3GPP TSG CN Plenary Meeting #17 **NP-020454**
4th – 6th September 2002 Biarritz, FRANCE.

Source: TSG CN WG4

Title: TEI5

Agenda item: 8.8

Document for: APPROVAL

Spec	CR	Rev	Doc-2nd-Level	Phase	Subject	Cat	Ver_C
23.083	010		N4-020808	Rel5	Determining the basic service for MT calls	F	5.0.0
29.002	465		N4-020841	Rel5	Clarification on Call Deflection	F	5.2.0
23.003	053		N4-020892	Rel5	Restructuring the IMEI to combine the TAC and FAC in Annex B	F	5.3.0
24.080	024		N4-020895	Rel5	Correction of references to FACILITY information element	F	5.1.0
23.018	110		N4-020917	Rel5	Minor corrections to Process ICH_MSC	F	5.3.0
24.080	019	3	N4-021000	Rel5	Compatible upgrade to ASN.1:1997 of 24.080	F	5.1.0
29.002	437	3	N4-021001	Rel5	Compatible upgrade to ASN.1:1997 of 29002	F	5.2.0
23.018	109	1	N4-021039	Rel5	Determining the basic service for MT calls	F	5.3.0
29.002	470	1	N4-021040	Rel5	Correction to the usage of "Roaming not allowed" error	F	5.2.0
29.060	325	2	N4-021011	Rel5	RAB Setup Information for IPv6	F	5.2.0
29.060	329	1	N4-021110	Rel5	Addition of PCO IE to Update PDP context procedures	F	5.2.0
29.002	473	2	N4-021108	Rel5	Available codecs list and selected codec indication	F	5.2.0

3GPP TSG CN WG4 Meeting #15 **N4-020892**
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **23.003** **CR** **053** ⌘ **rev** **-** ⌘ Current version: **5.3.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Restructuring the IMEI to combine the TAC and FAC in Annex B
Source:	⌘	CN4
Work item code:	⌘	TEI5 ***Date:*** ⌘ 16/07/2002
Category:	⌘	**F** ***Release:*** ⌘ Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	CR 23.003-044 submitted to CN4#14 as N4-020713, and approved in CN #16, did not reflect the changed structure of the IMEI in the procedure for computing the Luhn check digit, described in Annex B. This change is proposed as a non-critical correction, agreed by consensus
Summary of change:	⌘	Replace the separate 6-digit TAC (Type Approval Code) and 2-digit FAC (Final Assembly Code) with an 8-digit TAC (Type *Allocation* Code) in Annex B.
Consequences if not approved:	⌘	Misalignment between the definitions of the IMEI structure in subclause 6.2 and Annex B

Clauses affected:	⌘	Annex B

Other specs affected: ⌘

Y	N	
	X	Other core specifications
	X	Test specifications
	X	O&M Specifications

⌘

Other comments: ⌘

Annex B (normative): IMEI Check Digit computation

B.1 Representation of IMEI

The International Mobile station Equipment Identity and Software Version Number (IMEISV), as defined in TS 23.003, is a 16 digit decimal number composed of four distinct elements:

- an 86 digit Type ~~Approval~~ Allocation Code (TAC);
- ~~a 2 digit Final Assembly Code (FAC);~~
- a 6 digit Serial Number (SNR); and
- a 2 digit Software Version Number (SVN).

The IMEISV is formed by concatenating these ~~four~~ three elements as illustrated below:

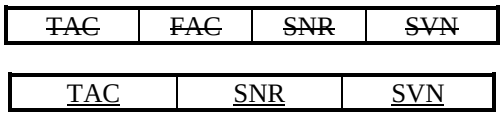


Figure A.1: Composition of the IMEISV

The IMEI is complemented by a check digit as defined in section 3. The Luhn Check Digit (CD) is computed on the 14 most significant digits of the IMEISV, that is on the value obtained by ignoring the SVN digits.

The method for computing the Luhn check is defined in Annex B of the International Standard "Identification cards - Numbering system and registration procedure for issuer identifiers" (ISO/IEC 7812) [3].

In order to specify precisely how the CD is computed for the IMEI, it is necessary to label the individual digits of the IMEISV, excluding the SVN. This is done as follows:

The (14 most significant) digits of the IMEISV are labelled D14 D13 ... D1, where:

- TAC = D14 D13 ... D79 (with D79 the least significant digit of TAC);
- ~~FAC = D8 D7 (with D7 the least significant digit of FAC); and~~
- SNR = D6 D5 ... D1 (with D1 the least significant digit of SNR).

B.2 Computation of CD for an IMEI

Computation of CD from the IMEI proceeds as follows:

Step 1: Double the values of the odd labelled digits D1, D3, D5 ... D13 of the IMEI.

Step 2: Add together the individual digits of all the seven numbers obtained in Step 1, and then add this sum to the sum of all the even labelled digits D2, D4, D6 ... D14 of the IMEI.

Step 3: If the number obtained in Step 2 ends in 0, then set CD to be 0. If the number obtained in Step 2 does not end in 0, then set CD to be that number subtracted from the next higher number which does end in 0.

B.3 Example of computation

IMEI (14 most significant digits):

TAC								SNR					
D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1
2	6	0	5	3	1	7	9	3	1	1	3	8	3

TAC								SNR					
D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1
2	6	0	5	3	1	7	9	3	1	1	3	8	3

Step 1:

2	6	0	5	3	1	7	9	3	1	1	3	8	3
	x2		x2		x2		x2		x2		x2		x2
	12		10		2		18		2		6		6
2	6	0	5	3	1	7	9	3	1	1	3	8	3
	x2		x2		x2		x2		x2		x2		x2
	12		10		2		18		2		6		6

Step 2:

2 + 1 + 2 + 0 + 1 + 0 + 3 + 2 + 7 + 1 + 8 + 3 + 2 + 1 + 6 + 8 + 6 = 53

Step 3:

CD = 60 - 53 = 7

3GPP TSG CN WG4 Meeting #15 ***N4-020917***
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **23.018** **CR** **110** ⌘ **rev** **-** ⌘ Current version: **5.3.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Minor corrections to Process ICH_MSC
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	17/07/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	In Process ICH_MSC: • On sheet 5, the Procedure CAMEL_Check_ORLCF_VMSC is incorrectly stated to be in 3GPP TS 23.079; it should be 3GPP TS 23.078. • On sheet 3, some comment boxes referencing 3GPP TS 23.078 are missing for some CAMEL procedures. • There is misalignment between the sheet numbers for the textual comments and the actual SDL sheet numbers.
Summary of change:	⌘	In Process ICH_MSC: • On sheet 5, the comment box is corrected to state that Procedure CAMEL_Check_ORLCF_VMSC is in 3GPP TS 23.078. • On sheet 2 and sheet 3, added some comment boxes referencing 3GPP TS 23.078 for CAMEL procedures. • Aligned sheet numbers in textual comments with the correct SDL sheet numbers.
Consequences if not approved:	⌘	Confusion for implementors in Process ICH_MSC over which specification Procedure CAMEL_Check_ORLCF_VMSC is defined in as well as all CAMEL procedures shown on sheet 2, and confusion when reading textual comments.

Clauses affected:	⌘	7.3.1			
Other specs affected:	⌘	**Y**	**N**		⌘
			X	Other core specifications	
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

7 Functional requirements of network entities

7.3 MT call

7.3.1 Functional requirements of serving MSC

7.3.1.1 Process ICH_MSC

Sheet 1: the rules for converting the ISDN BC/LLC/HLC to a bearer service or teleservice are specified in 3GPP TS 29.007 [30].

Sheet 1: the task "Store UUS information (if received)" is executed only if the VMSC supports UUS.

Sheet 1: the variables TCH allocated, ACM sent, Answer sent and Network connect sent are global data, accessible to the procedures Establish_Terminating_TCH_If_Required, Send_ACM_If_Required, Send_Answer_If_Required and Send_Network_Connect_If_Required.

Sheet 1: the variables UUS result sent, UUS1 implicit active, UUS1 explicit active, UUS2 active, UUS3 active and UUS CF interaction are specific to UUS. They are accessible to all UUS specific procedures.

Sheet 1: the handling starting with the input signal "Continue CAMEL handling" is specific to CAMEL phase 3 or later. If the VMSC does not support CAMEL phase 3 or later, this signal will not be received from the VLR.

Sheet 1: the procedure CAMEL_ICH_MSC_INIT is specific to CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 1: The variable "On_Hold" is used only if the VMSC supports Call Hold.

Sheet 1, sheet ~~4~~15, sheet 9: the process CAMEL_ICH_LEG1_MSC is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 2: the procedure Process_Access_Request_MSC is specified in subclause 7.1.1.2.

Sheet 2: the signal input Complete Call will be received in the state Wait_For_Page_Request only if the MSC/VLR supports pre-paging.

Sheet 2, sheet 3: the suggested mapping from values of the Send Info For Incoming Call negative response information element to values of the ISUP release cause (see ITU-T Recommendation Q.850 [37]) is shown in table 2. The mapping used is a matter for the network operator, depending on the telephony signalling system used.

Table 2: Suggested mapping of Send Info For Incoming Call (SIFIC) negative responses to ISUP release causes

SIFIC negative response	ISUP release cause number	ISUP release cause name
Absent subscriber	20	Subscriber absent
Busy subscriber	17	User busy
CUG reject (Called party SS interaction violation)	21	Call rejected
Forwarding violation	21	Call rejected
Impossible call completion	111	Protocol error, unspecified
No subscriber reply	19	No answer from user (user alerted)
System failure	111	Protocol error, unspecified
Unallocated roaming number	111	Protocol error, unspecified

Sheet 2, sheet 3, sheet 6~~5, sheet 7~~, sheet 8, sheet 10, sheet 12: the procedure CAMEL_MT_GMSC_DISC4 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "No" exit of the test "Result=Reconnect?".

Sheet 2, sheet 54, ~~sheet 7~~, sheet 8, sheet 10, sheet 11, sheet 12: the procedure CAMEL_MT_GMSC_DISC6 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 3: the procedure CAMEL_MT_GMSC_DISC5 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "No" exit of the test "Result=Reconnect?".

Sheet 3: the procedure CD_Reject is specific to Call Deflection; it is specified in 3GPP TS 23.072 [11].

Sheet 3: the procedure Process_Call_Waiting is specific to Call Waiting; it is specified in 3GPP TS 23.083 [16].

Sheet 3: the task "Store CW treatment indicator for this call if received in SII2" is executed only if the VMSC supports CAMEL phase 3 or later.

Sheet 3: ~~I~~if the VMSC does not support CAMEL phase 3 or later, the procedure Complete_Call_In_MSC and the procedure Process_Call_Waiting ~~MSC~~ will not return a "Reconnect" result.

Sheet 3: the processing in the branch starting with the input signal "Process Call Waiting" is specific to Call Waiting. If the VMSC does not support Call Waiting, this signal will not be received from the VLR.

~~Sheet 3: the procedure Process_Call_Waiting is specific to Call Waiting; it is specified in 3GPP TS 23.083 [16].~~

~~Sheet 3, sheet 8, the procedure CD_Reject is specific to Call Deflection; it is specified in 3GPP TS 23.072 [11].~~

Sheet 3, sheet 108: the procedure CCBS_Set_Diagnostic_For_Release is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 3, sheet 54, sheet 610, sheet 11, sheet 12, sheet 13: the procedure CCBS_Check_Last_Call is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

~~Sheet 3, sheet 11, sheet 13: signals are sent to and received from the process Subs_FSM; it is specified in subclause 7.4.~~

Sheet 34: the procedure UUS_ICH_Check_Support is specific to UUS; it is specified in 3GPP TS 23.087 [20].

Sheet 4, ~~sheet 9, sheet 15~~: the procedure~~s~~ CAMEL_ICH_LEG2_MSC ~~and CAMEL_ICH_LEG2_CF_MSC are~~ isspecific to CAMEL phase 4 or later; ~~they are~~it is specified in 3GPP TS 23.078 [12].

Sheet 9: the procedure CAMEL_ICH_LEG2_CF_MSC is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 54: the procedure CAMEL_Check_ORLCF_VMSC is specific to CAMEL phase 2 or later; it is specified in 3GPP TS 23.078 [12].

- If the VLR does not support CAMEL or no CAMEL information is available for the subscriber, then ORLCF may take place ('ORLCF' result from CAMEL_Check_ORLCF_VMSC).

  If CAMEL information is available for the subscriber and the GMSC supports the required CAMEL phase, then ORLCF may take place. The Resume Call Handling request shall include the relevant CAMEL information ('ORLCF' result from CAMEL_Check_ORLCF_VMSC).

  If CAMEL information is available for the subscriber but the GMSC does not support the required CAMEL phase, then ORLCF shall not take place ('VMSCCF' result from CAMEL_Check_ORLCF_VMSC).

Sheet 54: the procedure Handle_ORLCF_VMSC is specific to Support of Optimal Routeing. It is specified in 3GPP TS 23.07989 [13]. If the VMSC does not support Optimal Routeing, processing continues from the "Continue" exit of the test "Result~~Forwarding Failed~~?".

Sheet 5, sheet 64, ~~sheet 9~~, sheet 11: the procedures CD_Failure and CD_Success are specific to Call Deflection; they are specified in 3GPP TS 23.072 [11].

Sheet 65: the procedure CAMEL_MT_VMSC_Notify_CF is specific to CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 65: If the VMSC does not support CAMEL phase 3 or later, processing starts with the possible call of the procedure CCBS_Check_Last_Call.

Sheet 65: The task "set redirection information" includes the mapping of the MSISDN parameter received in the Send Info For Incoming Call ack message to the redirecting number of the IAM message and the setting of the presentation indicator of the redirecting number of the IAM message according to the value of the Redirecting presentation parameter received in the Send Info For Incoming Call ack message.

Sheet 65: it is an operator option whether to send an Address Complete message if the VLR returns forwarding information. If the VMSC sends an Address Complete message, it shall include the called party's status field of the Backward call indicator set to "no indication".

Sheet 65, sheet 87: the procedure Send_ACM_If_Required is specified in subclause 7.2.1.3.

Sheet 65: the procedure Activate_CF_Process is specified in subclause 7.2.1.7.

Sheet 65: the procedure UUS_ICH_Set_Info_In_IAM is specific to UUS, it is specified in 3GPP TS 23.087 [20].

Sheet 65: the called party address sent in the IAM to the process MT_CF_MSC is the Forwarded-to number received in the Perform Call Forwarding ack.

Sheet 65: the procedure CAMEL_Store_Destination_Address is specific to CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

~~Sheet 65; the procedure CD_Success is specific to Call Deflection; it is specified in 3GPP TS 23.0 72 [11].~~

Sheet 76: The processing on this sheet is specific to CAMEL phase 3 or later. If the VMSC does not support CAMEL phase 3 or later, the input signal Int_Release Call will not be received.

Sheet 87: the procedure CAMEL_MT_GMSC_ANSWER is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "Pass" exit of the test "Result?".

Sheet 87: the procedure Handle_COLP_Forwarding_Interaction_MSC is specified in subclause 7.2.1.6.

Sheet 87: the procedure Send_Answer_If_Required is specified in subclause 7.2.1.4.

Sheet 87: the procedure Send_Network_Connect_If_Required is specified in subclause 7.2.1.5.

Sheet 87: the procedure CAMEL_MT_MSC_ALERTING is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 4 or later, processing continues from the "Pass" exit of the test "Result?".

Sheet 108: the procedure CCBS_MT_MSC_Check_Forwarding is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 119: the processing on this sheet is specific to CAMEL phase 3 or later. If the VMSC does not support CAMEL phase 3 or later, the input signal Send Info For MT Reconnected Call ack will not be received.

Sheet 119: the procedure Handle_ORLCF_VMSC is specific to OR; it is specified in 3GPP TS 23.079 [13]. If the VMSC does not support OR, processing continues from the "No" exit of the test "Result = Forwarding Failed?".

Sheet 1311, sheet 1412: the procedure CAMEL_MT_GMSC_DISC1 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 1311, sheet 1412: the procedure CAMEL_MT_GMSC_-DISC2 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "No" exit of the test "Result=Reconnect?".

Sheet 1311: the procedure UUS_MSC_Check_UUS1_UUI is specific to UUS; it is specified in 3GPP TS 23.087 [20].

Sheet 1412: after the VMSC has sent an IAM to the process MT_CF_MSC, it acts as a transparent relay for messages received from the GMSC and the process MT_CF_MSC. Any message other than Address Complete, Connect, Answer or Release causes no change of state in the process ICH_MSC.

Sheet 1513: The processing on this sheet is specific to CAMEL phase 3 or later. If the VMSC does not support CAMEL phase 3 or later, the input signal Int_Release Call will not be received.

Sheet 1614: the procedure Process_Hold_Request is specific to Call Hold; it is specified in 3GPP TS 23.083[16].

Sheet 1614: the procedure Process_Retrieve_request is specific to Call_Hold; it is specified in 3GPP TS 23.083[16].

~~Sheet 15: the procedures CAMEL_ICH_LEG2_MSC and CAMEL_ICH_LEG2_CF_MSC are specific to CAMEL phase 4 or later; they are specified in 3GPP TS 23.078 [12].~~

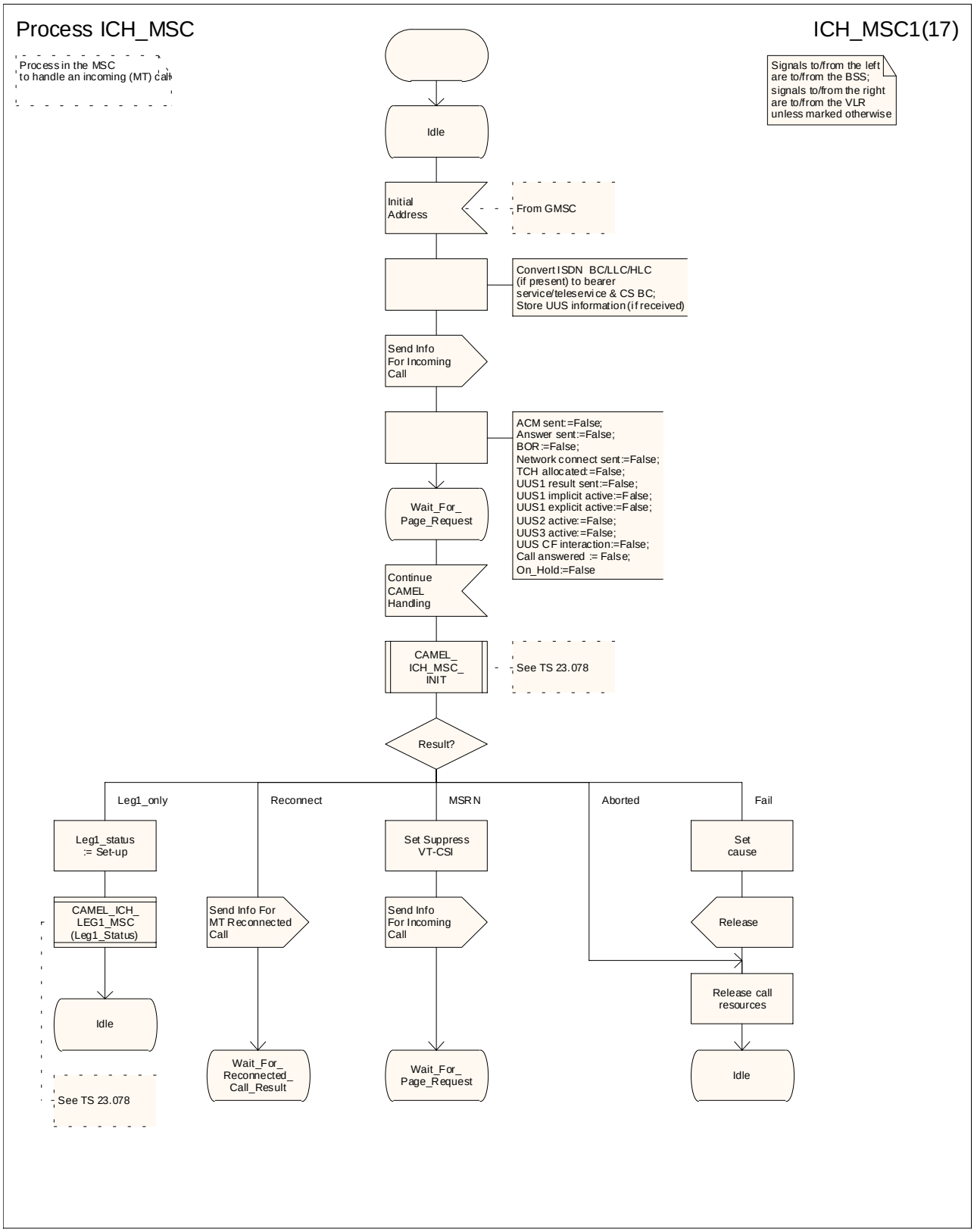
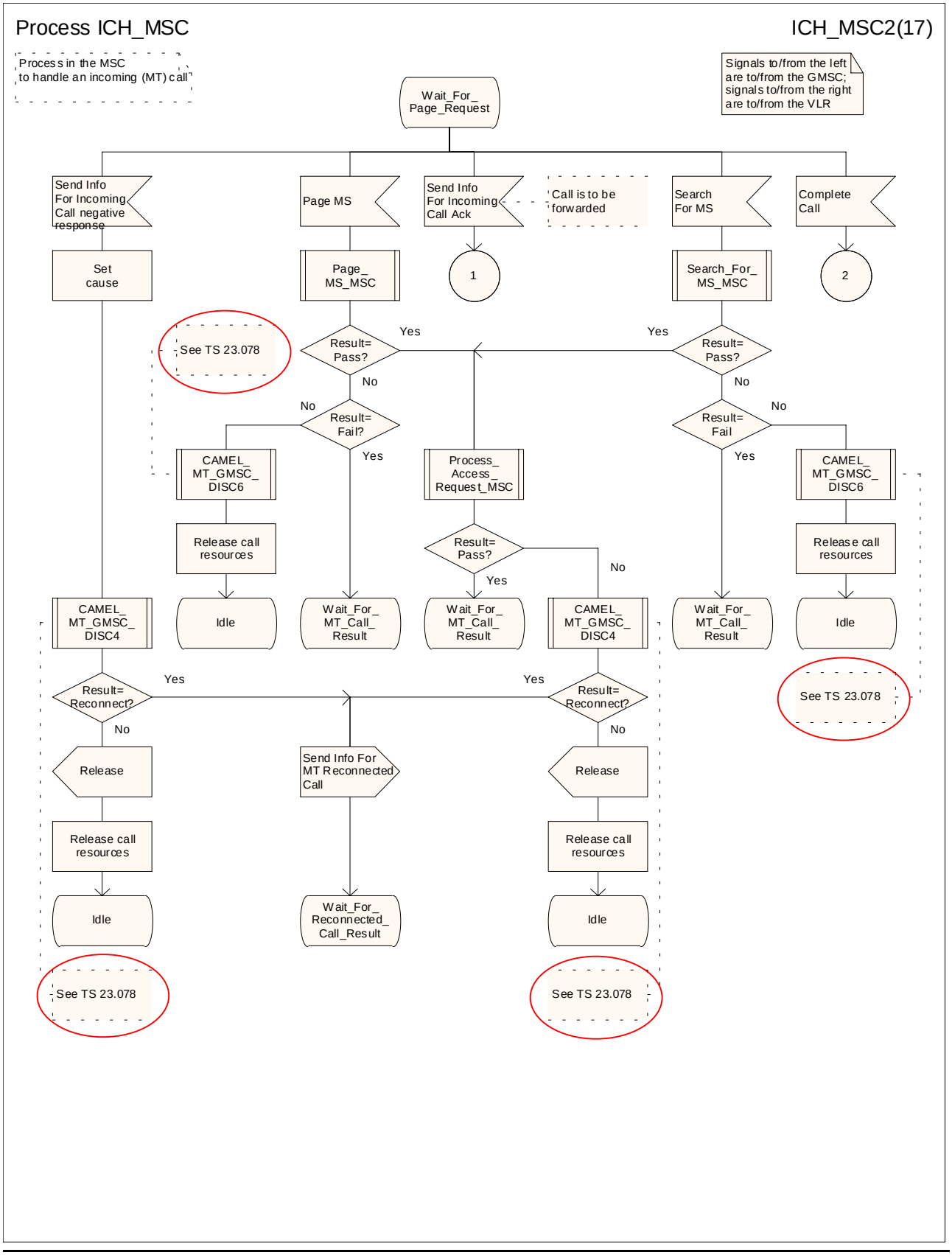


Figure 67a: Process ICH_MSC (sheet 1)


Process ICH_MSC
ICH_MSC2(17)
Process in the MSC to handle an incoming (MT) call
Signals to/from the left are to/from the GMSC; signals to/from the right are to/from the VLR
Wait_For_ Page_Request
Send Info For Incoming Call negative response
Page MS
Send Info For Incoming Call Ack
Call is to be forwarded
Search For MS
Complete Call
Set cause
Page_ MS_MSC
1
Search_For_ MS_MSC
2
See TS 23.078
Result= Pass?
Yes
No
Result= Fail?
Result= Fail
CAMEL_ MT_GMSC_ DISC6
Process_ Access_ Request_MSC
Release call resources
Idle
Wait_For_ MT_Call_ Result
CAMEL_ MT_GMSC_ DISC4
Result= Reconnect?
Release
Release call resources
Idle
See TS 23.078
Send Info For MT Reconnected Call
Wait_For_ Reconnected_ Call_Result

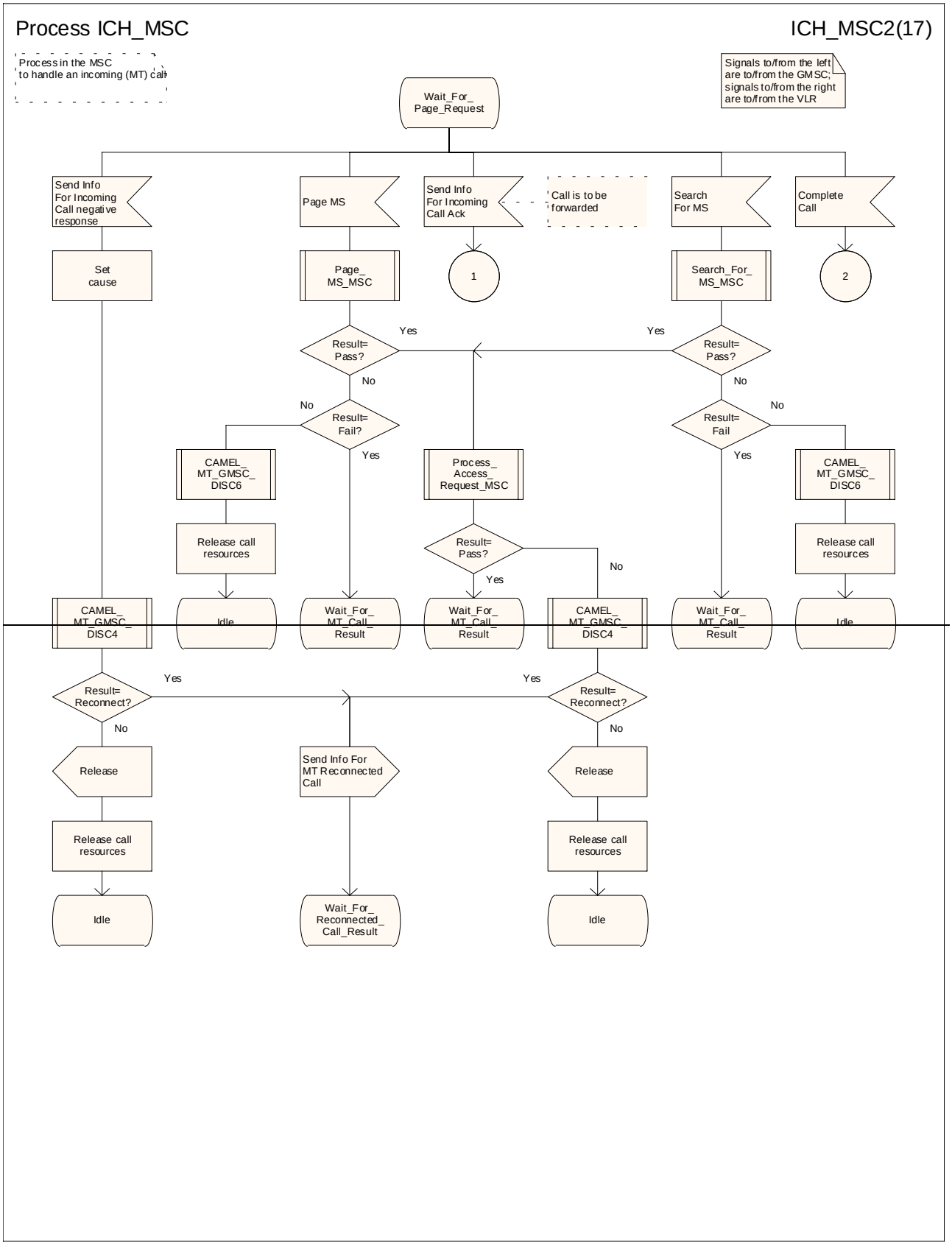
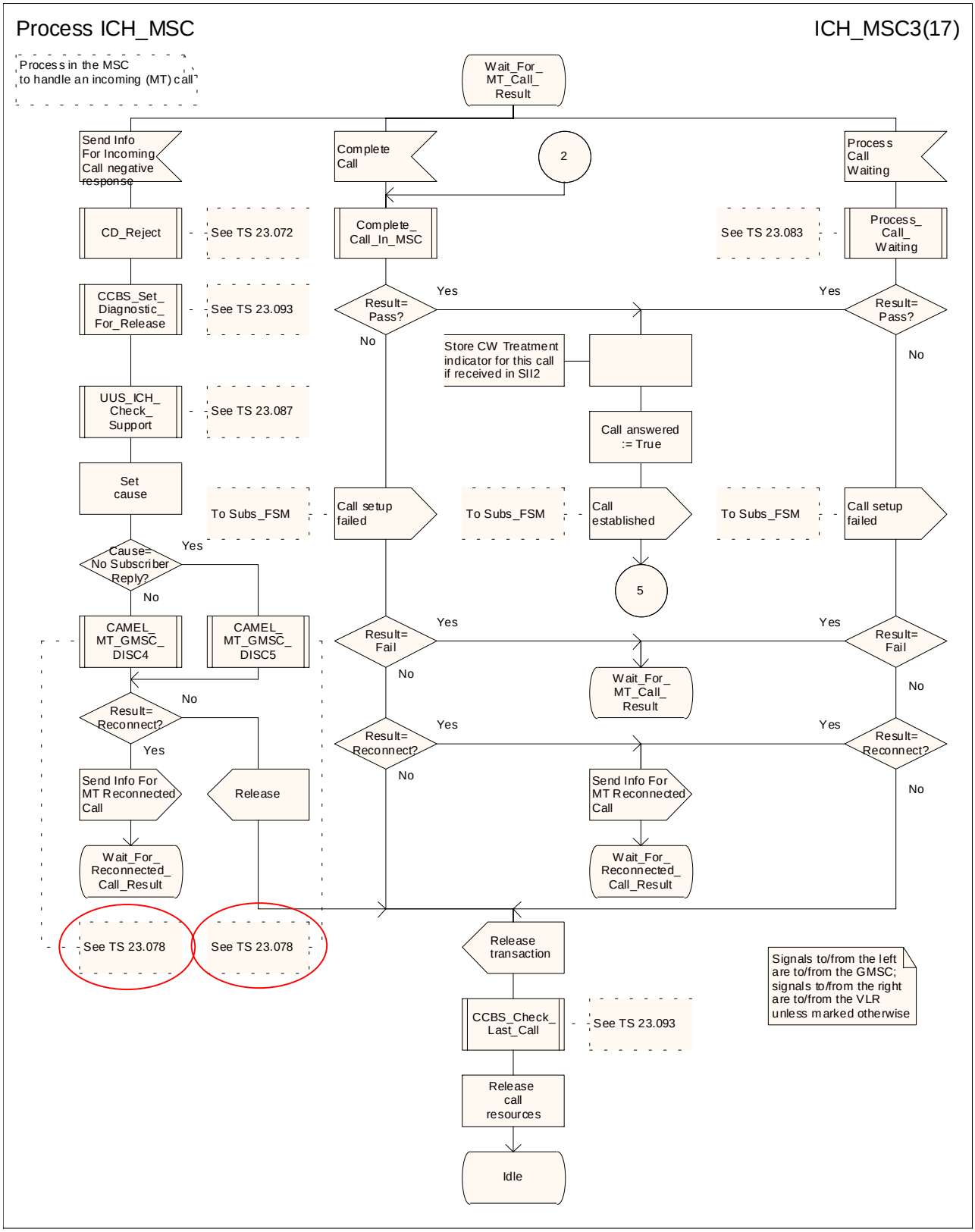


Figure 67b: Process ICH_MSC (sheet 2)


Process ICH_MSC
ICH_MSC3(17)
Process in the MSC to handle an incoming (MT) call
Wait_For_ MT_Call_ Result
Send Info For Incoming Call negative response
Complete Call
2
Process Call Waiting
CD_Reject
See TS 23.072
Complete_ Call_In_MSC
See TS 23.083
Process_ Call_ Waiting
CCBS_Set_ Diagnostic_ For_Release
See TS 23.093
Result= Pass?
Yes
No
Store CW Treatment indicator for this call if received in SII2
UUS_ICH_ Check_ Support
See TS 23.087
Call answered := True
Set cause
To Subs_FSM
Call setup failed
To Subs_FSM
Call established
To Subs_FSM
Call setup failed
Cause= No Subscriber Reply?
Yes
No
5
CAMEL_ MT_GMSC_ DISC4
CAMEL_ MT_GMSC_ DISC5
Result= Fail
Wait_For_ MT_Call_ Result
Result= Reconnect?
Send Info For MT Reconnected Call
Release
Wait_For_ Reconnected_ Call_Result
See TS 23.078
See TS 23.078
Release transaction
CCBS_Check_ Last_Call
See TS 23.093
Release call resources
Idle
Signals to/from the left are to/from the GMSC; signals to/from the right are to/from the VLR unless marked otherwise

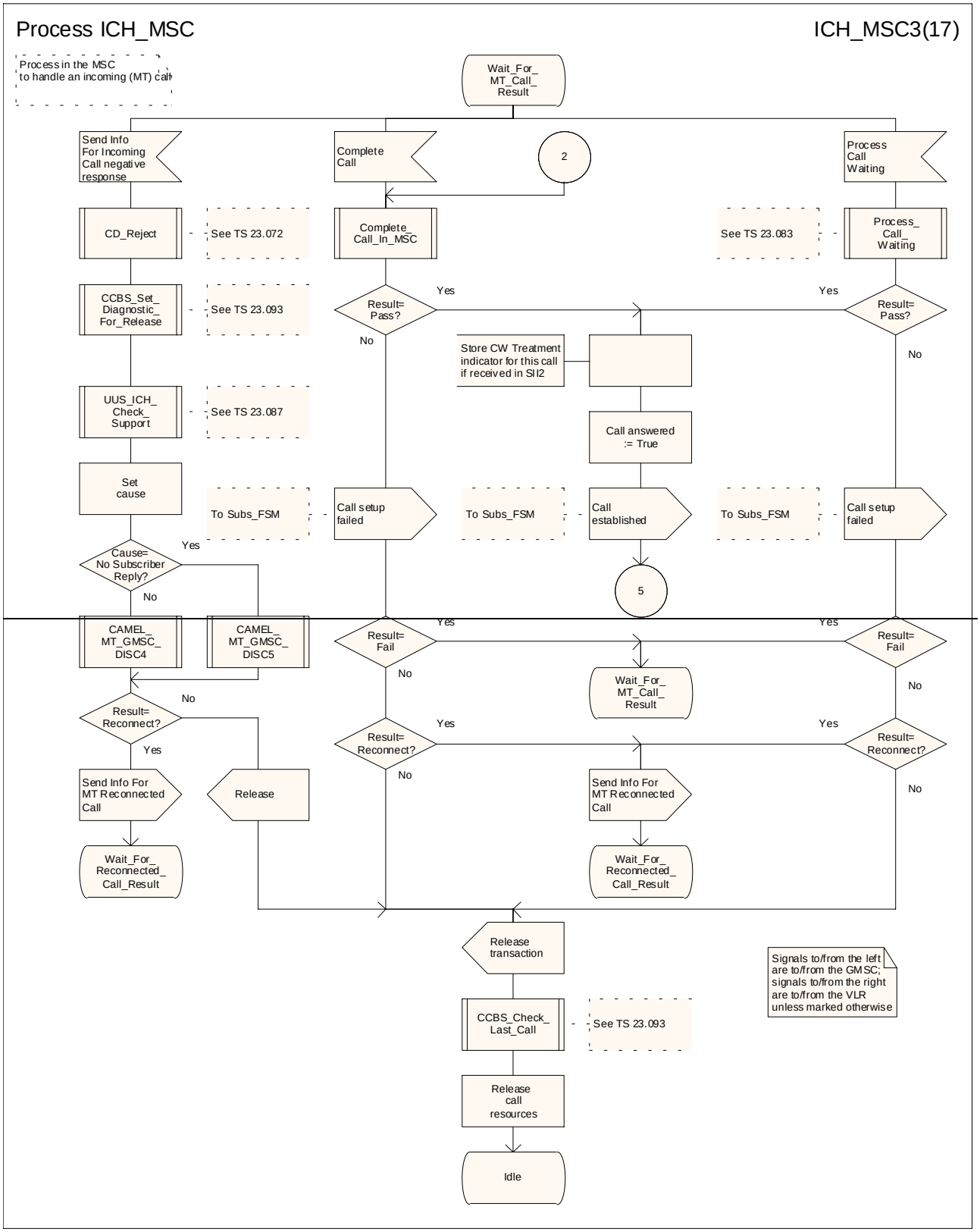


Figure 67c: Process ICH_MSC (sheet 3)

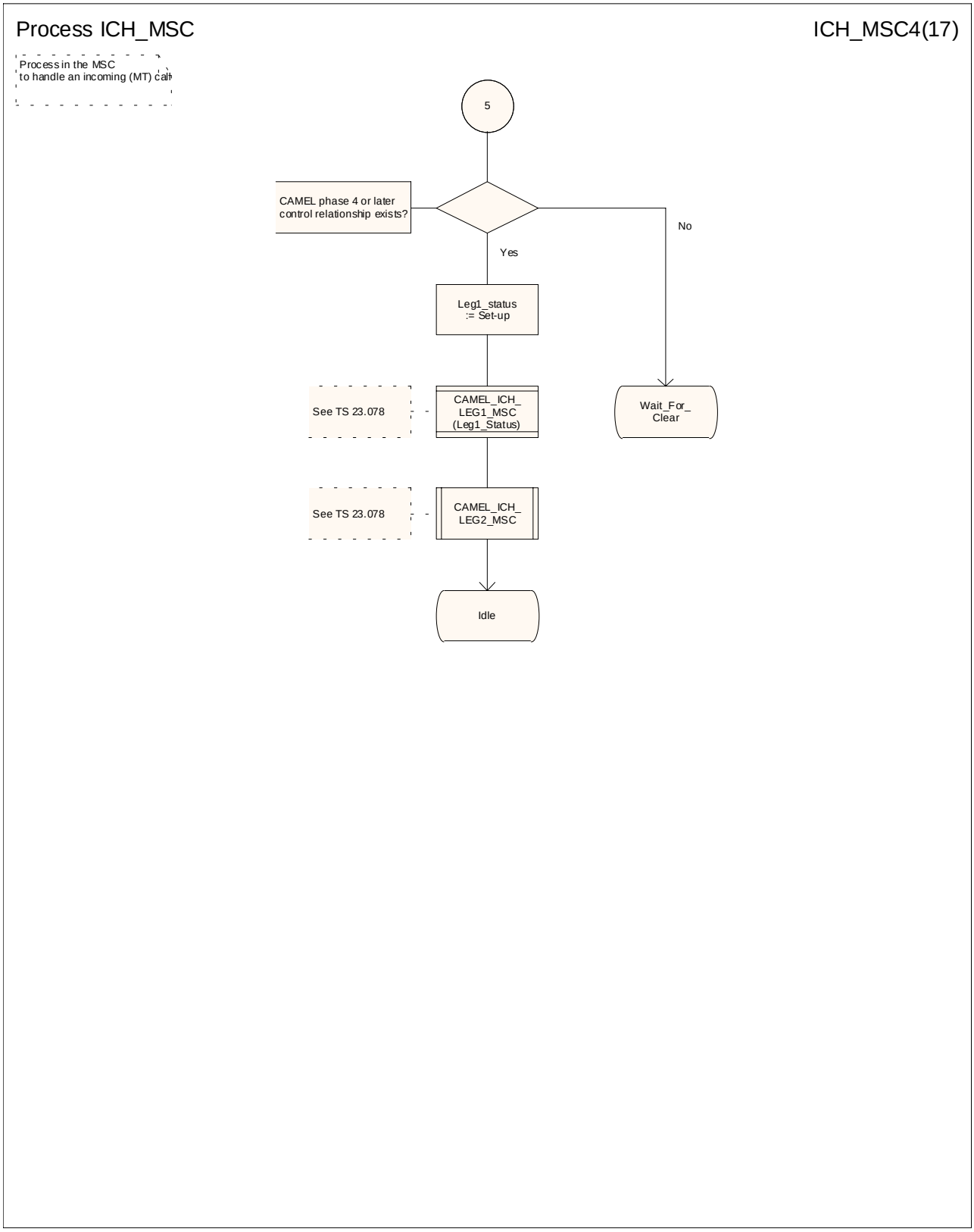
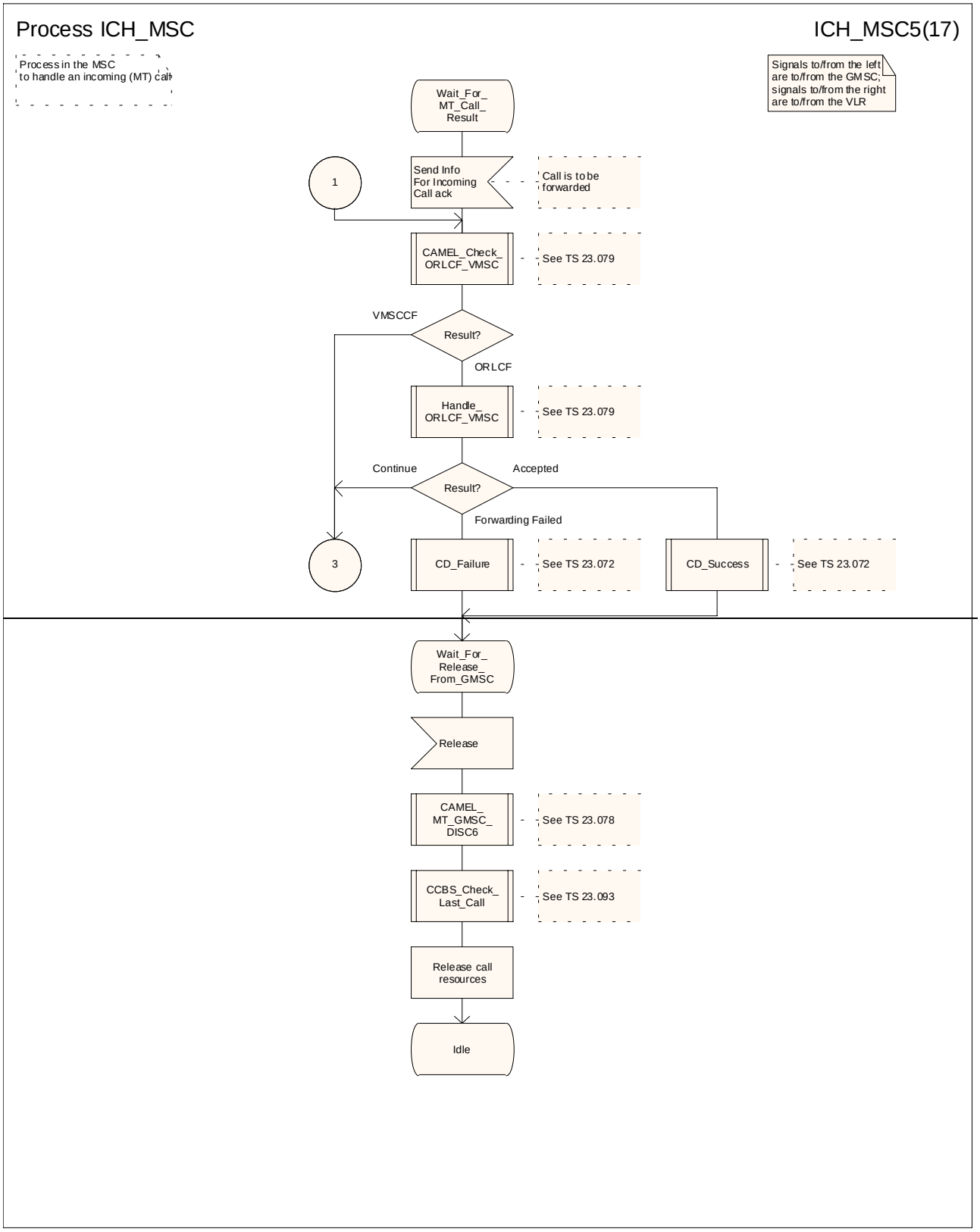


Figure 67d: Process ICH_MSC (sheet 4)


Process ICH_MSC
ICH_MSC5(17)
Process in the MSC to handle an incoming (MT) call
Signals to/from the left are to/from the GMSC; signals to/from the right are to/from the VLR
Wait_For_ MT_Call_ Result
1
Send Info For Incoming Call ack
Call is to be forwarded
CAMEL_Check_ ORLCF_VMSC
See TS 23.079
VMSCCF
Result?
ORLCF
Handle_ ORLCF_VMSC
See TS 23.079
Continue
Accepted
Result?
Forwarding Failed
3
CD_Failure
See TS 23.072
CD_Success
See TS 23.072
Wait_For_ Release_ From_GMSC
Release
CAMEL_ MT_GMSC_ DISC6
See TS 23.078
CCBS_Check_ Last_Call
See TS 23.093
Release call resources
Idle

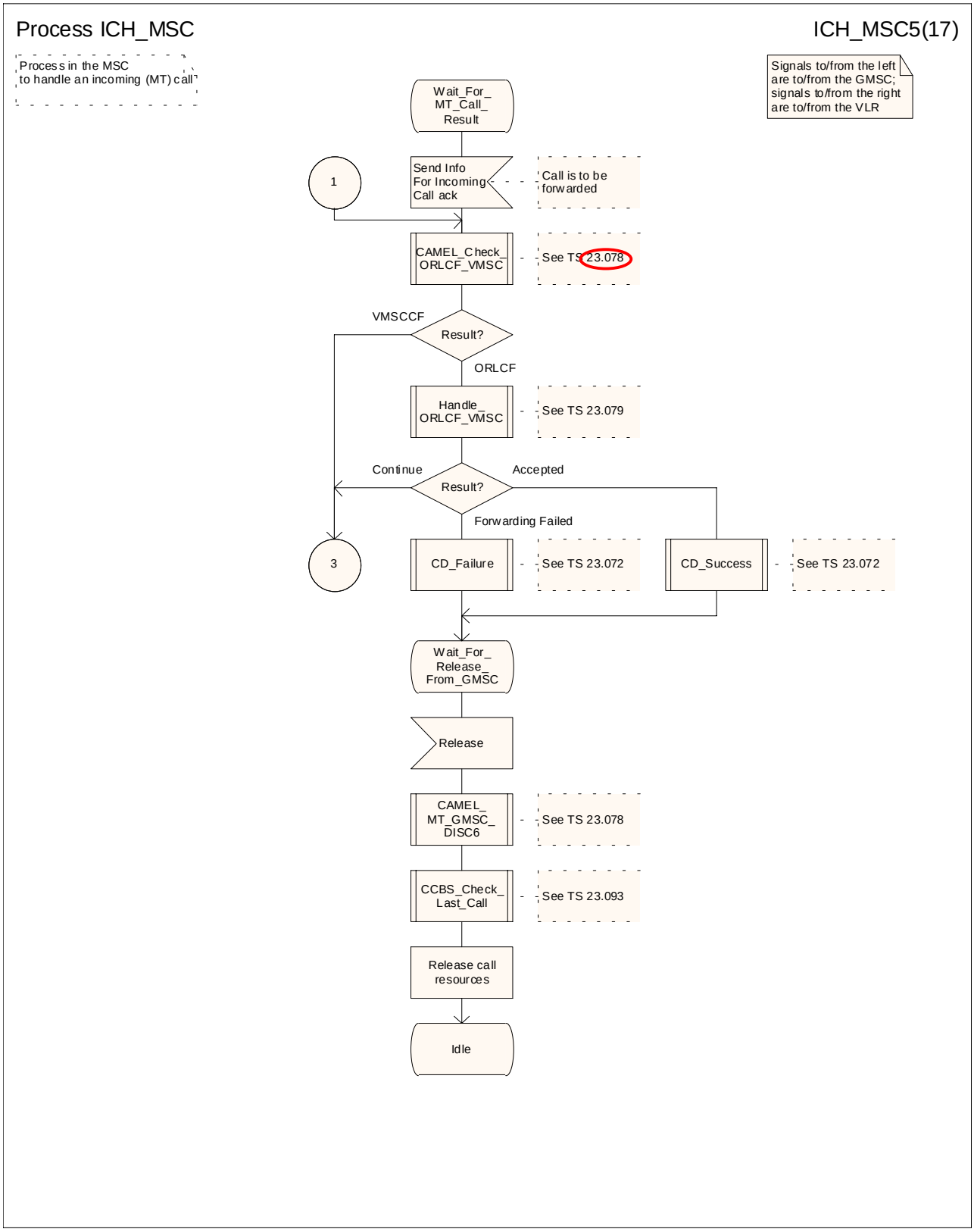


Figure 67e: Process ICH_MSC (sheet 5)

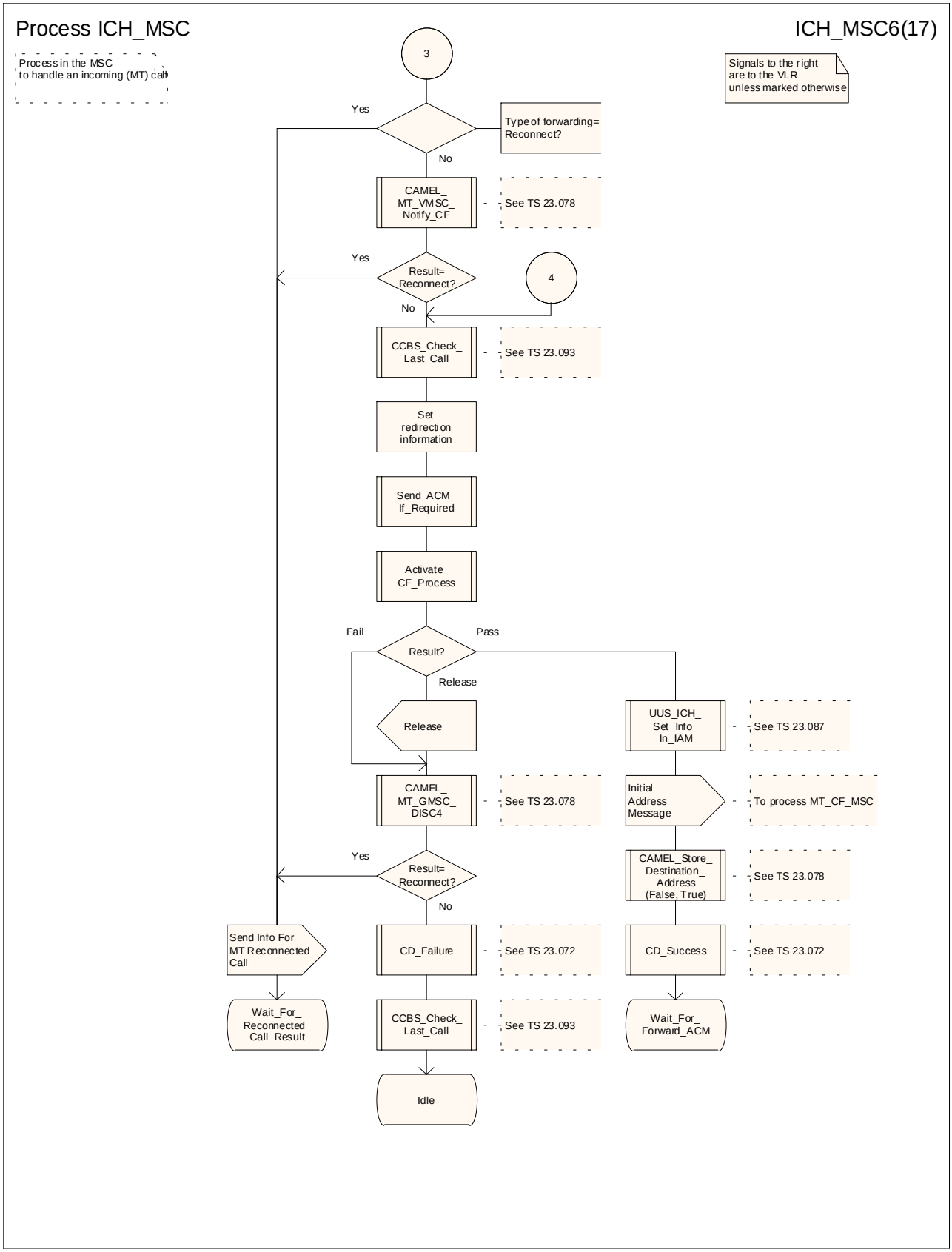


Figure 67f: Process ICH_MSC (sheet 6)

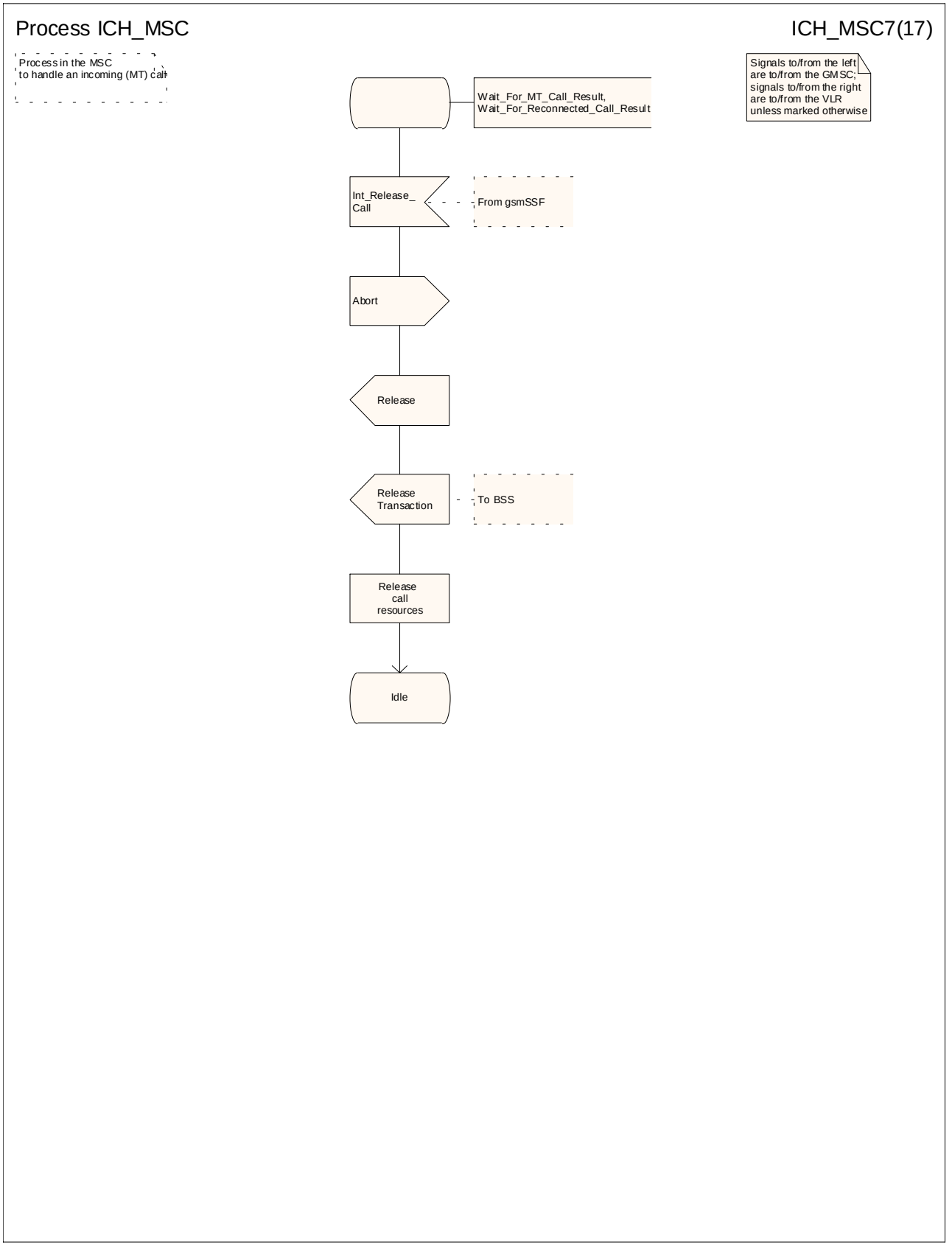


Figure 67g: Process ICH_MSC (sheet 7)

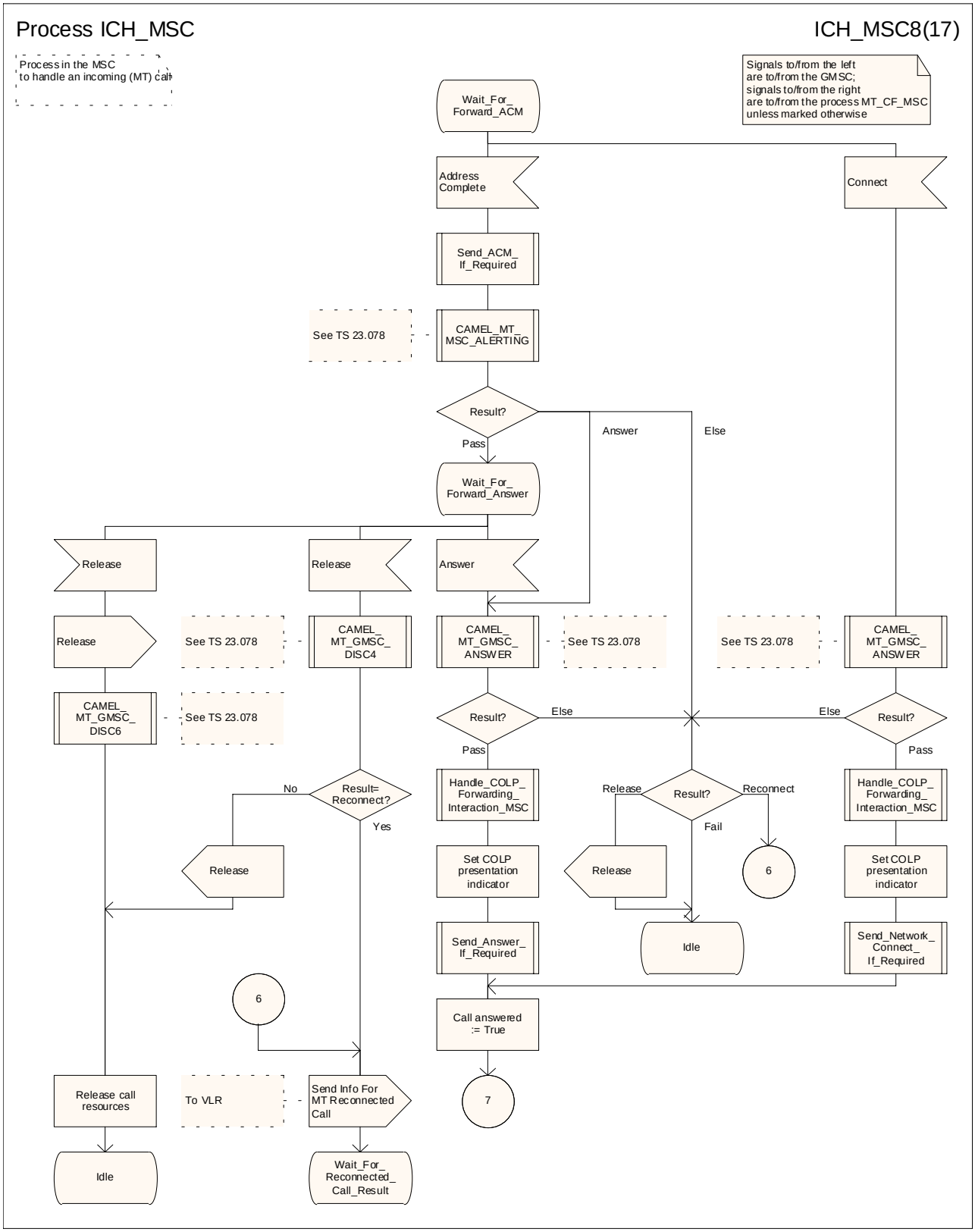


Figure 67h: Process ICH_MSC (sheet 8)

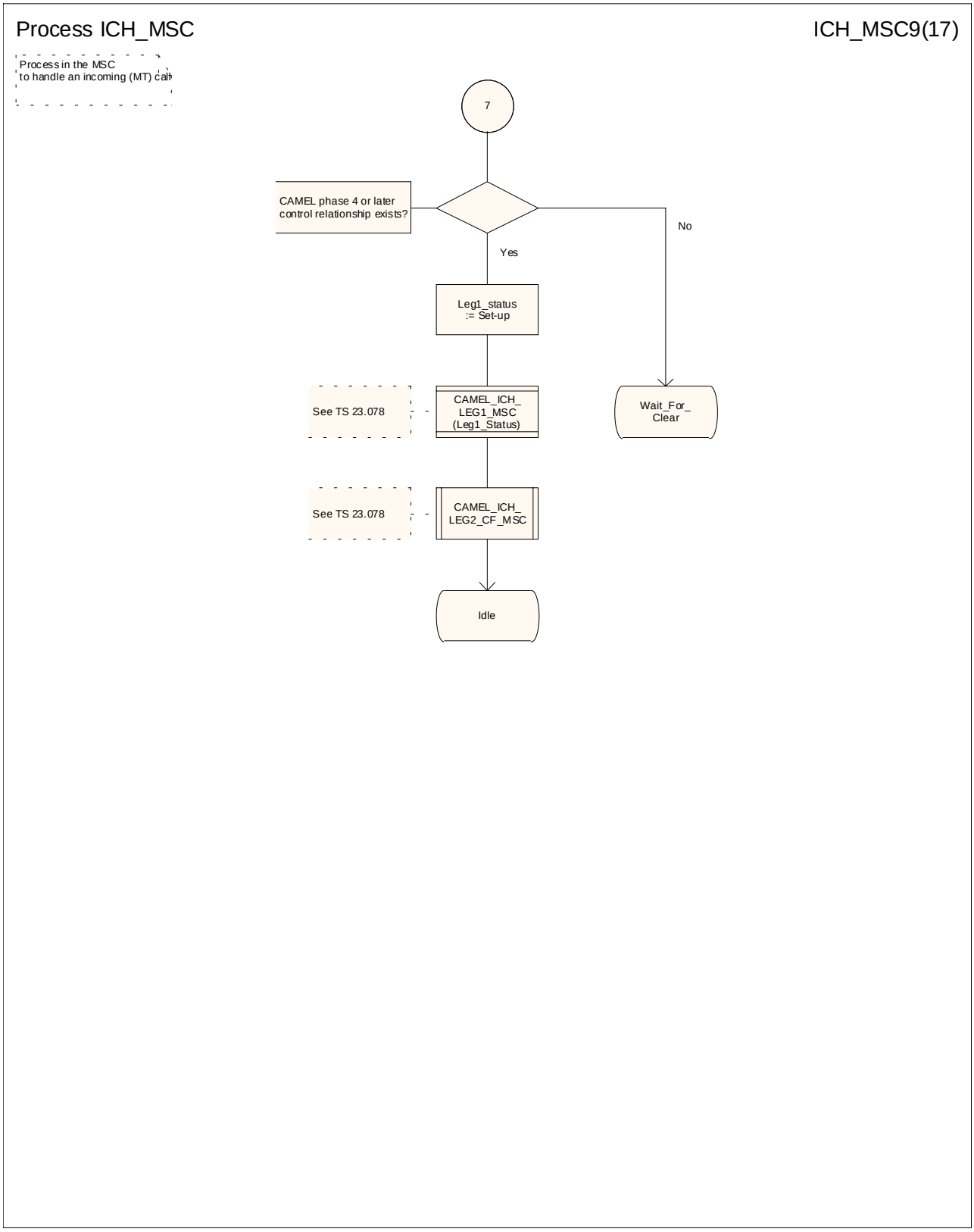


Figure 67i: Process ICH_MSC (sheet 9)

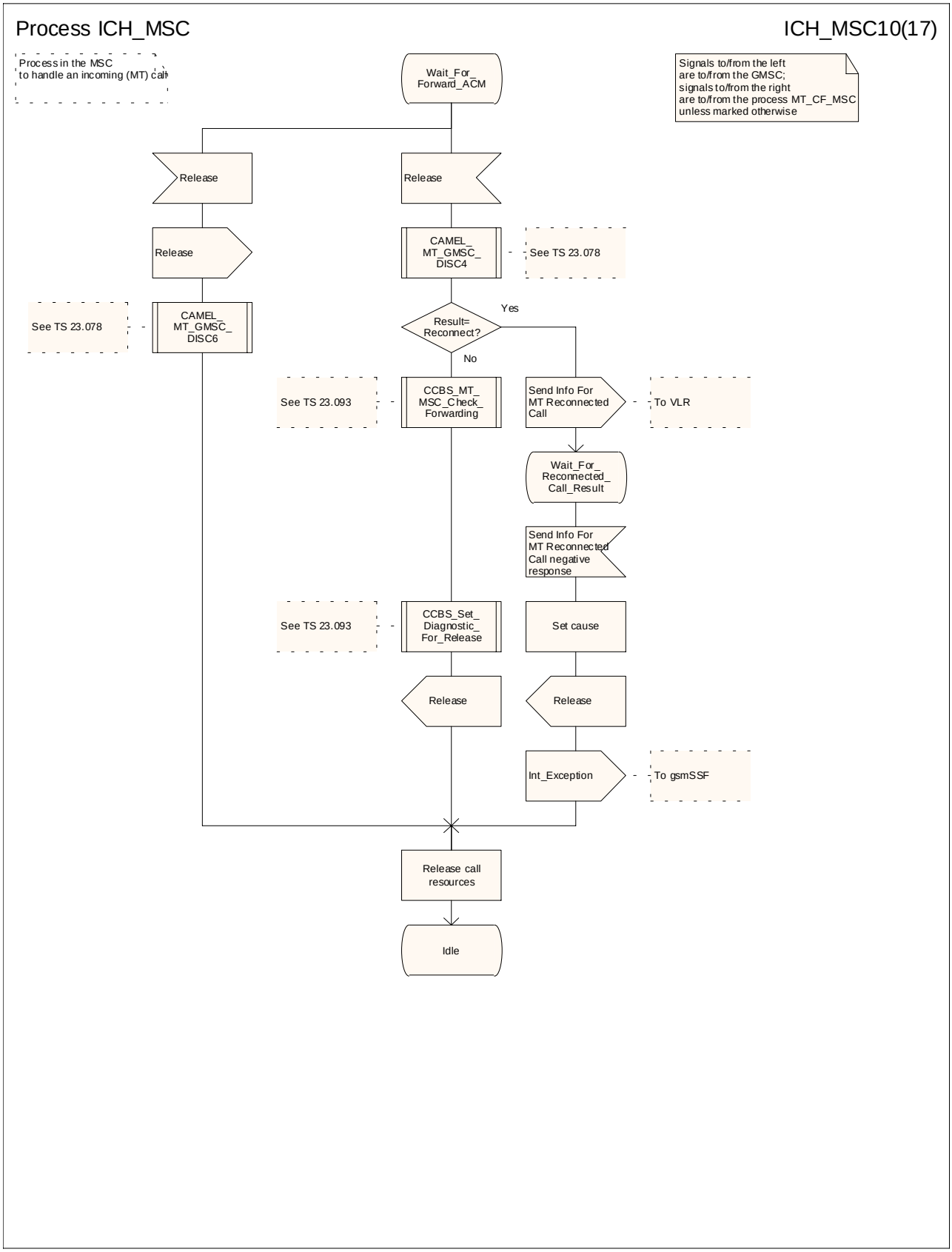


Figure 67j: Process ICH_MSC (sheet 10)

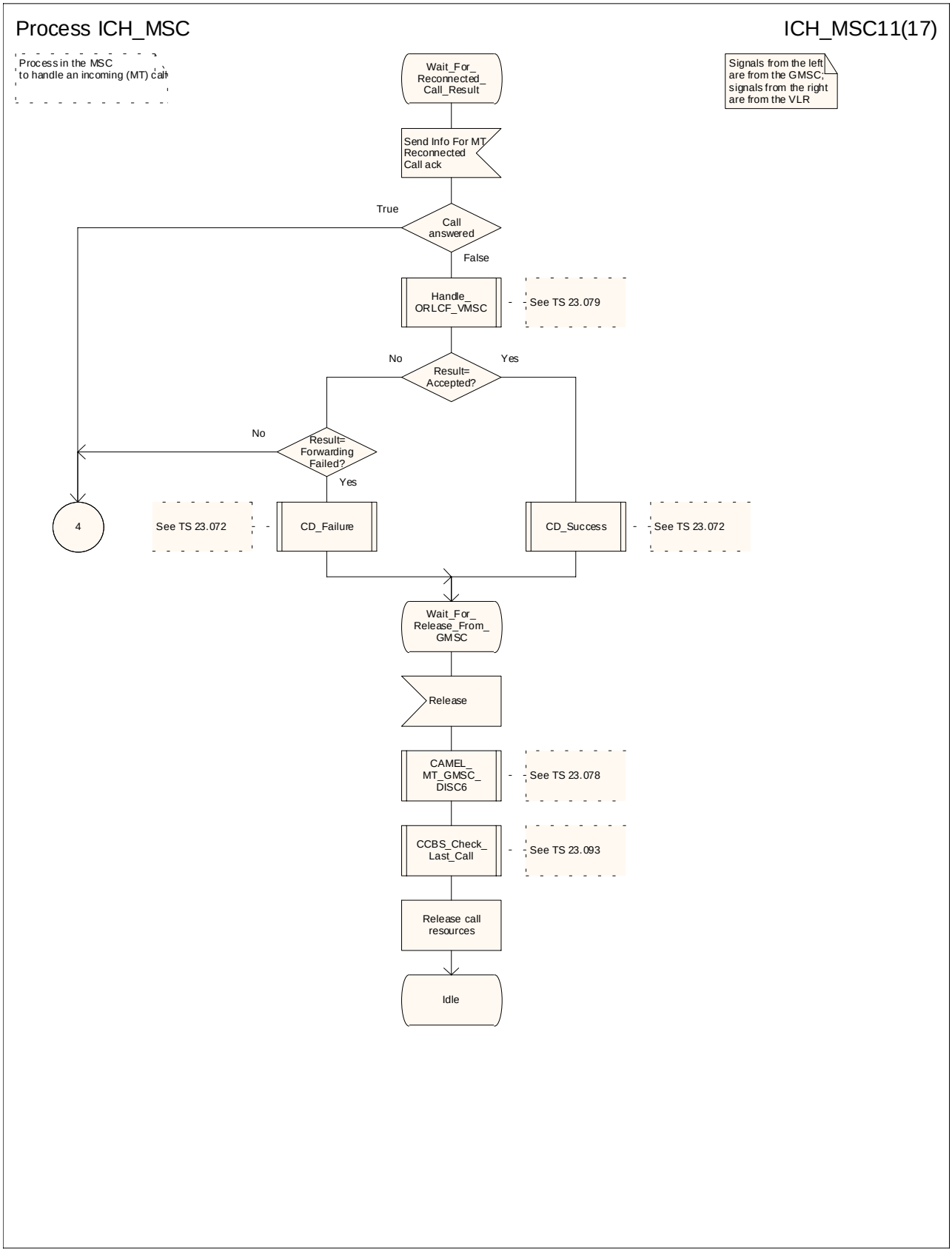


Figure 67k: Process ICH_MSC (sheet 11)

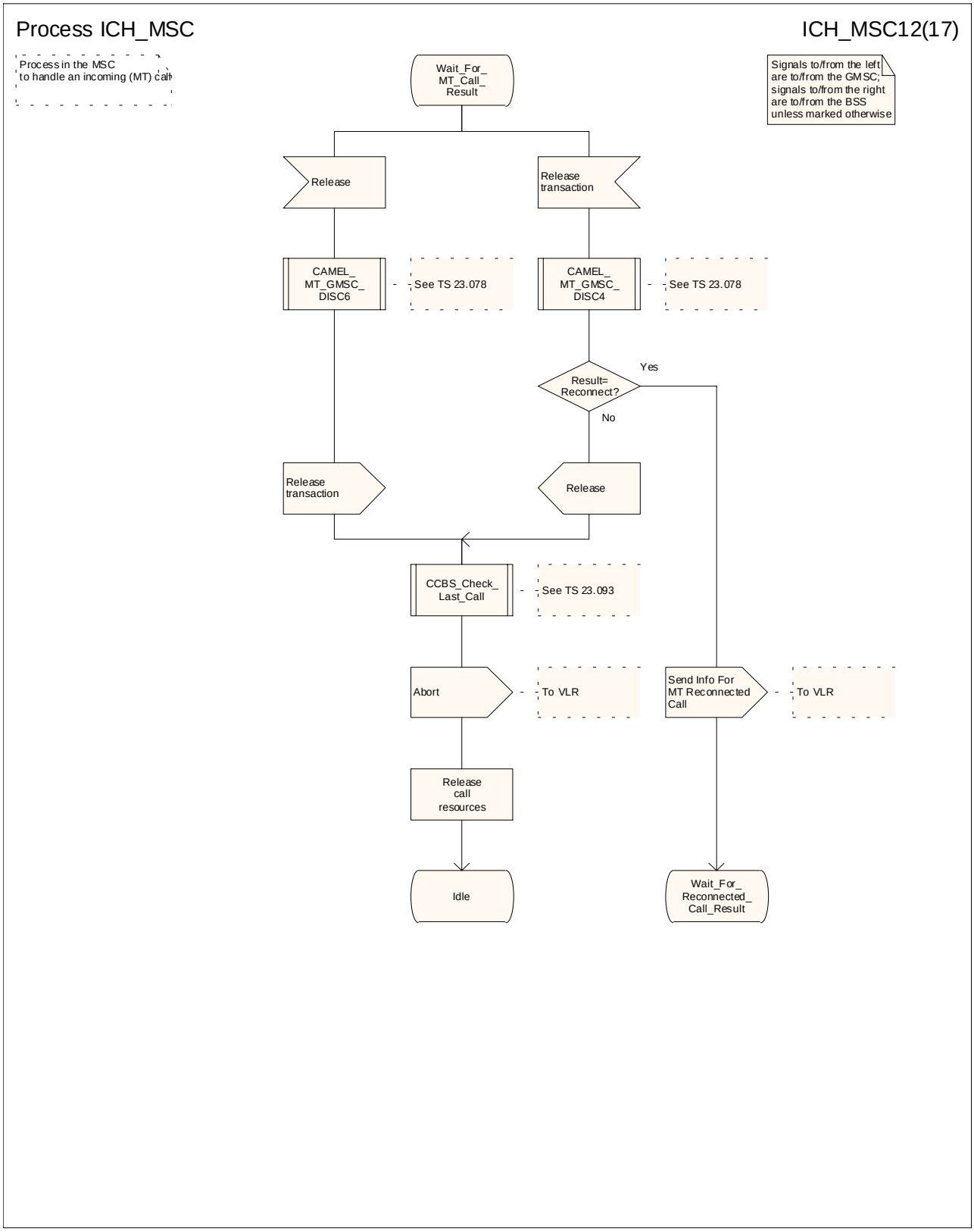


Figure 67l: Process ICH_MSC (sheet 12)

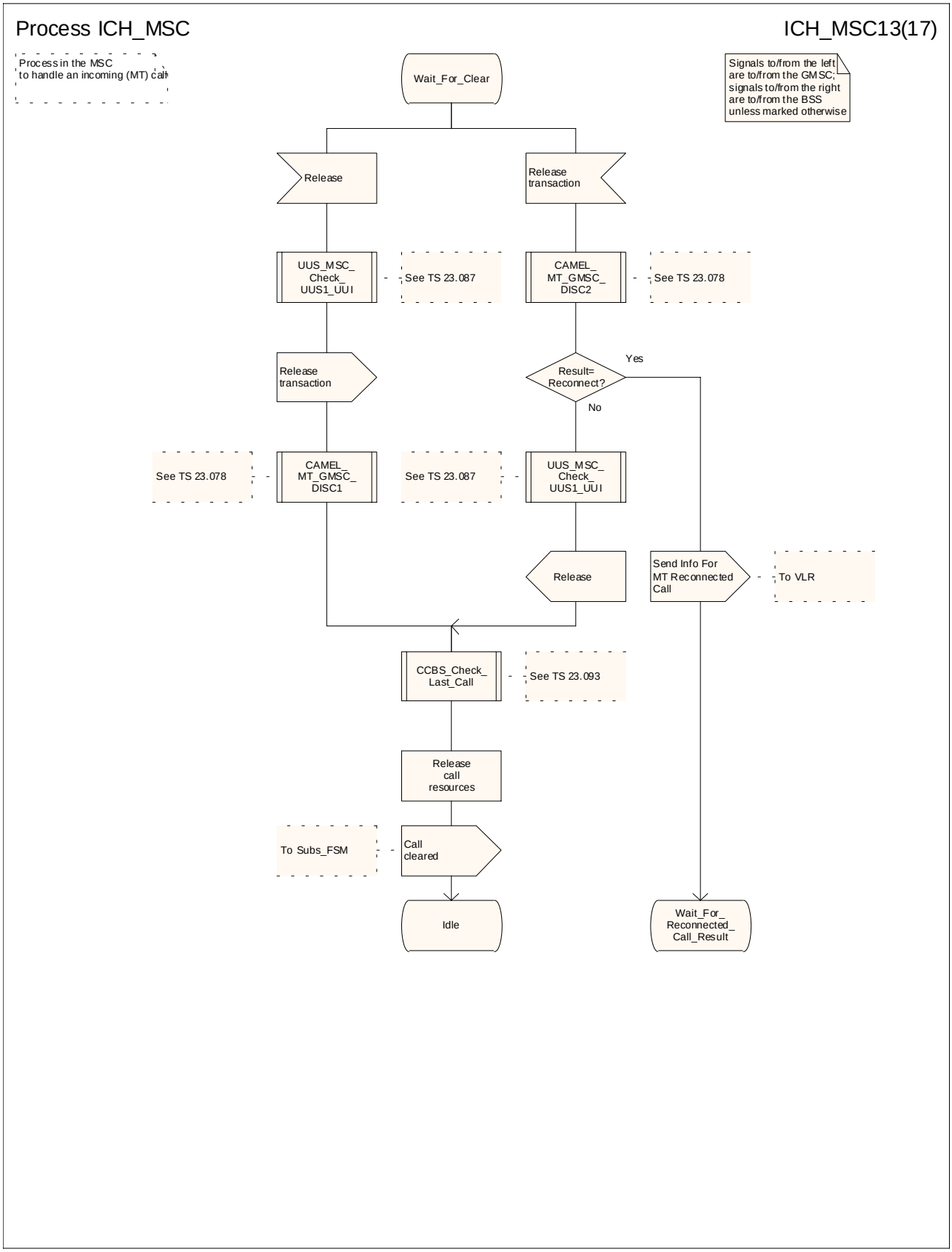


Figure 67m: Process ICH_MSC (sheet 13)

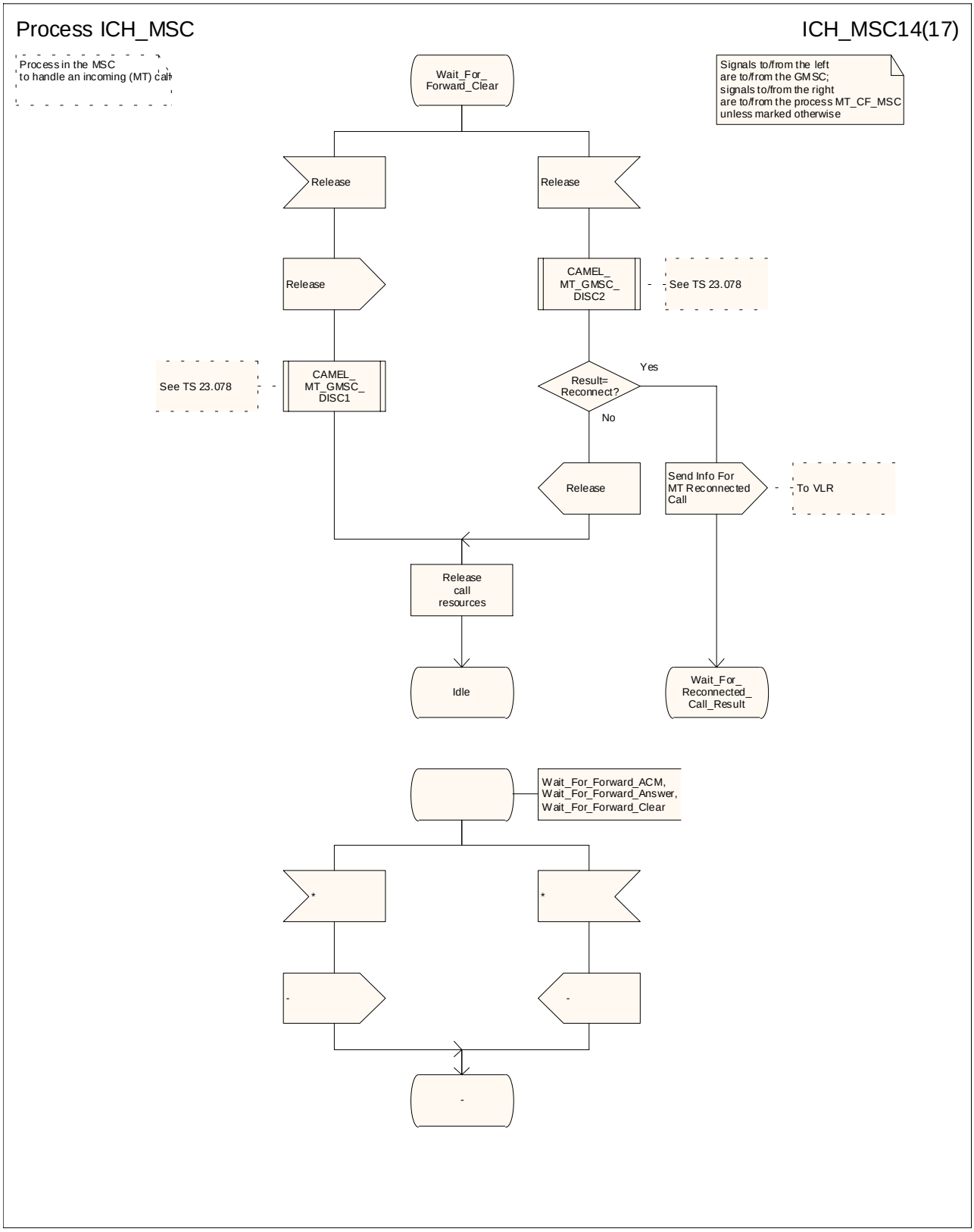


Figure 67n: Process ICH_MSC (sheet 14)

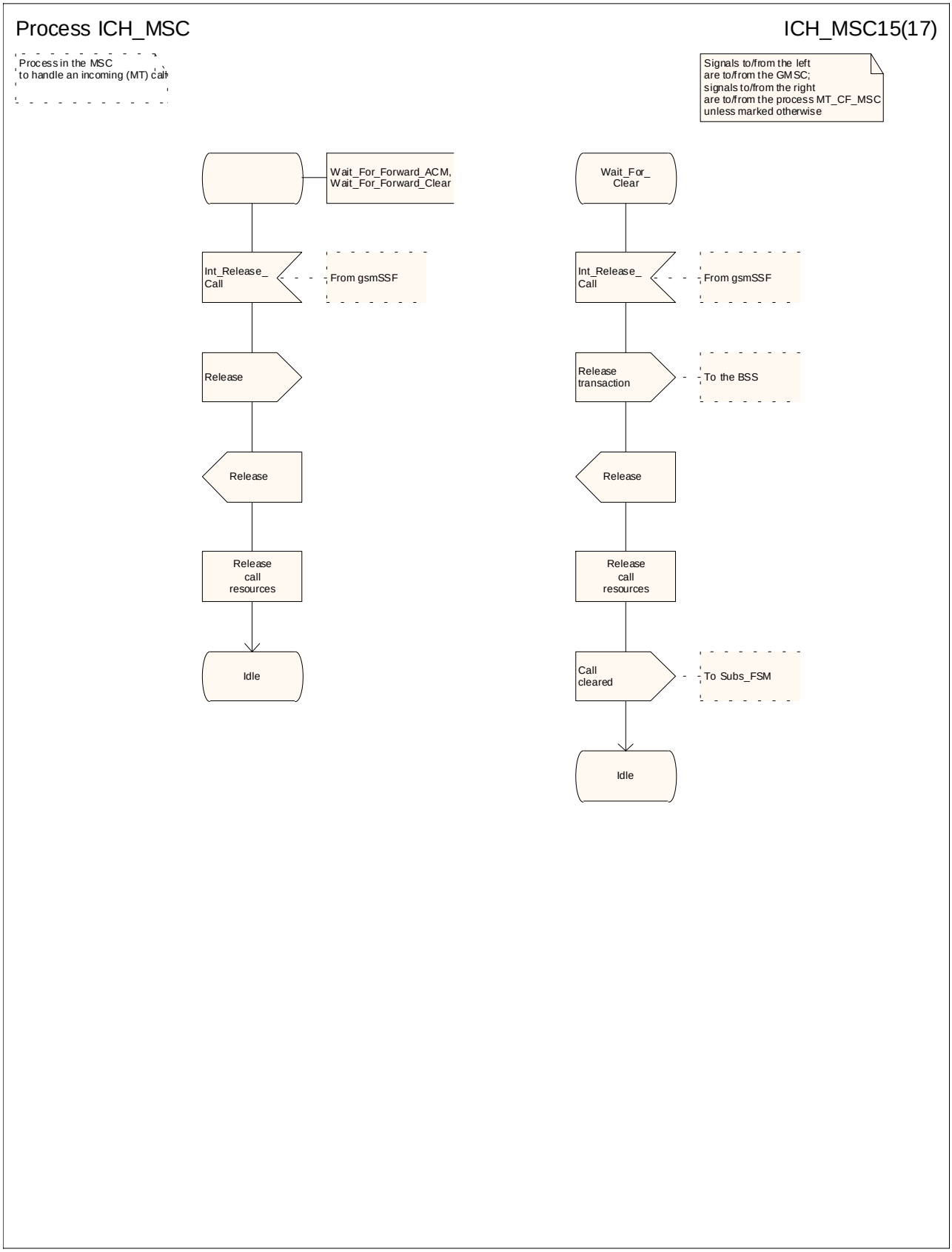


Figure 67o: Process ICH_MSC (sheet 15)

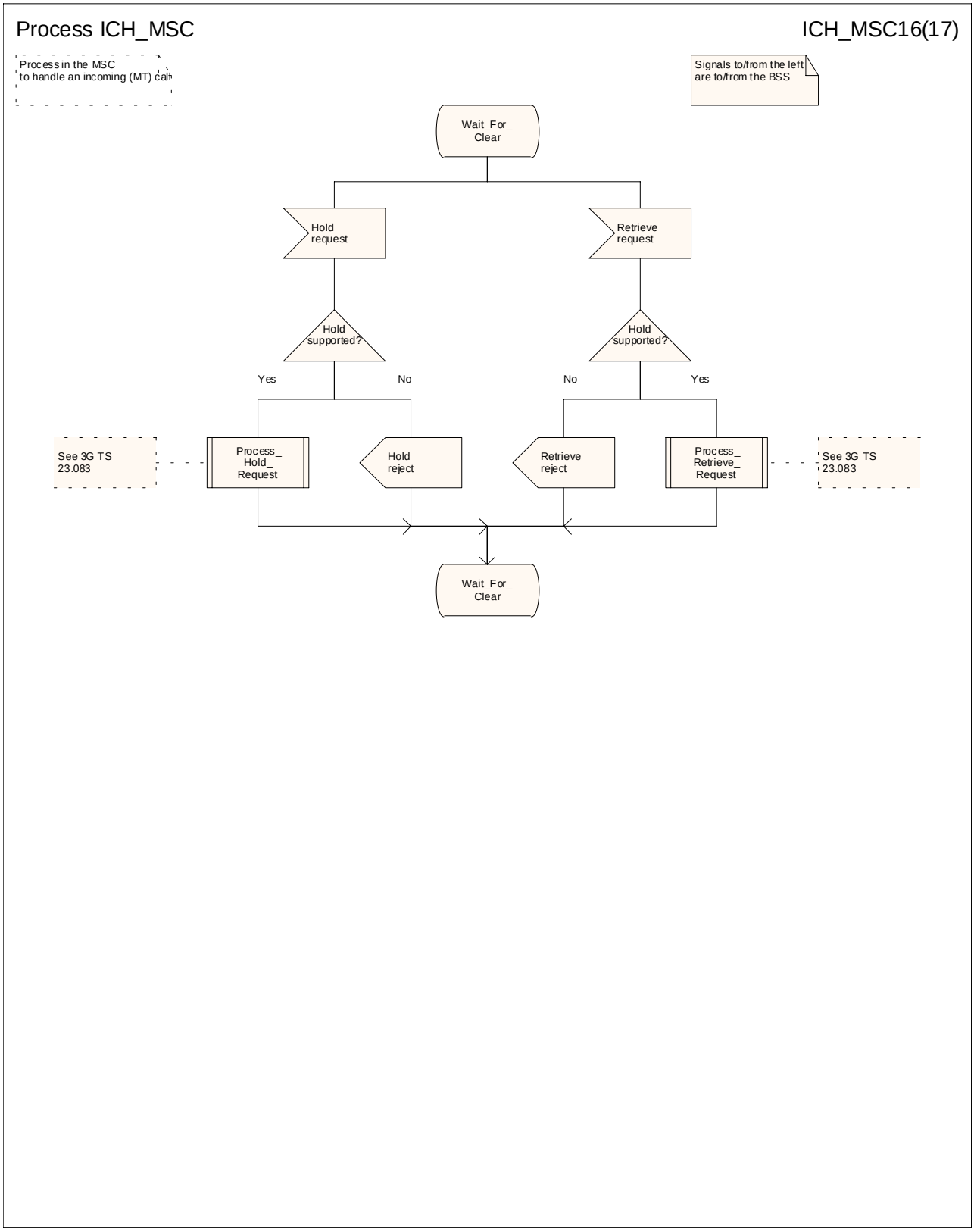


Figure 67p: Process ICH_MSC (sheet 16)

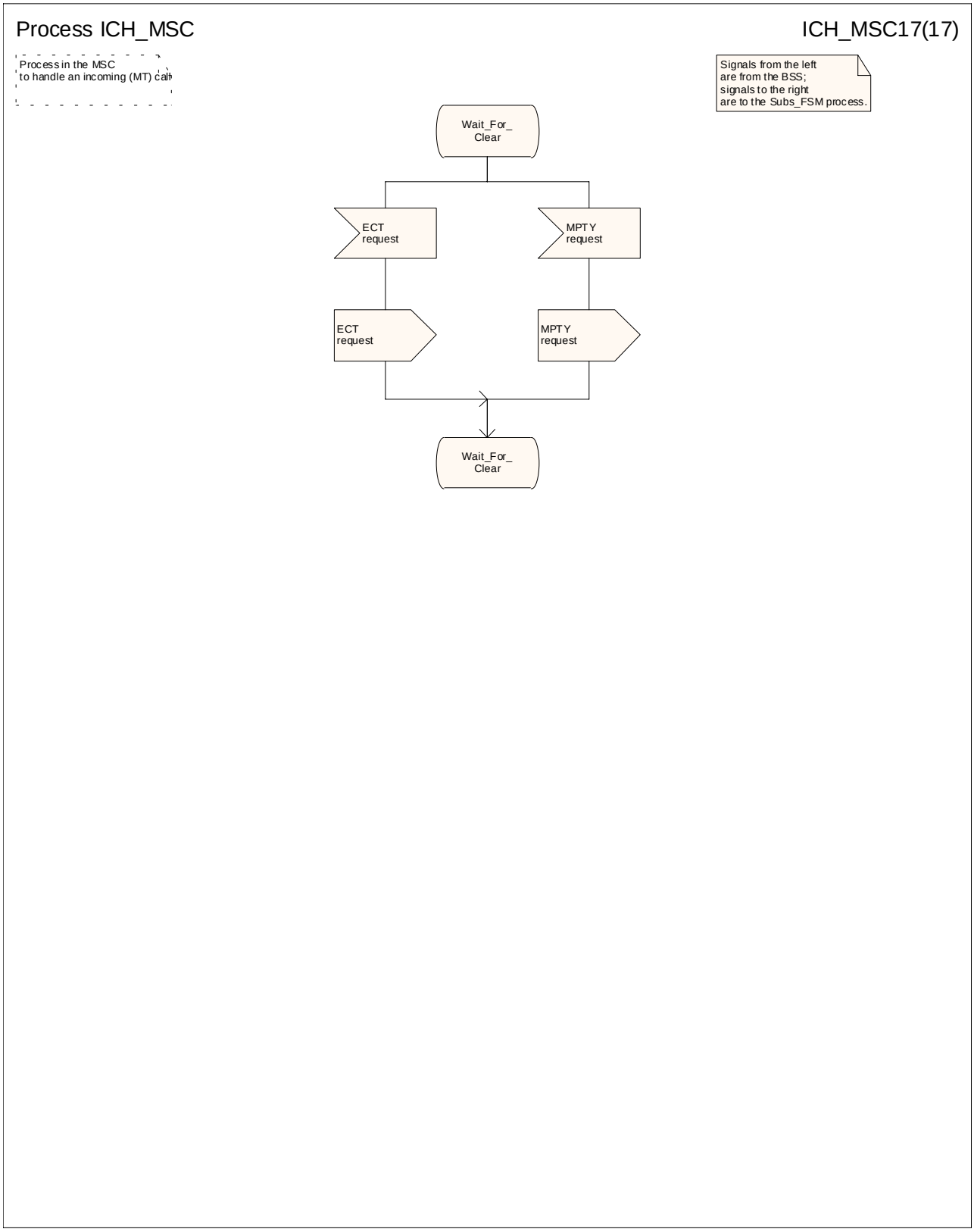


Figure 67q: Process ICH_MSC (sheet 17)

3GPP TSG CN WG4 Meeting #15 ***N4-020808***
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **23.083** **CR** **010** ⌘ **rev** **-** ⌘ Current version: **5.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☒

Title:	⌘	Determining the basic service for MT calls
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	27/06/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)

Reason for change:	⌘	CR 29.007-048r1, approved at CN #16, changed the rules for determining the basic service which applies for an MT call. The procedure Derive_CS_BC_MSC is no longer aligned with the rules defined in TS 29.007.
Summary of change:	⌘	Replace call of procedure to determine the PLMN BC for an MT call with reference to TS 29.007
Consequences if not approved:	⌘	Misaligned specifications

Clauses affected:	⌘	0.1; 1.2; Figure 1.5 (sheet 1 of 9)

	Y	N			
Other specs affected:	X		Other core specifications	⌘	CR 29.007-048r1 (approved in CN #16) CR 23.018-109
		X	Test specifications		
		X	O&M Specifications		

Other comments:	⌘	

****** First modified section ******

0.1 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document.*

[1] 3GPP TR 21.905: "3G Vocabulary".

[2] 3GPP TS 22.082: "Call Forwarding (CF) Supplementary Services - Stage 1".

[3] 3GPP TS 23.011: "Technical realization of supplementary services - General Aspects".

[4] 3GPP TS 24.008: "Mobile radio interface layer 3 specification; Core Network Protocols - Stage3".

[5] 3GPP TS 23.018: “Basic call handling”.

[6] 3GPP TS 23.078: "Customized Applications for Mobile network Enhanced Logic (CAMEL) - Phase 3; Stage 2".

[7] 3GPP TS 23.135: "Multicall supplementary service; Technical Realisation; Stage 2".

[8] 3GPP TS 23.087: "User-to-User Signalling (UUS) - Stage 2".

[9] 3GPP TS 23.093: "Technical realisation of Completion of Calls to Busy Subscriber (CCBS) - Stage 2".

[10] 3GPP TS 23.072: "Call Deflection (CD) supplementary service - Stage2".

[11] 3GPP TS 29.007: "General requirements on interworking between the Public Land Mobile Network (PLMN) and the Integrated Services Digital Network (ISDN) or Public Switched Telephone Network (PSTN)".

****** Next modified section ******

1.2 Functions and information flows

TS 24.008 specifies the procedures for call control. These shall also be used for waiting calls when applicable.

The following Mobile Additional Function has been identified for the call waiting service:

MAF013

Call waiting related authorizations examination

The ability of a PLMN component to determine the authorizations relating to call waiting. See figure 1.4.

Location: VLR

Definitions:

Subscriber B:

The subscriber who is provided by the network with the call waiting supplementary service. The subscriber B is always a mobile subscriber.

User B:

The user who reacts to call waiting at subscriber B. The user B is always a mobile user.

User C:

The user who has originated a call to subscriber B which causes the call waiting supplementary service to be invoked. The user C may be a mobile user.

User A:

The user who is engaged in a call with user B. The user A may be a mobile user.

Timer T1:

This timer corresponds to T303 + T310 (as defined in TS 24.008).

Timer T2:

Call Waiting Timer. This shall limit the duration of the call in the waiting condition.

Timer T3:

No Reply Condition Timer (see TS 22.082).

CFNRc:

Call Forwarding on Not Reachable (see TS 22.082).

CFNRy:

Call Forwarding on No Reply (see TS 22.082).

CW:

Call Waiting.

The procedure Process_Call _Waiting is shown in figure 1.5.

Sheet 1: the procedure Set_CLIP_Info_MSC is specific to CLIP; it is specified in 3GPP TS 23.018 [5].

Sheet 1: the VMSC derives the PLMN bearer capability required for the call according to the rules defined in 3GPP TS 29.007 [11].

~~Sheet 1: the procedure Derive_CS_BC_MSC is specified in 3GPP TS 23.018 [5].~~

Sheet 1: the VMSC and the MS may negotiate the bearer capability to be used for the call by the exchange of information in the Set-up and Call Confirmed messages.

Sheet 1: the Call Confirmed message indicates "busy" for the successful case.

Sheet 1: the procedure Establish_Terminating_TCH_Multicall1 is specific to Multicall; it is specified in 3GPP TS 23.135 [7]. If the VMSC does not support Multicall, processing continues from the “Yes” exit of the test “Result=Pass?”.

Sheet 1: the procedure UUS_ICH_UUS1_Implicit_Active is specific to UUS; it is specified in 3GPP TS 23.087 [8].

Sheet 1: the procedure CCBS_Report_Not_Idle is specific to CCBS; it is specified in 3GPP TS 23.093 [9].

Sheet 2, sheet 3, sheet 5: the procedure UUS_ICH_Check_Support is specific to UUS; it is specified in 3GPP TS 23.087 [8]. If the VMSC does not support UUS, processing continues from the "Yes" exit of the test "Result=Pass?" where the test follows the procedure call.

Sheet 2: the procedure CCBS_ICH_MSC_Report_Success is specific to CCBS; it is specified in 3GPP TS 23.093 [9].

Sheet 2: the task "UTU2Cnt:=0" is executed only if the VMSC supports UUS.

Sheet 2: the procedure CAMEL_Start_TNRy is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [6].

Sheet 2: the procedure Send_ACM_If_Required is specified in 3GPP TS 23.018 [5].

Sheet 2: The procedure CAMEL_MT_MSC_ALERTING is specific to CAMEL phase 4 or later; it is specifed in 3GPP TS 23.078 [6]. If the VMSC does not support CAMEL phase 4 or later, processing continues from the "Pass" exit of the test "Result?".

Sheet 2, sheet 8: the processing in the branch starting with the input "CD Request" is specific to Call Deflection; if the VMSC does not support Call Deflection the input is discarded.

Sheet 2, sheet 8: the procedure Handling_CD_MSC is specific to Call Deflection; it is specified in 3GPP TS 23.072 [10].

Sheet 2, sheet 3, sheet 6, sheet 7: the procedure CAMEL_MT_GMSC_DISC4 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [6]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "No" exit of the test "Result=Reconnect?".

Sheet 2, sheet 3, sheet 4, sheet 8: the procedure CCBS_ICH_MSC_Report_Failure is specific to CCBS; it is specified in 3GPP TS 23.093 [9].

Sheet 3, sheet 7: the Release transaction (reject) message covers all unsuccessful cases not otherwise indicated.

Sheet 4, sheet 7: the procedure UUS_MSC_Check_UUS1_UUI is specific to UUS; it is specified in 3GPP TS 23.087 [8].

Sheet 4, sheet 8: the procedure CAMEL_MT_GMSC_DISC6 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [6].

Sheet 5: the procedure CAMEL_Stop_TNRy is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [6].

Sheet 5: the procedure Establish_Terminating_TCH_If_Required is specified in 3GPP TS 23.018 [5].

Sheet 5: the procedure Establish_Terminating_TCH_Multicall is specific to Multicall; it is specified in 3GPP TS 23.135 [7].

Sheet 6: the procedure Handle_AoC_MT_MSC is specific to AoC; it is specified in 3GPP TS 23.018 [5]. If the VMSC does not support AoC, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 6: the procedure CAMEL_MT_GMSC_ANSWER is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [6]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "Pass" exit of the test "Result?" on sheet 6.

Sheet 6: the procedure Set_COL_Presentation_Indicator_MSC is specific to COLP; it is specified in 3GPP TS 23.018 [5].

Sheet 6: the procedure Send_Answer_If_Required is specified in 3GPP TS 23.018 [5]

Sheet 7: the input signal "CAMEL TNRy expired" will be received only if the VMSC supports CAMEL phase 3 or later.

Sheet 7: the procedure CAMEL_MT_GMSC_DISC5 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [6]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "No" exit of the test "Result=Reconnect?".

Sheet 7, sheet 8: the procedure UUS_ICH_Check_Forwarding is specific to UUS; it is specified in 3GPP TS 23.087 [8]. If the VMSC does not support UUS, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 7, sheet 8, sheet 9: the procedure UUS_MSC_Check_UUS1_UUI is specificto UUS; it is specified in 3GPP TS 23.087 [8].

Sheet 8: the procedures UUS_MSC_Check_UUS2_UUI_to_MS and UUS_MSC_Check_UUS2_UUI_to_NW are specific to UUS; they are specified in 3GPP TS 23.087 [8].

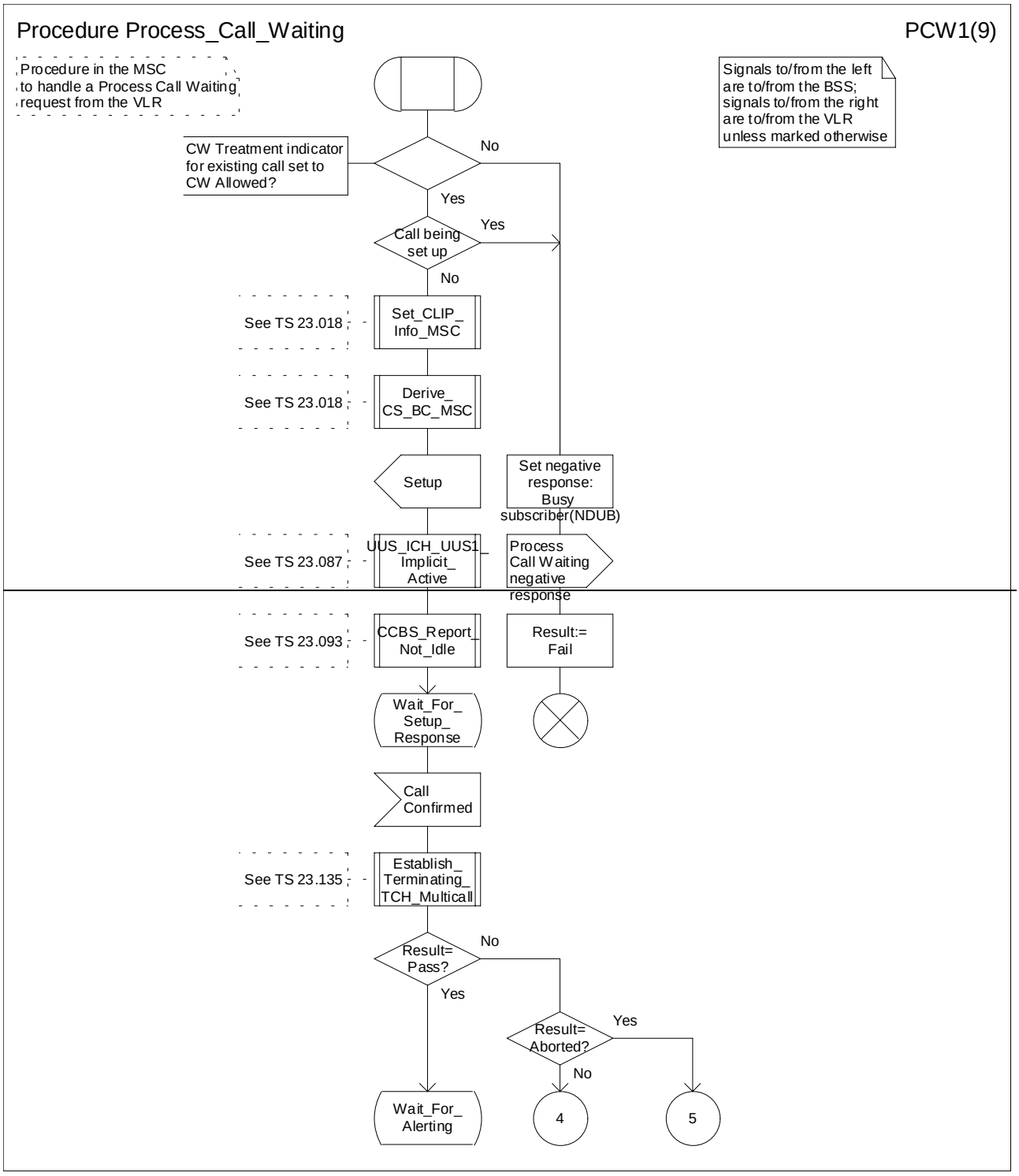
Sheet 8: the procedure CD_UUS_Interaction is specific to Call Deflection; it is specified in GSM 23.072 [10].

Sheet 9: the procedure CCBS_ICH_MSC_Report_Failure is specific to CCBS; it is specified in 23.093 [9].

Sheet 9: the procedure CAMEL_MT_GMSC_DISC6 is specific to CAMEL; it is specified in 23.078 [6].

The information flows are shown in figure 1.6. In these flows it is assumed that user A and user C are fixed users and that user B is a mobile user. Functions to be performed by the fixed ISDN are not shown in the information flows. Only the functions to be performed by the PLMN are shown.

****** Next modified section ******


Procedure Process_Call_Waiting
PCW1(9)
Procedure in the MSC to handle a Process Call Waiting request from the VLR
Signals to/from the left are to/from the BSS; signals to/from the right are to/from the VLR unless marked otherwise
CW Treatment indicator for existing call set to CW Allowed?
No
Yes
Call being set up
Yes
No
See TS 23.018
Set_CLIP_Info_MSC
See TS 23.018
Derive_CS_BC_MSC
Setup
Set negative response: Busy subscriber(NDUB)
See TS 23.087
UUS_ICH_UUS1_Implicit_Active
Process Call Waiting negative response
See TS 23.093
CCBS_Report_Not_Idle
Result:= Fail
Wait_For_Setup_Response
Call Confirmed
See TS 23.135
Establish_Terminating_TCH_Multicall
Result= Pass?
No
Yes
Result= Aborted?
Yes
No
Wait_For_Alerting
4
5

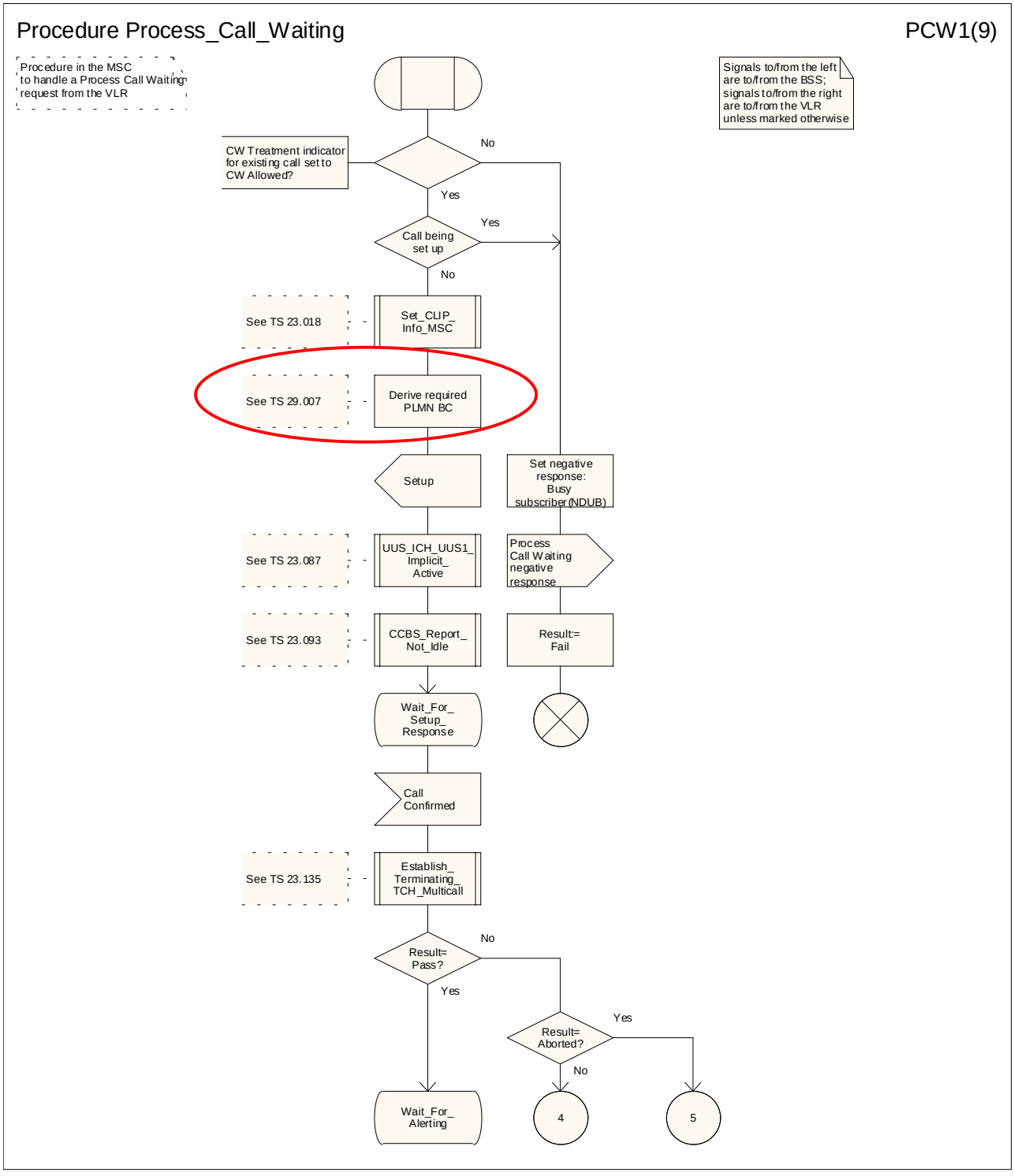


Figure 1.5 (sheet 1 of 9): Procedure Process_Call_Waiting

****** End of document ******

3GPP TSG CN WG4 Meeting #15 ***N4-021000***
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **24.080** **CR** **019** ⌘ **rev** **3** ⌘ Current version: **5.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Compatible upgrade to ASN.1:1997 of 24.080
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	29/07/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	The current version of the ASN.1 standard used in 24.080 is the 1988 edition. This old version will be withdrawn this Summer from the ITU-T and ISO lists of standards.
Summary of change:	⌘	The syntax of the ASN.1 modules has been updated to conform to the 1997 edition of the standard. The semantics of the specification has not been changed. The binary (BER) encoding is backwards compatible (i.e., an old implementation of 24.080 will be able to interoperate with an application that implements this CR). The references to the ASN.1 standards are updated. In each OID, the "ccitt" arc has been replaces by its synonymous "itu-t" according to ITU-T Rec. X.660 Amd. 2.
Consequences if not approved:	⌘	This 3GPP specification will reference standards (namely, X.208/X.209) that users won't be able to buy after Summer 2002 (note that the 1997 edition of ASN.1 is already freely available). Moreover the proposed changes give a more formal specification, hence more reliable implementations. Relying to X.208 instead of X.680 could also prevent the update towards new features offered by the latest edition of the ASN.1 standard (new character string types, extensibility...).

Clauses affected:	⌘	1.1, 3.6.1, 4

Other specs affected: ⌘	Y	N		⌘
		X	Other core specifications	
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘	

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

****** Start of modifications ******

1.1 References

…..

[11] ITU-T Recommendation X.680: "Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation".

[11b] ITU-T Recommendation X.681: "Information technology – Abstract Syntax Notation One (ASN.1): Information object specification".

[12] ITU-T Recommendation X.690: "Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)".~~ITU-T Recommendation X.208: "Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1)".~~

~~[12] ITU-T Recommendation X.209: "Specification of Abstract Syntax Notation One (ASN.1)".~~

[13] ITU-T Recommendation X.880: "Data networks and open system communication - Open System Interconnection - Service definitions - Remote operations: Concepts, model and notation".

~~ITU-T Recommendation Q.773: "Transaction capabilities formats and encoding".~~

****** Next modification ******

3.6.1 Component (octet 3 etc.)

This subclause provides the formats and encoding of components in the Facility information element. Formats and encoding methods make use of and is a subset of ~~CCITT~~ITU-T Rec.~~ommendation~~ Q.773 (Transaction Capabilities formats and Encoding) and T/S 43/BB. The used part of ~~CCITT~~ITU-T Rec.~~ommendation~~ Q.773 respectively T/S 43/BB is almost the same as the Component Portion of TC messages. The only difference is that returnResultNotLast is not used.

This subclause is further based on:

- ~~CCITT~~ITU-T Rec.~~ommendation~~ X.680~~208~~ (Abstract Syntax Notation One (ASN.1): Specification of Basic Notation~~Specification of Abstract Syntax Notation One (ASN.1)~~);
- ~~CCITT~~ITU-T Rec.~~ommendation~~ X.690~~209~~ (Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)~~Specification of basic encoding rules for Abstract Syntax Notation One~~);

and is consistent with these ~~CCITT~~ITU-T R~~r~~ecommendations.

~~The CCITT Recommendations X.208 and X.209 formal description language is not used.~~

The parameters in tables 3.3 to 3.6 may be one of the following:

- a Sequence of Parameters;
- a Set of Parameters;
- a specific Parameter with its own tag (i.e. not part of a Sequence or Set);
- nothing at all (i.e. absent).

NOTE: Concerning the general rules for encoding (structure of encoding, identifier octets, length octets, etc.) see ~~CCITT~~ITU-T Rec~~.ommendations~~ X.690~~208 and X.209~~. For these general rules the same exceptions apply as stated in TS 29.002. This holds also for tables 3.3 to 3.6.

Table 3.3: Invoke component

Invoke component	Reference	Mandatory indication
Component type tag Component length	3.6.2 X.690~~209~~	M
Invoke ID tag Invoke ID length Invoke ID	3.6.3 X.690~~209~~ 3.6.3	M
Linked ID tag Linked ID length Linked ID	3.6.3 X.690~~209~~ 3.6.3	O
Operation Code tag Operation Code length Operation Code	3.6.4 X.690~~209~~ 3.6.4	M
Parameters	4	O

Table 3.4: Return Result component

Return Result component	Reference	Mandatory indication
Component type tag Component length	3.6.2 X.690~~209~~	M
Invoke ID tag Invoke ID length Invoke ID	3.6.3 X.690~~209~~ 3.6.3	M
Sequence tag Sequence length	3.6.5 X.690~~209~~	O (note)
Operation Code tag Operation Code length Operation Code	3.6.4 X.690~~209~~ 3.6.4	O (note)
Parameters	4	O (note)
NOTE: Omitted if the Return Result component does not include any parameters.		

Table 3.5: Return Error component

Return Error component	Reference	Mandatory indication
Component type tag Component length	3.6.2 X.690~~209~~	M
Invoke ID tag Invoke ID length Invoke ID	3.6.3 X.690~~209~~ 3.6.3	M
Error Code tag Error Code length Error Code	3.6.6 X.690~~209~~ 3.6.6	M
Parameters	4	O

Table 3.6: Reject component

Reject component	Reference	Mandatory indication
Component type tag Component length	3.6.2 X.690~~209~~	M
Invoke ID tag (note) Invoke ID length Invoke ID	3.6.3 X.690~~209~~ 3.6.3	M
Problem Code tag Problem Code length Problem Code	3.6.7 X.690~~209~~ 3.6.7	M
NOTE: If the Invoke ID is not available, Universal Null (table 3.9) with length = 0 shall be used.		

****** Next modification ******

4 Supplementary services operation specifications

4.1 General

This clause specifies the abstract syntax for the Supplementary Service protocol using the Abstract Syntax Notation One (ASN.1), defined in ~~CCITT~~ITU-T Rec.~~ommendation~~ X.680 and X.681 ~~X.208 (1998)~~.

The mapping of OPERATION and ERROR to components is defined in clause 3 of this specification.

The encoding rules which are applicable to the defined abstract syntax are the Basic Encoding Rules for Abstract Syntax Notation One, defined in ~~CCITT~~ITU-T Rec.~~ommendation~~ X.690~~209 (1998)~~ with the same exceptions as stated in TS 29.002. For each Supplementary Service parameter which has to be transferred by a Supplementary Service message, there is a PDU field (an ASN.1 NamedType) whose ASN.1 identifier has the same name as the corresponding parameter, except for the differences required by the ASN.1 notation (blanks between words are removed, the first letter of the first word is lower-case and the first letter of the following words are capitalized (e.g. "bearer service" is mapped to "bearerService"). In addition some words may be abbreviated as follows:

- ms mobile subscriber;
- ss supplementary services;
- cug closed user group.

The ASN.1 data type which follows the ~~key~~words "ARGUMENT", "PARAMETER" or "RESULT" (for information objects of class "OPERATION" and "ERROR") is always optional from a syntactic point of view. However, except specific mention, it has to be considered as mandatory from a semantic point of view. When in an invoke component, a mandatory element is missing in any component or inner data structure, a reject component is returned with the problem code "Mistyped Parameter". When an optional element is missing in an invoke component or in an inner data structure while it is required by the context, an error component is returned; the associated type of error is "DataMissing".

In case an element is defined as mandatory in the protocol description (TS 24.080 including imports from TS 29.002), but is not present according to the service description (stage 1 to stage 3), the ASN.1 protocol description takes precedence over the diagrams in the TS 24.08x and 24.09x-series of technical specifications.

When possible operations and errors are imported from TS 29.002 thereby making the MSC transparent to most of the messages sent to or from the MS.

Timer values for operations which require timers are shown as ASN.1 comments.

~~Ellipsis Notation~~The extension marker "…" shall be used in the same way as described in TS 29.002 and shall be supported on the radio interface by the MS and the network for all operations defined in this specification including those imported from TS 29.002.

4.2 Operation types

Table 4.1 summarizes the operations defined for supplementary services in this specification and shows which of these operations are call related and call independent. The terms "call related" and "call independent" are defined in TS 24.010.

Table 4.1: Relevance of supplementary service operations

Operation name	Call related SS	Call independent SS
rRegisterSS	-	+
eEraseSS	-	+
aActivateSS	-	+
dDeactivateSS	-	+
iInterrogateSS	-	+
rRegisterPassword	-	+
gGetPassword	-	+
pProcessUnstructuredSS-Data	+	+
fForwardCheckSS-Indication	-	+
pProcessUnstructuredSS-Request	-	+
uUnstructuredSS-Request	-	+
uUnstructuredSS-Notify	-	+
fForwardChargeAdvice	+	-
nNotifySS	+	-
fForwardCUG-Info	+	-
bBuildMPTY	+	-
hHoldMPTY	+	-
rRetrieveMPTY	+	-
sSplitMPTY	+	-
eExplicitCT	+	-
aAccessRegisterCCEntry	+	-
eEraseCCEntry	-	+
cCallDeflection	+	-
uUserUserService	+	-
lcsLCS-LocationNotification	-	+
lcsLCS-MOLR	-	+
NOTE: The pProcessUnstructuredSS-Data operation may be used call related by a GSM Phase 1 MS.		

The following ASN.1 module defines operations by allocating them a local value. For the involved operations the same local values as in MAP are allocated.

```
SS-Operations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-Operations (0) version8 (8)}

DEFINITIONS ::=

BEGIN

EXPORTS

-- exports operation types

-- operations defined in this specification
pProcessUnstructuredSS-Data, nNotifySS, fForwardChargeAdvice, fForwardCUG-Info, bBuildMPTY,
hHoldMPTY, rRetrieveMPTY, sSplitMPTY, eExplicitCT, aAccessRegisterCCEntry, cCallDeflection,
uUserUserService,
lcsLCS-LocationNotification, lcsLCS-MOLR;

IMPORTS

OPERATION FROM
Remote-Operations-Information-Objects {
    joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

-- The MAP operations:
-- rRegisterSS, eEraseSS, aActivateSS, dDeactivateSS, iInterrogateSS, rRegisterPassword,
-- gGetPassword, pProcessUnstructuredSS-Request, uUnstructuredSS-Request, uUnstructuredSS-Notify
-- fForwardCheckSS-Indication
```

```
-- are imported from MAP-Operations in SS-Protocol module.

-- imports SS-data types
NotifySS-Arg,
ForwardChargeAdviceArg,
ForwardCUG-InfoArg,
SS-UserData,
AccessRegisterCCEntryArg,
CallDeflectionArg,
UserUserServiceArg,
LocationNotificationArg,
LocationNotificationRes,
LCS-MOLRArg,
LCS-MOLRRes
FROM SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-DataTypes (2) version8 (8)}

-- imports MAP-SS-data types
RegisterCC-EntryRes
FROM MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

-- imports MAP-errors
iIllegalSS-Operation, ssSS-ErrorStatus, ssSS-NotAvailable, ssSS-SubscriptionViolation,
ssSS-Incompatibility, sSystemFailure, fFacilityNotSupported, cCallBarred, uUnexpectedDataValue,
sShortTermDenial, lLongTermDenial, dDataMissing, fForwardingViolation, fForwardingFailed,
pPositionMethodFailure
FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-Errors (10) version8 (8)}

-- imports SS-Errors
rResourcesNotAvailable, mMaxNumberOfMPTY-ParticipantsExceeded, dDeflectionToServedSubscriber,
iInvalidDeflectedToNumber, sSpecialServiceCode, rRejectedByUser, rRejectedByNetwork
FROM SS-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-Errors (1) version8 (8)}
;

-- operation types definition

pProcessUnstructuredSS-Data::=  OPERATION ::= { -- Timer T(PUSSD)= 15s to 30s
    ARGUMENT
    ss-UserData     SS-UserData
    RESULT
    ss-UserData     SS-UserData
                    -- optional
    ERRORS          {
                    sSystemFailure |
                    uUnexpectedDataValue}
    CODE            local:19 }

nNotifySS ::=   OPERATION ::= {
    ARGUMENT
    notifySS-Arg    NotifySS-Arg
    CODE            local:16 }

fForwardChargeAdvice ::=    OPERATION ::= { -- Timer T(AoC)= 1s to 40s
    ARGUMENT
    forwardChargeAdviceArg  ForwardChargeAdviceArg
    RETURN RESULT TRUE
    CODE            local:125 }

fForwardCUG-Info ::=    OPERATION ::= {
    ARGUMENT
    forwardCUG-InfoArg  ForwardCUG-InfoArg
    CODE            local:120 }

bBuildMPTY ::=  OPERATION ::= { -- Timer T(BuildMPTY)= 5s to 30s
    RETURN RESULT TRUE
    ERRORS  {
            iIllegalSS-Operation |
            ssSS-ErrorStatus |
            ssSS-NotAvailable |
            ssSS-Incompatibility |
            sSystemFailure |
```

```
        rResourcesNotAvailable |-
        mMaxNumberOfMPTY-ParticipantsExceeded}
    CODE    local:124 }

holdMPTY ::=   OPERATION ::= { -- Timer T(HoldMPTY)= 5s to 30s
    RETURN RESULT TRUE
    ERRORS  {
        iIllegalSS-Operation |-
        ssSS-ErrorStatus |-
        ssSS-Incompatibility |-
        fFacilityNotSupported |-
        sSystemFailure}
    CODE    local:123 }

rRetrieveMPTY ::=   OPERATION ::= { -- Timer T(RetrieveMPTY)= 5s to 30s
    RETURN RESULT TRUE
    ERRORS  {
        iIllegalSS-Operation |-
        ssSS-ErrorStatus |-
        ssSS-Incompatibility |-
        fFacilityNotSupported |-
        sSystemFailure}
    CODE    local:122 }

sSplitMPTY ::=  OPERATION ::= { -- Timer T(SplitMPTY)= 5s to 30s
    RETURN RESULT TRUE
    ERRORS  {
        iIllegalSS-Operation |-
        ssSS-ErrorStatus |-
        ssSS-Incompatibility |-
        fFacilityNotSupported |-
        sSystemFailure}
    CODE    local:121 }

eExplicitCT ::= OPERATION ::= { -- Timer T(ECT)= 5s to 15s
    RETURN RESULT TRUE
    ERRORS  {
        iIllegalSS-Operation |-
        ssSS-ErrorStatus |-
        ssSS-NotAvailable |-
        ssSS-Incompatibility |-
        fFacilityNotSupported |-
        sSystemFailure |-
        rResourcesNotAvailable |-
        cCallBarred}
    CODE    local:126 }

aAccessRegisterCCEntry  ::= OPERATION ::= { -- Timer T(AccRegCCEntry)= 30s
    ARGUMENT
    accessRegisterCCEntryArg    AccessRegisterCCEntryArg
    RESULT
    registerCCEntryRes    RegisterCC-EntryRes
    ERRORS      {
        sSystemFailure |-
        dDataMissing |-
        uUnexpectedDataValue |-
        cCallBarred |-
        iIllegalSS-Operation |-
        ssSS-ErrorStatus |-
        ssSS-Incompatibility |-
        sShortTermDenial |-
        lLongTermDenial |-
        fFacilityNotSupported}
    CODE        local:119 }

-- the timer value is defined by T308, see also in TS 24.008 for definition of timer T308

cCallDeflection ::= OPERATION ::= { -- Timer T(CD)= 30s
    ARGUMENT
    callDeflectionArg   CallDeflectionArg
    RETURN RESULT TRUE
    ERRORS      {
        iIllegalSS-Operation- |
        ssSS-ErrorStatus- |
        ssSS-NotAvailable- |
        ssSS-Incompatibility- |
        fFacilityNotSupported- |
        sSystemFailure- |
```

```
            rResourcesNotAvailable,
            fForwardingViolation,
            cCallBarred,
            dDeflectionToServedSubscriber,
            iInvalidDeflectedToNumber,
            sSpecialServiceCode,
            fForwardingFailed}
    CODE    local:117 }

-- the timer value is defined by T305, see also in TS 24.008 for definition of timer T305
-- extensionContainer shall not be used with this operation

uUserUserService ::=    OPERATION ::= { -- Timer T(UUS3)= 10s
    ARGUMENT
    userUserServiceArg  UserUserServiceArg
    RETURN RESULT TRUE
    ERRORS      {
            iIllegalSS-Operation,
            ssSS-ErrorStatus,
            ssSS-NotAvailable,
            ssSS-Incompatibility,
            fFacilityNotSupported,
            sSystemFailure,
            rResourcesNotAvailable,
            rRejectedByNetwork,
            rRejectedByUser}
    CODE    local:118 }

-- The timer value for UUS3 is 10s; it is applicable only if UUS3 is activated by FACILITY
-- message. If UUS service (UUS1, UUS2 or UUS3) is activated by SETUP message, no timers are
-- needed. In those cases Return Result or Return Error must be received within certain call
-- control messages, see GSM 04.87.
-- extensionContainer shall not be used with this operation.


lcsLCS-LocationNotification::=  OPERATION ::= { -- Timer T(LCSN)= 10s to 20s
    ARGUMENT
    locationNotificationArg     LocationNotificationArg
    RESULT
    locationNotificationRes     LocationNotificationRes
    ERRORS      {
            sSystemFailure,
            uUnexpectedDataValue}
    CODE    local:116 }


lcsLCS-MOLR::=  OPERATION ::= { -- Timer T(LCSL)= 10s to 30s
    ARGUMENT
    lcs-MOLRArg     LCS-MOLRArg
    RESULT
    lcs-MOLRRes     LCS-MOLRRes
    ERRORS      {
            sSystemFailure,
            uUnexpectedDataValue,
            dDataMissing,
            fFacilityNotSupported,
            ssSS-SubscriptionViolation,
            pPositionMethodFailure}
    CODE    local:115 }

END
```

4.2.1 Void

4.2.2 Operation type description

For each operationoperation type this subclause provides a brief prose description.

4.2.2.1 rRegisterSS (MS --> network)

This operation type is invoked by an MS to register data related to a supplementary service in the network. When no BasicService parameter is provided, the registration applies to all provisioned and applicable basic services.

4.2.2.2 eEraseSS (MS --> network)

This operation ~~type~~ is invoked by an MS to erase data related to a supplementary service in the network. When no BasicService parameter is provided, the erasure applies to all provisioned and applicable basic services.

4.2.2.3 aActivateSS (MS --> network)

This operation ~~type~~ is invoked by an MS to request the network for a supplementary service activation. When no BasicService parameter is provided, the activation applies to all provisioned and applicable basic services.

4.2.2.4 dDeactivateSS (MS --> network)

This operation ~~type~~ is invoked by an MS to request the network for a supplementary service deactivation. When no BasicService parameter is provided, the deactivation applies to all provisioned and applicable basic services.

4.2.2.5 iInterrogateSS (MS --> network)

This operation ~~type~~ is invoked by an MS to request the network for a supplementary service interrogation. When no BasicService parameter is provided, the interrogation applies to all provisioned and applicable basic services.

4.2.2.6 nNotifySS (network --> MS)

This operation ~~type~~ is invoked by the network to forward a supplementary service notification towards a mobile subscriber.

4.2.2.7 rRegisterPassword (MS --> network)

This operation ~~type~~ is invoked by an MS to register a new password related to the management by the subscriber himself of subscription data in the HLR. The operation "Register password" will be successful if the subscriber can provide the old password, the new password and the new password again as results of 3 subsequent operations "Get password".

4.2.2.8 gGetPassword (network --> MS)

This operation ~~type~~ is invoked by the network to request a password from the mobile subscriber. It may be used to allow the registration of a new password or the management of subscription data by the subscriber himself (e.g. modification of call barring activation status).

4.2.2.9 pProcessUnstructuredSS-Data (MS --> network)

This operation ~~type~~ is invoked by an MS to relay unstructured information in order to allow end to end SS operation between the MS and the network following specific rules (e.g. embedding of keypad commands). The operation is used in order to provide backward compatibility (see TS 24.090).

4.2.2.10 pProcessUnstructuredSS-Request (MS --> network)

This operation ~~type~~ is invoked by an MS to start an unstructured supplementary service data application in the network.

4.2.2.11 uUnstructuredSS-Request (network --> MS)

This operation ~~type~~ is invoked by the network to request unstructured information from the MS in order to perform an unstructured supplementary service data application.

4.2.2.12 uUnstructuredSS-Notify (network --> MS)

This operation ~~type~~ is invoked by the network to give an unstructured supplementary service notification to the mobile user.

4.2.2.13 fForwardCheckSSIndication (network --> MS)

This operation ~~type~~ is invoked by the network to indicate to the mobile subscriber that the status of supplementary services may not be correct in the network. The procedures for initiating ForwardCheckSSIndication are specified in TS 29.002.

4.2.2.14 fForwardChargeAdvice (network --> MS)

This operation ~~type~~ is invoked by the network to forward Advice of Charge information to the mobile subscriber.

4.2.2.15 bBuildMPTY (MS --> network)

This operation ~~type~~ is invoked by an MS to request the network to connect calls in a multi party call.

4.2.2.16 hHoldMPTY (MS --> network)

This operation ~~type~~ is invoked by an MS to put the MS-connection to a multi party call (invoked by that MS) on hold.

4.2.2.17 rRetrieveMPTY (MS --> network)

This operation ~~type~~ is invoked by an MS to request retrieval of a multi party call held by that MS.

4.2.2.18 sSplitMPTY (MS --> network)

This operation ~~type~~ is invoked by an MS to request a private communication with one of the remote parties in a multi party call invoked by that MS.

4.2.2.19 fForwardCUG-Info (MS --> network)

This operation ~~type~~ is used by an MS to explicitly invoke a CUG call.

4.2.2.20 eExplicitCT (MS --> Network)

This operation ~~type~~ is invoked by an MS to request the network to connect the two calls of the subscriber.

4.2.2.21 aAccessRegisterCCEntry (MS --> Network)

This operation ~~type~~ is invoked by an MS to activate a CCBS request in the network.

4.2.2.22 cCallDeflection (MS --> Network)

This operation ~~type~~ is invoked by an MS to request the network to deflect the incoming call to a specified destination.

4.2.2.23 uUserUserService (MS --> Network, Network --> MS)

This operation ~~type~~ is invoked by an MS to request the network to allow an MS to send/receive information to/from another subscriber in association with a call.

4.2.2.24 lcsLCS-LocationNotification (network --> MS)

This operation ~~type~~ is invoked by the network to request a verification from the mobile subscriber for the attempted location request or to notify the subscriber about authorized location request.

4.2.2.25 lcsLCS-MOLR (MS --> Network)

This operation ~~type~~ is invoked by an MS to request the network to start location procedure, which is used to provide the MS location estimate, location assistance data or deciphering keys for broadcast assistance data.

4.3 Error ~~type~~s

4.3.1 Error ~~type~~s ASN.1 specification

The following ASN.1 module provides an ASN.1 specification of errors. Errors from MAP are imported in the SS-Protocol module in subclause 4.5. The module defines errors by allocating them a local value. For the involved errors the same local values as in MAP are allocated.

```
SS-Errors {
   ~~ccitt~~itu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-Errors (1) version8 (8)}

DEFINITIONS ::=

BEGIN

IMPORTS

ERROR FROM
Remote-Operations-Information-Objects {joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}~~TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}~~;

-- The MAP errors
-- u~~U~~nknownSubscriber, b~~B~~earerServiceNotProvisioned, t~~T~~eleserviceNotProvisioned,
-- i~~I~~llegalSS-Operation, ss~~SS~~-ErrorStatus, ss~~SS~~-NotAvailable, ss~~SS~~-SubscriptionViolation,
-- ss~~SS~~-Incompatibility, s~~S~~ystemFailure, d~~D~~ataMissing, u~~U~~nexpectedDataValue, f~~F~~acilityNotSupported,
-- pw~~PW~~-RegistrationFailure, n~~N~~egativePW-Check, c~~C~~allBarred, n~~N~~umberOfPW-AttemptsViolation,
-- a~~A~~bsentSubscriber, i~~I~~llegalSubscriber, i~~I~~llegalEquipment, ussd~~USSD~~-Busy, u~~U~~nknownAlphabet,
-- f~~F~~orwardingViolation, f~~F~~orwardingFailed
-- are imported from MAP-Errors in SS-Protocol module.

-- error ~~type~~s definition
r~~R~~esourcesNotAvailable ~~::=~~ ERROR ::= {
    CODE    local:127 }
m~~M~~axNumberOfMPTY-ParticipantsExceeded ~~::=~~ ERROR ::= {
    CODE    local:126 }
i~~I~~nvalidDeflectedToNumber ~~::=~~ ERROR ::= {
    CODE    local:125 }
s~~S~~pecialServiceCode ~~::=~~ ERROR ::= {
    CODE    local:124 }
d~~D~~eflectionToServedSubscriber ~~::=~~ ERROR ::= {
    CODE    local:123 }
r~~R~~ejectedByNetwork ~~::=~~ ERROR ::= {
    CODE    local:122 }
r~~R~~ejectedByUser ~~::=~~ ERROR ::= {
    CODE    local:121 }

END
```

4.3.2 Error ~~type~~s description

For each error ~~type~~ this subclause provides a brief prose description.

4.3.2.1 u~~U~~nknownSubscriber

This error is returned by the network when it is requested to perform an operation concerning an unknown subscriber.

4.3.2.2 b~~B~~earerServiceNotProvisioned

This error is returned by the network when it is requested to perform an operation on a supplementary service and not even a subset of the requested bearer service group has been subscribed to.

4.3.2.3 tTeleServiceNotProvisioned

This error is returned by the network when it is requested to perform an operation on a supplementary service and not even a subset of the requested teleservice group has been subscribed to.

4.3.2.4 iIllegalSS-Operation

This error is returned by the network when it is requested to perform an illegal operation which is defined as not applicable for the relevant supplementary service(s) (e.g. registration request for a service which must be registered by the administration). For the definition of the allowed operations for the individual supplementary services, see TS 24.08x and 24.09x-series of technical specifications.

4.3.2.5 ssSS-ErrorStatus

This error is returned by the network when it is requested to perform an operation which is not compatible with the current status of the relevant supplementary service. The current status may be given as additional information by use of the SS-parameter.

4.3.2.6 ssSS-NotAvailable

This error is returned by the network when it is requested to perform an operation on a supplementary service which is not available in the current location area.

4.3.2.7 ssSS-SubscriptionViolation

This error is returned by the network when it is requested to perform an operation on a supplementary service, transgressing the subscription restrictions. The nature of the restriction or the transgressed options may be sent as parameters.

4.3.2.8 ssSS-Incompatibility

This error is returned by the network when it is requested for a supplementary service operation incompatible with the status of an other supplementary service or with the teleservice or bearer service for which the operation is requested. This error shall only be used if the operation is not compatible for even a subset of the teleservice group or bearer service group specified in the request. The identity and status of the conflicting service may also be indicated. The additional information may contain the SS-code parameter, the Basic Service Group parameter and the SS-status parameter.

4.3.2.9 sSSystemFailure

This error is returned by the network, when it cannot perform an operation because of a failure in the network.

4.3.2.10 dDDataMissing

This error is returned by the network when an optional parameter is missing in an invoke component or an inner data structure, while it is required by the context of the request.

4.3.2.11 uUUnexpectedDataValue

This error is returned by the network when it receives a parameter with an unexpected value, without type violation.

4.3.2.12 pPPasswordRegistrationFailure

This error is returned when a password registration procedure fails because of abnormal subscriber inputs. A more specific diagnostic may be passed as error parameter and indicates situations such as:

- invalid password format;
- new passwords mismatch.

4.3.2.13 nNegativePasswordCheck

This error is returned to indicate the negative result of a password check because the subscriber has not provided the required password or has provided a password which does not match the valid one.

4.3.2.14 fFacilityNotSupported

This error is returned by the network receiving a request about a facility which is not supported in the PLMN.

4.3.2.15 rResourcesNotAvailable

This error is returned by the network to the MS if temporarily there are no resources to support e.g. a multi party call available in the network.

4.3.2.16 mMaxNumberOfMPTY-ParticipantsExceeded

This error is returned by the network to the MS if the request must be rejected because the number of subscribers to join a multi party call would exceed the maximum value.

4.3.2.17 cCallBarred

This error is returned by the network to the MS when call independent subscriber control procedures are barred by the operator. The parameter "operator barring" shall be included.

4.3.2.18 nNumberOfPW-AttemptsViolation

This error is returned by the network to the MS when the maximum number of wrong password attempts is exceeded.

4.3.2.19 aAbsentSubscriber

This error is returned when the subscriber has activated the detach service or the system detects the absence condition. This error is not used on the radio interface but only between network entities.

4.3.2.20 iIllegalSubscriber

This error is returned when illegality of the access has been established by use of authentication procedure. This error is not used on the radio interface but only between network entities.

4.3.2.21 iIllegalEquipment

This error is returned when the IMEI check procedure has shown that the IMEI is blacklisted or not white—listed. This error is not used on the radio interface but only between network entities.

4.3.2.22 ussdUSSD-Busy

This error is returned by the MS to the network when the MS is not able to process the unstructured supplementary service data operation due to an on-going MMI input of the user or an already existing call independent supplementary service transaction.

4.3.2.23 uUnknownAlphabet

This error is returned by the MS or the network when the alphabet/language used for the unstructured supplementary service data operation is not known by the network or the MS.

4.3.2.24 iInvalidDeflectedToNumber

This error is returned if the requested deflected-to number is invalid.

4.3.2.25 sSpecialServiceCode

This error is returned if diversion to a special service code was requested.

4.3.2.26 dDeflectionToServedSubscriber

This error is returned if a diversion to the served subscriber's number was requested.

4.3.2.27 rRejectedByNetwork

This error is returned by the network when the network rejects User-to-User Signalling service request.

4.3.2.28 rRejectedByUser

This error is returned by the remote party when the remote party rejects User-to-User Signalling service request.

4.3.2.29 pPositionMethodFailure

This error is returned by the network when the network is unable to obtain any of the location information requested or none of the information obtained satisfies the requested LCS QoS or if requested LCS assistance data could not be transferred or requested deciphering keys for broadcast assistance data could not be returned.

4.4 Data types and identifiers

4.4.1 General

The data types used in the SS protocol specifications are described in the ASN.1 module provided in subclause 4.4.2, while subclause 4.4.3 provides an overview of the identifiers used in SS ASN.1 specifications.

~~Since size constraints are subject to modifications~~ Nnamed values have been defined in the following module for the upper boundaries of the value ranges associated to several sub-type specifications.

4.4.2 ASN.1 data types

This subclause provides an ASN.1 module defining the abstract data types in operations and errors specification. Only data types which are specific for this specification are defined. All other data types are imported from MAP together with the import of operations and errors.

```
SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-DataTypes (2) version8 (8)}

DEFINITIONS

IMPLICIT TAGS ::=

BEGIN

-- exports all data types defined in this module

IMPORTS

SS-Code
FROM MAP-SS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-SS-Code (15) version8 (8)}

-- imports MAP-SS-DataTypes
SS-Status, USSD-DataCodingScheme, USSD-String, CCBS-Feature
-- USSD-DataCodingScheme, USSD-String were introduced because of CNAP.
FROM MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-SS-DataTypes (14) version8 (8)}

CUG-Index,
```

```
NotificationToMSUser
FROM MAP-MS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-MS-DataTypes (11) version8 (8)}

maxSignalInfoLength,
ISDN-AddressString,
ISDN-SubaddressString,
AlertingPattern,
LCSClientExternalID,
AddressString,
LCSServiceTypeID
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-CommonDataTypes (18) version8 (8)}

LocationType,
LCSClientName,
LCS-QoS,
Horizontal-Accuracy,
ResponseTime,
Ext-GeographicalInformation,
SupportedGADShapes,
Add-GeographicalInformation,
LCSRequestorID,
LCSCodeword
FROM MAP-LCS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-LCS-DataTypes (25) version8 (8)}

;

-- data types definition

SS-UserData ::= IA5String (SIZE (1.. maxSignalInfoLength))

NotifySS-Arg ::= SEQUENCE{
    ss-Code                     [1]     SS-Code OPTIONAL,
    ss-Status                   [4]     SS-Status OPTIONAL,
    ss-Notification             [5]     SS-Notification OPTIONAL,
    callIsWaiting-Indicator     [14]    NULL OPTIONAL,
    callOnHold-Indicator        [15]    CallOnHold-Indicator OPTIONAL,
    mpty-Indicator              [16]    NULL OPTIONAL,
    cug-Index                   [17]    CUG-Index OPTIONAL,
    clirSuppressionRejected     [18]    NULL OPTIONAL,
    ... ,
    ect-Indicator               [19]    ECT-Indicator OPTIONAL,
    nameIndicator               [20]    NameIndicator OPTIONAL,
    ccbs-Feature                [21]    CCBS-Feature OPTIONAL,
    alertingPattern             [22]    AlertingPattern OPTIONAL,
    multicall-Indicator         [23]    Multicall-Indicator OPTIONAL}

--  The nameIndicator is defined because of CNAP.

Multicall-Indicator ::= ENUMERATED {
        nbr-SNexceeded (0),
        nbr-Userexceeded (1)}

ForwardChargeAdviceArg ::= SEQUENCE{
    ss-Code                     [0]     SS-Code,
    chargingInformation         [1]     ChargingInformation,
    ...}

SS-Notification ::= OCTET STRING (SIZE (1))

--    Bit 8 7 6 5 4 00000 (Unused)

--  Bit 3   Call is forwarded indication to A-subscriber
--      (calling subscriber)
--  0   No information content
--  1   Outgoing call has been forwarded to C

--  Bit 2   Call is forwarded indication to B-subscriber
--      (forwarding subscriber)
--  0   No information content
--  1   Incoming call has been forwarded to C

--  Bit 1   Call is forwarded indication to C-subscriber
```

```
--      (forwarded-to subscriber)
--  0   No information content
--  1   Incoming call is a forwarded call

ChargingInformation ::= SEQUENCE{
    e1  [1] E1 OPTIONAL,
    e2  [2] E2 OPTIONAL,
    e3  [3] E3 OPTIONAL,
    e4  [4] E4 OPTIONAL,
    e5  [5] E5 OPTIONAL,
    e6  [6] E6 OPTIONAL,
    e7  [7] E7 OPTIONAL,
    ...}

E1 ::= INTEGER (0..max10TimesUnitsPerTime)
max10TimesUnitsPerTime INTEGER ::= 8191

E2 ::= INTEGER (0..max10TimesTimeInterval)
max10TimesTimeInterval INTEGER ::= 8191

E3 ::= INTEGER (0..max100TimesScalingFactor)
max100TimesScalingFactor INTEGER ::= 8191

E4 ::= INTEGER (0..max10TimesIncrement)
max10TimesIncrement INTEGER ::= 8191

E5 ::= INTEGER (0..max10TimesIncrementPerDataInterval)
max10TimesIncrementPerDataInterval INTEGER ::= 8191

E6 ::= INTEGER (0..maxNumberOfSegmentsPerDataInterval)
maxNumberOfSegmentsPerDataInterval INTEGER ::= 8191

E7 ::= INTEGER (0..max10TimesInitialTime)
max10TimesInitialTime INTEGER ::= 8191

CallOnHold-Indicator    ::= ENUMERATED {
            callRetrieved (0),
            callOnHold (1)}

ForwardCUG-InfoArg ::= SEQUENCE {
    cug-Index           [0] CUG-Index OPTIONAL,
    suppressPrefCUG     [1] NULL OPTIONAL,
    suppressOA          [2] NULL OPTIONAL,
    ...}

ECT-Indicator   ::= SEQUENCE {
    ect-CallState       [0] ECT-CallState,
    rdn [1] RDN OPTIONAL,
    ...}

ECT-CallState   ::= ENUMERATED {
        alerting (0),
        active (1)}

    NameIndicator ::= SEQUENCE {
        callingName     [0] Name OPTIONAL,
        ...}

    Name ::= CHOICE {
        namePresentationAllowed     [0] NameSet,
        presentationRestricted      [1] NULL,
        nameUnavailable             [2] NULL,
        namePresentationRestricted  [3] NameSet}

    NameSet ::= SEQUENCE {
        dataCodingScheme        [0] USSD-DataCodingScheme,
        lengthInCharacters      [1] INTEGER,
        nameString              [2] USSD-String,
        ...}

-- NameIndicator, Name and NameSet are defined because of CNAP.
-- The USSD-DataCodingScheme shall indicate use of the default alphabet through the
-- following encoding:
--    bit  7   6   5   4   3   2   1   0
--       | 0   0   0   0 | 1   1   1   1|

RDN ::= CHOICE {
    presentationAllowedAddress              [0] RemotePartyNumber,
```

```
    presentationRestricted                  [1] NULL,
    numberNotAvailableDueToInterworking     [2] NULL,
    presentationRestrictedAddress           [3] RemotePartyNumber}

RemotePartyNumber   ::= SEQUENCE {
    partyNumber             [0] ISDN-AddressString,
    partyNumberSubaddress   [1] ISDN-SubaddressString OPTIONAL,
    ...}

AccessRegisterCCEntryArg    ::= SEQUENCE {
    ...}

CallDeflectionArg   ::= SEQUENCE {
    deflectedToNumber       [0] AddressString,
    deflectedToSubaddress   [1] ISDN-SubaddressString OPTIONAL,
    ...}

UserUserServiceArg ::= SEQUENCE {
    uUS-Service     [0] UUS-Service,
    uUS-Required    [1] BOOLEAN,
    ... }

UUS-Service ::= ENUMERATED {
    uUS1 (1),
    uUS2 (2),
    uUS3 (3),
    ... }

-- exception handling:
-- In case of UUS-Service with any other value, indicated as "UUS required",
-- but not understood by the MS, the call will be cleared.

LocationNotificationArg ::= SEQUENCE {
    notificationType    [0] NotificationToMSUser,
    locationType        [1] LocationType,
    lcsClientExternalID [2] LCSClientExternalID     OPTIONAL,
    lcsClientName       [3] LCSClientName           OPTIONAL,
    ... ,
    lcsRequestorID      [4] LCSRequestorID          OPTIONAL,
    lcsCodeword         [5] LCSCodeword             OPTIONAL,
    lcsServiceTypeID    [6] LCSServiceTypeID        OPTIONAL }
-- exception handling:
-- At reception of an unrecognised notificationType value the receiver shall reject the
-- operation with a return error cause of unexpected data value.
-- At reception of an unrecognised locationType value the receiver shall reject the
-- operation with a return error cause of unexpected data value.


LocationNotificationRes ::= SEQUENCE {
    verificationResponse        [0] VerificationResponse OPTIONAL,
    ...}

VerificationResponse::= ENUMERATED {
    permissionDenied    (0),
    permissionGranted   (1),
    ... }

-- exception handling:
-- an unrecognized value shall be treated the same as value 0 (permissionDenied)

LCS-MOLRArg ::= SEQUENCE {
    molr-Type           [0] MOLR-Type,
    locationMethod      [1] LocationMethod          OPTIONAL,
    lcs-QoS             [2] LCS-QoS                 OPTIONAL,
    lcsClientExternalID [3] LCSClientExternalID     OPTIONAL,
    mlc-Number          [4] ISDN-AddressString      OPTIONAL,
    gpsAssistanceData   [5] GPSAssistanceData       OPTIONAL,
    ...,
    supportedGADShapes  [6] SupportedGADShapes      OPTIONAL}
-- The parameter locationMethod shall be included if and only if the molr-Type is set to value
-- deCipheringKeys or assistanceData.
-- The parameter gpsAssistanceData shall be included if and only if the molr-Type is set to value
-- assistanceData and locationMethod is set to value assistedGPS.

MOLR-Type::= ENUMERATED {
    locationEstimate            (0),
    assistanceData              (1),
    deCipheringKeys             (2),
```

```
    ... }
-- exception handling:
-- an unrecognized value shall be rejected by the receiver with a return error cause of
-- unexpected data value.

LocationMethod::= ENUMERATED {
    msBasedEOTD         (0),
    msAssistedEOTD      (1),
    assistedGPS         (2),
    ...,
    msBasedOTDOA        (3),
    msAssistedOTDOA     (4)
    }
-- exception handling:
-- When this parameter is received with value msBasedEOTD or msAssistedEOTD and the MS
-- is camped on an UMTS Service Area then the receiver shall reject it
-- with a return error cause of unexpected data value.
-- When this parameter is received with value msBasedOTDOA or msAssistedOTDOA and the MS
-- is camped on a GSM Cell then the receiver shall reject it with a return error cause of
-- unexpected data value.
-- an unrecognized value shall be rejected by the receiver with a return error cause of
-- unexpected data value.

GPSAssistanceData::= OCTET STRING (SIZE (1..38))
-- Octets 1 to 38 are coded in the same way as the octets 3 to 7+2n of Requested GPS Data IE
-- in GSM 09.31.

LCS-MOLRRes::= SEQUENCE {
    locationEstimate        [0] Ext-GeographicalInformation     OPTIONAL,
    decipheringKeys         [1] DecipheringKeys                 OPTIONAL,
    ...,
    add-LocationEstimate    [2] Add-GeographicalInformation         OPTIONAL}
-- Parameters locationEstimate or add-LocationEstimate (one but not both)
-- shall be included if and only if the
-- molr-Type in LocationRequestArg was set to value locationEstimate.
-- Parameter add-LocationEstimate shall not be included if the supportedGADShapes
-- parameter was not received in the LCS-MOLRArg.
-- The locationEstimate and the add-locationEstimate parameters shall not be sent if
-- the supportedGADShapes parameter has been received in LCS-MOLRArg
-- and the shape encoded in locationEstimate or add-LocationEstimate is not marked
-- as supported in supportedGADShapes. In such a case LCS-MOLRArg
-- shall be rejected with error FacilityNotSupported with additional indication
-- shapeOfLocationEstimateNotSupported.
-- Parameter decipheringKeys shall be included if and only if the molr-Type
-- in LocationRequestArg was set to value deCipheringKeys.
--


DecipheringKeys::= OCTET STRING (SIZE (15))
-- Octets in DecipheringKeys are coded in the same way as the octets 3 to 17 of Deciphering Key IE
-- in GSM 09.31. I.e. these octets contain Current Deciphering Key, Next Deciphering Key and
-- Ciphering Key Flag.



END
```

4.4.3 Identifiers definition

The parameters which are described in the following subclauses correspond to the identifiers used in operation and error ~~types~~ descriptions.

4.4.3.1 chargingInformation

The chargingInformation identifier refers to the necessary information for the Advice of Charge supplementary service (see TS 22.024).

4.4.3.2 e1

The e1 identifier refers to 10 times the number of LPLMN units per time interval in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.3 e2

The e2 identifier refers to 10 times the length of the time interval in seconds in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.4 e3

The e3 identifier refers to 100 times the scaling factor to convert from LPLMN units to HPLMN units in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.5 e4

The e4 identifier refers to 10 times the LPLMN increment in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.6 e5

The e5 identifier refers to 10 times the number of LPLMN units incremented per data interval in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.7 e6

The e6 identifier refers to the number of segments per data interval in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.8 e7

The e7 identifier refers to 10 times the length of the initial time interval in seconds in connection with the Advice of Charge supplementary service, see TS 22.024.

4.4.3.9 ss-Code

The ss-Code identifier refers to the code which identify a supplementary service or a group of supplementary services.

4.4.3.10 ss-Notification

The ss-Notification identifier refers to one or several supplementary service notifications which have to be forwarded to a mobile subscriber.

4.4.3.11 ss-Status

The ss-Status identifier refers to the status of a supplementary service.

4.4.3.12 callIsWaiting-Indicator

The callIsWaiting-Indicator identifier refers to the indication given to the mobile station that the call is waiting.

4.4.3.13 callOnhold-Indicator

The callOnHold-Indicator identifier refers to the indication given to the mobile station that the call has been put on hold or has been retrieved.

4.4.3.14 mpty-Indicator

The mpty-Indicator identifier refers to the indication given to the mobile station that the multi party call has been invoked.

4.4.3.15 forwardCUG-InfoArg

The forwardCUG-InfoArg identifier refers to the indication given from the mobile subscriber to the network in connection with explicit invocation of a CUG call.

4.4.3.16 cug-Index

The cug-Index identifier refers to the index of a CUG given in an explicit invocation of a CUG call.

4.4.3.17 suppressPrefCUG

The suppressPrefCUG identifier refers to the mobile subscribers request to the network to prohibit the use of the preferential CUG.

4.4.3.18 suppressOA

The suppressOA identifier refers to the mobile subscribers request to the network to prohibit the use of the subscriber option "OA allowed".

4.4.3.19 clirSuppressionRejected

The clirSuppressionRejected identifier refers to the indication given to the mobile station that the CLIR suppression request has been rejected.

4.4.3.20 ect-Indicator

The ect-Indicator identifier refers to the indication given to the mobile station that the call was transferred.

4.4.3.21 ect-CallState

The ect-CallState identifier refers to the state of the call to the other remote party in which Explicit Call Transfer was invoked.

4.4.3.22 rdn

The Rdn identifier refers to the line identity information of the other remote party.

4.4.3.23 presentationAllowedAddress

The presentationAllowedAddress identifier refers to the line identity of the other remote party that is allowed to be presented.

4.4.3.24 presentationRestricted

The presentationRestricted identifier refers to the restriction of presentation of the line identity of the other remote party.

Also, the identifier refers to the restriction of presentation of the name identity of the calling party to the called party.

4.4.3.25 numberNotAvailableDueToInterworking

The numberNotAvailableDueToInterworking identifier refers to the unavailability of the line identity of the other remote party.

4.4.3.26 presentationRestrictedAddress

The presentationRestrictedAddress identifier refers to the line identity of the other remote party which presentation restriction is overridden.

4.4.3.27 partyNumber

The partyNumber identifier refers to the remote party number.

4.4.3.28 partyNumberSubaddress

The partyNumberSubaddress identifier refers to remote party number subaddress.

4.4.3.29 nameIndicator

The nameIndicator identifier refers to the indication given to the mobile station that the name presentation has been invoked.

4.4.3.30 namePresentationAllowed

The namePresentationAllowed identifier refers to the presentation of the calling party's name identity to the called party.

4.4.3.31 nameUnavailable

The nameUnavailable identifier refers to the unavailability of the calling party's name identity to be offered to the called party.

4.4.3.32 namePresentationRestricted

The namePresentationRestricted identifier refers to the calling party's name identity to be offered to the called party with which presentation restriction is overridden.

4.4.3.33 deflectedToNumber

The DeflectedToNumber identifier refers to a party an incoming shall be deflected to.

4.4.3.34 deflectedToSubaddress

The DeflectedToSubaddress identifier refers to a subaddress an incoming call shall be deflected to.

4.4.3.35 uUS-Service

The uUS-Service identifier refers to the UUS service (service 1, service 2 or service 3) to be requested.

4.4.3.36 uUS-Required

The uUS-Required identifier refers to the option ("UUS required" or "UUS not required") given when requesting the UUS service.

4.4.3.37 locationNotificationArg

The locationNotificationArg identifier refers to the location notification request which is sent to the MS by the network.

4.4.3.38 notificationType

The notificationType identifier refers to the type of location notification (notification or privacy verification).

4.4.3.39 locationNotificationRes

The locationNotificationRes identifier refers to the location notification response which is sent to the network by the MS.

4.4.3.40 verificationResponse

The VerificationResponse identifier refers to the privacy verification response given by the MS user.

4.4.3.41 lcs-MOLRArg

The lcs-MOLRArg identifier refers to the MO-LR request parameters which are sent to the network by the MS.

4.4.3.42 molr-Type

The molr-Type identifier refers to the type of MO-LR.

4.4.3.43 locationMethod

The locationMethod identifier refers to the location method, for which assistance data is requested by the MS.

4.4.3.44 gpsAssistanceData

The gpsAssistanceData identifier refers to the indication, which GPS assistance data is requested by the MS.

4.4.3.45 lcs-MOLRRes

The lcs-MOLRRes identifier refers to the MO-LR response parameters which are sent to the MS by the network.

4.4.3.46 decipheringKeys

The decipheringKeys identifier refers to the set of deciphering keys, that contains Current Deciphering Key, Next Deciphering Key and Ciphering Key Flag.

4.4.3.47 multicall-Indicator

The multicall-Indicator identifier refers to the indication given to the mobile station that the number of active bearers has exceeded the maximum number.

4.5 Operations and errors implementation

For the actual implementation of supplementary services, operations and errors have to be defined by value. The following ASN.1 module, imports operation ~~type~~s from the ASN.1 module described in subclause 4.2 and operations and error ~~type~~s from MAP. ~~It defines operations by allocating operations and errors a local value. For the involved operations and errors the same local values as in MAP are allocated.~~

```
SS-Protocol {
   ~~ccitt~~itu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Access (2) modules (3) ss-Protocol (3) version8 (8)}

DEFINITIONS ::=

BEGIN

IMPORTS

OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)}

-- imports operation ~~types~~

-- imports operation ~~type~~ from MAP-MobileServiceOperations
f~~F~~orwardCheckSS-Indication
FROM MAP-MobileServiceOperations {
   ~~ccitt~~itu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-MobileServiceOperations (5) version8 (8)}

-- imports operation ~~types~~ from MAP-SupplementaryServiceOperations
```

```
rRegisterSS, eEraseSS, aActivateSS, dDeactivateSS, iInterrogateSS, rRegisterPassword, gGetPassword,
pProcessUnstructuredSS-Request, uUnstructuredSS-Request, uUnstructuredSS-Notify, eEraseCC-Entry
FROM MAP-SupplementaryServiceOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-SupplementaryServiceOperations (8) version8 (8)}

-- imports operation types from SS-Operations
pProcessUnstructuredSS-Data, nNotifySS, fForwardChargeAdvice, bBuildMPTY, hHoldMPTY, rRetrieveMPTY,
sSplitMPTY, eExplicitCT, fForwardCUG-Info, aAccessRegisterCCEntry, cCallDeflection,
uUserUserService,
lcsLCS-LocationNotification, lcsLCS-MOLR
FROM SS-Operations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-Operations (0) version8 (8)}

-- imports error types

-- imports error types from MAP-Errors
UnknownSubscriber, BearerServiceNotProvisioned, TeleserviceNotProvisioned,
IllegalSS-Operation, SS-ErrorStatus, SS-NotAvailable, SS-SubscriptionViolation,
SS-Incompatibility, SystemFailure, DataMissing, UnexpectedDataValue, PW-RegistrationFailure,
NegativePW-Check, FacilityNotSupported, CallBarred, NumberOfPW-AttemptsViolation,
AbsentSubscriber, IllegalSubscriber, IllegalEquipment, USSD-Busy, UnknownAlphabet,
ShortTermDenial, LongTermDenial, ForwardingViolation, ForwardingFailed, PositionMethodFailure

FROM MAP-Errors {
   ccitt identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-Errors (10) version8 (8)}

-- imports error types from SS-Errors
ResourcesNotAvailable, MaxNumberOfMPTY-ParticipantsExceeded,
InvalidDeflectedToNumber, SpecialServiceCode, DeflectionToServedSubscriber,
RejectedByNetwork, RejectedByUser

FROM SS-Errors {
   ccitt identified-organization (4) etsi (0) mobileDomain (0) gsm-Access (2) modules (3)
   ss-Errors (1) version8 (8)}
;

Supported-SS-Operations OPERATION ::= {forwardCheckSS-Indication | registerSS | eraseSS |
    activateSS | deactivateSS | interrogateSS | registerPassword | getPassword |
    processUnstructuredSS-Request | unstructuredSS-Request | unstructuredSS-Notify | eraseCC-Entry |
    processUnstructuredSS-Data | notifySS | forwardChargeAdvice | buildMPTY | holdMPTY |
    retrieveMPTY | splitMPTY | explicitCT | forwardCUG-Info | accessRegisterCCEntry |
    callDeflection | userUserService | lcs-LocationNotification | lcs-MOLR}


-- allocation of local values to operations

registerSS  RegisterSS ::= localValue 10
eraseSS     EraseSS ::= localValue 11
activateSS  ActivateSS ::= localValue 12
deactivateSS    DeactivateSS ::= localValue 13
interrogateSS   InterrogateSS ::= localValue 14
notifySS    NotifySS ::= localValue 16
registerPassword    RegisterPassword ::= localValue 17
getPassword     GetPassword ::= localValue 18
processUnstructuredSS-Data  ProcessUnstructuredSS-Data ::= localValue 19
forwardCheckSS-Indication   ForwardCheckSS-Indication ::= localValue 38
processUnstructuredSS-Request   ProcessUnstructuredSS-Request ::= localValue 59
unstructuredSS-Request  UnstructuredSS-Request ::= localValue 60
unstructuredSS-Notify   UnstructuredSS-Notify ::= localValue 61
eraseCCEntry    EraseCC-Entry ::= localValue 77
callDeflection  CallDeflection ::= localValue 117
userUserService UserUserService ::= localValue 118
accessRegisterCCEntry   AccessRegisterCCEntry ::= localValue 119
forwardCUG-Info ForwardCUG-Info ::= localValue 120
splitMPTY   SplitMPTY ::= localValue 121
retrieveMPTY    RetrieveMPTY ::= localValue 122
holdMPTY    HoldMPTY ::= localValue 123
buildMPTY   BuildMPTY ::= localValue 124
forwardChargeAdvice ForwardChargeAdvice ::= localValue 125
explicitCT  ExplicitCT ::= localValue 126
lcs-LocationNotification    LCS-LocationNotification::= localValue 116
lcs-MOLR    LCS-MOLR::= localValue 115


-- allocation of local values to errors
```

```
unknownSubscriber   UnknownSubscriber ::= localValue 1
illegalSubscriber   IllegalSubscriber ::= localValue 9
bearerServiceNotProvisioned BearerServiceNotProvisioned ::= localValue 10
teleserviceNotProvisioned   TeleserviceNotProvisioned ::= localValue 11
illegalEquipment    IllegalEquipment ::= localValue 12
callBarred  CallBarred ::= localValue 13
illegalSS-Operation IllegalSS-Operation ::= localValue 16
ss-ErrorStatus  SS-ErrorStatus ::= localValue 17
ss-NotAvailable SS-NotAvailable ::= localValue 18
ss-SubscriptionViolation    SS-SubscriptionViolation ::= localValue 19
ss-Incompatibility  SS-Incompatibility ::= localValue 20
facilityNotSupported    FacilityNotSupported ::= localValue 21
absentSubscriber    AbsentSubscriber ::= localValue 27
shortTermDenial ShortTermDenial ::= localValue 29
longTermDenial  LongTermDenial ::= localValue 30
systemFailure   SystemFailure ::= localValue 34
dataMissing DataMissing ::= localValue 35
unexpectedDataValue UnexpectedDataValue ::= localValue 36
pw-RegistrationFailure  PW-RegistrationFailure ::= localValue 37
negativePW-Check    NegativePW-Check ::= localValue 38
numberOfPW-AttemptsViolation    NumberOfPW-AttemptsViolation ::= localValue 43
positionMethodFailure  PositionMethodFailure ::= localValue 54
unknownAlphabet UnknownAlphabet ::= localValue 71
ussd-Busy   USSD-Busy ::= localValue 72
-- nbr-SbExceeded   Nbr-SbExceeded ::= localValue 120
--.editor's note:   a CR is needed to resolve the unresolved TypeReference "Nbr-SbExceeded".
--                  CR 011r1 is not complete!
rejectedByUser  RejectedByUser ::= localValue 121
rejectedByNetwork   RejectedByNetwork ::= localValue 122
deflectionToServedSubscriber    DeflectionToServedSubscriber ::= localValue 123
specialServiceCode  SpecialServiceCode ::= localValue 124
invalidDeflectedToNumber    InvalidDeflectedToNumber ::= localValue 125
maxNumberOfMPTY-ParticipantsExceeded    MaxNumberOfMPTY-ParticipantsExceeded ::= localValue 126
resourcesNotAvailable   ResourcesNotAvailable ::= localValue 127

END
```

3GPP TSG CN WG4 Meeting #15 ***N4-020895***
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **24.080** **CR** **024** ⌘ **rev** **-** ⌘ Current version: **5.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☐

Title:	⌘	Correction of references to FACILITY information element
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	09/07/2002
Category:	⌘	**F**
Release:	⌘	REL-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	In some sections there's a wrong reference to the section defining the FACILITY Ie
Summary of change:	⌘	References to 3.5 are replaced with references to 3.6 where necessary
Consequences if not approved:	⌘	Inconsistent references leading to confusion

Clauses affected:	⌘	2.3, 2.4.1, 2.4.2, 2.5

Other specs affected: ⌘	Y	N	
		N	Other core specifications
		N	Test specifications
		N	O&M Specifications

⌘

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

****** FIRST MODIFIED SECTION ******

2.3 Facility

This message is sent by the mobile station or the network to request or acknowledge a supplementary service. It is used when information is to be conveyed and the transaction already exists, but is not to be released. The supplementary service to be invoked, and its associated parameters, are specified in the Facility information element (see table 2.2).

Table 2.2: FACILITY message content

IEI	Information element	Type / Reference	Presence	Format	Length
	Supplementary service protocol discriminator	Protocol discriminator 3.2	M	V	1/2
	Transaction identifier	Transaction identifier 3.3	M	V	1/2
	Facility message type	Message type 3.4	M	V	1
	Facility	Facility 3.6~~5~~	M	LV	2-?

2.4 Register

2.4.1 Register (network to MS direction)

This message is sent by the network to the mobile station to assign a new transaction identifier for call independent supplementary service control and to request or acknowledge a supplementary service (see table 2.3).

Table 2.3: REGISTER message content (network to MS direction)

IEI	Information element	Type / Reference	Presence	Format	Length
	Supplementary service protocol discriminator	Protocol discriminator 3.2	M	V	1/2
	Transaction identifier	Transaction identifier 3.3	M	V	1/2
	Register message type	Message type 3.4	M	V	1
1C	Facility	Facility **3.6~~5~~**	M	TLV	2-?

2.4.2 Register (MS to network direction)

This message is sent by the mobile station to the network to assign a new transaction identifier for call independent supplementary service control and to request or acknowledge a supplementary service (see table 2.4).

Table 2.4: REGISTER message content (MS to network direction)

IEI	Information element	Type / Reference	Presence	Format	Length
	Supplementary service protocol discriminator	Protocol discriminator 3.2	M	V	1/2
	Transaction identifier	Transaction identifier 3.3	M	V	1/2
	Register Message type	Message type 3.4	M	V	1
1C	Facility	Facility 3.6~~5~~	M	TLV	2-?
7F	SS version	SS version indicator 3.8.2	O	TLV	3

2.4.2.1 SS version

This information element shall be included if the supplementary service operation being invoked is implemented according to the phase 2 or higher protocol version.

2.5 Release complete

This message is sent by the mobile station or the network to release a transaction used for call independent supplementary service control. It may also request or acknowledge a supplementary service (see table 2.5).

Table 2.5: RELEASE COMPLETE message content

IEI	Information element	Type / Reference	Presence	Format	Length
	Supplementary service protocol discriminator	Protocol discriminator 3.2	M	V	1/2
	Transaction identifier	Transaction identifier 3.3	M	V	1/2
	Release Complete message type	Message type 3.4	M	V	1
08	Cause	Cause TS 24.008	O	TLV	4-32
1C	Facility	Facility 3.6~~5~~	O	TLV	2-?

****** END OF MODIFICATIONS ******

3GPP TSG CN WG4 Meeting #15 ***N4-021039***
Helsinki, Finland, 29th July – 2nd August 2002 **(Revision of N4-020807)**

CR-Form-v7

CHANGE REQUEST

⌘ **23.018** **CR** **109** ⌘ **rev** **1** ⌘ Current version: **5.3.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Determining the basic service for MT calls
Source:	⌘	CN4
Work item code:	⌘	TEI_5
Date:	⌘	30/07/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)

Reason for change:	⌘	CR 29.007-048r1, approved at CN #16, changed the rules for determining the basic service which applies for an MT call. The procedures Derive_Requested_Basic_Service_HLR and Derive_CS_BC_MSC are no longer aligned with the rules defined in TS 29.007. Furthermore, the procedure Derive_Requested_Basic_Service_VLR has no counterpart in TS 29.007.
Summary of change:	⌘	Replace explicit definitions of procedures to determine the basic service & PLMN BC for an MT call with references to TS 29.007
Consequences if not approved:	⌘	Misaligned specifications

Clauses affected:	⌘	1; 3.1; 3.2; 6; 7; 7.1.1.3; Figures 8a & 8b; 7.2.2.3; 7.2.2.7; Figures 44c, 46, 50, 55a, 55b, 55c, 55e, 55f & 55g; 7.3.1.4; Figures 67a, 70a & 72; 7.3.2.1; 7.3.2.2; Figures 76a & 77; 8.1.2; 8.1.3; 8.1.9; 8.1.24; 8.1.36; 8.1.39; 8.3.1;

	Y	N			
Other specs affected:	X		Other core specifications	⌘	CR 29.007-048r1 (approved in CN #16) CR 29.007-053 CR 23.083-010
		X	Test specifications		
		X	O&M Specifications		

Other comments:	⌘	Incorrect references to GSM specifications have been corrected. References to the GSM bearer capability and CS bearer capability have been replaced by references to the PLMN bearer capability, to align with 29.007. References to GSM basic services and supplementary services have been replaced by references to GSM or UMTS basic services and supplementary

	services. References to the UMTS MS have been replaced by references to the UMTS UE.

****** First modified section ******

1 Scope

The present document specifies the technical realisation of the handling of calls originated by a UMTS or GSM mobile subscriber and calls directed to a UMTS or GSM mobile subscriber, up to the point where the call is established. Normal release of the call after establishment is also specified.

In the present document, the term MS is used to denote a UMTS UE or GSM MS, as appropriate.

The handling of DTMF signalling and Off-Air Call set-up (OACSU) are not described in the present document.

The details of the effects of UMTS or GSM supplementary services on the handling of a call are described in the relevant 23.07x, 23.08x and 23.09x series of specifications.

The specification of the handling of a request from the HLR for subscriber information is not part of basic call handling, but is required for both CAMEL (3GPP TS 23.078 [12]) and optimal routeing (3GPP TS 23.079 [13]). The use of the Provide Subscriber Information message flow is shown in 3GPP TS 23.078 [12] and 3GPP TS 23.079 [13].

The specification of the handling of data calls re-routed to a SIWFS is described in 3GPP TS 23.054 [8].

The logical separation of the MSC and VLR (shown in clauses 4, 5 and 7), and the messages transferred between them (described in clause 8) are the basis of a model used to define the externally visible behaviour of the MSC/VLR, which is a single physical entity. They do not impose any requirement except the definition of the externally visible behaviour.

If there is any conflict between the present document and the corresponding stage 3 specifications (3GPP TS 24.008 [26], 3GPP TS 25.413 [27], ~~GSM~~3GPP TS 48.008 [2] and 3GPP TS 29.002 [29]), the stage 3 specification shall prevail.

****** Next modified section ******

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following definitions apply:

A subscriber: The calling mobile subscriber.

B subscriber: The mobile subscriber originally called by the A subscriber.

C subscriber: The subscriber to whom the B subscriber has requested that calls be forwarded. The C subscriber may be fixed or mobile.

~~**Circuit Switched Bearer Capability:** The information transferred over the UMTS or GSM access interface to define the information transfer capabilities to be used between the MS and the network for a circuit switched connection.~~

Location Information: Information to define the whereabouts of the MS, and the age of the information defining the whereabouts.

PLMN Bearer Capability: The information transferred over the UMTS or GSM access interface to define the information transfer capabilities to be used between the MS and the network for a circuit-switched connection.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

A&O	Active & Operative
ACM	Address Complete Message
ANM	ANswer Message
AoC	Advice of Charge
BC	Bearer Capability
BOIC-exHC&BOIZC	Barring of Outgoing International Calls except those directed to the HPLMN Country & Barring of Outgoing InterZonal Calls
BOIZC	Barring of Outgoing InterZonal Calls
BOIZC-exHC	Barring of Outgoing InterZonal Calls except those directed to the HPLMN Country
CCBS	Completion of Calls to Busy Subscriber
CFB	Call Forwarding on Busy
CFNRc	Call Forwarding on mobile subscriber Not Reachable
CFNRy	Call Forwarding on No Reply
CFU	Call Forwarding Unconditional
CLIP	Calling Line Identity Presentation
CLIR	Calling Line Identity Restriction
COLP	COnnected Line identity Presentation
COLR	COnnected Line identity Restriction
~~CS BC~~	~~Circuit Switched Bearer Capability~~
CUG	Closed User Group
CW	Call Waiting
FTN	Forwarded-To Number
FTNW	Forwarded-To NetWork
GMSCB	Gateway MSC of the B subscriber
GPRS	General Packet Radio Service
HLC	Higher Layer Compatibility
HLRB	The HLR of the B subscriber
HPLMNB	The HPLMN of the B subscriber
IAM	Initial Address Message
IPLMN	Interrogating PLMN - the PLMN containing GMSCB
IWU	Inter Working Unit
LLC	Lower Layer Compatibility
MO	Mobile Originated
MPTY	MultiParTY
MT	Mobile Terminated
NDUB	Network Determined User Busy
NRCT	No Reply Call Timer
PLMN BC	(GSM or UMTS) PLMN Bearer Capability
PRN	Provide Roaming Number
SGSN	Serving GPRS support node
SIFIC	Send Information For Incoming Call
SIFOC	Send Information For Outgoing Call
SIWF	Shared Inter Working Function
SIWFS	SIWF Server. SIWFS is the entity where the used IWU is located.
SRI	Send Routeing Information
UDUB	User Determined User Busy
VLRA	The VLR of the A subscriber
VLRB	The VLR of the B subscriber
VMSCA	The Visited MSC of the A subscriber
VMSCB	The Visited MSC of the B subscriber
VPLMNA	The Visited PLMN of the A subscriber
VPLMNB	The Visited PLMN of the B subscriber

****** Next modified section ******

6 Principles for interactions with supplementary services

This clause specifies the principles used to describe the invocation of the GSM or UMTS supplementary services which were standardised when the present document was drafted. Registration, erasure, activation, deactivation and interrogation are call-independent operations; they are therefore outside the scope of the present document. Descriptions may be found in the stage 2 specifications for each supplementary service.

In the modelling used in the present document, each supplementary service which a network entity supports is managed by a supplementary service handler, which handles data in the entity in which it runs. The call handling processes defined in the present document use the data to define the contents of messages to other entities. The basic call handling processes defined in the present document interact with the supplementary service handlers as shown in the SDL diagrams and the supporting text. If a network entity does not support a supplementary service, it bypasses the interaction with the handler for that supplementary service. Exceptions to this general principle are described later in this clause.

****** Next modified section ******

7 Functional requirements of network entities

The text in this clause is a supplement to the definition in the SDL diagrams; it does not duplicate the information in the SDL diagrams.

The entities described in this clause interwork with other entities over four different types of interface:

- The Iu interface, used to interwork between the MSC and the UTRAN or the UMTS UE~~MS~~;
- The A interface, used to interwork between the MSC and the GSM BSS or the GSM MS;
- The C, D & F interfaces, used to interwork between the MSC & HLR (C), VLR & HLR (D) and MSC & EIR (F);
- Telephony signalling interfaces, used to interwork between an MSC and another exchange.

The protocols used over the Iu interface are RANAP, which is specified in 3GPP TS 25.413 [27], for interworking with the UTRAN and DTAP, which is specified in 3GPP TS 24.008 [26], for interworking with the MS.

The protocols used over the A interface are BSSMAP, which is specified in ~~GSM~~3GPP TS 48.008 [2], for interworking with the BSS and DTAP, which is specified in 3GPP TS 24.008 [26], for interworking with the MS.

The protocol used over the C, D & F interfaces is MAP, which is specified in 3GPP TS 29.002 [29].

For the purposes of the present document, the protocol used over telephony signalling interfaces is ISUP, which is specified in ITU-T Recommendations Q.761[33], Q.762 [34], Q.763 [35] and Q.764 [36]; other telephony signalling systems may be used instead.

The present document shows the call handling application processes interworking with a protocol handler for each of the protocols listed above. Each protocol defines supervision timers. If a supervision timer expires before a distant entity responds to a signal, the handling is as defined in the appropriate protocol specification. In general, the protocol handler reports timer expiry to the application as an error condition or negative response. Where a timer is shown in the present document, therefore, it is an **application** timer rather than a **protocol** timer. Interworking with the protocol handlers uses functional signal names which do not necessarily have a one-to-one correspondence with the names of messages used in the protocols.

An MSC which receives an IAM from an originating exchange may react in three different ways:

- It acts as a transit exchange, i.e. it relays the IAM to a destination exchange determined by analysis of the called party address, and thereafter relays other telephony signalling between the originating and destination exchange until the connection is released. This behaviour is not specific to UMTS or GSM;

- It acts as a terminating exchange, i.e. it attempts to connect the call to an MS currently registered in the service area of the MSC;

- It acts as a GMSC, i.e. it interrogates an HLR for information to route the call. If the HLR returns routeing information, the MSC uses the routeing information from the HLR to construct an IAM, which it sends to a destination exchange determined by analysis of the routeing information from the HLR.

Annex A describes the method which the MSC uses to decide how to process the IAM.

The SDL diagrams in this clause show the handling for a number of optional features and services. If the handling consists only of a call to a procedure specific to the feature or service, the procedure call is omitted if the entity does not support an optional feature or service. If the handling consists of more than a call to a procedure specific to the feature or service, the text associated with each SDL diagram specifies the handling which applies if the entity does not support an optional feature or service. For simplicity of description, it is assumed that support for Operator Determined Barring and the Call Forwarding and Call Barring supplementary services is mandatory.

**** Next modified section ****

7.1.1.3 Procedure OG_Call_Setup_MSC

Sheet 1: the variables Alerting sent, MS connected and Reconnect are global data, accessible to the procedures CCBS_Check_OG_Call, CCBS_OCH_Report_Failure, CCBS_OCH_Report_Success, CCBS_Check_If_CCBS_Possible, Send_Alerting_If_Required and Send_Access_Connect_If_Required.

Sheet 1: the variable UUS1 result sent is specific to UUS. This variable is accessible to all UUS specific procedures.

Sheet 1: the procedure UUS_OCH_Check_Setup is specific to UUS; it is specified in 3GPP TS 23.087 [20].

Sheet 1: the VMSC converts the ~~GSM~~ PLMN bearer capability negotiated between the VMSC and the MS to a ~~GSM~~ basic service according to the rules defined in 3GPP TS 27.001 [28].

Sheet 1: the procedure CAMEL_N_CSI_CHECK_MSC is specific to CAMEL Phase 3 or later, it is specified in 3GPP TS 23.078 [12].

Sheet 1: the procedure Check_OG_Multicall_MSC is specific to Multicall; it is specified in 3GPP TS 23.135 [25]. If the VMSC does not support Multicall, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 1: the variable "On_Hold" is used only if the VMSC supports Call Hold.

Sheet 1, sheet 2, sheet 3, sheet 6: the procedure CCBS_OCH_Report_Failure is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 1, sheet 2, sheet 6, sheet 7, sheet 9: at any stage after the Set-up has been received, the MS may terminate the transaction with the network by sending a Release transaction request.

Sheet 2, sheet 3, sheet 4, sheet 5, sheet 6, sheet 7, sheet 8, sheet 9: signals are sent to and received from the process Subs_FSM as described in subclause 7.4.

Sheet 3: the procedure Set_CLI_Presentation_Indicator_MSC is specific to CLIR. If the VMSC does not support CLIR, processing continues from the "Yes" exit of the test "Result=Call allowed?".

Sheet 3: the procedure CAMEL_OCH_MSC_INIT is specific to CAMEL; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 3: the procedure CAMEL_MO_Dialled_Services is specific to CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "Pass" exit of the test "Result?".

Sheet 3: the procedure CCBS_Check_OG_Call is specific to CCBS; it is specified in 3GPP TS 23.093 [23]. If the VMSC does not support CCBS, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 3: the procedure MOBILE_NUMBER_PORTABILITY_IN_OQoD is specific to Mobile Number Portability; it is specified in 3GPP TS 23.066 [10].

Sheet 3: the procedure UUS_OCH_Set_Info_In_IAM is specific to UUS; it is specified in 3GPP TS 23.087 [20].

Sheet 3: the procedure CAMEL_Store_Destination_Address is specific to CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 3: the procedure CCBS_OCH_Report_Success is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 3, sheet 5: the procedure CAMEL_OCH_LEG1_MSC is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 4, sheet 7: the procedures CAMEL_Start_TNRy and CAMEL_Stop_TNRy are specific to CAMEL phase 2 or later; they are specified in 3GPP TS 23.078 [12].

Sheet 4: the task "UTU2Cnt := 0" is executed only if the VMSC supports UUS

Sheet 4: the procedure CAMEL_OCH_MSC_ALERTING is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 4 or later, processing continues from the "Pass" exit of the test "Result?".

Sheet 5: the procedure CAMEL_OCH_MSC_ANSWER is specific to CAMEL; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 5: the procedure Set_COLP_Info_MSC is specific to COLP.

Sheet 5: the procedure Handle_AoC_MO_MSC is specific to AoC.

Sheet 5: the task "Store CW treatment indicator for this call if received in SII2" is executed only if the VMSC supports CAMEL phase 3 or later.

Sheet 5: The process CAMEL_OCH_LEG2_MSC is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 6: the procedures CCBS_Check_If_CCBS_Possible and CCBS_Activation_MSC are specific to CCBS; they are specified in 3GPP TS 23.093 [23]. The task "Store CCBS Result" is executed only if the VMSC supports CCBS. If the VMSC does not support CCBS, processing continues from the "CCBS Not Possible" exit of the test "CCBS Result".

Sheet 6, sheet 7: the procedure CAMEL_OCH_MSC_DISC3 is specific to CAMEL Phase 1; it is specified in 3GPP TS 23.078 [12].

Sheet 6, sheet 7: the procedure CAMEL_OCH_MSC_DISC4 is specific to CAMEL Phase 2 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 6, sheet 6: the procedure CAMEL_OCH_MSC1 is specific to CAMEL phase 2 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 2 or later, processing continues from the "No" exit of the test "Result=Reconnect?"

Sheet 6, sheet 7, sheet 9: the processing in the branch beginning with the Int_Release_Call input will occur only if the MSC supports CAMEL.

Sheet 7, sheet 9: the procedure UUS_MSC_Check_UUS1_UUI is specific to UUS; it is specified in 3GPP TS 23.087 [20].

Sheet 8: the input signal TNRy expired and all the subsequent processing are specific to CAMEL phase 2 or later, and will occur only if the VMSC supports CAMEL phase 2 or later. The procedure CAMEL_OCH_MSC2 is specified in 3GPP TS 23.078 [12].

Sheet 8: the input signal User To User is specific to UUS; it is discarded if the VMSC does not support UUS.

Sheet 8: the procedures UUS_MSC_Check_UUS2_UUI_to_MS and UUS_MSC_Check_UUS2_UUI_to_NW are specific to UUS; they are specified in 3GPP TS 23.087 [20].

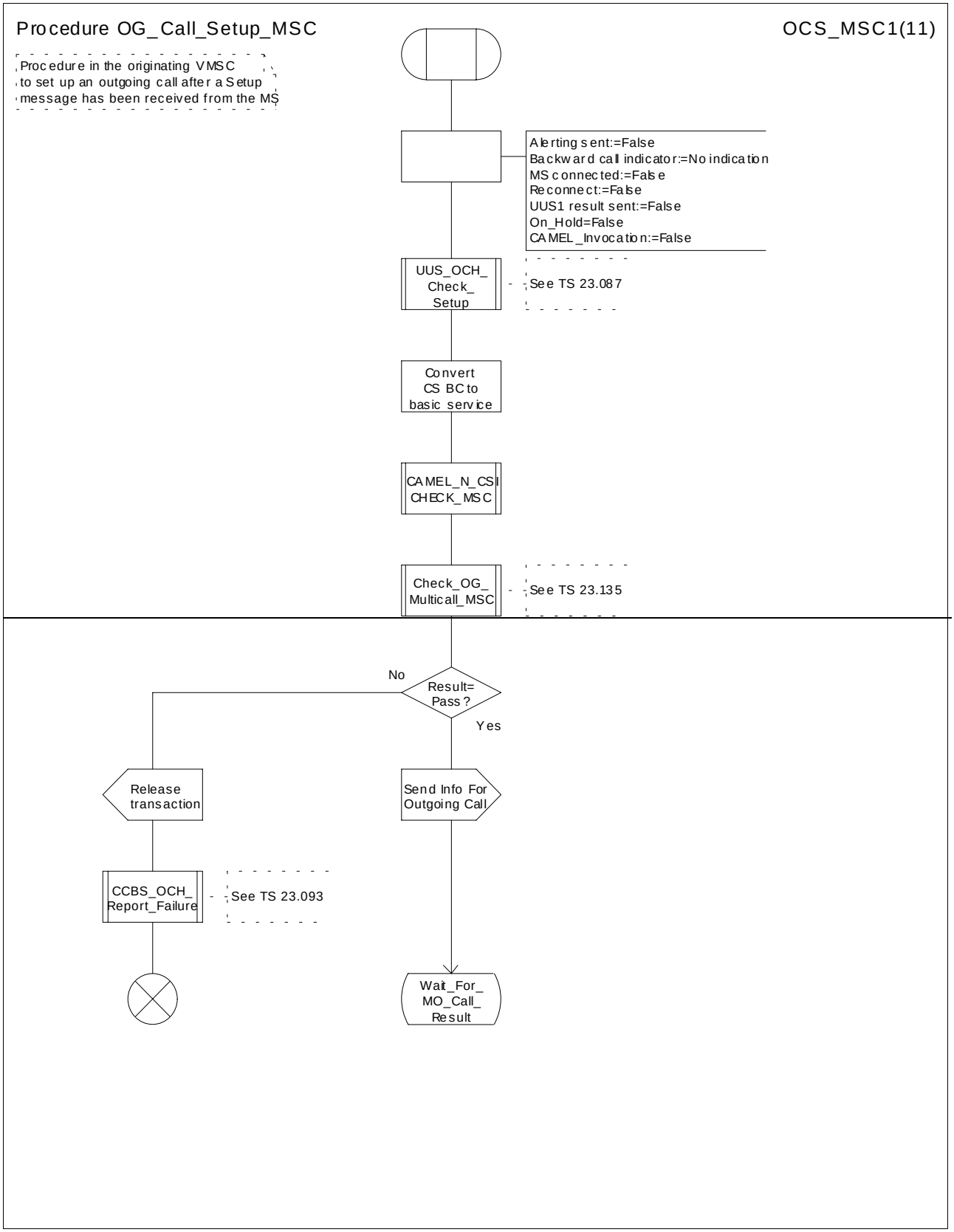
Sheet 9: the procedure CAMEL_OCH_MSC_DISC1 is specific to CAMEL; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL, processing continues from the "No" exit of the test "Result=CAMEL handling?".

Sheet 9: the procedure CAMEL_OCH_MSC_DISC2 is specific to CAMEL; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL, processing continues from the "No" exit of the test "Result=CAMEL handling?".

Sheet 10: the procedure Process_Hold_Request is specific to Call Hold; it is specified in 3GPP TS 23.083[16].

Sheet 10: the procedure Process_Retrieve_request is specific to Call Hold; it is specified in 3GPP TS 23.083[16].

****** Next modified section ******


Procedure OG_Call_Setup_MSC
OCS_MSC1(11)
Procedure in the originating VMSC to set up an outgoing call after a Setup message has been received from the MS
Alerting sent:=False
Backward call indicator:=No indication
MS connected:=False
Reconnect:=False
UUS1 result sent:=False
On_Hold=False
CAMEL_Invocation:=False
UUS_OCH_ Check_ Setup
See TS 23.087
Convert CS BC to basic service
CAMEL_N_CSI CHECK_MSC
Check_OG_ Multicall_MSC
See TS 23.135
No
Result= Pass ?
Yes
Release transaction
Send Info For Outgoing Call
CCBS_OCH_ Report_Failure
See TS 23.093
Wait_For_ MO_Call_ Result

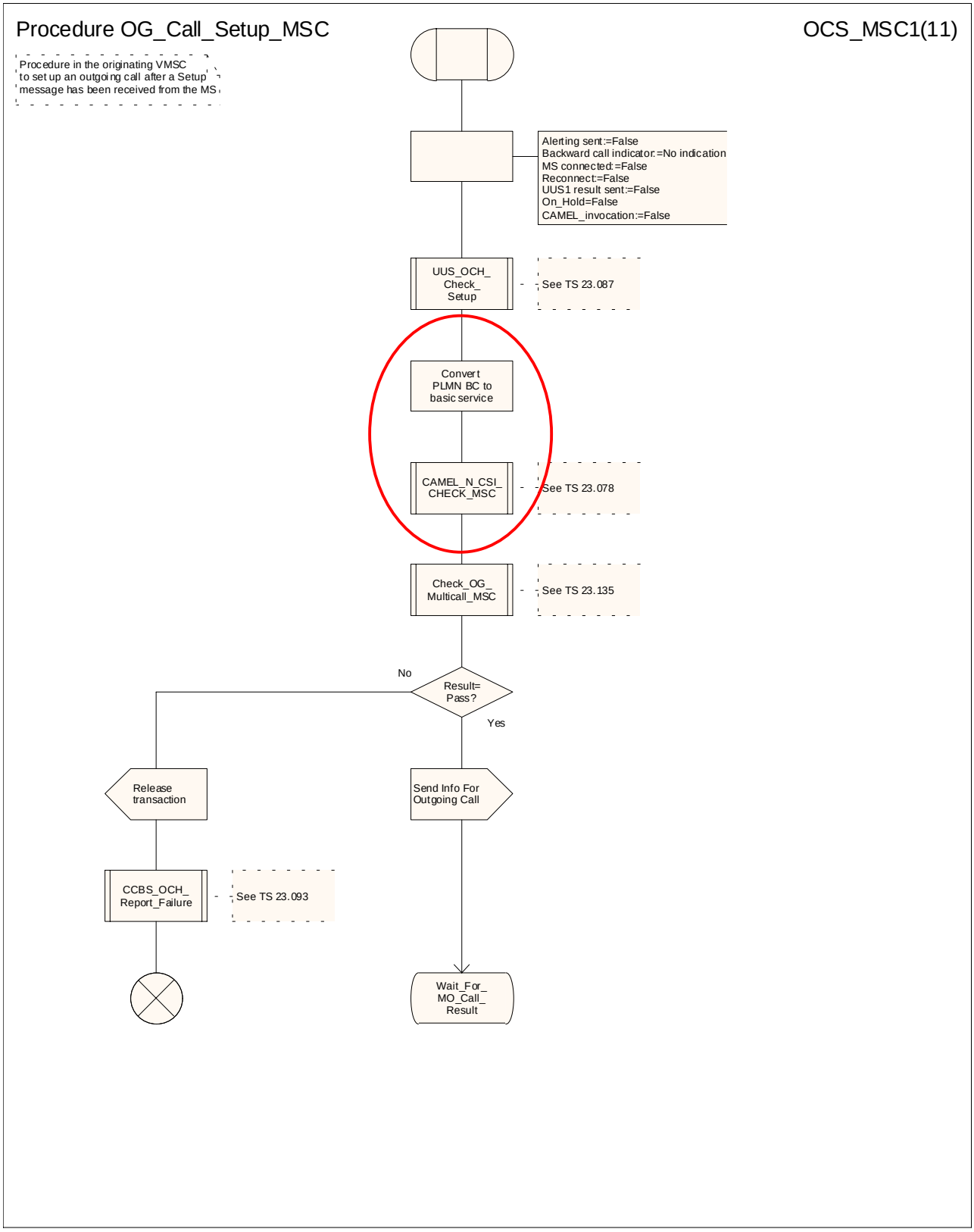
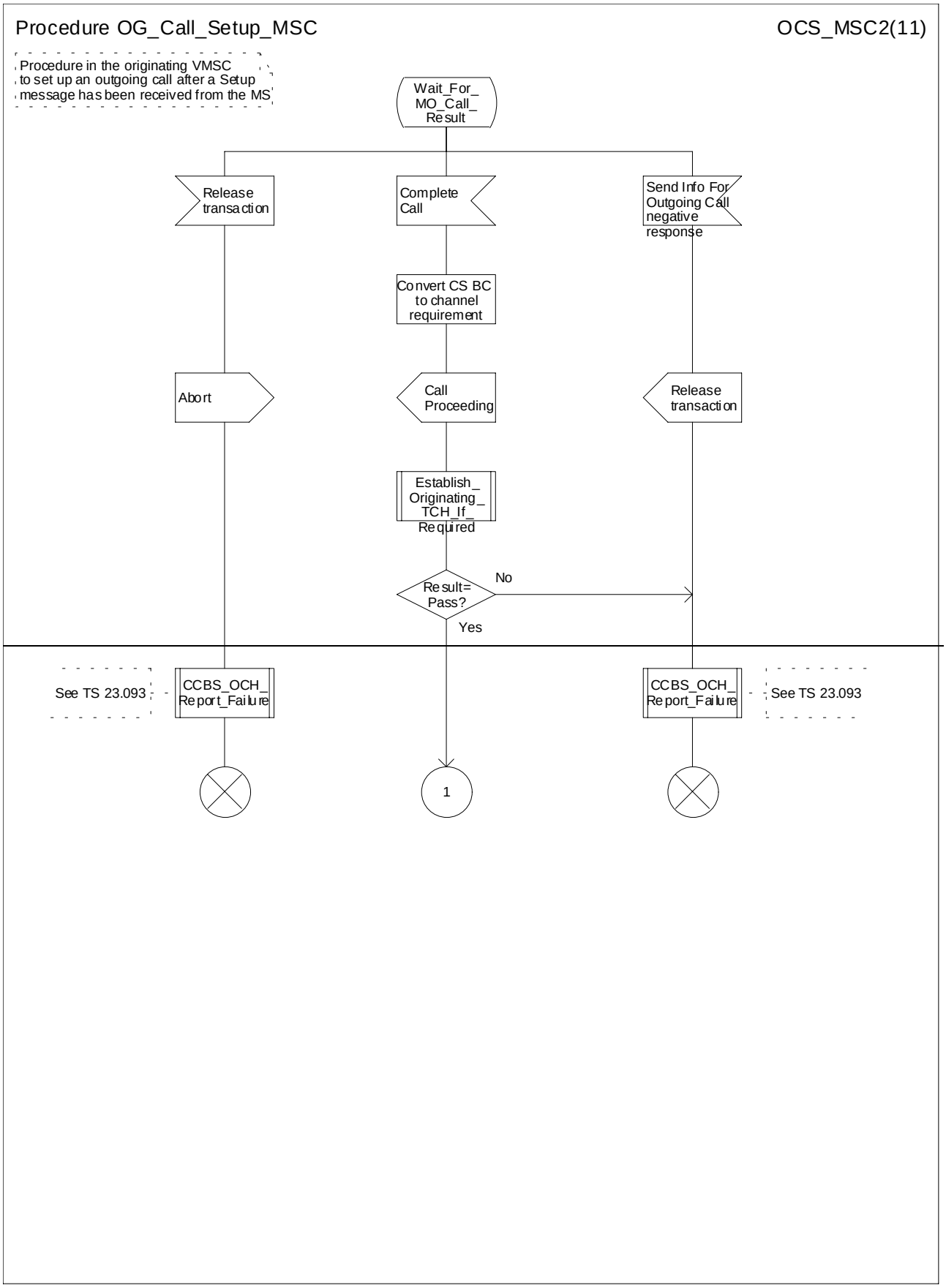


Figure 8a: Procedure OG_Call_Setup_MSC (sheet 1)


Procedure OG_Call_Setup_MSC
OCS_MSC2(11)
Procedure in the originating VMSC to set up an outgoing call after a Setup message has been received from the MS
Wait_For_MO_Call_Result
Release transaction
Complete Call
Send Info For Outgoing Call negative response
Convert CS BC to channel requirement
Abort
Call Proceeding
Release transaction
Establish_Originating_TCH_If_Required
Result= Pass?
No
Yes
See TS 23.093
CCBS_OCH_Report_Failure
CCBS_OCH_Report_Failure
See TS 23.093
1

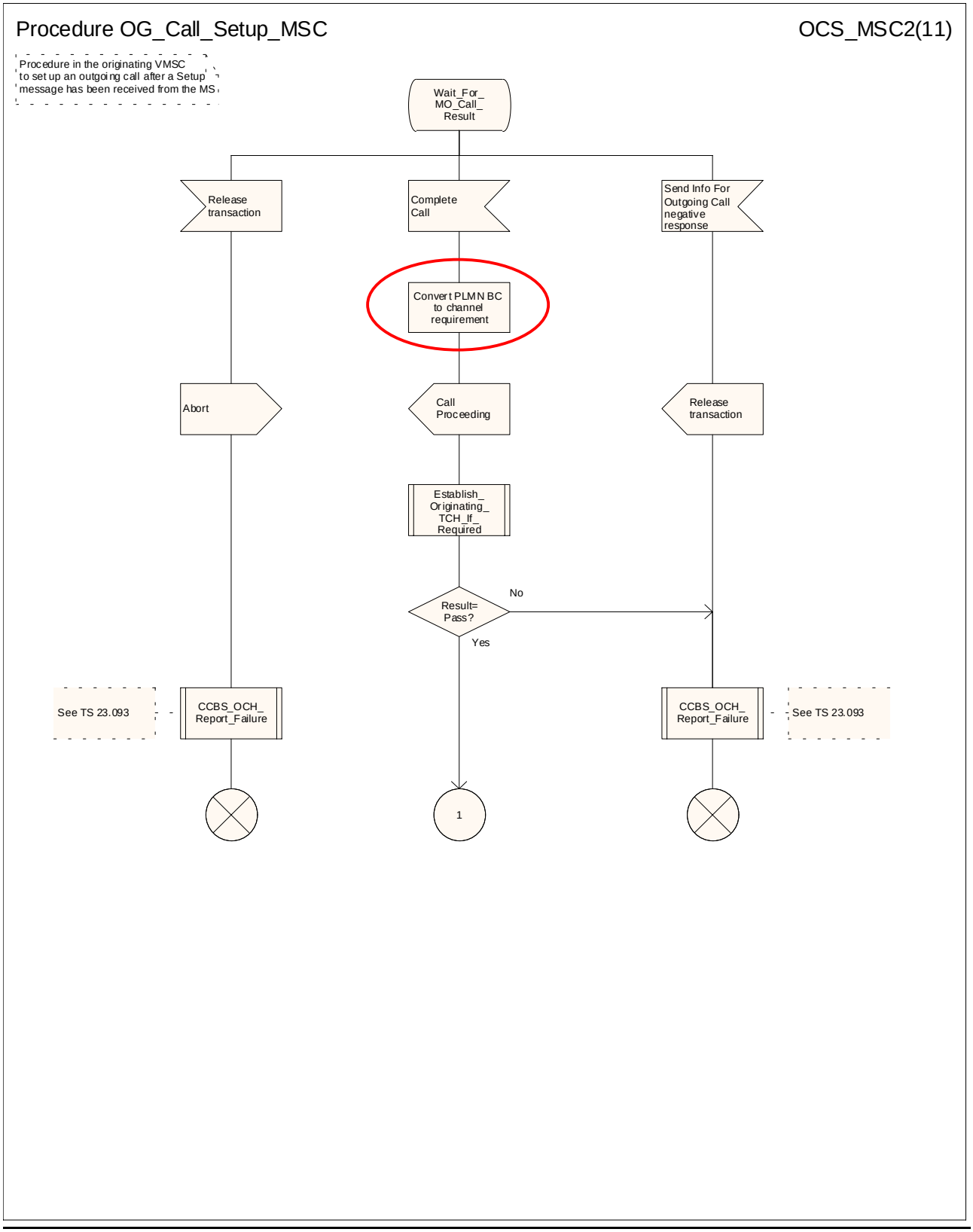


Figure 8b: Procedure OG_Call_Setup _MSC (sheet 2)

****** Next modified section ******

7.2.2.3 Procedure Subscription_Check_HLR

The HLR derives the possible PLMN bearer capability to populate the parameter in the Provide Roaming Number request according to the rules defined in 3GPP TS 29.007 [30].

If the HLR is able to determine the PLMN bearer capability or equivalent ISDN compatibility information to be sent to the VLR in the Provide Roaming Number request, it applies the corresponding PLMN bearer service or teleservice for handling the call. If the HLR is not able to determine any compatibility information to be sent to the VLR in the Provide Roaming Number request, it applies a default basic service according to the requirements of the operator.

It is an implementation option to carry out the check for operator determined barring of incoming calls before the check on provisioning of the requested basic service.

The test "gsmSCF Initiated Call?" is specific to CAMEL phase 4 or later. If the HLR does not support CAMEL phase 4 or later, processing continues from the "No" exit.

The negative response "Call barred" indicates whether the reason is operator determined barring or supplementary service barring, according to the result returned by the procedure Check_IC_Barring.

The negative response "CUG reject" indicates whether the reason is:

- Incoming calls barred within CUG;
- Requested basic service violates CUG constraints;
- Subscriber not member of CUG;

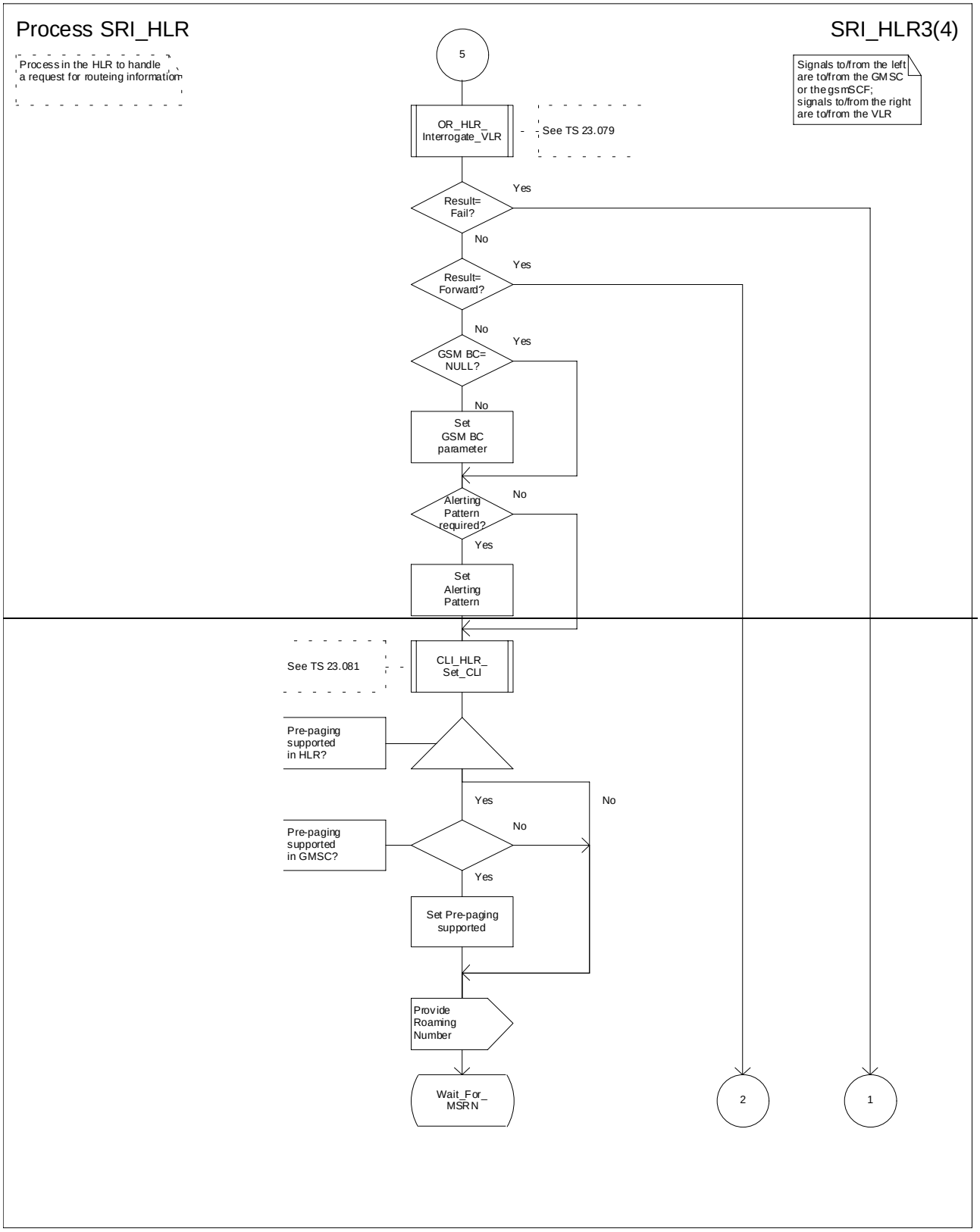
according to the cause returned by the procedure IC_CUG_Check.

****** Next modified section ******

7.2.2.7 ~~Procedure Derive_Requested_Basic_Service_HLR~~Void

~~The rules for deriving a Circuit switched bearer capability from ISDN compatibility information or the MSISDN of the B subscriber are specified in 3GPP TS 29.007 [30]. If a GSM bearer capability cannot be derived from the ISDN compatibility information or the MSISDN of the B subscriber, the HLR applies a default basic service according to the requirements of the operator.~~

****** Next modified section ******


Process SRI_HLR
SRI_HLR3(4)
Process in the HLR to handle a request for routeing information
Signals to/from the left are to/from the GMSC or the gsmSCF; signals to/from the right are to/from the VLR
5
OR_HLR_ Interrogate_VLR
See TS 23.079
Result= Fail?
Yes
No
Result= Forward?
Yes
No
GSM BC= NULL?
Yes
No
Set GSM BC parameter
Alerting Pattern required?
No
Yes
Set Alerting Pattern
See TS 23.081
CLI_HLR_ Set_CLI
Pre-paging supported in HLR?
Yes
No
Pre-paging supported in GMSC?
No
Yes
Set Pre-paging supported
Provide Roaming Number
Wait_For_ MSRN
2
1

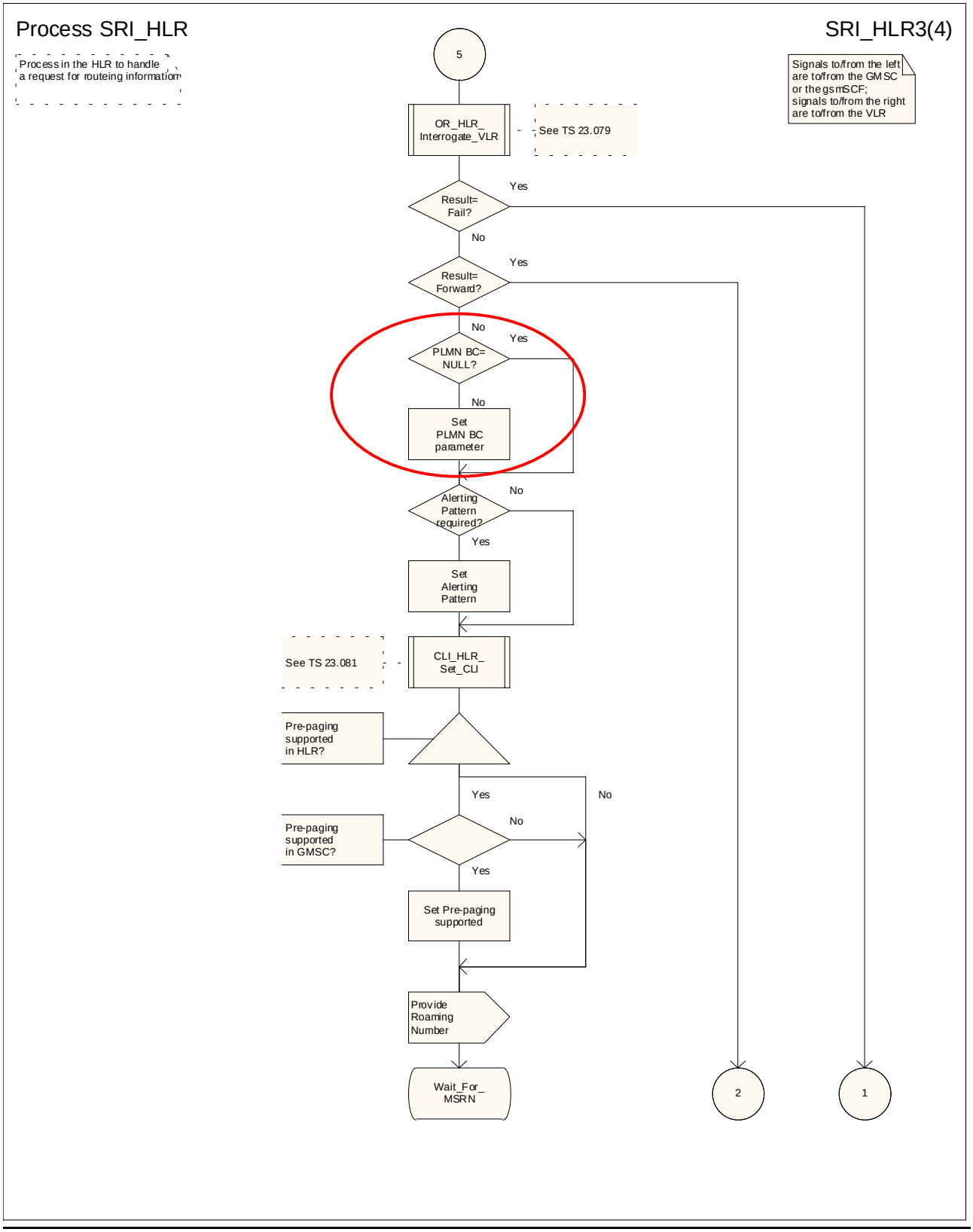
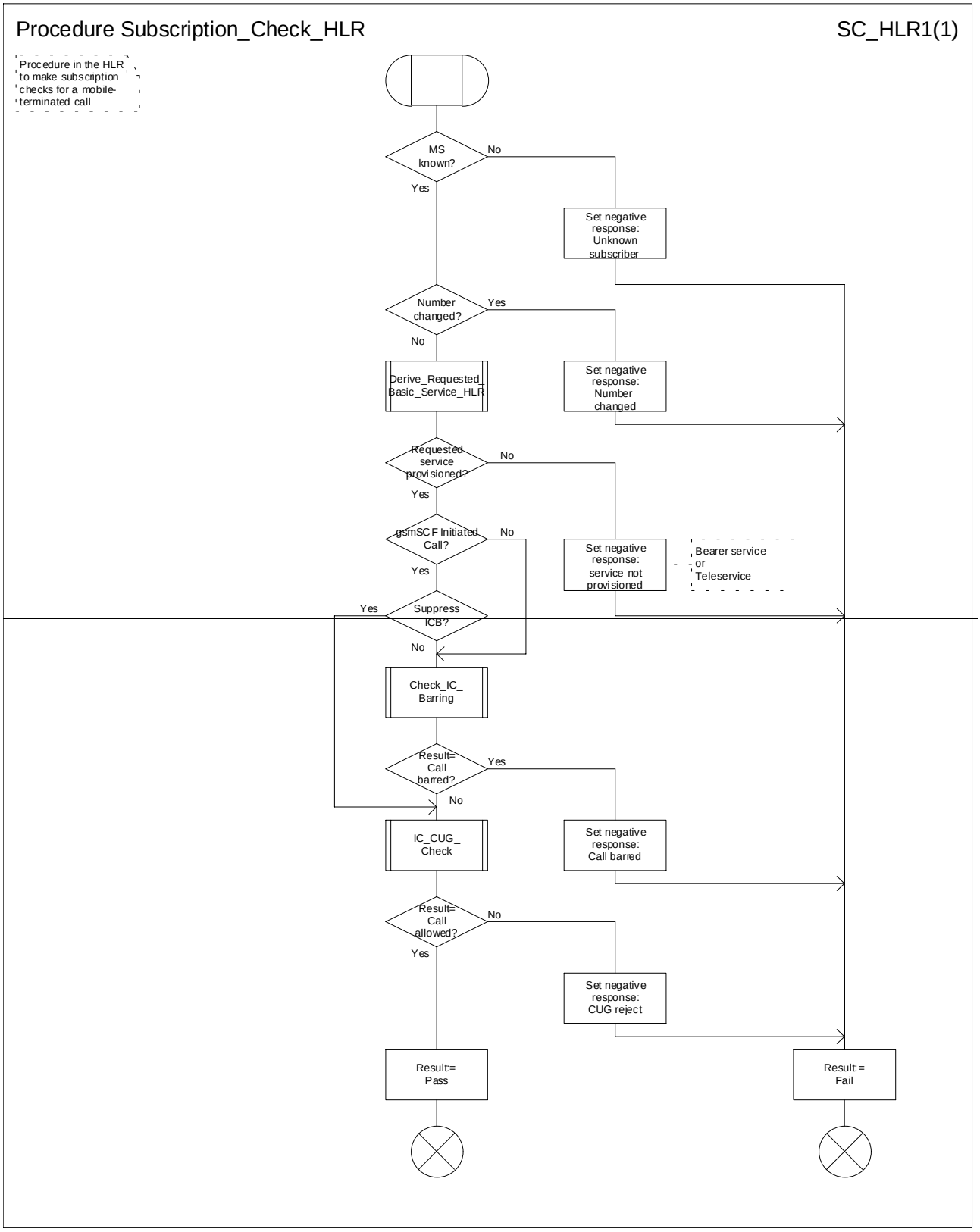


Figure 44c: Process SRI_HLR (sheet 3)

****** Next modified section ******


Procedure Subscription_Check_HLR
SC_HLR1(1)
Procedure in the HLR to make subscription checks for a mobile-terminated call
MS known?
No
Yes
Set negative response: Unknown subscriber
Number changed?
Yes
No
Derive_Requested Basic_Service_HLR
Set negative response: Number changed
Requested service provisioned?
No
Yes
gsmSCF Initiated Call?
No
Yes
Set negative response: service not provisioned
Bearer service or Teleservice
Yes
Suppress ICB?
No
Check_IC_ Barring
Result= Call barred?
Yes
No
IC_CUG_ Check
Set negative response: Call barred
Result= Call allowed?
No
Yes
Set negative response: CUG reject
Result= Pass
Result= Fail

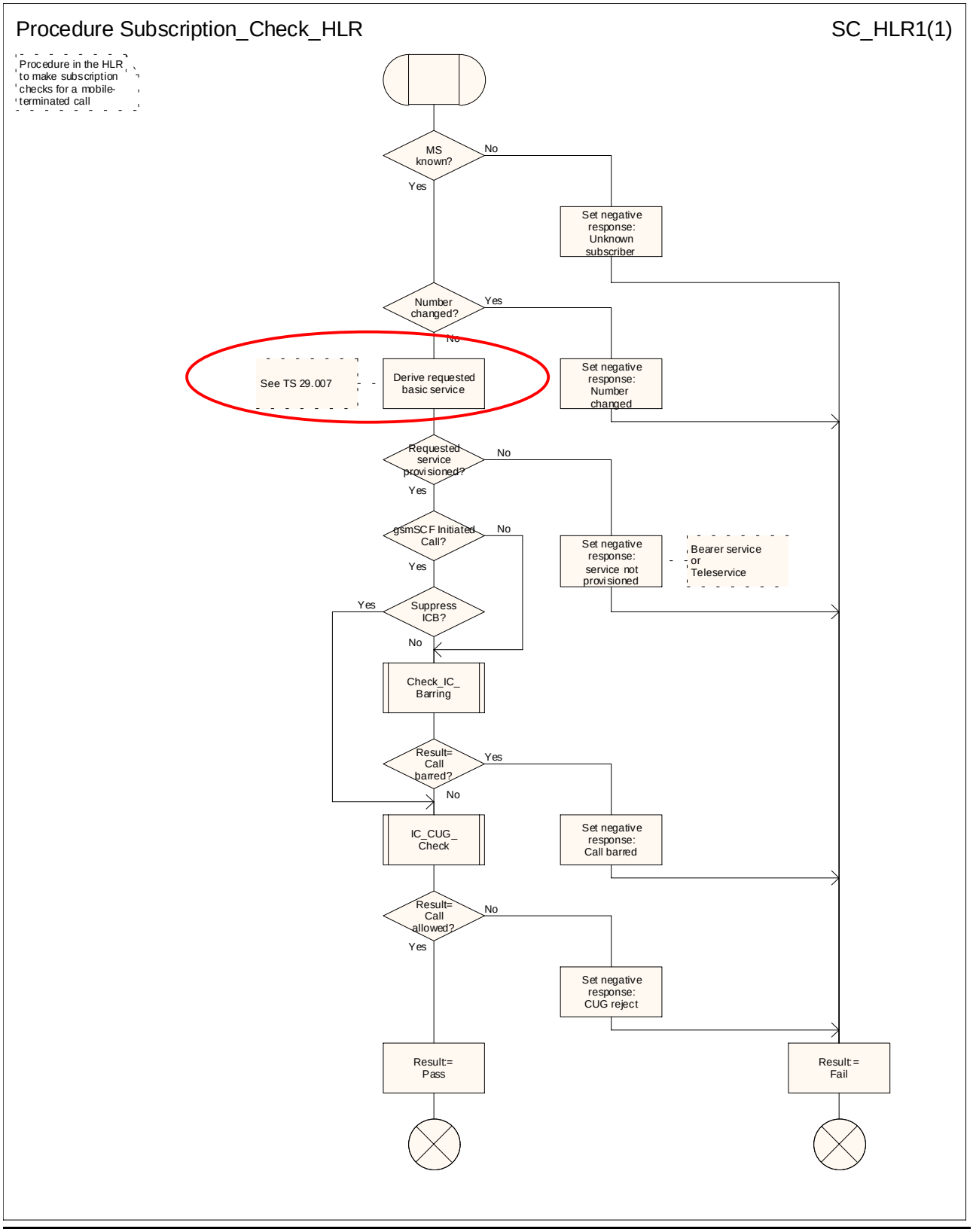


Figure 46: Procedure Subscription_Check_HLR

****** Next modified section ******

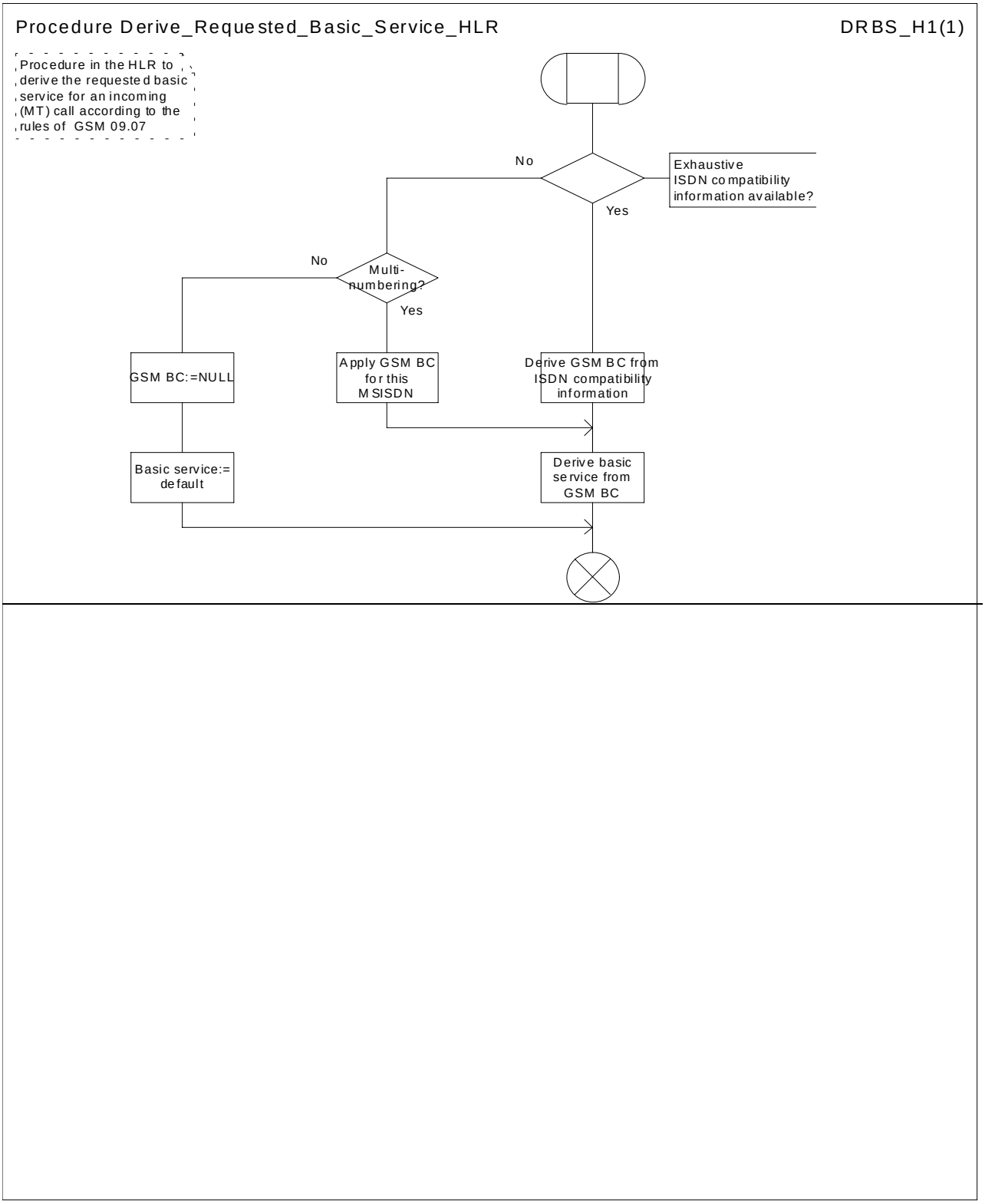
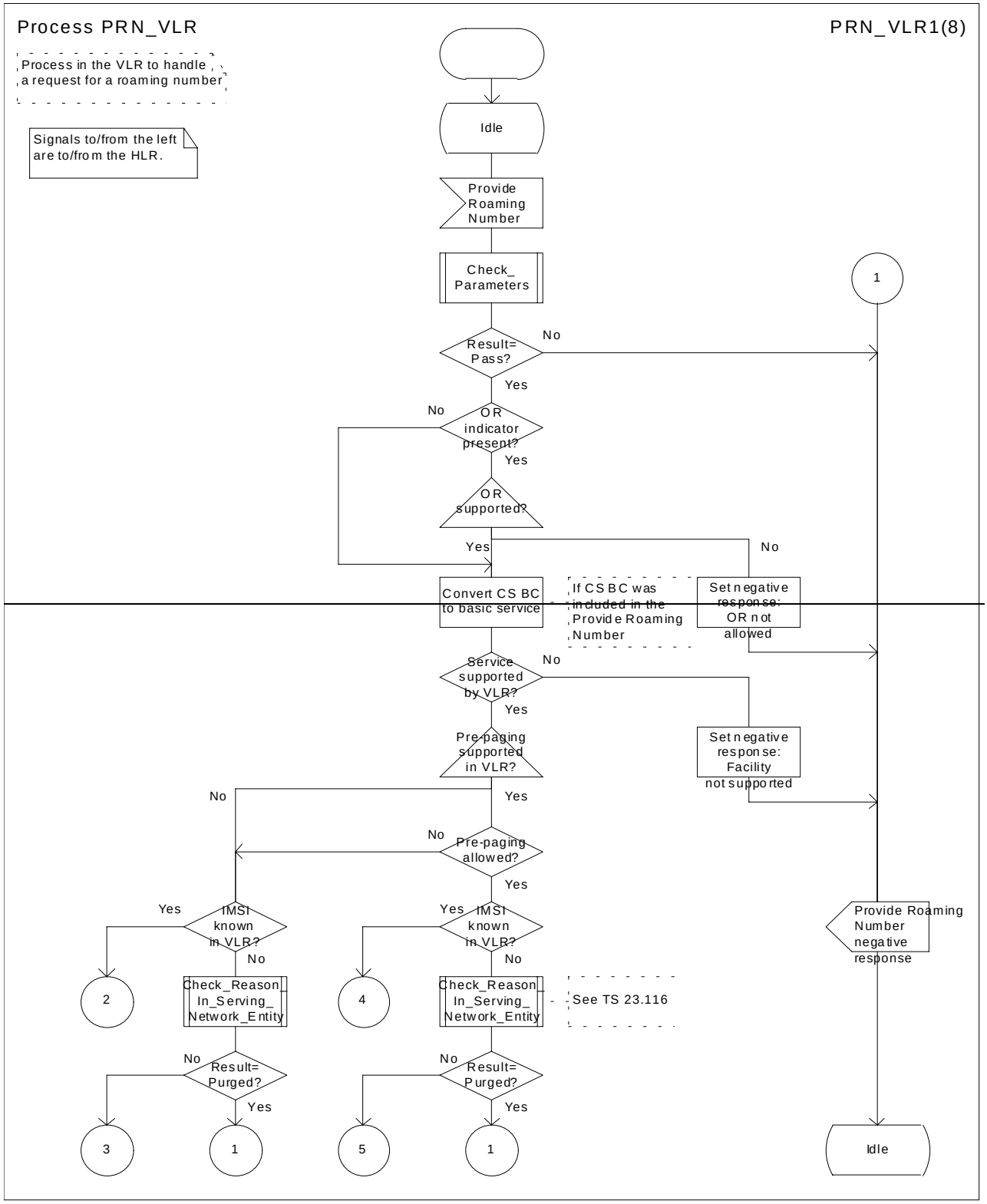


Figure 50: ~~Procedure Derive_Requested_Basic_Service_HLR~~Void

****** Next modified section ******


Process PRN_VLR
PRN_VLR1(8)
Process in the VLR to handle a request for a roaming number
Signals to/from the left are to/from the HLR.
Idle
Provide Roaming Number
Check_ Parameters
1
Result= Pass?
No
Yes
No
OR indicator present?
Yes
OR supported?
Yes
No
Convert CS BC to basic service
If CSBC was included in the Provide Roaming Number
Set negative response: OR not allowed
Service supported by VLR?
No
Yes
Pre-paging supported in VLR?
Set negative response: Facility not supported
No
Yes
No
Pre-paging allowed?
Yes
Yes
IMSI known in VLR?
No
Yes
IMSI known in VLR?
No
Provide Roaming Number negative response
2
Check_Reason In_Serving_ Network_Entity
4
Check_Reason In_Serving_ Network_Entity
See TS 23.116
No
Result= Purged?
Yes
No
Result= Purged?
Yes
3
1
5
1
Idle

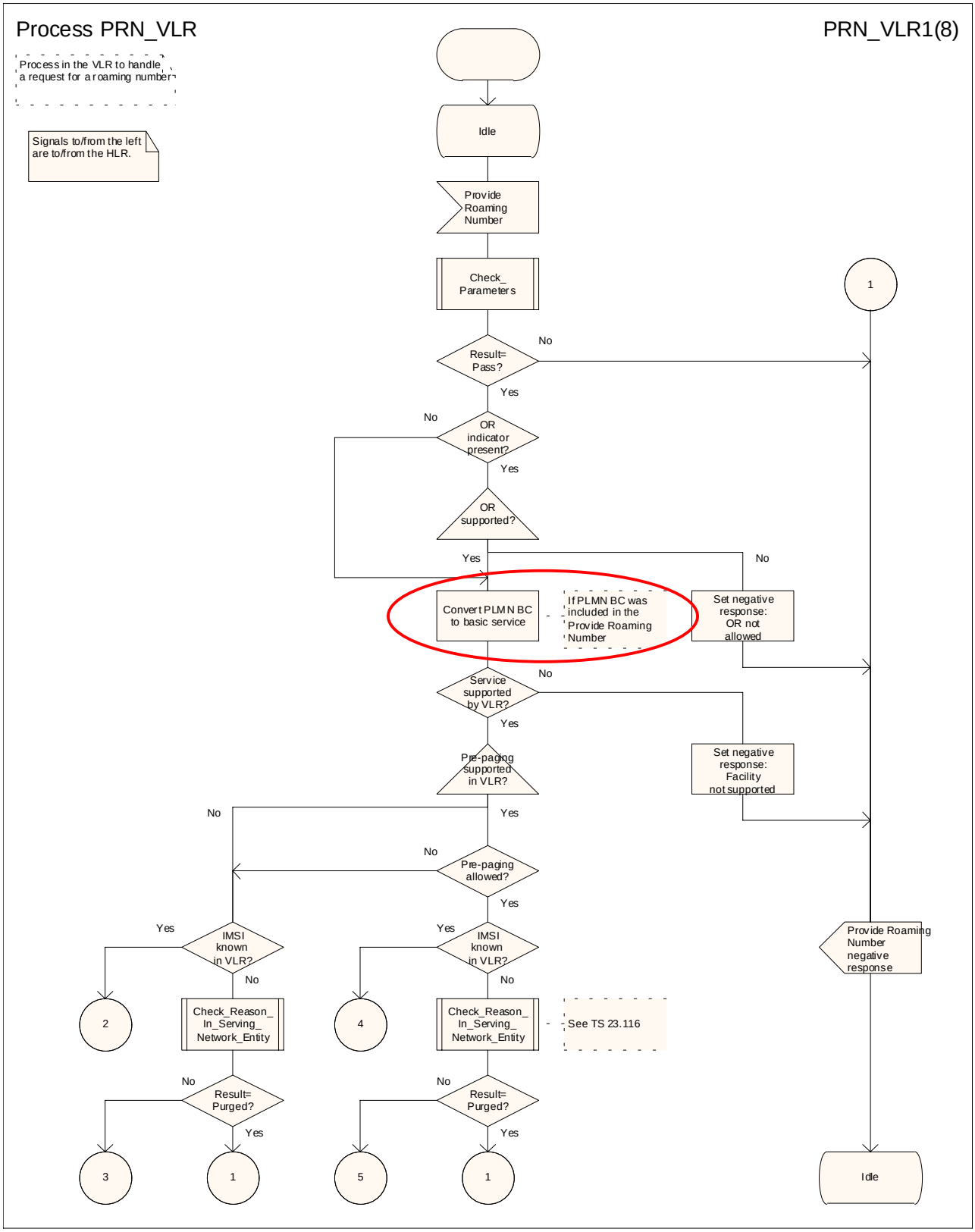
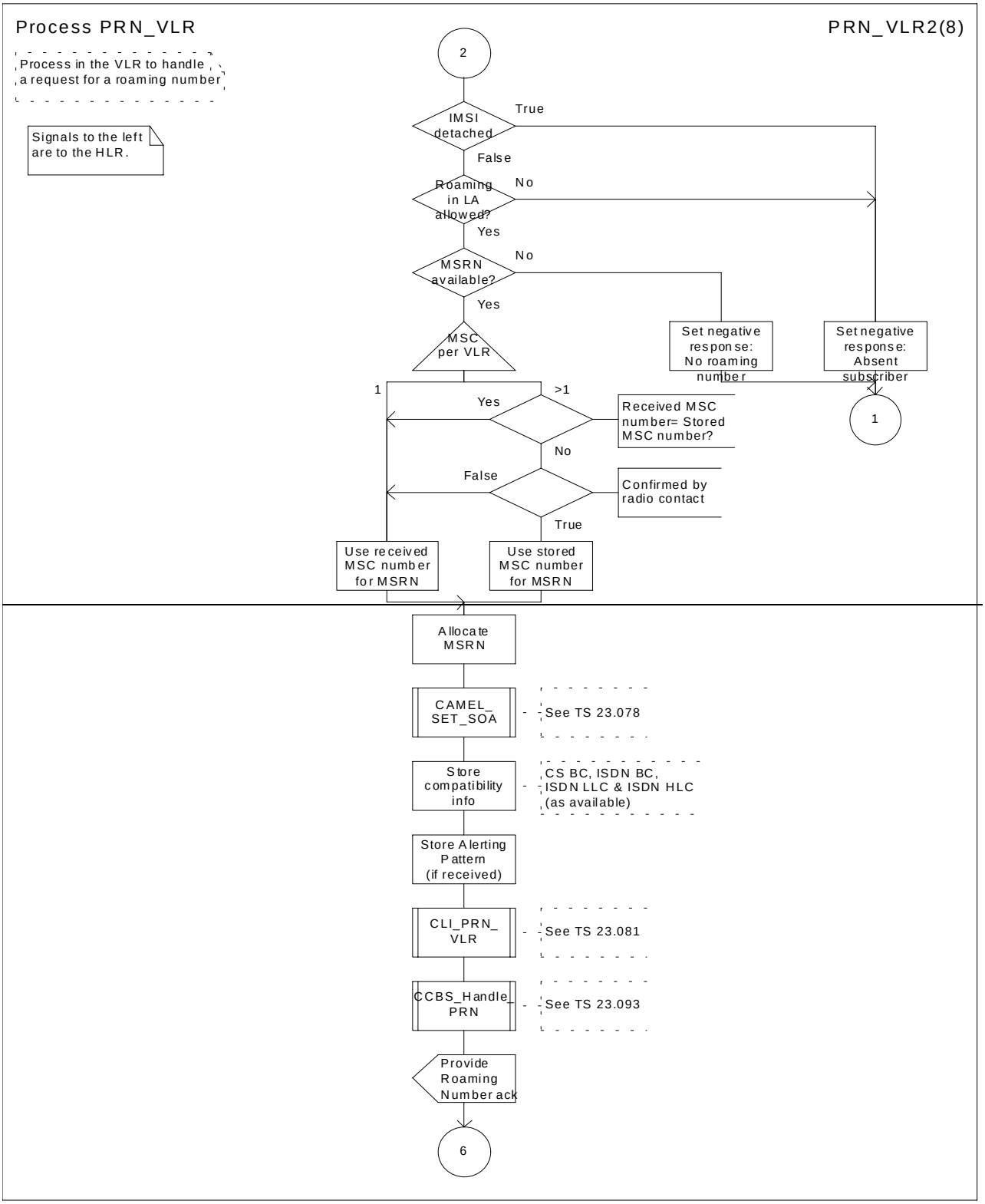


Figure 55a: Process PRN_VLR (sheet 1)


Process PRN_VLR
PRN_VLR2(8)
Process in the VLR to handle a request for a roaming number
Signals to the left are to the HLR.
2
IMSI detached
True
False
Roaming in LA allowed?
No
Yes
MSRN available?
No
Yes
MSC per VLR
Set negative response: No roaming number
Set negative response: Absent subscriber
1
>1
Yes
Received MSC number= Stored MSC number?
No
False
Confirmed by radio contact
True
Use received MSC number for MSRN
Use stored MSC number for MSRN
Allocate MSRN
CAMEL_ SET_SOA
See TS 23.078
Store compatibility info
CS BC, ISDN BC, ISDN LLC & ISDN HLC (as available)
Store Alerting Pattern (if received)
CLI_PRN_ VLR
See TS 23.081
CCBS_Handle_ PRN
See TS 23.093
Provide Roaming Number ack
6

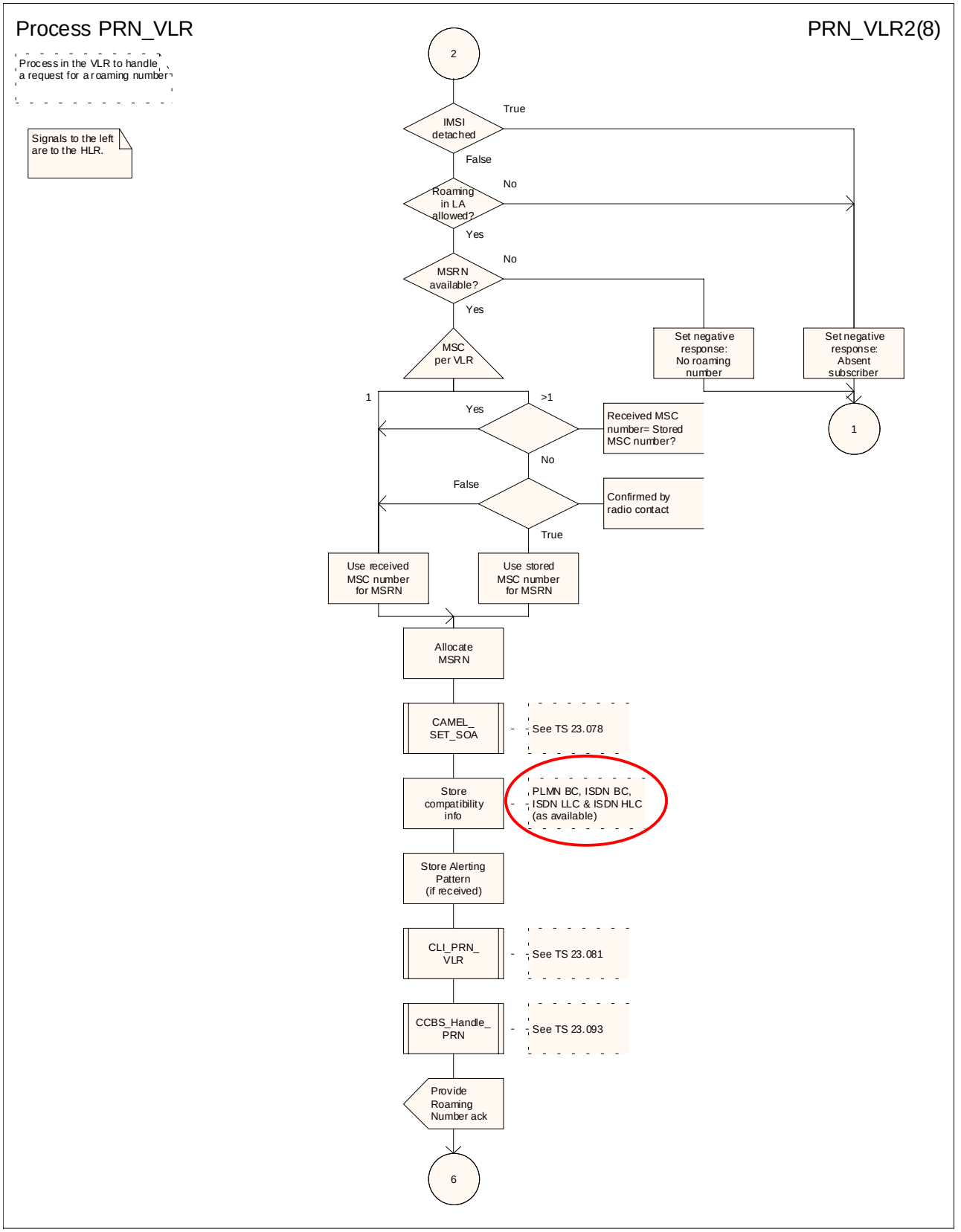
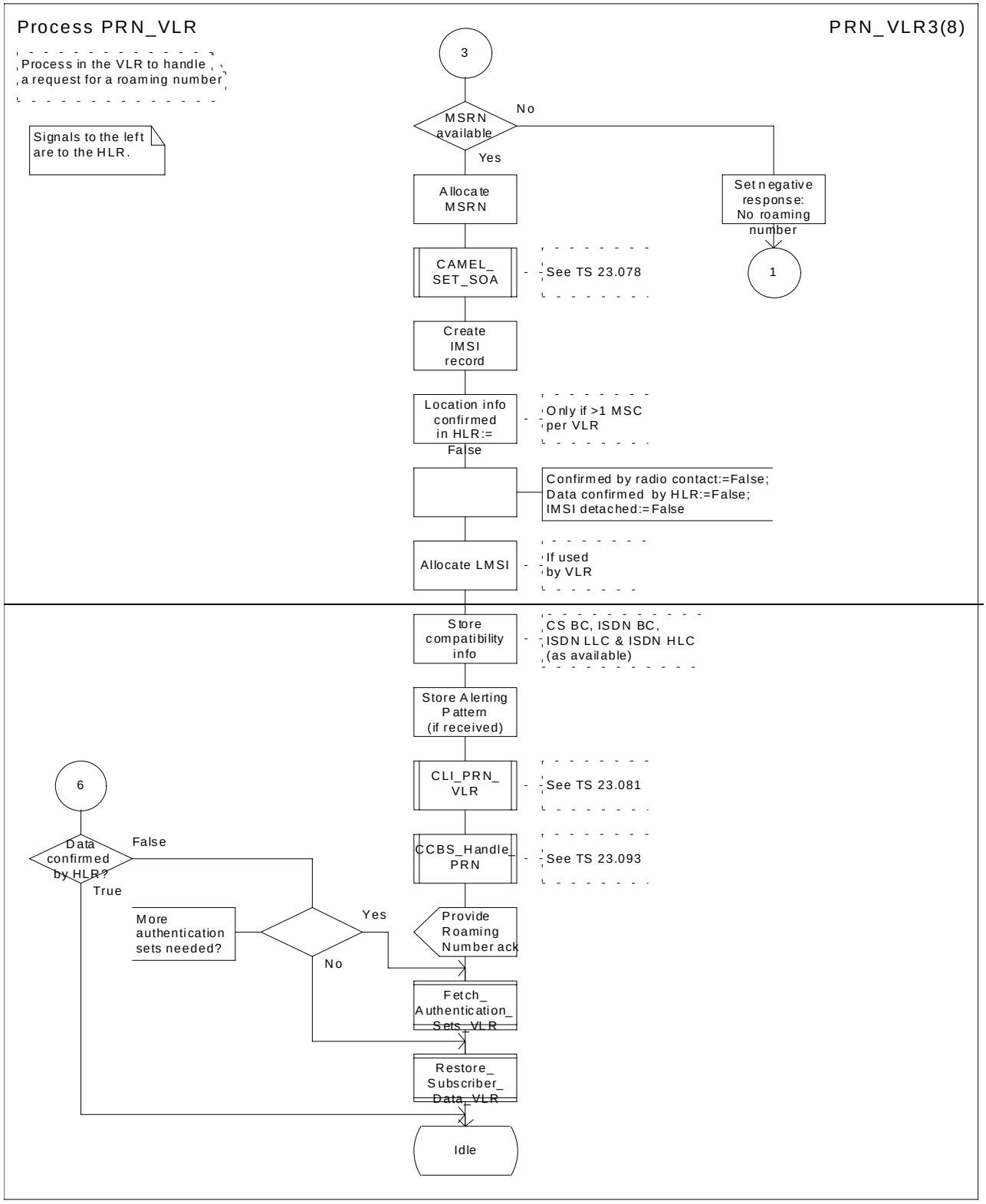


Figure 55b: Process PRN_VLR (sheet 2)


Process PRN_VLR
PRN_VLR3(8)
Process in the VLR to handle a request for a roaming number
Signals to the left are to the HLR.
3
MSRN available
No
Yes
Allocate MSRN
Set negative response: No roaming number
1
CAMEL_ SET_SOA
See TS 23.078
Create IMSI record
Location info confirmed in HLR:= False
Only if >1 MSC per VLR
Confirmed by radio contact:=False; Data confirmed by HLR:=False; IMSI detached:=False
Allocate LMSI
If used by VLR
Store compatibility info
CS BC, ISDN BC, ISDN LLC & ISDN HLC (as available)
Store Alerting Pattern (if received)
CLI_PRN_ VLR
See TS 23.081
CCBS_Handle_ PRN
See TS 23.093
6
Data confirmed by HLR?
False
True
More authentication sets needed?
Yes
No
Provide Roaming Number ack
Fetch_ Authentication_ Sets_VLR
Restore_ Subscriber_ Data_VLR
Idle

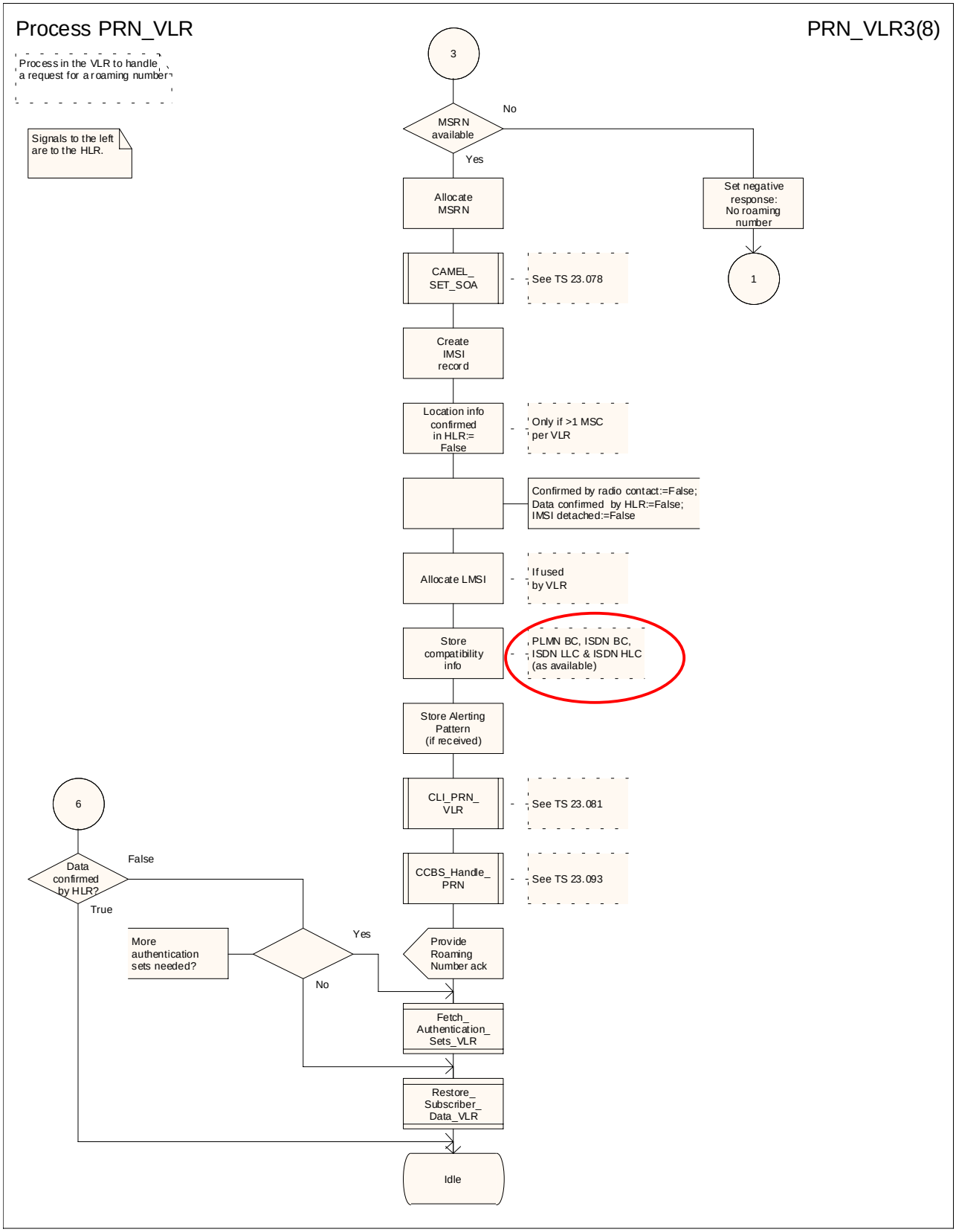


Figure 55c: Process PRN_VLR (sheet 3)

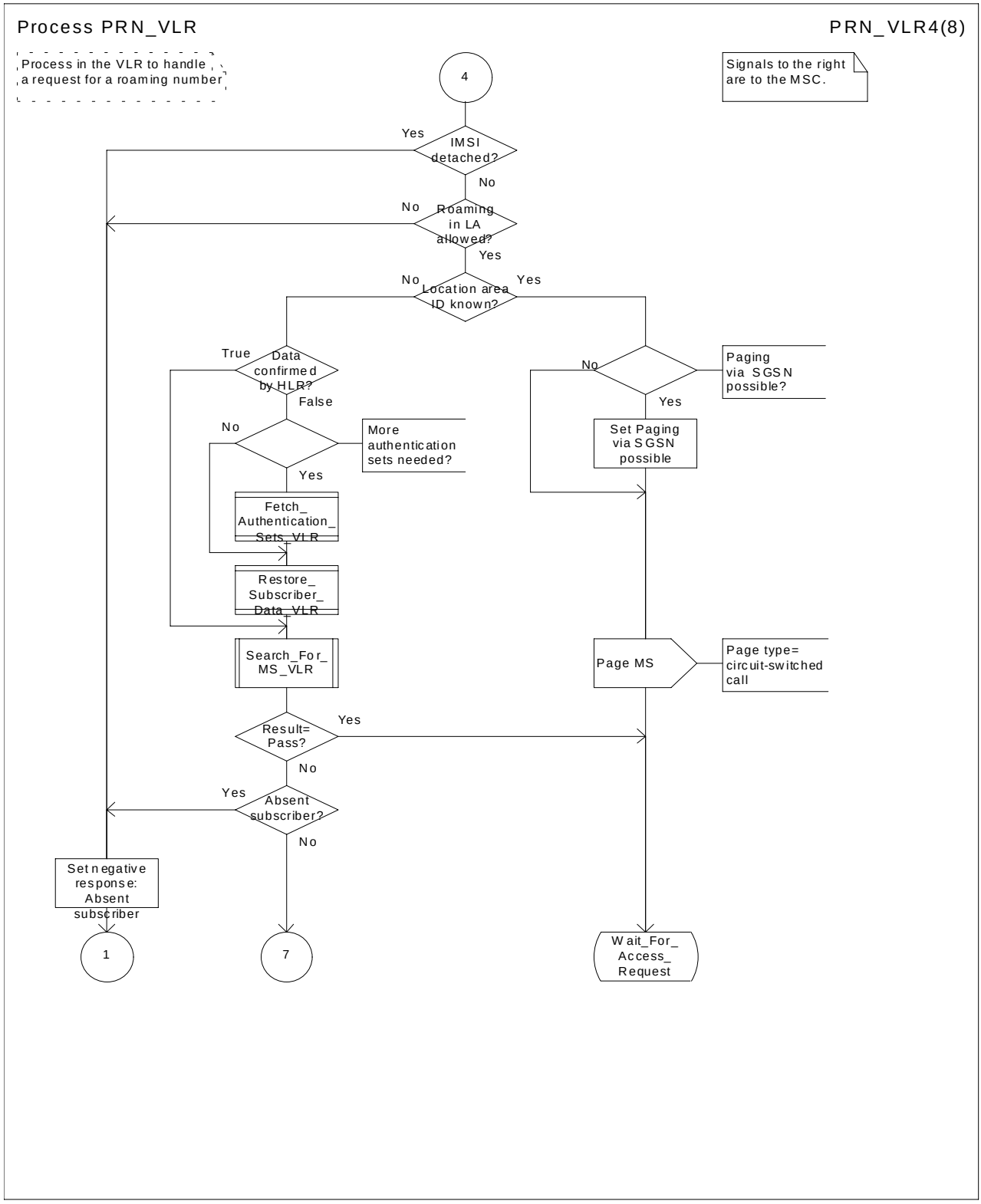
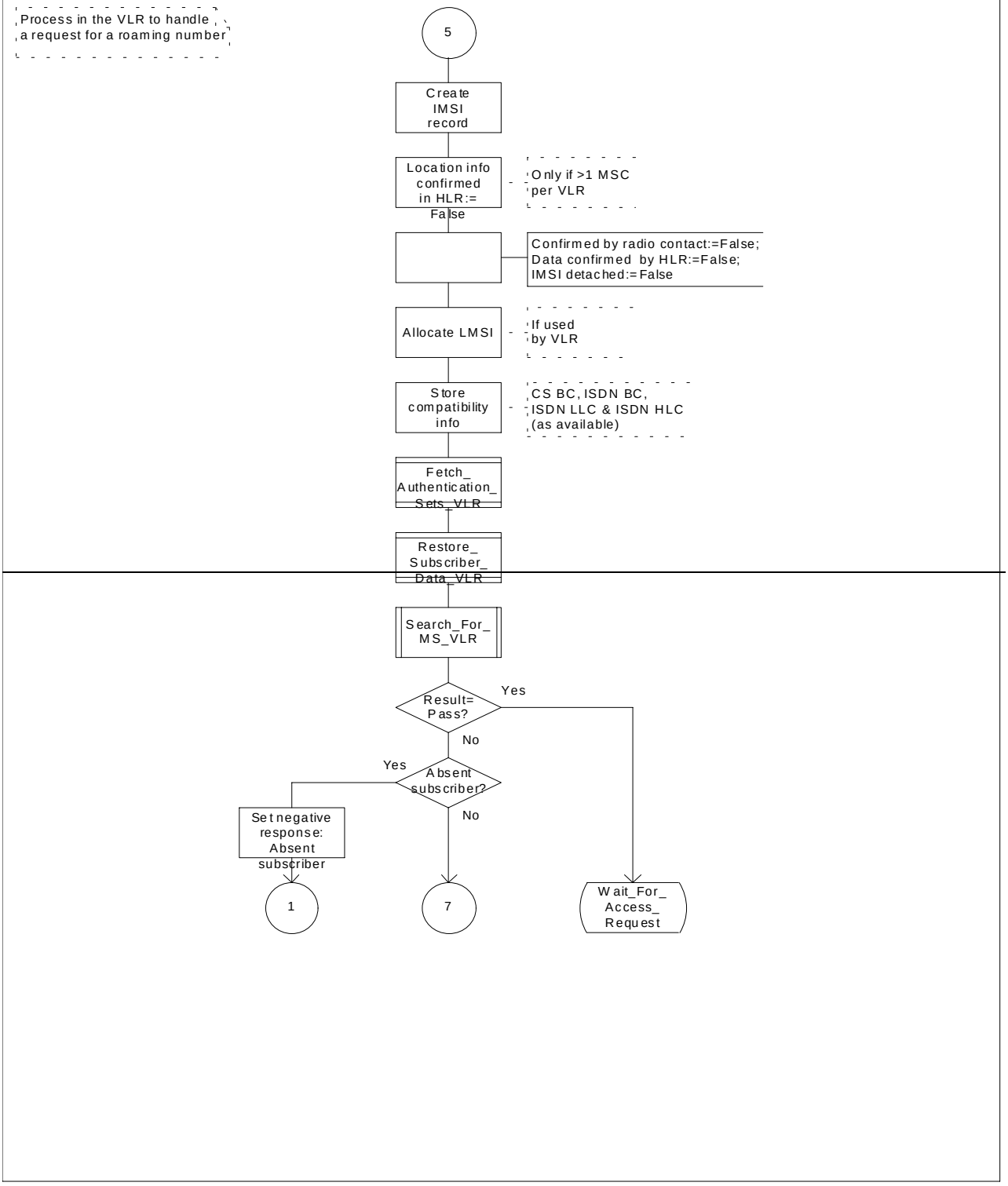


Figure 54d: Process PRN_VLR (sheet 4)

Process PRN_VLR PRN_VLR5(8)



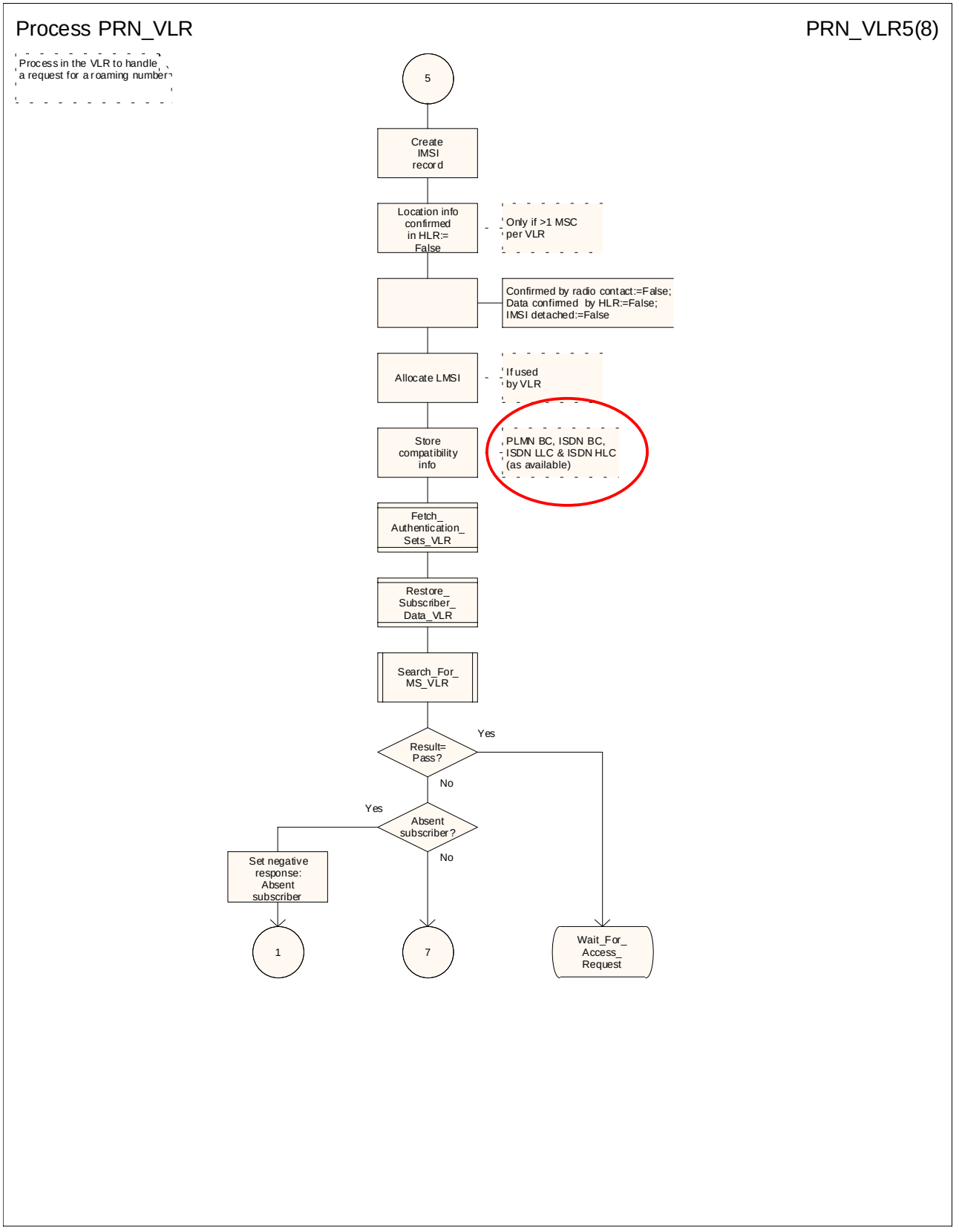
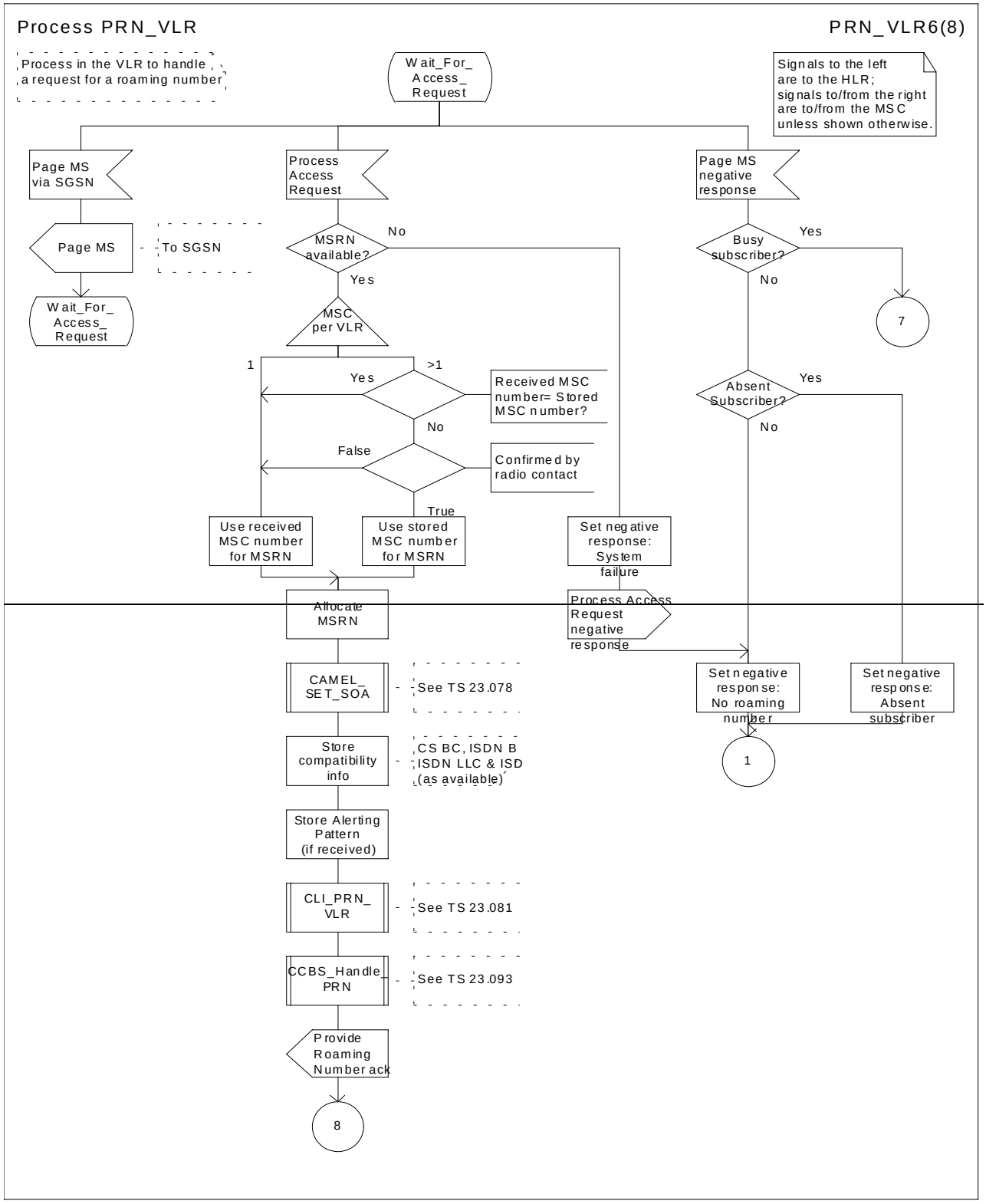


Figure 54e: Process PRN_VLR (sheet 5)


Process PRN_VLR
PRN_VLR6(8)
Process in the VLR to handle a request for a roaming number
Wait_For_ Access_ Request
Signals to the left are to the HLR; signals to/from the right are to/from the MSC unless shown otherwise.
Page MS via SGSN
Process Access Request
Page MS negative response
Page MS
To SGSN
MSRN available?
No
Yes
Busy subscriber?
Yes
No
Wait_For_ Access_ Request
MSC per VLR
7
1
>1
Yes
Received MSC number= Stored MSC number?
Absent Subscriber?
Yes
No
No
False
Confirmed by radio contact
True
Use received MSC number for MSRN
Use stored MSC number for MSRN
Set negative response: System failure
Allocate MSRN
Process Access Request negative response
CAMEL_ SET_SOA
See TS 23.078
Set negative response: No roaming number
Set negative response: Absent subscriber
Store compatibility info
CS BC, ISDN B ISDN LLC & ISD (as available)
1
Store Alerting Pattern (if received)
CLI_PRN_ VLR
See TS 23.081
CCBS_Handle_ PRN
See TS 23.093
Provide Roaming Number ack
8

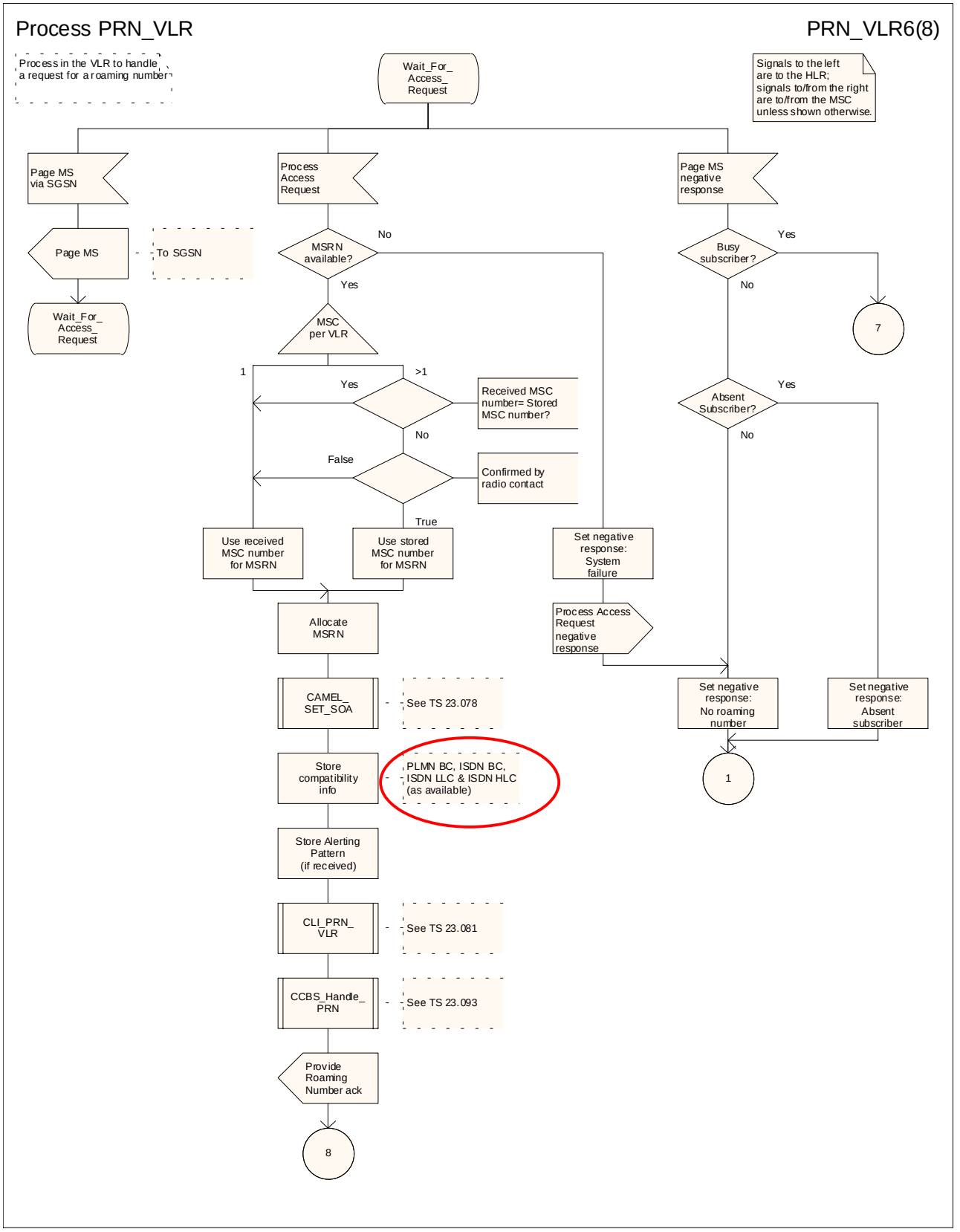
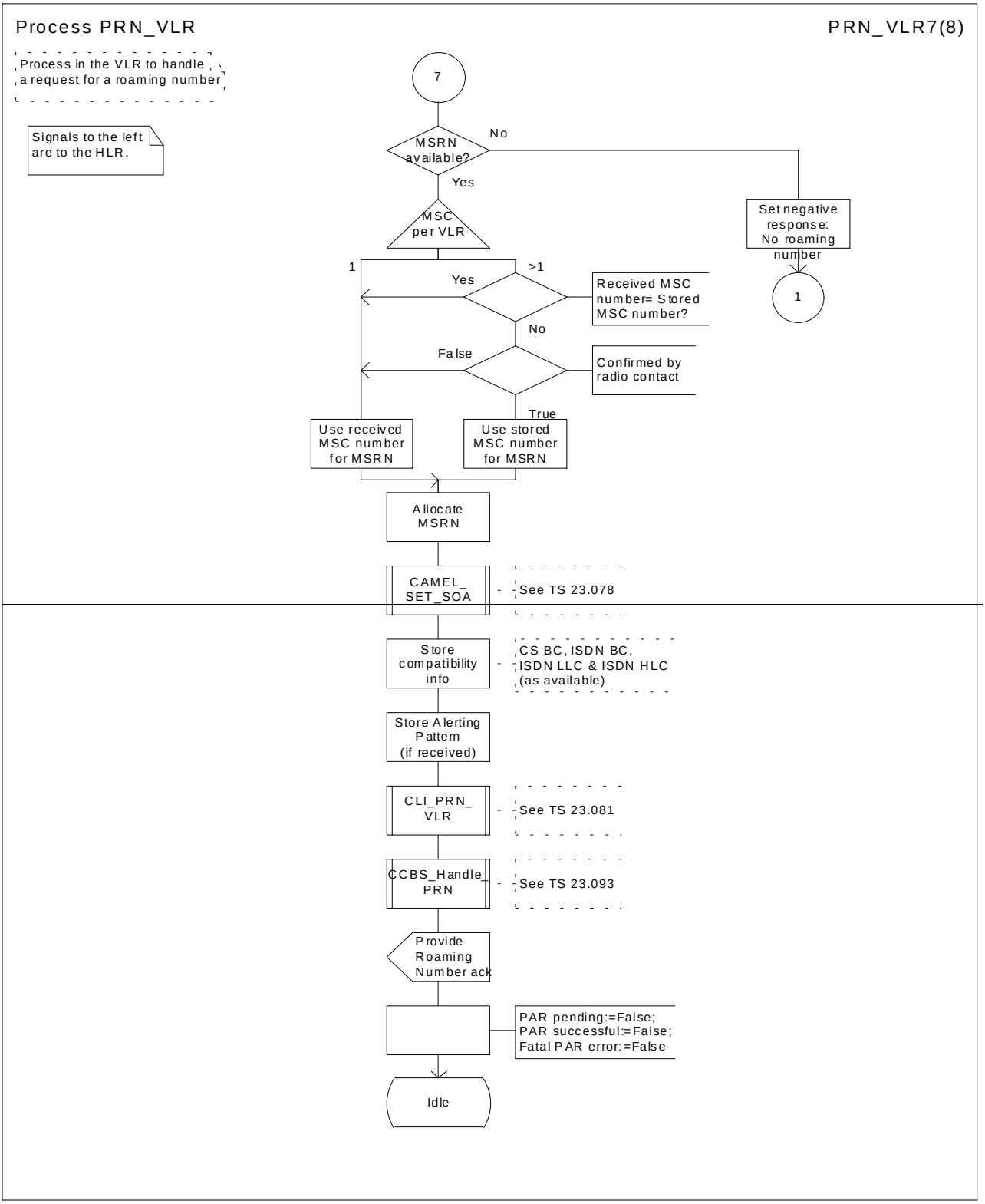


Figure 54f: Process PRN_VLR (sheet 6)


Process PRN_VLR
PRN_VLR7(8)
Process in the VLR to handle a request for a roaming number
Signals to the left are to the HLR.
7
MSRN available?
No
Yes
Set negative response: No roaming number
1
MSC per VLR
1
>1
Yes
Received MSC number= Stored MSC number?
No
False
Confirmed by radio contact
True
Use received MSC number for MSRN
Use stored MSC number for MSRN
Allocate MSRN
CAMEL_ SET_SOA
See TS 23.078
Store compatibility info
CS BC, ISDN BC, ISDN LLC & ISDN HLC (as available)
Store Alerting Pattern (if received)
CLI_PRN_ VLR
See TS 23.081
CCBS_Handle_ PRN
See TS 23.093
Provide Roaming Number ack
PAR pending:=False;
PAR successful:=False;
Fatal PAR error:=False
Idle

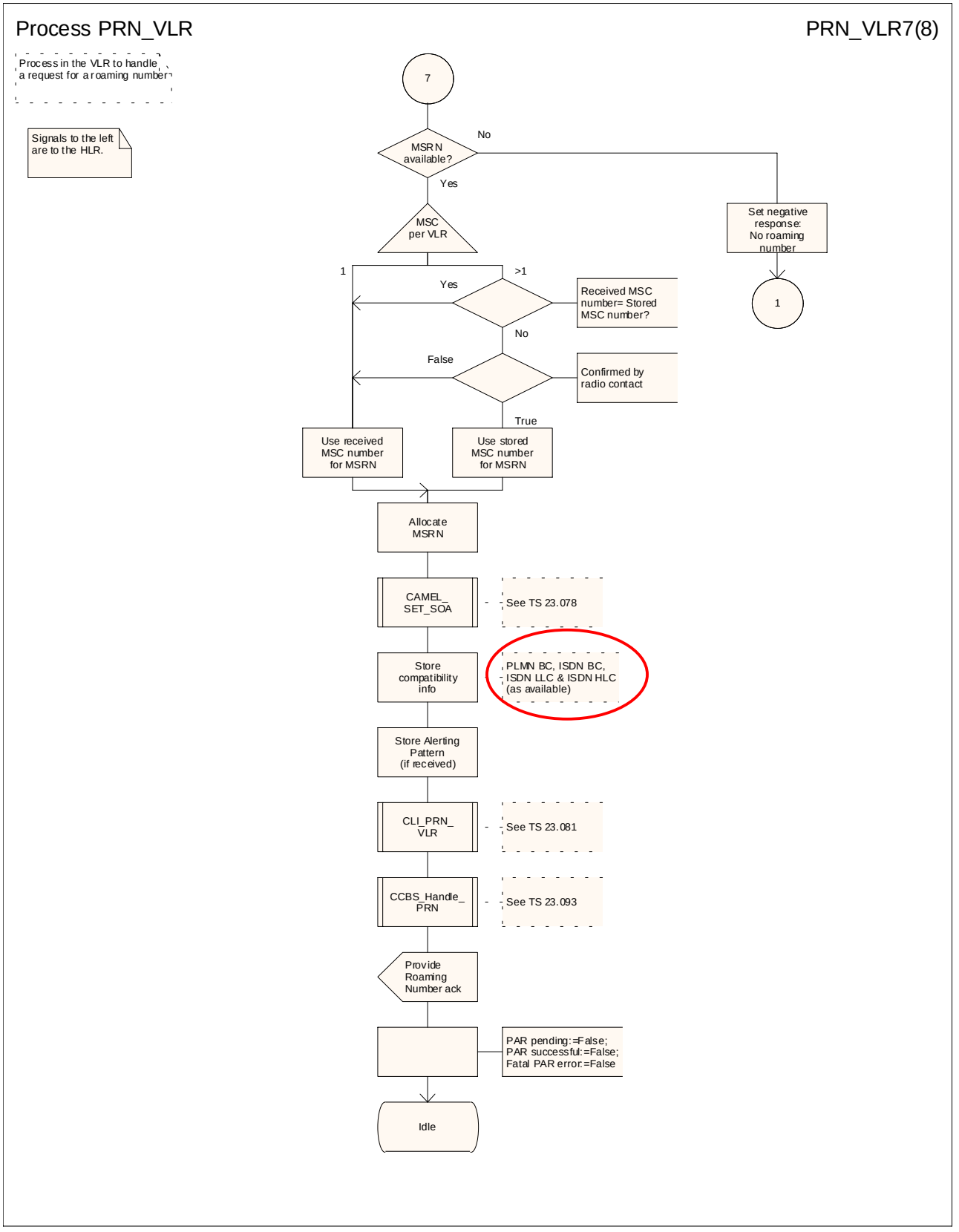


Figure 54g: Process PRN_VLR (sheet 7)

****** Next modified section ******

7.3.1.4 Procedure Complete_Call_In_MSC

Sheet 1: the procedure Set_CLIP_Info_MSC is specific to CLIP.

Sheet 1: the VMSC derives the PLMN bearer capability required for the call according to the rules defined in 3GPP TS 29.007 [30].

Sheet 1, sheet 2: the VMSC and the MS may negotiate the bearer capability to be used for the call by the exchange of information in the Set-up and Call Confirmed messages.

Sheet 1: the procedure UUS_ICH_UUS1_Implicit_Active is specific to UUS, it is specified in 3GPP TS 23.087 [20].

Sheet 1: the procedure CCBS_Report_Not_Idle is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 2: the procedure Establish_Terminating_TCH_Multicall is specific to Multicall; it is specified in 3GPP TS 23.135 [25].

Sheet 2: the test "Result=Rejected?" can take the "Yes" exit only if the procedure Establish_Terminating_TCH_Multicall was called.

Sheet 2, sheet 3, sheet 4, sheet 5, sheet 6, sheet 7: the procedure CAMEL_MT_GMSC_DISC4 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "No" exit of the test "Result=Reconnect?".

Sheet 2, sheet 3, sheet 6, sheet 9, sheet 10: the procedure CAMEL_MT_GMSC_DISC6 is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 2, sheet 5, sheet 9: the procedure CCBS_ICH_MSC_Report_Failure is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 3, sheet 5: the procedure CCBS_ICH_MSC_Report_Success is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 3: the procedure CAMEL_Start_TNRy is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 3: the procedure CAMEL_MT_MSC_ALERTING is specific to CAMEL phase 4 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 4 or later, processing continues from the "Pass" exit of the test "Result?".

Sheet 3, sheet 6: the procedure UUS_ICH_Check_Support is specific to UUS, it is specified in 3GPP TS 23.087 [20]. If the VMSC does not support UUS, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 3: the task "UTU2Cnt:=0" is executed only if the VMSC supports UUS.

Sheet 3: the procedure Send_ACM_If_Required is specified in subclause 7.2.1.3.

Sheet 3, sheet 6: the procedure Establish_Terminating_TCH_Multicall is specific to Multicall; it is specified in 3GPP TS 23.135 [25]. If the VMSC does not support Multicall, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 4, sheet 7: the procedure Handle_AoC_MT_MSC is specific to AoC. If the VMSC does not support AoC, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 4, sheet 7: the procedure CAMEL_MT_GMSC_ANSWER is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VMSC does not support CAMEL phase 3 or later, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 4, sheet 7: the procedure Set_COL_Presentation_Indicator_MSC is specific to COLP.

Sheet 4: the procedure Send_Network_Connect_If_Required is specified in subclause 7.2.1.5.

Sheet 5, sheet 11: the processing in the branch starting with the input "CD Request" is specific to Call Deflection; if the VMSC does not support Call Deflection the input is discarded.

Sheet 5, sheet 11: the procedure Handling_CD_MSC is specific to Call Deflection; it is specified in 3GPP TS 23.072 [11].

Sheet 6: the procedure CAMEL_Stop_TNRy is called if the VMSC supports CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12].

Sheet 7: the procedure Send_Answer_If_Required is specified in subclause 7.2.1.4.

Sheet 8: the input signal "CAMEL TNRy expired" will be received only if the VMSC supports CAMEL phase 3 or later.

Sheet 8, sheet 11: the procedure UUS_ICH_Check_Forwarding is specific to UUS, it is specified in 3GPP TS 23.087 [20]. If the VMSC does not support UUS, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 9, sheet 10: the procedure UUS_MSC_Check_UUS1_UUI is specific to UUS; it is specified in 3GPP TS 23.087 [20].

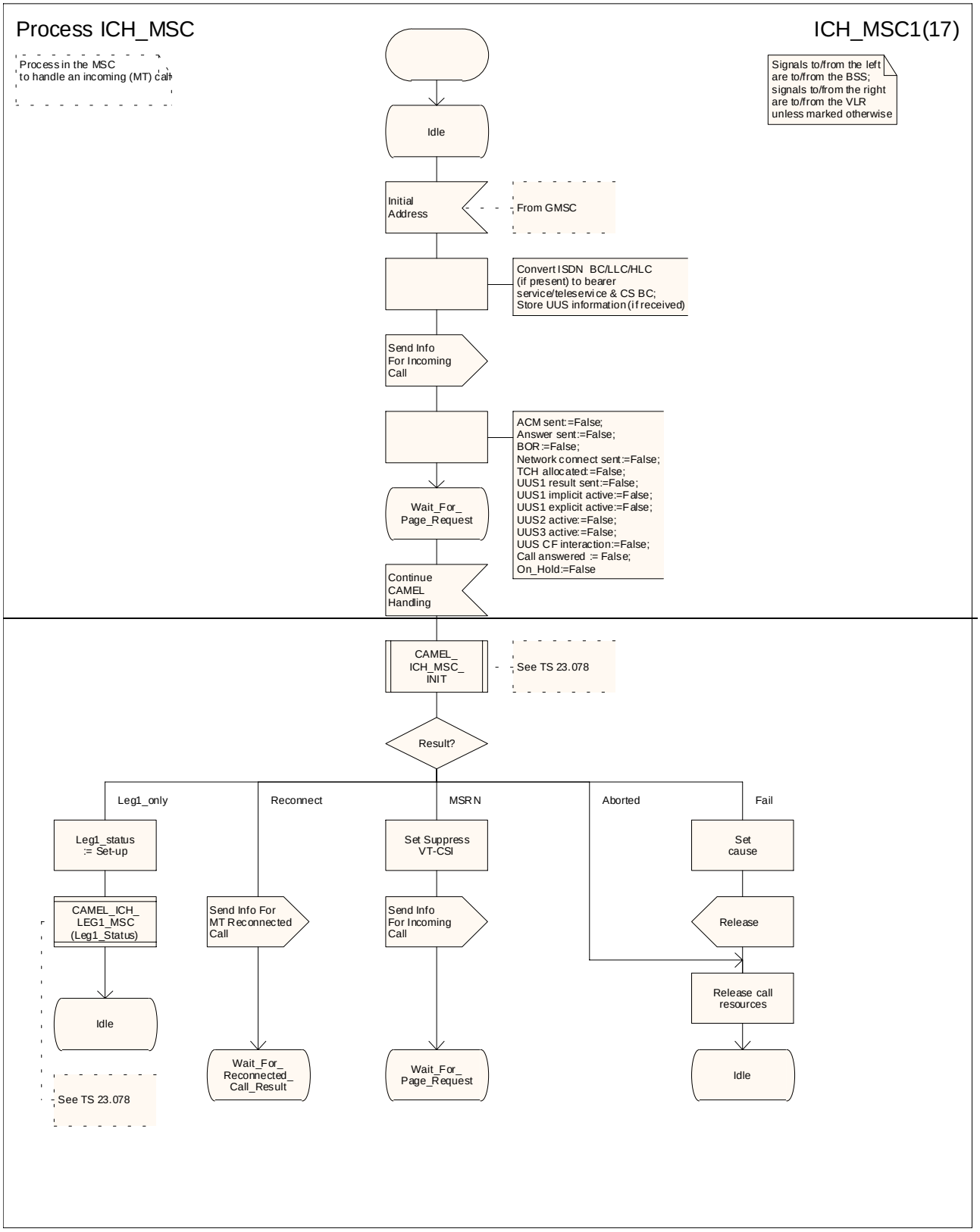
Sheet 11: the procedures UUS_MSC_Check_UUS2_UUI_to MS and UUS_MSC_Check_UUS2_UUI_to NW are specific to UUS, they are specified in 3GPP TS 23.087 [20].

Sheet 11: the procedure CD_UUS_Interaction is specific to Call Deflection; it is specified in 3GPP TS 23.072 [11].

****** Next modified section ******

7.3.1.7 ~~Procedure Derive_GSM_BC_MSC~~Void

****** Next modified section ******


Process ICH_MSC
ICH_MSC1(17)
Process in the MSC to handle an incoming (MT) call
Signals to/from the left are to/from the BSS; signals to/from the right are to/from the VLR unless marked otherwise
Idle
Initial Address
From GMSC
Convert ISDN BC/LLC/HLC (if present) to bearer service/teleservice & CS BC; Store UUS information (if received)
Send Info For Incoming Call
ACM sent:=False; Answer sent:=False; BOR:=False; Network connect sent:=False; TCH allocated:=False; UUS1 result sent:=False; UUS1 implicit active:=False; UUS1 explicit active:=False; UUS2 active:=False; UUS3 active:=False; UUS CF interaction:=False; Call answered := False; On_Hold:=False
Wait_For_ Page_Request
Continue CAMEL Handling
CAMEL_ ICH_MSC_ INIT
See TS 23.078
Result?
Leg1_only
Reconnect
MSRN
Aborted
Fail
Leg1_status := Set-up
CAMEL_ICH_ LEG1_MSC (Leg1_Status)
Idle
See TS 23.078
Send Info For MT Reconnected Call
Wait_For_ Reconnected_ Call_Result
Set Suppress VT-CSI
Send Info For Incoming Call
Wait_For_ Page_Request
Set cause
Release
Release call resources
Idle

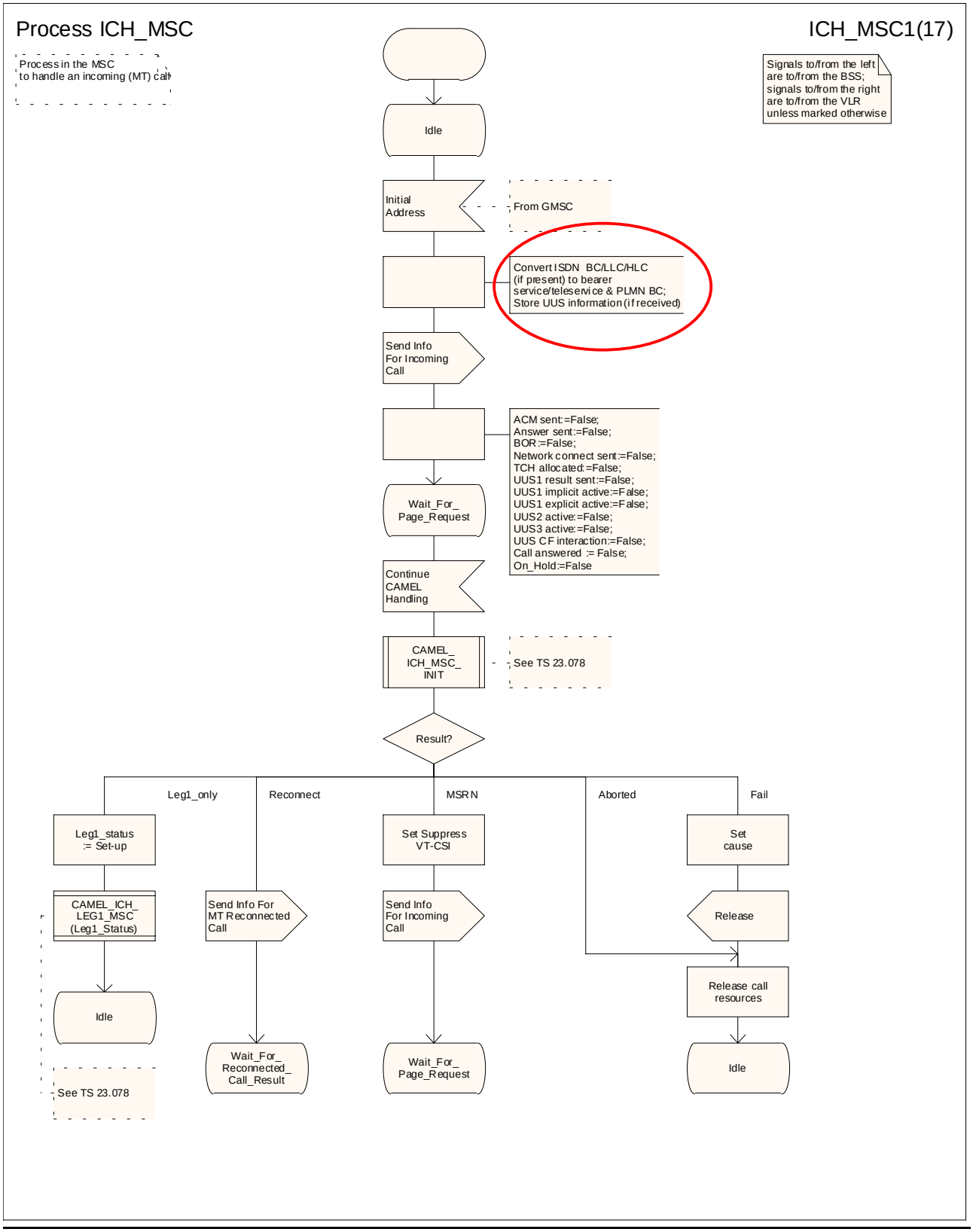
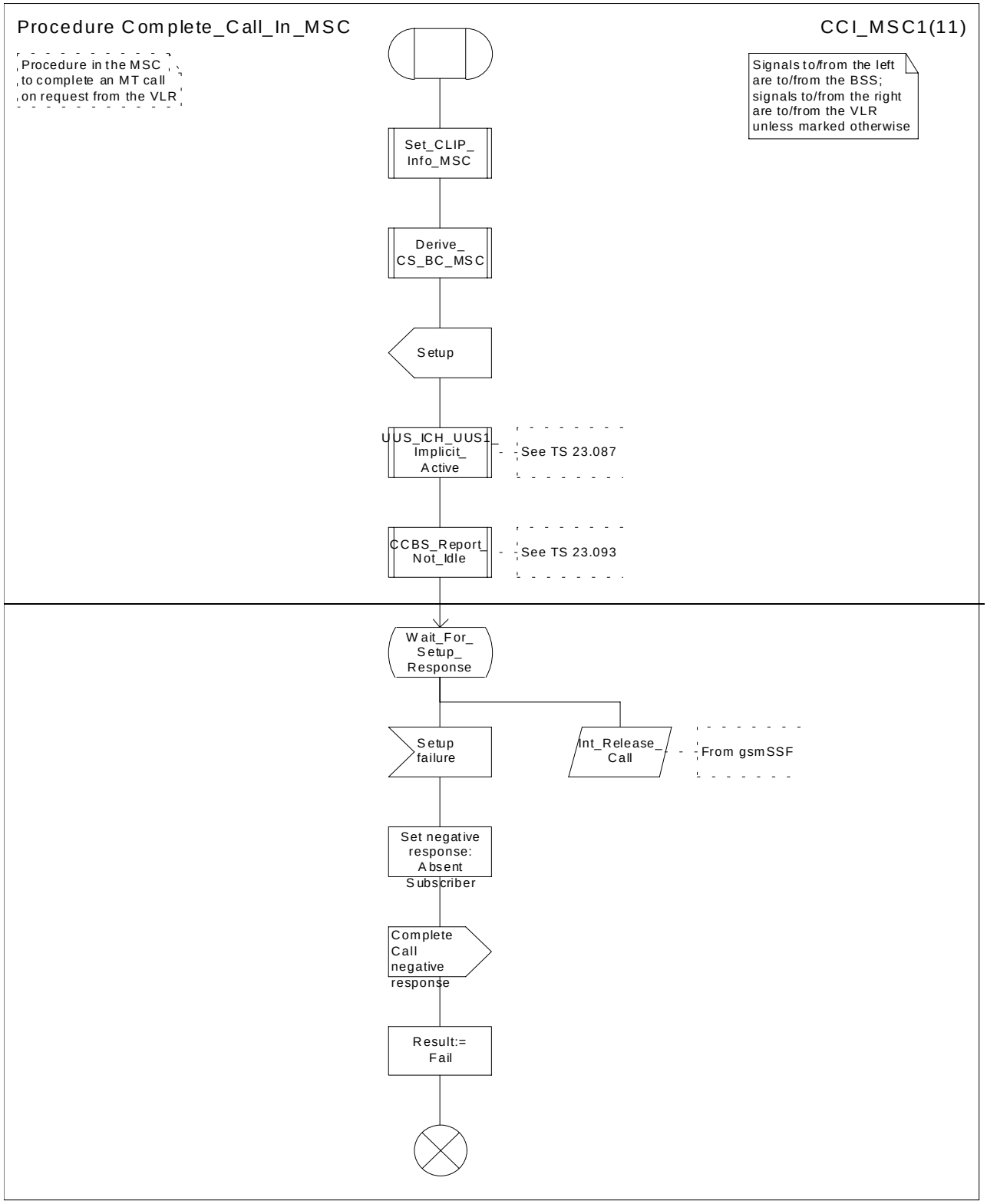


Figure 67a: Process ICH_MSC (sheet 1)

****** Next modified section ******


Procedure Complete_Call_In_MSC
CCI_MSC1(11)
Procedure in the MSC to complete an MT call on request from the VLR
Signals to/from the left are to/from the BSS; signals to/from the right are to/from the VLR unless marked otherwise
Set_CLIP_ Info_MSC
Derive_ CS_BC_MSC
Setup
UUS_ICH_UUS1_ Implicit_ Active
See TS 23.087
CCBS_Report_ Not_Idle
See TS 23.093
Wait_For_ Setup_ Response
Setup failure
Int_Release_ Call
From gsmSSF
Set negative response: Absent Subscriber
Complete Call negative response
Result:= Fail

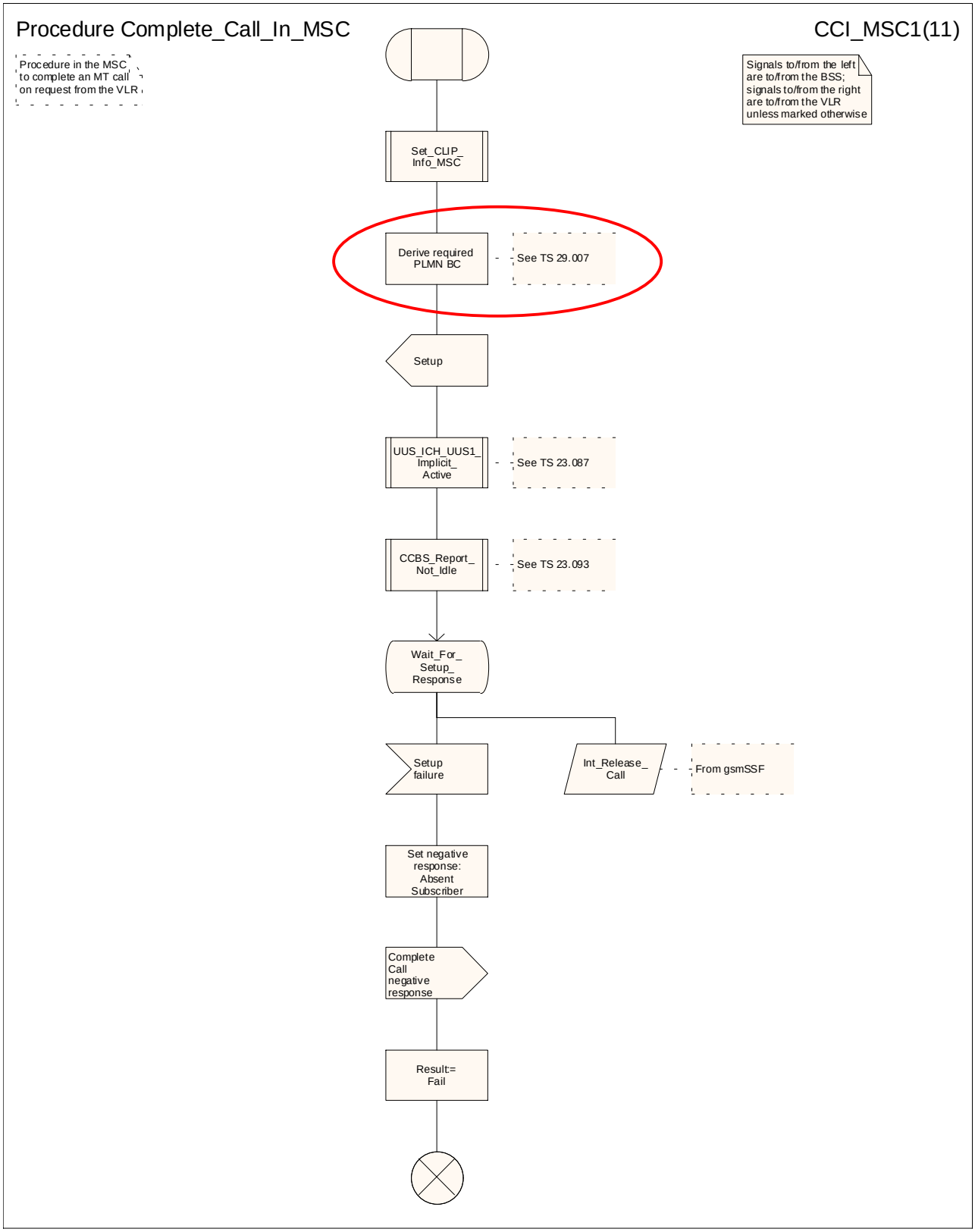


Figure 70a: Procedure Complete_Call_In_MSC (sheet 1)

****** Next modified section ******

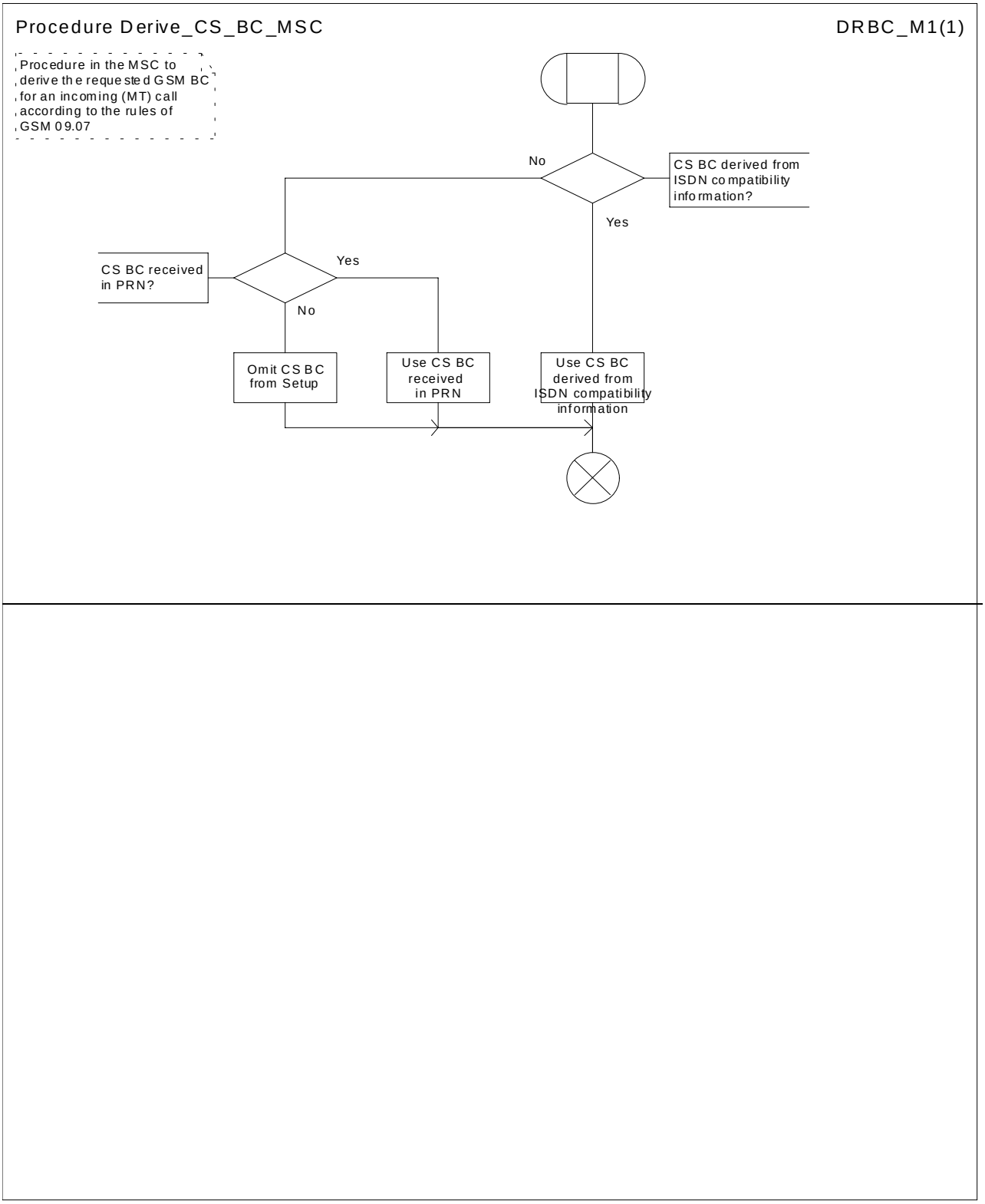


Figure 72: ~~Derive_CS_BC_MSC~~Void

**** Next modified section ****

7.3.2.1 Process ICH_VLR

Sheet 1: if the MSRN received in the Send Info For Incoming Call is not allocated or there is no IMSI record for the IMSI identified by the MSRN, this is treated as an unknown MSRN.

Sheet 1: the procedure CAMEL_ICH_VLR is specific to CAMEL phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the VLR does not support CAMEL phase 3 or later, processing continues from the possible call of the procedure CCBS_ICH_Set_CCBS_Call_Indicator.

Sheet 1: the procedure CCBS_ICH_Set_CCBS_Call_Indicator is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 1: the VLR derives the basic service required for the call according to the rules defined in 3GPP TS 29.007 [30].

Sheet 1, sheet 2, sheet 5: the procedure CCBS_ICH_VLR_Report_Failure is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 1, sheet 3: the procedure CCBS_ICH_Report_Not_Reachable is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 2: this process communicates with the matching instance of the process PRN_VLR, which is linked by the MSRN.

Sheet 2: the test "Paging via SGSN possible" takes the "yes" exit if:

- the Gs interface is implemented; and
- there is an association established for the MS between the MSC/VLR and the SGSN.

Sheet 3: the test "NDUB?" takes the "Yes" exit if the Page MS negative response or the Search for MS negative response had the value Busy Subscriber (NDUB).

Sheet 3: the procedure Get_CW_Subscription_Info_VLR is specific to Call Waiting. If the VLR does not support Call Waiting, processing continues from the "No" exit of the test "CW available?".

Sheet 3: the procedure Get_CW_Subscription_Info_Multicall_VLR is specific to Multicall; it is specified in 3GPP TS 23.135 [34]. If the VLR does not support both Multicall and Call Waiting, processing continues from the "No" exit of the test "CW available?".

Sheet 3: the VLR uses the basic service returned in the Page MS negative response or the Search for MS negative response Busy Subscriber (More calls possible) to determine whether call waiting is available.

Sheet 3: the procedure Get_LI_Subscription_Info_MT_VLR is specific to CLIP and COLR. If the VLR supports neither CLIP nor COLR, the procedure call is omitted.

Sheet3: the procedure Get_AoC_Subscription_Info_VLR is specific to AoC; it is specified in subclause 7.1.2.15.

Sheet 3 sheet 6: the procedure CLI_ICH_VLR_Add_CLI is specific to Enhanced CLI Handling. It is specified in 3GPP TS 23.081 [14].

Sheet 3: the procedure CCBS_ICH_Handle_NDUB is specific to CCBS; it is specified in 3GPP TS 23.093 [23]. If the VLR does not support CCBS, processing continues from the "Forward" exit of the test "Result".

Sheet 3: the procedure Process_Access_Request_VLR is specified in subclause 7.1.2.2.

Sheet 3: the output signal Page MS towards the SGSN includes the Location area identity parameter.

Sheet 3: if the VLR does not support CUG, handling continues from the "No" exit of the test "CUG info present?".

Sheet 4, sheet 6: the procedure CAMEL_CHECK_SII2_CDTI is specific to CAMEL Phase 3 or later; it is specified in 3GPP TS 23.078 [12]. If the GMSC does not support CAMEL Phase 3 or later, processing continues from the "Yes" exit of the test "Result = Pass?".

Sheet 5, sheet 6: the procedure CD_Authorization is specific to Call Deflection, it is specified in 3GPP TS 23.072 [11]. If the VLR does not support Call Deflection, processing continues from the "Yes" exit of the test "Result=Aborted?".

Sheet 5, sheet 6: the procedure CCBS_ICH_Handle_UDUB is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 6: the test "NDUB?" is executed only if the VLR supports CCBS. If the VLR does not support CCBS, processing continues from connector 5.

Sheet 7: the procedure CCBS_ICH_Set_CCBS_Target is specific to CCBS; it is specified in 3GPP TS 23.093 [23].

Sheet 7: the procedure Handle_CFNRc is specified in subclause 7.2.2.11.

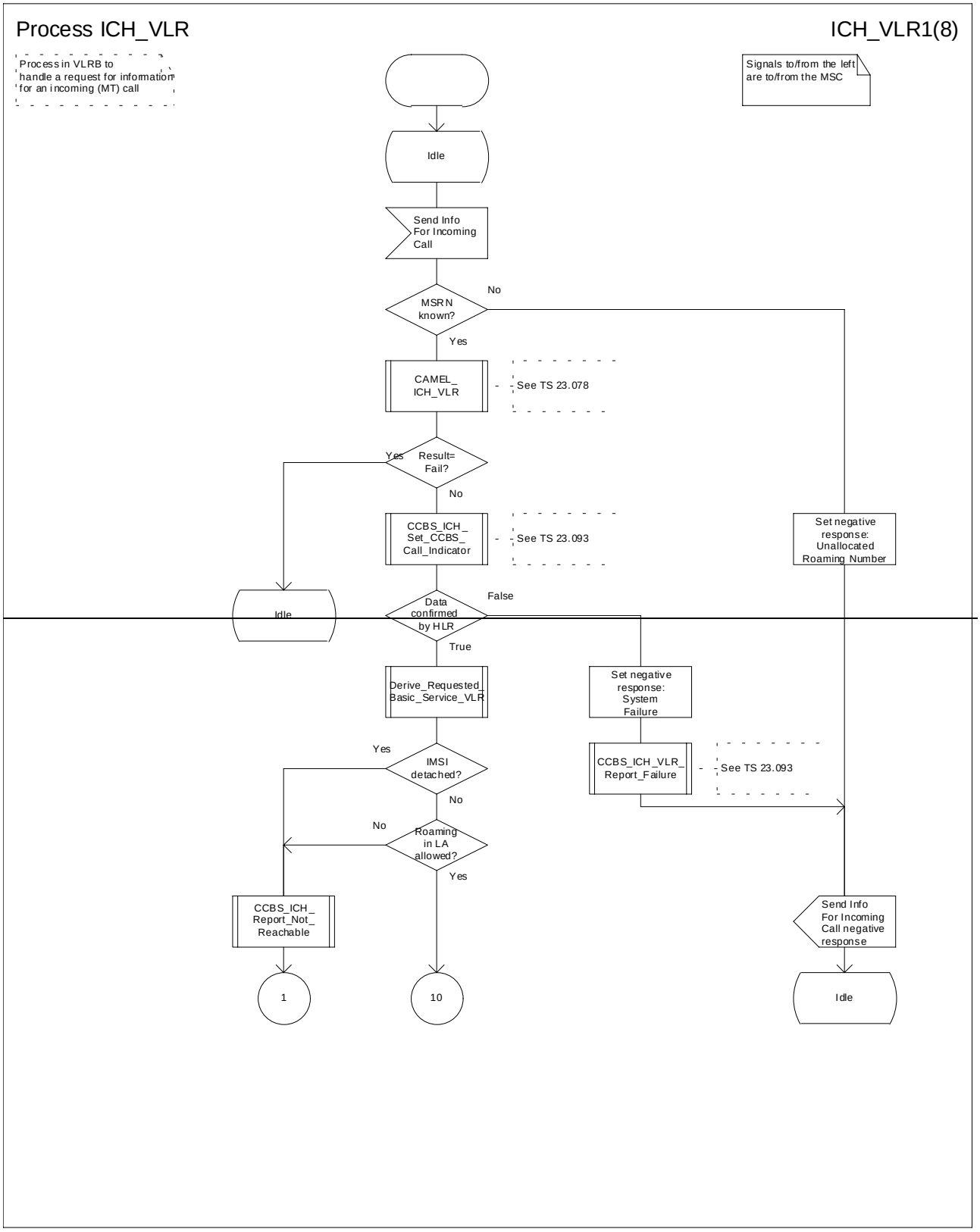
Sheet 8: the procedure Forward_CUG_Check is specific to CUG; it is specified in subclause 7.2.2.6. If the VLR does not support CUG, processing continues from the "Yes" exit of the test "Result=Call allowed?".

Sheet 8: the procedures CAMEL_O_CSI_Check_VLR, and CAMEL_D_CSI_Check_VLR are specific to CAMEL phase 3 or later; they are specified in 3GPP TS 23.078 [12].

7.3.2.2 ~~Procedure Derive_Requested_Basic_Service_VLR~~Void

~~If the VLR did not receive a basic service for the call in the Send Info For Incoming Call, and did not receive a GSM bearer capability in the Provide Roaming Number, it applies a default basic service according to the requirements of the operator.~~

****** Next modified section ******


Process ICH_VLR
ICH_VLR1(8)
Process in VLRB to handle a request for information for an incoming (MT) call
Signals to/from the left are to/from the MSC
Idle
Send Info For Incoming Call
MSRN known?
No
Yes
CAMEL_ ICH_VLR
See TS 23.078
Result= Fail?
Yes
No
CCBS_ICH_ Set_CCBS_ Call_Indicator
See TS 23.093
Set negative response: Unallocated Roaming Number
Idle
Data confirmed by HLR
False
True
Derive_Requested_ Basic_Service_VLR
Set negative response: System Failure
IMSI detached?
Yes
No
CCBS_ICH_VLR_ Report_Failure
See TS 23.093
Roaming in LA allowed?
No
Yes
CCBS_ICH_ Report_Not_ Reachable
Send Info For Incoming Call negative response
1
10
Idle

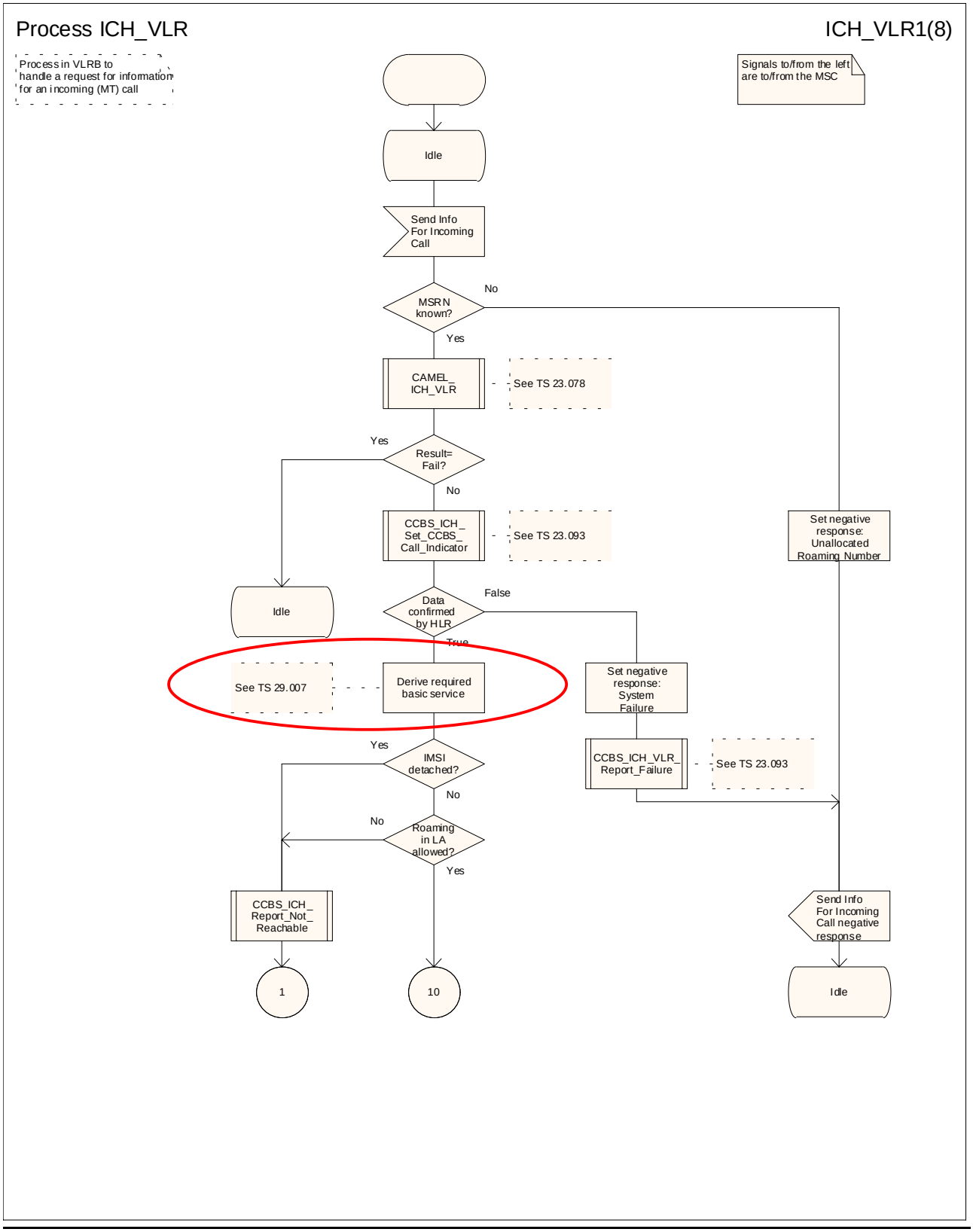


Figure 76a: Process ICH_VLR (sheet 1)

****** Next modified section ******

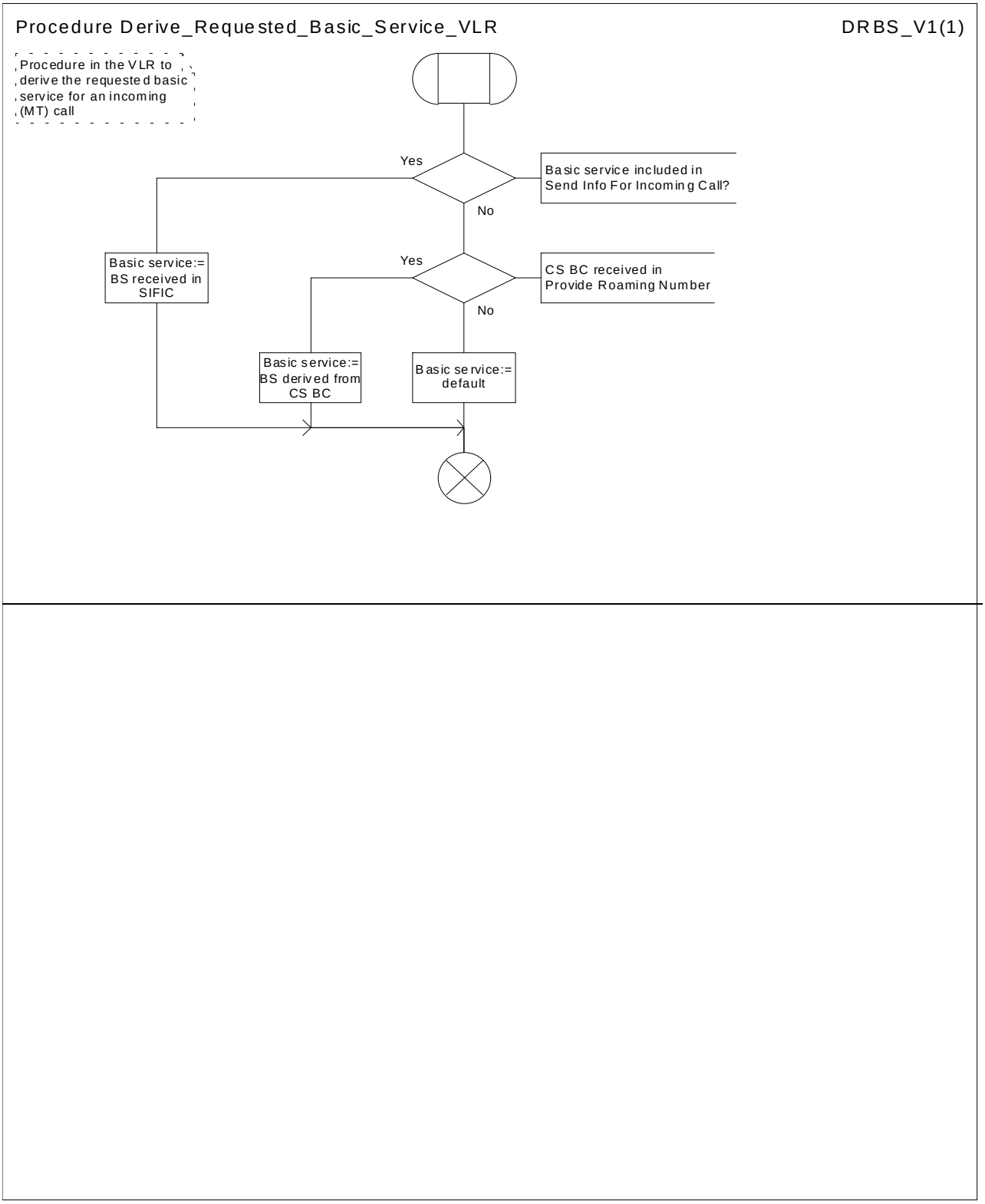


Figure 77: ~~Procedure Derive_Requested_Basic_Service_VLR~~Void

****** Next modified section ******

8.1.2 Authenticate

The following information elements are required for authentication of a UMTS ~~MS~~UE:

Information element name	Required	Description
RAND(I)	M	Random number challenge to be sent to the MS (3GPP TS 33.102 [32])
AUTN(I)	M	Authentication token to be sent to the MS (3GPP TS 33.102 [32])

The following information elements are required for authentication of a GSM MS:

Information element name	Required	Description
RAND	M	Random number challenge to be sent to the MS (3GPP TS 43.020 [1])
CKSN	M	Cipher key sequence number to be sent to the MS (3GPP TS 43.020 [1])

8.1.3 Authenticate ack

The following information element is required for authentication of a UMTS ~~MS~~UE:

Information element name	Required	Description
RES(I)	M	Result returned by the MS (3GPP TS 33.102 [32])

The following information element is required for authentication of a GSM MS:

Information element name	Required	Description
SRES	M	Signature result returned by the MS (3GPP TS 43.020 [1])

****** Next modified section ******

8.1.9 Complete Call

The following information elements are required:

Information element name	Required	Description
MSISDN	C	MSISDN of the MS for which the Complete Call is sent. Shall be present for an ordinary MO call, for an MT call and for an emergency call when the MS is registered in the VLR; otherwise shall be absent.
IMEI	C	IMEI of the mobile for which the Complete Call is sent. Shall be present for an emergency call when the mobile is identified only by its IMEI; otherwise shall be absent.
Category	C	Category of the MS for which the Complete Call is sent. Shall be present for an ordinary MO call and for an emergency call when the MS is registered in the VLR; otherwise shall be absent.
~~GSM~~ PLMN bearer capability	C	Shall be present for an MT call according to the rules defined in 3GPP TS 29.007 [30]~~if it was received in the Provide Roaming Number; otherwise shall be absent~~.
ISDN bearer capability	C	Shall be present for an MT call if it was received in the Provide Roaming Number; otherwise shall be absent.
ISDN low layer compatibility	C	Shall be present for an MT call if it was received in the Provide Roaming Number; otherwise shall be absent.
ISDN high layer compatibility	C	Shall be present for an MT call if it was received in the Provide Roaming Number; otherwise shall be absent.
CLIP provision	C	Indicates that CLIP is provisioned. Shall be present for an MT call if CLIP is provisioned; otherwise shall be absent.
CLIR override provision	C	Indicates that the CLIR override subscription option of CLIP is provisioned. Shall be present for an MT call if CLIP is provisioned with the CLIR override subscription option and the MS is registered in the HPLMN country; otherwise shall be absent.
CLIR provision	C	Indicates that CLIR is provisioned. Shall be present for an MO call if CLIR is provisioned; otherwise shall be absent.
CLIR mode	C	Indicates the mode in which CLIR is provisioned: permanent, temporary (default presentation allowed) or temporary (default presentation restricted). Shall be present for an MO call if CLIR is provisioned; otherwise shall be absent.
COLP provision	C	Indicates that COLP is provisioned. Shall be present for an MO call if COLP is provisioned; otherwise shall be absent.
COLR override provision	C	Indicates that the COLR override subscription option of COLP is provisioned. Shall be present for an MO call if COLP is provisioned with the COLR override subscription option and the MS is registered in the HPLMN country; otherwise shall be absent.
COLR provision	C	Indicates that COLR is provisioned. Shall be present for an MT call if COLR is provisioned; otherwise shall be absent.
No Reply Condition Timer	C	Value of timer to be used to determine the No subscriber reply condition. Shall be present for an MT call if the Call Forwarding on No Reply service is active and operative; otherwise shall be absent.

(continued)

The following information elements are required (concluded):

Information element name	Required	Description
CUG index	C	For the definition of this IE, see 3GPP TS 23.085 [18]. May be present (as a network operator option) for an ordinary MO call if the call is a CUG call; shall be present for an MT call if the call is a CUG call; otherwise shall be absent.
CUG interlock	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present for an ordinary MO call if the call is a CUG call; otherwise shall be absent.
CUG outgoing access	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present for an ordinary MO call if the call is a CUG call with outgoing access; otherwise shall be absent.
Advice of Charge provision	C	Indicates whether Advice of Charge (Information) or Advice of Charge (Charging) is provisioned. Shall be present for an ordinary MO call or an MT call if Advice of Charge is provisioned; otherwise shall be absent.
Alerting Pattern	C	Shall be present for an MT call if it was received in the Provide Roaming Number and if the feature is supported by the MSC/VLR; otherwise shall be absent.
NAEA preferred Carrier Id	O	The preferred carrier identity identifying the carrier to be used to route the interexchange call if the call requires routing via an interexchange carrier. This parameter may be included at the discretion of the VLR operator.

**** Next modified section ****

8.1.24 Process Call Waiting

The following information elements are required:

Information element name	Required	Description
MSISDN	M	MSISDN of the MS for which the Process Call Waiting is sent.
~~GSM~~ PLMN bearer capability	C	Shall be present according to the rules defined in 3GPP TS 29.007 [30]~~if it was received in the Provide Roaming Number for the waiting call; otherwise shall be absent~~.
ISDN bearer capability	C	Shall be present if it was received in the Provide Roaming Number for the waiting call; otherwise shall be absent.
ISDN low layer compatibility	C	Shall be present if it was received in the Provide Roaming Number for the waiting call; otherwise shall be absent.
ISDN high layer compatibility	C	Shall be present if it was received in the Provide Roaming Number for the waiting call; otherwise shall be absent.
CLIP provision	C	Indicates that CLIP is provisioned. Shall be present if CLIP is provisioned; otherwise shall be absent.
CLIR override provision	C	Indicates that the CLIR override subscription option of CLIP is provisioned. Shall be present if CLIP is provisioned with the CLIR override subscription option and the MS is registered in the HPLMN country; otherwise shall be absent.
COLR provision	C	Indicates that COLR is provisioned. Shall be present if COLR is provisioned; otherwise shall be absent.
No Reply Condition Timer	C	Value of timer to be used to determine the No subscriber reply condition. Shall be present if the Call Forwarding on No Reply service is active and operative; otherwise shall be absent.
CUG index	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present if the waiting call is a CUG call; otherwise shall be absent.
Advice of Charge provision	C	Indicates whether Advice of Charge (Information) or Advice of Charge (Charging) is provisioned. Shall be present if Advice of Charge is provisioned; otherwise shall be absent.

****** Next modified section ******

8.1.36 Send Info For Incoming Call

The following information elements are required:

Information element name	Required	Description
MSRN	M	Mobile Station Roaming Number received in the IAM.
Bearer service	C	~~GSM b~~Bearer service required for the MT call. Shall be present if the MSC was able to derive a ~~GSM~~ bearer service from ISDN BC/LLC/HLC information received in the IAM; otherwise shall be absent.
Teleservice	C	~~GSM t~~Teleservice required for the MT call. Shall be present if the MSC was able to derive a ~~GSM~~ teleservice from ISDN BC/LLC/HLC information received in the IAM; otherwise shall be absent.
Dialled number	C	Number dialled by the calling subscriber. Shall be present if it was received in the IAM; otherwise shall be absent.
Number of forwarding	C	Number of times the incoming call has already been forwarded. Shall be present if it was received in the IAM; otherwise shall be absent.
CUG interlock	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present if it was received in the IAM; otherwise shall be absent.
CUG outgoing access	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present if it was received in the IAM; otherwise shall be absent.

****** Next modified section ******

8.1.39 Send Info For Outgoing Call

The following information elements are required:

Information element name	Required	Description
Called number	M	E.164 number of the call destination.
Bearer service	C	Bearer service required for the MO call, derived from the ~~GSM~~ PLMN bearer capability information received in the set-up request from the MS. One of bearer service or teleservice shall be present.
Teleservice	C	Teleservice required for the MO call, derived from the ~~GSM~~ PLMN bearer capability information received in the set-up request from the MS or from the emergency set-up request from the MS. One of bearer service or teleservice shall be present.
CUG index	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present if it was received in the set-up request from the MS.
Suppress preferential CUG	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present if it was received in the set-up request from the MS.
Suppress CUG outgoing access	C	For the definition of this IE, see 3GPP TS 23.085 [18]. Shall be present if it was received in the set-up request from the MS.

****** Next modified section ******

8.3.1 Provide Roaming Number

The following information elements are required:

Information element name	Required	Description
IMSI	M	IMSI of the B subscriber (see 3GPP TS 23.003 [5]).
MSC number	M	E.164 number which identifies VMSCB (see 3GPP TS 23.003 [5]).
MSISDN	O	E.164 number which identifies the B subscriber. It shall be present if the following 3 conditions are all satisfied: 1. the MSISDN is different from the basic MSISDN 2. the subscriber has VT-CSI stored in HLR 3. the VLR has indicated support for CAMEL Phase 3 or later It may be present if the HLR requires it to be included in the call data record.
LMSI	C	Local Mobile Subscriber Identity. Shall be present if the LMSI was sent to HLRB at location updating.
~~GSM~~ PLMN bearer capability	C	Information to define the ~~GSM~~ PLMN bearer capability required for the call. For alternate speech/fax, alternate speech/data or speech followed by data calls this information element shall contain two ~~GSM~~ PLMN bearer capabilities, as specified in 3GPP TS 24.008. May be present if the HLR can determine the required ~~GSM~~ PLMN bearer capability from ISDN compatibility information received in the Send Routeing Info message, or from the MSISDN if a multi-numbering scheme is used; otherwise shall be absent. If the ISDN BC and ISDN LLC IEs are present, the PLMN ~~GSM~~ bearer capability IE shall be absent.
ISDN BC	C	ISDN bearer capability. May be present if the HLR received it in the Send Routeing Info message, otherwise shall be absent. If the PLMN ~~GSM~~ bearer capability IE is present, the ISDN BC IE shall be absent.
ISDN LLC	C	ISDN lower layer compatibility. May be present if the HLR received it in the Send Routeing Info message, otherwise shall be absent. If the PLMN ~~GSM~~ bearer capability IE is present, the ISDN LLC IE shall be absent.
ISDN HLC	C	ISDN higher layer compatibility. Shall be present if the HLR received it in the Send Routeing Info message, otherwise shall be absent.
Alerting Pattern	C	Shall be present if the HLR has determined an alerting category or an alerting level for the MT call configuration; otherwise shall be absent.
Pre-paging supported	C	Shall be present if the HLR has determined that pre-paging is supported in the GMSC and the HLR, otherwise shall be absent.

****** End of document ******

3GPP TSG CN WG4 Meeting #15 ***N4-021001***
Helsinki, Finland, 29th July – 2nd August 2002

****** Start of modifications ******

2 References

•••••

[66b] ETR 091: "ETSI object identifier tree; Common domain Mobile domain"

•••••

[93] ITU-T Recommendation X.680: "Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation" ~~X.208 (1988): "Specification of Abstract Syntax Notation One (ASN.1)"~~.

[93b] ITU-T Recommendation X.681: "Information technology – Abstract Syntax Notation One (ASN.1): Information object specification"

[94] ITU-T Recommendation X.690: "Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)" ~~X.209 (1988): "Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1)"~~.

•••••

[116] ITU-T Recommendation Q.850~~,~~ (May 1998): "Usage of cause and location in the Digital Subscriber Signalling System No. 1 and the Signalling System No. 7 ISDN User Part".

•••••

[125] ITU-T Recommendation X.880: "Data networks and open system communication - Open System Interconnection - Service definitions - Remote operations: Concepts, model and notation".

****** next modification ******

17 Abstract syntax of the MAP protocol

17.1 General

This clause specifies the Abstract Syntaxes for the Mobile Application Part as well as the associated set of Operations and Errors, using the Abstract Syntax Notation One (ASN.1), defined in ~~CCITT~~ ITU-T Recommendations X.680~~208 (1988) or X.680 (1994)~~ and X.681 with additions as defined in clause 17.1.4 on Compatibility Considerations and the OPERATION and ERROR external ~~MACROs~~information object classes, defined in ~~CCITT~~ITU-T Recommendation X.880~~Q.773~~.

The Abstract Syntax is defined for all interfaces specified in clause 4.4 except for the A- and B-interfaces.

The Mobile Application Part protocol is defined by two Abstract Syntaxes:

- one Abstract Syntax which encompass all Operations~~;~~ and
- Errors identified by the various MAP subsystem numbers.

This Abstract Syntax represents the set of values each of which is a value of the ASN.1 type TCAPMessages. TCMessage~~Type~~ as defined in ~~CCITT~~ITU-T Recommendation Q.773 with the ~~ANY DEFINED BY~~ component relation constraint sections resolved by the operation and error codes included in the ASN.1 modules MAP-~~*~~Operations and MAP-Errors~~Protocol~~. However, only the subset of this abstract syntax which is required by the procedures defined for an entity needs to be supported.~~:~~

- one Abstract Syntax identified by the OBJECT IDENTIFIER value MAP-DialogueInformation.map-DialogueAS.

This Abstract Syntax represents the set of values each of which is a value of the ASN.1 type MAP-~~-~~DialogueInformation.MAP-DialoguePDU. Such a value of the ASN.1 single-ASN.1-type element is contained within the user-information element of the TCAPMessages.DialoguePortion ASN.1 type. This Abstract Syntax name is to be used as a direct reference.

17.1.1 Encoding rules

The encoding rules which are applicable to the defined Abstract Syntaxes are the Basic Encoding Rules for Abstract Syntax Notation One, defined in ~~CCITT~~ITU-T Recommendation X.690 with the same exceptions as in ~~CCITT~~ITU-T Recommendation Q.773, clause 4 Message Representation.

When the definite form is used for length encoding, a data value of length less than 128 octets must have the length encoded in the short form.

When the long form is employed to code a length, the minimum number of octets shall be used to code the length field.

OCTET STRING values and BIT STRING values must be encoded in a primitive form.

There is no restriction to the use of empty constructors (e.g. an empty SEQUENCE type). That is, the encoding of the content of any data value shall consist of zero, one or more octets.

17.1.2 Use of TC

The mapping of OPERATION and ERROR to TC components is defined in ETS 300 287 (version 2) which is based on ~~CCITT~~ITU-T Recommendation Q.773 ~~(1992)~~.

NOTE 1: The class of an operation is not stated explicitly but is specified as well in the ASN.1 operation ~~type~~ definition.

Class 1: RESULT and ERROR appear in ASN.1 operation ~~type~~ definition.

Class 2: only ERROR appears in ASN.1 operation ~~type~~ definition.

Class 3: only RESULT appears in ASN.1 operation ~~type~~ definition.

Class 4: both RESULT and ERROR do not appear in ASN.1 operation ~~type~~ definition.

The ~~ASN.1 data type which follows the keywords~~field "ARGUMENT", "PARAMETER" or "RESULT" (for information objects of class OPERATION and ERROR) is always optional from a syntactic point of view. However, except when specifically mentioned with the ASN.1 comment "~~«~~-- optional"~~»~~ , the "~~«~~parameter"~~»~~ part of a component has to be considered as mandatory from a semantic point of view.

When an optional element is missing in an invoke component or in an inner data structure while it is required by the context, an error component is returned if specified in the information object associated with the operation ~~type~~; the associated type of error is "DataMissing". This holds also when the entire parameter of an invoke component is missing while it is required by the context.

NOTE 2: When a mandatory element is missing in the parameter or inner data structure of any component, a reject component is returned (if the dialogue still exists). The problem code to be used is "Mistyped parameter".

The Timer Values used in the operation ~~type~~ definitions are indicated as ASN.1 comments. The Timer Value Ranges are:

s = from 3 seconds to 10 seconds;

m = from 15 seconds to 30 seconds;

ml = from 1 minute to 10 minutes;

l = from 28 hours to 38 hours.

17.1.2.1 Use of Global Operation and Error codes defined outside MAP

An entity supporting an application context greater than 2 shall be capable of receiving an operation or error code, within an application context defined in GSM 2~~0~~9.002, encoded as either an Object Identifier (as defined in ~~CCITT~~ITU-T Recommendation X.690 ~~(1994)~~) or an integer value (as defined in clause 17.5). Related restrictions regarding the use of Object Identifiers are as follows:

- The length of the Object Identifier shall not exceed 16 octets and the number of components of the Object Identifier shall not exceed 16.
- Object Identifiers shall be used only for operations or errors defined outside of GSM 2~~0~~9.002.
- Global error codes may be sent only in response to a global operation. If a standard operation is received then a global error code shall not be sent in response.

Handling of an unknown operation codes by the receiving entity is defined in clause 15.1.1.

17.1.3 Use of information elements defined outside MAP

An information element or a set of information elements (messages) transparently carried in the Mobile Application Part but defined in other recommendations/technical specifications are handled in one of the following ways:

i) The contents of each information element (without the octets encoding the identifier and the length in the recommendation/technical specification where it is defined unless explicitly stated otherwise) is carried as the

value of an ASN.1 ~~NamedT~~type derived from the OCTET STRING data type. Additionally, the internal structure may be explained by means of comments. In case of misalignment the referred to recommendation/technical specification takes precedence.

ii) The complete information element (including the octets encoding the identifier and the length in the recommendation/technical specification where it is defined) or set of information elements and the identity of the associated protocol are carried as the value of the ExternalSignalInfo data type defined in the present document. Where more than one information element is carried, the information elements are sent contiguously with no filler octets between them.

17.1.4 Compatibility considerations

The following ASN.1 modules conform to ITU-T~~CCITT~~ Recommendations X.680~~208 (1988)~~ and~~or~~ X.6810 ~~(1994) (the only module which makes use of X.680 is MAP-ExtensionDataTypes), but in addition Ellipsis Notation~~. An extension marker ("...")~~ notation)~~ is used ~~as described in ITU-T Recommendation X.680 Amendment 1 (1995)~~ wherever future protocol extensions are foreseen.

The "..." construct applies only to SEQUENCE and ENUMERATED data types. An entity supporting a version greater than 1 shall not reject an unsupported extension following "..." of that SEQUENCE or ENUMERATED data type. The Encoding Rules from clause 17.1.1 apply to every element of the whole Transfer Syntax especially to the ASN.1 type EXTERNAL.

The extension container "privateExtensionList" is defined in this specification in order to carry extensions which are defined outside this specification. Private extensions can be defined by, for example, network operators, manufacturers, and regional standardisation bodies.

Private extensions shall:

1) if included in operations of an AC of V2, follow the extension marker and be tagged using PRIVATE tags up to and including 29.

NOTE: This type of extension is in most cases used only within a PLMN.

2) if included in operations of an AC of V3 or higher: be included only in the Private Extension Container that is defined in the specification.

NOTE: This type of extension can be used between PLMNs.

Private extensions shall not be included in v2 supplementary service operations.

Private extensions shall not be included within user error for RegisterCCEntry and EraseCCEntry operations.

PCS extensions shall be included in the PCS Extension Container that is defined in this specification.

In order to improve extensibility, a few error parameters have been defined as a CHOICE between the version 2 description and a SEQUENCE including the version 2 description and an extension container. Operations used in a v2-application-context must consider only the first alternative while operations used in a vn-application-context (n>2) must consider only the second alternative.

17.1.5 Structure of the Abstract Syntax of MAP

For each MAP parameter which has to be transferred by a MAP Protocol Data Unit (MAP message), there is a PDU field (an ASN.1 ~~NamedT~~type) which~~ose ASN.1 identifier~~ has the same name as the corresponding parameter, except for the differences required by the ASN.1 notation (blanks between words are removed or replaced by hyphen, the first letter of the first word is capital~~lower case~~ and the first letter of each of the following words is~~are~~ capitalised, e.g. "no reply condition time" is mapped to "N~~n~~oReplyConditionTime"). Additionally some words may be abbreviated as follows:

bs basic service

ch call handling

cug closed user group

ho handover

ic incoming call

id identity

info information

mm mobility management

lcs location services

ms mobile service

oc outgoing call

om operation & maintenance

pw Password

sm short message service

ss supplementary service

st secure transport

The MAP protocol is composed of several ASN.1 modules dealing with either operations, errors, data types, and, if applicable, split into those dealing with mobile services, call handling services, supplementary services and short message services. For operations and errors ~~no~~the code values are ~~assigned~~given as parameters, ~~but only the operation and error types~~ in order to allow use of the defined ~~types~~information objects also by other protocols (e.g. 3GPP TS 24.080 [38]). ~~The values (operation codes and error codes) are defined in a separate module.~~ The ASN.1 source lines are preceded by line-numbers at the left margin in order to enable the usage of the cross-reference in annex A.

The module containing the definition of the operation packages for MAP is:

1. MAP-OperationPackages.

The module containing the definition of the application contexts for MAP is:

2. MAP-ApplicationContexts.

The module containing the data types for the Abstract Syntax to be used for TCAPMessages.DialoguePortion for MAP is:

3. MAP-DialogueInformation.

The module containing the supported operations~~operation codes and error codes for MAP~~ is:

4. MAP-Protocol.

The modules containing all operation ~~type~~ definitions for MAP are:

5. MAP-MobileServiceOperations;
6. MAP-OperationAndMaintenanceOperations;
7. MAP-CallHandlingOperations;
8. MAP-SupplementaryServiceOperations;
9. MAP-ShortMessageServiceOperations;
10. MAP-Group-Call-Operations;
11. MAP-LocationServiceOperations;
12. MAP-SecureTransportOperations.

The module containing all error ~~type~~ definitions for MAP is:

13. MAP-Errors.

Modules containing all data type definitions for MAP are:

14. MAP-MS-DataTypes;
15. MAP-OM-DataTypes;
16. MAP-CH-DataTypes;
17. MAP-SS-DataTypes;
18. MAP-SS-Code;
19. MAP-SM-DataTypes;
20. MAP-ER-DataTypes;
21. MAP-CommonDataTypes;
22. MAP-TS-Code;
23. MAP-BS-Code;
24. MAP-ExtensionDataTypes;
25. MAP-GR-DataTypes;
26. MAP-LCS-DataTypes;
27. MAP-ST-DataTypes.

References are made also to modules defined outside of the present document. They are defined in the technical specification Mobile Services Domain, ~~and~~ technical specification Transaction Capability and ITU-T Recommendation X.880 respectively:

MobileDomainDefinitions;

TCAPMessages, ~~;~~ DialoguePDUs;

~~DialoguePDUs~~Remote-Operations-Information-Objects.

17.1.6 Application Contexts

The following informative table lists the latest versions of the Application Contexts used in this specification, with the operations used by them and, where applicable, whether or not the operation description is exactly the same as for previous versions. Information in 17.6 & 17.7 relates only to the ACs in this table.

AC Name	AC Version	Operations Used	Comments
locationCancellationContext	v3	cancelLocation	
equipmentMngtContext	v2	checkIMEI	
imsiRetrievalContext	v2	sendIMSI	
infoRetrievalContext	v3	sendAuthenticationInfo	
interVlrInfoRetrievalContext	v3	sendIdentification	
handoverControlContext	v3	prepareHandover forwardAccessSignalling sendEndSignal processAccessSignalling prepareSubsequentHandover	the syntax of this operation has been extended in comparison with release 98 version
mwdMngtContext	v3	readyForSM	
msPurgingContext	v3	purgeMS	
shortMsgAlertContext	v2	alertServiceCentre	
resetContext	v2	reset	

AC Name	AC Version	Operations Used	Comments
networkUnstructuredSsContext	v2	processUnstructuredSS-Request unstructuredSS-Request unstructuredSS-Notify	
tracingContext	v3	activateTraceMode deactivateTraceMode	
networkFunctionalSsContext	v2	registerSS eraseSS activateSS deactivateSS registerPassword interrogateSS getPassword	
shortMsgMO-RelayContext	v3	mo-forwardSM	
shortMsgMT-RelayContext	v3	mt-forwardSM	
shortMsgGatewayContext	v3	sendRoutingInfoForSM reportSM-DeliveryStatus InformServiceCentre	the syntax of this operation has been extended in comparison with release 96 version
networkLocUpContext	v3	updateLocation forwardCheckSs-Indication restoreData insertSubscriberData activateTraceMode	the syntax is the same in v1 & v2
gprsLocationUpdateContext	v3	updateGprsLocation insertSubscriberData activateTraceMode	
subscriberDataMngtContext	v3	insertSubscriberData deleteSubscriberData	
roamingNumberEnquiryContext	v3	provideRoamingNumber	
locationInfoRetrievalContext	v3	sendRoutingInfo	
gprsNotifyContext	v3	noteMsPresentForGprs	
gprsLocationInfoRetrievalContext	v4	sendRoutingInfoForGprs	
failureReportContext	v3	failureReport	
callControlTransferContext	v4	resumeCallHandling	
subscriberInfoEnquiryContext	v3	provideSubscriberInfo	
anyTimeEnquiryContext	v3	anyTimeInterrogation	
anyTimeInfoHandlingContext	v3	anyTimeSubscriptionInterrogation anyTimeModification	
ss-InvocationNotificationContext	v3	ss-InvocationNotification	
sIWFSAllocationContext	v3	provideSIWFSNumber sIWFSSignallingModify	
groupCallControlContext	v3	prepareGroupCall processGroupCallSignalling forwardGroupCallSignalling sendGroupCallEndSignal	
reportingContext	v3	setReportingState statusReport remoteUserFree	
callCompletionContext	v3	registerCC-Entry eraseCC-Entry	
istAlertingContext	v3	istAlert	
ImmediateTerminationContext	v3	istCommand	
locationSvcEnquiryContext	v3	provideSubscriberLocation subscriberLocationReport	
locationSvcGatewayContext	v3	sendRoutingInfoForLCS	
mm-EventReportingContext	v3	noteMM-Event	
subscriberDataModificationNotificationContext	v3	noteSubscriberDataModified	
authenticationFailureReportContext	v3	authenticationFailureReport	

AC Name	AC Version	Operations Used	Comments
secureTransportHandlingContext	v3	secureTransportClass1 secureTransportClass2 secureTransportClass3 secureTransportClass4	

NOTE (*): The syntax of the operations is not the same as in previous versions unless explicitly stated

17.2 Operation packages

17.2.1 General aspects

This clause describes the operation-packages which are used to build the application-contexts defined in clause 17.3.

Each operation-package is a specification of the roles of a pair of communicating objects (i.e. a pair of MAP-Providers), in terms of operations which they can invoke of each other.

The grouping of operations into one or several packages does not necessarily imply any grouping in terms of Application Service Elements.

The following ASN.1 ~~MACRO~~ information object class is used to describe operation-packages in this clause:

```
OPERATION-PACKAGE ::= CLASS {
     &Both      OPERATION          OPTIONAL,
     &Consumer  OPERATION          OPTIONAL,
     &Supplier  OPERATION          OPTIONAL,
     &id        OBJECT IDENTIFIER UNIQUE OPTIONAL }~~MACRO ::=~~

~~BEGIN~~

~~TYPE NOTATION ::= Symmetric | ConsumerInvokes SupplierInvokes |~~
~~empty~~

~~VALUE NOTATION ::= value(VALUE OBJECT IDENTIFIER)~~
~~Symmetric ::= "OPERATIONS" "{" OperationList "}"~~
~~ConsumerInvokes ::= "CONSUMER INVOKES" "{" OperationList "}"~~
~~SupplierInvokes ::= "SUPPLIER INVOKES" "{" OperationList "}" | empty~~
~~OperationList ::= Operation | OperationList "," Operation~~
~~Operation ::= value(OPERATION)~~

~~END~~
WITH SYNTAX {
     [ OPERATIONS       &Both ]
     [ CONSUMER INVOKES &Supplier ]
     [ SUPPLIER INVOKES &Consumer ]
     [ ID               &id ] }
```

Since the application-context definitions provided in clause 17.3 use only an informal description technique, only the type notation is used in the following clauses to define operation-packages.

The following definitions are used throughout this clause (n>=2):

- v1-only operation: An operation which shall be used only in v1 application-contexts;
- vn-only operation: An operation which shall be used only in vn application-contexts;
- v(n-1)-operation: An operation whose specification has not been modified since the MAP v(n-1) specifications or if the modifications are considered as not affecting v(n-1) implementations;
- v(n-1)-equivalent operation: The version of an operation which excludes all the information elements and errors which have been added since the MAP v(n-1) specification;
- vn-only package: An operation package which contains only vn-only operations;
- v(n-1)-package: An operation package which contains only v(n-1)- operations.

The names of vn-packages are suffixed by "-vn" where n>=2.

For each operation package which is not vn-only (n>=2) and which does not include only v(n-1)-operations, there is a v(n-1)-equivalent package. Except when a definition is explicitly provided in the following clauses, the v(n-1)-equivalent package includes the v(n-1)-equivalent operations of the operations which belong to this package.

17.2.2 Packages specifications

17.2.2.1 Location updating

This operation package includes the operations required for location management procedures between HLR and VLR.

```
lLocationUpdatingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          updateLocation}
     SUPPLIER INVOKES {
          forwardCheckSs-Indication} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.2 Location cancellation

This operation package includes the operations required for location cancellation and MS purging procedures between HLR and VLR and between HLR and SGSN.

```
lLocationCancellationPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          cancelLocation} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.3 Roaming number enquiry

This operation package includes the operations required for roaming number enquiry procedures between HLR and VLR.

```
rRoamingNumberEnquiryPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR if Consumer is HLR
     CONSUMER INVOKES {
          provideRoamingNumber} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.4 Information retrieval

This operation package includes the operation required for the authentication information retrieval procedure between HLR and VLR and between HLR and SGSN.

```
iInfoRetrievalPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     -- Supplier is HLR if Consumer is SGSN
     CONSUMER INVOKES {
          sendAuthenticationInfo} }
```

The v2-equivalent package is defined as follows:

```
iInfoRetrievalPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     -- Supplier is HLR if Consumer is SGSN
     CONSUMER INVOKES {
          sendAuthenticationInfo} }
```

The v1-equivalent package is defined as follows:

```
iInfoRetrievalPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR or VLR if Consumer is VLR
     -- Supplier is HLR if Consumer is SGSN
     CONSUMER INVOKES {
          sendParameters} }
```

17.2.2.5 Inter-VLR information retrieval

This operation package includes the operations required for inter VLR information retrieval procedures.

```
iInterVlrInfoRetrievalPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR if Consumer is VLR
     CONSUMER INVOKES {
          sendIdentification} }
```

The v2-equivalent package is defined as follows:

```
iInterVlrInfoRetrievalPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR if Consumer is VLR
     CONSUMER INVOKES {
          sendIdentification} }
```

The v1-equivalent package is : iInfoRetrievalPackage-v1.

17.2.2.6 IMSI retrieval

This operation package includes the operation required for the IMSI retrieval procedure between HLR and VLR.

```
IMSIimsi-RetrievalPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          sendIMSI} }
```

This package is v2 only.

17.2.2.7 Call control transfer

This operation package includes the operation required for the call control transfer procedure between VMSC and GMSC.

```
cCallControlTransferPackage-v4 ::= OPERATION-PACKAGE ::= {
     -- Supplier is GMSC if Consumer is VMSC
     CONSUMER INVOKES {
          resumeCallHandling} }
```

The v3-equivalent package can be determined according to the rules described in clause 17.2.1.

17.2.2.8 Secure transport

This operation package includes the operations required for the secure transport of MAP messages between any MAP entities.

```
sSecureTransportHandlingPackage-v3 ::= OPERATION-PACKAGE ::= {
     CONSUMER INVOKES {
          sSecureTransportClass1 |       -- to be used if the original operation is a
                                         -- TCAP class 1 operation
          sSecureTransportClass2 |       -- to be used if the original operation is a
                                         -- TCAP class 2 operation
          sSecureTransportClass3 |       -- to be used if the original operation is a
                                         -- TCAP class 3 operation
          sSecureTransportClass4}        -- to be used if the original operation is a
                                         -- TCAP class 4 operation
     }
```

This package is v3 only.

17.2.2.9 Void

17.2.2.10 Interrogation

This operation package includes the operations required for interrogation procedures between MSC and HLR or NPLR or between HLR and gsmSCF.

```
iInterrogationPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR or NPLR if Consumer is MSC
     -- Supplier is HLR if Consumer is gsmSCF
     CONSUMER INVOKES {
          sendRoutingInfo} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.11 Void

17.2.2.12 Handover Control

This operation package includes the operations required for handover procedures between MSCs.

```
hHandoverControlPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is MSCB if Consumer is MSCA
     CONSUMER INVOKES {
          prepareHandover |
          forwardAccessSignalling}
     SUPPLIER INVOKES {
          sendEndSignal |
          processAccessSignalling |
          prepareSubsequentHandover} }
```

The v2-equivalent package can be determined according to the rules described in clause 17.2.1.

The v1-equivalent package is defined as follows.

```
hHandoverControlPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is MSCB if Consumer is MSCA
     CONSUMER INVOKES {
          performHandover |
          forwardAccessSignalling |
          traceSubscriberActivity}
     SUPPLIER INVOKES {
          sendEndSignal |
          noteInternalHandover |
          processAccessSignalling |
          performSubsequentHandover} }
```

17.2.2.13 Subscriber Data management stand alone

This operation package includes the operations required for stand alone subscriber data management procedures between HLR and VLR or between HLR and SGSN.

```
sSubscriberDataMngtStandAlonePackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          insertSubscriberData |
          deleteSubscriberData} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.14 Equipment management

This operation package includes the operations required for equipment management procedures between EIR and MSC or between EIR and SGSN.

```
eEquipmentMngtPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is EIR if Consumer is MSC
     -- Supplier is EIR if Consumer is SGSN
     CONSUMER INVOKES {
          checkIMEI} }
```

The v1-equivalent package can be determined according to the rules described in clause 17.2.1.

17.2.2.15 Subscriber data management

This operation package includes the operations required for subscriber data management procedures between HLR and VLR or between HLR and SGSN.

```
sSubscriberDataMngtPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          insertSubscriberData} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.16 Location register restart

This operation package includes the operations required for location register restart procedures between HLR and VLR or between HLR and SGSN.

```
rResetPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          reset} }
```

The v1-equivalent package can be determined according to the rules described in clause 17.2.1.

17.2.2.17 Tracing stand-alone

This operation package includes the operations required for stand alone tracing procedures between HLR and VLR or between HLR and SGSN.

```
tTracingStandAlonePackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          activateTraceMode |
          deactivateTraceMode} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.18 Functional SS handling

This operation package includes the operations required for functional supplementary services procedures between VLR and HLR.

```
fFunctionalSsPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          registerSS |
          eraseSS |
          activateSS |
          deactivateSS |
          registerPassword |
          interrogateSS}
     SUPPLIER INVOKES {
          getPassword} }
```

The v1-equivalent package can be determined according to the rules described in clause 17.2.1.

17.2.2.19 Tracing

This operation package includes the operations required for tracing procedures between HLR and VLR or between HLR and SGSN.

```
tTracingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          activateTraceMode} }
```

The v1-equivalent and v2-equivalent packages can be determined according to the rules described in clause 17.2.1.

17.2.2.20 Binding

This operation package includes the operation required to initialise a supplementary service procedure between VLR and HLR or between gsmSCF and HLR.

```
bBindingPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     -- Supplier is gsmSCF if Consumer is HLR
     CONSUMER INVOKES {
          beginSubscriberActivity} }
```

This package is v1 only.

17.2.2.21 Unstructured SS handling

This operation package includes the operations required for unstructured supplementary services procedures between VLR and HLR, between the HLR and the gsmSCF, and between HLR and HLR.

```
uUnstructuredSsPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     -- Supplier is gsmSCF or HLR if Consumer is HLR
     CONSUMER INVOKES {
          processUnstructuredSS-Request}
     SUPPLIER INVOKES {
          unstructuredSS-Request,
          unstructuredSS-Notify} }
```

The v1-equivalent package is defined as follows:

```
uUnstructuredSsPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     -- Supplier is gsmSCF if Consumer is HLR
     CONSUMER INVOKES {
          processUnstructuredSS-Data} }
```

17.2.2.22 MO Short message relay services

This operation package includes the operations required for short message relay service procedures between IWMSC and VMSC or between GMSC and MSC or between SGSN and IWMSC.

```
mo-MOShortMsgRelayPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is IWMSC if Consumer is MSC
     -- Supplier is IWMSC if Consumer is SGSN
     CONSUMER INVOKES {
          moMO-forwardSM} }
```

The v2-equivalent package is defined as follows:

```
sShortMsgRelayPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is IWMSC if Consumer is MSC
     -- Supplier is MSC or SGSN if Consumer is GMSC
     -- Supplier is IWMSC if Consumer is SGSN
     CONSUMER INVOKES {
          forwardSM} }
```

The v1-equivalent package can be determined according to the rules described in clause 17.2.1.

17.2.2.23 Short message gateway services

This operation package includes the operations required for short message service gateway procedures between MSC and HLR.

```
sShortMsgGatewayPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is GMSC
     CONSUMER INVOKES {
          sendRoutingInfoForSM,
          reportSM-DeliveryStatus}
     SUPPLIER INVOKES {
          informServiceCentre} }
```

The v2-equivalent package can be determined according to the rules described in clause 17.2.1.

The v1-equivalent package is defined as follows:

```
sShortMsgGatewayPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is GMSC
     CONSUMER INVOKES {
          sendRoutingInfoForSM,
          reportSMDeliveryStatus} }
```

17.2.2.24 MT Short message relay services

This operation package includes the operations required for short message relay service procedures between GMSC and MSC or between GMSC and SGSN.

```
mt-MTShortMsgRelayPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is MSC or SGSN if Consumer is GMSC
     CONSUMER INVOKES {
          mtMT-forwardSM}
```

The v2-equivalent package is: **sShortMsgRelayPackage-v2**

17.2.2.25 Void

17.2.2.26 Message waiting data management

This operation package includes the operations required for short message waiting data procedures between HLR and VLR, between HLR and SGSN.

```
mMwdMngtPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is SGSN
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          readyForSM} }
```

The v2-equivalent package can be determined according to the rules described in clause 17.2.1.

The v1-equivalent package is defined as follows:

```
mMwdMngtPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          noteSubscriberPresent} }
```

17.2.2.27 Alerting

This operation package includes the operations required for alerting between HLR and IWMSC.

```
aAlertingPackage-v2 ::= OPERATION-PACKAGE ::= {
     -- Supplier is IWMSC if Consumer is HLR
     CONSUMER INVOKES {
          alertServiceCentre} }
```

The v1-equivalent package is defined as follows.

```
aAlertingPackage-v1 ::= OPERATION-PACKAGE ::= {
     -- Supplier is IWMSC if Consumer is HLR
     CONSUMER INVOKES {
          alertServiceCentreWithoutResult} }
```

17.2.2.28 Data restoration

This operation package includes the operations required for VLR data restoration between HLR and VLR.

```
dDataRestorationPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          restoreData} }
```

The v2-equivalent package can be determined according to the rules described in clause 17.2.1.

The v1-equivalent package is: iInfoRetrievalPackage-v1

17.2.2.29 Purging

This operation package includes the operations required for purging between HLR and VLR or between HLR and SGSN.

```
pPurgingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     -- Supplier is HLR if Consumer is SGSN
     CONSUMER INVOKES {
          purgeMS} }
```

The v2-equivalent package can be determined according to the rules described in clause 17.2.1.

17.2.2.30 Subscriber information enquiry

This operation package includes the operations required for subscriber information enquiry procedures between HLR and VLR or between HLR and SGSN.

```
sSubscriberInformationEnquiryPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR or SGSN if Consumer is HLR
     CONSUMER INVOKES {
          provideSubscriberInfo} }
```

This package is v3 only.

17.2.2.31 Any time information enquiry

This operation package includes the operations required for any time information enquiry procedures between gsmSCF and HLR or between gsmSCF and GMLC.

```
aAnyTimeInformationEnquiryPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR or GMLC if Consumer is gsmSCF
     CONSUMER INVOKES {
          anyTimeInterrogation} }
```

This package is v3 only.

17.2.2.32 Group Call Control

This operation package includes the operations required for group call and broadcast call procedures between MSCs.

```
gGroupCallControlPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is relay MSC if Consumer is anchor MSC
     CONSUMER INVOKES {
          prepareGroupCall |
          forwardGroupCallSignalling}
     SUPPLIER INVOKES {
          sendGroupCallEndSignal |
          processGroupCallSignalling} }
```

This package is v3 only.

17.2.2.33 Provide SIWFS number

This operation package includes the operations required between VMSC and SIWF for requesting resources from an SIWF.

```
pProvideSIWFSNumberPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is SIWF if Consumer is VMSC
     CONSUMER INVOKES {
          provideSIWFSNumber} }
```

This package is v3 only.

17.2.2.34 SIWFS Signalling Modify

This operation package includes the operations required for the modification of the resources in an SIWF between the VMSC and SIWF.

```
siwfs-SIWFSSignallingModifyPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is SIWF if Consumer is VMSC
     CONSUMER INVOKES {
          sIWFSSignallingModify} }
```

This package is v3 only.

17.2.2.35 Gprs location updating

This operation package includes the operations required for the gprs location management procedures between HLR and SGSN.

```
gGprsLocationUpdatingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is SGSN
     CONSUMER INVOKES {
          updateGprsLocation} }
```

This package is v3 only.

17.2.2.36 Gprs Interrogation

This operation package includes the operations required for interrogation procedures between HLR and GGSN.

```
gGprsInterrogationPackage-v4 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is GGSN
     CONSUMER INVOKES {
          sendRoutingInfoForGprs} }
```

The v3-equivalent package is defined as follows.

```
gGprsInterrogationPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is GGSN
     CONSUMER INVOKES {
          sendRoutingInfoForGprs} }
```

17.2.2.37 Failure reporting

This operation package includes the operations required for failure reporting between HLR and GGSN.

```
fFailureReportingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is GGSN
     CONSUMER INVOKES {
          failureReport} }
```

This package is v3 only.

17.2.2.38 GPRS notifying

This operation package includes the operations required for notifying that GPRS subscriber is present between HLR and GGSN.

```
gGprsNotifyingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is GGSN if Consumer is HLR
     CONSUMER INVOKES {
          noteMsPresentForGprs} }
```

This package is v3 only.

17.2.2.39 Supplementary Service invocation notification

This operation package includes the operations required for Supplementary Service invocation notification procedures between the MSC and the gsmSCF and between the HLR and the gsmSCF.

```
ssSS-InvocationNotificationPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is gsmSCF if Consumer is MSC
     -- Supplier is gsmSCF if Consumer is HLR
     CONSUMER INVOKES {
          ss-InvocationNotification} }
```

This package is v3 only.

17.2.2.40 Set Reporting State

This operation package includes the operation required for procedures between HLR and VLR to set the reporting state.

```
sSetReportingStatePackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR if Consumer is HLR
     CONSUMER INVOKES {
          setReportingState} }
```

This package is v3 only.

17.2.2.41 Status Report

This operation package includes the operation required for procedures between VLR and HLR to report call results and events.

```
sStatusReportPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          statusReport} }
```

This package is v3 only.

17.2.2.42 Remote User Free

This operation package includes the operation required by the HLR to indicate to the VLR that the remote user is free.

```
rRemoteUserFreePackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VLR if Consumer is HLR
     CONSUMER INVOKES {
          remoteUserFree} }
```

This package is v3 only.

17.2.2.43 Call Completion

This operation package includes the operations required for procedures between VLR and HLR for subscriber control of call completion services.

```
cCallCompletionPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VLR
     CONSUMER INVOKES {
          registerCC-Entry |
          eraseCC-Entry} }
```

This package is v3 only.

17.2.2.44 Location service gateway services

This operation package includes the operations required for location service gateway procedures between GMLC and HLR.

```
lLocationSvcGatewayPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is GMLC
     CONSUMER INVOKES {
          sendRoutingInfoForLCS} }
```

This package is v3 only.

17.2.2.45 Location service enquiry

This operation package includes the operations required for the location service enquiry procedures between GMLC and MSC and between GMLC and SGSN.

```
lLocationSvcEnquiryPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is MSC or SGSN if Consumer is GMLC
     CONSUMER INVOKES {
          provideSubscriberLocation} }
```

This package is v3 only.

17.2.2.45A Location service reporting

This operation package includes the operations required for the location service enquiry procedures between MSC and GMLC and between SGSN and GMLC.

```
lLocationSvcReportingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is GMLC if Consumer is MSC
     -- Supplier is GMLC if Consumer is SGSN
     CONSUMER INVOKES {
          subscriberLocationReport} }
```

17.2.2.46 Void

17.2.2.47 Void

17.2.2.48 Void

17.2.2.49 IST Alerting

This operation package includes the operation required for alerting procedures between the MSC (Visited MSC or Gateway MSC) and HLR.

```
istIST-AlertingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is HLR if Consumer is VMSC
     -- Supplier is HLR if Consumer is GMSC
     CONSUMER INVOKES {
          istAlert} }
```

This package is v3 only.

17.2.2.50 Service Termination

This operation package includes the operation required for immediate service termination procedures between the HLR and the Visited MSC or between the HLR and the Gateway MSC.

```
sServiceTerminationPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is VMSC or GMSC if Consumer is HLR
     CONSUMER INVOKES {
          istCommand} }
```

This package is v3 only.

17.2.2.51 Mobility Management event notification

This operation package includes the operations required for Mobility Management event notification procedures between VLR and gsmSCF.

```
mmMM-EventReportingPackage-v3 ::= OPERATION-PACKAGE ::= {
     -- Supplier is gsmSCF if Consumer is VLR
     CONSUMER INVOKES {
          noteMM-Event} }
```

This package is v3 only.

17.2.2.52 Any time information handling

This operation package includes the operations required for any time information handling procedures between gsmSCF and HLR.

```
aAnyTimeInformationHandlingPackage-v3 ::= OPERATION-PACKAGE ::= {
    -- Supplier is HLR if Consumer is gsmSCF
    CONSUMER INVOKES {
        anyTimeSubscriptionInterrogation ,
        anyTimeModification} }
```

This package is v3 only.

17.2.2.53 Subscriber Data modification notification

This operation package includes the operations required for Subscriber Data modification notification procedures between HLR and gsmSCF.

```
sSubscriberDataModificationNotificationPackage-v3 ::= OPERATION-PACKAGE ::= {
    -- Supplier is gsmSCF if Consumer is HLR
    CONSUMER INVOKES {
        noteSubscriberDataModified} }
```

This package is v3 only.

17.2.2.54 Authentication Failure Report

This operation package includes the operation required for procedures between VLR and HLR or the SGSN and the HLR for reporting of authentication failures.

```
aAuthenticationFailureReportPackage-v3 ::= OPERATION-PACKAGE ::= {
    -- Supplier is HLR if Consumer is VLR
    -- Supplier is HLR if Consumer is SGSN
    CONSUMER INVOKES {
        authenticationFailureReport} }
```

This package is v3 only.

17.3 Application contexts

17.3.1 General aspects

An application-context is assigned for each dialogue established by a MAP-user. In the present document each application-context is assigned a name which is supplied in the MAP-OPEN Req primitive by the MAP-User and transmitted to the peer under certain circumstances.

The following ASN.1 ~~MACRO~~ information object class is used to describe the main aspects of application-contexts in the following clauses:

```
APPLICATION-CONTEXT ::= CLASS {
     &Symmetric            OPERATION-PACKAGE OPTIONAL,
     &InitiatorConsumerOf  OPERATION-PACKAGE OPTIONAL,
     &ResponderConsumerOf  OPERATION-PACKAGE OPTIONAL,
     &code                 OBJECT IDENTIFIER }~~MACRO ::=~~

~~BEGIN~~

~~TYPE NOTATION ::= Symmetric | InitiatorConsumerOf
ResponderConsumerOf | empty~~

~~VALUE NOTATION ::= value(VALUE OBJECT IDENTIFIER)~~

~~Symmetric ::= "OPERATIONS OF" "{" PackageList "}"~~

~~InitiatorConsumerOf ::= "INITIATOR CONSUMER OF" "{" PackageList "}"~~

~~ResponderConsumerOf ::= "RESPONDER CONSUMER OF" "{" PackageList "}"~~
~~          | empty~~

~~PackageList ::= Package | PackageList "," Package~~

~~Package ::= value(OPERATION-PACKAGE)~~
~~          | type -- shall reference a package type~~

~~END~~
WITH SYNTAX {
     [ OPERATIONS OF          &Symmetric ]
     [ INITIATOR CONSUMER OF &InitiatorConsumerOf
       RESPONDER CONSUMER OF &ResponderConsumerOf ]
     ID &code }
```

The following definitions are used throughout this clause:

- v1-application-context: An application-context which contains only v1-packages and uses only TC v1 facilities;
- v1 context set: the set of v1-application-contexts defined in the present document.
- vn-application-context (n>=2): An application-context which contains only vn-packages;

The names of v1-application-contexts are suffixed by "-v1" while other names are suffixed by "-vn" where n>=2.

Application-contexts which do not belong to the v1 context set use v2 TC facilities.

The last component of each application-context-name (i.e. the last component of the object identifier value) assigned to an application-context which belongs to the v1 context set indicates explicitly "version1".

For each application-context which does not belong to the "v1 context set" there is a v1-equivalent application context. This is a v1-application-context which includes the v1-equivalents of the packages included in the original context.

Each application-context uses the abstract-syntax associated with the operation-packages it includes and uses the transfer-syntax derived from it by applying the encoding rules defined in clause 17.1.1.

ACs which do not belong to the v1 context set require the support of the abstract-syntax identified by the object identifier value: MAP-DialogueInformation.map-Dialogue-AS defined in clause 17.4.

17.3.2 Application context definitions

17.3.2.1 Void

17.3.2.2 Location Updating

This application context is used between HLR and VLR for location updating procedures.

```
networkLocUpContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          locationUpdatingPackage-v3 |
          dataRestorationPackage-v3}
     RESPONDER CONSUMER OF {
          subscriberDataMngtPackage-v3 |
          tracingPackage-v3}
::=  ID   {map-ac networkLocUp(1) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   {map-ac networkLocUp(1) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac networkLocUp(1) version1(1)}
```

17.3.2.3 Location Cancellation

This application context is used between HLR and VLR or between HLR and SGSN for location cancellation procedures. For the HLR - SGSN interface only version 3 of this application context is applicable.

```
locationCancellationContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR or SGSN if Initiator is HLR
     INITIATOR CONSUMER OF {
          locationCancellationPackage-v3}
::=  ID   {map-ac locationCancel(2) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   map-ac locationCancel(2) version2(2)
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   map-ac locationCancel(2) version1(1)
```

17.3.2.4 Roaming number enquiry

This application context is used between HLR and VLR for roaming number enquiry procedures.

```
roamingNumberEnquiryContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR if Initiator is HLR
     INITIATOR CONSUMER OF {
          roamingNumberEnquiryPackage-v3}
::=  ID   {map-ac roamingNbEnquiry(3) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   {map-ac roamingNbEnquiry(3) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac roamingNbEnquiry(3) version1(1)}
```

17.3.2.5 Void

17.3.2.6 Location Information Retrieval

This application-context is used between GMSC and HLR or between GMSC and NPLR or between gsmSCF and HLR when retrieving location information. For the GMSC - NPLR interface version 1, version 2 and version 3 of this application context are applicable.

```
locationInfoRetrievalContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR or NPLR if Initiator is GMSC
     -- Responder is HLR if Initiator is gsmSCF
     INITIATOR CONSUMER OF {
          iInterrogationPackage-v3}
::=  ID   {map-ac locInfoRetrieval(5) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   {map-ac locInfoRetrieval(5) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac locInfoRetrieval(5) version1(1)}
```

17.3.2.7 Call control transfer

This application context is used for the call control transfer procedure between the VMSC and the GMSC.

```
callControlTransferContext-v4 APPLICATION-CONTEXT ::= {
     -- Responder is GMSC if Initiator is VMSC
     INITIATOR CONSUMER OF {
          cCallControlTransferPackage-v4}
::=  ID   {map-ac callControlTransfer(6) version4(4)} }
```

The following application-context-name is assigned to the v3-equivalent application-context:

```
     ID   {map-ac callControlTransfer(6) version3(3)}
```

17.3.2.8 Secure transport

This application context is used for the secure transport of MAP messages between any MAP entities.

```
sSecureTransportHandlingContext-v3 APPLICATION-CONTEXT ::= {
     INITIATOR CONSUMER OF {
          sSecureTransportHandlingPackage-v3}
::=  ID   {map-ac secureTransportHandling(40) version3(3)} }
```

This application-context is v3 only.

17.3.2.9 - 17.3.2.10 Void

17.3.2.11 Location registers restart

This application context is used between HLR and VLR or between HLR and SGSN for location register restart procedures. For the HLR - SGSN interface version 1 and version 2 of this application context are applicable.

```
resetContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is VLR or SGSN if Initiator is HLR
     INITIATOR CONSUMER OF {
          rResetPackage-v2}
::=  ID   {map-ac reset(10) version2(2)} }
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac reset(10) version1(1)}
```

17.3.2.12 Handover control

This application context is used for handover procedures between MSCs.

```
handoverControlContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is MSCB if Initiator is MSCA
     INITIATOR CONSUMER OF {
          hHandoverControlPackage-v3}
::= ID   {map-ac handoverControl(11) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   {map-ac handoverControl(11) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac handoverControl(11) version1(1)}
```

17.3.2.13 IMSI Retrieval

This application context is used for IMSI retrieval between HLR and VLR.

```
imsiRetrievalContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          imsi-IMSIRetrievalPackage-v2}
::= ID   {map-ac imsiRetrieval(26) version2(2)} }
```

This application-context is v2 only.

17.3.2.14 Equipment Management

This application context is used for equipment checking between MSC and EIR or between SGSN and EIR. For the SGSN - EIR interface version 1 and version 2 of this application context are applicable:

```
equipmentMngtContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is EIR if Initiator is MSC
     -- Responder is EIR if Initiator is SGSN
     INITIATOR CONSUMER OF {
          eEquipmentMngtPackage-v2}
::= ID   {map-ac equipmentMngt(13) version2(2)} }
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac equipmentMngt(13) version1(1)}
```

17.3.2.15 Information retrieval

This application context is used for authentication information retrieval between HLR and VLR or between HLR and SGSN. For the HLR - SGSN interface version 1 and version 2 and version 3 of this application context are applicable.

```
infoRetrievalContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     -- Responder is HLR if Initiator is SGSN
     INITIATOR CONSUMER OF {
          iInfoRetrievalPackage-v3}
::= ID   {map-ac infoRetrieval(14) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
infoRetrievalContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     -- Responder is HLR if Initiator is SGSN
     INITIATOR CONSUMER OF {
          iInfoRetrievalPackage-v2}
::= ID   {map-ac infoRetrieval(14) version2(2)} }
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
    ID   {map-ac infoRetrieval(14) version1(1)}
```

17.3.2.16 Inter-VLR information retrieval

This application context is used for information retrieval between VLRs.

```
interVlrInfoRetrievalContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          iInterVlrInfoRetrievalPackage-v3}
::= ID   {map-ac interVlrInfoRetrieval(15) version3(3)} }
```

The v2-equivalent application-context is:

```
interVlrInfoRetrievalContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is VLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          iInterVlrInfoRetrievalPackage-v2}
::= ID   {map-ac interVlrInfoRetrieval(15) version2(2)} }
```

The v1-equivalent application-context is:

```
    ID   {map-ac infoRetrieval(14) version1(1)}
```

17.3.2.17 Stand Alone Subscriber Data Management

This application context is used for stand alone subscriber data management between HLR and VLR or between HLR and SGSN. For the HLR - SGSN interface only version 3 of this application context is applicable:

```
subscriberDataMngtContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR or SGSN if Initiator is HLR
     INITIATOR CONSUMER OF {
          sSubscriberDataMngtStandAlonePackage-v3}
::= ID   {map-ac subscriberDataMngt(16) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
    ID   {map-ac subscriberDataMngt(16) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
    ID   {map-ac subscriberDataMngt(16) version1(1)}
```

17.3.2.18 Tracing

This application context is used between HLR and VLR or between HLR and SGSN for stand alone tracing control procedures. For the HLR - SGSN interface version 1, version 2 and version 3 of this application context are applicable.

```
tracingContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR or SGSN if Initiator is HLR
     INITIATOR CONSUMER OF {
          tTracingStandAlonePackage-v3}
::= ID   {map-ac tracing(17) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
    ID   {map-ac tracing(17) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
    ID   {map-ac tracing(17) version1(1)}
```

17.3.2.19 Network functional SS handling

This application context is used for functional-like SS handling procedures between VLR and HLR.

```
networkFunctionalSsContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is HLR, Initiator is VLR
     INITIATOR CONSUMER OF {
          fFunctionalSsPackage-v2}
::= ID   {map-ac networkFunctionalSs(18) version2(2)} }
```

The v1-equivalent application-context is defined as follows:

```
networkFunctionalSsContext-v1 APPLICATION-CONTEXT ::= {
     -- Responder is HLR, Initiator is VLR
     INITIATOR CONSUMER OF {
          fFunctionalSsPackage-v1 |,
          uUnstructuredSsPackage-v1 |,
          bBindingPackage-v1}
::= ID   {map-ac networkFunctionalSs(18) version1(1)} }
```

17.3.2.20 Network unstructured SS handling

This application context is used for handling stimuli-like procedures between HLR and VLR, between the HLR and gsmSCF, and between HLR and HLR.

```
networkUnstructuredSsContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is HLR, Initiator is VLR
     -- Responder is VLR, Initiator is HLR
     -- Responder is gsmSCF, Initiator is HLR
     -- Responder is HLR, Initiator is gsmSCF
     -- Responder is HLR, Initiator is HLR
     OPERATIONS OF {
          uUnstructuredSsPackage-v2}
::= ID   {map-ac networkUnstructuredSs(19) version2(2)} }
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac networkFunctionalSs(18) version1(1)}
```

17.3.2.21 Short Message Gateway

This application context is used for short message gateway procedures.

```
shortMsgGatewayContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is GMSC
     INITIATOR CONSUMER OF {
          sShortMsgGatewayPackage-v3}
::= ID   {map-ac shortMsgGateway(20) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   {map-ac shortMsgGateway(20) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
     ID   {map-ac shortMsgGateway(20) version1(1)}
```

17.3.2.22 Mobile originating Short Message Relay

This application context is used between MSC and IWMSC or between SGSN and IWMSC for mobile originating short message relay procedures. For the SGSN - IWMSC interface version 1, version 2 and version 3 of this application context are applicable.

```
shortMsgMO-RelayContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is IWMSC if Initiator is MSC
     -- Responder is IWMSC if Initiator is SGSN
     INITIATOR CONSUMER OF {
          mo-MOShortMsgRelayPackage-v3}
::= ID   {map-ac shortMsgMO-Relay(21) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
    ID   {map-ac shortMsgMO-Relay(21) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
    ID   {map-ac shortMsg-Relay(21) version1(1)}
```

17.3.2.23 Void

17.3.2.24 Short message alert

This application context is used for short message alerting procedures.

```
shortMsgAlertContext-v2 APPLICATION-CONTEXT ::= {
     -- Responder is IWMSC if Initiator is HLR
     INITIATOR CONSUMER OF {
          aAlertingPackage-v2}
::= ID   {map-ac shortMsgAlert(23) version2(2)} }
```

The following application-context-name is symbolically assigned to the v1-equivalent application-context:

```
    ID   {map-ac shortMsgAlert(23) version1(1)}
```

17.3.2.25 Short message waiting data management

This application context is used between VLR and HLR or between SGSN and HLR for short message waiting data management procedures. For the SGSN - HLR interface only version 3 of this application context is applicable.

```
mwdMngtContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is SGSN
     -- Responder is HLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          mMwdMngtPackage-v3}
::= ID   {map-ac mwdMngt(24) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
    ID   {map-ac mwdMngt(24) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
    ID   {map-ac mwdMngt(24) version1(1)}
```

17.3.2.26 Mobile terminating Short Message Relay

This application context is used between GMSC and MSC or between GMSC and SGSN for mobile terminating short message relay procedures. For the GMSC - SGSN interface version 2 and version 3 of this application context and the equivalent version 1 application context are applicable.

```
shortMsgMT-RelayContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is MSC or SGSN if Initiator is GMSC
     INITIATOR CONSUMER OF {
          mt-MTShortMsgRelayPackage-v3}
::= ID   {map-ac shortMsgMT-Relay(25) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
    ID   {map-ac shortMsgMT-Relay(25) version2(2)}
```

The following application-context-name is assigned to the v1-equivalent application-context:

```
    ID   {map-ac shortMsgMO-Relay(21) version1(1)}
```

17.3.2.27 MS purging

This application context is used between HLR and VLR or between HLR and SGSN for MS purging procedures. For the SGSN - HLR interface only version 3 of this application context is applicable.

```
msPurgingContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     -- Responder is HLR if Initiator is SGSN
     INITIATOR CONSUMER OF {
          purgingPackage-v3}
::= ID   {map-ac msPurging(27) version3(3)} }
```

The following application-context-name is assigned to the v2-equivalent application-context:

```
     ID   {map-ac msPurging(27) version2(2)}
```

17.3.2.28 Subscriber information enquiry

This application context is used between HLR and VLR or between HLR and SGSN for subscriber information enquiry procedures.

```
subscriberInfoEnquiryContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR or SGSN if Initiator is HLR
     INITIATOR CONSUMER OF {
          sSubscriberInformationEnquiryPackage-v3}
::= ID   {map-ac subscriberInfoEnquiry(28) version3(3)} }
```

This application-context is v3 only.

17.3.2.29 Any time information enquiry

This application context is used between gsmSCF and HLR or between gsmSCF and GMLC for any time information enquiry procedures.

```
anyTimeInfoEnquiryContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR or GMLC if Initiator is gsmSCF
     INITIATOR CONSUMER OF {
          aAnyTimeInformationEnquiryPackage-v3}
::= ID   {map-ac anyTimeInfoEnquiry(29) version3(3)} }
```

This application-context is v3 only.

17.3.2.30 Group Call Control

This application context is used between anchor MSC and relay MSC for group call and broadcast call procedures.

```
groupCallControlContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is relay MSC if Initiator is anchor MSC
     INITIATOR CONSUMER OF {
          gGroupCallControlPackage-v3}
::= ID   {map-ac groupCallControl(31) version3(3)} }
```

This application-context is v3 only.

17.3.2.31 Provide SIWFS Number

This application context is used for activation or modification of SIWF resources.

```
sIWFSAllocationContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is SIWF if Initiater is VMSC
     INITIATOR CONSUMER OF {
          pProvideSIWFSNumberPackage-v3 |
          siwfs-SIWFSSignallingModifyPackage-v3}
::= ID   {map-ac sIWFSAllocation (12) version3(3)} }
```

This application-context is v3 only.

17.3.2.32 Gprs Location Updating

This application context is used between HLR and SGSN for gprs location updating procedures.

```
gprsLocationUpdateContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is SGSN
     INITIATOR CONSUMER OF {
          gGprsLocationUpdatingPackage-v3}
     RESPONDER CONSUMER OF {
          sSubscriberDataMngtPackage-v3 |
          tTracingPackage-v3}
::= ID   {map-ac gprsLocationUpdate(32) version3(3)} }
```

This application-context is v3 only.

17.3.2.33 Gprs Location Information Retreival

This application context is used between HLR and GGSN when retrieving gprs location information.

```
gprsLocationInfoRetrievalContext-v4 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is GGSN
     INITIATOR CONSUMER OF {
          gGprsInterrogationPackage-v4}
::= ID   {map-ac gprsLocationInfoRetrieval(33) version4(4)} }
```

The following application-context-name is assigned to the v3-equivalent application-context:

```
     ID   {map-ac gprsLocationInfoRetrieval(33) version3(3)}
```

17.3.2.34 Failure Reporting

This application context is used between HLR and GGSN to inform that network requested PDP-context activation has failed.

```
failureReportContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is GGSN
     INITIATOR CONSUMER OF {
          fFailureReportingPackage-v3}
::= ID   {map-ac failureReport(34) version3(3)} }
```

This application-context is v3 only.

17.3.2.35 GPRS Notifying

This application context is used between HLR and GGSN for notifying that GPRS subscriber is present again.

```
gprsNotifyContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is GGSN if Initiator is HLR
     INITIATOR CONSUMER OF {
          gGprsNotifyingPackage-v3}
::= ID   {map-ac gprsNotify(35) version3(3)} }
```

This application-context is v3 only.

17.3.2.36 Supplementary Service invocation notification

This application context is used between the MSC and the gsmSCF and between the HLR and the gsmSCF for Supplementary Service invocation notification procedures.

```
ss-InvocationNotificationContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is gsmSCF, Initiator is MSC
     -- Responder is gsmSCF, Initiator is HLR
     INITIATOR CONSUMER OF {
          ssSS-InvocationNotificationPackage-v3}
::= ID   {map-ac ss-InvocationNotification(36) version3(3)} }
```

This application-context is v3 only.

17.3.2.37 Reporting

This application context is used between HLR and VLR for reporting procedures.

```
reportingContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VLR if Initiator is HLR
     -- Responder is HLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          sSetReportingStatePackage-v3 |
          sStatusReportPackage-v3 |
          rRemoteUserFreePackage-v3}
     RESPONDER CONSUMER OF {
          sSetReportingStatePackage-v3|,
          sStatusReportPackage-v3}
::= ID   {map-ac reporting(7) version3(3)} }
```

This application-context is v3 only.

17.3.2.38 Call Completion

This application context is used between VLR and the HLR for subscriber control of call completion services.

```
callCompletionContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     INITIATOR CONSUMER OF {
          cCallCompletionPackage-v3}
::= ID   {map-ac callCompletion(8) version3(3)} }
```

This application-context is v3 only.

17.3.2.39 Location Service Gateway

This application context is used for location service gateway procedures.

```
locationSvcGatewayContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is GMLC
     INITIATOR CONSUMER OF {
          locationSvcGatewayPackage-v3}
::= ID   {map-ac locationSvcGateway(37) version3(3)} }
```

17.3.2.40 Location Service Enquiry

This application context is used for location service enquiry procedures.

```
locationSvcEnquiryContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is MSC or SGSN if Initiator is GMLC
     -- Responder is GMLC if Initiator is MSC
     -- Responder is GMLC if Initiator is SGSN
     INITIATOR CONSUMER OF {
          locationSvcEnquiryPackage-v3 |
          locationSvcReportingPackage-v3}
::= ID   {map-ac locationSvcEnquiry(38) version3 (3)} }
```

17.3.2.41 Void

17.3.2.42 Void

17.3.2.43 Void

17.3.2.44 IST Alerting

This application context is used between MSC (Visited MSC or Gateway MSC) and HLR for alerting services within IST procedures.

```
istAlertingContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VMSC
     -- Responder is HLR if Initiator is GMSC
     INITIATOR CONSUMER OF {
          istIST-AlertingPackage-v3}
::= ID   {map-ac alerting(4) version3(3)} }
```

This application-context is v3 only.

17.3.2.45 Service Termination

This application context is used between HLR and MSC (Visited MSC or Gateway MSC) for service termination services within IST procedures.

```
serviceTerminationContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is VMSC or GMSC if Initiator is HLR
     INITIATOR CONSUMER OF {
          sServiceTerminationPackage-v3}
::= ID   {map-ac serviceTermination(9) version3(3)} }
```

This application-context is v3 only.

17.3.2.46 Mobility Management event notification

This application context is used between VLR and gsmSCF for Mobility Management event notification procedures.

```
mm-EventReportingContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is gsmSCF, Initiator is VLR
     INITIATOR CONSUMER OF {
          mmMM-EventReportingPackage-v3}
::= ID   {map-ac mm-EventReporting(42) version3(3)} }
```

This application-context is v3 only.

17.3.2.47 Any time information handling

This application context is used between gsmSCF and HLR for any time information handling procedures.

```
anyTimeInfohandlingContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is gsmSCF
     INITIATOR CONSUMER OF {
          aAnyTimeInformationHandlingPackage-v3}
::= ID   {map-ac anyTimeInfoHandling(43) version3(3)} }
```

This application-context is v3 only.

17.3.2.48 Subscriber Data modification notification

This application context is used between HLR and gsmSCF for Subscriber Data modification notification procedures.

```
subscriberDataModificationNotificationContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is gsmSCF, Initiator is HLR
     INITIATOR CONSUMER OF {
          sSubscriberDataModificationNotificationPackage-v3}
::= ID   {map-ac subscriberDataModificationNotification(22) version3(3)} }
```

This application-context is v3 only.

17.3.2.49 Authentication Failure Report

This application context is used between VLR and HLR or SGSN and HLR for reporting of authentication failures.

```
authenticationFailureReportContext-v3 APPLICATION-CONTEXT ::= {
     -- Responder is HLR if Initiator is VLR
     -- Responder is HLR if Initiator is SGSN
     INITIATOR CONSUMER OF {
          aAuthenticationFailureReportPackage-v3 }
::= ID   {map-ac authenticationFailureReport(39) version3(3)} }
```

This application-context is v3 only.

17.3.3 ASN.1 Module for application-context-names

The following ASN.1 module summarises the application-context-name assigned to MAP application-contexts.

```
MAP-ApplicationContexts {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ApplicationContexts (2) version8 (8)}

DEFINITIONS

::=

BEGIN


-- EXPORTS everything


IMPORTS
   gsm-NetworkId,
   ac-Id
```

```
FROM MobileDomainDefinitions {
   ccittitu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
   mobileDomainDefinitions (0) version1 (1)}
;

-- application-context-names

map-ac  OBJECT IDENTIFIER ::= {gsm-NetworkId ac-Id}

networkLocUpContext-v3  OBJECT IDENTIFIER ::=
     {map-ac networkLocUp(1) version3(3)}

locationCancellationContext-v3  OBJECT IDENTIFIER ::=
     {map-ac locationCancel(2) version3(3)}

roamingNumberEnquiryContext-v3  OBJECT IDENTIFIER ::=
     {map-ac roamingNbEnquiry(3) version3(3)}

authenticationFailureReportContext-v3  OBJECT IDENTIFIER ::=
     {map-ac authenticationFailureReport(39) version3(3)}

locationInfoRetrievalContext-v3  OBJECT IDENTIFIER ::=
     {map-ac locInfoRetrieval(5) version3(3)}

resetContext-v2  OBJECT IDENTIFIER ::=
     {map-ac reset(10) version2(2)}

handoverControlContext-v3  OBJECT IDENTIFIER ::=
     {map-ac handoverControl(11) version3(3)}

equipmentMngtContext-v2  OBJECT IDENTIFIER ::=
     {map-ac equipmentMngt(13) version2(2)}

infoRetrievalContext-v3  OBJECT IDENTIFIER ::=
     {map-ac infoRetrieval(14) version3(3)}

interVlrInfoRetrievalContext-v3  OBJECT IDENTIFIER ::=
     {map-ac interVlrInfoRetrieval(15) version3(3)}

subscriberDataMngtContext-v3  OBJECT IDENTIFIER ::=
     {map-ac subscriberDataMngt(16) version3(3)}

tracingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac tracing(17) version3(3)}

networkFunctionalSsContext-v2  OBJECT IDENTIFIER ::=
     {map-ac networkFunctionalSs(18) version2(2)}

networkUnstructuredSsContext-v2  OBJECT IDENTIFIER ::=
     {map-ac networkUnstructuredSs(19) version2(2)}

shortMsgGatewayContext-v3  OBJECT IDENTIFIER ::=
     {map-ac shortMsgGateway(20) version3(3)}

shortMsgMO-RelayContext-v3  OBJECT IDENTIFIER ::=
     {map-ac shortMsgMO-Relay(21) version3(3)}

shortMsgAlertContext-v2  OBJECT IDENTIFIER ::=
     {map-ac shortMsgAlert(23) version2(2)}

mwdMngtContext-v3  OBJECT IDENTIFIER ::=
     {map-ac mwdMngt(24) version3(3)}

shortMsgMT-RelayContext-v3  OBJECT IDENTIFIER ::=
     {map-ac shortMsgMT-Relay(25) version3(3)}

imsiRetrievalContext-v2  OBJECT IDENTIFIER ::=
     {map-ac imsiRetrieval(26) version2(2)}

msPurgingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac msPurging(27) version3(3)}

subscriberInfoEnquiryContext-v3  OBJECT IDENTIFIER ::=
     {map-ac subscriberInfoEnquiry(28) version3(3)}
```

```
anyTimeInfoEnquiryContext-v3  OBJECT IDENTIFIER ::=
     {map-ac anyTimeInfoEnquiry(29) version3(3)}

callControlTransferContext-v4  OBJECT IDENTIFIER ::=
     {map-ac callControlTransfer(6) version4(4)}

ss-InvocationNotificationContext-v3  OBJECT IDENTIFIER ::=
     {map-ac ss-InvocationNotification(36) version3(3)}

sIWFSAllocationContext-v3  OBJECT IDENTIFIER ::=
     {map-ac sIWFSAllocation(12) version3(3)}

groupCallControlContext-v3  OBJECT IDENTIFIER ::=
     {map-ac groupCallControl(31) version3(3)}

gprsLocationUpdateContext-v3  OBJECT IDENTIFIER ::=
     {map-ac gprsLocationUpdate(32) version3(3)}

gprsLocationInfoRetrievalContext-v4  OBJECT IDENTIFIER ::=
     {map-ac gprsLocationInfoRetrieval(33) version4(4)}

failureReportContext-v3  OBJECT IDENTIFIER ::=
     {map-ac failureReport(34) version3(3)}

gprsNotifyContext-v3  OBJECT IDENTIFIER ::=
     {map-ac gprsNotify(35) version3(3)}

reportingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac reporting(7) version3(3)}

callCompletionContext-v3  OBJECT IDENTIFIER ::=
     {map-ac callCompletion(8) version3(3)}

istAlertingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac istAlerting(4) version3(3)}

serviceTerminationContext-v3  OBJECT IDENTIFIER ::=
     {map-ac immediateTermination(9) version3(3)}

locationSvcGatewayContext-v3  OBJECT IDENTIFIER ::=
     {map-ac locationSvcGateway(37) version3(3)}

locationSvcEnquiryContext-v3  OBJECT IDENTIFIER ::=
     {map-ac locationSvcEnquiry(38) version3(3)}


mm-EventReportingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac mm-EventReporting(42) version3(3)}

anyTimeInfoHandlingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac anyTimeInfoHandling(43) version3(3)}

subscriberDataModificationNotificationContext-v3  OBJECT IDENTIFIER ::=
     {map-ac subscriberDataModificationNotification(22) version3(3)}

secureTransportHandlingContext-v3  OBJECT IDENTIFIER ::=
     {map-ac secureTransportHandling(40) version3(3)}

-- The following Object Identifiers are reserved for application-contexts
-- contexts existing in previous versions of the protocol
```

-- AC Name & Version	Object Identifier	
--		
-- networkLocUpContext-v1	map-ac networkLocUp (1)	version1 (1)
-- networkLocUpContext-v2	map-ac networkLocUp (1)	version2 (2)
-- locationCancellationContext-v1	map-ac locationCancellation (2)	version1 (1)
-- locationCancellationContext-v2	map-ac locationCancellation (2)	version2 (2)
-- roamingNumberEnquiryContext-v1	map-ac roamingNumberEnquiry (3)	version1 (1)
-- roamingNumberEnquiryContext-v2	map-ac roamingNumberEnquiry (3)	version2 (2)
-- locationInfoRetrievalContext-v1	map-ac locationInfoRetrieval (5)	version1 (1)
-- locationInfoRetrievalContext-v2	map-ac locationInfoRetrieval (5)	version2 (2)
-- resetContext-v1	map-ac reset (10)	version1 (1)
-- handoverControlContext-v1	map-ac handoverControl (11)	version1 (1)
-- handoverControlContext-v2	map-ac handoverControl (11)	version2 (2)
-- equipmentMngtContext-v1	map-ac equipmentMngt (13)	version1 (1)
-- infoRetrievalContext-v1	map-ac infoRetrieval (14)	version1 (1)
-- infoRetrievalContext-v2	map-ac infoRetrieval (14)	version2 (2)
-- interVlrInfoRetrievalContext-v2	map-ac interVlrInfoRetrieval (15)	version2 (2)
-- subscriberDataMngtContext-v1	map-ac subscriberDataMngt (16)	version1 (1)
-- subscriberDataMngtContext-v2	map-ac subscriberDataMngt (16)	version2 (2)
-- tracingContext-v1	map-ac tracing (17)	version1 (1)
-- tracingContext-v2	map-ac tracing (17)	version2 (2)
-- networkFunctionalSsContext-v1	*map-ac networkFunctionalSs (18)*	*version1 (1)*
-- shortMsgGatewayContext-v1	map-ac shortMsgGateway (20)	version1 (1)
-- shortMsgGatewayContext-v2	map-ac shortMsgGateway (20)	version2 (2)
-- shortMsgRelayContext-v1	map-ac shortMsgRelay (21)	version1 (1)
-- shortMsgAlertContext-v1	map-ac shortMsgAlert (23)	version1 (1)
-- mwdMngtContext-v1	*map-ac mwdMngt (24)*	*version1 (1)*
-- mwdMngtContext-v2	map-ac mwdMngt (24)	version2 (2)
-- shortMsgMT-RelayContext-v2	map-ac shortMsgMT-Relay (25)	version2 (2)
-- msPurgingContext-v2	map-ac msPurging (27)	version2 (2)
-- callControlTransferContext-v3	map-ac callControlTransferContext (6)	version3 (3)
-- gprsLocationInfoRetrievalContext-v3	map-ac gprsLocationInfoRetrievalContext (33)	version3 (3)

```
END
```

17.4 MAP Dialogue Information

```
MAP-DialogueInformation {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-DialogueInformation (3) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS
   map-DialogueAS,
   MAP-DialoguePDU,
   map-ProtectedDialogueAS,
   MAP-ProtectedDialoguePDU

;

IMPORTS
   gsm-NetworkId,
   as-Id
FROM MobileDomainDefinitions {
   ccittitu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
   mobileDomainDefinitions (0) version1 (1)}

   AddressString
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network(1) modules (3) map-CommonDataTypes (18) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}

   SecurityHeader,
   ProtectedPayload
```

```
FROM MAP-ST-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ST-DataTypes (27) version8 (8)}

;


-- abstract syntax name for MAP-DialoguePDU
```

```
map-DialogueAS  OBJECT IDENTIFIER ::=
     {gsm-NetworkId as-Id map-DialoguePDU (1) version1 (1)}
```

```
MAP-DialoguePDU ::= CHOICE {
     map-open                          [0] MAP-OpenInfo,
     map-accept                        [1] MAP-AcceptInfo,
     map-close                         [2] MAP-CloseInfo,
     map-refuse                        [3] MAP-RefuseInfo,
     map-userAbort                     [4] MAP-UserAbortInfo,
     map-providerAbort                 [5] MAP-ProviderAbortInfo}
```

```
MAP-OpenInfo ::= SEQUENCE {
     destinationReference              [0] AddressString               OPTIONAL,
     originationReference              [1] AddressString               OPTIONAL,
     ...,
     extensionContainer                ExtensionContainer              OPTIONAL
     -- extensionContainer must not be used in version 2
     }
```

```
MAP-AcceptInfo ::= SEQUENCE {
     ...,
     extensionContainer                ExtensionContainer              OPTIONAL
     -- extensionContainer must not be used in version 2
     }
```

```
MAP-CloseInfo ::= SEQUENCE {
     ...,
     extensionContainer                ExtensionContainer              OPTIONAL
     -- extensionContainer must not be used in version 2
     }
```

```
MAP-RefuseInfo ::= SEQUENCE {
     reason    Reason,
     ...,
     extensionContainer                ExtensionContainer              OPTIONAL,
     -- extensionContainer must not be used in version 2
     alternativeApplicationContext     OBJECT IDENTIFIER               OPTIONAL
     -- alternativeApplicationContext must not be used in version 2
     }
```

```
Reason ::= ENUMERATED {
     noReasonGiven                     (0),
     invalidDestinationReference       (1),
     invalidOriginatingReference       (2),
     encapsulatedAC-NotSupported       (3) ,
     transportProtectionNotAdequate    (4)}
     -- encapsulatedAC-NotSupported and transportProtectionNotAdequate must not be used in
     -- dialogues with an AC different from secureTransportHandling
```

```
MAP-UserAbortInfo ::= SEQUENCE {
     map-UserAbortChoice               MAP-UserAbortChoice,
     ...,
     extensionContainer                ExtensionContainer              OPTIONAL
     -- extensionContainer must not be used in version 2
     }
```

```
MAP-UserAbortChoice ::= CHOICE {
     userSpecificReason                [0] NULL,
     userResourceLimitation            [1] NULL,
     resourceUnavailable               [2] ResourceUnavailableReason,
     applicationProcedureCancellation  [3] ProcedureCancellationReason}
```

```
ResourceUnavailableReason ::= ENUMERATED {
     shortTermResourceLimitation  (0),
     longTermResourceLimitation  (1)}
```

```
ProcedureCancellationReason ::= ENUMERATED {
     handoverCancellation  (0),
     radioChannelRelease  (1),
     networkPathRelease  (2),
     callRelease  (3),
     associatedProcedureFailure  (4),
     tandemDialogueRelease  (5),
     remoteOperationsFailure  (6)}
```

```
MAP-ProviderAbortInfo ::= SEQUENCE {
     map-ProviderAbortReason             MAP-ProviderAbortReason,
     ...,
     extensionContainer                  ExtensionContainer                OPTIONAL
     -- extensionContainer must not be used in version 2
     }
```

```
MAP-ProviderAbortReason ::= ENUMERATED {
     abnormalDialogue  (0),
     invalidPDU  (1)}
```

-- abstract syntax name for MAP-ProtectedDialoguePDU

```
map-ProtectedDialogueAS  OBJECT IDENTIFIER ::=
     {gsm-NetworkId as-Id map-ProtectedDialoguePDU (3) version1 (1)}
```

```
MAP-ProtectedDialoguePDU ::= SEQUENCE {
     encapsulatedAC                      OBJECT IDENTIFIER,
     securityHeader                      SecurityHeader                    OPTIONAL,
     protectedPayload                    ProtectedPayload                  OPTIONAL,
     ...}
     -- The protectedPayload carries the result of applying the security function
     -- defined in 3G TS 33.200 to the encoding of the securely transported
     -- MAP-DialoguePDU
```

```
END
```

17.5 MAP operation and error codes

```
MAP-Protocol {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Protocol (4) version8 (8)}

DEFINITIONS

::=

BEGIN

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)}

   uUpdateLocation,
   cCancelLocation,
   pPurgeMS,
   sSendIdentification,
   uUpdateGprsLocation,
   pPrepareHandover,
   sSendEndSignal,
   pProcessAccessSignalling,
   fForwardAccessSignalling,
   pPrepareSubsequentHandover,
   sSendAuthenticationInfo,
   aAuthenticationFailureReport,
   cCheckIMEI,
   iInsertSubscriberData,
   dDeleteSubscriberData,
   rReset,
   fForwardCheckSS-Indication,
   rRestoreData,
   pProvideSubscriberInfo,
   aAnyTimeInterrogation,
   aAnyTimeSubscriptionInterrogation,
   aAnyTimeModification,
   sSendRoutingInfoForGprs,
   fFailureReport,
```

```
    nNoteMsPresentForGprs,
    nNoteMM-Event,
    nNoteSubscriberDataModified


FROM MAP-MobileServiceOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-MobileServiceOperations (5)
    version8 (8)}

    aActivateTraceMode,
    dDeactivateTraceMode,
    sSendIMSI
FROM MAP-OperationAndMaintenanceOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-OperationAndMaintenanceOperations (6)
    version8 (8)}

    sSendRoutingInfo,
    pProvideRoamingNumber,
    rResumeCallHandling,
    pProvideSIWFSNumber,
    siwfs-SIWFSSignallingModify,
    sSetReportingState,
    sStatusReport,
    rRemoteUserFree,
    istIST-Alert,
    istIST-Command
FROM MAP-CallHandlingOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-CallHandlingOperations (7)
    version8 (8)}

    rRegisterSS,
    eEraseSS,
    aActivateSS,
    dDeactivateSS,
    iInterrogateSS,
    pProcessUnstructuredSS-Request,
    uUnstructuredSS-Request,
    uUnstructuredSS-Notify,
    rRegisterPassword,
    gGetPassword,
    ssSS-InvocationNotification,
    rRegisterCC-Entry,
    eEraseCC-Entry
FROM MAP-SupplementaryServiceOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-SupplementaryServiceOperations (8)
    version8 (8)}

    sSendRoutingInfoForSM,
    moMO-ForwardSM,
    mtMT-ForwardSM,
    rReportSM-DeliveryStatus,
    aAlertServiceCentre,
    iInformServiceCentre,
    rReadyForSM
FROM MAP-ShortMessageServiceOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-ShortMessageServiceOperations (9)
    version8 (8)}

    pPrepareGroupCall,
    pProcessGroupCallSignalling,
    fForwardGroupCallSignalling,
    sSendGroupCallEndSignal
FROM MAP-Group-Call-Operations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-Group-Call-Operations (22)
    version8 (8)}
```

```
    pProvideSubscriberLocation,
    sSendRoutingInfoForLCS,
    sSubscriberLocationReport
FROM MAP-LocationServiceOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-LocationServiceOperations (24)
    version8 (8)}

sSecureTransportClass1,
sSecureTransportClass2,
sSecureTransportClass3,
sSecureTransportClass4

FROM MAP-SecureTransportOperations {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-SecureTransportOperations (26)
    version8 (8)}

    SystemFailure,
    DataMissing,
    UnexpectedDataValue,
    FacilityNotSupported,
    UnknownSubscriber,
    NumberChanged,
    UnknownMSC,
    UnidentifiedSubscriber,
    UnknownEquipment,
    RoamingNotAllowed,
    IllegalSubscriber,
    IllegalEquipment,
    BearerServiceNotProvisioned,
    TeleserviceNotProvisioned,
    NoHandoverNumberAvailable,
    SubsequentHandoverFailure,
    TracingBufferFull,
    OR-NotAllowed,
    NoRoamingNumberAvailable,
    AbsentSubscriber,
    BusySubscriber,
    NoSubscriberReply,
    CallBarred,
    ForwardingViolation,
    ForwardingFailed,
    CUG-Reject,
    ATI-NotAllowed,
    IllegalSS-Operation,
    SS-ErrorStatus,
    SS-NotAvailable,
    SS-SubscriptionViolation,
    SS-Incompatibility,
    UnknownAlphabet,
    USSD-Busy,
    PW-RegistrationFailure,
    NegativePW-Check,
    NumberOfPW-AttemptsViolation,
    SubscriberBusyForMT-SMS,
    SM-DeliveryFailure,
    MessageWaitingListFull,
    AbsentSubscriberSM,
    ResourceLimitation,
    NoGroupCallNumberAvailable,
    ShortTermDenial,
    LongTermDenial,
    IncompatibleTerminal,
    UnauthorizedRequestingNetwork,
    UnauthorizedLCSClient,
    PositionMethodFailure,
    UnknownOrUnreachableLCSClient,
    ATSI-NotAllowed,
    ATM-NotAllowed,
    InformationNotAvailable,
    MM-EventNotSupported,
    TargetCellOutsideGroupCallArea,
    SecureTransportError



FROM MAP-Errors {
```

```
    ccitt identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-Errors (10) version8 (8)}
;
Supported-MAP-Operations OPERATION ::= {updateLocation | cancelLocation | purgeMS |
    sendIdentification | updateGprsLocation | prepareHandover | sendEndSignal |
    processAccessSignalling | forwardAccessSignalling | prepareSubsequentHandover |
    sendAuthenticationInfo | authenticationFailureReport | checkIMEI | insertSubscriberData |
    deleteSubscriberData | reset | forwardCheckSS-Indication | restoreData | provideSubscriberInfo |
    anyTimeInterrogation | anyTimeSubscriptionInterrogation | anyTimeModification |
    sendRoutingInfoForGprs | failureReport |noteMsPresentForGprs | noteMM-Event |
    noteSubscriberDataModified | activateTraceMode | deactivateTraceMode | sendIMSI |
    sendRoutingInfo | provideRoamingNumber | resumeCallHandling | provideSIWFSNumber |
    siwfs-SignallingModify | setReportingState | statusReport | remoteUserFree | ist-Alert |
    ist-Command | registerSS | eraseSS | activateSS | deactivateSS | interrogateSS |
    processUnstructuredSS-Request | unstructuredSS-Request | unstructuredSS-Notify |
    registerPassword | getPassword | ss-InvocationNotification | registerCC-Entry | eraseCC-Entry |
    sendRoutingInfoForSM | mo-ForwardSM | mt-ForwardSM | reportSM-DeliveryStatus |
    alertServiceCentre | informServiceCentre | readyForSM | prepareGroupCall |
    processGroupCallSignalling | forwardGroupCallSignalling | sendGroupCallEndSignal |
    provideSubscriberLocation | sendRoutingInfoForLCS | subscriberLocationReport |
    secureTransportClass1 |secureTransportClass2 | secureTransportClass3 | secureTransportClass4}

-- location registration operation codes

updateLocation    UpdateLocation ::= localValue 2
cancelLocation    CancelLocation ::= localValue 3
purgeMS    PurgeMS ::= localValue 67
sendIdentification    SendIdentification ::= localValue 55

-- handover operation codes

prepareHandover    PrepareHandover ::= localValue 68
sendEndSignal    SendEndSignal ::= localValue 29
processAccessSignalling    ProcessAccessSignalling ::= localValue 33
forwardAccessSignalling    ForwardAccessSignalling ::= localValue 34
prepareSubsequentHandover    PrepareSubsequentHandover ::=
    localValue 69

-- authentication operation codes

sendAuthenticationInfo    SendAuthenticationInfo ::= localValue 56
authenticationFailureReport    AuthenticationFailureReport ::= localValue 15

-- IMEI MANAGEMENT operation codes

checkIMEI    CheckIMEI ::= localValue 43

-- subscriber management operation codes

insertSubscriberData    InsertSubscriberData ::= localValue 7
deleteSubscriberData    DeleteSubscriberData ::= localValue 8

-- fault recovery operation codes

reset    Reset ::= localValue 37
forwardCheckSS-Indication    ForwardCheckSS-Indication ::=
    localValue 38
restoreData    RestoreData ::= localValue 57

-- operation and maintenance operation codes

activateTraceMode    ActivateTraceMode ::= localValue 50
deactivateTraceMode    DeactivateTraceMode ::= localValue 51
sendIMSI    SendIMSI ::= localValue 58

-- call handling operation codes
```

```
sendRoutingInfo  SendRoutingInfo ::= localValue 22
provideRoamingNumber  ProvideRoamingNumber ::= localValue 4
resumeCallHandling  ResumeCallHandling ::= localValue 6
provideSIWFSNumber  ProvideSIWFSNumber ::= localValue 31
sIWFSSignallingModify  SIWFSSignallingModify ::= localValue 32
setReportingState  SetReportingState ::= localValue 73
statusReport StatusReport ::= localValue 74
remoteUserFree RemoteUserFree ::= localValue 75
istAlert IST-Alert ::= localValue 87
istCommand IST-Command ::= localValue 88
```

```
-- supplementary service handling operation codes
```

```
registerSS  RegisterSS ::= localValue 10
eraseSS  EraseSS ::= localValue 11
activateSS  ActivateSS ::= localValue 12
deactivateSS  DeactivateSS ::= localValue 13
interrogateSS  InterrogateSS ::= localValue 14
processUnstructuredSS-Request  ProcessUnstructuredSS-Request ::=
    localValue 59
unstructuredSS-Request  UnstructuredSS-Request ::= localValue 60
unstructuredSS-Notify  UnstructuredSS-Notify ::= localValue 61
registerPassword  RegisterPassword ::= localValue 17
getPassword  GetPassword ::= localValue 18
registerCC-Entry  RegisterCC-Entry ::= localValue 76
eraseCC-Entry  EraseCC-Entry ::= localValue 77
```

```
-- short message service operation codes
```

```
sendRoutingInfoForSM  SendRoutingInfoForSM ::= localValue 45
mo-forwardSM  MO-ForwardSM ::= localValue 46
mt-forwardSM  MT-ForwardSM ::= localValue 44
reportSM-DeliveryStatus  ReportSM-DeliveryStatus ::= localValue 47
informServiceCentre  InformServiceCentre ::= localValue 63
alertServiceCentre  AlertServiceCentre ::= localValue 64
readyForSM  ReadyForSM ::= localValue 66
```

```
-- provide subscriber info operation codes
```

```
provideSubscriberInfo  ProvideSubscriberInfo ::= localValue 70
```

```
-- any time interrogation operation codes
```

```
anyTimeInterrogation  AnyTimeInterrogation ::= localValue 71
```

```
-- any time information handling operation codes
```

```
anyTimeSubscriptionInterrogation  AnyTimeSubscriptionInterrogation ::= localValue 62
anyTimeModification  AnyTimeModification ::= localValue 65
```

```
-- subscriber data modification notification operation codes
```

```
noteSubscriberDataModified  NoteSubscriberDataModified ::= localValue 5
```

```
-- supplementary service invocation notification operation codes
```

```
ss-InvocationNotification  SS-InvocationNotification ::= localValue 72
```

```
--Group Call operation codes
```

```
prepareGroupCall  PrepareGroupCall ::= localValue 39
sendGroupCallEndSignal  SendGroupCallEndSignal ::= localValue 40
processGroupCallSignalling  ProcessGroupCallSignalling ::= localValue 41
forwardGroupCallSignalling  ForwardGroupCallSignalling ::= localValue 42
```

```
-- gprs location updating operation codes
```

```
updateGprsLocation  UpdateGprsLocation ::= localValue 23
```

```
-- gprs location information retrieval operation codes
```

```
sendRoutingInfoForGprs  SendRoutingInfoForGprs ::= localValue 24
```

```
-- failure reporting operation codes

failureReport  FailureReport ::= localValue 25

-- GPRS notification operation codes

noteMsPresentForGprs  NoteMsPresentForGprs ::= localValue 26

-- Location service operation codes

provideSubscriberLocation  ProvideSubscriberLocation ::= localValue 83
sendRoutingInfoForLCS  SendRoutingInfoForLCS ::= localValue 85
subscriberLocationReport  SubscriberLocationReport ::= localValue 86


-- Mobility Management operation codes

noteMM-Event  NoteMM-Event ::= localValue 89

-- Secure transport operation codes

secureTransportClass1 SecureTransportClass1 ::= localValue 78
secureTransportClass2 SecureTransportClass2 ::= localValue 79
secureTransportClass3 SecureTransportClass3 ::= localValue 80
secureTransportClass4 SecureTransportClass4 ::= localValue 81


-- generic error codes

systemFailure  SystemFailure ::= localValue 34
dataMissing  DataMissing ::= localValue 35
unexpectedDataValue  UnexpectedDataValue ::= localValue 36
facilityNotSupported  FacilityNotSupported ::= localValue 21
incompatibleTerminal  IncompatibleTerminal ::= localValue 28
resourceLimitation  ResourceLimitation ::= localValue 51


-- identification and numbering error codes

unknownSubscriber  UnknownSubscriber ::= localValue 1
numberChanged  NumberChanged ::= localValue 44
unknownMSC  UnknownMSC ::= localValue 3
unidentifiedSubscriber  UnidentifiedSubscriber ::= localValue 5
unknownEquipment  UnknownEquipment ::= localValue 7


-- subscription error codes

roamingNotAllowed  RoamingNotAllowed ::= localValue 8
illegalSubscriber  IllegalSubscriber ::= localValue 9
illegalEquipment  IllegalEquipment ::= localValue 12
bearerServiceNotProvisioned  BearerServiceNotProvisioned ::=
    localValue 10
teleserviceNotProvisioned  TeleserviceNotProvisioned ::=
    localValue 11


-- handover error codes

noHandoverNumberAvailable  NoHandoverNumberAvailable ::=
    localValue 25
subsequentHandoverFailure  SubsequentHandoverFailure ::=
    localValue 26
targetCellOutsideGroupCallArea  TargetCellOutsideGroupCallArea ::=
    localValue 42


-- operation and maintenance error codes

tracingBufferFull  TracingBufferFull ::= localValue 40


-- call handling error codes
```

```
noRoamingNumberAvailable  NoRoamingNumberAvailable ::= localValue 39
absentSubscriber  AbsentSubscriber ::= localValue 27
busySubscriber  BusySubscriber ::= localValue 45
noSubscriberReply  NoSubscriberReply ::= localValue 46
callBarred  CallBarred ::= localValue 13
forwardingFailed  ForwardingFailed ::= localValue 47
or-NotAllowed  OR-NotAllowed ::= localValue 48
forwardingViolation  ForwardingViolation ::= localValue 14
cug-Reject  CUG-Reject ::= localValue 15

-- any time interrogation error codes

ati-NotAllowed  ATI-NotAllowed ::= localValue 49

-- any time information handling error codes
atsi-NotAllowed  ATSI-NotAllowed ::= localValue 60
atm-NotAllowed  ATM-NotAllowed ::= localValue 61
informationNotAvailable  InformationNotAvailable ::= localValue 62

-- Group Call error codes
noGroupCallNumberAvailable  NoGroupCallNumberAvailable ::= localValue 50

-- supplementary service error codes

illegalSS-Operation  IllegalSS-Operation ::= localValue 16
ss-ErrorStatus  SS-ErrorStatus ::= localValue 17
ss-NotAvailable  SS-NotAvailable ::= localValue 18
ss-SubscriptionViolation  SS-SubscriptionViolation ::= localValue 19
ss-Incompatibility  SS-Incompatibility ::= localValue 20
unknownAlphabet  UnknownAlphabet ::= localValue 71
ussd-Busy  USSD-Busy ::= localValue 72
pw-RegistrationFailure  PW-RegistrationFailure ::= localValue 37
negativePW-Check  NegativePW-Check ::= localValue 38
numberOfPW-AttemptsViolation  NumberOfPW-AttemptsViolation ::=
    localValue 43
shortTermDenial  ShortTermDenial ::= localValue 29
longTermDenial  LongTermDenial ::= localValue 30

-- short message service error codes

subscriberBusyForMT-SMS  SubscriberBusyForMT-SMS ::= localValue 31
sm-DeliveryFailure  SM-DeliveryFailure ::= localValue 32
messageWaitingListFull  MessageWaitingListFull ::= localValue 33
absentsubscriberSM  AbsentSubscriberSM ::= localValue 6

-- location service error codes

unauthorizedRequestingNetwork  UnauthorizedRequestingNetwork::= localValue 52
unauthorizedLCSClient  UnauthorizedLCSClient ::= localValue 53
positionMethodFailure  PositionMethodFailure ::= localValue 54
unknownOrUnreachableLCSClient  UnknownOrUnreachableLCSClient ::= localValue 58

-- Mobility Management error codes
mm-EventNotSupported  MM-EventNotSupported ::= localValue 59

-- Secure transport error codes

secureTransportError SecureTransportError ::= localValue 4

-- The following operation codes are reserved for operations
-- existing in previous versions of the protocol
```

-- Operation Name	AC used	Oper. Code
--		
-- sendParameters	map-ac infoRetrieval (14) version1 (1)	local Value :9
-- processUnstructuredSS-Data	map-ac networkFunctionalSs (18) version1 (1)	local Value :19
-- performHandover	map-ac handoverControl (11) version1 (1)	local Value :28
-- performSubsequentHandover	map-ac handoverControl (11) version1 (1)	local Value :30
-- noteInternalHandover	map-ac handoverControl (11) version1 (1)	local Value :35
-- noteSubscriberPresent	map-ac mwdMngt (24) version1 (1)	local Value :48
-- alertServiceCentreWithoutResult	map-ac shortMsgAlert (23) version1 (1)	local Value :49
-- traceSubscriberActivity	map-ac handoverControl (11) version1 (1)	local Value :52
-- beginSubscriberActivity	map-ac networkFunctionalSs (18) version1 (1)	local Value :54

```
-- The following error codes are reserved for errors
-- existing in previous versions of the protocol
```

-- Error Name	AC used	Error Code
--		
-- unknownBaseStation	map-ac handoverControl (11) version1 (1)	local Value :2
-- invalidTargetBaseStation	map-ac handoverControl (11) version1 (1)	local Value :23
-- noRadioResourceAvailable	map-ac handoverControl (11) version1 (1)	local Value :24

```
END
```

17.6 MAP operations and error types

17.6.1 Mobile Service Operations

```
MAP-MobileServiceOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-MobileServiceOperations (5)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS

   -- location registration operations
   uUpdateLocation,
   cCancelLocation,
   pPurgeMS,
   sSendIdentification,

   -- gprs location registration operations
   uUpdateGprsLocation,

   -- subscriber information enquiry operations
   pProvideSubscriberInfo,

   -- any time information enquiry operations
   aAnyTimeInterrogation,

   -- any time information handling operations
   aAnyTimeSubscriptionInterrogation,
   aAnyTimeModification,

   -- subscriber data modification notification operations
   nNoteSubscriberDataModified,


   -- handover operations
   pPrepareHandover,
   sSendEndSignal,
   pProcessAccessSignalling,
   fForwardAccessSignalling,
   pPrepareSubsequentHandover,

   -- authentication management operations
   sSendAuthenticationInfo,
   aAuthenticationFailureReport,
```

```
    -- IMEI management operations
    cCheckIMEI,

    -- subscriber management operations
    iInsertSubscriberData,
    dDeleteSubscriberData,

    -- fault recovery operations
    rReset,
    fForwardCheckSS-Indication,
    rRestoreData,

-- gprs location information retrieval operations
    sSendRoutingInfoForGprs,

    -- failure reporting operations
    fFailureReport,

    -- gprs notification operations
    nNoteMsPresentForGprs,

    -- Mobility Management operations
    nNoteMM-Event




;

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)} TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   sSystemFailure,
   dDataMissing,
   uUnexpectedDataValue,
   uUnknownSubscriber,
   uUnknownMSC,
   uUnidentifiedSubscriber,
   uUnknownEquipment,
   rRoamingNotAllowed,
   atiATI-NotAllowed,
   nNoHandoverNumberAvailable,
   sSubsequentHandoverFailure,
   aAbsentSubscriber,
   mmMM-EventNotSupported,
   atsiATSI-NotAllowed,
   atmATM-NotAllowed,
   bBearerServiceNotProvisioned,
   tTeleserviceNotProvisioned,
   cCallBarred,
   iIllegalSS-Operation,
   ssSS-ErrorStatus,
   ssSS-NotAvailable,
   ssSS-Incompatibility,
   ssSS-SubscriptionViolation,
   iInformationNotAvailable,
   tTargetCellOutsideGroupCallArea


FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

   UpdateLocationArg,
   UpdateLocationRes,
   CancelLocationArg,
   CancelLocationRes,
   PurgeMS-Arg,
   PurgeMS-Res,
   SendIdentificationArg,
   SendIdentificationRes,
   UpdateGprsLocationArg,
   UpdateGprsLocationRes,
   PrepareHO-Arg,
   PrepareHO-Res,
```

```
    ForwardAccessSignalling-Arg,
    ProcessAccessSignalling-Arg,
    SendEndSignal-Arg,
    SendEndSignal-Res,
    PrepareSubsequentHO-Res,
    PrepareSubsequentHO-Arg,
    SendAuthenticationInfoArg,
    SendAuthenticationInfoRes,
    AuthenticationFailureReportArg,
    AuthenticationFailureReportRes,
    EquipmentStatus,
    InsertSubscriberDataArg,
    InsertSubscriberDataRes,
    DeleteSubscriberDataArg,
    DeleteSubscriberDataRes,
    ResetArg,
    RestoreDataArg,
    RestoreDataRes,
    ProvideSubscriberInfoArg,
    ProvideSubscriberInfoRes,
    AnyTimeSubscriptionInterrogationArg,
    AnyTimeSubscriptionInterrogationRes,
    AnyTimeModificationArg,
    AnyTimeModificationRes,
    NoteSubscriberDataModifiedArg,
    NoteSubscriberDataModifiedRes,
    AnyTimeInterrogationArg,
    AnyTimeInterrogationRes,
    SendRoutingInfoForGprsArg,
    SendRoutingInfoForGprsRes,
    FailureReportArg,
    FailureReportRes,
    NoteMsPresentForGprsArg,
    NoteMsPresentForGprsRes,
    NoteMM-EventArg,
    NoteMM-EventRes


FROM MAP-MS-DataTypes {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-MS-DataTypes (11) version8 (8)}

    IMEI
FROM MAP-CommonDataTypes {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}
;


-- location registration operations

uUpdateLocation ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          updateLocationArg                    UpdateLocationArg
     RESULT
          updateLocationRes                    UpdateLocationRes
     ERRORS {
          sSystemFailure |
          dDataMissing |
          uUnexpectedDataValue |
          uUnknownSubscriber |
          rRoamingNotAllowed}
     CODE local:2 }

cCancelLocation ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          cancelLocationArg                    CancelLocationArg
     RESULT
          cancelLocationRes                    CancelLocationRes
               -- optional
     ERRORS {
          dDataMissing |
          uUnexpectedDataValue}
     CODE local:3 }
```

```
pPurgeMS ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          purgeMS-Arg                    PurgeMS-Arg
     RESULT
          purgeMS-Res                    PurgeMS-Res
               -- optional
     ERRORS{
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:67 }
```

```
sSendIdentification ::= OPERATION ::= {                         --Timer s
     ARGUMENT
          sendIdentificationArg          SendIdentificationArg
     RESULT
          sendIdentificationRes          SendIdentificationRes
     ERRORS {
          dDataMissing |,
          uUnidentifiedSubscriber}
     CODE local:55 }
```

-- gprs location registration operations

```
uUpdateGprsLocation ::= OPERATION ::= {                         --Timer m
     ARGUMENT
          updateGprsLocationArg          UpdateGprsLocationArg
     RESULT
          updateGprsLocationRes          UpdateGprsLocationRes
     ERRORS {
          sSystemFailure |,
          uUnexpectedDataValue |,
          uUnknownSubscriber |,
          rRoamingNotAllowed}
     CODE local:23 }
```

-- subscriber information enquiry operations

```
pProvideSubscriberInfo ::= OPERATION ::= {                      --Timer m
     ARGUMENT
          provideSubscriberInfoArg       ProvideSubscriberInfoArg
     RESULT
          provideSubscriberInfoRes       ProvideSubscriberInfoRes
     ERRORS {
          dDataMissing |,
          uUnexpectedDataValue}
     CODE local:70 }
```

-- any time information enquiry operations

```
aAnyTimeInterrogation ::= OPERATION ::= {                       --Timer m
     ARGUMENT
          anyTimeInterrogationArg        AnyTimeInterrogationArg
     RESULT
          anyTimeInterrogationRes        AnyTimeInterrogationRes
     ERRORS {
          sSystemFailure, |
          atiATI-NotAllowed, |
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:71 }
```

-- any time information handling operations

```
aAnyTimeSubscriptionInterrogation ::= OPERATION ::= {                    --Timer m
     ARGUMENT
          anyTimeSubscriptionInterrogationArg  AnyTimeSubscriptionInterrogationArg
     RESULT
          anyTimeSubscriptionInterrogationRes  AnyTimeSubscriptionInterrogationRes
     ERRORS {
          atsiATSI-NotAllowed |
          dDataMissing |
          uUnexpectedDataValue |
          uUnknownSubscriber |
          bBearerServiceNotProvisioned |
          tTeleserviceNotProvisioned |
          cCallBarred |
          iIllegalSS-Operation |
          ssSS-NotAvailable |
          iInformationNotAvailable}
     CODE local:62 }
```

```
aAnyTimeModification ::= OPERATION ::= {                                 --Timer m
     ARGUMENT
          anyTimeModificationArg            AnyTimeModificationArg
     RESULT
          anyTimeModificationRes            AnyTimeModificationRes
     ERRORS {
          atmATM-NotAllowed |
          dDataMissing |
          uUnexpectedDataValue |
          uUnknownSubscriber |
          bBearerServiceNotProvisioned |
          tTeleserviceNotProvisioned |
          cCallBarred |
          iIllegalSS-Operation |
          ssSS-SubscriptionViolation |
          ssSS-ErrorStatus |
          ssSS-Incompatibility |
          iInformationNotAvailable}
     CODE local:65 }
```

-- subscriber data modification notification operations

```
nNoteSubscriberDataModified ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          noteSubscriberDataModifiedArg     NoteSubscriberDataModifiedArg
     RESULT
          noteSubscriberDataModifiedRes     NoteSubscriberDataModifiedRes
               -- optional
     ERRORS {
          dDataMissing |
          uUnexpectedDataValue |
          uUnknownSubscriber}
     CODE local:5 }
```

-- handover operations

```
pPrepareHandover ::= OPERATION ::= {                                     --Timer m
     ARGUMENT
          prepareHO-Arg                     PrepareHO-Arg
     RESULT
          prepareHO-Res                     PrepareHO-Res
     ERRORS {
          sSystemFailure |
          dDataMissing |
          uUnexpectedDataValue |
          nNoHandoverNumberAvailable |
          tTargetCellOutsideGroupCallArea }
     CODE local:68 }
```

```
sSendEndSignal ::= OPERATION ::= {                                       --Timer l
     ARGUMENT
          sendEndSignal-Arg                 SendEndSignal-Arg
     RESULT
          sendEndSignal-Res                 SendEndSignal-Res

     CODE local:29 }
```

```
pProcessAccessSignalling ::= OPERATION ::= {                              --Timer s
     ARGUMENT
          processAccessSignalling-Arg      ProcessAccessSignalling-Arg
     CODE local:33 }

fForwardAccessSignalling ::= OPERATION ::= {                              --Timer s
     ARGUMENT
          forwardAccessSignalling-Arg      ForwardAccessSignalling-Arg
     CODE local:34 }

pPrepareSubsequentHandover ::= OPERATION ::= {                            --Timer m
     ARGUMENT
          prepareSubsequentHO-Arg          PrepareSubsequentHO-Arg
     RESULT
          prepareSubsequentHO-Res          PrepareSubsequentHO-Res
     ERRORS {
          uUnexpectedDataValue, |
          dDataMissing, |
          uUnknownMSC, |
          sSubsequentHandoverFailure}
     CODE local:69 }
```

-- authentication management operations

```
sSendAuthenticationInfo ::= OPERATION ::= {                               --Timer m
     ARGUMENT
          sendAuthenticationInfoArg        SendAuthenticationInfoArg
          -- optional
          -- within a dialogue sendAuthenticationInfoArg shall not be present in
          -- subsequent invoke components. If received in a subsequent invoke component
          -- it shall be discarded.

     RESULT
          sendAuthenticationInfoRes        SendAuthenticationInfoRes
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:56 }

aAuthenticationFailureReport ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          authenticationFailureReportArg   AuthenticationFailureReportArg
     RESULT
          authenticationFailureReportRes   AuthenticationFailureReportRes
          -- optional
     ERRORS {
          sSystemFailure, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:15 }
```

-- IMEI management operations

```
cCheckIMEI ::= OPERATION ::={                                             --Timer m
     ARGUMENT
          imei                             IMEI
     RESULT
          equipmentStatus                  EquipmentStatus
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnknownEquipment}
     CODE local:43 }
```

-- subscriber management operations

```
iInsertSubscriberData ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          insertSubscriberDataArg          InsertSubscriberDataArg
     RESULT
          insertSubscriberDataRes          InsertSubscriberDataRes
          -- optional
     ERRORS {
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnidentifiedSubscriber}
     CODE local:7 }

dDeleteSubscriberData ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          deleteSubscriberDataArg          DeleteSubscriberDataArg
     RESULT
          deleteSubscriberDataRes          DeleteSubscriberDataRes
          -- optional
     ERRORS {
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnidentifiedSubscriber}
     CODE local:8 }

-- fault recovery operations

rReset ::= OPERATION ::= {                                         --Timer m
     ARGUMENT
          resetArg                         ResetArg
     CODE local:37 }

fForwardCheckSS-Indication ::= OPERATION ::= {                     --Timer s
     CODE local:38 }

rRestoreData ::= OPERATION ::= {                                   --Timer m
     ARGUMENT
          restoreDataArg                   RestoreDataArg
     RESULT
          restoreDataRes                   RestoreDataRes
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:57 }

-- gprs location information retrieval operations

sSendRoutingInfoForGprs ::= OPERATION ::= {                        --Timer m
     ARGUMENT
          sendRoutingInfoForGprsArg        SendRoutingInfoForGprsArg
     RESULT
          sendRoutingInfoForGprsRes        SendRoutingInfoForGprsRes
     ERRORS {
          aAbsentSubscriber, |
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber, |
          cCallBarred }
     CODE local:24 }

-- failure reporting operations

fFailureReport ::= OPERATION ::= {                                 --Timer m
     ARGUMENT
          failureReportArg                 FailureReportArg
     RESULT
          failureReportRes                 FailureReportRes
               -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:25 }
```

```
-- gprs notification operations

nNoteMsPresentForGprs ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          noteMsPresentForGprsArg          NoteMsPresentForGprsArg
     RESULT
          noteMsPresentForGprsRes          NoteMsPresentForGprsRes
               -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:26 }

nNoteMM-Event ::= OPERATION ::= {                                  --Timer m
     ARGUMENT
          noteMM-EventArg                  NoteMM-EventArg
     RESULT
          noteMM-EventRes                  NoteMM-EventRes
     ERRORS {
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber, |
          mmMM-EventNotSupported}
     CODE local:89 }

END
```

17.6.2 Operation and Maintenance Operations

```
MAP-OperationAndMaintenanceOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-OperationAndMaintenanceOperations (6)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS
   aActivateTraceMode,
   dDeactivateTraceMode,
   sSendIMSI
;

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   sSystemFailure,
   dDataMissing,
   uUnexpectedDataValue,
   fFacilityNotSupported,
   uUnknownSubscriber,
   uUnidentifiedSubscriber,
   tTracingBufferFull
FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

   ActivateTraceModeArg,
   ActivateTraceModeRes,
   DeactivateTraceModeArg,
   DeactivateTraceModeRes
FROM MAP-OM-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-OM-DataTypes (12) version8 (8)}

   ISDN-AddressString,
   IMSI
FROM MAP-CommonDataTypes {
```

```
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}
;

aActivateTraceMode ::= OPERATION ::= {                                  --Timer m
     ARGUMENT
          activateTraceModeArg ----------ActivateTraceModeArg
     RESULT
          activateTraceModeRes ----------ActivateTraceModeRes
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          fFacilityNotSupported, |
          uUnidentifiedSubscriber, |
          tTracingBufferFull}
     CODE local:50 }

dDeactivateTraceMode ::= OPERATION ::= {                                --Timer m
     ARGUMENT
          deactivateTraceModeArg ----------DeactivateTraceModeArg
     RESULT
          deactivateTraceModeRes ----------DeactivateTraceModeRes
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          fFacilityNotSupported, |
          uUnidentifiedSubscriber}
     CODE local:51 }

sSendIMSI ::= OPERATION ::= {                                           --Timer m
     ARGUMENT
          msisdn ----------ISDN-AddressString
     RESULT
          imsi ----------IMSI
     ERRORS {
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:58 }

END
```

17.6.3 Call Handling Operations

```
MAP-CallHandlingOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CallHandlingOperations (7)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS
   sSendRoutingInfo,
   pProvideRoamingNumber,
   rResumeCallHandling,
   pProvideSIWFSNumber,
   siwfs-SIWFSSignallingModify,
   sSetReportingState,
   sStatusReport,
   rRemoteUserFree,
   istIST-Alert,
   istIST-Command
;

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
```

```
    ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

    sSystemFailure,
    dDataMissing,
    uUnexpectedDataValue,
    fFacilityNotSupported,
    orOR-NotAllowed,
    uUnknownSubscriber,
    nNumberChanged,
    bBearerServiceNotProvisioned,
    tTeleserviceNotProvisioned,
    nNoRoamingNumberAvailable,
    aAbsentSubscriber,
    bBusySubscriber,
    nNoSubscriberReply,
    cCallBarred,
    fForwardingViolation,
    fForwardingFailed,
    cugCUG-Reject,
    rResourceLimitation,
    iIncompatibleTerminal,
    uUnidentifiedSubscriber

FROM MAP-Errors {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

    SendRoutingInfoArg,
    SendRoutingInfoRes,
    ProvideRoamingNumberArg,
    ProvideRoamingNumberRes,
    ResumeCallHandlingArg,
    ResumeCallHandlingRes,
    ProvideSIWFSNumberArg,
    ProvideSIWFSNumberRes,
    SIWFSSignallingModifyArg,
    SIWFSSignallingModifyRes,
    SetReportingStateArg,
    SetReportingStateRes,
    StatusReportArg,
    StatusReportRes,
    RemoteUserFreeArg,
    RemoteUserFreeRes,
    IST-AlertArg,
    IST-AlertRes,
    IST-CommandArg,
IST-CommandRes
FROM MAP-CH-DataTypes {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-CH-DataTypes (13) version8 (8)}

;

sSendRoutingInfo ::= OPERATION ::= {                                    --Timer m
-- The timer is set to the upper limit of the range if the GMSC supports pre-paging.
     ARGUMENT
          sendRoutingInfoArg                SendRoutingInfoArg
     RESULT
          sendRoutingInfoRes                SendRoutingInfoRes
     ERRORS {
          sSystemFailure |
          dDataMissing |
          uUnexpectedDataValue |
          fFacilityNotSupported |
          orOR-NotAllowed |
          uUnknownSubscriber |
          nNumberChanged |
          bBearerServiceNotProvisioned |
          tTeleserviceNotProvisioned |
          aAbsentSubscriber |
          bBusySubscriber |
          nNoSubscriberReply |
          cCallBarred |
          cugCUG-Reject |
          fForwardingViolation}
     CODE local:22 }
```

```
pProvideRoamingNumber ::= OPERATION ::= {                                --Timer m
-- The timer is set to the upper limit of the range if the HLR supports pre-paging.
     ARGUMENT
          provideRoamingNumberArg ----------ProvideRoamingNumberArg
     RESULT
          provideRoamingNumberRes ----------ProvideRoamingNumberRes
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          fFacilityNotSupported, |
          orOR-NotAllowed, |
          aAbsentSubscriber, |
          nNoRoamingNumberAvailable}
     CODE local:4 }
```

```
rResumeCallHandling ::= OPERATION ::= {                                  --Timer m
     ARGUMENT
          resumeCallHandlingArg ----------ResumeCallHandlingArg
     RESULT
          resumeCallHandlingRes ----------ResumeCallHandlingRes
          -- optional
     ERRORS {
          fForwardingFailed, |
          orOR-NotAllowed, |
          uUnexpectedDataValue, |
          dDataMissing }
     CODE local:6 }
```

```
pProvideSIWFSNumber ::= OPERATION ::= {                                  --Timer m
     ARGUMENT
          provideSIWFSNumberArg ----------ProvideSIWFSNumberArg
     RESULT
          provideSIWFSNumberRes ----------ProvideSIWFSNumberRes
     ERRORS {
          rResourceLimitation, |
          dDataMissing, |
          uUnexpectedDataValue, |
          sSystemFailure}
     CODE local:31 }
```

```
siwfs-SIWFSSignallingModify ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          sIWFSSignallingModifyArg ----------SIWFSSignallingModifyArg
     RESULT
          sIWFSSignallingModifyRes ----------SIWFSSignallingModifyRes
          -- optional
     ERRORS {
          rResourceLimitation, |
          dDataMissing, |
          uUnexpectedDataValue, |
          sSystemFailure}
     CODE local:32 }
```

```
sSetReportingState ::= OPERATION ::= {                                   --Timer m
     ARGUMENT
          setReportingStateArg ----------SetReportingStateArg
     RESULT
          setReportingStateRes ----------SetReportingStateRes
          -- optional
     ERRORS {
          sSystemFailure, |
          uUnidentifiedSubscriber, |
          uUnexpectedDataValue, |
          dDataMissing, |
          rResourceLimitation, |
          fFacilityNotSupported}
     CODE local:73 }
```

```
sStatusReport ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          statusReportArg ------------------------StatusReportArg
     RESULT
          statusReportRes ------------------------StatusReportRes
          -- optional
     ERRORS {
          uUnknownSubscriber, |
          sSystemFailure, |
          uUnexpectedDataValue, |
          dDataMissing}
     CODE local:74 }
```

```
rRemoteUserFree ::= OPERATION ::= {                                  --Timer ml
     ARGUMENT
          remoteUserFreeArg ----------------------RemoteUserFreeArg
     RESULT
          remoteUserFreeRes ----------------------RemoteUserFreeRes
     ERRORS {
          uUnexpectedDataValue, |
          dDataMissing, |
          iIncompatibleTerminal, |
          aAbsentSubscriber, |
          sSystemFailure, |
          bBusySubscriber}
     CODE local:75 }
```

```
istIST-Alert ::= OPERATION ::= {                                     --Timer m
     ARGUMENT
          istAlertArg ----------------------------IST-AlertArg
     RESULT
          istAlertRes ----------------------------IST-AlertRes
          -- optional
     ERRORS {
          uUnexpectedDataValue, |
          rResourceLimitation, |
          uUnknownSubscriber, |
          sSystemFailure, |
          fFacilityNotSupported}
     CODE local:87 }
```

```
istIST-Command ::= OPERATION ::= {                                   --Timer m
     ARGUMENT
          istCommandArg --------------------------IST-CommandArg
     RESULT
          istCommandRes --------------------------IST-CommandRes
          -- optional
     ERRORS {
          uUnexpectedDataValue, |
          rResourceLimitation, |
          uUnknownSubscriber, |
          sSystemFailure, |
          fFacilityNotSupported}
     CODE local:88 }
```

```
END
```

17.6.4 Supplementary service operations

```
MAP-SupplementaryServiceOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SupplementaryServiceOperations (8)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS
   rRegisterSS,
   eEraseSS,
   aActivateSS,
   dDeactivateSS,
   iInterrogateSS,
   pProcessUnstructuredSS-Request,
```

```
    uUnstructuredSS-Request,
    uUnstructuredSS-Notify,
    rRegisterPassword,
    gGetPassword,
    ssSS-InvocationNotification,
    rRegisterCC-Entry,
    eEraseCC-Entry
;

IMPORTS
    OPERATION
FROM Remote-Operations-Information-Objects {
    joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
    ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

    sSystemFailure,
    dDataMissing,
    uUnexpectedDataValue,
    uUnknownSubscriber,
    bBearerServiceNotProvisioned,
    tTeleserviceNotProvisioned,
    cCallBarred,
    iIllegalSS-Operation,
    ssSS-ErrorStatus,
    ssSS-NotAvailable,
    ssSS-SubscriptionViolation,
    ssSS-Incompatibility,
    pwPW-RegistrationFailure,
    nNegativePW-Check,
    nNumberOfPW-AttemptsViolation,
    uUnknownAlphabet,
    ussdUSSD-Busy,
    aAbsentSubscriber,
    iIllegalSubscriber,
    iIllegalEquipment,
    sShortTermDenial,
    lLongTermDenial,
    fFacilityNotSupported
FROM MAP-Errors {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

    RegisterSS-Arg,
    SS-Info,
    SS-ForBS-Code,
    InterrogateSS-Res,
    USSD-Arg,
    USSD-Res,
    Password,
    GuidanceInfo,
    SS-InvocationNotificationArg,
    SS-InvocationNotificationRes,
    RegisterCC-EntryArg,
    RegisterCC-EntryRes,
    EraseCC-EntryArg,
    EraseCC-EntryRes
FROM MAP-SS-DataTypes {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

    SS-Code
FROM MAP-SS-Code {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-SS-Code (15) version8 (8)}
;


-- supplementary service handling operations
```

```
rRegisterSS ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          registerSS-Arg                    RegisterSS-Arg
     RESULT
          ss-Info                           SS-Info
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          bBearerServiceNotProvisioned, |
          tTeleserviceNotProvisioned, |
          cCallBarred, |
          iIllegalSS-Operation, |
          ssSS-ErrorStatus, |
          ssSS-Incompatibility}
     CODE local:10 }

eEraseSS ::= OPERATION ::= {                                       --Timer m
     ARGUMENT
          ss-ForBS                          SS-ForBS-Code
     RESULT
          ss-Info                           SS-Info
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          bBearerServiceNotProvisioned, |
          tTeleserviceNotProvisioned, |
          cCallBarred, |
          iIllegalSS-Operation, |
          ssSS-ErrorStatus
          }
     CODE local:11 }

aActivateSS ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          ss-ForBS                          SS-ForBS-Code
     RESULT
          ss-Info                           SS-Info
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          bBearerServiceNotProvisioned, |
          tTeleserviceNotProvisioned, |
          cCallBarred, |
          iIllegalSS-Operation, |
          ssSS-ErrorStatus, |
          ssSS-SubscriptionViolation, |
          ssSS-Incompatibility, |
          nNegativePW-Check, |
          nNumberOfPW-AttemptsViolation}
     CODE local:12 }

dDeactivateSS ::= OPERATION ::= {                                  --Timer m
     ARGUMENT
          ss-ForBS                          SS-ForBS-Code
     RESULT
          ss-Info                           SS-Info
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          bBearerServiceNotProvisioned, |
          tTeleserviceNotProvisioned, |
          cCallBarred, |
          iIllegalSS-Operation, |
          ssSS-ErrorStatus, |
          ssSS-SubscriptionViolation, |
          nNegativePW-Check, |
          nNumberOfPW-AttemptsViolation}
     CODE local:13 }
```

```
iInterrogateSS ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          ss-ForBS                              SS-ForBS-Code
     RESULT
          interrogateSS-Res                     InterrogateSS-Res
     ERRORS {
          sSystemFailure,
          dDataMissing,
          uUnexpectedDataValue,
          bBearerServiceNotProvisioned,
          tTeleserviceNotProvisioned,
          cCallBarred,
          iIllegalSS-Operation,
          ssSS-NotAvailable}
     CODE local:14 }
```

```
pProcessUnstructuredSS-Request ::= OPERATION ::= {                    --Timer 10
minutes
     ARGUMENT
          ussd-Arg                              USSD-Arg
     RESULT
          ussd-Res                              USSD-Res
     ERRORS {
          sSystemFailure,
          dDataMissing,
          uUnexpectedDataValue,
          uUnknownAlphabet,
          cCallBarred}
     CODE local:59 }
```

```
uUnstructuredSS-Request ::= OPERATION ::= {                           --Timer ml
     ARGUMENT
          ussd-Arg                              USSD-Arg
     RESULT
          ussd-Res                              USSD-Res
          -- optional
     ERRORS {
          sSystemFailure,
          dDataMissing,
          uUnexpectedDataValue,
          aAbsentSubscriber,
          iIllegalSubscriber,
          iIllegalEquipment,
          uUnknownAlphabet,
          ussdUSSD-Busy}
     CODE local:60 }
```

```
uUnstructuredSS-Notify ::= OPERATION ::= {                            --Timer ml
     ARGUMENT
          ussd-Arg                              USSD-Arg
     RETURN RESULT TRUE
     ERRORS {
          sSystemFailure,
          dDataMissing,
          uUnexpectedDataValue,
          aAbsentSubscriber,
          iIllegalSubscriber,
          iIllegalEquipment,
          uUnknownAlphabet,
          ussdUSSD-Busy}
     CODE local:61 }
```

```
rRegisterPassword ::= OPERATION ::= {                                    --Timer ml
     ARGUMENT
          ss-Code                              SS-Code
     RESULT
          newPassword                          Password
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          cCallBarred, |
          ssSS-SubscriptionViolation, |
          pwPW-RegistrationFailure, |
          nNegativePW-Check, |
          nNumberOfPW-AttemptsViolation}
     LINKED {
          gGetPassword}
     CODE local:17 }
```

```
gGetPassword ::= OPERATION ::= {                                         --Timer m
     ARGUMENT
          guidanceInfo                         GuidanceInfo
     RESULT
          currentPassword                      Password
     CODE local:18 }
```

```
ssSS-InvocationNotification ::= OPERATION ::= {                          --Timer m
     ARGUMENT
          ss-InvocationNotificationArg         SS-InvocationNotificationArg
     RESULT
          ss-InvocationNotificationRes         SS-InvocationNotificationRes
          -- optional
     ERRORS {
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber}
     CODE local:72 }
```

```
rRegisterCC-Entry ::= OPERATION ::= {                                    --Timer m
     ARGUMENT
          registerCC-EntryArg                  RegisterCC-EntryArg
     RESULT
          registerCC-EntryRes                  RegisterCC-EntryRes
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          cCallBarred, |
          iIllegalSS-Operation, |
          ssSS-ErrorStatus, |
          ssSS-Incompatibility, |
          sShortTermDenial, |
          lLongTermDenial, |
          fFacilityNotSupported}
     CODE local:76 }
```

```
eEraseCC-Entry ::= OPERATION ::= {                                       --Timer m
     ARGUMENT
          eraseCC-EntryArg                     EraseCC-EntryArg
     RESULT
          eraseCC-EntryRes                     EraseCC-EntryRes
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          cCallBarred, |
          iIllegalSS-Operation, |
          ssSS-ErrorStatus}
     CODE local:77 }
```

```
END
```

17.6.5 Short message service operations

```
MAP-ShortMessageServiceOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ShortMessageServiceOperations (9)
   version8 (8)}
```

```
DEFINITIONS

::=

BEGIN

EXPORTS
   sSendRoutingInfoForSM,
   moMO-ForwardSM,
   mtMT-ForwardSM,
   rReportSM-DeliveryStatus,
   aAlertServiceCentre,
   iInformServiceCentre,
   rReadyForSM
;

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   sSystemFailure,
   dDataMissing,
   uUnexpectedDataValue,
   fFacilityNotSupported,
   uUnknownSubscriber,
   uUnidentifiedSubscriber,
   iIllegalSubscriber,
   iIllegalEquipment,
   tTeleserviceNotProvisioned,
   cCallBarred,
   sSubscriberBusyForMT-SMS,
   smSM-DeliveryFailure,
   mMessageWaitingListFull,
   aAbsentSubscriberSM
FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

   RoutingInfoForSM-Arg,
   RoutingInfoForSM-Res,
   MO-ForwardSM-Arg,
   MO-ForwardSM-Res,
   MT-ForwardSM-Arg,
   MT-ForwardSM-Res,
   ReportSM-DeliveryStatusArg,
   ReportSM-DeliveryStatusRes,
   AlertServiceCentreArg,
   InformServiceCentreArg,
   ReadyForSM-Arg,
   ReadyForSM-Res
FROM MAP-SM-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SM-DataTypes (16) version8 (8)}


;
```

```
sSendRoutingInfoForSM ::= OPERATION::= {                                --Timer m
     ARGUMENT
          routingInfoForSM-Arg                RoutingInfoForSM-Arg
     RESULT
          routingInfoForSM-Res                RoutingInfoForSM-Res
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          fFacilityNotSupported, |
          uUnknownSubscriber, |
          tTeleserviceNotProvisioned, |
          cCallBarred, |
          aAbsentSubscriberSM}
     CODE local:45 }
```

```
moMO-ForwardSM ::= OPERATION ::= {                                      --Timer ml
     ARGUMENT
          mo-forwardSM-Arg                    MO-ForwardSM-Arg
     RESULT
          mo-forwardSM-Res                    MO-ForwardSM-Res
          -- optional
     ERRORS {
          sSystemFailure, |
          uUnexpectedDataValue, |
          fFacilityNotSupported, |
          smSM-DeliveryFailure}
     CODE local:46 }
```

```
mtMT-ForwardSM ::= OPERATION ::= {                                      --Timer ml
     ARGUMENT
          mt-forwardSM-Arg                    MT-ForwardSM-Arg
     RESULT
          mt-forwardSM-Res                    MT-ForwardSM-Res
          -- optional
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue, |
          fFacilityNotSupported, |
          uUnidentifiedSubscriber, |
          iIllegalSubscriber, |
          iIllegalEquipment, |
          sSubscriberBusyForMT-SMS, |
          smSM-DeliveryFailure, |
          aAbsentSubscriberSM}
     CODE local:44 }
```

```
rReportSM-DeliveryStatus ::= OPERATION ::= {                            --Timer s
     ARGUMENT
          reportSM-DeliveryStatusArg          ReportSM-DeliveryStatusArg
     RESULT
          reportSM-DeliveryStatusRes          ReportSM-DeliveryStatusRes
          -- optional
     ERRORS {
          dDataMissing, |
          uUnexpectedDataValue, |
          uUnknownSubscriber, |
          mMessageWaitingListFull}
     CODE local:47 }
```

```
aAlertServiceCentre ::= OPERATION ::= {                                 --Timer s
     ARGUMENT
          alertServiceCentreArg               AlertServiceCentreArg
     RETURN RESULT TRUE
     ERRORS {
          sSystemFailure, |
          dDataMissing, |
          uUnexpectedDataValue}
     CODE local:64 }
```

```
iInformServiceCentre ::= OPERATION ::= {                                --Timer s
     ARGUMENT
          informServiceCentreArg              InformServiceCentreArg
     CODE local:63 }
```

```
rReadyForSM ::= OPERATION ::= {                                    --Timer m
    ARGUMENT
        readyForSM-Arg                    ReadyForSM-Arg
    RESULT
        readyForSM-Res                    ReadyForSM-Res
        -- optional
    ERRORS {
        dDataMissing, |
        uUnexpectedDataValue, |
        fFacilityNotSupported, |
        uUnknownSubscriber}
    CODE local:66 }

END
```

17.6.6 Errors

```
MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS

   -- generic errors
   sSystemFailure,
   dDataMissing,
   uUnexpectedDataValue,
   fFacilityNotSupported,
   iIncompatibleTerminal,
   rResourceLimitation,

   -- identification and numbering errors
   uUnknownSubscriber,
   nNumberChanged,
   uUnknownMSC,
   uUnidentifiedSubscriber,
   uUnknownEquipment,

   -- subscription errors
   rRoamingNotAllowed,
   iIllegalSubscriber,
   iIllegalEquipment,
   bBearerServiceNotProvisioned,
   tTeleserviceNotProvisioned,

   -- handover errors
   nNoHandoverNumberAvailable,
   sSubsequentHandoverFailure,
   tTargetCellOutsideGroupCallArea,

   -- operation and maintenance errors
   tTracingBufferFull,

   -- call handling errors
   orOR-NotAllowed,
   nNoRoamingNumberAvailable,
   bBusySubscriber,
   nNoSubscriberReply,
   aAbsentSubscriber,
   cCallBarred,
   fForwardingViolation,
   fForwardingFailed,
   cugCUG-Reject,

   -- any time interrogation errors
   atiATI-NotAllowed,

   -- any time information handling errors
   atsiATSI-NotAllowed,
   atmATM-NotAllowed,
   iInformationNotAvailable,
```

```
    -- supplementary service errors
    illegalSS-Operation,
    ssSS-ErrorStatus,
    ssSS-NotAvailable,
    ssSS-SubscriptionViolation,
    ssSS-Incompatibility,
    uUnknownAlphabet,
    ussdUSSD-Busy,
    pwPW-RegistrationFailure,
    nNegativePW-Check,
    nNumberOfPW-AttemptsViolation,
    sShortTermDenial,
    lLongTermDenial,

    -- short message service errors
    sSubscriberBusyForMT-SMS,
    smSM-DeliveryFailure,
    mMessageWaitingListFull,
    aAbsentSubscriberSM,

    -- Group Call errors
    nNoGroupCallNumberAvailable,

    -- location service errors
    uUnauthorizedRequestingNetwork,
    uUnauthorizedLCSClient,
    pPositionMethodFailure,
    uUnknownOrUnreachableLCSClient,

    -- Mobility Management errors
    mmMM-EventNotSupported,

    -- Secure transport errors
    sSecureTransportError


;

IMPORTS
   ERROR
FROM Remote-Operations-Information-Objects {joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)TCAPMessages {
  ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   SS-Status
FROM MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

   SS-IncompatibilityCause,
   PW-RegistrationFailureCause,
   SM-DeliveryFailureCause,
   SystemFailureParam,
   DataMissingParam,
   UnexpectedDataParam,
   FacilityNotSupParam,
   UnknownSubscriberParam,
   NumberChangedParam,
   UnidentifiedSubParam,
   RoamingNotAllowedParam,
   IllegalSubscriberParam,
   IllegalEquipmentParam,
   BearerServNotProvParam,
   TeleservNotProvParam,
   TracingBufferFullParam,
   NoRoamingNbParam,
   OR-NotAllowedParam,
   AbsentSubscriberParam,
   BusySubscriberParam,
   NoSubscriberReplyParam,
   CallBarredParam,
   ForwardingViolationParam,
   ForwardingFailedParam,
   CUG-RejectParam,
   ATI-NotAllowedParam,
   SubBusyForMT-SMS-Param,
   MessageWaitListFullParam,
   AbsentSubscriberSM-Param,
   ResourceLimitationParam,
```

```
   NoGroupCallNbParam,
   IncompatibleTerminalParam,
   ShortTermDenialParam,
   LongTermDenialParam,
   UnauthorizedRequestingNetwork-Param,
   UnauthorizedLCSClient-Param,
   PositionMethodFailure-Param,
   UnknownOrUnreachableLCSClient-Param,
   MM-EventNotSupported-Param,
   ATSI-NotAllowedParam,
   ATM-NotAllowedParam,
   IllegalSS-OperationParam,
   SS-NotAvailableParam,
   SS-SubscriptionViolationParam,
   InformationNotAvailableParam,
   TargetCellOutsideGCA-Param,
   SecureTransportErrorParam


FROM MAP-ER-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ER-DataTypes (17) version8 (8)}
;


-- generic errors

sSystemFailure ::= ERROR ::= {
     PARAMETER
          systemFailureParam                SystemFailureParam
          -- optional
     CODE local:34 }

dDataMissing ::= ERROR ::= {
     PARAMETER
          dataMissingParam                  DataMissingParam
          -- optional
          -- DdataMissingParam must not be used in version <3
     CODE local:35 }

uUnexpectedDataValue ::= ERROR ::={
     PARAMETER
          unexpectedDataParam               UnexpectedDataParam
          -- optional
          -- UunexpectedDataParam must not be used in version <3
     CODE local:36 }

fFacilityNotSupported ::= ERROR ::= {
     PARAMETER
          facilityNotSupParam               FacilityNotSupParam
          -- optional
          -- FfacilityNotSupParam must not be used in version <3
     CODE local:21 }

iIncompatibleTerminal ::= ERROR ::= {
     PARAMETER
          incompatibleTerminalParam         IncompatibleTerminalParam
          -- optional
     CODE local:28 }

rResourceLimitation ::= ERROR ::= {
     PARAMETER
          resourceLimitationParam           ResourceLimitationParam
          -- optional
     CODE local:51 }

-- identification and numbering errors

uUnknownSubscriber ::= ERROR ::= {
     PARAMETER
          unknownSubscriberParam            UnknownSubscriberParam
          -- optional
          -- UunknownSubscriberParam must not be used in version <3
     CODE local:1 }
```

```
nNumberChanged ::= ERROR ::= {
     PARAMETER
          numberChangedParam ----------NumberChangedParam
          -- optional
     CODE local:44 }

uUnknownMSC ::= ERROR ::= {
     CODE local:3 }

uUnidentifiedSubscriber ::= ERROR ::= {
     PARAMETER
          unidentifiedSubParam ----------UnidentifiedSubParam
          -- optional
          -- Uunidentifiedsubparam must not be used in version <3
     CODE local:5 }

uUnknownEquipment ::= ERROR ::= {
     CODE local:7 }
```

-- subscription errors

```
rRoamingNotAllowed ::= ERROR ::= {
     PARAMETER
          roamingNotAllowedParam ----------RoamingNotAllowedParam
     CODE local:8 }

iIllegalSubscriber ::= ERROR ::= {
     PARAMETER
          illegalSubscriberParam ----------IllegalSubscriberParam
          -- optional
          -- Iillegalsubscriberparam must not be used in version <3
     CODE local:9 }

iIllegalEquipment ::= ERROR ::= {
     PARAMETER
          illegalEquipmentParam ----------IllegalEquipmentParam
          -- optional
          -- Iillegalequipmentparam must not be used in version <3
     CODE local:12 }

bBearerServiceNotProvisioned ::= ERROR ::= {
     PARAMETER
          bearerServNotProvParam ----------BearerServNotProvParam
          -- optional
          -- Bbearerservnotprovparam must not be used in version <3
     CODE local:10 }

tTeleserviceNotProvisioned ::= ERROR ::= {
     PARAMETER
          teleservNotProvParam ----------TeleservNotProvParam
          -- optional
          -- Tteleservnotprovparam must not be used in version <3
     CODE local:11 }
```

-- handover errors

```
nNoHandoverNumberAvailable ::= ERROR ::= {
     CODE local:25 }

sSubsequentHandoverFailure ::= ERROR ::= {
     CODE local:26 }

tTargetCellOutsideGroupCallArea ::= ERROR ::= {
     PARAMETER
          targetCellOutsideGCA-Param ----------TargetCellOutsideGCA-Param
          -- optional
     CODE local:42 }
```

```
-- operation and maintenance errors

tTracingBufferFull ::= ERROR ::= {
    PARAMETER
        tracingBufferFullParam ---------- TracingBufferFullParam
        -- optional
    CODE local: 40 }

-- call handling errors

nNoRoamingNumberAvailable ::= ERROR ::= {
    PARAMETER
        noRoamingNbParam ---------- NoRoamingNbParam
        -- optional
    CODE local:39 }

aAbsentSubscriber ::= ERROR ::= {
    PARAMETER
        absentSubscriberParam ---------- AbsentSubscriberParam
        -- optional
        -- AabsentSubscriberParam must not be used in version <3
    CODE local:27 }

bBusySubscriber ::= ERROR ::= {
    PARAMETER
        busySubscriberParam ---------- BusySubscriberParam
        -- optional
    CODE local:45 }

nNoSubscriberReply ::= ERROR ::= {
    PARAMETER
        noSubscriberReplyParam ---------- NoSubscriberReplyParam
        -- optional
    CODE local:46 }

cCallBarred ::= ERROR ::= {
    PARAMETER
        callBarredParam ---------- CallBarredParam
        -- optional
    CODE local:13 }

fForwardingViolation ::= ERROR ::= {
    PARAMETER
        forwardingViolationParam ---------- ForwardingViolationParam
        -- optional
    CODE local:14 }

fForwardingFailed ::= ERROR ::= {
    PARAMETER
        forwardingFailedParam ---------- ForwardingFailedParam
        -- optional
    CODE local:47 }

cugCUG-Reject ::= ERROR ::= {
    PARAMETER
        cug-RejectParam ---------- CUG-RejectParam
        -- optional
    CODE local:15 }

orOR-NotAllowed ::= ERROR ::= {
    PARAMETER
        or-NotAllowedParam ---------- OR-NotAllowedParam
        -- optional
    CODE local:48 }

-- any time interrogation errors
atiATI-NotAllowed ::= ERROR ::= {
    PARAMETER
        ati-NotAllowedParam ---------- ATI-NotAllowedParam
        -- optional
    CODE local:49 }
```

```
-- any time information handling errors
atsiATSI-NotAllowed ::= ERROR ::= {
     PARAMETER
          atsi-NotAllowedParam ATSI-NotAllowedParam
          -- optional
     CODE local:60 }

atmATM-NotAllowed ::= ERROR ::= {
     PARAMETER
          atm-NotAllowedParam ATM-NotAllowedParam
          -- optional
     CODE local:61 }

iInformationNotAvailable ::= ERROR ::= {
     PARAMETER
          informationNotAvailableParam InformationNotAvailableParam
          -- optional
     CODE local:62 }

-- supplementary service errors

iIllegalSS-Operation ::= ERROR ::= {
     PARAMETER
          illegalSS-OperationParam IllegalSS-OperationParam
          -- optional
          -- IillegalSS-OperationParam must not be used in version <3
     CODE local:16 }

ssSS-ErrorStatus ::= ERROR ::= {
     PARAMETER
          ss-Status SS-Status
          -- optional
     CODE local:17 }

ssSS-NotAvailable ::= ERROR ::= {
     PARAMETER
          ss-NotAvailableParam SS-NotAvailableParam
          -- optional
          -- SSss-NotAvailableParam must not be used in version <3
     CODE local:18 }

ssSS-SubscriptionViolation ::= ERROR ::= {
     PARAMETER
          ss-SubscriptionViolationParam SS-SubscriptionViolationParam
          -- optional
          -- SSss-SubscriptionViolationNotAvailableParam must not be used in version <3
     CODE local:19 }

ssSS-Incompatibility ::= ERROR ::= {
     PARAMETER
          ss-IncompatibilityCause SS-IncompatibilityCause
          -- optional
     CODE local:20 }

uUnknownAlphabet ::= ERROR ::= {
     CODE local:71 }

ussdUSSD-Busy ::= ERROR ::= {
     CODE local:72 }

pwPW-RegistrationFailure ::= ERROR ::= {
     PARAMETER
          pw-RegistrationFailureCause PW-RegistrationFailureCause
     CODE local:37 }

nNegativePW-Check ::= ERROR ::= {
     CODE local:38 }

nNumberOfPW-AttemptsViolation ::= ERROR ::= {
     CODE local:43 }
```

```
sShortTermDenial ::= ERROR ::= {
     PARAMETER
          shortTermDenialParam ------------------ShortTermDenialParam
          -- optional
     CODE local:29 }
```

```
lLongTermDenial ::= ERROR ::= {
     PARAMETER
          longTermDenialParam ------------------LongTermDenialParam
          -- optional
     CODE local:30 }
```

-- short message service errors

```
sSubscriberBusyForMT-SMS ::= ERROR ::= {
     PARAMETER
          subBusyForMT-SMS-Param ------------SubBusyForMT-SMS-Param
          -- optional
     CODE local:31 }
```

```
smSM-DeliveryFailure ::= ERROR ::= {
     PARAMETER
          sm-DeliveryFailureCause ------------SM-DeliveryFailureCause
     CODE local:32 }
```

```
mMessageWaitingListFull ::= ERROR ::= {
     PARAMETER
          messageWaitListFullParam ------------MessageWaitListFullParam
          -- optional
     CODE local:33 }
```

```
aAbsentSubscriberSM ::= ERROR ::= {
     PARAMETER
          absentSubscriberSM-Param ------------AbsentSubscriberSM-Param
          -- optional
     CODE local:6 }
```

-- Group Call errors

```
nNoGroupCallNumberAvailable ::= ERROR ::= {
     PARAMETER
          noGroupCallNbParam ------------------NoGroupCallNbParam
          -- optional
     CODE local:50 }
```

-- location service errors

```
uUnauthorizedRequestingNetwork ::= ERROR ::= {
     PARAMETER
          unauthorizedRequestingNetwork-Param --UnauthorizedRequestingNetwork-Param
          -- optional
     CODE local:52 }
```

```
uUnauthorizedLCSClient ::= ERROR ::= {
     PARAMETER
          unauthorizedLCSClient-Param ------UnauthorizedLCSClient-Param
          -- optional
     CODE local:53 }
```

```
pPositionMethodFailure ::= ERROR ::= {
     PARAMETER
          positionMethodFailure-Param ------PositionMethodFailure-Param
          -- optional
     CODE local:54 }
```

```
uUnknownOrUnreachableLCSClient ::= ERROR ::= {
     PARAMETER
          unknownOrUnreachableLCSClient-Param --UnknownOrUnreachableLCSClient-Param
          -- optional
     CODE local:58 }
```

```
mmMM-EventNotSupported ::= ERROR ::= {
    PARAMETER
        mm-EventNotSupported-Param      MM-EventNotSupported-Param
        -- optional
    CODE local:59 }
```

```
   -- Secure transport errors
```

```
sSecureTransportError ::= ERROR ::= {
    PARAMETER
        secureTransportErrorParam       SecureTransportErrorParam
    CODE local:4 }
```

```
END
```

17.6.7 Group Call operations

```
MAP-Group-Call-Operations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Group-Call-Operations (22)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS
   pPrepareGroupCall,
   sSendGroupCallEndSignal,
   fForwardGroupCallSignalling,
   pProcessGroupCallSignalling
;

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   sSystemFailure,
   uUnexpectedDataValue,
   nNoGroupCallNumberAvailable
FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

   PrepareGroupCallArg,
   PrepareGroupCallRes,
   SendGroupCallEndSignalArg,
   SendGroupCallEndSignalRes,
   ForwardGroupCallSignallingArg,
   ProcessGroupCallSignallingArg
FROM MAP-GR-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-GR-DataTypes (23) version8 (8)}


;
```

```
pPrepareGroupCall ::= OPERATION ::= {                                   --Timer m
    ARGUMENT
        prepareGroupCallArg             PrepareGroupCallArg
    RESULT
        prepareGroupCallRes             PrepareGroupCallRes
    ERRORS {
        sSystemFailure, |
        nNoGroupCallNumberAvailable, |
        uUnexpectedDataValue}
    CODE local:39 }
```

```
sSendGroupCallEndSignal ::= OPERATION ::= {                              --Timer l
     ARGUMENT
          sendGroupCallEndSignalArg          SendGroupCallEndSignalArg
     RESULT
          sendGroupCallEndSignalRes          SendGroupCallEndSignalRes
     CODE local:40 }

pProcessGroupCallSignalling ::= OPERATION ::= {                          --Timer s
     ARGUMENT
          processGroupCallSignallingArg      ProcessGroupCallSignallingArg
     CODE local:41 }

fForwardGroupCallSignalling ::= OPERATION ::= {                          --Timer s
     ARGUMENT
          forwardGroupCallSignallingArg      ForwardGroupCallSignallingArg
     CODE local:42 }

END
```

17.6.8 Location service operations

```
MAP-LocationServiceOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-LocationServiceOperations (24)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS
   pProvideSubscriberLocation,
   sSendRoutingInfoForLCS,
   sSubscriberLocationReport
;

IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   sSystemFailure,
   dDataMissing,
   uUnexpectedDataValue,
   fFacilityNotSupported,
   uUnknownSubscriber,
   aAbsentSubscriber,
   uUnauthorizedRequestingNetwork,
   uUnauthorizedLCSClient,
   pPositionMethodFailure,
   rResourceLimitation,
   uUnknownOrUnreachableLCSClient,
   uUnidentifiedSubscriber,
   iIllegalEquipment,
   iIllegalSubscriber
FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

   RoutingInfoForLCS-Arg,
   RoutingInfoForLCS-Res,
   ProvideSubscriberLocation-Arg,
   ProvideSubscriberLocation-Res,
   SubscriberLocationReport-Arg,
   SubscriberLocationReport-Res
FROM MAP-LCS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-LCS-DataTypes (25) version8 (8)}
;
```

```
sSendRoutingInfoForLCS ::= OPERATION ::= {                              --Timer m
     ARGUMENT
          routingInfoForLCS-Arg            RoutingInfoForLCS-Arg
     RESULT
          routingInfoForLCS-Res            RoutingInfoForLCS-Res
     ERRORS {
          sSystemFailure |
          dDataMissing |
          uUnexpectedDataValue |
          fFacilityNotSupported |
          uUnknownSubscriber |
          aAbsentSubscriber |
          uUnauthorizedRequestingNetwork }
     CODE local:85 }

pProvideSubscriberLocation ::= OPERATION ::= {                          --Timer ml
     ARGUMENT
          provideSubscriberLocation-Arg    ProvideSubscriberLocation-Arg
     RESULT
          provideSubscriberLocation-Res    ProvideSubscriberLocation-Res
     ERRORS {
          sSystemFailure |
          dDataMissing |
          uUnexpectedDataValue |
          fFacilityNotSupported |
          uUnidentifiedSubscriber |
          iIllegalSubscriber |
          iIllegalEquipment |
          aAbsentSubscriber |
          uUnauthorizedRequestingNetwork |
          uUnauthorizedLCSClient |
          pPositionMethodFailure }
     CODE local:83 }

sSubscriberLocationReport ::= OPERATION ::= {                           --Timer m
     ARGUMENT
          subscriberLocationReport-Arg     SubscriberLocationReport-Arg
     RESULT
          subscriberLocationReport-Res     SubscriberLocationReport-Res
     ERRORS {
          sSystemFailure |
          dDataMissing |
          rResourceLimitation |
          uUnexpectedDataValue |
          uUnknownSubscriber |
          uUnauthorizedRequestingNetwork |
          uUnknownOrUnreachableLCSClient}
     CODE local:86 }


END
```

17.6.9 Secure transport operations

```
MAP-SecureTransportOperations {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SecureTransportOperations (26)
   version8 (8)}

DEFINITIONS

::=

BEGIN

EXPORTS
   sSecureTransportClass1,
   sSecureTransportClass2,
   sSecureTransportClass3,
   sSecureTransportClass4
;
```

```
IMPORTS
   OPERATION
FROM Remote-Operations-Information-Objects {
   joint-iso-itu-t remote-operations(4)
  informationObjects(5) version1(0)}TCAPMessages {
   ccitt recommendation q 773 modules (2) messages (1) version2 (2)}

   dDataMissing,
   sSecureTransportError,
   uUnexpectedDataValue

FROM MAP-Errors {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-Errors (10) version8 (8)}

   SecureTransportArg,
   SecureTransportRes

FROM MAP-ST-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ST-DataTypes (27) version8 (8)}
;

sSecureTransportClass1 ::= OPERATION::= {          --Timer shall be the same as for the
                                                   --securely transported operation
     ARGUMENT
          secureTransportArg                SecureTransportArg
     RESULT
          secureTransportRes                SecureTransportRes
     ERRORS {
          sSecureTransportError, |
          dDataMissing, |
          uUnexpectedDataValue}
     CODE local:78 }

sSecureTransportClass2 ::= OPERATION ::= {                              --Timer shall be
the same as for the
                                                   --securely transported operation
     ARGUMENT
          secureTransportArg                SecureTransportArg
     ERRORS {
          sSecureTransportError, |
          dDataMissing, |
          uUnexpectedDataValue}
     CODE local:79 }

sSecureTransportClass3 ::= OPERATION ::= {                              --Timer shall be
the same as for the
                                                   --securely transported operation
     ARGUMENT
          secureTransportArg                SecureTransportArg
     RESULT
          secureTransportRes                SecureTransportRes
     CODE local:80 }

sSecureTransportClass4 ::= OPERATION ::= {                              --Timer shall be
the same as for the
                                                   --securely transported operation
     ARGUMENT
          secureTransportArg                SecureTransportArg
     CODE local:81 }


END
```

17.7 MAP constants and data types

17.7.1 Mobile Service data types

```
MAP-MS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-MS-DataTypes (11) version8 (8)}
```

```
DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS

   -- location registration types
   UpdateLocationArg,
   UpdateLocationRes,
   CancelLocationArg,
   CancelLocationRes,
   PurgeMS-Arg,
   PurgeMS-Res,
   SendIdentificationArg,
   SendIdentificationRes,
   UpdateGprsLocationArg,
   UpdateGprsLocationRes,
   IST-SupportIndicator,
   SupportedLCS-CapabilitySets,

   -- gprs location registration types
   GSN-Address,

   -- handover types
   ForwardAccessSignalling-Arg,
   PrepareHO-Arg,
   PrepareHO-Res,
   PrepareSubsequentHO-Arg,
   PrepareSubsequentHO-Res,
   ProcessAccessSignalling-Arg,
   SendEndSignal-Arg,
   SendEndSignal-Res,


   -- authentication management types
   SendAuthenticationInfoArg,
   SendAuthenticationInfoRes,
   AuthenticationFailureReportArg,
   AuthenticationFailureReportRes,

   -- security management types
   EquipmentStatus,
   Kc,

   -- subscriber management types
   InsertSubscriberDataArg,
   InsertSubscriberDataRes,
   LSAIdentity,
   DeleteSubscriberDataArg,
   DeleteSubscriberDataRes,
   Ext-QoS-Subscribed,
   SubscriberData,
   ODB-Data,
   SubscriberStatus,
   ZoneCodeList,
   maxNumOfZoneCodes,
   O-CSI,
   D-CSI,
   O-BcsmCamelTDPCriteriaList,
   T-BCSM-CAMEL-TDP-CriteriaList,
   SS-CSI,
   ServiceKey,
   DefaultCallHandling,
   CamelCapabilityHandling,
   BasicServiceCriteria,
   SupportedCamelPhases,
   maxNumOfCamelTDPData,
   CUG-Index,
   CUG-Info,
   CUG-Interlock,
   InterCUG-Restrictions,
   IntraCUG-Options,
   NotificationToMSUser,
   QoS-Subscribed,
   IST-AlertTimerValue,
```

```
    T-CSI,
    T-BcsmTriggerDetectionPoint,
    APN,

    -- fault recovery types
    ResetArg,
    RestoreDataArg,
    RestoreDataRes,

-- provide subscriber info types
    GeographicalInformation,
    MS-Classmark2,
    GPRSMSClass,

    -- subscriber information enquiry types
    ProvideSubscriberInfoArg,
    ProvideSubscriberInfoRes,
    SubscriberInfo,
    LocationInformation,
    LocationInformationGPRS,
    RAIdentity,
    SubscriberState,
    GPRSChargingID,

    -- any time information enquiry types
    AnyTimeInterrogationArg,
    AnyTimeInterrogationRes,

    -- any time information handling types
    AnyTimeSubscriptionInterrogationArg,
    AnyTimeSubscriptionInterrogationRes,
    AnyTimeModificationArg,
    AnyTimeModificationRes,

    -- subscriber data modification notification types
    NoteSubscriberDataModifiedArg,
    NoteSubscriberDataModifiedRes,

    -- gprs location information retrieval types
    SendRoutingInfoForGprsArg,
    SendRoutingInfoForGprsRes,

    -- failure reporting types
    FailureReportArg,
    FailureReportRes,

    -- gprs notification types
    NoteMsPresentForGprsArg,
    NoteMsPresentForGprsRes,

    -- Mobility Management types
    NoteMM-EventArg,
    NoteMM-EventRes



;

IMPORTS
    maxNumOfSS,
    SS-SubscriptionOption,
    SS-List,
    SS-ForBS-Code,
    Password
FROM MAP-SS-DataTypes {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

    SS-Code
FROM MAP-SS-Code {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-SS-Code (15) version8 (8)}

    Ext-BearerServiceCode
FROM MAP-BS-Code {
    ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
    gsm-Network (1) modules (3) map-BS-Code (20) version8 (8)}

    Ext-TeleserviceCode
```

```
FROM MAP-TS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-TS-Code (19) version8 (8)}


   AddressString,
   ISDN-AddressString,
   ISDN-SubaddressString,
   FTN-AddressString,
   AccessNetworkSignalInfo,
   IMSI,
   IMEI,
   TMSI,
   HLR-List,
   LMSI,
   Identity,
   GlobalCellId,
   CellGlobalIdOrServiceAreaIdOrLAI,
   Ext-BasicServiceCode,
   NAEA-PreferredCI,
   EMLPP-Info,
   MC-SS-Info,
   SubscriberIdentity,
   AgeOfLocationInformation,
   LCSClientExternalID,
   LCSClientInternalID,
   Ext-SS-Status,
   LCSServiceTypeID



FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}

   AbsentSubscriberDiagnosticSM
FROM MAP-ER-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ER-DataTypes (17) version8 (8)}


;


-- location registration types
```

```
UpdateLocationArg ::= SEQUENCE {
    imsi                                IMSI,

    msc-Number                          [1] ISDN-AddressString,
    vlr-Number                          ISDN-AddressString,
    lmsi                                [10] LMSI OPTIONAL,
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ... ,
    vlr-Capability                      [6] VLR-Capability              OPTIONAL,
    informPreviousNetworkEntity         [11] NULL                       OPTIONAL,
    cs-LCS-NotSupportedByUE             [12] NULL                       OPTIONAL }
```

```
VLR-Capability ::= SEQUENCE{
    supportedCamelPhases                [0] SupportedCamelPhases        OPTIONAL,
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ... ,
    solsaSupportIndicator               [2] NULL                        OPTIONAL,
    istSupportIndicator                 [1] IST-SupportIndicator        OPTIONAL,
    superChargerSupportedInServingNetworkEntity   [3] SuperChargerInfo  OPTIONAL,
    longFTN-Supported                   [4]  NULL                       OPTIONAL,
    supportedLCS-CapabilitySets         [5]  SupportedLCS-CapabilitySets OPTIONAL }
```

```
SuperChargerInfo ::= CHOICE {
    sendSubscriberData                  [0] NULL,
    subscriberDataStored                [1] AgeIndicator }
```

```
AgeIndicator ::= OCTET STRING (SIZE (1..6))
```

```
    -- The internal structure of this parameter is implementation specific.
```

```
IST-SupportIndicator ::=  ENUMERATED {
    basicISTSupported                   (0),
    istCommandSupported                 (1),
    ...}
-- exception handling:
-- reception of values > 1 shall be mapped to ' istCommandSupported '
```

```
SupportedLCS-CapabilitySets ::= BIT STRING {
    lcsCapabilitySet1 (0),
    lcsCapabilitySet2 (1),
    lcsCapabilitySet3 (2) } (SIZE (2..16))
-- Core network signalling capability set1 indicates LCS Release98 or Release99 version.
-- Core network signalling capability set2 indicates LCS Release4.
-- Core network signalling capability set3 indicates LCS Release5 or later version.
-- A node shall mark in the BIT STRING all LCS capability sets it supports.
-- If no bit is set then the sending node does not support LCS.
-- If the parameter is not sent by an VLR then the VLR may support at most capability set1.
-- If the parameter is not sent by an SGSN then no support for LCS is assumed.
-- An SGSN is not allowed to indicate support of capability set1.
-- Other bits than listed above shall be discarded.
```

```
UpdateLocationRes ::= SEQUENCE {
    hlr-Number                          ISDN-AddressString,

    extensionContainer                  ExtensionContainer              OPTIONAL,
    ... }
```

```
CancelLocationArg ::= [3] SEQUENCE {
    identity                            Identity,
    cancellationType                    CancellationType                OPTIONAL,
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ...}
```

```
CancellationType ::= ENUMERATED {
    updateProcedure                     (0),
    subscriptionWithdraw                (1),
    ...}
    -- The HLR shall not send values other than listed above
```

```
CancelLocationRes ::= SEQUENCE {
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ...}
```

```
PurgeMS-Arg ::= [3] SEQUENCE {
    imsi                                IMSI,
    vlr-Number                          [0] ISDN-AddressString          OPTIONAL,
    sgsn-Number                         [1]  ISDN-AddressString         OPTIONAL,
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ...}
```

```
PurgeMS-Res ::= SEQUENCE {
    freezeTMSI                          [0]  NULL                       OPTIONAL,
    freezeP-TMSI                        [1]  NULL                       OPTIONAL,
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ...}
```

```
SendIdentificationArg ::= SEQUENCE {
    tmsi                                TMSI,
    numberOfRequestedVectors            NumberOfRequestedVectors        OPTIONAL,
    -- if segmentation is used, numberOfRequestedVectors shall be present in
    -- the first segment and shall not be present in subsequent segments. If received
    -- in a subsequent segment it shall be discarded.
    segmentationProhibited              NULL                            OPTIONAL,
    -- if segmentation is prohibited the previous VLR shall not send the result
    -- within a TC-CONTINUE message.
    extensionContainer                  ExtensionContainer              OPTIONAL,
    ...}
```

```
SendIdentificationRes ::= [3] SEQUENCE {
    imsi                            IMSI                            OPTIONAL,
    -- IMSI must be present if SendIdentificationRes is not segmented.
    -- If the TC-Continue segmentation option is taken the IMSI must be
    -- present in one segmented transmission of SendIdentificationRes.
    authenticationSetList           AuthenticationSetList           OPTIONAL,
    currentSecurityContext          [2]CurrentSecurityContext       OPTIONAL,
    extensionContainer              [3] ExtensionContainer          OPTIONAL,
    ...}
```

```
-- authentication management types
```

```
AuthenticationSetList ::= CHOICE {
    tripletList                     [0] TripletList,
    quintupletList                  [1] QuintupletList }
```

```
TripletList ::= SEQUENCE SIZE (1..5) OF
                                    AuthenticationTriplet
```

```
QuintupletList ::= SEQUENCE SIZE (1..5) OF
                                    AuthenticationQuintuplet
```

```
AuthenticationTriplet ::= SEQUENCE {
    rand                            RAND,
    sres                            SRES,
    kc                              Kc,
    ...}
```

```
AuthenticationQuintuplet ::= SEQUENCE {
    rand                            RAND,
    xres                            XRES,
    ck                              CK,
    ik                              IK,
    autn                            AUTN,
    ...}
```

```
CurrentSecurityContext ::= CHOICE {
    gsm-SecurityContextData         [0] GSM-SecurityContextData,
    umts-SecurityContextData        [1] UMTS-SecurityContextData }
```

```
GSM-SecurityContextData ::= SEQUENCE {
    kc                              Kc,
    cksn                            Cksn,
    ... }
```

```
UMTS-SecurityContextData ::= SEQUENCE {
    ck                              CK,
    ik                              IK,
    ksi                             KSI,
    ... }
```

```
RAND ::= OCTET STRING (SIZE (16))
```

```
SRES ::= OCTET STRING (SIZE (4))
```

```
Kc ::= OCTET STRING (SIZE (8))
```

```
XRES ::= OCTET STRING (SIZE (4..16))
```

```
CK ::= OCTET STRING (SIZE (16))
```

```
IK ::= OCTET STRING (SIZE (16))
```

```
AUTN ::= OCTET STRING (SIZE (16))
```

```
AUTS ::= OCTET STRING (SIZE (14))
```

```
Cksn ::= OCTET STRING (SIZE (1))
    -- The internal structure is defined in 3G TS 24.008
```

```
KSI ::= OCTET STRING (SIZE (1))
    -- The internal structure is defined in 3G TS 24.008
```

```
AuthenticationFailureReportArg ::= SEQUENCE {
     imsi                              IMSI,
     failureCause                      FailureCause,
     extensionContainer                ExtensionContainer                OPTIONAL,
     ... ,
     re-attempt                        BOOLEAN                           OPTIONAL,
     accessType                        AccessType                        OPTIONAL,
     rand                              RAND                              OPTIONAL,
     vlr-Number                        [0] ISDN-AddressString            OPTIONAL,
     sgsn-Number                       [1] ISDN-AddressString            OPTIONAL }
```

```
AccessType ::= ENUMERATED {
     call (0),
     emergencyCall (1),
     locationUpdating (2),
     supplementaryService (3),
     shortMessage (4),
     gprsAttach (5),
     routingAreaUpdating (6),
     serviceRequest (7),
     pdpContextActivation (8),
     pdpContextDeactivation (9),
     ...}
     -- exception handling:
     -- received values greater than 9 shall be ignored.
```

```
AuthenticationFailureReportRes ::= SEQUENCE {
     extensionContainer                ExtensionContainer                OPTIONAL,
     ...}
```

```
FailureCause ::= ENUMERATED {
     wrongUserResponse  (0),
     wrongNetworkSignature  (1)}
```

-- gprs location registration types

```
UpdateGprsLocationArg ::= SEQUENCE {
     imsi                              IMSI,
     sgsn-Number                       ISDN-AddressString,
     sgsn-Address                      GSN-Address,
     extensionContainer                ExtensionContainer                OPTIONAL,
     ... ,
     sgsn-Capability                   [0] SGSN-Capability               OPTIONAL,
     informPreviousNetworkEntity       [1]  NULL                         OPTIONAL,
     ps-LCS-NotSupportedByUE           [2]  NULL                         OPTIONAL }
```

```
SGSN-Capability ::= SEQUENCE{
     solsaSupportIndicator             NULL                              OPTIONAL,
     extensionContainer                [1] ExtensionContainer            OPTIONAL,
     ... ,
     superChargerSupportedInServingNetworkEntity   [2] SuperChargerInfo     OPTIONAL ,
     gprsEnhancementsSupportIndicator  [3] NULL                          OPTIONAL,
     supportedCamelPhases              [4] SupportedCamelPhases          OPTIONAL,
     supportedLCS-CapabilitySets       [5]  SupportedLCS-CapabilitySets  OPTIONAL }
```

```
GSN-Address ::= OCTET STRING (SIZE (5..17))
     -- Octets are coded according to TS 3GPP TS 23.003 [17]
```

```
UpdateGprsLocationRes ::= SEQUENCE {
     hlr-Number                        ISDN-AddressString,
     extensionContainer                ExtensionContainer                OPTIONAL,
     ...}
```

-- handover types

```
ForwardAccessSignalling-Arg ::= [3] SEQUENCE {
     an-APDU                           AccessNetworkSignalInfo,
     integrityProtectionInfo           [0] IntegrityProtectionInformation  OPTIONAL,
     encryptionInfo                    [1] EncryptionInformation           OPTIONAL,
     keyStatus                         [2]  KeyStatus                    OPTIONAL,
     allowedGSM-Algorithms             [4]  AllowedGSM-Algorithms        OPTIONAL,
```

```
    allowedUMTS-Algorithms              [5]  AllowedUMTS-Algorithms       OPTIONAL,
    radioResourceInformation            [6] RadioResourceInformation      OPTIONAL,
    extensionContainer                  [3]  ExtensionContainer           OPTIONAL,
    ...,
    radioResourceList                   [7]  RadioResourceList            OPTIONAL,
    bssmap-ServiceHandover              [9]  BSSMAP-ServiceHandover       OPTIONAL,
    ranap-ServiceHandover               [8]  RANAP-ServiceHandover        OPTIONAL }
```

```
AllowedGSM-Algorithms ::= OCTET STRING (SIZE (1))
    -- internal structure is coded as Algorithm identifier octet from
    -- Permitted Algorithms defined in 3G TS 48.008
    -- A node shall mark all GSM algorithms that are allowed in MSC-B
```

```
AllowedUMTS-Algorithms ::= SEQUENCE {
    integrityProtectionAlgorithms       [0]  PermittedIntegrityProtectionAlgorithms
    OPTIONAL,
    encryptionAlgorithms                [1]  PermittedEncryptionAlgorithms  OPTIONAL,
    extensionContainer                  [2]  ExtensionContainer           OPTIONAL,
    ...}
```

```
PermittedIntegrityProtectionAlgorithms ::=
         OCTET STRING (SIZE (1..maxPermittedIntegrityProtectionAlgorithmsLength))
    -- Octets contain a complete PermittedIntegrityProtectionAlgorithms data type
    -- as defined in 3G TS 25.413, encoded according to the encoding scheme
    -- mandated by 3G TS 25.413.
    -- Padding bits are included, if needed, in the least significant bits of the
    -- last octet of the octet string.
```

```
maxPermittedIntegrityProtectionAlgorithmsLength INTEGER ::= 9
```

```
PermittedEncryptionAlgorithms ::=
         OCTET STRING (SIZE (1..maxPermittedEncryptionAlgorithmsLength))
    -- Octets contain a complete PermittedEncryptionAlgorithms data type
    -- as defined in 3G TS 25.413, encoded according to the encoding scheme
    -- mandated by 3G TS 25.413
    -- Padding bits are included, if needed, in the least significant bits of the
    -- last octet of the octet string.
```

```
maxPermittedEncryptionAlgorithmsLength INTEGER ::= 9
```

```
KeyStatus ::= ENUMERATED {
    old  (0),
    new  (1),
    ...}
    -- exception handling:
    -- received values in range 2-31 shall be treated as "old"
    -- received values greater than 31 shall be treated as "new"
```

```
PrepareHO-Arg ::= [3] SEQUENCE {
    targetCellId                        [0] GlobalCellId                  OPTIONAL,
    ho-NumberNotRequired                NULL                              OPTIONAL,
    targetRNCId                         [1] RNCId                         OPTIONAL,
    an-APDU                             [2] AccessNetworkSignalInfo       OPTIONAL,
    multipleBearerRequested             [3] NULL                          OPTIONAL,
    imsi                                [4] IMSI                          OPTIONAL,
    integrityProtectionInfo             [5] IntegrityProtectionInformation  OPTIONAL,
    encryptionInfo                      [6] EncryptionInformation          OPTIONAL,
    radioResourceInformation            [7] RadioResourceInformation      OPTIONAL,
    allowedGSM-Algorithms               [9]  AllowedGSM-Algorithms        OPTIONAL,
    allowedUMTS-Algorithms              [10] AllowedUMTS-Algorithms       OPTIONAL,
    radioResourceList                   [11] RadioResourceList            OPTIONAL,
    extensionContainer                  [8] ExtensionContainer            OPTIONAL,
    ... ,
    rab-Id                              [12] RAB-Id                       OPTIONAL,
    bssmap-ServiceHandover              [13] BSSMAP-ServiceHandover       OPTIONAL,
    ranap-ServiceHandover               [14] RANAP-ServiceHandover        OPTIONAL
     }
```

```
BSSMAP-ServiceHandover ::= OCTET STRING (SIZE (1))
    -- Octets are coded according the Service Handover information element in
    -- 3G TS 48.008.
```

```
RANAP-ServiceHandover ::= OCTET STRING (SIZE (1))
     -- Octet contains a complete Service-Handover data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included in the least significant bits.
```

```
RadioResourceList ::= SEQUENCE SIZE (2.. maxNumOfRadioResources) OF
                                   RadioResource
```

```
RadioResource ::= SEQUENCE {
     radioResourceInformation            RadioResourceInformation,
     rab-Id                              RAB-Id,
     -- RAB Identity is needed to relate the radio resources with the radio access bearers.
     ...}
```

```
maxNumOfRadioResources  INTEGER ::= 7
```

```
PrepareHO-Res ::= [3] SEQUENCE {
     handoverNumber                      [0] ISDN-AddressString            OPTIONAL,
     relocationNumberList                [1]  RelocationNumberList         OPTIONAL,
     an-APDU                             [2]  AccessNetworkSignalInfo      OPTIONAL,
     multicallBearerInfo                 [3]  MulticallBearerInfo          OPTIONAL,
     multipleBearerNotSupported          NULL                              OPTIONAL,
     selectedUMTS-Algorithms             [5]  SelectedUMTS-Algorithms      OPTIONAL,
     chosenRadioResourceInformation      [6] ChosenRadioResourceInformation OPTIONAL,
     extensionContainer                  [4]  ExtensionContainer           OPTIONAL,
     ...}
```

```
SelectedUMTS-Algorithms ::= SEQUENCE {
     integrityProtectionAlgorithm        [0]  ChosenIntegrityProtectionAlgorithm   OPTIONAL,
     encryptionAlgorithm                 [1]  ChosenEncryptionAlgorithm    OPTIONAL,
     extensionContainer                  [2]  ExtensionContainer           OPTIONAL,
     ...}
```

```
ChosenIntegrityProtectionAlgorithm ::= OCTET STRING (SIZE (1))
     -- Octet contains a complete IntegrityProtectionAlgorithm data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included in the least significant bits.
```

```
ChosenEncryptionAlgorithm ::= OCTET STRING (SIZE (1))
     -- Octet contains a complete EncryptionAlgorithm data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included in the least significant bits.
```

```
ChosenRadioResourceInformation ::= SEQUENCE {
     chosenChannelInfo                   [0] ChosenChannelInfo             OPTIONAL,
     chosenSpeechVersion                 [1] ChosenSpeechVersion           OPTIONAL,
     ...}
```

```
ChosenChannelInfo ::= OCTET STRING (SIZE (1))
     -- Octets are coded according the Chosen Channel information element in 3G TS 48.008
```

```
ChosenSpeechVersion ::= OCTET STRING (SIZE (1))
     -- Octets are coded according the Speech Version (chosen) information element in 3G TS
     -- 48.008
```

```
PrepareSubsequentHO-Arg ::= [3] SEQUENCE {
     targetCellId                        [0] GlobalCellId                  OPTIONAL,
     targetMSC-Number                    [1] ISDN-AddressString,
     targetRNCId                         [2] RNCId                         OPTIONAL,
     an-APDU                             [3]  AccessNetworkSignalInfo      OPTIONAL,
     selectedRab-Id                      [4]  RAB-Id                       OPTIONAL,
     extensionContainer                  [5]  ExtensionContainer           OPTIONAL,
     ...}
```

```
PrepareSubsequentHO-Res ::= [3] SEQUENCE {
    an-APDU                          AccessNetworkSignalInfo,
    extensionContainer               [0]  ExtensionContainer          OPTIONAL,
    ...}
```

```
ProcessAccessSignalling-Arg ::= [3] SEQUENCE {
    an-APDU                          AccessNetworkSignalInfo,
    selectedUMTS-Algorithms          [1]  SelectedUMTS-Algorithms     OPTIONAL,
    selectedGSM-Algorithm            [2]  SelectedGSM-Algorithm       OPTIONAL,
    chosenRadioResourceInformation   [3] ChosenRadioResourceInformation OPTIONAL,
    selectedRab-Id                   [4] RAB-Id                       OPTIONAL,
    extensionContainer               [0]  ExtensionContainer          OPTIONAL,
    ...}
```

```
SelectedGSM-Algorithm ::= OCTET STRING (SIZE (1))
    -- internal structure is coded as Algorithm identifier octet from Chosen Encryption
    -- Algorithm defined in 3G TS 48.008
    -- A node shall mark only the selected GSM algorithm
```

```
SendEndSignal-Arg ::= [3] SEQUENCE {
    an-APDU                          AccessNetworkSignalInfo,
    extensionContainer               [0]  ExtensionContainer          OPTIONAL,
    ...}
```

```
SendEndSignal-Res ::= SEQUENCE {
    extensionContainer               [0]  ExtensionContainer          OPTIONAL,
    ...}
```

```
RNCId ::= OCTET STRING (SIZE (7))
    -- The internal structure is defined as follows:
    -- octet 1 bits 4321             Mobile Country Code 1st digit
    --         bits 8765             Mobile Country Code 2nd digit
    -- octet 2 bits 4321             Mobile Country Code 3rd digit
    --         bits 8765             Mobile Network Code 3rd digit
    --                               or filler (1111) for 2 digit MNCs
    -- octet 3 bits 4321             Mobile Network Code 1st digit
    --         bits 8765             Mobile Network Code 2nd digit
    -- octets 4 and 5                Location Area Code according to 3G TS 24.008
    -- octets 6 and 7                RNC Id value according to 3G TS 25.413
```

```
RelocationNumberList ::= SEQUENCE SIZE (1..maxNumOfRelocationNumber) OF
                                     RelocationNumber
```

```
MulticallBearerInfo ::= INTEGER (1..maxNumOfRelocationNumber)
```

```
RelocationNumber ::= SEQUENCE {
    handoverNumber                   ISDN-AddressString,
    rab-Id                           RAB-Id,
    -- RAB Identity is needed to relate the calls with the radio access bearers.
    ...}
```

```
RAB-Id ::= INTEGER (1..maxNrOfRABs)
```

```
maxNrOfRABs INTEGER ::= 255
```

```
maxNumOfRelocationNumber  INTEGER ::= 7
```

```
RadioResourceInformation ::= OCTET STRING (SIZE (3..13))
    -- Octets are coded according the Channel Type information element in 3G TS 48.008
```

```
IntegrityProtectionInformation ::= OCTET STRING (SIZE (18..maxNumOfIntegrityInfo))
    -- Octets contain a complete IntegrityProtectionInformation data type
    -- as defined in 3G TS 25.413, encoded according to the encoding scheme
    -- mandated by 3G TS 25.413
    -- Padding bits are included, if needed, in the least significant bits of the
    -- last octet of the octet string.
```

```
maxNumOfIntegrityInfo INTEGER ::= 100
```

```
EncryptionInformation ::= OCTET STRING (SIZE (18..maxNumOfEncryptionInfo))
     -- Octets contain a complete EncryptionInformation data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included, if needed, in the least significant bits of the
     -- last octet of the octet string.
```

```
maxNumOfEncryptionInfo INTEGER ::= 100
```

-- authentication management types

```
SendAuthenticationInfoArg ::= SEQUENCE {
     imsi                                [0] IMSI,
     numberOfRequestedVectors            NumberOfRequestedVectors,
     segmentationProhibited              NULL                              OPTIONAL,
     -- if segmentation is prohibited the HLR shall not send the result within
     -- a TC-CONTINUE message.
     immediateResponsePreferred          [1] NULL                           OPTIONAL,
     -- if present, the HLR may send an immediate response with the available authentication
     -- vectors (see § 8.5.2 for more information).
     re-synchronisationInfo              Re-synchronisationInfo            OPTIONAL,
     extensionContainer                  [2] ExtensionContainer            OPTIONAL,
     ...,
     requestingNodeType                  [3] RequestingNodeType            OPTIONAL}
```

```
NumberOfRequestedVectors ::= INTEGER (1..5)
```

```
Re-synchronisationInfo ::= SEQUENCE {
     rand                                RAND,
     auts                                AUTS,
     ...}
```

```
SendAuthenticationInfoRes ::= [3] SEQUENCE {
     authenticationSetList               AuthenticationSetList             OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
RequestingNodeType ::= ENUMERATED {
     vlr  (0),
     sgsn  (1),
     ...}
     -- exception handling:
     -- received values in the range 2-15 shall be treated as "vlr"
     -- received values greater than 15 shall be treated as "sgsn"
```

-- security management types

```
EquipmentStatus ::= ENUMERATED {
     whiteListed  (0),
     blackListed  (1),
     greyListed  (2)}
```

-- subscriber management types

```
InsertSubscriberDataArg ::= SEQUENCE {
     imsi                                  [0] IMSI                              OPTIONAL,
     COMPONENTS OF                         SubscriberData,
     extensionContainer                    [14] ExtensionContainer               OPTIONAL,
     ... ,
     naea-PreferredCI                      [15] NAEA-PreferredCI                 OPTIONAL,
     -- naea-PreferredCI is included at the discretion of the HLR operator.
     gprsSubscriptionData                  [16] GPRSSubscriptionData             OPTIONAL,
     roamingRestrictedInSgsnDueToUnsupportedFeature [23]                         NULL
                                                                                 OPTIONAL,
     networkAccessMode                     [24] NetworkAccessMode                OPTIONAL,
     lsaInformation                        [25] LSAInformation                   OPTIONAL,
     lmu-Indicator                         [21] NULL                             OPTIONAL,
     lcsInformation                        [22] LCSInformation                   OPTIONAL,
          istAlertTimer                    [26] IST-AlertTimerValue              OPTIONAL,
          superChargerSupportedInHLR       [27] AgeIndicator                     OPTIONAL,
          mc-SS-Info                       [28] MC-SS-Info                       OPTIONAL,
     cs-AllocationRetentionPriority        [29] CS-AllocationRetentionPriority   OPTIONAL,
     sgsn-CAMEL-SubscriptionInfo           [17] SGSN-CAMEL-SubscriptionInfo      OPTIONAL,
     chargingCharacteristics               [18] ChargingCharacteristics          OPTIONAL
 }
     -- If the Network Access Mode parameter is sent, it shall be present only in
     -- the first sequence if seqmentation is used
```

```
CS-AllocationRetentionPriority ::= OCTET STRING (SIZE (1))
     -- This data type encodes each priority level defined in TS 23.107 as the binary value
     -- of the priority level.
```

```
IST-AlertTimerValue ::= INTEGER (15..255)
```

```
LCSInformation ::= SEQUENCE {
     gmlc-List [0]                         GMLC-List OPTIONAL,
     lcs-PrivacyExceptionList              [1]  LCS-PrivacyExceptionList      OPTIONAL,
     molr-List                             [2]  MOLR-List                     OPTIONAL,
     ...,
     add-lcs-PrivacyExceptionList          [3]  LCS-PrivacyExceptionList      OPTIONAL }
     -- add-lcs-PrivacyExceptionList may be sent only if lcs-PrivacyExceptionList is
     -- present and contains four instances of LCS-PrivacyClass. If the mentioned condition
     -- is not satisfied the receiving node shall discard add-lcs-PrivacyExceptionList.
     -- If an LCS-PrivacyClass is received both in lcs-PrivacyExceptionList and in
     -- add-lcs-PrivacyExceptionList with the same SS-Code, then the error unexpected
     -- data value shall be returned.
```

```
GMLC-List ::= SEQUENCE SIZE (1..maxNumOfGMLC) OF
                                           ISDN-AddressString
     -- if segmentation is used, the complete GMLC-List shall be sent in one segment
```

```
maxNumOfGMLC  INTEGER ::= 5
```

```
NetworkAccessMode ::= ENUMERATED {
     bothMSCAndSGSN                        (0),
     onlyMSC                               (1),
     onlySGSN                              (2),
     ...}
     -- if unknown values are received in NetworkAccessMode
     -- they shall be discarded.
```

```
GPRSDataList ::= SEQUENCE SIZE (1..maxNumOfPDP-Contexts) OF
                                           PDP-Context
```

```
maxNumOfPDP-Contexts  INTEGER ::= 50
```

```
PDP-Context ::= SEQUENCE {
     pdp-ContextId                         ContextId,
     pdp-Type                              [16] PDP-Type,
     pdp-Address                           [17] PDP-Address                   OPTIONAL,
     qos-Subscribed                        [18] QoS-Subscribed,
     vplmnAddressAllowed                   [19] NULL OPTIONAL,
     apn                                   [20] APN,
     extensionContainer                    [21] ExtensionContainer            OPTIONAL,
     ... ,
     ext-QoS-Subscribed                    [0] Ext-QoS-Subscribed             OPTIONAL,
     pdp-ChargingCharacteristics           [1] ChargingCharacteristics        OPTIONAL }
     -- qos-Subscribed shall be discarded if ext-QoS-Subscribed is received and supported
```

```
ContextId ::= INTEGER (1..maxNumOfPDP-Contexts)
```

```
GPRSSubscriptionData ::= SEQUENCE {
     completeDataListIncluded            NULL                                OPTIONAL,

          -- If segmentation is used, completeDataListIncluded may only be present in the
          -- first segment.
     gprsDataList                        [1]  GPRSDataList,
     extensionContainer                  [2] ExtensionContainer              OPTIONAL,
     ... }
```

```
SGSN-CAMEL-SubscriptionInfo ::= SEQUENCE {
     gprs-CSI                            [0]  GPRS-CSI                       OPTIONAL,
     mo-sms-CSI                          [1]  SMS-CSI                        OPTIONAL,
     extensionContainer                  [2]  ExtensionContainer             OPTIONAL,
     ...,
     mt-sms-CSI                          [3]  SMS-CSI                        OPTIONAL,
     mt-smsCAMELTDP-CriteriaList         [4]  MT-smsCAMELTDP-CriteriaList    OPTIONAL,
     mg-csi                              [5]  MG-CSI                         OPTIONAL
     }
```

```
GPRS-CSI ::= SEQUENCE {
     gprs-CamelTDPDataList               [0] GPRS-CamelTDPDataList           OPTIONAL,
     camelCapabilityHandling             [1] CamelCapabilityHandling         OPTIONAL,
     extensionContainer                  [2] ExtensionContainer              OPTIONAL,
     notificationToCSE                   [3]  NULL                           OPTIONAL,
     csi-Active                          [4]  NULL                           OPTIONAL,
     ...}
--   notificationToCSE and csi-Active shall not be present when GPRS-CSI is sent to SGSN.
--   They may only be included in ATSI/ATM ack/NSDC message.
--   GPRS-CamelTDPData and  camelCapabilityHandling shall be present in
--   the GPRS-CSI sequence.
--   If GPRS-CSI is segmented, gprs-CamelTDPDataList and camelCapabilityHandling shall be
--   present in the first segment
```

```
GPRS-CamelTDPDataList ::= SEQUENCE SIZE (1..maxNumOfCamelTDPData) OF
     GPRS-CamelTDPData
--   GPRS-CamelTDPDataList shall not contain more than one instance of
--   GPRS-CamelTDPData containing the same value for gprs-TriggerDetectionPoint.
```

```
GPRS-CamelTDPData ::= SEQUENCE {
     gprs-TriggerDetectionPoint          [0] GPRS-TriggerDetectionPoint,
     serviceKey                          [1] ServiceKey,
     gsmSCF-Address                      [2] ISDN-AddressString,
     defaultSessionHandling              [3] DefaultGPRS-Handling,
     extensionContainer                  [4] ExtensionContainer              OPTIONAL,
     ...
     }
```

```
DefaultGPRS-Handling ::= ENUMERATED {
     continueTransaction (0) ,
     releaseTransaction (1) ,
     ...}
-- exception handling:
-- reception of values in range 2-31 shall be treated as "continueTransaction"
-- reception of values greater than 31 shall be treated as "releaseTransaction"
```

```
GPRS-TriggerDetectionPoint ::= ENUMERATED {
     attach                                (1),
     attachChangeOfPosition                (2),
     pdp-ContextEstablishment              (11),
     pdp-ContextEstablishmentAcknowledgement  (12),
     pdp-ContextChangeOfPosition           (14),
     ... }
-- exception handling:
-- For GPRS-CamelTDPData sequences containing this parameter with any
-- other value than the ones listed the receiver shall ignore the whole
-- GPRS-CamelTDPDatasequence.
```

```
APN ::=  OCTET STRING (SIZE (2..63))
     -- Octets are coded according to TS 3GPP TS 23.003 [17]
```

```
PDP-Type ::= OCTET STRING (SIZE (2))
     -- Octets are coded according to TS 3GPP TS 29.060 [105]
```

```
PDP-Address ::= OCTET STRING (SIZE (1..16))
     -- Octets are coded according to TS 3GPP TS 29.060 [105]

     -- The possible size values are:
     -- 1-7 octets  X.25 address type
     --  4  octets  IPv4 address type
     -- 16  octets  Ipv6 address type
```

```
QoS-Subscribed ::= OCTET STRING (SIZE (3))
     -- Octets are coded according to TS 3GPP TS 24.008 [35].
```

```
Ext-QoS-Subscribed ::= OCTET STRING (SIZE (1..9))
     -- OCTET 1:
     --  Allocation/Retention Priority (This octet encodes each priority level defined in
     --     23.107 as the binary value of the priority level, declaration in 29.060)
     -- Octets 2-9 are coded according to 3G TS 24.008 Quality of Service Octets
     -- 6-13.
```

```
ChargingCharacteristics ::= OCTET STRING (SIZE (2))
     -- Octets are coded according to 3G TS 32.015.
```

```
LSAOnlyAccessIndicator ::= ENUMERATED {
     accessOutsideLSAsAllowed  (0),
     accessOutsideLSAsRestricted (1)}
```

```
LSADataList ::= SEQUENCE SIZE (1..maxNumOfLSAs) OF
                                        LSAData
```

```
maxNumOfLSAs  INTEGER ::= 20
```

```
LSAData ::= SEQUENCE {
     lsaIdentity                    [0] LSAIdentity,
     lsaAttributes                  [1] LSAAttributes,
     lsaActiveModeIndicator         [2] NULL                        OPTIONAL,
     extensionContainer             [3] ExtensionContainer          OPTIONAL,
     ...}
```

```
LSAInformation ::= SEQUENCE {
     completeDataListIncluded       NULL                            OPTIONAL,

          -- If segmentation is used, completeDataListIncluded may only be present in the
          -- first segment.
     lsaOnlyAccessIndicator         [1]  LSAOnlyAccessIndicator     OPTIONAL,
     lsaDataList                    [2]  LSADataList                OPTIONAL,
     extensionContainer             [3] ExtensionContainer          OPTIONAL,
     ...}
```

```
LSAIdentity ::= OCTET STRING (SIZE (3))
     -- Octets are coded according to TS 3GPP TS 23.003 [17]
```

```
LSAAttributes ::= OCTET STRING (SIZE (1))
     -- Octets are coded according to TS 3GPP TS 48.008 [49]
```

```
SubscriberData ::= SEQUENCE {
     msisdn                                [1] ISDN-AddressString            OPTIONAL,
     category                              [2] Category                      OPTIONAL,
     subscriberStatus                      [3] SubscriberStatus              OPTIONAL,
     bearerServiceList                     [4] BearerServiceList             OPTIONAL,
     -- The exception handling for reception of unsupported / not allocated
     -- bearerServiceCodes is defined in section 8.8.1
     teleserviceList                       [6] TeleserviceList               OPTIONAL,
     -- The exception handling for reception of unsupported / not allocated
     -- teleserviceCodes is defined in section 8.8.1
     provisionedSS                         [7] Ext-SS-InfoList               OPTIONAL,
     odb-Data                              [8] ODB-Data                      OPTIONAL,
     roamingRestrictionDueToUnsupportedFeature  [9] NULL                     OPTIONAL,
     regionalSubscriptionData              [10] ZoneCodeList                 OPTIONAL,
     vbsSubscriptionData                   [11] VBSDataList                  OPTIONAL,
     vgcsSubscriptionData                  [12] VGCSDataList                 OPTIONAL,
     vlrCamelSubscriptionInfo              [13] VlrCamelSubscriptionInfo     OPTIONAL
     }
```

```
Category ::= OCTET STRING (SIZE (1))
     -- The internal structure is defined in ~~CCITT~~ITU-T Rec Q.763.
```

```
SubscriberStatus ::= ENUMERATED {
     serviceGranted  (0),
     operatorDeterminedBarring  (1)}
```

```
BearerServiceList ::= SEQUENCE SIZE (1..maxNumOfBearerServices) OF
                                       Ext-BearerServiceCode
```

```
maxNumOfBearerServices  INTEGER ::= 50
```

```
TeleserviceList ::= SEQUENCE SIZE (1..maxNumOfTeleservices) OF
                                       Ext-TeleserviceCode
```

```
maxNumOfTeleservices  INTEGER ::= 20
```

```
ODB-Data ::= SEQUENCE {
     odb-GeneralData                   ODB-GeneralData,
     odb-HPLMN-Data                    ODB-HPLMN-Data                    OPTIONAL,
     extensionContainer                ExtensionContainer                OPTIONAL,
     ...}
```

```
ODB-GeneralData ::= BIT STRING {
     allOG-CallsBarred  (0),
     internationalOGCallsBarred  (1),
     internationalOGCallsNotToHPLMN-CountryBarred  (2),
     interzonalOGCallsBarred (6),
     interzonalOGCallsNotToHPLMN-CountryBarred (7),
     interzonalOGCallsAndInternationalOGCallsNotToHPLMN-CountryBarred (8),
     premiumRateInformationOGCallsBarred  (3),
     premiumRateEntertainementOGCallsBarred  (4),
     ss-AccessBarred  (5),
     allECT-Barred (9),
     chargeableECT-Barred (10),
     internationalECT-Barred (11),
     interzonalECT-Barred (12),
     doublyChargeableECT-Barred (13),
     multipleECT-Barred (14),
     allPacketOrientedServicesBarred (15),
     roamerAccessToHPLMN-AP-Barred  (16),
     roamerAccessToVPLMN-AP-Barred  (17),
     roamingOutsidePLMNOG-CallsBarred  (18),
     allIC-CallsBarred  (19),
     roamingOutsidePLMNIC-CallsBarred  (20),
     roamingOutsidePLMNICountryIC-CallsBarred  (21),
     roamingOutsidePLMN-Barred  (22),
     roamingOutsidePLMN-CountryBarred  (23),
     registrationAllCF-Barred  (24),
     registrationCFNotToHPLMN-Barred  (25),
     registrationInterzonalCF-Barred  (26),
     registrationInterzonalCFNotToHPLMN-Barred  (27)} (SIZE (15..32))
     -- exception handling: reception of unknown bit assignments in the
     -- ODB-GeneralData type shall be treated like unsupported ODB-GeneralData
     -- When the ODB-GeneralData type is removed from the HLR for a given subscriber,
     -- in NoteSubscriberDataModified operation sent toward the gsmSCF
     -- all bits shall be set to “0“.
```

```
ODB-HPLMN-Data ::= BIT STRING {
     plmn-SpecificBarringType1  (0),
     plmn-SpecificBarringType2  (1),
     plmn-SpecificBarringType3  (2),
     plmn-SpecificBarringType4  (3)} (SIZE (4..32))
     -- exception handling: reception of unknown bit assignments in the
     -- ODB-HPLMN-Data type shall be treated like unsupported ODB-HPLMN-Data
     -- When the ODB-HPLMN-Data type is removed from the HLR for a given subscriber,
     -- in NoteSubscriberDataModified operation sent toward the gsmSCF
     -- all bits shall be set to “0“.
```

```
Ext-SS-InfoList ::= SEQUENCE SIZE (1..maxNumOfSS) OF
                                    Ext-SS-Info
```

```
Ext-SS-Info ::= CHOICE {
     forwardingInfo                 [0] Ext-ForwInfo,
     callBarringInfo                [1] Ext-CallBarInfo,
     cug-Info                       [2] CUG-Info,
     ss-Data                        [3] Ext-SS-Data,
     emlpp-Info                     [4] EMLPP-Info}
```

```
Ext-ForwInfo ::= SEQUENCE {
     ss-Code                        SS-Code,
     forwardingFeatureList          Ext-ForwFeatureList,
     extensionContainer             [0] ExtensionContainer          OPTIONAL,
     ...}
```

```
Ext-ForwFeatureList ::= SEQUENCE SIZE (1..maxNumOfExt-BasicServiceGroups) OF
                                    Ext-ForwFeature
```

```
Ext-ForwFeature ::= SEQUENCE {
     basicService                   Ext-BasicServiceCode            OPTIONAL,
     ss-Status                      [4] Ext-SS-Status,
     forwardedToNumber              [5] ISDN-AddressString          OPTIONAL,
     -- When this data type is sent from an HLR which supports CAMEL Phase 2
     -- to a VLR that supports CAMEL Phase 2 the VLR shall not check the
     -- format of the number
     forwardedToSubaddress          [8] ISDN-SubaddressString       OPTIONAL,
     forwardingOptions              [6] Ext-ForwOptions             OPTIONAL,
     noReplyConditionTime           [7] Ext-NoRepCondTime           OPTIONAL,
     extensionContainer             [9] ExtensionContainer          OPTIONAL,
     ...,
     longForwardedToNumber          [10] FTN-AddressString          OPTIONAL }
```

```
Ext-ForwOptions ::= OCTET STRING (SIZE (1..5))

     -- OCTET 1:

     --  bit 8: notification to forwarding party
     --   0  no notification
     --   1  notification

     --  bit 7: redirecting presentation
     --   0 no presentation
     --   1  presentation

     --  bit 6: notification to calling party
     --   0  no notification
     --   1  notification

     --  bit 5: 0 (unused)

     --  bits 43: forwarding reason
     --   00  ms not reachable
     --   01  ms busy
     --   10  no reply
     --   11  unconditional

     -- bits 21: 00 (unused)

     -- OCTETS 2-5: reserved for future use. They shall be discarded if
     -- received and not understood.
```

```
Ext-NoRepCondTime ::= INTEGER (1..100)
    -- Only values 5-30 are used.
    -- Values in the ranges 1-4 and 31-100 are reserved for future use
    -- If received:
    --       values 1-4 shall be mapped on to value 5
    --       values 31-100 shall be mapped on to value 30
```

```
Ext-CallBarInfo ::= SEQUENCE {
    ss-Code                         SS-Code,
    callBarringFeatureList          Ext-CallBarFeatureList,
    extensionContainer              ExtensionContainer              OPTIONAL,
    ...}
```

```
Ext-CallBarFeatureList ::= SEQUENCE SIZE (1..maxNumOfExt-BasicServiceGroups) OF
                                    Ext-CallBarringFeature
```

```
Ext-CallBarringFeature ::= SEQUENCE {
    basicService                    Ext-BasicServiceCode            OPTIONAL,
    ss-Status                       [4] Ext-SS-Status,
    extensionContainer              ExtensionContainer              OPTIONAL,
    ...}
```

```
CUG-Info ::= SEQUENCE {
    cug-SubscriptionList            CUG-SubscriptionList,
    cug-FeatureList                 CUG-FeatureList                 OPTIONAL,
    extensionContainer              [0] ExtensionContainer          OPTIONAL,
    ...}
```

```
CUG-SubscriptionList ::= SEQUENCE SIZE (0..maxNumOfCUG) OF
                                    CUG-Subscription
```

```
CUG-Subscription ::= SEQUENCE {
    cug-Index CUG-Index,
    cug-Interlock                   CUG-Interlock,
    intraCUG-Options                IntraCUG-Options,
    basicServiceGroupList           Ext-BasicServiceGroupList       OPTIONAL,
    extensionContainer              [0] ExtensionContainer          OPTIONAL,
    ...}
```

```
CUG-Index ::= INTEGER (0..32767)
    -- The internal structure is defined in ETS 300 138.
```

```
CUG-Interlock ::= OCTET STRING (SIZE (4))
```

```
IntraCUG-Options ::= ENUMERATED {
    noCUG-Restrictions  (0),
    cugIC-CallBarred  (1),
    cugOG-CallBarred  (2)}
```

```
maxNumOfCUG  INTEGER ::= 10
```

```
CUG-FeatureList ::= SEQUENCE SIZE (1..maxNumOfExt-BasicServiceGroups) OF
                                    CUG-Feature
```

```
Ext-BasicServiceGroupList ::= SEQUENCE SIZE (1..maxNumOfExt-BasicServiceGroups) OF
                                    Ext-BasicServiceCode
```

```
maxNumOfExt-BasicServiceGroups  INTEGER ::= 32
```

```
CUG-Feature ::= SEQUENCE {
    basicService                    Ext-BasicServiceCode            OPTIONAL,
    preferentialCUG-Indicator       CUG-Index OPTIONAL,
    interCUG-Restrictions           InterCUG-Restrictions,
    extensionContainer              ExtensionContainer              OPTIONAL,
    ...}
```

```
InterCUG-Restrictions ::= OCTET STRING (SIZE (1))

     -- bits 876543: 000000 (unused)
     -- Exception handling:
     -- bits 876543 shall be ignored if received and not understood

     -- bits 21
     --   00  CUG only facilities
     --   01  CUG with outgoing access
     --   10  CUG with incoming access
     --   11  CUG with both outgoing and incoming access
```

```
Ext-SS-Data ::= SEQUENCE {
     ss-Code                         SS-Code,
     ss-Status [4] Ext-SS-Status,
     ss-SubscriptionOption           SS-SubscriptionOption           OPTIONAL,
     basicServiceGroupList           Ext-BasicServiceGroupList       OPTIONAL,
     extensionContainer              [5] ExtensionContainer          OPTIONAL,
     ...}
```

```
LCS-PrivacyExceptionList ::= SEQUENCE SIZE (1..maxNumOfPrivacyClass) OF
                                     LCS-PrivacyClass
```

```
maxNumOfPrivacyClass  INTEGER ::= 4
```

```
LCS-PrivacyClass ::= SEQUENCE {
     ss-Code                         SS-Code,
     ss-Status                       Ext-SS-Status,
     notificationToMSUser            [0] NotificationToMSUser        OPTIONAL,
     -- notificationToMSUser may be sent only for SS-codes callSessionRelated
     -- and callSessionUnrelated. If not received for SS-codes callSessionRelated
     -- and callSessionUnrelated,
     -- the default values according to 3G TS 23.271 shall be assumed.
     externalClientList              [1] ExternalClientList          OPTIONAL,
     -- externalClientList may be sent only for SS-code callSessionUnrelated to a
     -- visited node that does not support LCS Release 4 or later versions.
     -- externalClientList may be sent only for SS-codes callSessionUnrelated and
     -- callSessionRelated to a visited node that supports LCS Release 4 or later versions.
     plmnClientList                  [2] PLMNClientList              OPTIONAL,
     -- plmnClientList may be sent only for SS-code plmnoperator.
     extensionContainer              [3] ExtensionContainer          OPTIONAL,
     ...,
     ext-externalClientList          [4] Ext-ExternalClientList      OPTIONAL,
     -- Ext-externalClientList may be sent only if the visited node supports LCS Release 4 or
     -- later versions, the user did specify more than 5 clients, and White Book SCCP is used.
     serviceTypeList                 [5]  ServiceTypeList            OPTIONAL
     -- serviceTypeList may be sent only for SS-code serviceType and if the visited node
     -- supports LCS Release 5 or later versions.
     --
     -- if segmentation is used, the complete LCS-PrivacyClass shall be sent in one segment
}
```

```
ExternalClientList ::= SEQUENCE SIZE (0..maxNumOfExternalClient) OF
                                     ExternalClient
```

```
maxNumOfExternalClient  INTEGER ::= 5
```

```
PLMNClientList ::= SEQUENCE SIZE (1..maxNumOfPLMNClient) OF
                                     LCSClientInternalID
```

```
maxNumOfPLMNClient  INTEGER ::= 5
```

```
Ext-ExternalClientList ::= SEQUENCE SIZE (1..maxNumOfExt-ExternalClient) OF
                                     ExternalClient
```

```
maxNumOfExt-ExternalClient  INTEGER ::= 35
```

```
ExternalClient ::= SEQUENCE {
     clientIdentity                  LCSClientExternalID,
     gmlc-Restriction                [0] GMLC-Restriction            OPTIONAL,
     notificationToMSUser            [1] NotificationToMSUser        OPTIONAL,
     -- If notificationToMSUser is not received, the default value according to
     -- 3G TS 23.271 shall be assumed.
     extensionContainer              [2] ExtensionContainer          OPTIONAL,
     ... }
```

```
GMLC-Restriction ::= ENUMERATED {
     gmlc-List                          (0),
     home-Country                       (1) ,
     ... }
-- exception handling:
-- At reception of any other value than the ones listed the receiver shall ignore
-- GMLC-Restriction.
```

```
NotificationToMSUser ::= ENUMERATED {
     notifyLocationAllowed              (0),
     notifyAndVerify-LocationAllowedIfNoResponse   (1),
     notifyAndVerify-LocationNotAllowedIfNoResponse(2),
     ...,
     locationNotAllowed (3) }
-- exception handling:
-- At reception of any other value than the ones listed the receiver shall ignore
-- NotificationToMSUser.
```

```
ServiceTypeList ::= SEQUENCE SIZE (1..maxNumOfServiceType) OF
                                        ServiceType
```

```
maxNumOfServiceType  INTEGER ::= 32
```

```
ServiceType ::= SEQUENCE {
     serviceTypeIdentity                LCSServiceTypeID,
     gmlc-Restriction                   [0] GMLC-Restriction            OPTIONAL,
     notificationToMSUser               [1] NotificationToMSUser        OPTIONAL,
     -- If notificationToMSUser is not received, the default value according to
     -- 3G TS 23.271 shall be assumed.
     extensionContainer                 [2] ExtensionContainer          OPTIONAL,
     ... }
```

```
MOLR-List ::= SEQUENCE SIZE (1..maxNumOfMOLR-Class) OF
                                        MOLR-Class
```

```
maxNumOfMOLR-Class  INTEGER ::= 3
```

```
MOLR-Class ::= SEQUENCE {
     ss-Code                            SS-Code,
     ss-Status                          Ext-SS-Status,
     extensionContainer                 [0] ExtensionContainer          OPTIONAL,
     ...}
```

```
ZoneCodeList ::= SEQUENCE SIZE (1..maxNumOfZoneCodes)
                                        OF ZoneCode
```

```
ZoneCode ::= OCTET STRING (SIZE (2))
     -- internal structure is defined in TS 3GPP TS 23.003 [17]
```

```
maxNumOfZoneCodes  INTEGER ::= 10
```

```
InsertSubscriberDataRes ::= SEQUENCE {
     teleserviceList                    [1] TeleserviceList             OPTIONAL,
     bearerServiceList                  [2] BearerServiceList           OPTIONAL,
     ss-List                            [3] SS-List                     OPTIONAL,
     odb-GeneralData                    [4] ODB-GeneralData             OPTIONAL,
     regionalSubscriptionResponse       [5] RegionalSubscriptionResponse  OPTIONAL,
     supportedCamelPhases               [6] SupportedCamelPhases        OPTIONAL,
     extensionContainer                 [7] ExtensionContainer          OPTIONAL,
     ...}
```

```
RegionalSubscriptionResponse ::= ENUMERATED {
     networkNode-AreaRestricted         (0),
     tooManyZoneCodes                   (1),
     zoneCodesConflict                  (2),
     regionalSubscNotSupported          (3)}
```

```
DeleteSubscriberDataArg ::= SEQUENCE {
     imsi                                [0] IMSI,
     basicServiceList                    [1] BasicServiceList              OPTIONAL,
     -- The exception handling for reception of unsupported/not allocated
     -- basicServiceCodes is defined in section 6.8.2
     ss-List                             [2] SS-List                       OPTIONAL,
     roamingRestrictionDueToUnsupportedFeature [4] NULL                    OPTIONAL,
     regionalSubscriptionIdentifier      [5] ZoneCode                      OPTIONAL,
     vbsGroupIndication                  [7] NULL                          OPTIONAL,
     vgcsGroupIndication                 [8] NULL   OPTIONAL,
     camelSubscriptionInfoWithdraw       [9] NULL   OPTIONAL,
     extensionContainer                  [6] ExtensionContainer OPTIONAL,
     ...,
     gprsSubscriptionDataWithdraw        [10] GPRSSubscriptionDataWithdraw OPTIONAL,
     roamingRestrictedInSgsnDueToUnsuppportedFeature [11] NULL             OPTIONAL,
     lsaInformationWithdraw              [12] LSAInformationWithdraw       OPTIONAL,
     gmlc-ListWithdraw                   [13] NULL                         OPTIONAL,
     istInformationWithdraw              [14] NULL                         OPTIONAL,
     specificCSI-Withdraw                [15] SpecificCSI-Withdraw         OPTIONAL }
```

```
SpecificCSI-Withdraw ::= BIT STRING {
     o-csi (0),
     ss-csi (1),
     tif-csi (2),
     d-csi (3),
     vt-csi (4),
     mo-sms-csi (5),
     m-csi (6),
     gprs-csi (7),
     t-csi (8),
     mt-sms-csi (9),
     mg-csi (10) } (SIZE(8..32))
-- exception handling:
-- bits 11 to 31 shall be ignored if received.
-- Bit 8 is only applicable for the NoteSubscriberDataModified operation.
```

```
GPRSSubscriptionDataWithdraw ::= CHOICE {
     allGPRSData                         NULL,
     contextIdList                       ContextIdList}
```

```
ContextIdList ::= SEQUENCE SIZE (1..maxNumOfPDP-Contexts) OF
                                         ContextId
```

```
LSAInformationWithdraw ::= CHOICE {
     allLSAData                          NULL,
     lsaIdentityList                     LSAIdentityList }
```

```
LSAIdentityList ::= SEQUENCE SIZE (1..maxNumOfLSAs) OF
                                         LSAIdentity
```

```
BasicServiceList ::= SEQUENCE SIZE (1..maxNumOfBasicServices) OF
                                         Ext-BasicServiceCode
```

```
maxNumOfBasicServices  INTEGER ::= 70
```

```
DeleteSubscriberDataRes ::= SEQUENCE {
     regionalSubscriptionResponse        [0] RegionalSubscriptionResponse  OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
VlrCamelSubscriptionInfo ::= SEQUENCE {
     o-CSI                               [0] O-CSI                         OPTIONAL,
     extensionContainer                  [1] ExtensionContainer            OPTIONAL,
     ...,
     ss-CSI                              [2] SS-CSI                        OPTIONAL,
     o-BcsmCamelTDP-CriteriaList         [4] O-BcsmCamelTDPCriteriaList    OPTIONAL,
     tif-CSI                             [3] NULL                          OPTIONAL,
     m-CSI                               [5] M-CSI                         OPTIONAL,
     mo-sms-CSI                          [6] SMS-CSI                       OPTIONAL,
     vt-CSI                              [7] T-CSI                         OPTIONAL,
     t-BCSM-CAMEL-TDP-CriteriaList       [8] T-BCSM-CAMEL-TDP-CriteriaList OPTIONAL,
     d-CSI                               [9] D-CSI                         OPTIONAL,
     mt-sms-CSI                          [10] SMS-CSI                      OPTIONAL,
     mt-smsCAMELTDP-CriteriaList         [11] MT-smsCAMELTDP-CriteriaList  OPTIONAL
     }
```

```
MT-smsCAMELTDP-CriteriaList ::= SEQUENCE SIZE (1.. maxNumOfCamelTDPData) OF
     MT-smsCAMELTDP-Criteria
```

```
MT-smsCAMELTDP-Criteria ::= SEQUENCE {
     sms-TriggerDetectionPoint           SMS-TriggerDetectionPoint,
     tpdu-TypeCriterion                  [0]  TPDU-TypeCriterion            OPTIONAL,
     ... }
```

```
TPDU-TypeCriterion ::= SEQUENCE SIZE (1..maxNumOfTPDUTypes) OF
     MT-SMS-TPDU-Type
```

```
maxNumOfTPDUTypes INTEGER ::= 5
```

```
MT-SMS-TPDU-Type ::= ENUMERATED {
     sms-DELIVER                         (0),
     sms-SUBMIT-REPORT                   (1),
     sms-STATUS-REPORT                   (2),
     ... }

--   exception handling:
--   For TPDU-TypeCriterion sequences containing this parameter with any
--   other value than the ones listed above the receiver shall ignore
--   the whole TPDU-TypeCriterion sequence.
--   In CAMEL phase 4, sms-SUBMIT-REPORT shall not be used and a received TPDU-TypeCriterion
--   sequence containing sms-SUBMIT-REPORT shall be wholly ignored.
```

```
D-CSI ::= SEQUENCE {
     dp-AnalysedInfoCriteriaList         [0] DP-AnalysedInfoCriteriaList   OPTIONAL,
     camelCapabilityHandling             [1] CamelCapabilityHandling       OPTIONAL,
     extensionContainer                  [2] ExtensionContainer            OPTIONAL,
     notificationToCSE                   [3]  NULL                         OPTIONAL,
     csi-Active                          [4]  NULL                         OPTIONAL,
     ...}
--   notificationToCSE and csi-Active shall not be present when D-CSI is sent to VLR/GMSC.
--   They may only be included in ATSI/ATM ack/NSDC message.
--   DP-AnalysedInfoCriteria and  camelCapabilityHandling shall be present in
--   the D-CSI sequence.
--   If D-CSI is segmented, dp-AnalysedInfoCriteriaList and camelCapabilityHandling shall be
--   present in the first segment
```

```
DP-AnalysedInfoCriteriaList  ::= SEQUENCE SIZE (1..maxNumOfDP-AnalysedInfoCriteria) OF
                                     DP-AnalysedInfoCriterium
```

```
maxNumOfDP-AnalysedInfoCriteria INTEGER ::= 10
```

```
DP-AnalysedInfoCriterium ::= SEQUENCE {
     dialledNumber                       ISDN-AddressString,
     serviceKey                          ServiceKey,
     gsmSCF-Address                      ISDN-AddressString,
     defaultCallHandling                 DefaultCallHandling,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
SS-CSI ::= SEQUENCE {
     ss-CamelData                        SS-CamelData,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...,
     notificationToCSE                   [0]  NULL                         OPTIONAL,
     csi-Active                          [1]  NULL                         OPTIONAL
--   notificationToCSE and csi-Active shall not be present when SS-CSI is sent to VLR.
--   They may only be included in ATSI/ATM ack/NSDC message.
}
```

```
SS-CamelData  ::= SEQUENCE {
     ss-EventList                        SS-EventList,
     gsmSCF-Address                      ISDN-AddressString,
     extensionContainer                  [0] ExtensionContainer            OPTIONAL,
     ...}
```

```
SS-EventList  ::= SEQUENCE SIZE (1..maxNumOfCamelSSEvents) OF SS-Code
     -- Actions for the following SS-Code values are defined in CAMEL Phase 3:
     -- ect                              SS-Code ::= '00110001'B
     -- multiPTY                         SS-Code ::= '01010001'B
     -- cd                               SS-Code ::= '00100100'B
     -- ccbs                             SS-Code ::= '01000100'B
     -- all other SS codes shall be ignored
     -- When SS-CSI is sent to the VLR, it shall not contain a marking for ccbs.
     -- If the VLR receives SS-CSI containing a marking for ccbs, the VLR shall discard the
     -- ccbs marking in SS-CSI.
```

```
maxNumOfCamelSSEvents INTEGER ::= 10
```

```
O-CSI ::= SEQUENCE {
     o-BcsmCamelTDPDataList              O-BcsmCamelTDPDataList,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...,
     camelCapabilityHandling             [0] CamelCapabilityHandling       OPTIONAL,
     notificationToCSE                   [1]  NULL                         OPTIONAL,
     csiActive                           [2]  NULL                         OPTIONAL}
--   notificationtoCSE and csiActive shall not be present when O-CSI is sent to VLR/GMSC.
--   They may only be included in ATSI/ATM ack/NSDC message.
```

```
O-BcsmCamelTDPDataList ::= SEQUENCE SIZE (1..maxNumOfCamelTDPData) OF
     O-BcsmCamelTDPData
     -- O-BcsmCamelTDPDataList shall not contain more than one instance of
     -- O-BcsmCamelTDPData containing the same value for o-BcsmTriggerDetectionPoint.
     -- For CAMEL Phase 2, this means that only one instance of O-BcsmCamelTDPData is allowed
     -- with o-BcsmTriggerDetectionPoint being equal to DP2.
```

```
maxNumOfCamelTDPData  INTEGER ::= 10
```

```
O-BcsmCamelTDPData ::= SEQUENCE {
     o-BcsmTriggerDetectionPoint         O-BcsmTriggerDetectionPoint,
     serviceKey                          ServiceKey,
     gsmSCF-Address                      [0] ISDN-AddressString,
     defaultCallHandling                 [1] DefaultCallHandling,
     extensionContainer                  [2] ExtensionContainer            OPTIONAL,
     ...
     }
```

```
ServiceKey ::= INTEGER (0..2147483647)
```

```
O-BcsmTriggerDetectionPoint ::= ENUMERATED {
     collectedInfo (2),
     ...,
     routeSelectFailure (4) }
     -- exception handling:
     -- For O-BcsmCamelTDPData sequences containing this parameter with any
     -- other value than the ones listed the receiver shall ignore the whole
     -- O-BcsmCamelTDPDatasequence.
     -- For O-BcsmCamelTDP-Criteria sequences containing this parameter with any
     -- other value than the ones listed the receiver shall ignore the whole
     -- O-BcsmCamelTDP-Criteria sequence.
```

```
O-BcsmCamelTDPCriteriaList ::= SEQUENCE SIZE (1..maxNumOfCamelTDPData) OF
     O-BcsmCamelTDP-Criteria
```

```
T-BCSM-CAMEL-TDP-CriteriaList ::= SEQUENCE SIZE (1..maxNumOfCamelTDPData) OF
     T-BCSM-CAMEL-TDP-Criteria
```

```
O-BcsmCamelTDP-Criteria ::= SEQUENCE {
     o-BcsmTriggerDetectionPoint         O-BcsmTriggerDetectionPoint,
     destinationNumberCriteria           [0] DestinationNumberCriteria     OPTIONAL,
     basicServiceCriteria                [1] BasicServiceCriteria          OPTIONAL,
     callTypeCriteria                    [2] CallTypeCriteria              OPTIONAL,
     ...,
     o-CauseValueCriteria                [3] O-CauseValueCriteria          OPTIONAL,
     extensionContainer                  [4] ExtensionContainer            OPTIONAL }
```

```
T-BCSM-CAMEL-TDP-Criteria ::= SEQUENCE {
     t-BCSM-TriggerDetectionPoint        T-BcsmTriggerDetectionPoint,
     basicServiceCriteria                [0] BasicServiceCriteria          OPTIONAL,
     t-CauseValueCriteria                [1] T-CauseValueCriteria          OPTIONAL,
     ... }
```

```
DestinationNumberCriteria  ::= SEQUENCE {
    matchType                          [0] MatchType,
    destinationNumberList              [1] DestinationNumberList         OPTIONAL,
    destinationNumberLengthList        [2] DestinationNumberLengthList   OPTIONAL,
    -- one or both of destinationNumberList and destinationNumberLengthList
    -- shall be present
    ...}
```

```
DestinationNumberList  ::= SEQUENCE SIZE   (1..maxNumOfCamelDestinationNumbers) OF
                                   ISDN-AddressString
    -- The receiving entity shall not check the format of a number in
    -- the dialled number list
```

```
DestinationNumberLengthList  ::= SEQUENCE SIZE (1..maxNumOfCamelDestinationNumberLengths)
OF
                                        INTEGER(1..maxNumOfISDN-AddressDigits)
```

```
BasicServiceCriteria  ::= SEQUENCE SIZE(1..maxNumOfCamelBasicServiceCriteria) OF
    Ext-BasicServiceCode
```

```
maxNumOfISDN-AddressDigits  INTEGER ::= 15
```

```
maxNumOfCamelDestinationNumbers  INTEGER ::= 10
```

```
maxNumOfCamelDestinationNumberLengths  INTEGER ::= 3
```

```
maxNumOfCamelBasicServiceCriteria  INTEGER ::= 5
```

```
CallTypeCriteria       ::= ENUMERATED {
    forwarded                          (0),
    notForwarded                       (1)}
```

```
MatchType       ::= ENUMERATED {
    inhibiting                         (0),
    enabling                           (1)}
```

```
O-CauseValueCriteria  ::= SEQUENCE SIZE(1..maxNumOfCAMEL-O-CauseValueCriteria) OF
    CauseValue
```

```
T-CauseValueCriteria  ::= SEQUENCE SIZE(1..maxNumOfCAMEL-T-CauseValueCriteria) OF
    CauseValue
```

```
maxNumOfCAMEL-O-CauseValueCriteria  INTEGER ::= 5
```

```
maxNumOfCAMEL-T-CauseValueCriteria  INTEGER ::= 5
```

```
CauseValue ::= OCTET STRING (SIZE(1))
-- Type extracted from Cause parameter in ITU-T Recommendation Q.763.
-- For the use of cause value refer to ITU-T Recommendation Q.850.
```

```
DefaultCallHandling ::= ENUMERATED {
    continueCall (0) ,
    releaseCall (1) ,
    ...}
    -- exception handling:
    -- reception of values in range 2-31 shall be treated as "continueCall"
    -- reception of values greater than 31 shall be treated as "releaseCall"
```

```
CamelCapabilityHandling ::= INTEGER(1..16)
    -- value 1 = CAMEL phase 1,
    -- value 2 = CAMEL phase 2,
    -- value 3 = CAMEL Phase 3,
    -- value 4 = CAMEL phase 4:
    -- reception of values greater than 4 shall be treated as CAMEL phase 4.
```

```
SupportedCamelPhases ::= BIT STRING {
    phase1 (0),
    phase2 (1),
    phase3 (2),
    phase4 (3)} (SIZE (1..16))
-- A node shall mark in the BIT STRING all CAMEL Phases it supports.
-- Other values than listed above shall be discarded.
```

```
SMS-CSI ::= SEQUENCE {
     sms-CAMEL-TDP-DataList              [0] SMS-CAMEL-TDP-DataList        OPTIONAL,
     camelCapabilityHandling             [1] CamelCapabilityHandling       OPTIONAL,
     extensionContainer                  [2] ExtensionContainer            OPTIONAL,
     notificationToCSE                   [3] NULL                          OPTIONAL,
     csi-Active                          [4] NULL                          OPTIONAL,
     ...}
--   notificationToCSE and csi-Active shall not be present
--   when MO-SMS-CSI or MT-SMS-CSI is sent to VLR or SGSN.
--   They may only be included in ATSI/ATM ack/NSDC message.
--   SMS-CAMEL-TDP-Data and  camelCapabilityHandling shall be present in
--   the SMS-CSI sequence.
--   If SMS-CSI is segmented, sms-CAMEL-TDP-DataList and camelCapabilityHandling shall be
--   present in the first segment
```

```
SMS-CAMEL-TDP-DataList ::= SEQUENCE SIZE (1..maxNumOfCamelTDPData) OF
     SMS-CAMEL-TDP-Data
--   SMS-CAMEL-TDP-DataList shall not contain more than one instance of
--   SMS-CAMEL-TDP-Data containing the same value for sms-TriggerDetectionPoint.
```

```
SMS-CAMEL-TDP-Data ::= SEQUENCE {
     sms-TriggerDetectionPoint           [0] SMS-TriggerDetectionPoint,
     serviceKey                          [1] ServiceKey,
     gsmSCF-Address                      [2] ISDN-AddressString,
     defaultSMS-Handling                 [3] DefaultSMS-Handling,
     extensionContainer                  [4] ExtensionContainer            OPTIONAL,
     ...
     }
```

```
SMS-TriggerDetectionPoint ::= ENUMERATED {
     sms-CollectedInfo (1),
     ...,
     sms-DeliveryRequest (2)
     }
--   exception handling:
--   For SMS-CAMEL-TDP-Data and MT-smsCAMELTDP-Criteria sequences containing this
--   parameter with any other value than the ones listed the receiver shall ignore
--   the whole sequence.
--
--   If this parameter is received with any other value than sms-CollectedInfo
--   in an SMS-CAMEL-TDP-Data sequence contained in mo-sms-CSI, then the receiver shall
--   ignore the whole SMS-CAMEL-TDP-Data sequence.
--
--   If this parameter is received with any other value than sms-DeliveryRequest
--   in an SMS-CAMEL-TDP-Data sequence contained in mt-sms-CSI then the receiver shall
--   ignore the whole SMS-CAMEL-TDP-Data sequence.
--
--   If this parameter is received with any other value than sms-DeliveryRequest
--   in an MT-smsCAMELTDP-Criteria sequence then the receiver shall
--   ignore the whole MT-smsCAMELTDP-Criteria sequence.
```

```
DefaultSMS-Handling ::= ENUMERATED {
     continueTransaction (0) ,
     releaseTransaction (1) ,
     ...}
--   exception handling:
--   reception of values in range 2-31 shall be treated as "continueTransaction"
--   reception of values greater than 31 shall be treated as "releaseTransaction"
```

```
M-CSI ::= SEQUENCE {
     mobilityTriggers                    MobilityTriggers,
     serviceKey                          ServiceKey,
     gsmSCF-Address                      [0]  ISDN-AddressString,
     extensionContainer                  [1]  ExtensionContainer           OPTIONAL,
     notificationToCSE                   [2] NULL                          OPTIONAL,
     csi-Active                          [3] NULL                          OPTIONAL,
     ...}
--   notificationToCSE and csi-Active shall not be present when M-CSI is sent to VLR.
--   They may only be included in ATSI/ATM ack/NSDC message.
```

```
MG-CSI ::= SEQUENCE {
     mobilityTriggers                    MobilityTriggers,
     serviceKey                          ServiceKey,
     gsmSCF-Address                      [0]  ISDN-AddressString,
     extensionContainer                  [1]  ExtensionContainer          OPTIONAL,
     notificationToCSE                   [2] NULL                         OPTIONAL,
     csi-Active                          [3] NULL                         OPTIONAL,
     ...}
--   notificationToCSE and csi-Active shall not be present when MG-CSI is sent to SGSN.
--   They may only be included in ATSI/ATM ack/NSDC message.
```

```
MobilityTriggers  ::= SEQUENCE SIZE (1..maxNumOfMobilityTriggers) OF
     MM-Code
```

```
maxNumOfMobilityTriggers INTEGER ::= 10
```

```
MM-Code ::= OCTET STRING (SIZE (1))
--   This type is used to indicate a Mobility Management event.
--   Actions for the following MM-Code values are defined in CAMEL Phase 4:
--
--   CS domain MM events:
--   Location-update-in-same-VLR         MM-Code ::= '00000000'B
--   Location-update-to-other-VLR        MM-Code ::= '00000001'B
--   IMSI-Attach                         MM-Code ::= '00000010'B
--   MS-initiated-IMSI-Detach            MM-Code ::= '00000011'B
--   Network-initiated-IMSI-Detach       MM-Code ::= '00000100'B
--
--   PS domain MM events:
--   Routeing-Area-update-in-same-SGSN   MM-Code ::= '10000000'B
--   Routeing-Area-update-to-other-SGSN  MM-Code ::= '10000001'B
--   GPRS-Attach                         MM-Code ::= '10000010'B
--   MS-initiated-GPRS-Detach            MM-Code ::= '10000011'B
--   Network-initiated-GPRS-Detach       MM-Code ::= '10000100'B
--   Network-initiated-transfer-to-MS-not-reachable-for-paging
--                                       MM-Code ::= '10000101'B
--
--   If the MSC receives any other MM-code than the ones listed above for the
--   CS domain, then the MSC shall ignore that MM-code.
--   If the SGSN receives any other MM-code than the ones listed above for the
--   PS domain, then the SGSN shall ignore that MM-code.
```

```
T-CSI ::= SEQUENCE {
     t-BcsmCamelTDPDataList              T-BcsmCamelTDPDataList,
     extensionContainer                  ExtensionContainer               OPTIONAL,
     ...,
     camelCapabilityHandling             [0] CamelCapabilityHandling      OPTIONAL,
     notificationToCSE                   [1] NULL                         OPTIONAL,
     csi-Active                          [2] NULL                         OPTIONAL}
--   notificationToCSE and csi-Active shall not be present when VT-CSI/T-CSI is sent
--   to VLR/GMSC.
--   They may only be included in ATSI/ATM ack/NSDC message.
```

```
T-BcsmCamelTDPDataList ::= SEQUENCE SIZE (1..maxNumOfCamelTDPData) OF
     T-BcsmCamelTDPData
     --- T-BcsmCamelTDPDataList shall not contain more than one instance of
     --- T-BcsmCamelTDPData containing the same value for t-BcsmTriggerDetectionPoint.
     --- For CAMEL Phase 2, this means that only one instance of T-BcsmCamelTDPData is
allowed
     --- with t-BcsmTriggerDetectionPoint being equal to DP12.
     --- For CAMEL Phase 3, more TDP's are allowed.
```

```
T-BcsmCamelTDPData ::= SEQUENCE {
     t-BcsmTriggerDetectionPoint         T-BcsmTriggerDetectionPoint,
     serviceKey                          ServiceKey,
     gsmSCF-Address                      [0] ISDN-AddressString,
     defaultCallHandling                 [1] DefaultCallHandling,
     extensionContainer                  [2] ExtensionContainer           OPTIONAL,
     ...}
```

```
T-BcsmTriggerDetectionPoint ::= ENUMERATED {
     termAttemptAuthorized (12),
     ... ,
     tBusy (13),
     tNoAnswer (14)}
     -- exception handling:
     -- For T-BcsmCamelTDPData sequences containing this parameter with any other
     -- value than the ones listed above, the receiver shall ignore the whole
     -- T-BcsmCamelTDPData sequence.
```

-- gprs location information retrieval types

```
SendRoutingInfoForGprsArg ::= SEQUENCE {
     imsi                                [0] IMSI,
     ggsn-Address                        [1] GSN-Address             OPTIONAL,
     ggsn-Number                         [2]  ISDN-AddressString,
     extensionContainer                  [3] ExtensionContainer      OPTIONAL,
     ...}
```

```
SendRoutingInfoForGprsRes ::= SEQUENCE {
     sgsn-Address                        [0] GSN-Address,
     ggsn-Address                        [1]  GSN-Address            OPTIONAL,
     mobileNotReachableReason            [2]  AbsentSubscriberDiagnosticSM    OPTIONAL,
     extensionContainer                  [3] ExtensionContainer      OPTIONAL,
     ...}
```

-- failure report types

```
FailureReportArg ::= SEQUENCE {
     imsi                                [0] IMSI,
     ggsn-Number                         [1] ISDN-AddressString      ,
     ggsn-Address                        [2] GSN-Address             OPTIONAL,
     extensionContainer                  [3] ExtensionContainer      OPTIONAL,
     ...}
```

```
FailureReportRes ::= SEQUENCE {
     ggsn-Address                        [0] GSN-Address             OPTIONAL,
     extensionContainer                  [1] ExtensionContainer      OPTIONAL,
     ...}
```

-- gprs notification types

```
NoteMsPresentForGprsArg ::= SEQUENCE {
     imsi                                [0] IMSI,
     sgsn-Address                        [1] GSN-Address,
     ggsn-Address                        [2] GSN-Address             OPTIONAL,
     extensionContainer                  [3] ExtensionContainer      OPTIONAL,
     ...}
```

```
NoteMsPresentForGprsRes ::= SEQUENCE {
     extensionContainer                  [0] ExtensionContainer      OPTIONAL,
     ...}
```

-- fault recovery types

```
ResetArg ::= SEQUENCE {
     hlr-Number                      ISDN-AddressString,
     hlr-List                        HLR-List                    OPTIONAL,
     ...}
```

```
RestoreDataArg ::= SEQUENCE {
     imsi                            IMSI,
     lmsi                            LMSI                        OPTIONAL,
     extensionContainer              ExtensionContainer          OPTIONAL,
     ... ,
     vlr-Capability                  [6] VLR-Capability          OPTIONAL }
```

```
RestoreDataRes ::= SEQUENCE {
     hlr-Number                      ISDN-AddressString,
     msNotReachable                  NULL                        OPTIONAL,
     extensionContainer              ExtensionContainer          OPTIONAL,
     ...}
```

-- VBS/VGCS types

```
VBSDataList ::= SEQUENCE SIZE (1..maxNumOfVBSGroupIds) OF
                                        VoiceBroadcastData
```

```
VGCSDataList ::= SEQUENCE SIZE (1..maxNumOfVGCSGroupIds) OF
                                        VoiceGroupCallData
```

```
maxNumOfVBSGroupIds  INTEGER ::= 50
```

```
maxNumOfVGCSGroupIds  INTEGER ::= 50
```

```
VoiceGroupCallData  ::= SEQUENCE {
     groupId                            GroupId,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
VoiceBroadcastData ::= SEQUENCE {
     groupid                            GroupId,
     broadcastInitEntitlement           NULL                            OPTIONAL,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
GroupId  ::= OCTET STRING (SIZE (3))
     -- Refers to the Group Identification as specified in GSM TS 03.03
     -- and 03.68/ 03.69
```

-- provide subscriber info types

```
ProvideSubscriberInfoArg ::= SEQUENCE {
     imsi     [0] IMSI,
     lmsi     [1] LMSI                  OPTIONAL,
     requestedInfo                      [2] RequestedInfo,
     extensionContainer                 [3] ExtensionContainer          OPTIONAL,
     ...}
```

```
ProvideSubscriberInfoRes ::= SEQUENCE {
     subscriberInfo                     SubscriberInfo,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
SubscriberInfo ::= SEQUENCE {
     locationInformation                [0] LocationInformation         OPTIONAL,
     subscriberState                    [1] SubscriberState             OPTIONAL,
     extensionContainer                 [2] ExtensionContainer          OPTIONAL,
     ... ,
     locationInformationGPRS            [3] LocationInformationGPRS     OPTIONAL,
     ps-SubscriberState                 [4] PS-SubscriberState          OPTIONAL,
     imei                               [5] IMEI                        OPTIONAL,
     ms-Classmark2                      [6] MS-Classmark2               OPTIONAL,
     gprs-MS-Class                      [7] GPRSMSClass                 OPTIONAL}

--   If the HLR receives locationInformation, subscriberState or ms-Classmark2 from an SGSN
--   it shall discard them.
--   If the HLR receives locationInformationGPRS, ps-SubscriberState or gprs-MS-Class from
--   a VLR it shall discard them.
--   If the HLR receives parameters which it has not requested, it shall discard them.
```

```
MS-Classmark2 ::= OCTET STRING (SIZE (3))
     -- This parameter carries the value part of the MS Classmark 2 IE defined in
     -- 3GPP TS 24.008 [35].
```

```
GPRSMSClass ::= SEQUENCE {
     mSNetworkCapability                [0] MSNetworkCapability,
     mSRadioAccessCapability            [1] MSRadioAccessCapability     OPTIONAL
     }
```

```
MSNetworkCapability ::= OCTET STRING (SIZE (1..8))
     -- This parameter carries the value part of the MS Network Capability IE defined in
     -- 3GPP TS 24.008 [35].
```

```
MSRadioAccessCapability ::= OCTET STRING (SIZE (1..50))
     -- This parameter carries the value part of the MS Radio Access Capability IE defined in
     -- 3GPP TS 24.008 [35].
```

```
RequestedInfo ::= SEQUENCE {
     locationInformation                [0] NULL                            OPTIONAL,
     subscriberState                    [1] NULL                            OPTIONAL,
     extensionContainer                 [2] ExtensionContainer              OPTIONAL,
     ...,
     currentLocation                    [3] NULL                            OPTIONAL,
     requestedDomain                    [4] DomainType                      OPTIONAL,
     imei                               [6] NULL                            OPTIONAL,
     ms-classmark                       [5] NULL                            OPTIONAL }

--   currentLocation shall be absent if locationInformation is absent
```

```
DomainType ::=  ENUMERATED {
     cs-Domain                          (0),
     ps-Domain                          (1),
     ...}
-- exception handling:
-- reception of values > 1 shall be mapped to 'cs-Domain'
```

```
LocationInformation ::= SEQUENCE {
     ageOfLocationInformation           AgeOfLocationInformation          OPTIONAL,
     geographicalInformation            [0] GeographicalInformation       OPTIONAL,
     vlr-number                         [1] ISDN-AddressString            OPTIONAL,
     locationNumber                     [2] LocationNumber                OPTIONAL,
     cellGlobalIdOrServiceAreaIdOrLAI   [3] CellGlobalIdOrServiceAreaIdOrLAI   OPTIONAL,
     extensionContainer                 [4] ExtensionContainer            OPTIONAL,
     ... ,
     selectedLSA-Id                     [5] LSAIdentity                   OPTIONAL,
     msc-Number                         [6] ISDN-AddressString            OPTIONAL,
     geodeticInformation                [7] GeodeticInformation           OPTIONAL,
     currentLocationRetrieved           [8] NULL                          OPTIONAL,
     sai-Present                        [9] NULL                          OPTIONAL }
-- sai-Present indicates that the cellGlobalIdOrServiceAreaIdOrLAI parameter contains
-- a Service Area Identity.
-- currentLocationRetrieved shall be present
-- if the location information were retrieved after a successfull paging.
```

```
LocationInformationGPRS ::= SEQUENCE {
     cellGlobalIdOrServiceAreaIdOrLAI   [0] CellGlobalIdOrServiceAreaIdOrLAI OPTIONAL,
     routeingAreaIdentity               [1] RAIdentity                    OPTIONAL,
     geographicalInformation            [2] GeographicalInformation       OPTIONAL,
     sgsn-Number                        [3] ISDN-AddressString            OPTIONAL,
     selectedLSAIdentity                [4] LSAIdentity                   OPTIONAL,
     extensionContainer                 [5] ExtensionContainer            OPTIONAL,
     ...,
     sai-Present                        [6] NULL                          OPTIONAL,
     geodeticInformation                [7] GeodeticInformation           OPTIONAL,
     currentLocationRetrieved           [8] NULL                          OPTIONAL,
     ageOfLocationInformation           [9] AgeOfLocationInformation      OPTIONAL }
-- sai-Present indicates that the cellGlobalIdOrServiceAreaIdOrLAI parameter contains
-- a Service Area Identity.
-- currentLocationRetrieved shall be present if the location information
-- was retrieved after successful paging.
```

```
RAIdentity ::= OCTET STRING (SIZE (6))
-- Routing Area Identity is coded in accordance with 3GPP TS 29.060 [105].
-- It shall contain the value part defined in 3GPP TS 29.060 only. I.e. the 3GPP TS 29.060
-- type identifier octet shall not be included.
```

```
GeographicalInformation ::= OCTET STRING (SIZE (8))
--   Refers to geographical Information defined in 3GPP TS 23.032.
--   Only the description of an ellipsoid point with uncertainty circle
--   as specified in 3GPP TS 23.032 is allowed to be used
--   The internal structure according to 3GPP TS 23.032 is as follows:
--        Type of shape (ellipsoid point with uncertainty circle)          1 octet
--        Degrees of Latitude                                              3 octets
--        Degrees of Longitude                                             3 octets
--        Uncertainty code                                                 1 octet
```

```
GeodeticInformation ::= OCTET STRING (SIZE (10))
--   Refers to Calling Geodetic Location defined in Q.763 (1999).
--   Only the description of an ellipsoid point with uncertainty circle
--   as specified in Q.763 (1999) is allowed to be used
--   The internal structure according to Q.763 (1999) is as follows:
--        Screening and presentation indicators                          1 octet
--        Type of shape (ellipsoid point with uncertainty circle)        1 octet
--        Degrees of Latitude                                            3 octets
--        Degrees of Longitude                                           3 octets
--        Uncertainty code                                               1 octet
--        Confidence                                                     1 octet
```

```
LocationNumber ::= OCTET STRING (SIZE (2..10))
    -- the internal structure is defined in ~~CCITT~~ITU-T Rec Q.763
```

```
SubscriberState ::= CHOICE {
    assumedIdle                 [0] NULL,
    camelBusy                   [1] NULL,
    netDetNotReachable          NotReachableReason,
    notProvidedFromVLR          [2] NULL}
```

```
PS-SubscriberState ::= CHOICE {
    notProvidedFromSGSN                 [0] NULL,
    ps-Detached                         [1] NULL,
    ps-AttachedNotReachableForPaging    [2] NULL,
    ps-AttachedReachableForPaging       [3] NULL,
    ps-PDP-ActiveNotReachableForPaging  [4] PDP-ContextInfoList,
    ps-PDP-ActiveReachableForPaging     [5] PDP-ContextInfoList}
```

```
PDP-ContextInfoList ::= SEQUENCE SIZE (1..maxNumOfPDP-Contexts) OF
                        PDP-ContextInfo
```

```
PDP-ContextInfo ::= SEQUENCE {
    pdp-ContextIdentifier       [0] ContextId,
    pdp-ContextActive           [1] NULL                        OPTIONAL,
    pdp-Type                    [2] PDP-Type,
    pdp-Address                 [3] PDP-Address                 OPTIONAL,
    apn-Subscribed              [4] APN                         OPTIONAL,
    apn-InUse                   [5] APN                         OPTIONAL,
    nsapi                       [6] NSAPI                       OPTIONAL,
    transactionId               [7] TransactionId               OPTIONAL,
    teid-ForGnAndGp             [8] TEID                        OPTIONAL,
    teid-ForIu                  [9] TEID                        OPTIONAL,
    ggsn-Address                [10] GSN-Address                OPTIONAL,
    qos-Subscribed              [11] Ext-QoS-Subscribed         OPTIONAL,
    qos-Requested               [12] Ext-QoS-Subscribed         OPTIONAL,
    qos-Negotiated              [13] Ext-QoS-Subscribed         OPTIONAL,
    chargingId                  [14] GPRSChargingID             OPTIONAL,
    chargingCharacteristics     [15] ChargingCharacteristics    OPTIONAL,
    rnc-Address                 [16] GSN-Address                OPTIONAL,
    extensionContainer          [17] ExtensionContainer         OPTIONAL,
    ...}
```

```
NSAPI ::= INTEGER (0..15)
--   This type is used to indicate the Network layer Service Access Point
```

```
TransactionId ::= OCTET STRING (SIZE (1..2))
--   This type carries the value part of the transaction identifier which is used in the
--   session management messages on the access interface. The encoding is defined in
--   3GPP TS 24.008
```

```
TEID ::= OCTET STRING (SIZE (4))
--   This type carries the value part of the Tunnel Endpoint Identifier which is used to
--   distinguish between different tunnels between the same pair of entities which communicate
--   using the GPRS Tunnelling Protocol The encoding is defined in 3GPP TS 29.060.
```

```
GPRSChargingID ::= OCTET STRING (SIZE (4))
--   The Charging ID is a unique four octet value generated by the GGSN when
--   a PDP Context is activated. A Charging ID is generated for each activated context.
--   The encoding is defined in 3GPP TS 29.060.
```

```
NotReachableReason ::= ENUMERATED {
     msPurged (0),
     imsiDetached (1),
     restrictedArea (2),
     notRegistered (3)}
```

-- any time interrogation info types

```
AnyTimeInterrogationArg ::= SEQUENCE {
     subscriberIdentity                  [0] SubscriberIdentity,
     requestedInfo                       [1] RequestedInfo,
     gsmSCF-Address                      [3] ISDN-AddressString,
     extensionContainer                  [2] ExtensionContainer          OPTIONAL,
     ...}
```

```
AnyTimeInterrogationRes ::= SEQUENCE {
     subscriberInfo                      SubscriberInfo,
     extensionContainer                  ExtensionContainer              OPTIONAL,
     ...}
```

-- any time information handling types

```
AnyTimeSubscriptionInterrogationArg ::= SEQUENCE {
     subscriberIdentity                  [0] SubscriberIdentity,
     requestedSubscriptionInfo           [1] RequestedSubscriptionInfo,
     gsmSCF-Address                      [2] ISDN-AddressString,
     extensionContainer                  [3] ExtensionContainer          OPTIONAL,
     longFTN-Supported                   [4]  NULL                       OPTIONAL,
     ...}
```

```
AnyTimeSubscriptionInterrogationRes ::= SEQUENCE {
     callForwardingData                  [1] CallForwardingData          OPTIONAL,
     callBarringData                     [2] CallBarringData             OPTIONAL,
     odb-Info                            [3] ODB-Info                    OPTIONAL,
     camel-SubscriptionInfo              [4] CAMEL-SubscriptionInfo      OPTIONAL,
     supportedVLR-CAMEL-Phases           [5] SupportedCamelPhases        OPTIONAL,
     supportedSGSN-CAMEL-Phases          [6] SupportedCamelPhases        OPTIONAL,
     extensionContainer                  [7] ExtensionContainer          OPTIONAL,
     ...}
```

```
RequestedSubscriptionInfo ::= SEQUENCE {
     requestedSS-Info                    [1] SS-ForBS-Code               OPTIONAL,
     odb                                 [2] NULL                        OPTIONAL,
     requestedCAMEL-SubscriptionInfo     [3] RequestedCAMEL-SubscriptionInfo   OPTIONAL,
     supportedVLR-CAMEL-Phases           [4] NULL                        OPTIONAL,
     supportedSGSN-CAMEL-Phases          [5] NULL                        OPTIONAL,
     extensionContainer                  [6] ExtensionContainer          OPTIONAL,
     ...,
     additionalRequestedCAMEL-SubscriptionInfo
                                         [7] AdditionalRequestedCAMEL-SubscriptionInfo
                                                                         OPTIONAL }
```

```
RequestedCAMEL-SubscriptionInfo ::= ENUMERATED {
     o-CSI                               (0),
     t-CSI                               (1),
     vt-CSI                              (2),
     tif-CSI                             (3),
     gprs-CSI                            (4),
     mo-sms-CSI                          (5),
     ss-CSI                              (6),
     m-CSI                               (7),
     d-csi                               (8)}
```

```
AdditionalRequestedCAMEL-SubscriptionInfo ::= ENUMERATED {
     mt-sms-CSI                          (0),
     mg-csi                              (1),
     ...}
--   exception handling: unknown values shall be discarded by the receiver.
```

```
CallForwardingData ::= SEQUENCE {
     forwardingFeatureList               Ext-ForwFeatureList,
     notificationToCSE                   NULL                            OPTIONAL,
     extensionContainer                  [0] ExtensionContainer          OPTIONAL,
     ...}
```

```
CallBarringData ::= SEQUENCE {
     callBarringFeatureList              Ext-CallBarFeatureList,
     password                            Password                          OPTIONAL,
     wrongPasswordAttemptsCounter        WrongPasswordAttemptsCounter      OPTIONAL,
     notificationToCSE                   NULL                              OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
WrongPasswordAttemptsCounter ::= INTEGER (0..4)
```

```
ODB-Info ::= SEQUENCE {
     odb-Data                            ODB-Data,
     notificationToCSE                   NULL                              OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
CAMEL-SubscriptionInfo ::= SEQUENCE {
     o-CSI                          [0]  O-CSI                          OPTIONAL,
     o-BcsmCamelTDP-CriteriaList    [1]  O-BcsmCamelTDPCriteriaList     OPTIONAL,
     d-CSI                          [2]  D-CSI                          OPTIONAL,
     t-CSI                          [3]  T-CSI                          OPTIONAL,
     t-BCSM-CAMEL-TDP-CriteriaList  [4]  T-BCSM-CAMEL-TDP-CriteriaList OPTIONAL,
     vt-CSI                         [5]  T-CSI                          OPTIONAL,
     vt-BCSM-CAMEL-TDP-CriteriaList [6]  T-BCSM-CAMEL-TDP-CriteriaList OPTIONAL,
     tif-CSI                        [7]  NULL                           OPTIONAL,
     tif-CSI-NotificationToCSE      [8]  NULL                           OPTIONAL,
     gprs-CSI                       [9]  GPRS-CSI                       OPTIONAL,
     mo-sms-CSI                     [10] SMS-CSI                        OPTIONAL,
     ss-CSI                         [11] SS-CSI                         OPTIONAL,
     m-CSI                          [12] M-CSI                          OPTIONAL,
     extensionContainer             [13] ExtensionContainer             OPTIONAL,
     ...,
     specificCSIDeletedList         [14] SpecificCSI-Withdraw           OPTIONAL,
     mt-sms-CSI                     [15] SMS-CSI                        OPTIONAL,
     mt-smsCAMELTDP-CriteriaList    [16] MT-smsCAMELTDP-CriteriaList    OPTIONAL,
     mg-csi                         [17] MG-CSI                         OPTIONAL
     }
```

```
AnyTimeModificationArg ::= SEQUENCE {
     subscriberIdentity             [0]  SubscriberIdentity,
     gsmSCF-Address                 [1]  ISDN-AddressString,
     modificationRequestFor-CF-Info [2]  ModificationRequestFor-CF-Info OPTIONAL,
     modificationRequestFor-CB-Info [3]  ModificationRequestFor-CB-Info OPTIONAL,
     modificationRequestFor-CSI     [4]  ModificationRequestFor-CSI     OPTIONAL,
     extensionContainer             [5]  ExtensionContainer             OPTIONAL,
     longFTN-Supported              [6]  NULL                           OPTIONAL,
     ...,
     modificationRequestFor-ODB-data [7]  ModificationRequestFor-ODB-data OPTIONAL }
```

```
AnyTimeModificationRes ::= SEQUENCE {
     ss-InfoFor-CSE                 [0]  Ext-SS-InfoFor-CSE             OPTIONAL,
     camel-SubscriptionInfo         [1]  CAMEL-SubscriptionInfo         OPTIONAL,
     extensionContainer             [2]  ExtensionContainer             OPTIONAL,
     ...,
     odb-Info                       [3]  ODB-Info                       OPTIONAL }
```

```
ModificationRequestFor-CF-Info ::= SEQUENCE {
     ss-Code                        [0]  SS-Code,
     basicService                   [1]  Ext-BasicServiceCode           OPTIONAL,
     ss-Status                      [2]  Ext-SS-Status                  OPTIONAL,
     forwardedToNumber              [3]  AddressString                  OPTIONAL,
     forwardedToSubaddress          [4]  ISDN-SubaddressString          OPTIONAL,
     noReplyConditionTime           [5]  Ext-NoRepCondTime              OPTIONAL,
     modifyNotificationToCSE        [6]  ModificationInstruction        OPTIONAL,
     extensionContainer             [7]  ExtensionContainer             OPTIONAL,
     ...}
```

```
ModificationRequestFor-CB-Info ::= SEQUENCE {
     ss-Code                        [0]  SS-Code,
     basicService                   [1]  Ext-BasicServiceCode           OPTIONAL,
     ss-Status                      [2]  Ext-SS-Status                  OPTIONAL,
     password                       [3]  Password                       OPTIONAL,
     wrongPasswordAttemptsCounter   [4]  WrongPasswordAttemptsCounter   OPTIONAL,
     modifyNotificationToCSE        [5]  ModificationInstruction        OPTIONAL,
     extensionContainer             [6]  ExtensionContainer             OPTIONAL,
     ...}
```

```
ModificationRequestFor-ODB-data ::= SEQUENCE {
     odb-data                          [0]  ODB-Data                    OPTIONAL,
     modifyNotificationToCSE           [1]  ModificationInstruction     OPTIONAL,
     extensionContainer                [2]  ExtensionContainer          OPTIONAL,
     ...}
```

```
ModificationRequestFor-CSI ::= SEQUENCE {
     requestedCamel-SubscriptionInfo      [0]  RequestedCAMEL-SubscriptionInfo,
     modifyNotificationToCSE              [1]  ModificationInstruction      OPTIONAL,
     modifyCSI-State                      [2]  ModificationInstruction      OPTIONAL,
     extensionContainer                   [3]  ExtensionContainer           OPTIONAL,
     ...,
     additionalRequestedCAMEL-SubscriptionInfo
                                          [4] AdditionalRequestedCAMEL-SubscriptionInfo
                                                                            OPTIONAL }
```

```
ModificationInstruction ::= ENUMERATED {
     deactivate                        (0),
     activate                          (1)}
```

-- subscriber data modification notification types

```
NoteSubscriberDataModifiedArg ::= SEQUENCE {
     imsi                              IMSI,
     msisdn                            ISDN-AddressString,
     forwardingInfoFor-CSE             [0] Ext-ForwardingInfoFor-CSE     OPTIONAL,
     callBarringInfoFor-CSE            [1] Ext-CallBarringInfoFor-CSE    OPTIONAL,
     odb-Info                          [2] ODB-Info                      OPTIONAL,
     camel-SubscriptionInfo            [3] CAMEL-SubscriptionInfo        OPTIONAL,
     allInformationSent                [4] NULL                          OPTIONAL,
     extensionContainer                ExtensionContainer                OPTIONAL,
     ...}
```

```
NoteSubscriberDataModifiedRes ::= SEQUENCE {
     extensionContainer                ExtensionContainer                OPTIONAL,
     ...}
```

-- mobility management event notificatioon info types

```
NoteMM-EventArg::= SEQUENCE {
     serviceKey                        ServiceKey,
     eventMet                          [0]  MM-Code,
     imsi                              [1]  IMSI,
     msisdn                            [2]  ISDN-AddressString,
     locationInformation               [3]  LocationInformation          OPTIONAL,
     supportedCAMELPhases              [5]  SupportedCamelPhases         OPTIONAL,
     extensionContainer                [6]  ExtensionContainer           OPTIONAL,
     ...,
     locationInformationGPRS           [7]  LocationInformationGPRS      OPTIONAL
}
```

```
NoteMM-EventRes ::= SEQUENCE {
     extensionContainer                ExtensionContainer                OPTIONAL,
     ...}
```

```
Ext-SS-InfoFor-CSE ::= CHOICE {
     forwardingInfoFor-CSE             [0] Ext-ForwardingInfoFor-CSE,
     callBarringInfoFor-CSE            [1] Ext-CallBarringInfoFor-CSE
     }
```

```
Ext-ForwardingInfoFor-CSE ::= SEQUENCE {
     ss-Code                           [0]  SS-Code,
     forwardingFeatureList             [1]  Ext-ForwFeatureList,
     notificationToCSE                 [2]  NULL                         OPTIONAL,
     extensionContainer                [3]  ExtensionContainer           OPTIONAL,
     ...}
```

```
Ext-CallBarringInfoFor-CSE ::= SEQUENCE {
     ss-Code                           [0]  SS-Code,
     callBarringFeatureList            [1]  Ext-CallBarFeatureList,
     password                          [2]  Password                     OPTIONAL,
     wrongPasswordAttemptsCounter      [3]  WrongPasswordAttemptsCounter OPTIONAL,
     notificationToCSE                 [4]  NULL                         OPTIONAL,
     extensionContainer                [5]  ExtensionContainer           OPTIONAL,
     ...}
```

```
END
```

17.7.2 Operation and maintenance data types

```
MAP-OM-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-OM-DataTypes (12) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS
   ActivateTraceModeArg,
   ActivateTraceModeRes,
   DeactivateTraceModeArg,
   DeactivateTraceModeRes
;

IMPORTS
   AddressString,
   IMSI
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}


;
```

```
ActivateTraceModeArg ::= SEQUENCE {
     imsi                              [0] IMSI                          OPTIONAL,
     traceReference                    [1] TraceReference,
     traceType                         [2] TraceType,
     omc-Id                            [3] AddressString                 OPTIONAL,
     extensionContainer                [4] ExtensionContainer            OPTIONAL,
     ...}
```

```
TraceReference ::= OCTET STRING (SIZE (1..2))
```

```
TraceType ::= INTEGER
     (0..255)
     -- Trace types are fully defined in TS GSM 12.08.
```

```
ActivateTraceModeRes ::= SEQUENCE {
     extensionContainer                [0] ExtensionContainer            OPTIONAL,
     ...}
```

```
DeactivateTraceModeArg ::= SEQUENCE {
     imsi                              [0] IMSI                          OPTIONAL,
     traceReference                    [1] TraceReference,
     extensionContainer                [2] ExtensionContainer            OPTIONAL,
     ...}
```

```
DeactivateTraceModeRes ::= SEQUENCE {
     extensionContainer                [0] ExtensionContainer            OPTIONAL,
     ...}
```

```
END
```

17.7.3 Call handling data types

```
MAP-CH-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CH-DataTypes (13) version8 (8)}

DEFINITIONS
```

```
IMPLICIT TAGS

::=

BEGIN

EXPORTS
   SendRoutingInfoArg,
   SendRoutingInfoRes,
   ProvideRoamingNumberArg,
   ProvideRoamingNumberRes,
   ResumeCallHandlingArg,
   ResumeCallHandlingRes,
   NumberOfForwarding,
   SuppressionOfAnnouncement,
   CallReferenceNumber,
   ProvideSIWFSNumberArg,
   ProvideSIWFSNumberRes,
   SIWFSSignallingModifyArg,
   SIWFSSignallingModifyRes,
   SetReportingStateArg,
   SetReportingStateRes,
   StatusReportArg,
   StatusReportRes,
   RemoteUserFreeArg,
   RemoteUserFreeRes,
   IST-AlertArg,
   IST-AlertRes,
   IST-CommandArg,
   IST-CommandRes
;

IMPORTS
   SubscriberInfo,
   SupportedCamelPhases,
   CUG-Interlock,
   O-CSI,
   D-CSI,
   O-BcsmCamelTDPCriteriaList,
   T-BCSM-CAMEL-TDP-CriteriaList,
   IST-SupportIndicator,
   IST-AlertTimerValue,
   T-CSI


FROM MAP-MS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-MS-DataTypes (11) version8 (8)}

   ForwardingOptions,
   SS-List,
   CCBS-Feature
FROM MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

   ISDN-AddressString,
   ISDN-SubaddressString,
   FTN-AddressString,
   ExternalSignalInfo,
   Ext-ExternalSignalInfo,
   IMSI,
   LMSI,
   Ext-BasicServiceCode,
   AlertingPattern,
   NAEA-PreferredCI


FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}
;
```

```
CUG-CheckInfo ::= SEQUENCE {
     cug-Interlock                    CUG-Interlock,
     cug-OutgoingAccess               NULL                              OPTIONAL,
     extensionContainer               ExtensionContainer                OPTIONAL,
     ...}
```

```
NumberOfForwarding ::= INTEGER (1..5)
```

```
SendRoutingInfoArg ::= SEQUENCE {
     msisdn                           [0] ISDN-AddressString,
     cug-CheckInfo                    [1] CUG-CheckInfo                 OPTIONAL,
     numberOfForwarding               [2] NumberOfForwarding            OPTIONAL,
     interrogationType                [3] InterrogationType,
     or-Interrogation                 [4] NULL                          OPTIONAL,
     or-Capability                    [5] OR-Phase                      OPTIONAL,
     gmsc-OrGsmSCF-Address            [6] ISDN-AddressString,
     callReferenceNumber              [7] CallReferenceNumber           OPTIONAL,
     forwardingReason                 [8] ForwardingReason              OPTIONAL,
     basicServiceGroup                [9] Ext-BasicServiceCode          OPTIONAL,
     networkSignalInfo                [10] ExternalSignalInfo           OPTIONAL,
     camelInfo                        [11] CamelInfo                    OPTIONAL,
     suppressionOfAnnouncement        [12] SuppressionOfAnnouncement    OPTIONAL,
     extensionContainer               [13] ExtensionContainer           OPTIONAL,
     ...,
     alertingPattern                  [14] AlertingPattern              OPTIONAL,
     ccbs-Call                        [15] NULL                          OPTIONAL,
     supportedCCBS-Phase              [16]  SupportedCCBS-Phase         OPTIONAL,
     additionalSignalInfo             [17] Ext-ExternalSignalInfo       OPTIONAL,
     istSupportIndicator              [18] IST-SupportIndicator         OPTIONAL,
     pre-pagingSupported              [19] NULL                         OPTIONAL,
     callDiversionTreatmentIndicator  [20] CallDiversionTreatmentIndicator   OPTIONAL,
     longFTN-Supported                [21] NULL                         OPTIONAL,
     suppress-VT-CSI                  [22] NULL                         OPTIONAL,
     suppressIncomingCallBarring      [23] NULL                         OPTIONAL,
     gsmSCF-InitiatedCall             [24] NULL                         OPTIONAL
      }
```

```
SuppressionOfAnnouncement ::= NULL
```

```
InterrogationType ::= ENUMERATED {
     basicCall  (0),
     forwarding  (1)}
```

```
OR-Phase ::= INTEGER (1..127)
```

```
CallReferenceNumber ::= OCTET STRING (SIZE (1..8))
```

```
ForwardingReason ::= ENUMERATED {
     notReachable  (0),
     busy  (1),
     noReply  (2)}
```

```
SupportedCCBS-Phase ::= INTEGER (1..127)
-- exception handling:
-- Only value 1 is used.
-- Values in the ranges 2-127 are reserved for future use.
-- If received values 2-127 shall be mapped on to value 1.
```

```
CallDiversionTreatmentIndicator ::= OCTET STRING (SIZE(1))
--   callDiversionAllowed (xxxx xx01)
--   callDiversionNotAllowed (xxxx xx10)
--   network default is call diversion allowed
```

```
SendRoutingInfoRes ::= [3] SEQUENCE {
    imsi                              [9] IMSI                          OPTIONAL,
    -- IMSI must be present if SendRoutingInfoRes is not segmented.
    -- If the TC-Result-NL segmentation option is taken the IMSI must be
    -- present in one segmented transmission of SendRoutingInfoRes.
    extendedRoutingInfo               ExtendedRoutingInfo               OPTIONAL,
    cug-CheckInfo                     [3] CUG-CheckInfo                 OPTIONAL,
    cugSubscriptionFlag               [6] NULL                          OPTIONAL,
    subscriberInfo                    [7] SubscriberInfo                OPTIONAL,
    ss-List                           [1] SS-List                       OPTIONAL,
    basicService                      [5] Ext-BasicServiceCode          OPTIONAL,
    forwardingInterrogationRequired   [4] NULL                          OPTIONAL,
    vmsc-Address                      [2] ISDN-AddressString            OPTIONAL,
    extensionContainer                [0] ExtensionContainer            OPTIONAL,
    ... ,
    naea-PreferredCI                  [10] NAEA-PreferredCI             OPTIONAL,
    -- naea-PreferredCI is included at the discretion of the HLR operator.
    ccbs-Indicators                   [11] CCBS-Indicators              OPTIONAL,
    msisdn                            [12] ISDN-AddressString           OPTIONAL,
    numberPortabilityStatus           [13] NumberPortabilityStatus      OPTIONAL,
    istAlertTimer                     [14] IST-AlertTimerValue          OPTIONAL,
    supportedCamelPhases              [15] SupportedCamelPhases         OPTIONAL
    }
```

```
NumberPortabilityStatus ::= ENUMERATED {
    notKnownToBePorted                (0),
    ownNumberPortedOut                (1),
    foreignNumberPortedToForeignNetwork  (2),
    ...}
    --   exception handling:
    --  reception of other values than the ones listed the receiver shall ignore the
    --  whole NumberPortabilityStatus
```

```
CCBS-Indicators ::= SEQUENCE {
    ccbs-Possible                     [0]  NULL                         OPTIONAL,
    keepCCBS-CallIndicator            [1]  NULL                         OPTIONAL,
    extensionContainer                [2]  ExtensionContainer           OPTIONAL,
    ...}
```

```
RoutingInfo ::= CHOICE {
    roamingNumber                     ISDN-AddressString,
    forwardingData                    ForwardingData}
```

```
ForwardingData ::= SEQUENCE {
    forwardedToNumber                 [5] ISDN-AddressString            OPTIONAL,
    -- When this datatype is sent from an HLR which supports CAMEL Phase 2
    -- to a GMSC which supports CAMEL Phase 2 the GMSC shall not check the
    -- format of the number
    forwardedToSubaddress             [4] ISDN-SubaddressString         OPTIONAL,
    forwardingOptions                 [6] ForwardingOptions             OPTIONAL,
    extensionContainer                [7] ExtensionContainer            OPTIONAL,
    ...,
    longForwardedToNumber             [8] FTN-AddressString             OPTIONAL}
```

```
ProvideRoamingNumberArg ::= SEQUENCE {
    imsi                              [0] IMSI,
    msc-Number                        [1] ISDN-AddressString,
    msisdn                            [2] ISDN-AddressString            OPTIONAL,
    lmsi                              [4] LMSI                          OPTIONAL,
    gsm-BearerCapability              [5] ExternalSignalInfo            OPTIONAL,
    networkSignalInfo                 [6] ExternalSignalInfo            OPTIONAL,
    suppressionOfAnnouncement         [7] SuppressionOfAnnouncement     OPTIONAL,
    gmsc-Address                      [8] ISDN-AddressString            OPTIONAL,
    callReferenceNumber               [9] CallReferenceNumber           OPTIONAL,
    or-Interrogation                  [10] NULL                         OPTIONAL,
    extensionContainer                [11] ExtensionContainer           OPTIONAL,
    ... ,
    alertingPattern                   [12] AlertingPattern              OPTIONAL,
    ccbs-Call                         [13] NULL                         OPTIONAL,
    supportedCamelPhasesInGMSC        [15] SupportedCamelPhases         OPTIONAL,
    additionalSignalInfo              [14] Ext-ExternalSignalInfo       OPTIONAL,
    orNotSupportedInGMSC              [16] NULL                         OPTIONAL,
    pre-pagingSupported               [17] NULL                         OPTIONAL,
    longFTN-Supported                 [18] NULL                         OPTIONAL,
    suppress-VT-CSI                   [19] NULL                         OPTIONAL
     }
```

```
ProvideRoamingNumberRes ::= SEQUENCE {
    roamingNumber                   ISDN-AddressString,
    extensionContainer              ExtensionContainer              OPTIONAL,
    ...}
```

```
ResumeCallHandlingArg ::= SEQUENCE {
    callReferenceNumber             [0] CallReferenceNumber         OPTIONAL,
    basicServiceGroup               [1] Ext-BasicServiceCode        OPTIONAL,
    forwardingData                  [2] ForwardingData              OPTIONAL,
    imsi                            [3] IMSI                        OPTIONAL,
    cug-CheckInfo                   [4] CUG-CheckInfo               OPTIONAL,
    o-CSI                           [5] O-CSI                       OPTIONAL,
    extensionContainer              [7] ExtensionContainer          OPTIONAL,
    ccbs-Possible                   [8]  NULL                       OPTIONAL,
    msisdn                          [9]  ISDN-AddressString         OPTIONAL,
    uu-Data                         [10] UU-Data                    OPTIONAL,
    allInformationSent              [11] NULL                       OPTIONAL,
    ...,
    d-csi                           [12] D-CSI                      OPTIONAL}
```

```
UU-Data ::= SEQUENCE {
    uuIndicator                     [0] UUIndicator                 OPTIONAL,
    uui                             [1] UUI                         OPTIONAL,
    uusCFInteraction                [2] NULL                        OPTIONAL,
    extensionContainer              [3] ExtensionContainer          OPTIONAL,
    ...}
```

```
UUIndicator ::= OCTET STRING (SIZE (1))
    -- Octets are coded according to ETS 300 356
```

```
UUI  ::= OCTET STRING (SIZE (1..131))
    -- Octets are coded according to ETS 300 356
```

```
ResumeCallHandlingRes ::= SEQUENCE {
    extensionContainer              ExtensionContainer              OPTIONAL,
    ...}
```

```
CamelInfo ::= SEQUENCE {
    supportedCamelPhases            SupportedCamelPhases,
    suppress-T-CSI                  NULL                            OPTIONAL,
    extensionContainer              ExtensionContainer              OPTIONAL,
    ...}
```

```
ExtendedRoutingInfo ::= CHOICE {
    routingInfo                     RoutingInfo,
    camelRoutingInfo                [8] CamelRoutingInfo}
```

```
CamelRoutingInfo ::= SEQUENCE {
    forwardingData                  ForwardingData                  OPTIONAL,
    gmscCamelSubscriptionInfo       [0] GmscCamelSubscriptionInfo,
    extensionContainer              [1] ExtensionContainer          OPTIONAL,
    ...}
```

```
GmscCamelSubscriptionInfo ::= SEQUENCE {
    t-CSI                           [0] T-CSI OPTIONAL,
    o-CSI                           [1] O-CSI OPTIONAL,
    extensionContainer              [2] ExtensionContainer          OPTIONAL,
    ...,
    o-BcsmCamelTDP-CriteriaList     [3] O-BcsmCamelTDPCriteriaList  OPTIONAL,
    t-BCSM-CAMEL-TDP-CriteriaList   [4]  T-BCSM-CAMEL-TDP-CriteriaList OPTIONAL,
    d-csi                           [5]  D-CSI                      OPTIONAL}
```

```
ProvideSIWFSNumberArg ::= SEQUENCE {
    gsm-BearerCapability            [0] ExternalSignalInfo,
    isdn-BearerCapability           [1] ExternalSignalInfo,
    call-Direction                  [2] CallDirection,
    b-Subscriber-Address            [3] ISDN-AddressString,
    chosenChannel                   [4] ExternalSignalInfo,
    lowerLayerCompatibility         [5] ExternalSignalInfo          OPTIONAL,
    highLayerCompatibility          [6] ExternalSignalInfo          OPTIONAL,
    extensionContainer              [7] ExtensionContainer          OPTIONAL,
    ...}
```

```
CallDirection ::= OCTET STRING (SIZE (1))
     -- OCTET 1

     -- bit 1 (direction of call)
     -- 0  Mobile Originated Call (MOC)
     -- 1  Mobile Terminated Call (MTC)
```

```
ProvideSIWFSNumberRes ::= SEQUENCE {
     sIWFSNumber                    [0] ISDN-AddressString,
     extensionContainer             [1] ExtensionContainer          OPTIONAL,
     ...}
```

```
SIWFSSignallingModifyArg ::= SEQUENCE {
     channelType                    [0] ExternalSignalInfo          OPTIONAL,
     chosenChannel                  [1] ExternalSignalInfo          OPTIONAL,
     extensionContainer             [2] ExtensionContainer          OPTIONAL,
     ...}
```

```
SIWFSSignallingModifyRes ::= SEQUENCE {
     chosenChannel                  [0] ExternalSignalInfo          OPTIONAL,
     extensionContainer             [1] ExtensionContainer          OPTIONAL,
     ...}
```

```
SetReportingStateArg ::= SEQUENCE {
     imsi                           [0]  IMSI                       OPTIONAL,
     lmsi                           [1]  LMSI                       OPTIONAL,
     ccbs-Monitoring                [2]  ReportingState             OPTIONAL,
     extensionContainer             [3]  ExtensionContainer         OPTIONAL,
     ...}
```

```
ReportingState ::= ENUMERATED {
     stopMonitoring                 (0),
     startMonitoring                (1),
     ...}
     -- exception handling:
     -- reception of values 2-10 shall be mapped to 'stopMonitoring'
     -- reception of values > 10 shall be mapped to 'startMonitoring'
```

```
SetReportingStateRes ::= SEQUENCE{
     ccbs-SubscriberStatus          [0]  CCBS-SubscriberStatus      OPTIONAL,
     extensionContainer             [1]  ExtensionContainer         OPTIONAL,
     ...}
```

```
CCBS-SubscriberStatus ::= ENUMERATED {
     ccbsNotIdle                    (0),
     ccbsIdle                       (1),
     ccbsNotReachable               (2),
     ...}
     --  exception handling:
     --  reception of values 3-10 shall be mapped to 'ccbsNotIdle'
     --  reception of values 11-20 shall be mapped to 'ccbsIdle'
     --  reception of values > 20 shall be mapped to 'ccbsNotReachable'
```

```
StatusReportArg ::= SEQUENCE{
     imsi                           [0]  IMSI,
     eventReportData                [1]  EventReportData            OPTIONAL,
     callReportdata                 [2]  CallReportData             OPTIONAL,
     extensionContainer             [3]  ExtensionContainer         OPTIONAL,
     ...}
```

```
EventReportData ::= SEQUENCE{
     ccbs-SubscriberStatus          [0]  CCBS-SubscriberStatus      OPTIONAL,
     extensionContainer             [1]  ExtensionContainer         OPTIONAL,
     ...}
```

```
CallReportData ::= SEQUENCE{
     monitoringMode                 [0]  MonitoringMode             OPTIONAL,
     callOutcome                    [1]  CallOutcome                OPTIONAL,
     extensionContainer             [2]  ExtensionContainer         OPTIONAL,
     ...}
```

```
MonitoringMode ::= ENUMERATED {
     a-side                              (0),
     b-side                              (1),
     ...}
     --   exception handling:
     --  reception of values 2-10 shall be mapped 'a-side'
     --  reception of values > 10 shall be mapped to 'b-side'
```

```
CallOutcome ::= ENUMERATED {
     success                             (0),
     failure                             (1),
     busy                                (2),
     ...}
     --   exception handling:
     --  reception of values 3-10 shall be mapped to 'success'
     --  reception of values 11-20 shall be mapped to 'failure'
     --  reception of values > 20 shall be mapped to 'busy'
```

```
StatusReportRes ::= SEQUENCE {
     extensionContainer                  [0]  ExtensionContainer           OPTIONAL,
     ...}
```

```
RemoteUserFreeArg ::= SEQUENCE{
     imsi                                [0]  IMSI,
     callInfo                            [1]  ExternalSignalInfo,
     ccbs-Feature                        [2]  CCBS-Feature,
     translatedB-Number                  [3]  ISDN-AddressString,
     replaceB-Number                     [4]  NULL                         OPTIONAL,
     alertingPattern                     [5]  AlertingPattern              OPTIONAL,
     extensionContainer                  [6]  ExtensionContainer           OPTIONAL,
     ...}
```

```
RemoteUserFreeRes ::= SEQUENCE{
     ruf-Outcome                         [0]  RUF-Outcome,
     extensionContainer                  [1]  ExtensionContainer           OPTIONAL,
     ...}
```

```
RUF-Outcome ::= ENUMERATED{
     accepted (0),
     rejected (1),
     noResponseFromFreeMS (2), -- T4 Expiry
     noResponseFromBusyMS (3), -- T10 Expiry
     udubFromFreeMS (4),
     udubFromBusyMS (5),
     ...}
     -- exception handling:
     -- reception of values 6-20 shall be mapped to 'accepted'
     -- reception of values 21-30 shall be mapped to 'rejected'
     -- reception of values 31-40 shall be mapped to 'noResponseFromFreeMS'
     -- reception of values 41-50 shall be mapped to 'noResponseFromBusyMS'
     -- reception of values 51-60 shall be mapped to 'udubFromFreeMS'
     -- reception of values > 60 shall be mapped to 'udubFromBusyMS'
```

```
IST-AlertArg ::= SEQUENCE{
     imsi                                [0]  IMSI,
     extensionContainer                  [1]  ExtensionContainer           OPTIONAL,
     ...}
```

```
IST-AlertRes ::= SEQUENCE{
     istAlertTimer                       [0]  IST-AlertTimerValue          OPTIONAL,
     istInformationWithdraw              [1]  NULL                         OPTIONAL,
     callTerminationIndicator            [2]  CallTerminationIndicator     OPTIONAL,
     extensionContainer                  [3]  ExtensionContainer           OPTIONAL,
     ...}
```

```
IST-CommandArg ::= SEQUENCE{
     imsi                                [0]  IMSI,
     extensionContainer                  [1]  ExtensionContainer           OPTIONAL,
     ...}
```

```
IST-CommandRes ::= SEQUENCE{
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
CallTerminationIndicator ::= ENUMERATED {
     terminateCallActivityReferred       (0),
     terminateAllCallActivities          (1),
     ...}
     -- exception handling:
     -- reception of values 2-10 shall be mapped to ' terminateCallActivityReferred '
     -- reception of values > 10 shall be mapped to ' terminateAllCallActivities '

     -- In MSCs not supporting linkage of all call activities, any value received shall
     -- be interpreted as ' terminateCallActivityReferred '
```

```
END
```

17.7.4 Supplementary service data types

```
MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS
   RegisterSS-Arg,
   SS-Info,
   SS-Status,
   SS-SubscriptionOption,
   SS-ForBS-Code,
   InterrogateSS-Res,
   USSD-Arg,
   USSD-Res,
   USSD-DataCodingScheme,
   USSD-String,
   Password,
   GuidanceInfo,
   SS-List,
   SS-InfoList,
   OverrideCategory,
   CliRestrictionOption,
   NoReplyConditionTime,
   ForwardingOptions,
   maxNumOfSS,
   SS-Data,
   SS-InvocationNotificationArg,
   SS-InvocationNotificationRes,
   CCBS-Feature,
   RegisterCC-EntryArg,
   RegisterCC-EntryRes,
   EraseCC-EntryArg,
   EraseCC-EntryRes
;

IMPORTS
   AddressString,
   ISDN-AddressString,
   ISDN-SubaddressString,
   FTN-AddressString,
   IMSI,
   BasicServiceCode,
   AlertingPattern,
   EMLPP-Priority,
   MaxMC-Bearers,
   MC-Bearers,
   ExternalSignalInfo

FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}
```

```
   SS-Code
FROM MAP-SS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-Code (15) version8 (8)}
;
```

```
RegisterSS-Arg ::= SEQUENCE {
     ss-Code                          SS-Code,
     basicService                     BasicServiceCode                 OPTIONAL,
     forwardedToNumber                [4] AddressString                OPTIONAL,
     forwardedToSubaddress            [6] ISDN-SubaddressString        OPTIONAL,
     noReplyConditionTime             [5] NoReplyConditionTime         OPTIONAL,
     ...,
     defaultPriority                  [7] EMLPP-Priority               OPTIONAL,
     nbrUser                          [8] MC-Bearers                   OPTIONAL,
     longFTN-Supported                [9]  NULL                        OPTIONAL }
```

```
NoReplyConditionTime ::= INTEGER (5..30)
```

```
SS-Info ::= CHOICE {
     forwardingInfo                   [0] ForwardingInfo,
     callBarringInfo                  [1] CallBarringInfo,
     ss-Data                          [3] SS-Data}
```

```
ForwardingInfo ::= SEQUENCE {
     ss-Code                          SS-Code                          OPTIONAL,
     forwardingFeatureList            ForwardingFeatureList,
     ...}
```

```
ForwardingFeatureList ::=
     SEQUENCE SIZE (1..maxNumOfBasicServiceGroups) OF
                                      ForwardingFeature
```

```
ForwardingFeature ::= SEQUENCE {
     basicService                     BasicServiceCode                 OPTIONAL,
     ss-Status                        [4] SS-Status                    OPTIONAL,
     forwardedToNumber                [5] ISDN-AddressString           OPTIONAL,
     forwardedToSubaddress            [8] ISDN-SubaddressString        OPTIONAL,
     forwardingOptions                [6] ForwardingOptions            OPTIONAL,
     noReplyConditionTime             [7] NoReplyConditionTime         OPTIONAL,
     ...,
     longForwardedToNumber            [9] FTN-AddressString            OPTIONAL }
```

```
SS-Status ::= OCTET STRING (SIZE (1))

     -- bits 8765: 0000 (unused)
     -- bits 4321: Used to convey the "P bit","R bit","A bit" and "Q bit",
     --            representing supplementary service state information
     --            as defined in TS 3GPP TS 23.011 [22]

     -- bit 4: "Q bit"

     -- bit 3: "P bit"

     -- bit 2: "R bit"

     -- bit 1: "A bit"
```

```
ForwardingOptions ::= OCTET STRING (SIZE (1))

     -- bit 8: notification to forwarding party
     --   0  no notification
     --   1  notification

     -- bit 7: redirecting presentation
     --   0 no presentation
     --   1  presentation

     -- bit 6: notification to calling party
     --   0  no notification
     --   1  notification

     -- bit 5: 0 (unused)

     -- bits 43: forwarding reason
     --   00  ms not reachable
     --   01  ms busy
     --   10  no reply
     --   11  unconditional when used in a SRI Result,
     --       or call deflection when used in a RCH Argument
     -- bits 21: 00 (unused)
```

```
CallBarringInfo ::= SEQUENCE {
     ss-Code                         SS-Code                         OPTIONAL,
     callBarringFeatureList          CallBarringFeatureList,
     ...}
```

```
CallBarringFeatureList ::= SEQUENCE SIZE (1..maxNumOfBasicServiceGroups) OF
                                     CallBarringFeature
```

```
CallBarringFeature ::= SEQUENCE {
     basicService                    BasicServiceCode                OPTIONAL,
     ss-Status [4] SS-Status         OPTIONAL,
     ...}
```

```
SS-Data ::= SEQUENCE {
     ss-Code                         SS-Code                         OPTIONAL,
     ss-Status                       [4] SS-Status                   OPTIONAL,
     ss-SubscriptionOption           SS-SubscriptionOption           OPTIONAL,
     basicServiceGroupList           BasicServiceGroupList           OPTIONAL,
     ...,
     defaultPriority                 EMLPP-Priority                  OPTIONAL,
     nbrUser                         [5] MC-Bearers                  OPTIONAL
     }
```

```
SS-SubscriptionOption ::= CHOICE {
     cliRestrictionOption            [2] CliRestrictionOption,
     overrideCategory                [1] OverrideCategory}
```

```
CliRestrictionOption ::= ENUMERATED {
     permanent  (0),
     temporaryDefaultRestricted  (1),
     temporaryDefaultAllowed  (2)}
```

```
OverrideCategory ::= ENUMERATED {
     overrideEnabled  (0),
     overrideDisabled  (1)}
```

```
SS-ForBS-Code ::= SEQUENCE {
     ss-Code                         SS-Code,
     basicService                    BasicServiceCode                OPTIONAL,
     ...,
     longFTN-Supported               [4]  NULL                       OPTIONAL }
```

```
GenericServiceInfo ::= SEQUENCE {
     ss-Status SS-Status,
     cliRestrictionOption            CliRestrictionOption            OPTIONAL,
     ...,
     maximumEntitledPriority         [0] EMLPP-Priority              OPTIONAL,
     defaultPriority                 [1] EMLPP-Priority              OPTIONAL,
     ccbs-FeatureList                [2] CCBS-FeatureList            OPTIONAL,
     nbrSB                           [3] MaxMC-Bearers               OPTIONAL,
     nbrUser                         [4] MC-Bearers                  OPTIONAL,
     nbrSN                           [5] MC-Bearers                  OPTIONAL }
```

```
CCBS-FeatureList ::= SEQUENCE SIZE (1..maxNumOfCCBS-Requests) OF
                                        CCBS-Feature
```

```
maxNumOfCCBS-Requests  INTEGER ::= 5
```

```
CCBS-Feature ::= SEQUENCE {
     ccbs-Index                         [0] CCBS-Index                    OPTIONAL,
     b-subscriberNumber                 [1] ISDN-AddressString            OPTIONAL,
     b-subscriberSubaddress             [2] ISDN-SubaddressString         OPTIONAL,
     basicServiceGroup                  [3] BasicServiceCode              OPTIONAL,
     ...}
```

```
CCBS-Index  ::= INTEGER (1..maxNumOfCCBS-Requests)
```

```
InterrogateSS-Res ::= CHOICE {
     ss-Status                          [0] SS-Status,
     basicServiceGroupList              [2] BasicServiceGroupList,
     forwardingFeatureList              [3] ForwardingFeatureList,
     genericServiceInfo                 [4]  GenericServiceInfo }
```

```
USSD-Arg ::= SEQUENCE {
     ussd-DataCodingScheme              USSD-DataCodingScheme,
     ussd-String                        USSD-String,
     ... ,
     alertingPattern                    AlertingPattern                   OPTIONAL,
     msisdn                             [0] ISDN-AddressString            OPTIONAL }
```

```
USSD-Res ::= SEQUENCE {
     ussd-DataCodingScheme              USSD-DataCodingScheme,
     ussd-String                        USSD-String,
     ...}
```

```
USSD-DataCodingScheme ::= OCTET STRING (SIZE (1))
     -- The structure of the USSD-DataCodingScheme is defined by
     -- the Cell Broadcast Data Coding Scheme as described in
     -- TS 3GPP TS 23.038 [25]
```

```
USSD-String ::= OCTET STRING (SIZE (1..maxUSSD-StringLength))
     -- The structure of the contents of the USSD-String is dependent
     -- on the USSD-DataCodingScheme as described in TS 3GPP TS 23.038 [25].
```

```
maxUSSD-StringLength  INTEGER ::= 160
```

```
Password ::= NumericString
     (FROM ("0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9"))
     (SIZE (4))
```

```
GuidanceInfo ::= ENUMERATED {
     enterPW  (0),
     enterNewPW  (1),
     enterNewPW-Again  (2)}
     -- How this information is really delivered to the subscriber
     -- (display, announcement, ...) is not part of this
     -- specification.
```

```
SS-List ::= SEQUENCE SIZE (1..maxNumOfSS) OF
                                        SS-Code
```

```
maxNumOfSS  INTEGER ::= 30
```

```
SS-InfoList ::= SEQUENCE SIZE (1..maxNumOfSS) OF
                                        SS-Info
```

```
BasicServiceGroupList ::= SEQUENCE SIZE (1..maxNumOfBasicServiceGroups) OF
                                        BasicServiceCode
```

```
maxNumOfBasicServiceGroups  INTEGER ::= 13
```

```
SS-InvocationNotificationArg ::= SEQUENCE {
     imsi                                [0] IMSI,
     msisdn                              [1] ISDN-AddressString,
     ss-Event                            [2] SS-Code,
     -- The following SS-Code values are allowed :
     -- ect                              SS-Code ::= '00110001'B
     -- multiPTY                         SS-Code ::= '01010001'B
     -- cd                               SS-Code ::= '00100100'B
     -- ccbs                             SS-Code ::= '01000100'B
     ss-EventSpecification               [3] SS-EventSpecification         OPTIONAL,
     extensionContainer                  [4] ExtensionContainer            OPTIONAL,
     ...,
     b-subscriberNumber                  [5]  ISDN-AddressString           OPTIONAL,
     ccbs-RequestState                   [6]  CCBS-RequestState            OPTIONAL
     }
```

```
CCBS-RequestState ::= ENUMERATED {
     request   (0),
     recall    (1),
     active    (2),
     completed (3),
     suspended (4),
     frozen    (5),
     deleted   (6)
     }
```

```
SS-InvocationNotificationRes ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...
     }
```

```
SS-EventSpecification ::= SEQUENCE SIZE (1..maxEventSpecification) OF
                                         AddressString
```

```
maxEventSpecification  INTEGER ::= 2
```

```
RegisterCC-EntryArg ::= SEQUENCE {
     ss-Code                             [0]  SS-Code,
     ccbs-Data                           [1]  CCBS-Data                    OPTIONAL,
     ...}
```

```
CCBS-Data ::= SEQUENCE {
     ccbs-Feature                        [0]  CCBS-Feature,
     translatedB-Number                  [1]  ISDN-AddressString,
     serviceIndicator                    [2]  ServiceIndicator             OPTIONAL,
     callInfo                            [3]  ExternalSignalInfo,
     networkSignalInfo                   [4]  ExternalSignalInfo,
     ...}
```

```
ServiceIndicator ::= BIT STRING {
     clir-invoked (0),
     camel-invoked (1)} (SIZE(2..32))
     -- exception handling:
     -- bits 2 to 31 shall be ignored if received and not understood
```

```
RegisterCC-EntryRes ::= SEQUENCE {
     ccbs-Feature                        [0] CCBS-Feature                  OPTIONAL,
     ...}
```

```
EraseCC-EntryArg ::= SEQUENCE {
     ss-Code                             [0]  SS-Code,
     ccbs-Index                          [1]  CCBS-Index                   OPTIONAL,
     ...}
```

```
EraseCC-EntryRes ::= SEQUENCE {
     ss-Code                             [0]  SS-Code,
     ss-Status                           [1] SS-Status                     OPTIONAL,
     ...}
```

```
END
```

17.7.5 Supplementary service codes

```
MAP-SS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-Code (15) version8 (8)}
```

```
DEFINITIONS

::=

BEGIN
```

```
SS-Code ::= OCTET STRING (SIZE (1))
     -- This type is used to represent the code identifying a single
     -- supplementary service, a group of supplementary services, or
     -- all supplementary services. The services and abbreviations
     -- used are defined in TS 3GPP TS 22.004 [5]. The internal structure is
     -- defined as follows:
     --
     -- bits 87654321: group (bits 8765), and specific service
     -- (bits 4321)
```

```
allSS                                   SS-Code ::= '00000000'B
     -- reserved for possible future use
     -- all SS
```

```
allLineIdentificationSS                 SS-Code ::= '00010000'B
     -- reserved for possible future use
     -- all line identification SS
clip                                    SS-Code ::= '00010001'B
     -- calling line identification presentation
clir                                    SS-Code ::= '00010010'B
     -- calling line identification restriction
colp                                    SS-Code ::= '00010011'B
     -- connected line identification presentation
colr                                    SS-Code ::= '00010100'B
     -- connected line identification restriction
mci                                     SS-Code ::= '00010101'B
     -- reserved for possible future use
     -- malicious call identification

allNameIdentificationSS                 SS-Code ::= '00011000'B
     -- all name identification SS
cnap                                    SS-Code ::= '00011001'B
     -- calling name presentation

     -- SS-Codes '00011010'B to '00011111'B are reserved for future
     -- NameIdentification Supplementary Service use.
```

```
allForwardingSS                         SS-Code ::= '00100000'B
     -- all forwarding SS
cfu                                     SS-Code ::= '00100001'B
     -- call forwarding unconditional
allCondForwardingSS                     SS-Code ::= '00101000'B
     -- all conditional forwarding SS
cfb                                     SS-Code ::= '00101001'B
     -- call forwarding on mobile subscriber busy
cfnry                                   SS-Code ::= '00101010'B
     -- call forwarding on no reply
cfnrc                                   SS-Code ::= '00101011'B
     -- call forwarding on mobile subscriber not reachable
cd                                      SS-Code ::= '00100100'B
     -- call deflection
```

```
allCallOfferingSS                       SS-Code ::= '00110000'B
     -- reserved for possible future use
     -- all call offering SS includes also all forwarding SS
ect                                     SS-Code ::= '00110001'B
          -- explicit call transfer
mah                                     SS-Code ::= '00110010'B
     -- reserved for possible future use
     -- mobile access hunting
```

```
allCallCompletionSS                     SS-Code ::= '01000000'B
     -- reserved for possible future use
     -- all Call completion SS
cw                                      SS-Code ::= '01000001'B
     -- call waiting
hold                                    SS-Code ::= '01000010'B
     -- call hold
ccbs-A                                  SS-Code ::= '01000011'B
     -- completion of call to busy subscribers, originating side
ccbs-B                                  SS-Code ::= '01000100'B
     -- completion of call to busy subscribers, destination side
     -- this SS-Code is used only in InsertSubscriberData and DeleteSubscriberData
mc                                      SS-Code ::= '01000101'B
     -- multicall
```

```
allMultiPartySS                         SS-Code ::= '01010000'B
     -- reserved for possible future use
     -- all multiparty SS
multiPTY                                SS-Code ::= '01010001'B
     -- multiparty
```

```
allCommunityOfInterest-SS               SS-Code ::= '01100000'B
     -- reserved for possible future use
     -- all community of interest SS
cug                                     SS-Code ::= '01100001'B
     -- closed user group
```

```
allChargingSS                           SS-Code ::= '01110000'B
     -- reserved for possible future use
     -- all charging SS
aoci                                    SS-Code ::= '01110001'B
     -- advice of charge information
aocc                                    SS-Code ::= '01110010'B
     -- advice of charge charging
```

```
allAdditionalInfoTransferSS             SS-Code ::= '10000000'B
     -- reserved for possible future use
     -- all additional information transfer SS
uus1                                    SS-Code ::= '10000001'B
     -- UUS1 user-to-user signalling
uus2                                    SS-Code ::= '10000010'B
     -- UUS2 user-to-user signalling
uus3                                    SS-Code ::= '10000011'B
     -- UUS3 user-to-user signalling
```

```
allBarringSS                            SS-Code ::= '10010000'B
     -- all barring SS
barringOfOutgoingCalls                  SS-Code ::= '10010001'B
     -- barring of outgoing calls
baoc                                    SS-Code ::= '10010010'B
     -- barring of all outgoing calls
boic                                    SS-Code ::= '10010011'B
     -- barring of outgoing international calls
boicExHC                                SS-Code ::= '10010100'B
     -- barring of outgoing international calls except those directed
     -- to the home PLMN
barringOfIncomingCalls                  SS-Code ::= '10011001'B
     -- barring of incoming calls
baic                                    SS-Code ::= '10011010'B
     -- barring of all incoming calls
bicRoam                                 SS-Code ::= '10011011'B
     -- barring of incoming calls when roaming outside home PLMN
     -- Country
```

```
allPLMN-specificSS                      SS-Code ::= '11110000'B
plmn-specificSS-1                       SS-Code ::= '11110001'B
plmn-specificSS-2                       SS-Code ::= '11110010'B
plmn-specificSS-3                       SS-Code ::= '11110011'B
plmn-specificSS-4                       SS-Code ::= '11110100'B
plmn-specificSS-5                       SS-Code ::= '11110101'B
plmn-specificSS-6                       SS-Code ::= '11110110'B
plmn-specificSS-7                       SS-Code ::= '11110111'B
plmn-specificSS-8                       SS-Code ::= '11111000'B
plmn-specificSS-9                       SS-Code ::= '11111001'B
plmn-specificSS-A                       SS-Code ::= '11111010'B
plmn-specificSS-B                       SS-Code ::= '11111011'B
plmn-specificSS-C                       SS-Code ::= '11111100'B
plmn-specificSS-D                       SS-Code ::= '11111101'B
plmn-specificSS-E                       SS-Code ::= '11111110'B
plmn-specificSS-F                       SS-Code ::= '11111111'B
```

```
allCallPrioritySS                       SS-Code ::= '10100000'B
     -- reserved for possible future use
     -- all call priority SS
emlpp                                   SS-Code ::= '10100001'B
     -- enhanced Multilevel Precedence Pre-emption (EMLPP) service
```

```
allLCSPrivacyException                  SS-Code ::= '10110000'B
     -- all LCS Privacy Exception Classes
universal                               SS-Code ::= '10110001'B
     -- allow location by any LCS client
callSessionRelated                      SS-Code ::= '10110010'B
     -- allow location by any value added LCS client to which a call/session
     -- is established from the target MS
callSessionUnrelated                    SS-Code ::= '10110011'B
     -- allow location by designated external value added LCS clients
plmnoperator                            SS-Code ::= '10110100'B
     -- allow location by designated PLMN operator LCS clients
serviceType                             SS-Code ::= '10110101'B
     -- allow location by LCS clients of a designated LCS service type
```

```
allMOLR-SS                              SS-Code ::= '11000000'B
     -- all Mobile Originating Location Request Classes
basicSelfLocation                       SS-Code ::= '11000001'B
     -- allow an MS to request its own location
autonomousSelfLocation                  SS-Code ::= '11000010'B
     -- allow an MS to perform self location without interaction
     -- with the PLMN for a predetermined period of time
transferToThirdParty                    SS-Code ::= '11000011'B
     -- allow an MS to request transfer of its location to another LCS client
```

```
END
```

17.7.6 Short message data types

```
MAP-SM-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SM-DataTypes (16) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS
   RoutingInfoForSM-Arg,
   RoutingInfoForSM-Res,
   MO-ForwardSM-Arg,
   MO-ForwardSM-Res,
   MT-ForwardSM-Arg,
   MT-ForwardSM-Res,
   ReportSM-DeliveryStatusArg,
   ReportSM-DeliveryStatusRes,
   AlertServiceCentreArg,
   InformServiceCentreArg,
   ReadyForSM-Arg,
   ReadyForSM-Res,
   SM-DeliveryOutcome,
```

```
   AlertReason,
   Additional-Number
;

IMPORTS
   AddressString,
   ISDN-AddressString,
   SignalInfo,
   IMSI,
   LMSI
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   AbsentSubscriberDiagnosticSM
FROM MAP-ER-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ER-DataTypes (17) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}
;
```

```
RoutingInfoForSM-Arg ::= SEQUENCE {
     msisdn                              [0] ISDN-AddressString,
     sm-RP-PRI                           [1] BOOLEAN,
     serviceCentreAddress                [2] AddressString,
     extensionContainer                  [6] ExtensionContainer          OPTIONAL,
     ... ,
     gprsSupportIndicator                [7]  NULL                       OPTIONAL,
     -- gprsSupportIndicator is set only if the SMS-GMSC supports
     -- receiving of two numbers from the HLR
     sm-RP-MTI                           [8] SM-RP-MTI                   OPTIONAL,
     sm-RP-SMEA                          [9] SM-RP-SMEA                  OPTIONAL }
```

```
SM-RP-MTI ::= INTEGER (0..10)
     -- 0 SMS Deliver
     -- 1 SMS Status Report
     -- other values are reserved for future use and shall be discarded if
     -- received
```

```
SM-RP-SMEA ::= OCTET STRING (SIZE (1..12))
     -- this parameter contains an address field which is encoded
     -- as defined in 3GPP TS 23.140. An address field contains 3 elements :
     --       address-length
     --       type-of-address
     --       address-value
```

```
RoutingInfoForSM-Res ::= SEQUENCE {
     imsi                                IMSI,
     locationInfoWithLMSI                [0] LocationInfoWithLMSI,
     extensionContainer                  [4] ExtensionContainer          OPTIONAL,
     ...}
```

```
LocationInfoWithLMSI ::= SEQUENCE {
     networkNode-Number                  [1] ISDN-AddressString,
     lmsi                                LMSI                            OPTIONAL,
     extensionContainer                  ExtensionContainer              OPTIONAL,
     ...,
     gprsNodeIndicator                   [5]  NULL                       OPTIONAL,
     -- gprsNodeIndicator is set only if the SGSN number is sent as the
     -- Network Node Number
     additional-Number                   [6] Additional-Number           OPTIONAL
     -- NetworkNode-number can be either msc-number or sgsn-number
     }
```

```
Additional-Number ::= CHOICE {
     msc-Number                          [0] ISDN-AddressString,
     sgsn-Number                         [1] ISDN-AddressString}
     -- additional-number can be either msc-number or sgsn-number
     -- if received networkNode-number is msc-number then the
     -- additional number is sgsn-number
     -- if received networkNode-number is sgsn-number then the
     -- additional number is msc-number
```

```
MO-ForwardSM-Arg ::= SEQUENCE {
     sm-RP-DA                            SM-RP-DA,
     sm-RP-OA                            SM-RP-OA,
     sm-RP-UI                            SignalInfo,
     extensionContainer                  ExtensionContainer              OPTIONAL,
     ... ,
     imsi                                IMSI                            OPTIONAL }
```

```
MO-ForwardSM-Res ::= SEQUENCE {
     sm-RP-UI                            SignalInfo                      OPTIONAL,
     extensionContainer                  ExtensionContainer              OPTIONAL,
     ...}
```

```
MT-ForwardSM-Arg ::= SEQUENCE {
     sm-RP-DA                            SM-RP-DA,
     sm-RP-OA                            SM-RP-OA,
     sm-RP-UI                            SignalInfo,
     moreMessagesToSend                  NULL                            OPTIONAL,
     extensionContainer                  ExtensionContainer              OPTIONAL,
     ...}
```

```
MT-ForwardSM-Res ::= SEQUENCE {
     sm-RP-UI                            SignalInfo                      OPTIONAL,
     extensionContainer                  ExtensionContainer              OPTIONAL,
     ...}
```

```
SM-RP-DA ::= CHOICE {
     imsi                                [0] IMSI,
     lmsi                                [1] LMSI,
     serviceCentreAddressDA              [4] AddressString,
     noSM-RP-DA                          [5] NULL}
```

```
SM-RP-OA ::= CHOICE {
     msisdn                              [2] ISDN-AddressString,
     serviceCentreAddressOA              [4] AddressString,
     noSM-RP-OA                          [5] NULL}
```

```
ReportSM-DeliveryStatusArg ::= SEQUENCE {
     msisdn                              ISDN-AddressString,
     serviceCentreAddress                AddressString,
     sm-DeliveryOutcome                  SM-DeliveryOutcome,
     absentSubscriberDiagnosticSM        [0] AbsentSubscriberDiagnosticSM
                                                                         OPTIONAL,
     extensionContainer                  [1] ExtensionContainer          OPTIONAL,
     ...,
     gprsSupportIndicator                [2]  NULL                       OPTIONAL,
     -- gprsSupportIndicator is set only if the SMS-GMSC supports
     -- handling of two delivery outcomes
     deliveryOutcomeIndicator            [3]  NULL                       OPTIONAL,
     -- DeliveryOutcomeIndicator is set when the SM-DeliveryOutcome
     -- is for GPRS
     additionalSM-DeliveryOutcome        [4]  SM-DeliveryOutcome         OPTIONAL,
     -- If received, additionalSM-DeliveryOutcome is for GPRS
     -- If DeliveryOutcomeIndicator is set, then AdditionalSM-DeliveryOutcome shall be absent
     additionalAbsentSubscriberDiagnosticSM   [5]  AbsentSubscriberDiagnosticSM OPTIONAL
     -- If received additionalAbsentSubscriberDiagnosticSM is for GPRS
     -- If DeliveryOutcomeIndicator is set, then AdditionalAbsentSubscriberDiagnosticSM
     -- shall be absent
     }
```

```
SM-DeliveryOutcome ::= ENUMERATED {
     memoryCapacityExceeded  (0),
     absentSubscriber  (1),
     successfulTransfer  (2)}
```

```
ReportSM-DeliveryStatusRes ::= SEQUENCE {
     storedMSISDN                         ISDN-AddressString               OPTIONAL,
     extensionContainer                   ExtensionContainer               OPTIONAL,
     ...}
```

```
AlertServiceCentreArg ::= SEQUENCE {
     msisdn                               ISDN-AddressString,
     serviceCentreAddress                 AddressString,
     ...}
```

```
InformServiceCentreArg ::= SEQUENCE {
     storedMSISDN                         ISDN-AddressString               OPTIONAL,
     mw-Status MW-Status                  OPTIONAL,
     extensionContainer                   ExtensionContainer               OPTIONAL,
     ... ,
     absentSubscriberDiagnosticSM         AbsentSubscriberDiagnosticSM     OPTIONAL,
     additionalAbsentSubscriberDiagnosticSM   [0]  AbsentSubscriberDiagnosticSM OPTIONAL }
     -- additionalAbsentSubscriberDiagnosticSM may be present only if
     -- absentSubscriberDiagnosticSM is present.
     -- if included, additionalAbsentSubscriberDiagnosticSM is for GPRS and
     -- absentSubscriberDiagnosticSM is for non-GPRS
```

```
MW-Status ::= BIT STRING {
     sc-AddressNotIncluded  (0),
     mnrf-Set  (1),
     mcef-Set  (2) ,
     mnrg-Set   (3)} (SIZE (6..16))
     -- exception handling:
     -- bits 4 to 15 shall be ignored if received and not understood
```

```
ReadyForSM-Arg ::= SEQUENCE {
     imsi                                 [0] IMSI,
     alertReason                          AlertReason,
     alertReasonIndicator                 NULL                             OPTIONAL,
     -- alertReasonIndicator is set only when the alertReason
     -- sent to HLR is for GPRS
     extensionContainer                   ExtensionContainer               OPTIONAL,
     ...}
```

```
ReadyForSM-Res ::= SEQUENCE {
     extensionContainer                   ExtensionContainer               OPTIONAL,
     ...}
```

```
AlertReason ::= ENUMERATED {
     ms-Present  (0),
     memoryAvailable  (1)}
```

```
END
```

17.7.7 Error data types

```
MAP-ER-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ER-DataTypes (17) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS
   RoamingNotAllowedParam,
   CallBarredParam,
   CUG-RejectParam,
   SS-IncompatibilityCause,
   PW-RegistrationFailureCause,
   SM-DeliveryFailureCause,
   SystemFailureParam,
   DataMissingParam,
   UnexpectedDataParam,
   FacilityNotSupParam,
   OR-NotAllowedParam,
```

```
    UnknownSubscriberParam,
    NumberChangedParam,
    UnidentifiedSubParam,
    IllegalSubscriberParam,
    IllegalEquipmentParam,
    BearerServNotProvParam,
    TeleservNotProvParam,
    TracingBufferFullParam,
    NoRoamingNbParam,
    AbsentSubscriberParam,
    BusySubscriberParam,
    NoSubscriberReplyParam,
    ForwardingViolationParam,
    ForwardingFailedParam,
    ATI-NotAllowedParam,
    SubBusyForMT-SMS-Param,
    MessageWaitListFullParam,
    AbsentSubscriberSM-Param,
    AbsentSubscriberDiagnosticSM,
    ResourceLimitationParam,
    NoGroupCallNbParam,
    IncompatibleTerminalParam,
    ShortTermDenialParam,
    LongTermDenialParam,
    UnauthorizedRequestingNetwork-Param,
    UnauthorizedLCSClient-Param,
    PositionMethodFailure-Param,
    UnknownOrUnreachableLCSClient-Param,
    MM-EventNotSupported-Param,
    SecureTransportErrorParam,
    ATSI-NotAllowedParam,
    ATM-NotAllowedParam,
    IllegalSS-OperationParam,
    SS-NotAvailableParam,
    SS-SubscriptionViolationParam,
    InformationNotAvailableParam,
    TargetCellOutsideGCA-Param



;

IMPORTS
   SS-Status
FROM MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-DataTypes (14) version8 (8)}

   SignalInfo,
   BasicServiceCode,
   NetworkResource
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   SecurityHeader,
   ProtectedPayload
FROM MAP-ST-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ST-DataTypes (27) version8 (8)}


   SS-Code
FROM MAP-SS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-Code (15) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}
;
```

```
RoamingNotAllowedParam ::= SEQUENCE {
     roamingNotAllowedCause              RoamingNotAllowedCause,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
RoamingNotAllowedCause ::= ENUMERATED {
     plmnRoamingNotAllowed  (0),
     operatorDeterminedBarring  (3)}
```

```
CallBarredParam ::= CHOICE {
     callBarringCause                    CallBarringCause,
     -- call BarringCause must not be used in version 3 and higher
     extensibleCallBarredParam           ExtensibleCallBarredParam
     -- extensibleCallBarredParam must not be used in version <3
     }
```

```
CallBarringCause ::= ENUMERATED {
     barringServiceActive  (0),
     operatorBarring  (1)}
```

```
ExtensibleCallBarredParam ::= SEQUENCE {
     callBarringCause                    CallBarringCause                  OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ... ,
     unauthorisedMessageOriginator       [1] NULL                          OPTIONAL }
```

```
CUG-RejectParam ::= SEQUENCE {
     cug-RejectCause                     CUG-RejectCause                   OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
CUG-RejectCause ::= ENUMERATED {
     incomingCallsBarredWithinCUG  (0),
     subscriberNotMemberOfCUG  (1),
     requestedBasicServiceViolatesCUG-Constraints  (5),
     calledPartySS-InteractionViolation  (7)}
```

```
SS-IncompatibilityCause ::= SEQUENCE {
     ss-Code                             [1] SS-Code                       OPTIONAL,
     basicService                        BasicServiceCode                  OPTIONAL,
     ss-Status                           [4] SS-Status                     OPTIONAL,
     ...}
```

```
PW-RegistrationFailureCause ::= ENUMERATED {
     undetermined  (0),
     invalidFormat  (1),
     newPasswordsMismatch  (2)}
```

```
SM-EnumeratedDeliveryFailureCause ::= ENUMERATED {
     memoryCapacityExceeded  (0),
     equipmentProtocolError  (1),
     equipmentNotSM-Equipped  (2),
     unknownServiceCentre  (3),
     sc-Congestion  (4),
     invalidSME-Address  (5),
     subscriberNotSC-Subscriber  (6)}
```

```
SM-DeliveryFailureCause ::= SEQUENCE {
     sm-EnumeratedDeliveryFailureCause   SM-EnumeratedDeliveryFailureCause,
     diagnosticInfo                      SignalInfo                        OPTIONAL,
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
AbsentSubscriberSM-Param ::= SEQUENCE {
     absentSubscriberDiagnosticSM        AbsentSubscriberDiagnosticSM      OPTIONAL,
     -- AbsentSubscriberDiagnosticSM can be either for non-GPRS
     -- or for GPRS
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...,
     additionalAbsentSubscriberDiagnosticSM   [0]  AbsentSubscriberDiagnosticSM OPTIONAL }
     -- if received, additionalAbsentSubscriberDiagnosticSM
     -- is for GPRS and absentSubscriberDiagnosticSM is
     -- for non-GPRS
```

```
AbsentSubscriberDiagnosticSM ::= INTEGER (0..255)
     -- AbsentSubscriberDiagnosticSM values are defined in ETS 300 536 (3GPP TS 23.140)
```

```
SystemFailureParam ::= CHOICE {
     networkResource                    NetworkResource,
     -- networkResource must not be used in version 3
     extensibleSystemFailureParam       ExtensibleSystemFailureParam
     -- extensibleSystemFailureParam must not be used in version <3
     }
```

```
ExtensibleSystemFailureParam ::= SEQUENCE {
     networkResource                    NetworkResource                 OPTIONAL,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
DataMissingParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
UnexpectedDataParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
FacilityNotSupParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...,
     shapeOfLocationEstimateNotSupported [0]  NULL                      OPTIONAL,
     neededLcsCapabilityNotSupportedInServingNode [1] NULL              OPTIONAL }
```

```
OR-NotAllowedParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
UnknownSubscriberParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...,
     unknownSubscriberDiagnostic        UnknownSubscriberDiagnostic     OPTIONAL}
```

```
UnknownSubscriberDiagnostic ::= ENUMERATED {
     imsiUnknown  (0),
     gprsSubscriptionUnknown  (1),
     ...,
     npdbMismatch  (2)}
     -- if unknown values are received in
     -- UunknownSubscriberDiagnostic they shall be discarded
```

```
NumberChangedParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
UnidentifiedSubParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
IllegalSubscriberParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
IllegalEquipmentParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
BearerServNotProvParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
TeleservNotProvParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
TracingBufferFullParam ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
NoRoamingNbParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
AbsentSubscriberParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...,
     absentSubscriberReason              [0] AbsentSubscriberReason        OPTIONAL}
```

```
AbsentSubscriberReason ::= ENUMERATED {
     imsiDetach (0),
     restrictedArea (1),
     noPageResponse (2),
     ... ,
     purgedMS (3)}
-- exception handling: at reception of other values than the ones listed the
-- AbsentSubscriberReason shall be ignored.
-- The AbsentSubscriberReason: purgedMS is defined for the Super-Charger feature
-- (see TS 23.116). If this value is received in a Provide Roaming Number response
-- it shall be mapped to the AbsentSubscriberReason: imsiDetach in the Send Routeing
-- Information response
```

```
BusySubscriberParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...,
     ccbs-Possible                       [0] NULL                          OPTIONAL,
     ccbs-Busy                           [1] NULL                          OPTIONAL}
```

```
NoSubscriberReplyParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
ForwardingViolationParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
ForwardingFailedParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
ATI-NotAllowedParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
ATSI-NotAllowedParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
ATM-NotAllowedParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
IllegalSS-OperationParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
SS-NotAvailableParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
SS-SubscriptionViolationParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
InformationNotAvailableParam ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ...}
```

```
SubBusyForMT-SMS-Param ::= SEQUENCE {
     extensionContainer                  ExtensionContainer                OPTIONAL,
     ... ,
     gprsConnectionSuspended             NULL                              OPTIONAL }
     -- If GprsConnectionSuspended is not understood it shall
     -- be discarded
```

```
MessageWaitListFullParam ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
ResourceLimitationParam ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
NoGroupCallNbParam ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
IncompatibleTerminalParam ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
ShortTermDenialParam ::= SEQUENCE {
     ...}
```

```
LongTermDenialParam ::= SEQUENCE {
     ...}
```

```
UnauthorizedRequestingNetwork-Param ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
UnauthorizedLCSClient-Param ::= SEQUENCE {
     unauthorizedLCSClient-Diagnostic     [0] UnauthorizedLCSClient-Diagnostic   OPTIONAL,
     extensionContainer                   [1] ExtensionContainer                 OPTIONAL,
     ... }
```

```
UnauthorizedLCSClient-Diagnostic ::= ENUMERATED {
     noAdditionalInformation (0),
     clientNotInMSPrivacyExceptionList (1),
     callToClientNotSetup (2),
     privacyOverrideNotApplicable (3),
     disallowedByLocalRegulatoryRequirements (4),
     ... }
--   exception handling:
--   any unrecognized value shall be ignored
```

```
PositionMethodFailure-Param ::= SEQUENCE {
     positionMethodFailure-Diagnostic     [0] PositionMethodFailure-Diagnostic   OPTIONAL,
     extensionContainer                   [1] ExtensionContainer                 OPTIONAL,
     ... }
```

```
PositionMethodFailure-Diagnostic ::= ENUMERATED {
     congestion  (0),
     insufficientResources  (1),
     insufficientMeasurementData  (2),
     inconsistentMeasurementData  (3),
     locationProcedureNotCompleted  (4),
     locationProcedureNotSupportedByTargetMS  (5),
     qoSNotAttainable  (6),
     positionMethodNotAvailableInNetwork  (7),
     positionMethodNotAvailableInLocationArea (8),
     ... }
--   exception handling:
--   any unrecognized value shall be ignored
```

```
UnknownOrUnreachableLCSClient-Param ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
MM-EventNotSupported-Param ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
TargetCellOutsideGCA-Param ::= SEQUENCE {
     extensionContainer                   ExtensionContainer                OPTIONAL,
     ...}
```

```
SecureTransportErrorParam ::= SEQUENCE {
     securityHeader                    SecurityHeader,
     protectedPayload                  ProtectedPayload                 OPTIONAL
     }
     -- The protectedPayload carries the result of applying the security function
     -- defined in 3G TS 33.200 to the encoding of the securely transported error
     -- parameter
```

```
END
```

17.7.8 Common data types

```
MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS

   -- general data types and values
   AddressString,
   ISDN-AddressString,
   maxISDN-AddressLength,
   FTN-AddressString,
   ISDN-SubaddressString,
   ExternalSignalInfo,
   Ext-ExternalSignalInfo,
   AccessNetworkSignalInfo,
   SignalInfo,
   maxSignalInfoLength,
   AlertingPattern,

   -- data types for numbering and identification
   IMSI,
   TMSI,
   Identity,
   SubscriberId,
   IMEI,
   HLR-List,
   LMSI,
   GlobalCellId,
   NetworkResource,
   NAEA-PreferredCI,
   NAEA-CIC,
   ASCI-CallReference,
   SubscriberIdentity,

   -- data types for CAMEL
   CellGlobalIdOrServiceAreaIdOrLAI,

   -- data types for subscriber management
   BasicServiceCode,
   Ext-BasicServiceCode,
   EMLPP-Info,
   EMLPP-Priority,
   MC-SS-Info,
   MaxMC-Bearers,
   MC-Bearers,
   Ext-SS-Status,

   -- data types for geographic location
   AgeOfLocationInformation,
   LCSClientExternalID,
   LCSClientInternalID,
   LCSServiceTypeID
;

IMPORTS
   TeleserviceCode,
   Ext-TeleserviceCode
```

```
FROM MAP-TS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-TS-Code (19) version8 (8)}

   BearerServiceCode,
   Ext-BearerServiceCode
FROM MAP-BS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-BS-Code (20) version8 (8)}

   SS-Code
FROM MAP-SS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SS-Code (15) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}
;


-- general data types

TBCD-STRING ::= OCTET STRING
     -- This type (Telephony Binary Coded Decimal String) is used to
     -- represent several digits from 0 through 9, *, #, a, b, c, two
     -- digits per octet, each digit encoded 0000 to 1001 (0 to 9),
     -- 1010 (*), 1011 (#), 1100 (a), 1101 (b) or 1110 (c); 1111 used
     -- as filler when there is an odd number of digits.

     -- bits 8765 of octet n encoding digit 2n
     -- bits 4321 of octet n encoding digit 2(n-1) +1

AddressString ::= OCTET STRING (SIZE (1..maxAddressLength))
     -- This type is used to represent a number for addressing
     -- purposes. It is composed of
     --   a)   one octet for nature of address, and numbering plan
     --        indicator.
     --   b)   digits of an address encoded as TBCD-String.

     -- a)     The first octet includes a one bit extension indicator, a
     --        3 bits nature of address indicator and a 4 bits numbering
     --        plan indicator, encoded as follows:

     -- bit 8: 1  (no extension)

     -- bits 765: nature of address indicator
     --   000  unknown
     --   001  international number
     --   010  national significant number
     --   011  network specific number
     --   100  subscriber number
     --   101  reserved
     --   110  abbreviated number
     --   111  reserved for extension

     -- bits 4321: numbering plan indicator
     --   0000  unknown
     --   0001  ISDN/Telephony Numbering Plan (Rec CCITTITU-T E.164)
     --   0010  spare
     --   0011  data numbering plan (CCITTITU-T Rec X.121)
     --   0100  telex numbering plan (CCITTITU-T Rec F.69)
     --   0101  spare
     --   0110  land mobile numbering plan (CCITTITU-T Rec E.212)
     --   0111  spare
     --   1000  national numbering plan
     --   1001  private numbering plan
     --   1111  reserved for extension

     --   all other values are reserved.

     -- b)     The following octets representing digits of an address
     --        encoded as a TBCD-STRING.

maxAddressLength  INTEGER ::= 20
```

```
ISDN-AddressString ::=
            AddressString (SIZE (1..maxISDN-AddressLength))
    -- This type is used to represent ISDN numbers.
```

```
maxISDN-AddressLength  INTEGER ::= 9
```

```
FTN-AddressString ::=
            AddressString (SIZE (1..maxFTN-AddressLength))
    -- This type is used to represent forwarded-to numbers.
    -- For long forwarded-to numbers (longer than 15 digits) NPI shall be unknown;
    -- if NAI = international the first digits represent the country code (CC)
    -- and the network destination code (NDC) as for E.164.
```

```
maxFTN-AddressLength  INTEGER ::= 15
```

```
ISDN-SubaddressString ::=
            OCTET STRING (SIZE (1..maxISDN-SubaddressLength))
    -- This type is used to represent ISDN subaddresses.
    -- It is composed of
    --   a)   one octet for type of subaddress and odd/even indicator.
    --   b)   20 octets for subaddress information.

    --   a)   The first octet includes a one bit extension indicator, a
    --        3 bits type of subaddress and a one bit odd/even indicator,
    --        encoded as follows:

    --   bit 8: 1  (no extension)

    --   bits 765: type of subaddress
    --        000  NSAP (X.213/ISO 8348 AD2)
    --        010  User Specified
    --        All other values are reserved

    --   bit 4: odd/even indicator
    --        0  even number of address signals
    --        1  odd number of address signals
    --        The odd/even indicator is used when the type of subaddress
    --        is "user specified" and the coding is BCD.

    --   bits 321: 000 (unused)

    --   b) Subaddress information.
    --   The NSAP X.213/ISO8348AD2 address shall be formatted as specified
    --   by octet 4 which contains the Authority and Format Identifier
    --   (AFI). The encoding is made according to the "preferred binary
    --   encoding" as defined in X.213/ISO834AD2. For the definition
    --   of this type of subaddress, see ~~CCITT~~ITU-T Rec I.334.

    --   For User-specific subaddress, this field is encoded according
    --   to the user specification, subject to a maximum length of 20
    --   octets. When interworking with X.25 networks BCD coding should
    --   be applied.
```

```
maxISDN-SubaddressLength  INTEGER ::= 21
```

```
ExternalSignalInfo ::= SEQUENCE {
    protocolId                      ProtocolId,
    signalInfo                      SignalInfo,
    -- Information about the internal structure is given in
    -- clause 7.6.9.
    extensionContainer              ExtensionContainer              OPTIONAL,
    -- extensionContainer must not be used in version 2
    ...}
```

```
SignalInfo ::= OCTET STRING (SIZE (1..maxSignalInfoLength))
```

```
maxSignalInfoLength  INTEGER ::= 200
    -- This NamedValue represents the theoretical maximum number of octets which is
    -- available to carry a single instance of the SignalInfo data type,
    -- without requiring segmentation to cope with the network layer service.
    -- However, the actual maximum size available for an instance of the data
    -- type may be lower, especially when other information elements
    -- have to be included in the same component.
```

```
ProtocolId ::= ENUMERATED {
     gsm-0408  (1),
     gsm-0806  (2),
     gsm-BSSMAP  (3),
     -- Value 3 is reserved and must not be used
     ets-300102-1  (4)}
```

```
Ext-ExternalSignalInfo ::= SEQUENCE {
     ext-ProtocolId                    Ext-ProtocolId,
     signalInfo                        SignalInfo,
     -- Information about the internal structure is given in
     -- clause 7.6.9.10
     extensionContainer                ExtensionContainer              OPTIONAL,
     ...}
```

```
Ext-ProtocolId ::= ENUMERATED {
     ets-300356  (1),
     ...
      }
-- exception handling:
-- For Ext-ExternalSignalInfo sequences containing this parameter with any
-- other value than the ones listed the receiver shall ignore the whole
-- Ext-ExternalSignalInfo sequence.
```

```
AccessNetworkSignalInfo ::= SEQUENCE {
     accessNetworkProtocolId           AccessNetworkProtocolId,
     signalInfo                        LongSignalInfo,
     -- Information about the internal structure is given in clause 7.6.9.1

     extensionContainer                ExtensionContainer              OPTIONAL,
     ...}
```

```
LongSignalInfo ::= OCTET STRING (SIZE (1..maxLongSignalInfoLength))
```

```
maxLongSignalInfoLength  INTEGER ::= 2560
     -- This Named Value represents the maximum number of octets which is available
     -- to carry a single instance of the LongSignalInfo data type using
     -- White Book SCCP with the maximum number of segments.
     -- It takes account of the octets used by the lower layers of the protocol, and
     -- other information elements which may be included in the same component.
```

```
AccessNetworkProtocolId ::= ENUMERATED {
     ts3G-48006   (1),
     ts3G-25413 (2),
     ...}
     -- exception handling:
     -- For AccessNetworkSignalInfo sequences containing this parameter with any
     -- other value than the ones listed the receiver shall ignore the whole
     -- AccessNetworkSignalInfo sequence.
```

```
AlertingPattern ::= OCTET STRING (SIZE (1) )
     -- This type is used to represent Alerting Pattern

     --   bits 8765 : 0000 (unused)

     --   bits 43 : type of Pattern
     --        00 level
     --        01 category
     --        10 category
     --        all other values are reserved.

     --   bits 21 : type of alerting

alertingLevel-0   AlertingPattern ::= '00000000'B
alertingLevel-1   AlertingPattern ::= '00000001'B
alertingLevel-2   AlertingPattern ::= '00000010'B
     -- all other values of Alerting level are reserved
     -- Alerting Levels are defined in GSM 02.07

alertingCategory-1   AlertingPattern ::= '00000100'B
alertingCategory-2   AlertingPattern ::= '00000101'B
alertingCategory-3   AlertingPattern ::= '00000110'B
alertingCategory-4   AlertingPattern ::= '00000111'B
alertingCategory-5   AlertingPattern ::= '00001000'B
     -- all other values of Alerting Category are reserved
     -- Alerting categories are defined in GSM 02.07
```

-- data types for numbering and identification

```
IMSI ::= TBCD-STRING (SIZE (3..8))
     -- digits of MCC, MNC, MSIN are concatenated in this order.
```

```
Identity ::= CHOICE {
     imsi                                IMSI,
     imsi-WithLMSI                       IMSI-WithLMSI}
```

```
IMSI-WithLMSI ::= SEQUENCE {
     imsi                                IMSI,
     lmsi                                LMSI,
     -- a special value 00000000 indicates that the LMSI is not in use
     ...}
```

```
ASCI-CallReference ::= TBCD-STRING (SIZE (1..8))
     -- digits of VGCS/VBC-area,Group-ID are concatenated in this order.
```

```
TMSI ::= OCTET STRING (SIZE (1..4))
```

```
SubscriberId ::= CHOICE {
     imsi                                [0] IMSI,
     tmsi                                [1] TMSI}
```

```
IMEI ::= TBCD-STRING (SIZE (8))
     --   Refers to International Mobile Station Equipment Identity
     --   and Software Version Number (SVN) defined in TS 3GPP TS 23.003 [17].
     --   If the SVN is not present the last octet shall contain the
     --   digit 0 and a filler.
     --   If present the SVN shall be included in the last octet.
```

```
HLR-Id ::= IMSI
     -- leading digits of IMSI, i.e. (MCC, MNC, leading digits of
     -- MSIN) forming HLR Id defined in TS 3GPP TS 23.003 [17].
```

```
HLR-List ::= SEQUENCE SIZE (1..maxNumOfHLR-Id) OF
                                         HLR-Id
```

```
maxNumOfHLR-Id  INTEGER ::= 50
```

```
LMSI ::= OCTET STRING (SIZE (4))
```

```
GlobalCellId ::= OCTET STRING (SIZE (5..7))
     -- Refers to Cell Global Identification defined in TS 3GPP TS 23.003 [17].
     -- The internal structure is defined as follows:
     -- octet 1 bits 4321                Mobile Country Code 1st digit
     --         bits 8765                Mobile Country Code 2nd digit
     -- octet 2 bits 4321                Mobile Country Code 3rd digit
     --         bits 8765                Mobile Network Code 3rd digit
     --                                  or filler (1111) for 2 digit MNCs
     -- octet 3 bits 4321                Mobile Network Code 1st digit
     --         bits 8765                Mobile Network Code 2nd digit
     -- octets 4 and 5                   Location Area Code according to TS 3GPP TS 24.008
[35]
     -- octets 6 and 7                   Cell Identity (CI) according to TS 3GPP TS 24.008
[35]
```

```
NetworkResource ::= ENUMERATED {
     plmn  (0),
     hlr  (1),
     vlr  (2),
     pvlr  (3),
     controllingMSC  (4),
     vmsc  (5),
     eir  (6),
     rss  (7)}
```

```
NAEA-PreferredCI ::= SEQUENCE {
     naea-PreferredCIC                  [0] NAEA-CIC,
     extensionContainer                 [1] ExtensionContainer          OPTIONAL,
     ...}
```

```
NAEA-CIC ::= OCTET STRING (SIZE (3))
     -- The internal structure is defined by the Carrier Identification
     -- parameter in ANSI T1.113.3. Carrier codes between “000” and “999” may
     -- be encoded as 3 digits using “000” to “999” or as 4 digits using
     -- “0000” to “0999”. Carrier codes between “1000” and “9999” are encoded
     -- using 4 digits.
```

```
SubscriberIdentity ::= CHOICE {
     imsi                               [0] IMSI,
     msisdn                             [1] ISDN-AddressString
     }
```

```
LCSClientExternalID ::= SEQUENCE {
     externalAddress                    [0] AddressString               OPTIONAL,
     extensionContainer                 [1] ExtensionContainer          OPTIONAL,
     ... }
```

```
LCSClientInternalID ::= ENUMERATED {
     broadcastService                   (0),
     o-andM-HPLMN                       (1),
     o-andM-VPLMN                       (2),
     anonymousLocation                  (3),
     targetMSsubscribedService          (4),
     ... }
-- for a CAMEL phase 3 PLMN operator client, the value targetMSsubscribedService shall be used
```

```
LCSServiceTypeID ::= INTEGER (0..127)
     -- the integer values 0-63 are reserved for Standard LCS service types
     -- the integer values 64-127 are reserved for Non Standard LCS service types
```

```
emergencyServices                       LCSServiceTypeID ::= 0
emergencyAlertServices                  LCSServiceTypeID ::= 1
personTracking                          LCSServiceTypeID ::= 2
fleetManagement                         LCSServiceTypeID ::= 3
assetManagement                         LCSServiceTypeID ::= 4
trafficCongestionReporting              LCSServiceTypeID ::= 5
roadsideAssistance                      LCSServiceTypeID ::= 6
routingToNearestCommercialEnterprise    LCSServiceTypeID ::= 7
navigation                              LCSServiceTypeID ::= 8
citySightseeing                         LCSServiceTypeID ::= 9
localizedAdvertising                    LCSServiceTypeID ::= 10
mobileYellowPages                       LCSServiceTypeID ::= 11
-- The values of LCSServiceTypeID are defined according to 3G TS 22.071.
```

-- data types for CAMEL

```
CellGlobalIdOrServiceAreaIdOrLAI ::= CHOICE {
     cellGlobalIdOrServiceAreaIdFixedLength   [0] CellGlobalIdOrServiceAreaIdFixedLength,
     laiFixedLength                           [1] LAIFixedLength}
```

```
CellGlobalIdOrServiceAreaIdFixedLength ::= OCTET STRING (SIZE (7))
     -- Refers to Cell Global Identification or Service Are Identification
     -- defined in 3G TS 23.003.
     -- The internal structure is defined as follows:
     -- octet 1 bits 4321                Mobile Country Code 1st digit
     --         bits 8765                Mobile Country Code 2nd digit
     -- octet 2 bits 4321                Mobile Country Code 3rd digit
     --         bits 8765                Mobile Network Code 3rd digit
     --                                  or filler (1111) for 2 digit MNCs
     -- octet 3 bits 4321                Mobile Network Code 1st digit
     --         bits 8765                Mobile Network Code 2nd digit
     -- octets 4 and 5                   Location Area Code according to 3G TS 24.008
     -- octets 6 and 7                   Cell Identity (CI) value or
     --                                  Service Area Code (SAC) value
     --                                  according to 3G TS 23.003
```

```
LAIFixedLength ::= OCTET STRING (SIZE (5))
     -- Refers to Location Area Identification defined in TS 3GPP TS 23.003 [17].
     -- The internal structure is defined as follows:
     -- octet 1 bits 4321                Mobile Country Code 1st digit
     --         bits 8765                Mobile Country Code 2nd digit
     -- octet 2 bits 4321                Mobile Country Code 3rd digit
     --         bits 8765                Mobile Network Code 3rd digit
     --                                  or filler (1111) for 2 digit MNCs
     -- octet 3 bits 4321                Mobile Network Code 1st digit
     --         bits 8765                Mobile Network Code 2nd digit
     -- octets 4 and 5                   Location Area Code according to TS 3GPP TS 24.008
[35]
```

-- data types for subscriber management

```
BasicServiceCode ::= CHOICE {
     bearerService                    [2] BearerServiceCode,
     teleservice                      [3] TeleserviceCode}
```

```
Ext-BasicServiceCode ::= CHOICE {
     ext-BearerService                [2] Ext-BearerServiceCode,
     ext-Teleservice                  [3] Ext-TeleserviceCode}
```

```
EMLPP-Info ::= SEQUENCE {
     maximumentitledPriority          EMLPP-Priority,
     defaultPriority                  EMLPP-Priority,
     extensionContainer               ExtensionContainer              OPTIONAL,
     ...}
```

```
EMLPP-Priority ::= INTEGER (0..15)
     -- The mapping from the values A,B,0,1,2,3,4 to the integer-value is
     -- specified as follows where A is the highest and 4 is the lowest
     -- priority level
     -- the integer values 7-15 are spare and shall be mapped to value 4
```

```
priorityLevelA                        EMLPP-Priority ::= 6
priorityLevelB                        EMLPP-Priority ::= 5
priorityLevel0                        EMLPP-Priority ::= 0
priorityLevel1                        EMLPP-Priority ::= 1
priorityLevel2                        EMLPP-Priority ::= 2
priorityLevel3                        EMLPP-Priority ::= 3
priorityLevel4                        EMLPP-Priority ::= 4
```

```
MC-SS-Info ::= SEQUENCE {
     ss-Code                          [0] SS-Code,
     ss-Status                        [1] Ext-SS-Status,
     nbrSB                            [2] MaxMC-Bearers,
     nbrUser                          [3] MC-Bearers,
     extensionContainer               [4] ExtensionContainer          OPTIONAL,
     ...}
```

```
MaxMC-Bearers ::= INTEGER (2..maxNumOfMC-Bearers)
```

```
MC-Bearers ::= INTEGER (1..maxNumOfMC-Bearers)
```

```
maxNumOfMC-Bearers  INTEGER ::= 7
```

```
Ext-SS-Status ::= OCTET STRING (SIZE (1..5))

     -- OCTET 1:
     --
     -- bits 8765: 0000 (unused)
     -- bits 4321: Used to convey the "P bit","R bit","A bit" and "Q bit",
     --            representing supplementary service state information
     --            as defined in TS 3GPP TS 23.011 [22]

     -- bit 4: "Q bit"

     -- bit 3: "P bit"

     -- bit 2: "R bit"

     -- bit 1: "A bit"

     -- OCTETS 2-5: reserved for future use. They shall be discarded if
     -- received and not understood.
```

```
   -- data types for geographic location

AgeOfLocationInformation ::= INTEGER (0..32767)
-- the value represents the elapsed time in minutes since the last
-- network contact of the mobile station (i.e. the actuality of the
-- location information).
-- value “0” indicates that the MS is currently in contact with the
--           network
-- value “32767” indicates that the location information is at least
--               32767 minutes old

END
```

17.7.9 Teleservice Codes

```
MAP-TS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-TS-Code (19) version8 (8)}

DEFINITIONS

::=

BEGIN
```

```
TeleserviceCode ::= OCTET STRING (SIZE (1))
     -- This type is used to represent the code identifying a single
     -- teleservice, a group of teleservices, or all teleservices. The
     -- services are defined in TS GSM 22.003 [4].
     -- The internal structure is defined as follows:

     -- bits 87654321: group (bits 8765) and specific service
     -- (bits 4321)
```

```
Ext-TeleserviceCode ::= OCTET STRING (SIZE (1..5))
     -- This type is used to represent the code identifying a single
     -- teleservice, a group of teleservices, or all teleservices. The
     -- services are defined in TS GSM 22.003 [4].
     -- The internal structure is defined as follows:

     -- OCTET 1:
     -- bits 87654321: group (bits 8765) and specific service
     -- (bits 4321)

     -- OCTETS 2-5: reserved for future use. If received the
    -- Ext-TeleserviceCode shall be
     -- treated according to the exception handling defined for the
     -- operation that uses this type.

     -- Ext-TeleserviceCode includes all values defined for TeleserviceCode.
```

```
allTeleservices                         TeleserviceCode ::= '00000000'B
```

```
allSpeechTransmissionServices           TeleserviceCode ::= '00010000'B
telephony                               TeleserviceCode ::= '00010001'B
emergencyCalls                          TeleserviceCode ::= '00010010'B
```

```
allShortMessageServices                 TeleserviceCode ::= '00100000'B
shortMessageMT-PP                       TeleserviceCode ::= '00100001'B
shortMessageMO-PP                       TeleserviceCode ::= '00100010'B
```

```
allFacsimileTransmissionServices        TeleserviceCode ::= '01100000'B
facsimileGroup3AndAlterSpeech           TeleserviceCode ::= '01100001'B
automaticFacsimileGroup3                TeleserviceCode ::= '01100010'B
facsimileGroup4                         TeleserviceCode ::= '01100011'B
```

```
-- The following non-hierarchical Compound Teleservice Groups
-- are defined in TS 3GPP TS 22.030:
allDataTeleservices                     TeleserviceCode ::= '01110000'B
     -- covers Teleservice Groups 'allFacsimileTransmissionServices'
     -- and 'allShortMessageServices'
allTeleservices-ExeptSMS                TeleserviceCode ::= '10000000'B
     -- covers Teleservice Groups 'allSpeechTransmissionServices' and
     -- 'allFacsimileTransmissionServices'
--
-- Compound Teleservice Group Codes are only used in call
-- independent supplementary service operations, i.e. they
-- are not used in InsertSubscriberData or in
-- DeleteSubscriberData messages.
```

```
allVoiceGroupCallServices               TeleserviceCode ::= '10010000'B
voiceGroupCall                          TeleserviceCode ::= '10010001'B
voiceBroadcastCall                      TeleserviceCode ::= '10010010'B
```

```
allPLMN-specificTS                      TeleserviceCode ::= '11010000'B
plmn-specificTS-1                       TeleserviceCode ::= '11010001'B
plmn-specificTS-2                       TeleserviceCode ::= '11010010'B
plmn-specificTS-3                       TeleserviceCode ::= '11010011'B
plmn-specificTS-4                       TeleserviceCode ::= '11010100'B
plmn-specificTS-5                       TeleserviceCode ::= '11010101'B
plmn-specificTS-6                       TeleserviceCode ::= '11010110'B
plmn-specificTS-7                       TeleserviceCode ::= '11010111'B
plmn-specificTS-8                       TeleserviceCode ::= '11011000'B
plmn-specificTS-9                       TeleserviceCode ::= '11011001'B
plmn-specificTS-A                       TeleserviceCode ::= '11011010'B
plmn-specificTS-B                       TeleserviceCode ::= '11011011'B
plmn-specificTS-C                       TeleserviceCode ::= '11011100'B
plmn-specificTS-D                       TeleserviceCode ::= '11011101'B
plmn-specificTS-E                       TeleserviceCode ::= '11011110'B
plmn-specificTS-F                       TeleserviceCode ::= '11011111'B
```

```
END
```

17.7.10 Bearer Service Codes

```
MAP-BS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-BS-Code (20) version8 (8)}

DEFINITIONS

::=

BEGIN
```

```
BearerServiceCode ::= OCTET STRING (SIZE (1))
     -- This type is used to represent the code identifying a single
     -- bearer service, a group of bearer services, or all bearer
     -- services. The services are defined in TS 3GPP TS 22.002 [3].
     -- The internal structure is defined as follows:
     --
     -- plmn-specific bearer services:
     -- bits 87654321: defined by the HPLMN operator

     -- rest of bearer services:
     -- bit 8: 0 (unused)
     -- bits 7654321: group (bits 7654), and rate, if applicable
     -- (bits 321)
```

```
Ext-BearerServiceCode ::= OCTET STRING (SIZE (1..5))
     -- This type is used to represent the code identifying a single
     -- bearer service, a group of bearer services, or all bearer
     -- services. The services are defined in TS 3GPP TS 22.002 [3].
     -- The internal structure is defined as follows:
     --
     -- OCTET 1:
     -- plmn-specific bearer services:
     -- bits 87654321: defined by the HPLMN operator
     --
     -- rest of bearer services:
     -- bit 8: 0 (unused)
     -- bits 7654321: group (bits 7654), and rate, if applicable
     -- (bits 321)

     -- OCTETS 2-5: reserved for future use. If received the
    -- Ext-TeleserviceCode shall be
     -- treated according to the exception handling defined for the
     -- operation that uses this type.


     -- Ext-BearerServiceCode includes all values defined for BearerServiceCode.
```

```
allBearerServices                    BearerServiceCode ::= '00000000'B
```

```
allDataCDA-Services                  BearerServiceCode ::= '00010000'B
dataCDA-300bps                       BearerServiceCode ::= '00010001'B
dataCDA-1200bps                      BearerServiceCode ::= '00010010'B
dataCDA-1200-75bps                   BearerServiceCode ::= '00010011'B
dataCDA-2400bps                      BearerServiceCode ::= '00010100'B
dataCDA-4800bps                      BearerServiceCode ::= '00010101'B
dataCDA-9600bps                      BearerServiceCode ::= '00010110'B
general-dataCDA                      BearerServiceCode ::= '00010111'B
```

```
allDataCDS-Services                  BearerServiceCode ::= '00011000'B
dataCDS-1200bps                      BearerServiceCode ::= '00011010'B
dataCDS-2400bps                      BearerServiceCode ::= '00011100'B
dataCDS-4800bps                      BearerServiceCode ::= '00011101'B
dataCDS-9600bps                      BearerServiceCode ::= '00011110'B
general-dataCDS                      BearerServiceCode ::= '00011111'B
```

```
allPadAccessCA-Services              BearerServiceCode ::= '00100000'B
padAccessCA-300bps                   BearerServiceCode ::= '00100001'B
padAccessCA-1200bps                  BearerServiceCode ::= '00100010'B
padAccessCA-1200-75bps               BearerServiceCode ::= '00100011'B
padAccessCA-2400bps                  BearerServiceCode ::= '00100100'B
padAccessCA-4800bps                  BearerServiceCode ::= '00100101'B
padAccessCA-9600bps                  BearerServiceCode ::= '00100110'B
general-padAccessCA                  BearerServiceCode ::= '00100111'B
```

```
allDataPDS-Services                  BearerServiceCode ::= '00101000'B
dataPDS-2400bps                      BearerServiceCode ::= '00101100'B
dataPDS-4800bps                      BearerServiceCode ::= '00101101'B
dataPDS-9600bps                      BearerServiceCode ::= '00101110'B
general-dataPDS                      BearerServiceCode ::= '00101111'B
```

```
allAlternateSpeech-DataCDA           BearerServiceCode ::= '00110000'B
```

```
allAlternateSpeech-DataCDS           BearerServiceCode ::= '00111000'B
```

```
allSpeechFollowedByDataCDA           BearerServiceCode ::= '01000000'B
```

```
allSpeechFollowedByDataCDS                  BearerServiceCode ::= '01001000'B
```

```
-- The following non-hierarchical Compound Bearer Service
-- Groups are defined in TS 3GPP TS 22.030:
allDataCircuitAsynchronous                  BearerServiceCode ::= '01010000'B
     -- covers "allDataCDA-Services", "allAlternateSpeech-DataCDA" and
     -- "allSpeechFollowedByDataCDA"
allAsynchronousServices                     BearerServiceCode ::= '01100000'B
     -- covers "allDataCDA-Services", "allAlternateSpeech-DataCDA",
     -- "allSpeechFollowedByDataCDA" and "allPadAccessCDA-Services"
allDataCircuitSynchronous                   BearerServiceCode ::= '01011000'B
     -- covers "allDataCDS-Services", "allAlternateSpeech-DataCDS" and
     -- "allSpeechFollowedByDataCDS"
allSynchronousServices                      BearerServiceCode ::= '01101000'B
     -- covers "allDataCDS-Services", "allAlternateSpeech-DataCDS",
     -- "allSpeechFollowedByDataCDS" and "allDataPDS-Services"
--
-- Compound Bearer Service Group Codes are only used in call
-- independent supplementary service operations, i.e. they
-- are not used in InsertSubscriberData or in
-- DeleteSubscriberData messages.
```

```
allPLMN-specificBS                          BearerServiceCode ::= '11010000'B
plmn-specificBS-1                           BearerServiceCode ::= '11010001'B
plmn-specificBS-2                           BearerServiceCode ::= '11010010'B
plmn-specificBS-3                           BearerServiceCode ::= '11010011'B
plmn-specificBS-4                           BearerServiceCode ::= '11010100'B
plmn-specificBS-5                           BearerServiceCode ::= '11010101'B
plmn-specificBS-6                           BearerServiceCode ::= '11010110'B
plmn-specificBS-7                           BearerServiceCode ::= '11010111'B
plmn-specificBS-8                           BearerServiceCode ::= '11011000'B
plmn-specificBS-9                           BearerServiceCode ::= '11011001'B
plmn-specificBS-A                           BearerServiceCode ::= '11011010'B
plmn-specificBS-B                           BearerServiceCode ::= '11011011'B
plmn-specificBS-C                           BearerServiceCode ::= '11011100'B
plmn-specificBS-D                           BearerServiceCode ::= '11011101'B
plmn-specificBS-E                           BearerServiceCode ::= '11011110'B
plmn-specificBS-F                           BearerServiceCode ::= '11011111'B
```

```
END
```

17.7.11 Extension data types

```
MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS

   PrivateExtension,
   ExtensionContainer;


-- IOC for private MAP extensions
```

```
MAP-EXTENSION  ::= CLASS {
    &ExtensionType                                              OPTIONAL,
    &extensionId                    OBJECT IDENTIFIER }
    -- The length of the Object Identifier shall not exceed 16 octets and the
    -- number of components of the Object Identifier shall not exceed 16
```

```
-- data types
```

```
ExtensionContainer ::= SEQUENCE {
    privateExtensionList            [0]PrivateExtensionList         OPTIONAL,
    pcs-Extensions                  [1]PCS-Extensions               OPTIONAL,
```

```
    ...}

PrivateExtensionList ::= SEQUENCE SIZE (1..maxNumOfPrivateExtensions) OF
                                    PrivateExtension

PrivateExtension ::= SEQUENCE {
    extId                           MAP-EXTENSION.&extensionId
                                    ({ExtensionSet}),
    extType                         MAP-EXTENSION.&ExtensionType
                                    ({ExtensionSet}{@extId})        OPTIONAL}

maxNumOfPrivateExtensions  INTEGER ::= 10

ExtensionSet                        MAP-EXTENSION ::=
        {...
         -- ExtensionSet is the set of all defined private extensions
    }

    -- Unsupported private extensions shall be discarded if received.


PCS-Extensions ::= SEQUENCE {
    ...}

END
```

17.7.12 Group Call data types

```
MAP-GR-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-GR-DataTypes (23) version8 (8)}

DEFINITIONS

IMPLICIT TAGS

::=

BEGIN

EXPORTS
   PrepareGroupCallArg,
   PrepareGroupCallRes,
   SendGroupCallEndSignalArg,
   SendGroupCallEndSignalRes,
   ForwardGroupCallSignallingArg,
   ProcessGroupCallSignallingArg
;

IMPORTS
   ISDN-AddressString,
   IMSI,
   EMLPP-Priority,
   ASCI-CallReference
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   Ext-TeleserviceCode
FROM MAP-TS-Code {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-TS-Code (19) version8 (8)}

   Kc
FROM MAP-MS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-MS-DataTypes (11) version8 (8)}


   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}
;
```

```
PrepareGroupCallArg ::= SEQUENCE {
     teleservice                        Ext-TeleserviceCode,
     asciCallReference                  ASCI-CallReference,
     codec-Info                         CODEC-Info,
     cipheringAlgorithm                 CipheringAlgorithm,
     groupKeyNumber                     [0] GroupKeyNumber              OPTIONAL,
     groupKey                           [1] Kc                          OPTIONAL,
     priority                           [2] EMLPP-Priority              OPTIONAL,
     uplinkFree                         [3] NULL                        OPTIONAL,
     extensionContainer                 [4] ExtensionContainer          OPTIONAL,
     ...}
```

```
PrepareGroupCallRes ::= SEQUENCE {
     groupCallNumber                    ISDN-AddressString,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
SendGroupCallEndSignalArg ::= SEQUENCE {
     imsi                               IMSI                            OPTIONAL,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
SendGroupCallEndSignalRes ::= SEQUENCE {
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
ForwardGroupCallSignallingArg ::= SEQUENCE {
     imsi                               IMSI                            OPTIONAL,
     uplinkRequestAck                   [0] NULL                        OPTIONAL,
     uplinkReleaseIndication            [1] NULL                        OPTIONAL,
     uplinkRejectCommand                [2] NULL                        OPTIONAL,
     uplinkSeizedCommand                [3] NULL                        OPTIONAL,
     uplinkReleaseCommand               [4] NULL                        OPTIONAL,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...,
     stateAttributes                    [5] StateAttributes             OPTIONAL }
```

```
ProcessGroupCallSignallingArg ::= SEQUENCE {
     uplinkRequest                      [0] NULL                        OPTIONAL,
     uplinkReleaseIndication            [1] NULL                        OPTIONAL,
     releaseGroupCall                   [2] NULL                        OPTIONAL,
     extensionContainer                 ExtensionContainer              OPTIONAL,
     ...}
```

```
GroupKeyNumber ::= INTEGER (0..15)
```

```
CODEC-Info ::= OCTET STRING (SIZE (5..10))
     -- Refers to channel type
     -- coded according to 3GPP TS 48.008 [49] and including Element identifier and Length
```

```
CipheringAlgorithm ::= OCTET STRING (SIZE (1))
     -- Refers to 'permitted algorithms' in 'encryption information'
     -- coded according to 3GPP TS 48.008 [49]:

     -- Bits 8-1
     -- 8765 4321
     -- 0000 0001                      No encryption
     -- 0000 0010                      GSM A5/1
     -- 0000 0100                      GSM A5/2
     -- 0000 1000                      GSM A5/3
     -- 0001 0000                      GSM A5/4
     -- 0010 0000                      GSM A5/5
     -- 0100 0000                      GSM A5/6
     -- 1000 0000                      GSM A5/7
```

```
StateAttributes ::= SEQUENCE {
     downlinkAttached                              [5] NULL                  OPTIONAL,
     uplinkAttached                                [6] NULL                  OPTIONAL,
     dualCommunication                             [7] NULL                  OPTIONAL,
     callOriginator                                [8] NULL                  OPTIONAL }

     -- Refers to 3GPP TS 44.068 for definitions of StateAttributes fields.


END
```

17.7.13 Location service data types

```
MAP-LCS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-LCS-DataTypes (25) version8 (8)}

DEFINITIONS
IMPLICIT TAGS
::=
BEGIN

EXPORTS
   RoutingInfoForLCS-Arg,
   RoutingInfoForLCS-Res,
   ProvideSubscriberLocation-Arg,
   ProvideSubscriberLocation-Res,
   SubscriberLocationReport-Arg,
   SubscriberLocationReport-Res,
   LocationType,
   LCSClientName,
   LCS-QoS,
   Horizontal-Accuracy,
   ResponseTime,
   Ext-GeographicalInformation,
   SupportedGADShapes,
   Add-GeographicalInformation,
   LCSRequestorID,
   LCSCodeword
;

IMPORTS
   AddressString,
   ISDN-AddressString,
   IMEI,
   IMSI,
   LMSI,
   SubscriberIdentity,
   AgeOfLocationInformation,
   LCSClientExternalID,
   LCSClientInternalID,
   LCSServiceTypeID
FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}

   ExtensionContainer
FROM MAP-ExtensionDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ExtensionDataTypes (21) version8 (8)}

   USSD-DataCodingScheme,
   USSD-String
FROM MAP-SS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0) gsm-Network (1) modules (3)
   map-SS-DataTypes (14) version8 (8)}

   APN
FROM MAP-MS-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-MS-DataTypes (11) version8 (8)}

   Additional-Number
FROM MAP-SM-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-SM-DataTypes (16) version8 (8)}
```

```
;


RoutingInfoForLCS-Arg ::= SEQUENCE {
     mlcNumber                          [0] ISDN-AddressString,
     targetMS                           [1] SubscriberIdentity,
     extensionContainer                 [2] ExtensionContainer            OPTIONAL,
     ...,
     lcsCodewordApplicability           [3]  LCSCodewordApplicability     OPTIONAL }

LCSCodewordApplicability ::= ENUMERATED {
     codewordCheckApplicable            (0),
     codewordCheckNotApplicable         (1),
     ...}
--   exception handling:
--   unrecognized values shall be ignored by the receiver.

RoutingInfoForLCS-Res ::= SEQUENCE {
     targetMS                           [0] SubscriberIdentity,
     lcsLocationInfo                    [1] LCSLocationInfo,
     extensionContainer                 [2] ExtensionContainer            OPTIONAL,
     ...,
     lcsCodewordNotification            [3]  NULL                         OPTIONAL
     -- lcsCodewordNotification may be present only if
     -- lcsCodewordApplicability was present in RoutingInfoForLCS-Arg.
     -- If received when lcsCodewordApplicability was not present in
     -- RoutingInfoForLCS-Arg then lcsCodewordNotification shall be ignored.
}

LCSLocationInfo ::= SEQUENCE {
     networkNode-Number                 ISDN-AddressString,
     -- NetworkNode-number can be either msc-number or sgsn-number
     lmsi                               [0] LMSI                          OPTIONAL,
     extensionContainer                 [1] ExtensionContainer            OPTIONAL,
     ... ,
     gprsNodeIndicator                  [2] NULL                          OPTIONAL,
     -- gprsNodeIndicator is set only if the SGSN number is sent as the Network Node Number
     additional-Number                  [3] Additional-Number             OPTIONAL
     }

ProvideSubscriberLocation-Arg ::= SEQUENCE {
     locationType                       LocationType,
     mlc-Number                         ISDN-AddressString,
     lcs-ClientID                       [0] LCS-ClientID                  OPTIONAL,
     privacyOverride                    [1] NULL                          OPTIONAL,
     imsi                               [2] IMSI                          OPTIONAL,
     msisdn                             [3] ISDN-AddressString            OPTIONAL,
     lmsi                               [4] LMSI                          OPTIONAL,
     imei                               [5] IMEI                          OPTIONAL,
     lcs-Priority                       [6] LCS-Priority                  OPTIONAL,
     lcs-QoS                            [7] LCS-QoS                       OPTIONAL,
     extensionContainer                 [8] ExtensionContainer            OPTIONAL,
     ... ,
     supportedGADShapes                 [9]  SupportedGADShapes           OPTIONAL,
     lcsServiceTypeID                   [10] LCSServiceTypeID             OPTIONAL,
     lcsCodeword                        [11] LCSCodeword                  OPTIONAL }

     -- one of imsi or msisdn is mandatory

LocationType ::= SEQUENCE {
     locationEstimateType               [0] LocationEstimateType,
     ...,
     deferredLocationEventType          [1] DeferredLocationEventType     OPTIONAL }

LocationEstimateType ::= ENUMERATED {
     currentLocation                    (0),
     currentOrLastKnownLocation         (1),
     initialLocation                    (2),
     ...,
     activateDeferredLocation           (3),
     cancelDeferredLocation             (4) }
--   exception handling:
--   a ProvideSubscriberLocation-Arg containing an unrecognized LocationEstimateType
--   shall be rejected by the receiver with a return error cause of unexpected data value
```

```
DeferredLocationEventType ::= BIT STRING {
     msAvailable                          (0) } (SIZE (1..16))
-- exception handling
-- a ProvideSubscriberLocation-Arg containing other values than listed above in
-- DeferredLocationEventType shall be rejected by the receiver with a return error cause of
-- unexpected data value.
```

```
LCS-ClientID ::= SEQUENCE {
     lcsClientType                        [0] LCSClientType,
     lcsClientExternalID                  [1] LCSClientExternalID          OPTIONAL,
     lcsClientDialedByMS                  [2] AddressString                OPTIONAL,
     lcsClientInternalID                  [3] LCSClientInternalID          OPTIONAL,
     lcsClientName                        [4] LCSClientName                OPTIONAL,
     ...,
     lcsAPN                               [5] APN                          OPTIONAL,
     lcsRequestorID                       [6] LCSRequestorID               OPTIONAL }
```

```
LCSClientType ::= ENUMERATED {
     emergencyServices                    (0),
     valueAddedServices                   (1),
     plmnOperatorServices                 (2),
     lawfulInterceptServices              (3),
     ... }
     --   exception handling:
     --   unrecognized values may be ignored if the LCS client uses the privacy override
     --   otherwise, an unrecognized value shall be treated as unexpected data by a receiver
     --   a return error shall then be returned if received in a MAP invoke
```

```
LCSClientName ::= SEQUENCE {
     dataCodingScheme                     [0] USSD-DataCodingScheme,
     nameString                           [2] NameString,
     ...}

-- The USSD-DataCodingScheme shall indicate use of the default alphabet through the
-- following encoding
--   bit  7 6 5 4 3 2 1 0
--        0 0 0 0 1 1 1 1
```

```
NameString ::= USSD-String (SIZE (1..maxNameStringLength))
```

```
maxNameStringLength  INTEGER ::= 63
```

```
LCSRequestorID ::= SEQUENCE {
     dataCodingScheme                     [0] USSD-DataCodingScheme,
     requestorIDString                    [1] RequestorIDString,
     ...}
```

```
RequestorIDString ::= USSD-String (SIZE (10..maxRequestorIDStringLength))
```

```
maxRequestorIDStringLength  INTEGER ::= 127
```

```
LCS-Priority ::= OCTET STRING (SIZE (1))
     -- 0 = highest priority
     -- 1 = normal priority
     -- all other values treated as 1
```

```
LCS-QoS ::= SEQUENCE {
     horizontal-accuracy                  [0] Horizontal-Accuracy          OPTIONAL,
     verticalCoordinateRequest            [1] NULL                         OPTIONAL,
     vertical-accuracy                    [2] Vertical-Accuracy            OPTIONAL,
     responseTime                         [3] ResponseTime                 OPTIONAL,
     extensionContainer                   [4] ExtensionContainer           OPTIONAL,
     ...}
```

```
Horizontal-Accuracy ::= OCTET STRING (SIZE (1))
     -- bit 8 = 0
     -- bits 7-1 = 7 bit Uncertainty Code defined in 3G TS 23.032. The horizontal location
     -- error should be less than the error indicated by the uncertainty code with 67%
     -- confidence.
```

```
Vertical-Accuracy ::= OCTET STRING (SIZE (1))
     -- bit 8 = 0
     -- bits 7-1 = 7 bit Vertical Uncertainty Code defined in 3G TS 23.032.
     -- The vertical location error should be less than the error indicated
     -- by the uncertainty code with 67% confidence.
```

```
ResponseTime ::= SEQUENCE {
     responseTimeCategory                ResponseTimeCategory,
     ...}
--   note: an expandable SEQUENCE simplifies later addition of a numeric response time.
```

```
ResponseTimeCategory ::= ENUMERATED {
     lowdelay  (0),
     delaytolerant  (1),
     ... }
--   exception handling:
--   an unrecognized value shall be treated the same as value 1 (delaytolerant)
```

```
SupportedGADShapes ::= BIT STRING {
     ellipsoidPoint  (0),
     ellipsoidPointWithUncertaintyCircle (1),
     ellipsoidPointWithUncertaintyEllipse (2),
     polygon (3),
     ellipsoidPointWithAltitude (4),
     ellipsoidPointWithAltitudeAndUncertaintyElipsoid (5),
     ellipsoidArc  (6) } (SIZE (7..16))
-- A node shall mark in the BIT STRING all Shapes defined in 3G TS 23.032 it supports.
-- exception handling: bits 7 to 15 shall be ignored if received.
```

```
LCSCodeword ::= SEQUENCE {
     dataCodingScheme                    [0] USSD-DataCodingScheme,
     lcsCodewordString                   [1] LCSCodewordString,
     ...}
```

```
LCSCodewordString ::= USSD-String (SIZE (1..maxLCSCodewordStringLength))
```

```
maxLCSCodewordStringLength  INTEGER ::= 127
```

```
ProvideSubscriberLocation-Res ::= SEQUENCE {
     locationEstimate                    Ext-GeographicalInformation,
     ageOfLocationEstimate               [0] AgeOfLocationInformation      OPTIONAL,
     extensionContainer                  [1] ExtensionContainer            OPTIONAL,
     ... ,
     add-LocationEstimate                [2] Add-GeographicalInformation   OPTIONAL,
     deferredmt-lrResponseIndicator      [3] NULL                          OPTIONAL }

--   if deferredmt-lrResponseIndicator is set, locationEstimate is ignored.

-- the add-LocationEstimate parameter shall not be sent to a node that did not indicate the
-- geographic shapes supported in the ProvideSubscriberLocation-Arg
-- The locationEstimate and the add-locationEstimate parameters shall not be sent if
-- the supportedGADShapes parameter has been received in ProvideSubscriberLocation-Arg
-- and the shape encoded in locationEstimate or add-LocationEstimate is not marked
-- as supported in supportedGADShapes. In such a case ProvideSubscriberLocation
-- shall be rejected with error FacilityNotSupported with additional indication
-- shapeOfLocationEstimateNotSupported
```

```
Ext-GeographicalInformation ::= OCTET STRING (SIZE (1..maxExt-GeographicalInformation))
    -- Refers to geographical Information defined in 3G TS 23.032.
    -- This is composed of 1 or more octets with an internal structure according to
    -- 3G TS 23.032
    -- Octet 1: Type of shape, only the following shapes in 3G TS 23.032 are allowed:
    --       (a) Ellipsoid point with uncertainty circle
    --       (b) Ellipsoid point with uncertainty ellipse
    --       (c) Ellipsoid point with altitude and uncertainty ellipsoid
    --       (d) Ellipsoid Arc
    --       (e) Ellipsoid Point
    -- Any other value in octet 1 shall be treated as invalid
    -- Octets 2 to 8 for case (a) – Ellipsoid point with uncertainty circle
    --       Degrees of Latitude                                        3 octets
    --       Degrees of Longitude                                       3 octets
    --       Uncertainty code                                           1 octet
    -- Octets 2 to 11 for case (b) – Ellipsoid point with uncertainty ellipse:
    --       Degrees of Latitude                                        3 octets
    --       Degrees of Longitude                                       3 octets
    --       Uncertainty semi-major axis                                1 octet
    --       Uncertainty semi-minor axis                                1 octet
    --       Angle of major axis                                        1 octet
    --       Confidence                                                 1 octet
    -- Octets 2 to 14 for case (c) – Ellipsoid point with altitude and uncertainty ellipsoid
    --       Degrees of Latitude                                        3 octets
    --       Degrees of Longitude                                       3 octets
    --       Altitude                                                   2 octets
    --       Uncertainty semi-major axis                                1 octet
    --       Uncertainty semi-minor axis                                1 octet
    --       Angle of major axis                                        1 octet
    --       Uncertainty altitude                                       1 octet
    --       Confidence                                                 1 octet
    -- Octets 2 to 13 for case (d) – Ellipsoid Arc
    --       Degrees of Latitude                                        3 octets
    --       Degrees of Longitude                                       3 octets
    --       Inner radius                                               2 octets
    --       Uncertainty radius                                         1 octet
    --       Offset angle                                               1 octet
    --       Included angle                                             1 octet
    --       Confidence                                                 1 octet
    -- Octets 2 to 7 for case (e) – Ellipsoid Point
    --       Degrees of Latitude                                        3 octets
    --       Degrees of Longitude                                       3 octets

    --
    -- An Ext-GeographicalInformation parameter comprising more than one octet and
    -- containing any other shape or an incorrect number of octets or coding according
    -- to 3G TS 23.032 shall be treated as invalid data by a receiver.
    --
    -- An Ext-GeographicalInformation parameter comprising one octet shall be discarded
    -- by the receiver if an Add-GeographicalInformation parameter is received
    -- in the same message.
    --
    -- An Ext-GeographicalInformation parameter comprising one octet shall be treated as
    -- invalid data by the receiver if an Add-GeographicalInformation parameter is not
    -- received in the same message.

maxExt-GeographicalInformation  INTEGER ::= 20
    -- the maximum length allows for further shapes in 3G TS 23.032 to be included in later
    -- versions of 3G TS 29.002

Add-GeographicalInformation ::= OCTET STRING (SIZE (1..maxAdd-GeographicalInformation))
    -- Refers to geographical Information defined in 3G TS 23.032.
    -- This is composed of 1 or more octets with an internal structure according to
    -- 3G TS 23.032
    -- Octet 1: Type of shape, all the shapes defined in 3G TS 23.032 are allowed:
    -- Octets 2 to n (where n is the total number of octets necessary to encode the shape
    -- according to 3G TS 23.032) are used to encode the shape itself in accordance with the
    -- encoding defined in 3G TS 23.032
    --
    -- An Add-GeographicalInformation parameter, whether valid or invalid, received
    -- together with a valid Ext-GeographicalInformation parameter in the same message
    -- shall be discarded.
    --
    -- An Add-GeographicalInformation parameter containing any shape not defined in
    -- 3G TS 23.032 or an incorrect number of octets or coding according to
    -- 3G TS 23.032 shall be treated as invalid data by a receiver if not received
    -- together with a valid Ext-GeographicalInformation parameter in the same message.
```

```
maxAdd-GeographicalInformation  INTEGER ::= 91
     -- the maximum length allows support for all the shapes currently defined in 3G TS 23.032

SubscriberLocationReport-Arg ::= SEQUENCE {
     lcs-Event                          LCS-Event,
     lcs-ClientID                       LCS-ClientID,
     lcsLocationInfo                    LCSLocationInfo,
     msisdn                             [0] ISDN-AddressString            OPTIONAL,
     imsi                               [1] IMSI                          OPTIONAL,
     imei                               [2] IMEI                          OPTIONAL,
     na-ESRD                            [3] ISDN-AddressString            OPTIONAL,
     na-ESRK                            [4] ISDN-AddressString            OPTIONAL,
     locationEstimate                   [5] Ext-GeographicalInformation   OPTIONAL,
     ageOfLocationEstimate              [6] AgeOfLocationInformation      OPTIONAL,
     extensionContainer                 [7] ExtensionContainer            OPTIONAL,
     ... ,
     add-LocationEstimate               [8] Add-GeographicalInformation   OPTIONAL,
     deferredmt-lrData                  [9] Deferredmt-lrData             OPTIONAL }

     -- one of msisdn or imsi is mandatory
     -- a location estimate that is valid for the locationEstimate parameter should
     -- be transferred in this parameter in preference to the add-LocationEstimate.
     -- the deferredmt-lrData parameter shall be included if and only if the lcs-Event
     -- indicates a deferredmt-lrResponse.
     -- if the lcs-Event indicates a deferredmt-lrResponse then the locationEstimate
     -- and the add-locationEstimate parameters shall not be sent if the
     -- supportedGADShapes parameter had been received in ProvideSubscriberLocation-Arg
     -- and the shape encoded in locationEstimate or add-LocationEstimate was not marked
     -- as supported in supportedGADShapes. In such a case terminationCause
     -- in deferredmt-lrData shall be present with value
     -- shapeOfLocationEstimateNotSupported.

Deferredmt-lrData ::= SEQUENCE {
     deferredLocationEventType          DeferredLocationEventType,
     terminationCause                   [0] TerminationCause              OPTIONAL,
     lcsLocationInfo                    [1] LCSLocationInfo               OPTIONAL,
     ...}
     -- lcsLocationInfo may be included only if a terminationCause is present
     -- indicating mt-lrRestart.

LCS-Event ::= ENUMERATED {
     emergencyCallOrigination  (0),
     emergencyCallRelease  (1),
     mo-lr  (2),
     ...,
     deferredmt-lrResponse  (3) }
     --   exception handling:
     --   a SubscriberLocationReport-Arg containing an unrecognized LCS-Event
     --   shall be rejected by a receiver with a return error cause of unexpected data value

TerminationCause ::= ENUMERATED {
     normal  (0),
     errorundefined  (1),
     internalTimeout  (2),
     congestion  (3),
     mt-lrRestart  (4),
     privacyViolation  (5),
     ...,
     shapeOfLocationEstimateNotSupported (6) }
-- mt-lrRestart shall be used to trigger the GMLC to restart the location procedure,
-- either because the sending node knows that the terminal has moved under coverage
-- of another MSC or SGSN (e.g. Send Identification received), or because the subscriber
-- has been deregistered due to a Cancel Location received from HLR.
--
-- exception handling
-- an unrecognized value shall be treated the same as value 1 (errorundefined)

SubscriberLocationReport-Res ::= SEQUENCE {
     extensionContainer                 ExtensionContainer                OPTIONAL,
     ...}


END
```

17.7.14 Secure transport data types

```
MAP-ST-DataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-ST-DataTypes (27) version8 (8)}

DEFINITIONS
IMPLICIT TAGS
::=
BEGIN

EXPORTS
   SecureTransportArg,
   SecureTransportRes,
   SecurityHeader,
   ProtectedPayload
;

IMPORTS
   IMSI

FROM MAP-CommonDataTypes {
   ccittitu-t identified-organization (4) etsi (0) mobileDomain (0)
   gsm-Network (1) modules (3) map-CommonDataTypes (18) version8 (8)}
;
```

```
SecureTransportArg ::= SEQUENCE {
     securityHeader                          SecurityHeader,
     protectedPayload                        ProtectedPayload            OPTIONAL
     }
     -- The protectedPayload carries the result of applying the security function
     -- defined in 3G TS 33.200 to the encoding of the argument of the securely
     -- transported operation
```

```
SecureTransportRes ::= SEQUENCE {
     securityHeader                          SecurityHeader,
     protectedPayload                        ProtectedPayload            OPTIONAL
     }
     -- The protectedPayload carries the result of applying the security function
     -- defined in 3G TS 33.200 to the encoding of the result of the securely
     -- transported operation
```

```
SecurityHeader ::= SEQUENCE {
     securityParametersIndex             SecurityParametersIndex,
     originalComponentIdentifier         OriginalComponentIdentifier,
     initialisationVector                InitialisationVector            OPTIONAL,
     ...}
```

```
ProtectedPayload ::= OCTET STRING(SIZE(1.. 3438))
     -- In protection mode 0 (noProtection) the ProtectedPayload carries the transfer
          -- syntax value of the component parameter identified by the
          -- originalComponentIdentifier.
     -- In protection mode 1 (integrityAuthenticity) the protectedPayload carries
          --  the transfer syntax value of the component
          -- parameter identified by the originalComponentIdentifier, followed by
          -- the 32 bit integrity check value.
          -- The integrity check value is the result of applying the hash algorithm
          -- to the concatenation of the transfer syntax value of the SecurityHeader,
          -- and the transfer syntax value of the component parameter.
     -- In protection mode 2 (confidentialityIntegrityAuthenticity) the protected
          -- payload carries the encrypted transfer syntax
          -- value of the component parameter identified by the
          -- originalComponentIdentifier, followed by the 32 bit integrity check value.
          -- The integrity check value is the result of applying the hash algorithm
          -- to the concatenation of the transfer syntax value of the SecurityHeader,
          -- and the encrypted transfer syntax value of the component parameter.
     -- See 33.200.
     -- The length of the protectedPayload is adjusted according to the capabilities of
     -- the lower protocol layers
```

```
SecurityParametersIndex ::= OCTET STRING (SIZE(4))
```

```
InitialisationVector ::= OCTET STRING (SIZE(14))
    -- the internal structure is defined as follows:
    -- Octets 1 to 4 : TVP. The TVP is a 32 bit time stamp. Its value is binary coded
    --                 and indicates the number of intervals of 100 milliseconds
    --                 elapsed since 1st January 2002, 0:00:00 UTC
    -- Octets 5 to 10: NE-Id. The NE-Id uniquely identifies the sending network entity
    --                 within the PLMN. It is the entity's E.164 number without CC and
    --                 NDC. It is TBCD-coded, padded with zeros.
    -- Octets 11 to 14: PROP. This 32 bit value is used to make the
    --                 InitialisationVector unique within the same TVP period.
    --                 The content is not standardized.
```

```
OriginalComponentIdentifier ::= CHOICE {
    operationCode                       [0] OperationCode,
    errorCode                           [1] ErrorCode,
    userInfo                            [2] NULL}
```

```
OperationCode ::= CHOICE {
    localValue                          INTEGER,
    globalValue                         OBJECT IDENTIFIER}
```

```
ErrorCode ::= CHOICE {
    localValue                          INTEGER,
    globalValue                         OBJECT IDENTIFIER}
```

```
END
```

Annex B (informative): Fully expanded ASN.1 sources for abstract syntaxes of MAP

Annex B is not part of the standard, it is included for information purposes only.

For every (Value)Assignment in the root ASN.1 module all the used defined types and defined values, which are defined within the ASN.1 module or imported from ASN.1 modules, are replaced by the constructs this type or value is composed of.

The fully expanded ASN.1 root module is itself a correct and equivalent representation of the MAP-Protocol.

It allows to see at all the parameters, including all nested ones for a specific operationcode or errorcode at once.

Note that for those operations which use a result without parameters the ~~key~~word "RESULT" is not shown. Empty results are only defined in the ASN.1 description in clause 17.

.....

3GPP TSG CN WG4 Meeting #15 **N4-020841**
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **29.002** **CR** **465** ⌘ **rev** **-** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☒

Title:	⌘	Clarification on Call Deflection
Source:	⌘	CN4
Work item code:	⌘	Call Deflection
Date:	⌘	03/07/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)

Reason for change:	⌘	To clarify that the supplementary service Call Deflection is one of the Call Forwarding supplementary services. The subscription options for Call Deflection can be sent from the HLR to the VLR within the ForwardingOptions Parameter.
Summary of change:	⌘	Add “including Call Deflection” to All Call Forwarding services.
Consequences if not approved:	⌘	It is not clear which parameter is used to send the subscription options for Call Deflection from HLR to VLR.

Clauses affected:	⌘	7.6.4.1

Other specs affected: ⌘	Y	N		⌘
		X	Other core specifications	
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

7.6.4.1 SS-Code

This parameter may refer to one supplementary service or a set of supplementary services as defined in 3GPP TS 22.004. For MAP this includes:

- Calling Line Identification Presentation service (CLIP);
- Calling Line Identification Restriction service (CLIR);
- Connected Line Identification Presentation service (COLP);
- Connected Line Identification Restriction service (COLR);
- Calling Name Presentation (CNAP);
- All Call Forwarding services, including Call Deflection;
- Call Waiting (CW);
- Call Hold (HOLD);
- Multi-Party service (MPTY);
- Closed User Group (CUG);
- All Charging services;
- All Call Restriction services;
- Explicit Call Transfer service (ECT);
- enhanced Multi-Level Precedence and Pre-emption service (eMLPP);
- Completion of Calls to Busy Subscriber, originating side (CCBS-A);
- Completion of Calls to Busy Subscriber, destination side (CCBS-B);
- All LCS privacy exceptions (see clause 7.6.4.44);
- Mobile Originating Location Request (MO-LR) (see clause 7.6.4.44A);
- Multicall (MC).

3GPP TSG CN WG4 Meeting #15 ***N4-021040***
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **29.002** **CR** **470** ⌘ **rev** **1** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Correction to the usage of "Roaming not allowed" error
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	19/07/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	In CN1#24, TDoc N1-021235 (source Motorola) was approved which discussed the potential problems of sending Cause #14 to the MS: *"Due to the requirements in 03.22 (section 3.1) that "The HPLMN shall not be stored on the list of "forbidden PLMNs for GPRS service." there appears to be a potential problem if a subscriber's HPLMN uses the rejection cause #14 when the network is using NMO I."* This CR proposes to add a health warning to HLR implementors. Proposed for agreement by concensus.
Summary of change:	⌘	Added a "health" warning advising that the error Roaming not Allowed (cause 'PLMN Not Allowed' or 'Operator determined Barring') should not be sent to an SGSN in the HPLMN otherwise undesirable behaviour may occur.
Consequences if not approved:	⌘	Undesirable behaviour in the HPLMN of GPRS enabled MEs.

Clauses affected:	⌘	19.1.1.4

	Y	N		
Other specs affected: ⌘		X	Other core specifications	⌘
		X	Test specifications	
		X	O&M Specifications	

Other comments:	⌘	

****** For Information Only – from TS 29.010 ******

3.2 Routeing area updating

	04.08	09.02	Notes
Forward message	GMM (ROUTEING AREA UPDATE REQUEST)	MAP_UPDATE_GPRS LOCATION request	
	MS classmark 1	-	
	MS classmark 4	-	
	GPRS Ciphering key seq number	-	
	Mobile station identity	IMSI	
	Old routeing area identification	-	
Positive results	GMM (ROUTEING AREA UPDATE ACCEPT)	MAP_UPDATE_GPRS LOCATION response	
	Routeing area identification	-	
	Mobile station identity	-	1
C	Mobile station	-	2
C	Reject: IMSI unknown in HLR	-	3
C	Reject: MSC temporarily not reacheable	-	4
Negative results	GMM (ROUTEING AREA UPDATE REJECT)	MAP_UPDATE_GPRS LOCATION response	
	Network failure	-	5
	GPRS services not allowed in this PLMN	Unknown HLR	
	GPRS services not allowed	Unknown subscriber (no GPRS subscription)	6
	GPRS services and non GPRS services not allowed	Unknown subscriber (IMSI unknown)	7
C	GPRS services not allowed	Unknown subscriber (no GPRS subscription)	8
C	GPRS services and non-GPRS services not allowed	Unknown subscriber (IMSI unknown)	9
	MS identity cannot be derived by the network	-	10
	GPRS services not allowed in this PLMN	Roaming not allowed: PLMN not allowed	
	LA not allowed	-	
	Roaming not allowed in this LA	-	
	No Suitable cells in location area	-	11
	GPRS services not allowed in this PLMN	Operator determined barring	
	Illegal MS	-	
	Illegal ME	-	
	Network failure	System Failure	
	Network failure	Unexpected data value	
	Network failure	MAP_U/P_ABORT	
	Network failure	MAP_NOTICE	
	Network failure	MAP_CLOSE	

**** Modified section ****

19.1.1.4 Detailed procedure in the HLR

Sheet 1: The procedure Super_Charged_Cancel_Location_HLR is specific to Super-Charger; it is specified in TS 23.116 [110]. If the previous SGSN and the originating HLR support the Super-Charger functionality, processing continues from the "Yes" exit of the test "Result=Pass?".

Sheet 2: The procedure Super_Charged_Location_Updating_HLR is specific to Super-Charger; it is specified in TS 23.116 [110]. If subscription data needs to be sent to the SGSN, processing continues from the "No" exit of the test "Result=Pass?".

When addressed by the SGSN, the following macros are used by the process Update_GPRS_Location_HLR:

- Receive_Open_indication, defined in clause 25.1;
- Check_indication, defined in clause 25.2;
- Insert_Subs_Data_In_SGSN_Framed_HLR, described in clause 19.4;
- Control_Tracing_HLR_with_SGSN, described in clause 25.9;

and the processes Cancel_Location_HLR (see clause 19.1.2) and Subscriber_Present_HLR (see clause 19.1.1.7) are invoked.

The location updating process in the HLR is activated by receipt of a MAP_UPDATE_GPRS_LOCATION indication (see figure 19.1.1/19):

- if there is a parameter problem in the indication, the error Unexpected Data Value is returned in the MAP_UPDATE_LOCATION response (see Check_indication macro defined in clause 25.2); if the subscriber is not known in the HLR, the error Unknown Subscriber (with diagnostic value set to “Imsi Unknown”) is returned in the response. In either case the process terminates;
- if Network Access Mode is set to “non-GPRS only” the error Unknown Subscriber (with diagnostic value set to “Gprs Subscription Unknown”) is returned in the response. The process terminates;
- tracing shall be set to deactive in the SGSN.
- if the SGSN number received in the MAP_UPDATE_GPRS_LOCATION indication differs from the one actually stored against the subscriber, the Cancel_Location_HLR process is started to cancel the subscriber data in the stored SGSN (see clause 19.1.2).

The next action will be to check whether the subscriber is allowed to roam into the PLMN indicated by the SGSN Number given in the MAP_UPDATE_GPRS_LOCATION indication:

- if the subscriber is not allowed to roam into the PLMN, the error Roaming not Allowed with cause 'PLMN Roaming Not Allowed' or 'Operator determined Barring', depending on the case, is returned in the MAP_UPDATE_GPRS_LOCATION response, and the routing information stored (SGSN number) is deleted (deregistration). The HLR operator should avoid sending the error Roaming not Allowed with cause 'PLMN Roaming Not Allowed' or 'Operator determined Barring' to an SGSN in the HPLMN because this may lead to undesirable behaviour by the MS;
- otherwise the HLR database will be updated with information received in the indication. The HLR sets the "MS purged for GPRS" flag to False and checks whether tracing is required for that subscriber. This is handled by the macro Control_Tracing_HLR-with_SGSN described in clause 25.9.

Thereafter, the macro Insert_Subs_Data_In_SGSN_Framed_HLR described in clause 19.4 is invoked. The outcome of this macro may be:

- aborted, in which case the process terminates;
- error, in which case the error System Failure is returned in the MAP_UPDATE_GPRS_LOCATION response and the process terminates;

- OK, indicating successful outcome of downloading the subscriber data to the SGSN.

The SUBSCRIBER_PRESENT_HLR process is then started to alert the Short Message Service Centre, if required (see clause 19.1.7).

Finally the HLR number is returned in the MAP_UPDATE_GPRS_LOCATION response.

In all cases where the HLR sends a MAP_UPDATE_GPRS_LOCATION response to the SGSN, the dialogue towards the SGSN is terminated by a MAP_CLOSE request with parameter Release Method indicating Normal Release.

3GPP TSG CN WG4 Meeting #15 ***N4-021108***
Helsinki, Finland, 29th July – 2nd August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **29.002** **CR** **473** ⌘ **rev** **2** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☒

Title:	⌘	Available codecs list and selected codec indication
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	02/08/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)

Reason for change: ⌘

Currently MSC-A can not indicate to 3G_MSC-B currently used codec or available codecs that can be used after inter-MSC handover/relocation. Any method based on reverse deducing of the codec (i.e. based on received RAB parameters) in 3G_MSC-B is not seen viable, since it is possible similar RAB parameters result from different codecs. Note that it is essential for 3G_MSC-B to know the currently used codec and available codecs for it to select a codec and to be able to allocate a correct transcoder in 3G_MSC/MGW. In R99 this problem does not exist since UMTS_AMR is the only possible codec. However, in later releases other codecs are possible as well and the currently specified relocation procedure is clearly requiring a correction.

In addition, the 3G_MSC-B needs to know the set of available UMTS codecs due to potential subsequent intersystem handover to UMTS within 3G_MSC-B. MSC-A/3G_MSC-A needs to always know the selected codec for subsequent relocation and charging purposes.

Summary of change: ⌘

Consequences if not approved: ⌘ Other codecs than UMTS_AMR/UMTS_AMR2 can not be used after Inter-MSC handover/relocation.

Clauses affected: ⌘ 7.6.6, 8.4.1, 8.4.3, 8.4.4, 17.7.1

Other specs affected: ⌘

Y	N		
X		Other core specifications	⌘ 23.009 CR 072
	X	Test specifications	
	X	O&M Specifications	

Other comments: ⌘	

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

7.6.6 Radio parameters

7.6.6.1 - 7.6.6.4 Void

7.6.6.5 BSSMAP Service Handover

This parameter refers to the Service Handover information element defined in 3GPP TS 48.008

7.6.6.6 RANAP Service Handover

This parameter refers to the Service Handover information element defined in 3GPP TS 25.413.

7.6.6.7 HO-Number Not Required

This parameter indicates that no handover or relocation number allocation is necessary.

7.6.6.8 Integrity Protection Information

This parameter refers to the Integrity Protection Information element defined in 3G TS 25.413.

7.6.6.9 Encryption Information

This parameter refers to the Encryption Information element defined in 3G TS 25.413.

7.6.6.10 Radio Resource Information

This parameter refers to the Channel Type information element defined in 3GPP TS 48.008 [49].

7.6.6.10A Radio Resource List

This parameter refers to list of RAB-id's and their associated Channel Type information elements defined in 3G TS 48.008.

7.6.6.10B Chosen Radio Resource Information

This parameter refers to the Chosen Channel and Speech Version information elements defined in 3G TS 48.008.

7.6.6.11 Key Status

This parameter refers to the Key Status element defined in 3G TS 25.413.

7.6.6.12 Selected UMTS Algorithms

This parameters identifies the UMTS integrity and optionally encryption algorithms selected by MSC-B. Coding of this parameter is defined in 3G TS 25.413.

7.6.6.13 Allowed GSM Algorithms

This parameters identifies the allowed GSM algorithms in MSC-B. Coding of this parameter is defined in 3G TS 48.008.

7.6.6.14 Allowed UMTS Algorithms

This parameters identifies the allowed UMTS algorithms in MSC-B. Coding of this parameter is defined in 3G TS 25.413.

7.6.6.15 Selected GSM Algorithm

This parameter identifies the GSM algorithm selected by GSM BSC controlled by MSC-B. Coding of this parameter is defined in 3G TS 48.008.

7.6.6.16 Currently Used Codec

This parameter indicates the currently used codec in MSC-A.

7.6.6.17 Available Codecs List

This parameter indicates the available codecs in the MSC-A and the associated modes in priority order (the first entry being the highest priority codec). MSC-B uses this information to select the associated transcoder resources.

7.6.6.18 Selected Codec

This parameter indicates the codec selected by MSC-B.

****** NEXT MODIFIED SECTION ******

8.4.1 MAP_PREPARE_HANDOVER service

8.4.1.1 Definition

This service is used between MSC-A and MSC-B (E-interface) when a call is to be handed over or relocated from MSC-A to MSC-B.

The MAP_PREPARE_HANDOVER service is a confirmed service using the primitives from table 8.4/1.

8.4.1.2 Service primitives

Table 8.4/1: MAP_PREPARE_HANDOVER

Parameter name	Request	Indication	Response	Confirm
Invoke Id	M	M(=)	M(=)	M(=)
Target Cell Id	C	C(=)		
Target RNC Id	C	C(=)		
HO-NumberNotRequired	C	C(=)		
IMSI	C	C(=)		
Integrity Protection Information	C	C(=)		
Encryption Information	C	C(=)		
Radio Resource Information	C	C(=)		
AN-APDU	C	C(=)	C	C(=)
Allowed GSM Algorithms	C	C(=)		
Allowed UMTS Algorithms	C	C(=)		
Radio Resource List	C	C(=)		
RAB ID	C	C(=)		
BSSMAP Service Handover	C	C(=)		
RANAP Service Handover	C	C(=)		
Currently Used Codec	C	C(=)		
Available Codecs List	C	C(=)		
Handover Number			C	C(=)
Relocation Number List			C	C(=)
Multicall Bearer Information			C	C(=)
Multiple Bearer Requested	C	C(=)		
Multiple Bearer Not Supported			C	C(=)
Selected UMTS Algorithms			C	C(=)

Chosen Radio Resource Information			C	C(=)
Selected Codec			C	C(=)
User error			C	C(=)
Provider error				O

8.4.1.3 Parameter use

Invoke Id

For definition of this parameter see clause 7.6.1.

Target Cell Id

For definition of this parameter see clause 7.6.2. This parameter is only included if the service is not in an ongoing transaction. This parameter shall also be excluded if the service is a part of the Inter-MSC SRNS Relocation procedure or the inter-system handover GSM to UMTS procedure described in 3G TS 23.009.

Target RNC Id

For definition of this parameter see clause 7.6.2. This parameter shall be included if the service is a part of the Inter-MSC SRNS Relocation procedure or the inter-system handover GSM to UMTS procedure described in 3G TS 23.009.

HO-Number Not Required

For definition of this parameter see clause 7.6.6.

IMSI

For definition of this parameter see clause 7.6.2. This UMTS parameter shall be included if:

- available and
- if the access network protocol is BSSAP and
- there is an indication that the MS also supports UMTS.

Integrity Protection Information

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if available and if the access network protocol is BSSAP.

Encryption Information

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if available and if the access network protocol is BSSAP.

Radio Resource Information

For definition of this parameter see clause 7.6.6. This GSM parameter shall be included if the access network protocol is RANAP and there is an indication that the UE also supports GSM. If the parameter Radio Resource List is sent , the parameter Radio Resource Information shall not be sent.

AN-APDU

For definition of this parameter see clause 7.6.9.

Allowed GSM Algorithms

For definition of this parameter see clause 7.6.6. This parameters includes allowed GSM algorithms. This GSM parameter shall be included if:

- the service is a part of the Inter-MSC SRNS Relocation procedure and
- Ciphering or Security Mode Setting procedure has been performed.and

- there is an indication that the UE also supports GSM.

Allowed UMTS Algorithms

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if all of the following conditions apply:

- access network protocol is BSSAP and
- Integrity Protection Information and Encryption Information are not available and

Ciphering or Security Mode Setting procedure has been performed.

Radio Resource List

For definition of this parameter see clause 7.6.6. This parameter shall be included if the access network protocol is RANAP and there is an indication that the UE also supports GSM. This parameter shall be sent when MSC-A requests multiple bearers to MSC-B. If the parameter Radio Resource Information is sent , the parameter Radio Resource List shall not be sent.

RAB ID

For definition of this parameter see subclause 7.6.2. This parameter shall be included when MSC-A supports multiple bearers and access network protocol is BSSAP and the RAB ID has a value other than 1.

BSSMAP Service Handover

For definition of this parameter see clause 7.6.6. It shall be present if it is available.

RANAP Service Handover

For definition of this parameter see clause 7.6.6. It shall be present if it is available.

Currently Used Codec

For definition of this parameter see subclause 7.6.6. This parameter shall be included if the call is a speech call.

Available Codecs List

For definition of this parameter see subclause 7.6.6. This parameter shall be included if the call is a speech call and the Currently Used Codec, if available, is not the only available codec. This parameter shall not be available if Currently Used Codec is not available.

Handover Number

For definition of this parameter see clause 7.6.2. This parameter shall be returned at handover, unless the parameter HO-NumberNotRequired is sent. If the parameter Handover Number is returned, the parameter Relocation Number List shall not be returned.

Relocation Number List

For definition of this parameter see clause 7.6.2. This parameter shall be returned at relocation, unless the parameter HO-NumberNotRequired is sent. If the parameter Relocation Number List is returned, the parameter Handover Number shall not be returned.

Multicall Bearer Information

For a definition of this parameter see clause 7.6.2. This parameter shall be returned at relocation in the case that MSC-B supports multiple bearers.

Multiple Bearer Requested

For a definition of this parameter see clause 7.6.2. This parameter shall be sent when MSC-A requests multiple bearers to MSC-B.

Multiple Bearer Not Supported

For a definition of this parameter see clause 7.6.2. This parameter shall be returned at relocation when MSC-B receives Multiple Bearer Requested parameter and MSC-B does not support multiple bearers.

Selected UMTS Algorithms

For definition of this parameter see clause 7.6.6. This parameters includes the UMTS integrity and optionally encryption algorithms selected by RNC under the control of MSC-B. This UMTS parameter shall be included if the service is a part of the inter MSC inter system handover from GSM to UMTS.

Chosen Radio Resource Information

For definition of this parameter see clause 7.6.6. This parameter shall be returned at relocation if the encapsulated PDU is RANAP RAB Assignment Response and MS is in GSM access.

Selected Codec

For definition of this parameter see subclause 7.6.6. This parameter shall be included always if MSC-B supports the selection of codec based on Available Codecs List, even if Selected Codec is equal to the Currently Used Codec received in the service request. This parameter shall not be included if Available Codecs List was not received in the service request.

User error

For definition of this parameter see clause 7.6.1. The following errors defined in clause 7.6.1 may be used, depending on the nature of the fault:

- No handover number available.
- Target cell outside group call area;
- System failure.
- Unexpected data value.
- Data Missing.

Provider error

See definition of provider errors in clause 7.6.1.

****** NEXT MODIFIED SECTION ******

8.4.3 MAP_PROCESS_ACCESS_SIGNALLING service

8.4.3.1 Definition

This service is used between MSC-B and MSC-A (E-interface) to pass information received on the A-interface or Iu-interface in MSC-B to MSC-A.

The MAP_PROCESS_ACCESS_SIGNALLING service is a non-confirmed service using the primitives from table 8.4/3.

8.4.3.2 Service primitives

Table 8.4/3: MAP_PROCESS_ACCESS_SIGNALLING

Parameter name	Request	Indication
Invoke Id	M	M(=)

AN-APDU	M	M(=)
Selected GSM Algorithm	C	C(=)
Selected UMTS Algorithms	C	C(=)
Chosen Radio Resource Information	C	C(=)
Selected RAB id	C	C(=)
Selected Codec	C	C(=)

8.4.3.3 Parameter use

Invoke Id

For definition of this parameter see clause 7.6.1.

AN-APDU

For definition of this parameter see clause 7.6.9.

Selected GSM algorithm

For definition of this parameter see clause 7.6.6. This parameter shall be present if the encapsulated PDU is Security Mode Complete and MS is in GSM access.

Selected UMTS Algorithms

For definition of this parameter see clause 7.6.6. This parameters includes the UMTS integrity and optionally encryption algorithms selected by RNC under the control of MSC-B. This UMTS parameter shall be included if the encapsulated PDU is BSSMAP Cipher Mode Complete and the MS is in UMTS, or an interystem handover to UMTS is performed in MSC-B, or in the case of intra MSC-B intra UMTS relocation.

Chosen Radio Resource Information

For definition of this parameter see clause 7.6.6. This parameter shall be sent if the encapsulated PDU is RANAP RAB Assignment Response and MS is in GSM access.

Selected RAB ID

The selected radio access bearer that was kept at subsequent intra-MSC handover from UMTS to GSM after multiple bearers were used.

Selected Codec

For definition of this parameter see subclause 7.6.6. This parameter shall be included if MSC-B changes the selected codec or in case of intersystem handover to UMTS is performed in MSC-B. This parameter shall not be included if Available Codecs List was not received either in the Prepare Handover service request or in the Forward Access Signalling service request.

****** NEXT MODIFIED SECTION ******

8.4.4 MAP_FORWARD_ACCESS_SIGNALLING service

8.4.4.1 Definition

This service is used between MSC-A and MSC-B (E-interface) to pass information to be forwarded to the A-interface or Iu-interface of MSC-B.

The MAP_FORWARD_ACCESS_SIGNALLING service is a non-confirmed service using the primitives from table 8.4/4.

8.4.4.2 Service primitives

Table 8.4/4: MAP_FORWARD_ACCESS_SIGNALLING

Parameter name	Request	Indication
Invoke Id	M	M(=)
Integrity Protection Information	C	C(=)
Encryption Information	C	C(=)
Key Status	C	C(=)
AN-APDU	M	M(=)
Allowed GSM Algorithms	C	C(=)
Allowed UMTS Algorithms	C	C(=)
Radio Resource Information	C	C(=)
Radio Resource List	C	C(=)
BSSMAP Service Handover	C	C(=)
RANAP Service Handover	C	C(=)
Currently Used Codec	C	C(=)
Available Codecs List	C	C(=)

8.4.4.3 Parameter use

For the definition and use of all parameters and errors, see clause 7.6.1.

Invoke Id

For definition of this parameter see clause 7.6.1.

Integrity Protection Information

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if available and if the encapsulated PDU is BSSMAP Cipher Mode Command.

Encryption Information

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if available and if the encapsulated PDU is BSSMAP Cipher Mode Command.

Key Status

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if available and if the encapsulated PDU is BSSMAP Cipher Mode Command.

AN-APDU

For definition of this parameter see clause 7.6.9.

Allowed GSM Algorithms

This parameters includes allowed GSM algorithms. This GSM parameter shall be included if the encapsulated PDU is RANAP Security Mode Command and there is an indication that the UE also supports GSM.

Allowed UMTS Algorithms

For definition of this parameter see clause 7.6.6. This UMTS parameter shall be included if Integrity Protection Information and Encryption Information are not available and the encapsulated PDU is BSSMAP Cipher Mode Command.

Radio Resource Information

For definition of this parameter see clause 7.6.6. This parameter shall be sent if the encapsulated PDU is RANAP RAB Assignment Request. If the parameter Radio Resource List is sent, the parameter Radio Resource Information shall not be sent.

Radio Resource List

For definition of this parameter see clause 7.6.6. This parameter shall be sent if the encapsulated PDU is RANAP RAB Assignment Request and MSC-A requests modification of multiple bearers. If the parameter Radio Resource Information is sent, the parameter Radio Resource List shall not be sent.

BSSMAP Service Handover

For definition of this parameter see clause 7.6.6. It shall be present if it is available and the encapsulated PDU is RANAP RAB Assignment Request or BSSMAP Assignment Request.

RANAP Service Handover

For definition of this parameter see clause 7.6.6. It shall be present if it is available and the encapsulated PDU is BSSMAP Assignment Request or RANAP RAB Assignment Request.

Currently Used Codec

For definition of this parameter see subclause 7.6.6. This parameter shall be included if the encapsulated PDU is RANAP RAB Assignment Request and the bearer is modified from data to speech.

Available Codecs List

For definition of this parameter see subclause 7.6.6. This parameter shall be included if the encapsulated PDU is RANAP RAB Assignment Request, the bearer is modified from data to speech and the Currently Used Codec, if available, is not the only available codec. This parameter shall not be included if Available Codecs List was not received in the service request.

****** NEXT MODIFIED SECTION ******

17.7 MAP constants and data types

17.7.1 Mobile Service data types

….

```
-- handover types

ForwardAccessSignalling-Arg ::= [3] SEQUENCE {
    an-APDU                          AccessNetworkSignalInfo,
    integrityProtectionInfo          [0] IntegrityProtectionInformation  OPTIONAL,
    encryptionInfo                   [1] EncryptionInformation           OPTIONAL,
    keyStatus                        [2]  KeyStatus                     OPTIONAL,
    allowedGSM-Algorithms            [4]  AllowedGSM-Algorithms         OPTIONAL,
    allowedUMTS-Algorithms           [5]  AllowedUMTS-Algorithms        OPTIONAL,
    radioResourceInformation         [6] RadioResourceInformation       OPTIONAL,
    extensionContainer               [3]  ExtensionContainer            OPTIONAL,
    ...,
    radioResourceList                [7]  RadioResourceList             OPTIONAL,
    bssmap-ServiceHandover           [9]  BSSMAP-ServiceHandover        OPTIONAL,
    ranap-ServiceHandover            [8]  RANAP-ServiceHandover         OPTIONAL,
    currentlyUsedCodec               [10] Codec                         OPTIONAL,
    availableCodecsList              [11] AvailableCodecsList           OPTIONAL }
    -- availableCodecsList shall be present only if currentlyUsedCodec is present,
    -- availableCodecsList shall be discarded by the receiving node if received
    -- when currentlyUsedCodec is not present.
```

```
AllowedGSM-Algorithms ::= OCTET STRING (SIZE (1))
    -- internal structure is coded as Algorithm identifier octet from
    -- Permitted Algorithms defined in 3G TS 48.008
    -- A node shall mark all GSM algorithms that are allowed in MSC-B
```

```
AllowedUMTS-Algorithms ::= SEQUENCE {
     integrityProtectionAlgorithms       [0]  PermittedIntegrityProtectionAlgorithms
     OPTIONAL,
     encryptionAlgorithms                [1]  PermittedEncryptionAlgorithms  OPTIONAL,
     extensionContainer                  [2]  ExtensionContainer             OPTIONAL,
     ...}
```

```
PermittedIntegrityProtectionAlgorithms ::=
          OCTET STRING (SIZE (1..maxPermittedIntegrityProtectionAlgorithmsLength))
     -- Octets contain a complete PermittedIntegrityProtectionAlgorithms data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413.
     -- Padding bits are included, if needed, in the least significant bits of the
     -- last octet of the octet string.
```

```
maxPermittedIntegrityProtectionAlgorithmsLength INTEGER ::= 9
```

```
PermittedEncryptionAlgorithms ::=
          OCTET STRING (SIZE (1..maxPermittedEncryptionAlgorithmsLength))
     -- Octets contain a complete PermittedEncryptionAlgorithms data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included, if needed, in the least significant bits of the
     -- last octet of the octet string.
```

```
maxPermittedEncryptionAlgorithmsLength INTEGER ::= 9
```

```
KeyStatus ::= ENUMERATED {
     old  (0),
     new  (1),
     ...}
     -- exception handling:
     -- received values in range 2-31 shall be treated as "old"
     -- received values greater than 31 shall be treated as "new"
```

```
PrepareHO-Arg ::= [3] SEQUENCE {
     targetCellId                        [0] GlobalCellId                    OPTIONAL,
     ho-NumberNotRequired                NULL                                OPTIONAL,
     targetRNCId                         [1] RNCId                           OPTIONAL,
     an-APDU                             [2] AccessNetworkSignalInfo         OPTIONAL,
     multipleBearerRequested             [3] NULL                            OPTIONAL,
     imsi                                [4] IMSI                            OPTIONAL,
     integrityProtectionInfo             [5] IntegrityProtectionInformation  OPTIONAL,
     encryptionInfo                      [6] EncryptionInformation           OPTIONAL,
     radioResourceInformation            [7] RadioResourceInformation        OPTIONAL,
     allowedGSM-Algorithms               [9]  AllowedGSM-Algorithms          OPTIONAL,
     allowedUMTS-Algorithms              [10] AllowedUMTS-Algorithms         OPTIONAL,
     radioResourceList                   [11] RadioResourceList              OPTIONAL,
     extensionContainer                  [8] ExtensionContainer              OPTIONAL,
     ... ,
     rab-Id                              [12] RAB-Id                         OPTIONAL,
     bssmap-ServiceHandover              [13] BSSMAP-ServiceHandover         OPTIONAL,
     ranap-ServiceHandover               [14] RANAP-ServiceHandover          OPTIONAL,
     currentlyUsedCodec                  [15] Codec                          OPTIONAL,
     availableCodecsList                 [16] AvailableCodecsList            OPTIONAL}
     -- availableCodecsList shall be present only if currentlyUsedCodec is present,
     -- availableCodecsList shall be discarded by the receiving node if received
     -- when currentlyUsedCodec is not present.
```

```
BSSMAP-ServiceHandover ::= OCTET STRING (SIZE (1))
     -- Octets are coded according the Service Handover information element in
     -- 3G TS 48.008.
```

```
RANAP-ServiceHandover ::= OCTET STRING (SIZE (1))
     -- Octet contains a complete Service-Handover data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included in the least significant bits.
```

```
RadioResourceList ::= SEQUENCE SIZE (2.. maxNumOfRadioResources) OF
                                     RadioResource
```

```
RadioResource ::= SEQUENCE {
     radioResourceInformation            RadioResourceInformation,
     rab-Id                              RAB-Id,
     -- RAB Identity is needed to relate the radio resources with the radio access bearers.
     ...}
```

```
maxNumOfRadioResources  INTEGER ::= 7
```

```
PrepareHO-Res ::= [3] SEQUENCE {
     handoverNumber                      [0] ISDN-AddressString            OPTIONAL,
     relocationNumberList                [1]  RelocationNumberList         OPTIONAL,
     an-APDU                             [2]  AccessNetworkSignalInfo      OPTIONAL,
     multicallBearerInfo                 [3]  MulticallBearerInfo          OPTIONAL,
     multipleBearerNotSupported          NULL                              OPTIONAL,
     selectedUMTS-Algorithms             [5]  SelectedUMTS-Algorithms      OPTIONAL,
     chosenRadioResourceInformation      [6] ChosenRadioResourceInformation OPTIONAL,
     extensionContainer                  [4]  ExtensionContainer           OPTIONAL,
     ...,
     selectedCodec                       [7] Codec                         OPTIONAL }
```

```
SelectedUMTS-Algorithms ::= SEQUENCE {
     integrityProtectionAlgorithm        [0]  ChosenIntegrityProtectionAlgorithm   OPTIONAL,
     encryptionAlgorithm                 [1]  ChosenEncryptionAlgorithm    OPTIONAL,
     extensionContainer                  [2]  ExtensionContainer           OPTIONAL,
     ...}
```

```
ChosenIntegrityProtectionAlgorithm ::= OCTET STRING (SIZE (1))
     -- Octet contains a complete IntegrityProtectionAlgorithm data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included in the least significant bits.
```

```
ChosenEncryptionAlgorithm ::= OCTET STRING (SIZE (1))
     -- Octet contains a complete EncryptionAlgorithm data type
     -- as defined in 3G TS 25.413, encoded according to the encoding scheme
     -- mandated by 3G TS 25.413
     -- Padding bits are included in the least significant bits.
```

```
ChosenRadioResourceInformation ::= SEQUENCE {
     chosenChannelInfo                   [0] ChosenChannelInfo             OPTIONAL,
     chosenSpeechVersion                 [1] ChosenSpeechVersion           OPTIONAL,
     ...}
```

```
ChosenChannelInfo ::= OCTET STRING (SIZE (1))
     -- Octets are coded according the Chosen Channel information element in 3G TS 48.008
```

```
ChosenSpeechVersion ::= OCTET STRING (SIZE (1))
     -- Octets are coded according the Speech Version (chosen) information element in 3G TS
     -- 48.008
```

```
PrepareSubsequentHO-Arg ::= [3] SEQUENCE {
     targetCellId                        [0] GlobalCellId                  OPTIONAL,
     targetMSC-Number                    [1] ISDN-AddressString,
     targetRNCId                         [2] RNCId                         OPTIONAL,
     an-APDU                             [3]  AccessNetworkSignalInfo      OPTIONAL,
     selectedRab-Id                      [4]  RAB-Id                       OPTIONAL,
     extensionContainer                  [5]  ExtensionContainer           OPTIONAL,
     ...}
```

```
PrepareSubsequentHO-Res ::= [3] SEQUENCE {
     an-APDU                             AccessNetworkSignalInfo,
     extensionContainer                  [0]  ExtensionContainer           OPTIONAL,
     ...}
```

```
ProcessAccessSignalling-Arg ::= [3] SEQUENCE {
    an-APDU                             AccessNetworkSignalInfo,
    selectedUMTS-Algorithms             [1]  SelectedUMTS-Algorithms       OPTIONAL,
    selectedGSM-Algorithm               [2]  SelectedGSM-Algorithm         OPTIONAL,
    chosenRadioResourceInformation      [3] ChosenRadioResourceInformation OPTIONAL,
    selectedRab-Id                      [4] RAB-Id                         OPTIONAL,
    extensionContainer                  [0]  ExtensionContainer            OPTIONAL,
    ...,
    selectedCodec                       [5] Codec                          OPTIONAL }
```

```
SelectedGSM-Algorithm ::= OCTET STRING (SIZE (1))
    -- internal structure is coded as Algorithm identifier octet from Chosen Encryption
    -- Algorithm defined in 3G TS 48.008
    -- A node shall mark only the selected GSM algorithm
```

```
SendEndSignal-Arg ::= [3] SEQUENCE {
    an-APDU                             AccessNetworkSignalInfo,
    extensionContainer                  [0]  ExtensionContainer            OPTIONAL,
    ...}
```

```
SendEndSignal-Res ::= SEQUENCE {
    extensionContainer                  [0]  ExtensionContainer            OPTIONAL,
    ...}
```

```
RNCId ::= OCTET STRING (SIZE (7))
    -- The internal structure is defined as follows:
    -- octet 1 bits 4321                Mobile Country Code 1st digit
    --         bits 8765                Mobile Country Code 2nd digit
    -- octet 2 bits 4321                Mobile Country Code 3rd digit
    --         bits 8765                Mobile Network Code 3rd digit
    --                                  or filler (1111) for 2 digit MNCs
    -- octet 3 bits 4321                Mobile Network Code 1st digit
    --         bits 8765                Mobile Network Code 2nd digit
    -- octets 4 and 5                   Location Area Code according to 3G TS 24.008
    -- octets 6 and 7                   RNC Id value according to 3G TS 25.413
```

```
RelocationNumberList ::= SEQUENCE SIZE (1..maxNumOfRelocationNumber) OF
                                        RelocationNumber
```

```
MulticallBearerInfo ::= INTEGER (1..maxNumOfRelocationNumber)
```

```
RelocationNumber ::= SEQUENCE {
    handoverNumber                      ISDN-AddressString,
    rab-Id                              RAB-Id,
    -- RAB Identity is needed to relate the calls with the radio access bearers.
    ...}
```

```
RAB-Id ::= INTEGER (1..maxNrOfRABs)
```

```
maxNrOfRABs INTEGER ::= 255
```

```
maxNumOfRelocationNumber  INTEGER ::= 7
```

```
RadioResourceInformation ::= OCTET STRING (SIZE (3..13))
    -- Octets are coded according the Channel Type information element in 3G TS 48.008
```

```
IntegrityProtectionInformation ::= OCTET STRING (SIZE (18..maxNumOfIntegrityInfo))
    -- Octets contain a complete IntegrityProtectionInformation data type
    -- as defined in 3G TS 25.413, encoded according to the encoding scheme
    -- mandated by 3G TS 25.413
    -- Padding bits are included, if needed, in the least significant bits of the
    -- last octet of the octet string.
```

```
maxNumOfIntegrityInfo INTEGER ::= 100
```

```
EncryptionInformation ::= OCTET STRING (SIZE (18..maxNumOfEncryptionInfo))
    -- Octets contain a complete EncryptionInformation data type
    -- as defined in 3G TS 25.413, encoded according to the encoding scheme
    -- mandated by 3G TS 25.413
    -- Padding bits are included, if needed, in the least significant bits of the
    -- last octet of the octet string.
```

```
maxNumOfEncryptionInfo INTEGER ::= 100
```

```
AvailableCodecsList ::= SEQUENCE {
    utranCodecList              [0] CodecList              OPTIONAL,
    geranCodecList              [1] CodecList              OPTIONAL,
    extensionContainer          [2] ExtensionContainer     OPTIONAL,
    ...}
```

```
CodecList ::= SEQUENCE {
    codec1                      [1] Codec,
    codec2                      [2] Codec                  OPTIONAL,
    codec3                      [3] Codec                  OPTIONAL,
    codec4                      [4] Codec                  OPTIONAL,
    codec5                      [5] Codec                  OPTIONAL,
    codec6                      [6] Codec                  OPTIONAL,
    codec7                      [7] Codec                  OPTIONAL,
    codec8                      [8] Codec                  OPTIONAL,
    extensionContainer          [9] ExtensionContainer     OPTIONAL,
    ...}
    -- Codecs are sent in priority order where codec1 has highest priority
```

```
Codec ::= OCTET STRING (SIZE (1..4))

    -- The internal structure is defined as follows:
    -- octet 1                  Coded as Codec Identification code in 3GPP TS 26.103
    -- octets 2,3,4             Parameters for the Codec as defined in 3GPP TS
    --                          26.103, if available, length depending on the codec
```

3GPP TSG-CN WG4 Meeting #15 ***Tdoc ⌘N4-021011***
Helsinki, Finland, 29th July – 2nd August 2002 **(revision of *N4-020986*)**

CR-Form-v7

CHANGE REQUEST

⌘ **29.060** **CR** **325** ⌘ **rev** **2** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☒

Title:	⌘	RAB Setup Information for IPv6
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	30/07/2002
Category:	⌘	**F**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	The Liaison Statement on Support of IPv6 on Iu (N4-020567) from RAN3, i.e. to support data forwarding during handovers between an IPv4-only RNC and an IPv6-capable RNC, the IPv6-capable RNC should be an IPv4/IPv6 dual stack RNC. At the SRNS Relocation procedure, the target RNC will not know the IP version capabilities of the source RNC. Therefore, a dual stack target RNC should supply both an IPv4 address and an IPv6 address to the source RNC, to be used for packet forwarding.
Summary of change:	⌘	Addition of the Additional RAB Setup Information IE in which the TEID and RNC IP address for IPv6 for a given NSAPI is sent.
Consequences if not approved:	⌘	Data forwarding between R99 (or Rel-4) RNC and Rel-5 RNC using IPv6 may not work.

Clauses affected:	⌘	7.5.7, 7.7, 7.7.45A (new chapter)

Other specs affected: ⌘

Y	N	
	X	Other core specifications
	X	Test specifications
	X	O&M Specifications

⌘

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

****** START OF MODIFICATION ******

7.5.7 Forward Relocation Response

The new SGSN shall send a Forward Relocation Response to the old SGSN as a response to a previous Forward Relocation Request.

Possible Cause values is:

- 'Request Accepted'.
- 'System failure'.
- 'Mandatory IE incorrect'.
- 'Mandatory IE missing'.
- 'Optional IE incorrect'.
- 'No resources available'.
- 'Invalid message format'.
- 'Relocation failure'.

RANAP Cause is mandatory if cause value is contained in RANAP message.

RAB Setup Information, UTRAN transparent container and RANAP Cause are information from the target RNC in the new SGSN.

One or more RAB Setup Information parameters shall be set in this message. This information element shall be included if the Cause contains the value 'Request accepted'.

The new SGSN shall include a SGSN Address for control plane. The old SGSN shall store this SGSN Address and use it when sending control plane messages for the MS to the new SGSN in the SRNS Relocation Procedure.

The Tunnel Endpoint Identifier Control Plane field specifies a Tunnel Endpoint Identifier that is chosen by the new SGSN. The old SGSN shall include this Tunnel Endpoint Identifier in the GTP header of all subsequent signalling messages that are sent from the old SGSN to the new SGSN. This information element shall be included if the Cause contains the value 'Request accepted'.

One or more Additional RAB Setup Information parameters may be sent in this message for IPv6. This information element shall be included if the Cause contains the value 'Request accepted' and there is at least one RAB assigned in the new SGSN.

The optional Private Extension contains vendor or operator specific information.

Table 30: Information Elements in a Forward Relocation

Information element	Presence requirement	Reference
Cause	Mandatory	7.7.1
Tunnel Endpoint Identifier Control Plane	Conditional	7.7.14
RANAP Cause	Conditional	7.7.18
SGSN Address for Control plane	Conditional	7.7.32
UTRAN transparent container	Optional	7.7.38
RAB Setup Information	Conditional	7.7.39
Additional RAB Setup Information	Conditional	7.7.45A
Private Extension	Optional	7.7.46

****** END OF MODIFICATION ******

****** START OF MODIFICATION ******

7.7 Information Elements

A GTP Signalling message may contain several information elements. The TLV (Type, Length, Value) or TV (Type, Value) encoding format shall be used for the GTP information elements. The information elements shall be sorted, with the Type fields in ascending order, in the signalling messages. The Length field contains the length of the information element excluding the Type and Length field.

For all the length fields, bit 8 of the lowest numbered octet is the most significant bit and bit 1 of the highest numbered octet is the least significant bit.

Within information elements, certain fields may be described as spare. These bits shall be transmitted with the value defined for them. To allow for future features, the receiver shall not evaluate these bits.

The most significant bit in the Type field is set to 0 when the TV format is used and set to 1 for the TLV format.

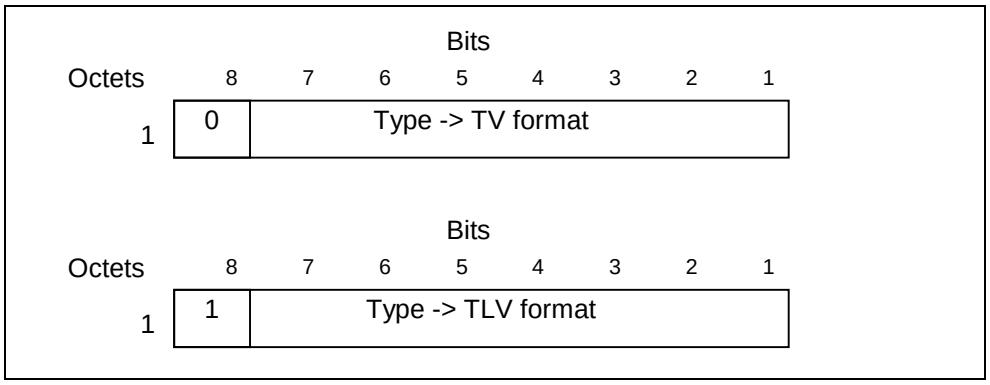


Figure 8: Type field for TV and TLV format

Table 37: Information Elements

IE Type Value	Format	Information Element	Reference
1	TV	Cause	7.7.1
2	"	International Mobile Subscriber Identity (IMSI)	7.7.2
3	"	Routeing Area Identity (RAI)	7.7.3
4	"	Temporary Logical Link Identity (TLLI)	7.7.4
5	"	Packet TMSI (P-TMSI)	7.7.5
6-7	Spare		
8	"	Reordering Required	7.7.6
9	"	Authentication Triplet	7.7.7
10	Spare		
11	"	MAP Cause	7.7.8
12	"	P-TMSI Signature	7.7.9
13	"	MS Validated	7.7.10
14	"	Recovery	7.7.11
15	"	Selection Mode	7.7.12
16	"	Tunnel Endpoint Identifier Data I	7.7.13
17	"	Tunnel Endpoint Identifier Control Plane	7.7.14
18	"	Tunnel Endpoint Identifier Data II	7.7.15
19	"	Teardown Ind	7.7.16
20	"	NSAPI	7.7.17
21	"	RANAP Cause	7.7.18
22	"	RAB Context	7.7.19
23	"	Radio Priority SMS	7.7.20
24	"	Radio Priority	7.7.21
25	"	Packet Flow Id	7.7.22
26	"	Charging Characteristics	7.7.23
27	"	Trace Reference	7.7.24
28	"	Trace Type	7.7.25
29	"	MS Not Reachable Reason	7.7.25A
30	"	Radio Priority LCS	7.7.25B
117-126	Reserved for the GPRS charging protocol (see GTP' in GSM 12.15)		
127	"	Charging ID	7.7.26
128	TLV	End User Address	7.7.27
129	"	MM Context	7.7.28
130	"	PDP Context	7.7.29
131	"	Access Point Name	7.7.30
132	"	Protocol Configuration Options	7.7.31
133	"	GSN Address	7.7.32
134	"	MS International PSTN/ISDN Number (MSISDN)	7.7.33
135	"	Quality of Service Profile	7.7.34
136	"	Authentication Quintuplet	7.7.35
137	"	Traffic Flow Template	7.7.36
138	"	Target Identification	7.7.37
139	"	UTRAN Transparent Container	7.7.38
140	"	RAB Setup Information	7.7.39
141	"	Extension Header Type List	7.7.40
142	"	Trigger Id	7.7.41
143	"	OMC Identity	7.7.42
144	"	RAN Transparent Container	7.7.43
145	"	PDP Context Prioritization	7.7.45
146	"	Additional RAB Setup Information	7.7.45A
239-250	Reserved for the GPRS charging protocol (see GTP' in GSM 12.15)		
251	"	Charging Gateway Address	7.7.44
252-254	Reserved for the GPRS charging protocol (see GTP' in GSM 12.15)		
255	"	Private Extension	7.7.46

****** END OF MODIFICATION ******

****** START OF MODIFICATION ******

7.7.45A Additional RAB Setup Information

If the target RNC successfully allocated resources associated with the NSAPI, the Additional RAB Setup Information IE contains the RNC Tunnel Endpoint Identifier and RNC IP address for data forwarding from source RNC to target RNC for IPv6. If the target RNC or the new SGSN failed to allocate resources the Additional RAB Setup Information IE contains only Length and NSAPI indicating that the source RNC shall release the resources associated with the NSAPI.

The spare bits x indicate unused bits, which shall be set to 0 by the sending side and which shall not be evaluated by the receiving side.

The format of the RNC IPv6 address is the same as the GSN address as defined in 3GPP TS 23.003. The Address Type and Address Length fields from 3GPP TS 23.003 are not included in the RNC IP Address field.

Octets	Bits 8	7	6	5	4	3	2	1
1	Type = 146 (Decimal)							
2-3	Length							
4	x	x	x	x	NSAPI			
5-8	Tunnel Endpoint Identifier Data							
9-n	RNC IP Address							

Figure X: Additional RAB Setup Information IE for data forwarding

Octets	Bits 8	7	6	5	4	3	2	1
1	Type = 146 (Decimal)							
2-3	Length=1							
4	x	x	x	x	NSAPI			

Figure Y: Additional RAB Setup Information IE for release of resources

****** END OF MODIFICATION ******

3GPP TSG CN WG4 Meeting #15 ***N4-021110***
Helsinki, Finland, 29th July – 2nd August 2002 **(revision of N4-020877)**

CR-Form-v7

CHANGE REQUEST

⌘ **29.060** **CR** **329** ⌘ **rev** **1** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [X] Radio Access Network [ ] Core Network [X]

Title:	⌘	Addition of PCO IE to Update PDP context procedures
Source:	⌘	CN4
Work item code:	⌘	TEI5
Date:	⌘	01/08/2002
Category:	⌘	**F**
Release:	⌘	REL-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	During CN1#24 meeting it was agreed to add PCO IE to other PDP context modification procedure (see the agreed CR to 24.008 in N1-021475). The reason for this is to avoid backward compatibility problems in future releases.
Summary of change:	⌘	PCO IE added to 7.3.3 Update PDP Context Request and 7.3.4 Update PDP Context Response
Consequences if not approved:	⌘	In Rel-6 CN groups will face the same backwards compatibility problems, as with TFT and PCO, and a non-optimal solution will have to be adopted again

Clauses affected:	⌘	7.3.3, 7.3.4

		Y	N			
Other specs affected:	⌘	X		Other core specifications	⌘	24.008 (CR 634r1)
			X	Test specifications		
			X	O&M Specifications		

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

7.3.3 Update PDP Context Request

An Update PDP Context Request message shall be sent from a SGSN to a GGSN as part of the GPRS Inter SGSN Routeing Update procedure or the PDP Context Modification procedure or to redistribute contexts due to load sharing. It shall be used to change the QoS and the path. In addition it shall be used if it is necessary to change the GTP version of a tunnel to a GGSN from GTP v0 to GTP v1. The message shall be sent by the new SGSN at the Inter SGSN Routeing Update procedure.

The NSAPI information element together with the Tunnel Endpoint Identifier in the GTP header unambiguously identifies a PDP Context in the GGSN.

The IMSI shall be included if the message is sent during an Inter SGSN change when changing the GTP version from GTP v0 to GTP v1; this is required, as the TEID in the header of the message is set to all zeros in this case.

The Tunnel Endpoint Identifier Data field specifies a downlink Tunnel Endpoint Identifier for G-PDUs which is chosen by the SGSN. The GGSN shall include this Tunnel Endpoint Identifier in the GTP header of all subsequent downlink G-PDUs that are related to the requested PDP context.

The Tunnel Endpoint Identifier Control Plane field specifies a downlink Tunnel Endpoint Identifier Control Plane messages which is chosen by the SGSN. The GGSN shall include this Tunnel Endpoint Identifier in the GTP header of all subsequent downlink control plane messages that are related to the requested PDP context. If the SGSN has already confirmed successful assignment of its Tunnel Endpoint Identifier Control Plane to the peer GGSN, this field shall not be present. The SGSN confirms successful assignment of its Tunnel Endpoint Identifier Control Plane to the GGSN when it receives any message with its assigned Tunnel Endpoint Identifier Control Plane in the GTP header from the GGSN.

The Quality of Service Profile information element shall include the QoS negotiated between the MS and SGSN at PDP Context activation or the new QoS negotiated in the PDP Context Modification procedure.

The SGSN shall include an SGSN Address for control plane and an SGSN address for user traffic, which may differ from that provided by the underlying network service (e.g. IP).

If an IPv4/IPv6 capable SGSN received IPv4 GGSN addresses from the old SGSN, it shall include IPv4 addresses in the fields SGSN Address for Control Plane and SGSN Address for User Traffic and IPv6 addresses in the fields Alternative SGSN Address for Control Plane and Alternative SGSN Address for User Traffic. Otherwise, an IPv4/IPv6 capable SGSN shall use only SGSN IPv6 addresses if it has GGSN IPv6 addresses available. If the GGSN supports IPv6 below GTP, it shall store and use the IPv6 SGSN addresses for communication with the SGSN and ignore the IPv4 SGSN addresses. If the GGSN supports only IPv4 below GTP, it shall store and use the IPv4 SGSN addresses for communication with the SGSN and ignore the IPv6 SGSN addresses. When active contexts are being redistributed due to load sharing, G-PDUs that are in transit across the Gn-interface are in an undetermined state and may be lost.

The SGSN shall include a Recovery information element into the Update PDP Context Request if the SGSN is in contact with the GGSN for the very first time or if the SGSN has restarted recently and the new Restart Counter value has not yet been indicated to the GGSN. The GGSN that receives a Recovery information element in the Update PDP Context Request message element shall handle it in the same way as when receiving an Echo Response message. The Update PDP Context Request message shall be considered as a valid update request for the PDP context indicated in the message.

The Traffic Flow Template (TFT) is used to distinguish between different user traffic flows.

The SGSN shall include Trace Reference, Trace Type, Trigger Id, and OMC Identity in the message if GGSN trace is activated while the PDP context is active. The SGSN shall copy Trace Reference, Trace Type, and OMC Identity from the trace request received from the HLR or OMC.

The optional Private Extension contains vendor or operator specific information.

The MS includes the Protocol Configuration Options (PCO) information element ~~may be included~~ in the request ~~when~~if the MS wishes to provide~~s~~ the GGSN with application specific parameters. The SGSN includes this IE in the Update PDP Context Request if the associated Modify PDP Context Request from the MS includes protocol configuration options. The SGSN shall copy the content of this IE transparently from the content of the PCO IE in the Modify PDP Context Request message.

Table 7: Information Elements in an SGSN-Initiated Update PDP Context Request

Information element	Presence requirement	Reference
IMSI	Conditional	7.7.2
Recovery	Optional	7.7.11
Tunnel Endpoint Identifier Data I	Mandatory	7.7.13
Tunnel Endpoint Identifier Control Plane	Conditional	7.7.14
NSAPI	Mandatory	7.7.17
Trace Reference	Optional	7.7.24
Trace Type	Optional	7.7.25
Protocol Configuration Options	Optional	7.7.31
SGSN Address for Control Plane	Mandatory	GSN Address 7.7.32
SGSN Address for User Traffic	Mandatory	GSN Address 7.7.32
Alternative SGSN Address for Control Plane	Conditional	GSN Address 7.7.32
Alternative SGSN Address for User Traffic	Conditional	GSN Address 7.7.32
Quality of Service Profile	Mandatory	7.7.34
TFT	Optional	7.7.36
Trigger Id	Optional	7.7.41
OMC Identity	Optional	7.7.42
Private Extension	Optional	7.7.46

An Update PDP Context Request may also be sent from a GGSN to a SGSN to re-negotiate the QoS of a PDP context. The GGSN-initiated Update PDP Context Request can also be used to provide a PDP address to the SGSN (and MS). The latter shall be used by GGSN when it acts as a DHCP Relay Agent or Mobil IP Foreign Agent. A GGSN may send an update PDP context to a SGSN to check that the PDP context is still active at the SGSN. In such a case, the GGSN shall include the optional IMSI IE, to add robustness against the case the SGSN has re-assigned the TEID to another PDP context (this may happen when the PDP context is dangling at the GGSN). Also, the "Quality of service profile" IE and the "End user Address" IE shall not be included in this case.

The Quality of Service Profile information element shall include the GGSN requested QoS.

The End User Address information element shall contain a valid IPv4 or IPv6 address.

The GGSN shall include a Recovery information element into the Update PDP Context Request if the GGSN has restarted recently and the new Restart Counter value has not yet been indicated to the SGSN. The SGSN that receives a Recovery information element in the Update PDP Context Request message element shall handle it in the same way as when receiving an Echo Response message. The Update PDP Context Request message shall be considered as a valid update request for the PDP context indicated in the message.

The NSAPI information element together with the Tunnel Endpoint Identifier in the GTP header unambiguously identifies a PDP Context in the SGSN.

The optional Private Extension contains vendor or operator specific information.

Table 8: Information Elements in a GGSN-Initiated Update PDP Context

Information element	Presence requirement	Reference
IMSI	optional	7.7.2
Recovery	Optional	7.7.11
NSAPI	Mandatory	7.7.17
End User Address	Optional	7.7.27
Quality of Service Profile	Optional	7.7.34
Private Extension	Optional	7.7.46

7.3.4 Update PDP Context Response

The message shall be sent from a GGSN node to a SGSN node as a response of an Update PDP Context Request.

If the SGSN receives an Update PDP Context Response with a Cause value other than 'Request accepted', it shall abort the update of the PDP context.

Only the Cause information element and optionally the Recovery information element shall be included in the response if the Cause contains another value than 'Request accepted'.

Possible Cause values are:

- 'Request Accepted'.
- 'Non-existent'.
- 'Service not supported'.
- 'System failure'.
- 'Semantic error in the TFT operation'.
- 'Syntactic error in the TFT operation'.
- 'Semantic errors in packet filter(s)'.
- 'Syntactic errors in packet filters(s)'.
- 'Mandatory IE incorrect'.
- 'Mandatory IE missing'.
- 'Optional IE incorrect'.
- 'Invalid message format'.

The Tunnel Endpoint Identifier Data field specifies an uplink Tunnel Endpoint Identifier for G-PDUs that is chosen by the GGSN. The SGSN shall include this Tunnel Endpoint Identifier in the GTP header of all subsequent uplink G-PDUs that are related to the requested PDP context. This information element shall be included if the Cause contains the value 'Request accepted'.

The Tunnel Endpoint Identifier Control Plane field specifies an uplink Tunnel Endpoint Identifier Control Plane messages which is chosen by the GGSN. The SGSN shall include this Tunnel Endpoint Identifier in the GTP header of all subsequent uplink control plane messages which are related to the requested PDP context. If the GGSN has already confirmed successful assignment of its Tunnel Endpoint Identifier Control Plane to the peer SGSN, this field shall not be present. The GGSN confirms successful assignment of its Tunnel Endpoint Identifier Control Plane to the SGSN when it receives any message with its assigned Tunnel Endpoint Identifier Control Plane in the GTP header from the SGSN.

The QoS values supplied in the Update PDP Context Request may be negotiated downwards by the GGSN. The negotiated values or the original value from SGSN is inserted in the Quality of Service Profile information element. This information element shall be included if the Cause contains the value 'Request accepted'.

The GGSN may start to forward T-PDUs after the Update PDP Context Response has been sent. The SGSN may start to forward T-PDUs when the Update PDP Context Response has been received. In this case the SGSN shall also be prepared to receive T-PDUs from the GGSN after it has sent an Update PDP Context Request but before an Update PDP Context Response has been received.

The GGSN shall include a GGSN address for user traffic, which may differ from that provided by the underlying network service (e.g. IP). IPv4/IPv6 capable GGSN shall include both its IP version addresses. If the Update PDP Context Request received from the SGSN included IPv6 SGSN addresses, an IPv4/IPv6 capable GGSN shall include an IPv6 address in the field GGSN Address for User Traffic and a corresponding IPv4 address in the field Alternative GGSN Address for User Traffic. If SGSN included only an IPv4 SGSN address in the request, IPv4/IPv6 capable GGSN shall include IPv4 address for user traffic in the field GGSN Address for User Traffic and IPv6 address in the field Alternative GGSN Address for User Traffic. The SGSN shall store the GGSN Addresses and use one of them when sending G-PDUs to the GGSN for the MS. When active contexts are being redistributed due to load sharing, G-PDUs that are in transit across the Gn-interface are in an undetermined state and may be lost.

The GGSN shall also include a GGSN address for control plane, which shall not differ from that provided at PDP context setup time and shall remain unchanged for the lifetime of the PDP context. If the Update PDP Context Request received from the SGSN included IPv6 SGSN addresses, an IPv4/IPv6 capable GGSN shall include an IPv6 address in the field GGSN Address for Control Plane and a corresponding IPv4 address in the field Alternative GGSN Address for Control Plane. If SGSN included only an IPv4 SGSN address in the request, IPv4/IPv6 capable GGSN shall include

IPv4 address for Control plane in the field GGSN Address for Control Plane and IPv6 address for Control plane in the field Alternative GGSN Address for Control Plane.

The GGSN Address for control plane and the GGSN Address for user traffic shall be included if the Cause contains the value 'Request accepted'. The Alternative GGSN Addresses shall be included if the GGSN supports IPv6 below GTP and the Cause contains the value 'Request accepted'.

The GGSN shall include the Recovery information element into the Update PDP Context Response if the GGSN is in contact with the SGSN for the first time or if the GGSN has restarted recently and the new Restart Counter value has not yet been indicated to the SGSN. The SGSN receiving the Recovery information element shall handle it as when an Echo Response message is received but shall consider the PDP context as updated and active if the response cause indicates a successful operation at the GGSN.

The Charging ID is used to identify all charging records produced in SGSN(s) and the GGSN for this PDP context. The Charging ID has been previously generated by the GGSN and is unique for this PDP context. If an inter-SGSN routing area update occurs, it is transferred to the new SGSN as part of each active PDP context. This information element shall be included if the Cause contains the value 'Request accepted'.

The Charging Gateway Address is the IP address of the recommended Charging Gateway Functionality to which the SGSN should transfer the Charging Detail Records (CDR) for this PDP Context.

The optional Private Extension contains vendor or operator specific information.

The GGSN includes the Protocol Configuration Options (PCO) information element ~~may be included~~ in the response ~~when~~if the GGSN wishes to~~-~~ provide~~s~~ the MS with application specific parameters.

Table 9: Information Elements in an Update PDP Context Response sent by a GGSN

Information element	Presence requirement	Reference
Cause	Mandatory	7.7.1
Recovery	Optional	7.7.11
Tunnel Endpoint Identifier Data I	Conditional	7.7.13
Tunnel Endpoint Identifier Control Plane	Conditional	7.7.14
Charging ID	Conditional	7.7.26
Protocol Configuration Options	Optional	7.7.31
GGSN Address for Control Plane	Conditional	GSN Address 7.7.32
GGSN Address for User Traffic	Conditional	GSN Address 7.7.32
Alternative GGSN Address for Control Plane	Conditional	GSN Address 7.7.32
Alternative GGSN Address for User Traffic	Conditional	GSN Address 7.7.32
Quality of Service Profile	Conditional	7.7.34
Charging Gateway Address	Optional	7.7.44
Private Extension	Optional	7.7.46

The message can also be sent from a SGSN node to a GGSN node as a response of a GGSN-initiated Update PDP Context Request.

If the GGSN receives an Update PDP Context Response with a Cause value other than 'Request accepted', it shall abort the update of the PDP context if the associated Update PDP Context Request was sent only to re-negotiate the QoS of a PDP context. Furthermore if the associated Update PDP Context Request included an 'End User Address' information element the GGSN shall delete the PDP context using the Delete PDP Context procedure and may notify the Operation and Maintenance network element.

Only the Cause information element and optionally the Recovery information element shall be included in the response if the Cause contains another value than 'Request accepted'.

Possible Cause values are the same as for the Update PDP Context Response sent by a GGSN. When the optional IMSI IE value differs from the IMSI IE value associated to the PDP context, the SGSN shall respond using the cause value 'Non-existent'.

The QoS values supplied in the Update PDP Context Request may be negotiated downwards by the SGSN. The negotiated values or the original value from GGSN is inserted in the Quality of Service Profile information element.

This information element shall be included if the Cause contains the value 'Request accepted' and a QoS information element was supplied in the corresponding request message.

The SGSN shall include the Recovery information element into the Update PDP Context Response if the SGSN has restarted recently and the new Restart Counter value has not yet been indicated to the GGSN. The GGSN receiving the Recovery information element shall handle it as when an Echo Response message is received but shall consider the PDP context as updated and active if the response cause indicates a successful operation at the SGSN.

Table 10: Information Elements in an Update PDP Context Response sent by a

Information element	Presence requirement	Reference
Cause	Mandatory	7.7.1
Recovery	Optional	7.7.11
Quality of Service Profile	Conditional	7.7.34
Private Extension	Optional	7.7.46