fsb from the other fsb in MkV and MkVI only.

The command removes all the software on the current FSB, stops the FSB synchronization services and reboots. If the FSB is primary the secondary FSB becomes the primary FSB. When the FSB is online again becomes the secondary FSB and at the same time restores the software running on the primary FSB.

This command is available in MkV/VI/VI+. It can be accessed by the console only.

- delete *backup-name*

The delete command deletes a node backup from the GSS server.

Example:

Input

```
delete 2006_0925_1212
```

In this example, the 2006_0925_1212 backup is deleted.

- remove

The remove command removes all stored backups on the GSS server.

- verify *backup-name*

The verify command verifies that a node backup on the GSS server fulfills a number of criteria.

- periodic_backup

periodic_backup *periodic-backup-start-time* [-r *hours*] [-t *tag-name*] [-l]

periodic_backup [-l]

periodic_backup [-k *job-id*]

The periodic_backup command creates lists and deletes periodic backup jobs. The time of day of the backup and the recurrence are input parameters to the command.

The periodic_backup command performs the same way as the 'manual' backup command.

Use the periodic_backup command in combination with the -l GSA operand and variable to list the current defined periodic backup jobs and job-id.

- list

The list command lists all node backups stored on the GSS server.

Example:

Input

```
list
```

Output

```
2007_1012_1110_UTC__initial_install
```

- version

The version command displays the current version of the GSA.

Example:

Input

```
version
```

Output

```
-----  
GSA version:
```

```
Build date           : Fri Oct 26 19:45:28 MEST 2007
```

```
Built by             : xxxxxx
```

```
fsbos_version:
```

```
1./main/ndpgsn_4_0/ndpgsn_4_0_nir20j/ndpgsn_4_0_eabln3  
7073
```

```
Welcome to MontaVista(R) Linux(R) Professional Edition  
4.0.1 (0502020)
```

```
Build label: NDPGSN_4_0_FSB_OS_LINUX_1R20J15
```

```
Build host: selnx036
```

```
Build date: Thu Nov 1 10:27:19 MET 2007
```

```
Disk reformatted: Mon Nov 12 17:19:46 UTC 2007  
-----
```

- help [*command-name*]

The help command shows help on available GSA commands.

- info

The info command displays information and network properties about the GSS and the local board running GSA.

Example:

Input

```
info
```

Output

```
time: 2007_1012_0832 UTC

disc synchronization:

0: cs:Connected st:Primary/Secondary ld:Consistent
1: cs:Connected st:Primary/Secondary ld:Consistent
2: cs:Connected st:Primary/Secondary ld:Consistent

primary_FSB:

front ETH0 : mac 00:01:EC:ED:E8:C2

: ip 10.45.0.15

: bcast 10.45.0.255

: mask 255.255.255.0

GSS server:

ip-address : 10.45.0.251

node path : /gsn/nodes/node186

backup dir : /gsn/nodes/node186/backups
```

- › **To enter GSA User Interface**
 1. Connect a cable to the FSB (MKV,MKVI or VI+) console port. Access to gsa prompt by enter "gsa"
 2. Connect to the NCB (MKIV) via O&M network. Access to gsa prompt by enter "gsa"
- › **BACKUP and RESTORE**
 - gsa> **backup** [-t TagName]
 - gsa> **restore** backup-name
 - gsa> **restore_fsb**
 - gsa> **delete** backup-name
 - gsa> **remove**
 - gsa> **verify** backup-name
 - gsa> **periodic_backup** YYYY_MMDD_hhmm [-r hours] [-t TagName]
- › **UTILITIES**
 - gsa> **list**
 - gsa> **version**
 - gsa> **help** [CommandName]
 - gsa> **info**

Figure 6-16: GSA Commands for backup and restore

BACKUP AND RESTORE PROCESS DESCRIPTION

The GSA offers functions to backup and restore the SGSN-MME. GSA consists of an application for creating, administering, and restoring backups. When a backup is taken, all software, all configuration data, and most files (the exception being charging data and log files) are copied from the SGSN-MME to an external storage server acting as the GSS server. This data can then be restored to the SGSN-MME in case a fallback is needed and the other fallback mechanisms of the SGSN-MME do not work.

Backups should be kept as clean as possible, for example, by deleting expendable checkpoints and Software Configurations (SCs), before the backup is taken.

BACKUP

To backup the SGSN-MME, the GSA mounts the file system of the GSS using NFS. For MkV/VI/VI+ hardware, the necessary parts of the FSB file system are archived to a backup directory on the disk of the GSS. For MkIV hardware, one of the NCB is backed up. This information is then used to restore both NCB and GPB PIUs.

The backup procedure includes preparing the SGSN-MME for backup, performing backup, and verifying backup. A backup should be performed after installation, after major configuration changes and after reconfiguration of the SGSN-MME.

MKV/VI /VI+

The backup command performs a dump of the root and core partitions, including the SC.

A backup is always created as a full set of file system dumps only on primary FSB boards. And FTP is used for transporting dumps to GSS server.

- › The GSA command '*backup*' is used to take backups of primary FSB's local filesystem.
- › All dumps of the FSB are gzipped to 3 files.
- › The backup files are transferred by ftp and stored on GSS server.

SGSN-MME (MKV/VI/VI+)

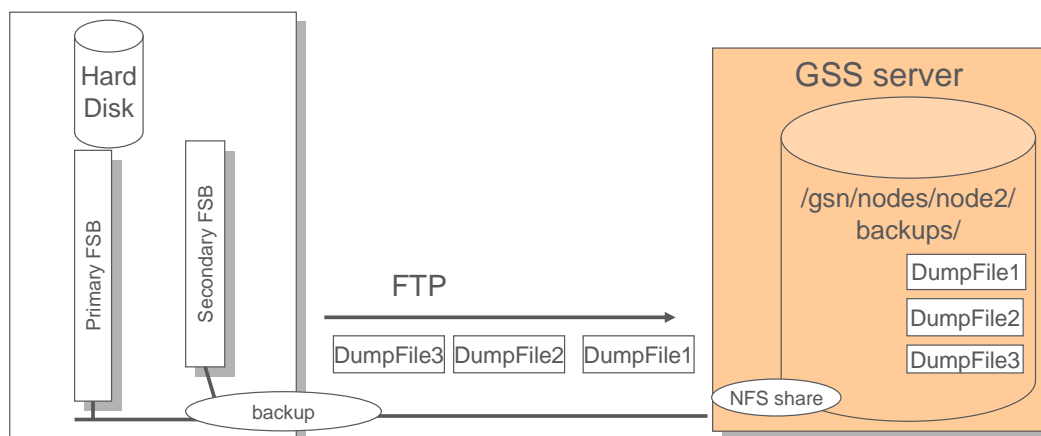


Figure 6-17: Backup Principal – MkV/VI/VI+

Examples

The following examples describe the use of the backup command.

Taking Backup

This example shows how to back up the software on the SGSN-MME with identity remsgsn03.

Input

```
backup -t test
```

Output

```
=====
2007-12-21          01:17:20          Backup->
169.254.226.254:/gsn/nodes/remsgsn03/backups/2007_1221
_0117_UTC__test

Using protocol: ftp

=====

Backup of /export/Core/.current_fsb_software to
169.254.226.254:/gsn/nodes/remsgsn03/backups/2007_1221
_0117_UTC__test/FSB_os.cpio.gz)

[OK] backup done of /export/Core/.current_fsb_software
to
(169.254.226.254:/gsn/nodes/remsgsn03/backups/2007_122
1_0117_UTC__test/FSB_os.cpio.gz)

Backup of /mnt/local/etc to
(169.254.226.254:/gsn/nodes/remsgsn03/backups/2007_122
1_0117_UTC__test/FSB_local.cpio.gz)

[OK] backup done of /mnt/local/etc to
(169.254.226.254:/gsn/nodes/remsgsn03/backups/2007_122
1_0117_UTC__test/FSB_local.cpio.gz)

[OK] backup done of /export/Core to
(169.254.226.254:/gsn/nodes/remsgsn03/backups/2007_122
1_0117_UTC__test/FSB_core.cpio.gz)

2007-12-21 01:35:33 End of Backup

[OK] FSB_os.cpio.gz exist

[OK] FSB_os.cpio.gz size

[OK] FSB_local.cpio.gz exist
```



```
[OK] FSB_local.cpio.gz size
[OK] FSB_core.cpio.gz exist
[OK] FSB_core.cpio.gz size
[OK] The backup
/gsn/nodes/remsgsn03/backups/2007_1221_0117.UTC__test
has all files needed for a restore
```

RESTORE

The restore procedure includes preparing the SGSN-MME for restore, restoring the backup, and verifying restoration. Restore is a fallback procedure that uses the backup to bring the SGSN-MME configuration back to the same state as when the dump was saved.

MkIV

To restore the SGSN-MME, the GSA reconfigures all GPB PIUs through the Dynamic Host Configuration Protocol (DHCP) and NFS. The SGSN-MME is then partitioned, the backup mounted from the GSS over NFS, and the archives are transferred and unpacked. The SGSN-MME returns to operation after a restart.

MkV/VI/VI+

The standalone FSB (FSBv4) transfers the backup archive using File Transfer Protocol (FTP) to the disk space of the standalone FSB, where it is unpacked. The SGSN-MME boots using the software residing on the standalone FSB.

Restore of a standalone FSB

The restore of a standalone FSB differs in how it is initiated from the node restore.

But only a few preparations are necessary in order to enable the GSS for board restoration. First and most important the DHCP table must be updated on the GSS with the Media Access Control (MAC) address for the new FSB, and remove the MAC address for the removed FSB.

Before the faulty FSB is removed there must be an FSB switchover if the faulty FSB was the primary FSB. That is to say, the secondary FSB becomes the primary FSB if the faulty FSB is primary FSB. The inserted FSB restores itself from the Primary FSB.

For detailed instructions on how to replace a standalone FSB board and what to prepare in GSS server use the CPI document “Replacing a standalone FSB 1/1531-ROJ 208 137/1 Uen S” and “GSN Support System (GSS) Configuration, INSTALLATION INSTR. 2/1531-AXB 250 05/8-1”.

- › The GSS must be prepared:
 - DHCP server must be updated with the new MAC for the new FSB then running.
- › The FSB starts up immediately when inserted.
 - If the question “Enter yes within 10 seconds to restore this FSB” appears during startup, enter yes.
 - In case the question does not appear, the FSB is prepared as a sparepart. The restore is done automatically.
- › The FSB now automatically restores itself from the primary FSB.
- › Replacing an FSB normally does not impact the running node.
- › Check CPI document “Replacing a standalone FSB 1/1531-ROJ 208 137/1 Uen S” and document “GSN Support System (GSS) Configuration, INSTALLATION INSTR. 2/1531-AXB 250 05/8-1 ”

Figure 6-18: Restore a standalone FSB in MkV/MkVI/VI+

Examples

Restoring a Backup

This example restores the backup 2010_1221_0117.UTC__test.

Input

```
restore 2010_1221_0117.UTC__test
```

Output

```
[OK] BACKUP_COMPLETE exist

[OK] FSB_os.cpio.gz exist

[OK] FSB_os.cpio.gz size

[OK] FSB_local.cpio.gz exist

[OK] FSB_local.cpio.gz size

[OK] FSB_core.cpio.gz exist

[OK] FSB_core.cpio.gz size

[OK] The backup 2010_1221_0117.UTC__test has all files
needed for a restore

Warning: All software on the FSBs that are common to all
the NCBs and GPBs

        on the node will be destroyed and replaced with
the backup,

169.254.226.254:/gsn/nodes/remsgsn03/backups/2010_1221_0117
.UTC__test

        If the current FSB OS is different from the
version on the backup,

        the FSBs will be up/down-graded with the backedup
OS version.

        The local FSB data will be replaced by the backup.

Proceed? [yes|no]

yes
```

Output

```
=====
=====
```

```
Restore of backup 2010_1221_0117.UTC__test started

=====
=====

-create temporary directory /tmp/DPE_CORE/tmp_fsb_local

-get      FSB_local      from      GSS      and      unpack      to
/tmp/DPE_CORE/tmp_fsb_local

-get      and      unpack      FSB_local.cpio.gz      to
/tmp/DPE_CORE/tmp_fsb_local

-remove FSB_local.cpio.gz

-restore FSB_local remotely on the secondary_FSB (fsb1)

[OK] fai_execute -t 60 fsb1 fai_fsbLocalFilesCopy succeeded

-restore FSB_local on the primary_FSB

---RESTORE FSB_local start:

-delete content of /mnt/local/etc

-copy FSB_local files from /tmp/DPE_CORE/tmp_fsb_local to
/mnt/local/etc

---RESTORE FSB_local done!

-remove temporary directory /tmp/DPE_CORE/tmp_fsb_local

-create temporary directory /export/Core/tmp_fsbos

-restore FSB_os files (uboot, kernel, os)

-get and unpack FSB_os.cpio.gz to /export/Core/tmp_fsbos

-remove FSB_os.cpio.gz

-fsbos_version:                current      =
1./main/ndpgsn_4_0/ndpgsn_4_0_nir20j/3,      new      =
1./main/ndpgsn_4_0/ndpgsn_4_0_nir20j/3

-flash FSB_os files (uboot, kernel, os) if different to
current

Wed Jan 16 04:43:45 UTC 2010 upgrade_check: Checking FSB
for U-Boot and FSBOS upgrades

Wed Jan 16 04:43:46 UTC 2010 upgrade_check: Using system
configuration at /export/Core/tmp_fsbos

Wed Jan 16 04:43:47 UTC 2010 upgrade_check: Current U-Boot
version: R1F
```

```
Wed Jan 16 04:43:47 UTC 2010 upgrade_check: New U-Boot
version: R1F

Wed Jan 16 04:43:47 UTC 2010 upgrade_check: No U-Boot
upgrade

Wed Jan 16 04:43:47 UTC 2010 upgrade_check: Current FSBOS
version: 1./main/ndpgsn_4_0/ndpgsn_4_0_nir20j/3

Wed Jan 16 04:43:47 UTC 2010 upgrade_check: New FSBOS
version: 1./main/ndpgsn_4_0/ndpgsn_4_0_nir20j/3

Wed Jan 16 04:43:47 UTC 2010 upgrade_check: No FSBOS
upgrade since versions are same

Wed Jan 16 04:43:47 UTC 2010 upgrade_check: fsb2 is done!

Wed Jan 16 04:43:48 UTC 2010 upgrade_check: no upgrades

-remove temporary directory /export/Core/tmp_fsbos

-restore set_action_in_flash(restore) on the secondary_FSB
(fsb1)

-prevent FSB reboot

- fai_announceNclIsAlive

Wed Jan 16 04:43:55 UTC 2010 dhcp: DHCP is stopped

Wed Jan 16 04:43:56 UTC 2010 dhcp: DHCP is started

fai_configAllClientBoot.sh off

Wed Jan 16 04:43:56 UTC 2010 fai_configAllClientBoot.sh:
DHCP is stopped

-Reboot all non FSB boards

-Stopping monitord

-Stopped monitord

-disconnect /export/Core

-end all processes that are using /export/Core

/export/Core:          1234c  1515c

-unexport all NFS directories

-unmount /export/Core

[OK] -umount /export/Core

pkill remote_control
```

```
-kick watchdog and format /dev/drbd0

Wed Jan 16 04:44:04 UTC 2010 gsa_cli: Creating EXT3 file
system on /dev/drbd0

-mount /export/Core to /dev/drbd0

-get and unpack FSB_core.cpio.gz to /export/Core

-remove FSB_core.cpio.gz

-disconnect /export/charging

-end all processes that are using /export/charging

-unexport all NFS directories

-unmount /export/charging

[OK] -unmount /export/charging

-kick watchdog and format /dev/drbd1

Wed Jan 16 04:45:50 UTC 2010 gsa_cli: Creating EXT3 file
system on /dev/drbd1

-mount /export/charging to /dev/drbd1

-disconnect /export/logs

-end all processes that are using /export/logs

-unexport all NFS directories

-unmount /export/logs

[OK] -unmount /export/logs

-kick watchdog and format /dev/drbd2

Wed Jan 16 04:46:16 UTC 2010 gsa_cli: Creating EXT3 file
system on /dev/drbd2

-mount /export/logs to /dev/drbd2

Saving the System Clock time to the Hardware
Clock...Hardware Clock updated to Wed Jan 16 04:46:23 UTC
2008.

Wed Jan 16 04:46:23 UTC 2010 gsa_cli: Saved current time to
HW clock

-export all NFS directories

-stop watchdog
```

```
=====
=====

Restore done, rebooting FSB now!

=====
=====

-node_restart -c

Wed Jan 16 04:46:24 UTC 2010 fai_nodeRestart.sh: Rebooting
FSBs

Wed Jan 16 04:46:24 UTC 2010 setup_sc_in_fs: Clean
/tmp/DPE_CORE/SCs

Wed Jan 16 04:46:24 UTC 2010 setup_sc_in_fs: Set up link
/tmp/DPE_CORE/SCs/sc.1

Wed Jan 16 04:46:25 UTC 2010 setup_sc_in_fs: Changed DHCP
configuration

Wed Jan 16 04:46:25 UTC 2010 fai_nodeRestart.sh: Sleeping
10 seconds for other FSB to be stopped
```

Intentionally Blank