

Source: CN2
Title: CR on Rel-5 WI CAMEL 4, Composite changes for CAMEL phase 4, stage 3
Agenda item: 8.3
Document for: APPROVAL

This document contains CR on Release-5 WI CAMEL 4 that is submitted to CN#16 Plenary for approval.

Spec	CR	Rev	Doc-2nd-Level	Phase	Subject	Cat	Ver_C
29.078	244	1	N2-020627	Rel-5	Composite changes for CAMEL phase 4, stage 3 (TS 29.078 for Rel-5)	B	4.4.0

N2-020627 document contains CR cover page and the following attachments:

- content of the CR in the 29078-500.zip which contains 29078-500.doc and extracted ASN.1 modules in separate text files
- 29078-500 Change History.doc that lists all the individual CRs that are approved in CN2 for TS 29.078, for the WI CAMEL4

Annex D (informative): Editorial comments

NOTE This Annex is for information only. It is to be used during the development. It shall be removed before this draft TS is presented to CN for approval.

Version	Comments
X.5.0	<p>The following CRs have been included:</p> <ul style="list-style-type: none"> - N2-000424 (CAMEL control of Optimal Routing) - N2-000426 (Introduction of Alerting DP) - N2-000540 (Introduction of CAMEL control of MT-SMS in 29.078) - N2-000543 (Enhancement to Play Announcement) - N2-010107 (FTN in forwarding notification to SCP) - N2-010109 (Corrections to CAMEL control of MT-SMS) - N2-010231 (Introduction of flexible tones in ApplyCharing Information Flow) - N2-010228 (Provide Location information in case a terminating call is alerted) - N2-010192 (trigger criteria for mid-call event) - N2-010194 (Mobile Terminated SMS) - N2-010141 (Triggering criteria for MT-SMS) <p>The following improvements have been made to the specification:</p> <ul style="list-style-type: none"> - references corrected; references numbers for 22.078 and 23.079 must still be added. - editorial & stylistic corrections and improvements - clause 11 has been split up into 3 separate chapters
d5.2.0	<ul style="list-style-type: none"> - The following CRs have been included:all CRs approved at CN Plenary meeting #12 - N2-010317 (Introduction of CPH operations (ASN.1))
d.5.3.0	<p>The following CRs have been included:</p> <ul style="list-style-type: none"> - N2-010606 (Introduction of CPH parameters in existing CAP operations) - N2-010607 (Introduction of DisconnectForwardConnectionWithArgument procedure) - N2-010608 (Introduction of DisconnectLeg procedure) - N2-010609 (Introduction of EntityReleased procedure) - N2-010610 (Introduction of InitiateCallAttempt procedure) - N2-010622 (Introduction of MoveLeg procedure) - N2-010623 (Introduction of SplitLeg procedure) - N2-010626 (Definitions of RNC & ENC operations and their parameters)
d.5.4.0	<p>The following CAMEL Phase 4 CRs have been included:</p> <ul style="list-style-type: none"> - N2-010638 (Ability to arm Mid_Call DP for the duration of a call) - N2-010692 (Providing the location information during ongoing call) - N2-010686 (Introduction of Reference Number for SMS) <p>The following CAMEL Phase 3 CRs have been included:</p> <ul style="list-style-type: none"> - N2-010587 (Corrections to ASN.1 syntax) - N2-010602 (Using gsmSCF address from GPRS-CSI for re-establishing TC dialogues)
d.5.5.0	<p>The following CAMEL Phase 4 CRs have been included:</p> <ul style="list-style-type: none"> - N2-010839 (Enhancement of the SendChargingInformation procedure for Call Party Handling) - N2-010840 (SrfConnection and tariff switch timers for Apply Charging and Apply Charging Report) <p>The following CAMEL Phase 3 CRs have been included:</p> <ul style="list-style-type: none"> - N2-010759 (Encoding of the InitialDPGPRS ChargingID parameter) - N2-010761 (Correction of the MAXIMUM-FOR-FCI-BILLING-CHARGING value) - N2-010813 (Correction of the MAXIMUM-FOR-SCI-BILLING-CHARGING value) - N2-010844 (Precision about default values for ServiceInteractionIndicatorsTwo parameters)
D5.5.1	Editorial corrections.
D5.6.0	<p>The following approved CAMEL Phase 3 CRs have been included (without change bars):</p> <ul style="list-style-type: none"> - N2-010892 (ApplyCharging shall be allowed in a control relationship only) - N2-010893 (Correction to IMPORT statements) - N2-010894 (Correction to reference for the encoding of Called Party Number) - N2-010982 (Correction to error handling description for Initial DP operations) - N2-010984 (The use of "White TCAP" shall be mandated for CAP) - N2-010994 (Correction to preconditions for ActivityTestGPRS) - N2-010995 (Correction to references for the encoding of APN) - N2-011011 (Correction to precondition of ContinueWithArgument) - N2-010970 (Correction to GPRS parameters encoding) <p>The following approved CAMEL Phase 4 CRs have been included (with change bars):</p> <ul style="list-style-type: none"> - N2-010873 (Introduction of Initiate Call Attempt ack) - N2-010964 (Route not permitted IE in ERB in the case of MF); tagging has been updated. - N2-011001 (Move of definition of GPRSChargingID to 29.002) <p>The following editorial corrections have been made:</p> <ul style="list-style-type: none"> - consistent spelling of gsmSSF state names (no bars). - Adding oChangeOfPosition and tChangeOfPosition to EventTypeBCSM, tags 50 and 51 (with change bars). - Adding oChangeOfPosition and tChangeOfPosition to tables 11-1 and 11-2 (with change bars).
D5.6.1	<p>Editorial changes (with change bars):</p> <ul style="list-style-type: none"> - consistent event naming; the DP names from 23.078 (table 4-2 and table 4-3) are used for the BCSM events, except for cases where there is no 1:1 relationship between and event and a DP, e.g. "answer", which can relate to DP "O_Answer" and to DP "T_Answer". - Unused references in chapter 2 have been removed; reference numbers have been corrected. - Consistent usage of terms "O-BCSM" and "T-BCSM". - Consistent usage of the terms "gsm_SSME_FSM" and "gprs_SSME_FSM". - Correction to tagging of parameters within eventSpecificInformation.
D5.7.0	<ul style="list-style-type: none"> - The following CRs have been included: N2-020014, N2-020061. - Rapporteur's comments have been added.

D5.7.1	<ul style="list-style-type: none"> - Approved CRs from the editor's meeting in Rijen, Netherlands, have been incorporated - Agreed principles have been applied (refer to report from the editor's meeting) - CR editor's notes have been incorporated to highlight outstanding issues
D5.7.2	<ul style="list-style-type: none"> - All change bars accepted - The following R99 CRs are included: N2-020181, N2-020190, N2-020214 and N2-020218. - CR Editor's notes are retained - ISUP messages shall be referred to as "ISUP IAM", "ISUP ANM", "ISUP CON" etc.
D5.7.3	<ul style="list-style-type: none"> - All change bars accepted - The following approved CRs from CN2#23 (Helsinki) are included: N2-020317, N2-020321, N2-020417, N2-020437, N2-020443, N2-020444, N2-020455 - Editorial corrections
D5.7.4	<ul style="list-style-type: none"> - Rapporteur's notes added
5.0.0	<p>Rel-5 version of TS 29.078 approved by CN Plenary</p> <p>The following CRs are included:</p> <p>N2-020622; N2-020503; N2-020588; N2-020589; N2-020591; N2-020592; N2-020593; N2-020600; N2-020612; N2-020614; N2-020625</p>

```

-- 5 Common CAP Types
-- 5.1 Data types
CAP-datatypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3) cap-datatypes(52) version4(3)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS

    Duration,
    Integer4,
    Interval,
    LegID,
    ServiceKey
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
modules(0) cs1-datatypes(2) version1(0)}

    BothwayThroughConnectionInd,
    CallSegmentID {},
    CriticalityType,
    MiscCallInfo
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
cs2(20) modules(0) in-cs2-datatypes(0) version1(0)}

    IMSI,
    ISDN-AddressString,
    Ext-BasicServiceCode,
    NAEA-CIC
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}

    Ext-QoS-Subscribed,
    GeographicalInformation,
    GSN-Address,
    LocationInformation,
    LSAIdentity,
    QoS-Subscribed,
    SubscriberState
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}

    CallReferenceNumber,
    SuppressionOfAnnouncement
FROM MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-CH-DataTypes(13) version6(6)}

    tc-Messages,
    classes
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

```

```
TCInvokeIdSet
FROM TCAPMessages tc-Messages
```

```
EXTENSION,
PARAMETERS-BOUND,
SupportedExtensions
FROM CAP-classes classes
```

```
ExtensionContainer
FROM MAP-ExtensionDataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-ExtensionDataTypes(21) version6(6)}
```

```
;
```

```
AccessPointName {PARAMETERS-BOUND: bound} ::= OCTET STRING (SIZE(
    bound.&minAccessPointNameLength .. bound.&maxAccessPointNameLength))
-- Indicates the AccessPointName, refer to 3GPP TS 24.008 [9] for the encoding.
-- It shall be coded as in the value part defined in 3GPP TS 24.008,
-- i.e. the 3GPP TS 24.008 IEI and 3GPP TS 24.008 octet length indicator
-- shall not be included.
```

```
AChBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING
(SIZE
    (bound.&minAChBillingChargingLength ..
bound.&maxAChBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of the
type --
    CAMEL-AChBillingChargingCharacteristics {bound}})
-- The AChBillingChargingCharacteristics parameter specifies the charging
related information
-- to be provided by the gsmSSF and the conditions on which this information has
to be reported
-- back to the gsmSCF with the ApplyChargingReport operation. The value of the
-- AChBillingChargingCharacteristics of type OCTET STRING carries a value of the
ASN.1 data type:
-- CAMEL-AChBillingChargingCharacteristics. The normal encoding rules are used
to encode this
-- value.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.
```

```
AChChargingAddress {PARAMETERS-BOUND : bound} ::= CHOICE {
    legID [2] LegID,
    srfConnection [50] CallSegmentID {bound}
}
```

```
AdditionalCallingPartyNumber {PARAMETERS-BOUND : bound} ::= Digits {bound}
-- Indicates the Additional Calling Party Number.
```

```
AlertingPattern ::= OCTET STRING (SIZE(3))
-- Indicates a specific pattern that is used to alert a subscriber
-- (e.g. distinctive ringing, tones, etc.).
-- The encoding of the last octet of this parameter is as defined in 3GPP TS
29.002 [11].
-- Only the trailing OCTET is used, the remaining OCTETS shall be sent as NULL
(zero)
```

```

-- The receiving side shall ignore the leading two OCTETS.

AOCBeforeAnswer ::= SEQUENCE {
    aOCInitial                [0] CAI-GSM0224,
    aOCSubsequent             [1] AOCSubsequent
                                OPTIONAL
}

AOCGPRS ::= SEQUENCE {
    aOCInitial                [0] CAI-GSM0224,
    aOCSubsequent             [1] AOCSubsequent
                                OPTIONAL
}

AOCSubsequent ::= SEQUENCE {
    cAI-GSM0224                [0] CAI-GSM0224 ,
    tariffSwitchInterval      [1] INTEGER (1..86400)
                                OPTIONAL
}
-- tariffSwitchInterval is measured in 1 second units

AppendFreeFormatData ::= ENUMERATED {
    overwrite (0),
    append (1)
}

ApplicationTimer ::= INTEGER (0..2047)
-- Used by the gsmSCF to set a timer in the gsmSSF. The timer is in seconds.

AssistingSSPIPRoutingAddress {PARAMETERS-BOUND : bound} ::= Digits {bound}
-- Indicates the destination address of the gsmSRF for the assist procedure.

AudibleIndicator ::= CHOICE {
    Tone                        BOOLEAN,
    burstList                   [1] Burstlist
}

BackwardServiceInteractionInd ::= SEQUENCE {
    conferenceTreatmentIndicator [1] OCTET STRING (SIZE(1))
                                OPTIONAL,
    -- acceptConferenceRequest   'xxxx xx01'B
    -- rejectConferenceRequest   'xxxx xx10'B
    -- if absent from Connect or ContinueWithArgument,
    -- then CAMEL service does not affect conference treatment
    callCompletionTreatmentIndicator [2] OCTET STRING (SIZE(1))
                                OPTIONAL,
    -- acceptCallCompletionServiceRequest 'xxxx xx01'B,
    -- rejectCallCompletionServiceRequest 'xxxx xx10'B
    -- if absent from Connect or ContinueWithArgument,
    -- then CAMEL service does not affect call completion treatment
    ...
}

BasicGapCriteria {PARAMETERS-BOUND : bound} ::= CHOICE {
    calledAddressValue          [0] Digits {bound},
    gapOnService                [2] GapOnService,
    calledAddressAndService     [29] SEQUENCE {

```



```

    }
-- burstInterval, toneDuration and toneInterval are measured in 100 millisecond
units

BurstList ::= SEQUENCE {
    warningPeriod [0] INTEGER (1..1200)
DEFAULT 30,
    bursts [1] Burst
}
-- warningPeriod is measured in 1 second units.

CAI-GSM0224 ::= SEQUENCE {
    e1 [0] INTEGER
(0..8191) OPTIONAL,
    e2 [1] INTEGER
(0..8191) OPTIONAL,
    e3 [2] INTEGER
(0..8191) OPTIONAL,
    e4 [3] INTEGER
(0..8191) OPTIONAL,
    e5 [4] INTEGER
(0..8191) OPTIONAL,
    e6 [5] INTEGER
(0..8191) OPTIONAL,
    e7 [6] INTEGER
(0..8191) OPTIONAL
}
-- Indicates Charge Advice Information to the Mobile Station. For information
regarding
-- parameter usage, refer to 3GPP TS 22.024 [2].

CalledPartyBCDNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minCalledPartyBCDNumberLength ..
bound.&maxCalledPartyBCDNumberLength))
-- Indicates the Called Party Number, including service selection information.
-- Refer to 3GPP TS 24.008 [9] for encoding.
-- This data type carries only the "type of number", "numbering plan
-- identification" and "number digit" fields defined in 3GPP TS 24.008 [9];
-- it does not carry the "called party BCD number IEI" or "length of called
-- party BCD number contents".

CalledPartyNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minCalledPartyNumberLength .. bound.&maxCalledPartyNumberLength))
-- Indicates the Called Party Number. Refer to ETS EN 300 356-1 [23] for
encoding.

-- A CalledPartyNumber may contain national-specific values of the Nature Of
Address
-- indicator. The filling-in of the national-specific Nature Of Address
indicator
-- values shall be done in accordance with the national ISUP of the gsmSSF
country, e.g.
-- ANSI T1.113-1995 [92].
-- In terms of ETS EN 300 356-1 [23], the Destination Address Field is not
present if the
-- destination address length is set to zero. This is the case e.g. when the
ANSI

```

-- ISUP Nature Of Address indicator indicates no number present, operator requested
-- (1110100) or no number present, cut-through call to carrier (1110101).
-- See also see 3GPP TS 23.078 [7].

CallingPartyNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
bound.&minCallingPartyNumberLength .. bound.&maxCallingPartyNumberLength))
-- Indicates the Calling Party Number. Refer to ETSI EN 300 356-1 [23] for
encoding.

CallResult {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
bound.&minCallResultLength .. bound.&maxCallResultLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
CAMEL-CallResult {bound}})
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.

-- This parameter provides the gsmSCF with the charging related information
previously requested
-- using the ApplyCharging operation. This shall include the partyToCharge
parameter as
-- received in the related ApplyCharging operation to correlate the result to
the request

CallSegmentFailure {PARAMETERS-BOUND : bound} ::= SEQUENCE {
callSegmentID [0] CallSegmentID {bound}
cause OPTIONAL, [2] Cause {bound}
OPTIONAL,
...
}

CallSegmentToCancel ::= SEQUENCE {
invokeID [0] Invoke ID
OPTIONAL,
callSegmentID [1] CallSegmentID
{bound} OPTIONAL,
...
}

CAMEL-AChBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= CHOICE {
timeDurationCharging [0] SEQUENCE {
maxCallPeriodDuration [0] INTEGER (1..864000),
releaseIfdurationExceeded [1] BOOLEAN DEFAULT FALSE,
tariffSwitchInterval [2] INTEGER (1..86400)
OPTIONAL,
audibleIndicator [3] AudibleIndicator
DEFAULT {tone FALSE},
extensions [4] Extensions
{bound} OPTIONAL,
...
}

-- tariffSwitchInterval is measured in 1 second units.
-- maxCallPeriodDuration is measured in 100 millisecond units

CAMEL-CallResult {PARAMETERS-BOUND : bound} ::= CHOICE {

```

    TimeDurationChargingResult          [0] SEQUENCE {
        partyToCharge                    [0] ReceivingSideID,
        timeInformation                   [1] TimeInformation,
        callActive                        [2] BOOLEAN DEFAULT
TRUE,
        callReleasedAtTcpExpiry          [3] NULL
        OPTIONAL,
        extensions                        [4] Extensions
{bound}
        aChChargingAddress               [5]
AChChargingAddress {bound}

    DEFAULT legID:receivingSideID:leg1,
        ...
    }
}

CAMEL-FCIBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= CHOICE{
    fCIBCCAMELsequence1                 [0] SEQUENCE {
        freeFormatData                  [0] OCTET
STRING (SIZE(
        bound.&minFCIBillingChargingDataLength ..
bound.&maxFCIBillingChargingDataLength)),
        partyToCharge                   [1] SendingSideID
DEFAULT sendingSideID : leg1,
        appendFreeFormatData           [2] AppendFreeFormatData
DEFAULT overwrite
    }
}

CAMEL-FCIGPRSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::=
SEQUENCE{
    fCIBCCAMELsequence1                 [0] SEQUENCE {
        freeFormatData                  [0] OCTET STRING
(SIZE(
        bound.&minFCIBillingChargingDataLength ..
bound.&maxFCIBillingChargingDataLength)),
        pDPID                           [1] PDPID OPTIONAL,
        appendFreeFormatData           [2] AppendFreeFormatData
DEFAULT overwrite,
        ...
    }
}

CAMEL-FCISMSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::=
CHOICE{
    fCIBCCAMELsequence1                 [0] SEQUENCE {
        freeFormatData                  [0] OCTET STRING
(SIZE(
        bound.&minFCIBillingChargingDataLength ..
bound.&maxFCIBillingChargingDataLength)),
        appendFreeFormatData           [1] AppendFreeFormatData
DEFAULT overwrite
    }
}

CAMEL-SCIBillingChargingCharacteristics ::= CHOICE {

```

```

        aOCBeforeAnswer                [0] AOCBeforeAnswer,
        aOCAfterAnswer                 [1] AOCSubsequent
    }

CAMEL-SCIGPRSBillingChargingCharacteristics ::= SEQUENCE {
    aOCGPRS                            [0] AOCGPRS,
    pDPID                              [1] PDPID
    OPTIONAL,
    ...
}

Carrier {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minCarrierLength .. bound.&maxCarrierLength))
-- This parameter is used for North America (na) only.
-- It contains the carrier selection field (first octet) followed by Carrier ID
-- information (North America (na)).

-- The Carrier selection is one octet and is encoded as:
-- 00000000      No indication
-- 00000001      Selected carrier identification code (CIC) pre subscribed and
not
--
--                input by calling party
-- 00000010      Selected carrier identification code (CIC) pre subscribed and
input by
--
--                calling party
-- 00000011      Selected carrier identification code (CIC) pre subscribed, no
--
--                indication of whether input by calling party
(undetermined)
-- 00000100      Selected carrier identification code (CIC) not pre subscribed
and
--
--                input by calling party
-- 00000101
--    to          Spare
-- 11111110
-- 11111111      Reserved

-- Refer to ANSI T1.113-1995 [92] for encoding of na carrier ID information (3
octets).

Cause {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minCauseLength .. bound.&maxCauseLength))
-- Indicates the cause for interface related information.
-- Refer to ETSI EN 300 356-1 [23] Cause parameter for encoding.
-- For the use of cause and location values refer to ITU-T Recommendation Q.850
[47]
-- Shall always include the cause value and shall also include the diagnostics
field,
-- if available.

CGEncountered ::= ENUMERATED {
    noCGencountered                (0),
    manualCGencountered            (1),
    scpOverload                    (2)
}
-- Indicates the type of automatic call gapping encountered, if any.

ChargeNumber {PARAMETERS-BOUND : bound} ::= LocationNumber {bound}

```

-- Information sent in either direction indicating the chargeable number for the call and
-- consisting of the odd/even indicator, nature of address indicator, numbering plan indicator,
-- and address signals.

-- Uses the LocationNumber format which is based on the Location Number format as defined
-- in ITU-T Recommendation Q.763 [45].
-- For example, the ChargeNumber may be a third party number to which a call is billed for
-- the 3rd party billing service. In this case, the calling party may request operator assistance
-- to charge the call to, for example, their home number.

-- For NA, this parameter uniquely identifies the chargeable number for a call sent into a North
-- American long distance carrier. It transports the ChargeNumber Parameter Field as defined in
-- ANSI T1.113-1995 [92]. This provides
-- - 1 octet for the nature of address indicator field, plus
-- - 1 octet for a numbering plan field, plus
-- - up to 5 octets for the address signal (up to 10 digits)

-- The Charge Number in ANSI T1.113-1995 [92] normally contains a 10 digit national number within
-- the North American Numbering Plan (NANP); longer (e.g. international) charge numbers are not
-- supported in ANSI T1.113-1995 [92].

```
ChargeUnitsComponent ::= SEQUENCE SIZE  
                        (bound.&minComponentLength .. bound.&maxComponentLength) OF  
SEQUENCE {  
    monitorComponent           [0] MonitorComponent,  
    reportingConditionUnits    [1] ReportingConditionUnits  
}
```

```
ChargeUnitsTotal ::= SEQUENCE {  
    reportingConditionUnits [0] ReportingConditionUnits  
}
```

```
ChargingCharacteristics ::= CHOICE {  
    MaxTransferredVolume      [0] INTEGER (1..4294967295),  
    maxElapsedTime            [1] INTEGER (1..86400)  
}  
-- maxTransferredVolume is measured in number of bytes  
-- maxElapsedTime is measured in seconds
```

```
ChargingResult ::= CHOICE {  
    transferredVolume         [0] TransferredVolume,  
    elapsedTime                [1] ElapsedTime  
}
```

```
ChargingRollOver ::= CHOICE {  
    transferredVolumeRollOver [0] TransferredVolumeRollOver,  
    elapsedTimeRollOver       [1] ElapsedTimeRollOver  
}
```

-- transferredVolumeRollOver shall be reported if ApplyChargingReportGPRS reports volume and
 -- a roll-over has occurred in one or more volume counters. Otherwise, it shall be absent.
 -- elapsedTimeRollOver shall be reported if ApplyChargingReportGPRS reports duration and
 -- a roll-over has occurred in one or more duration counters. Otherwise, it shall be absent.

```
CollectedDigits ::= SEQUENCE {
    MinimumNbOfDigits          [0] INTEGER (1..30)      DEFAULT
1,
    maximumNbOfDigits          [1] INTEGER (1..30),
    endOfReplyDigit            [2] OCTET STRING (SIZE
(1..2))                OPTIONAL,
    cancelDigit                [3] OCTET STRING (SIZE
(1..2))                OPTIONAL,
    startDigit                 [4] OCTET STRING (SIZE
(1..2))                OPTIONAL,
    firstDigitTimeOut          [5] INTEGER (1..127)
    OPTIONAL,
    interDigitTimeOut          [6] INTEGER (1..127)
    OPTIONAL,
    errorTreatment              [7] ErrorTreatment DEFAULT
stdErrorAndInfo,
    interruptableAnnInd        [8] BOOLEAN DEFAULT TRUE,
    voiceInformation            [9] BOOLEAN DEFAULT FALSE,
    voiceBack                  [10] BOOLEAN DEFAULT FALSE
}
```

-- The use of voiceBack and the support of voice recognition via voiceInformation
 -- is network operator specific.
 -- The endOfReplyDigit, cancelDigit, and startDigit parameters have been
 -- designated as OCTET STRING, and are to be encoded as BCD, one digit per octet
 -- only, contained in the four least significant bits of each OCTET. The
 -- following encoding shall
 -- be applied for the non-decimal characters:
 -- 1011 (*), 1100 (#).
 -- The usage is service dependent.
 -- firstDigitTimeOut and interDigitTimeOut are measured in seconds.

```
CollectedInfo ::= CHOICE {
    collectedDigits            [0] CollectedDigits
}
```

```
ConnectedNumberTreatmentInd ::= ENUMERATED {
    noINImpact                (0),
    presentationRestricted     (1),
    presentCalledINNumber      (2),
    presentCallINNumberRestricted (3)
}
```

-- This parameter is used to suppress or to display the connected number.

```
ControlType ::= ENUMERATED {
    sCPOverloaded              (0),
    manuallyInitiated          (1)
}
```

```
CompoundCriteria {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    BasicGapCriteria          [0] BasicGapCriteria {bound},
    scfID                     [1] ScfID {bound}
                                OPTIONAL
}
```

```
CorrelationID {PARAMETERS-BOUND : bound} ::= Digits {bound}
-- used by gsmSCF for correlation with a previous operation.
```

```
DateAndTime ::= OCTET STRING (SIZE(7))
-- DateAndTime is BCD encoded. The year digit indicating millenium occupies
bits
-- 0-3 of the first octet, and the year digit indicating century occupies bits
-- 4-7 of the first octet.
-- The year digit indicating decade occupies bits 0-3 of the second octet,
-- whilst the digit indicating the year within the decade occupies bits 4-7 of
-- the second octet.
-- The most significant month digit occupies bits 0-3 of the third octet,
-- and the least significant month digit occupies bits 4-7 of the third octet.
-- The most significant day digit occupies bits 0-3 of the fourth octet,
-- and the least significant day digit occupies bits 4-7 of the fourth octet.
-- The most significant hours digit occupies bits 0-3 of the fifth octet,
-- and the least significant digit occupies bits 4-7 of the fifth octet.
-- The most significant minutes digit occupies bits 0-3 of the sixth octet,
-- and the least significant digit occupies bits 4-7 of the sixth octet.
-- The most significant seconds digit occupies bits 0-3 of the seventh octet,
-- and the least seconds significant digit occupies bits 4-7 of the seventh
octet.
-- For the encoding of digits in an octet, refer to the timeAndtimezone
parameter.
```

```
DestinationRoutingAddress {PARAMETERS-BOUND : bound} ::= SEQUENCE SIZE(1) OF
    CalledPartyNumber {bound}
-- Indicates the Called Party Number.
```

```
Digits {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minDigitsLength .. bound.&maxDigitsLength))
-- Indicates the address signalling digits.
-- Refer to ETSI EN 300 356-1 [23] Generic Number & Generic Digits parameters
for encoding.
-- The coding of the subfields "NumberQualifier" in Generic Number and
"TypeOfDigits" in
-- Generic Digits are irrelevant to the CAP;
-- the ASN.1 tags are sufficient to identify the parameter.
-- The ISUP format does not allow to exclude these subfields,
-- therefore the value is network operator specific.
--
-- The following parameters shall use Generic Number:
-- - AdditionalCallingPartyNumber for InitialDP
-- - AssistingSSPIPRoutingAddress for EstablishTemporaryConnection
-- - CorrelationID for AssistRequestInstructions
-- - CalledAddressValue for all occurrences, CallingAddressValue for all
occurrences.
--
-- The following parameters shall use Generic Digits:
```

```

-- - CorrelationID in EstablishTemporaryConnection
-- - number in VariablePart
-- - digitsResponse in ReceivedInformationArg
-- - midCallEvents in oMidCallSpecificInfo and tMidCallSpecificInfo
--
-- In the digitsResponse and midCallevents, the digits may also include the '*',
'#',
-- a, b, c and d digits by using the IA5 character encoding scheme. If the BCD
even or
-- BCD odd encoding scheme is used, then the following encoding shall be applied
for the
-- non-decimal characters: 1011 (*), 1100 (#).

```

```

-- Note that when CorrelationID is transported in Generic Digits, then the
digits shall
-- always be BCD encoded.

```

```

DpSpecificCriteria ::= CHOICE {
    applicationTimer                [1] ApplicationTimer,
    midCallControlInfo              [2] MidCallControlInfo
}

```

```

-- The gsmSCF may set a timer in the gsmSSF for the No_Answer event.
-- If the user does not answer the call within the allotted time,
-- then the gsmSSF reports the event to the gsmSCF.
-- The gsmSCF may define a criterion for the detection of DTMF digits during
a call.

```

```

ElapsedTime ::= CHOICE {
    TimeGPRSIfNoTariffSwitch        [0] INTEGER (0..86400),
    timeGPRSIfTariffSwitch          [1] SEQUENCE {
        timeGPRSSinceLastTariffSwitch [0] INTEGER (0..86400),
        timeGPRSTariffSwitchInterval [1] INTEGER (0..86400)
        OPTIONAL
    }
}

```

```

-- timeGPRSIfNoTariffSwitch is measured in seconds
-- timeGPRSSinceLastTariffSwitch and timeGPRSTariffSwitchInterval are
measured in seconds

```

```

ElapsedTimeRollOver ::= CHOICE {
    rO-TimeGPRSIfNoTariffSwitch     [0] INTEGER (0..255),
    rO-TimeGPRSIfTariffSwitch       [1] SEQUENCE {
        rO-TimeGPRSSinceLastTariffSwitch [0] INTEGER (0..255)
        OPTIONAL,
        rO-TimeGPRSTariffSwitchInterval [1] INTEGER (0..255)
        OPTIONAL
    }
}

```

```

-- rO-TimeGPRSIfNoTariffSwitch, rO-TimeGPRSSinceLastTariffSwitch and
-- rO-TimeGPRSTariffSwitchInterval
-- present counters indicating the number of parameter range rollovers.

```

```

EndUserAddress {PARAMETERS-BOUND: bound} ::= SEQUENCE {
    pDPTYPEOrganization             [0] OCTET STRING
(SIZE(1)),
    pDPTYPENumber                   [1] OCTET STRING
(SIZE(1)),
}

```



```

        pDPAddress [2] OCTET STRING (SIZE(
            bound.&minPDPAddressLength .. bound.&maxPDPAddressLength))
            OPTIONAL
    }
-- Indicates the EndUserAddress, refer to 3GPP TS 29.060 [12] for the encoding.
-- The pDPTypeOrganization shall use the least significant 4 bits of the octet
encoded.
-- The sender of this parameter shall set the most significant 4 bits of the
octet to 1.
-- The receiver of this parameter shall ignore the most significant 4 bits of
this octet.

ErrorTreatment ::= ENUMERATED {
    stdErrorAndInfo (0),
    help (1),
    repeatPrompt (2)
}
-- stdErrorAndInfo means returning the "ImproperCallerResponse" error in the
event of an error
-- condition during collection of user info.

EvaluesAndTsw ::= SEQUENCE {
    evalues [0] CAI-GSM0224,
    tariffSwitchInterval [1] TariffSwitchInterval
}

EvaluesComponent ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE
{
    monitorComponent [0] MonitorComponent,
    reportingConditionEvaluate [1] ReportingConditionEvaluate
}

EvaluesTotal ::= SEQUENCE {
    ReportingConditionEvaluate [0] ReportingConditionEvaluate
}

EventSpecificChargingInformation {PARAMETERS-BOUND : bound} ::= CHOICE {
    infoChargeUnitsTotal [0] InfoChargeUnitsTotal,
    infoChargeUnitsComponents [1] InfoChargeUnitsComponents
{bound},
    infoEvaluateTotal [2] InfoEvaluateTotal,
    infoEvaluateComponents [3] InfoEvaluateComponents {bound}
}

EventSpecificInformationBCSM {PARAMETERS-BOUND : bound} ::= CHOICE {
    routeSelectFailureSpecificInfo [2] SEQUENCE {
        failureCause [0] Cause {bound}
        OPTIONAL,
        ...
    },
    oCalledPartyBusySpecificInfo [3] SEQUENCE {
        busyCause [0] Cause {bound}
        OPTIONAL,
        ...
    },
    oNoAnswerSpecificInfo [4] SEQUENCE {

```

```

-- no specific info defined --
...
},
oAnswerSpecificInfo [5] SEQUENCE {
  destinationAddress [50]
CalledPartyNumber {bound} OPTIONAL,
  or-Call [51] NULL
  forwardedCall [52] NULL
  OPTIONAL,
...
},
oMidCallSpecificInfo [6] SEQUENCE {
  midCallEvents [1] CHOICE {
    dTMFDigitsCompleted [3] Digits,
    dTMFDigitsTimeOut [4] Digits
  }
  OPTIONAL,
...
},
oDisconnectSpecificInfo [7] SEQUENCE {
  releaseCause [0] Cause {bound}
  OPTIONAL,
...
},
tBusySpecificInfo [8] SEQUENCE {
  busyCause [0] Cause {bound}
  callForwarded [50] NULL
  OPTIONAL,
  routeNotPermitted [51] NULL
  OPTIONAL,
  forwardingDestinationNumber [52] CalledPartyNumber
{bound} OPTIONAL,
...
},
tNoAnswerSpecificInfo [9] SEQUENCE {
  callForwarded [50] NULL
  OPTIONAL,
  forwardingDestinationNumber [52] CalledPartyNumber
{bound} OPTIONAL,
...
},
tAnswerSpecificInfo [10] SEQUENCE {
  destinationAddress [50]
CalledPartyNumber {bound} OPTIONAL,
  or-Call [51] NULL
  forwardedCall [52] NULL
  OPTIONAL,
...
},
tMidCallSpecificInfo [11] SEQUENCE {
  midCallEvents [1] CHOICE {
    dTMFDigitsCompleted [3] Digits,
    dTMFDigitsTimeOut [4] Digits

```

```

        }
        OPTIONAL,
        ...
    },
    tDisconnectSpecificInfo
        releaseCause
        [12] SEQUENCE {
            [0] Cause {bound}
        }
        OPTIONAL,
        ...
    },
    oTermSeizedSpecificInfo
        locationInformation
        [13] SEQUENCE {
            [50]
        }
    LocationInformation
    OPTIONAL,
    ...
    },
    callAcceptedSpecificInfo
        locationInformation
        [20] SEQUENCE {
            [50]
        }
    LocationInformation
    OPTIONAL,
    ...
    },
    oAbandonSpecificInfo
        routeNotPermitted
        [21] SEQUENCE {
            [50] NULL
        }
        OPTIONAL,
    ...
    },
    oChangeOfPositionSpecificInfo
        locationInformation
        [50] SEQUENCE {
            [50]
        }
    LocationInformation
    OPTIONAL,
    ...
    },
    tChangeOfPositionSpecificInfo
        locationInformation
        [51] SEQUENCE {
            [50]
        }
    LocationInformation
    OPTIONAL,
    ...
    }
}
-- Indicates the call related information specific to the event.

EventSpecificInformationSMS ::= CHOICE {
    o-smsFailureSpecificInfo
        failureCause
        [0] SEQUENCE {
            [0] MO-SMSCause
        }
        OPTIONAL,
    ...
    },
    o-smsSubmissionSpecificInfo
        -- no specific info defined-
        [1] SEQUENCE {
        }
    ...
    },
    t-smsFailureSpecificInfo
        failureCause
        [2] SEQUENCE {
            [0] MT-SMSCause
        }
        OPTIONAL,
    ...
    },
    t-smsDeliverySpecificInfo
        -- no specific info defined-
        [3] SEQUENCE {
        }
    ...
    }
}

```

```

EventTypeBCSM ::= ENUMERATED {
    collectedInfo (2),
    analyzedInformation (3),
    routeSelectFailure (4),
    oCalledPartyBusy (5),
    oNoAnswer (6),
    oAnswer (7),
    oMidCall (8),
    oDisconnect (9),
    oAbandon (10),
    termAttemptAuthorized (12),
    tBusy (13),
    tNoAnswer (14),
    tAnswer (15),
    tMidCall (16),
    tDisconnect (17),
    tAbandon (18),
    oTermSeized (19),
    callAccepted (27),
    oChangeOfPosition (50),
    tChangeOfPosition (51)
}
-- Indicates the BCSM detection point event.
-- Values collectedInfo, analyzedInformation and termAttemptAuthorized may be
used
-- for TDPs only.

EventTypeChargingPLMN {PARAMETERS-BOUND : bound} ::= CHOICE {
    ChargeUnitsTotal [0] ChargeUnitsTotal,
    ChargeUnitsComponent [1] ChargeUnitsComponent
{bound},
    EvaluesTotal [2] EvaluesTotal,
    EvaluesComponent [3] EvaluesComponent {bound}
}
-- This parameter indicates the charging event type.

EventTypeSMS ::= ENUMERATED {
    sms-CollectedInfo (1),
    o-smsFailure (2),
    o-smsSubmission (3),
    sms-DeliveryRequested (11),
    t-smsFailure (12),
    t-smsDelivery (13)
}
-- Values sms-CollectedInfo and sms-DeliveryRequested may be used for TDPs only.

Extensions {PARAMETERS-BOUND : bound} ::= SEQUENCE SIZE
(1..bound.&numOfExtensions) OF ExtensionField
ExtensionField ::= SEQUENCE {
    Type EXTENSION.&id
({SupportedExtensions {bound}}),
    -- shall identify
the value of an EXTENSION type
    criticality CriticalityType DEFAULT
ignore,

```

```

        value [1]
EXTENSION.&ExtensionType ({SupportedExtensions}{@type}),
    ...
}
-- This parameter indicates an extension of an argument data type.
-- Its content is network operator specific

FCIBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING
(SIZE(
    bound.&minFCIBillingChargingLength .. bound.&maxFCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
-
- CAMEL-FCIBillingChargingCharacteristics {bound}})
-- This parameter indicates the billing and/or charging characteristics.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.

FCIGPRSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET
STRING (SIZE(
    bound.&minFCIBillingChargingLength .. bound.&maxFCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
- CAMEL-FCIGPRSBillingChargingCharacteristics {bound}})
-- This parameter indicates the GPRS billing and/or charging characteristics.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.

FCISMSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING
(SIZE(
    bound.&minFCIBillingChargingLength .. bound.&maxFCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
- CAMEL-FCISMSBillingChargingCharacteristics {bound}})
-- This parameter indicates the SMS billing and/or charging characteristics.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.

ForwardServiceInteractionInd ::= SEQUENCE {
    conferenceTreatmentIndicator [1] OCTET STRING (SIZE(1))
        OPTIONAL,
    -- acceptConferenceRequest 'xxxx xx01'B
    -- rejectConferenceRequest 'xxxx xx10'B
    -- if absent from Connect or ContinueWithArgument,
    -- then CAMEL service does not affect conference treatment
    callDiversionTreatmentIndicator [2] OCTET STRING (SIZE(1))
        OPTIONAL,
    -- callDiversionAllowed 'xxxx xx01'B
    -- callDiversionNotAllowed 'xxxx xx10'B
    -- if absent from Connect or ContinueWithArgument,
    -- then CAMEL service does not affect call diversion treatment
    callingPartyRestrictionIndicator [4] OCTET STRING (SIZE(1))
        OPTIONAL,
    -- noINIImpact 'xxxx xx01'B
    -- presentationRestricted 'xxxx xx10'B
    -- if absent from Connect or ContinueWithArgument,
    -- then CAMEL service does not affect calling party restriction treatment
    ...
}

```

```

GapCriteria {PARAMETERS-BOUND : bound} ::= CHOICE {
    basicGapCriteria                BasicGapCriteria {bound},
    compoundGapCriteria              CompoundCriteria {bound}
}

GapIndicators ::= SEQUENCE {
    duration                        [0] Duration,
    gapInterval                     [1] Interval,
    ...
}
-- Indicates the call gapping characteristics.
-- No call gapping when gapInterval equals 0.

GapOnService ::= SEQUENCE {
    serviceKey                      [0] ServiceKey,
    ...
}

GapTreatment {PARAMETERS-BOUND : bound} ::= CHOICE {
    informationToSend               [0] InformationToSend {bound},
    releaseCause                   [1] Cause {bound}
}
-- The default value for Cause is the same as in ISUP.

GenericNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minGenericNumberLength .. bound.&maxGenericNumberLength))
-- Indicates a generic number. Refer to ETSI EN 300 356-1 [23] Generic number
for encoding.

GenericNumbers {PARAMETERS-BOUND : bound} ::= SET
SIZE(1..bound.&numOfGenericNumbers) OF GenericNumber {bound}

GPRS-QoS ::= CHOICE {
    short-QoS-format               [0] QoS-Subscribed,
    long-QoS-format                [1] Ext-QoS-Subscribed
}
-- Short-QoS-format shall be sent for QoS in pre GSM release 99 format.
-- Long-QoS-format shall be sent for QoS in GSM release 99 (and beyond) format.
-- Which of the two QoS formats shall be sent is determined by which QoS
-- format is available in the SGSN at the time of sending.
-- Refer to 3GPP TS 29.002 [11] for encoding details of QoS-Subscribed and
-- Ext-QoS-Subscribed.

GPRSCause {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE
    (bound.&minGPRSCauseLength .. bound.&maxGPRSCauseLength))
-- Shall only include the cause value.

-- 00000000 Unspecified
-- All other values shall be interpreted as "Unspecified".
--
-- This parameter indicates the cause for CAP interface related information.
-- The GPRSCause mapping to/from GTP cause values specified in the 3GPP TS
29.060 [12] and
-- to/from 3GPP TS 24.008 [9] GMM cause and SM cause values are outside scope of
this document.

GPRSEvent ::= SEQUENCE {

```

```

    gPRSEventType          [0] GPRSEventType,
    monitorMode            [1] MonitorMode
}
-- Indicates the GPRS event information for monitoring.

GPRSEventSpecificInformation {PARAMETERS-BOUND : bound} ::= CHOICE {

    attachChangeOfPositionSpecificInformation          [0] SEQUENCE {
        locationInformationGPRS          [0]
LocationInformationGPRS          OPTIONAL,
        ...
    },

    pdp-ContextchangeOfPositionSpecificInformation    [1] SEQUENCE {
        accessPointName                  [0]
AccessPointName {bound}          OPTIONAL,
        chargingID                       [1]   GPRSChargingID
                                           OPTIONAL,
        locationInformationGPRS          [2]
LocationInformationGPRS          OPTIONAL,
        endUserAddress                   [3]   EndUserAddress
{bound}                          OPTIONAL,
        qualityOfService                 [4]   QualityOfService
                                           OPTIONAL,
        timeAndTimezone {bound}          [5]
TimeAndTimezone {bound}          OPTIONAL,
        ...
        gGSNAddress                      [6]   GSN-Address
                                           OPTIONAL
    },

    detachSpecificInformation              [2] SEQUENCE {
        initiatingEntity                  [0] InitiatingEntity
                                           OPTIONAL,
        ...
        routeingAreaUpdate               [1]   NULL
                                           OPTIONAL
    },

    disconnectSpecificInformation          [3] SEQUENCE {
        initiatingEntity                  [0] InitiatingEntity
                                           OPTIONAL,
        ...
        routeingAreaUpdate               [1]   NULL
                                           OPTIONAL
    },

    pDPContextEstablishmentSpecificInformation        [4] SEQUENCE {
        accessPointName                  [0]
AccessPointName {bound}          OPTIONAL,
        endUserAddress                   [1]   EndUserAddress
{bound}                          OPTIONAL,
        qualityOfService                 [2]   QualityOfService
                                           OPTIONAL,

```

```

                locationInformationGPRS                [3]
LocationInformationGPRS                OPTIONAL,
                timeAndTimeZone                        [4]
                TimeAndTimezone {bound}              OPTIONAL,
                pdpInitiationType                    [5] PDPInitiationType
                OPTIONAL,
                ...,
                secondaryPDP-context                 [6] NULL
                OPTIONAL
            },

    pdpContextEstablishmentAcknowledgementSpecificInformation [5] SEQUENCE {
                accessPointName                       [0]
                AccessPointName {bound}              OPTIONAL,
                chargingID                             [1] GPRSChargingID
                OPTIONAL,
                endUserAddress                         [2] EndUserAddress
                OPTIONAL,
                qualityOfService                       [3] QualityOfService
                OPTIONAL,
                locationInformationGPRS                [4]
                LocationInformationGPRS                OPTIONAL,
                timeAndTimeZone                        [5]
                TimeAndTimezone {bound}              OPTIONAL,
                ...,
                gGSNAddress                            [6] GSN-Address
                OPTIONAL
            }
    }

```

```

GPRSEventType ::= ENUMERATED {
    attach (1),
    attachChangeOfPosition (2),
    detached (3),
    pdp-ContextEstablishment (11),
    pdp-ContextEstablishmentAcknowledgement (12),
    disonnect (13),
    pdp-ContextChangeOfPosition (14)
}

```

```

InbandInfo {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    messageID [0] MessageID {bound},
    numberOfRepetitions [1] INTEGER (1..127)
    OPTIONAL,
    duration [2] INTEGER (0..32767)
    OPTIONAL,
    interval [3] INTEGER (0..32767)
    OPTIONAL,
    ...
}

```

```

-- Interval is the time in seconds between each repeated announcement. Duration
is the total
-- amount of time in seconds, including repetitions and intervals.
-- The end of announcement is either the end of duration or numberOfRepetitions,
-- whatever comes first.
-- duration with value 0 indicates infinite duration

```



```

InfoChargeUnitsComponents ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE
{
    reportingConditionUnits          [0] ReportingConditionUnits,
    units                            [1] Units,
    monitorComponent                  [2] MonitorComponent
}

InfoChargeUnitsTotal ::= SEQUENCE {
    ReportingConditionUnits          [0] ReportingConditionUnits,
    Units                            [1] Units
}

InfoValueComponents ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE
{
    reportingConditionEvaluate        [0] ReportingConditionEvaluate,
    evaluatesAndTsw                   [1] EvaluatesAndTsw,
    monitorComponent                  [2] MonitorComponent
}

InfoValueTotal ::= SEQUENCE {
    ReportingConditionEvaluate        [0] ReportingConditionEvaluate,
    EvaluatesAndTsw                  [1] EvaluatesAndTsw
}

InformationToSend {PARAMETERS-BOUND : bound} ::= CHOICE {
    inbandInfo                        [0] InbandInfo {bound},
    tone                              [1] Tone
}

InitiatingEntity ::= ENUMERATED {
    mobileStation                      (0),
    sgsn                              (1),
    hlr                                (2),
    ggsn                              (3)
}

InvokeID ::= TCInvokeIdSet

IPRoutingAddress {PARAMETERS-BOUND : bound} ::= CalledPartyNumber {bound}
-- Indicates the routing address for the IP.

IPSSPCapabilities {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minIPSSPCapabilitiesLength .. bound.&maxIPSSPCapabilitiesLength))
-- Indicates the gsmSRF resources available. The parameter has two parts, a
standard and a
-- bilateral part. The standard part indicates capabilities defined as optional
in CAP V.2
-- that shall be recognised (but not necessarily supported) by a CAP V.2
gsmSCF. The bilateral
-- part contains further information that is not specified in this standard,
but which is set
-- according to bilateral agreements between network operators and/or equipment
vendors.

```

```

-- The last octet of the standard part is indicated by bit 7 being set to 0,
otherwise Bit 7 of
-- a standard part octet is set to 1 indicating that the standard part
continues in the following
-- octet. Coding is as follows:

-- Octet 1          Standard Part for CAP V.3
-- Bit      Value   Meaning
-- 0 0          IPRoutingAddress not supported
--           1      IPRoutingAddress supported
-- 1 0          VoiceBack not supported
--           1      VoiceBack supported
-- 2 0          VoiceInformation not supported, via speech
recognition
--           1      VoiceInformation supported, via speech
recognition
-- 3 0          VoiceInformation not supported, via voice
recognition
--           1      VoiceInformation supported, via voice
recognition
-- 4 0          Generation of voice announcements from Text
not supported
--           1      Generation of voice announcements from
Text supported
-- 5 -          Reserved
-- 6 -          Reserved
-- 7 0          End of standard part
--           1      This value is reserved in CAP V.3
--
-- Octets 2 to 4   Bilateral Part: Network operator/equipment vendor
specific

LegOrCallSegment ::= CHOICE {
    callSegmentID          [0] CallSegmentID {bound},
    legID                  [1] LegID
}

LegType ::= OCTET STRING (SIZE(1))
leg1 LegType      ::= '01'H
leg2 LegType      ::= '02'H

LocationNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE (
    bound.&minLocationNumberLength .. bound.&maxLocationNumberLength))
-- Indicates the Location Number for the calling party.
-- Refer to ETSI EN 300 356-1 [23] for encoding.

MessageID {PARAMETERS-BOUND : bound} ::= CHOICE {
    elementaryMessageID   [0] Integer4,
    text                  [1] SEQUENCE {
        messageContent    [0] IA5String (SIZE(
            bound.&minMessageContentLength ..
bound.&maxMessageContentLength)),
        attributes        [1] OCTET STRING
            (SIZE(
                bound.&minAttributesLength .. bound.&maxAttributesLength))
            OPTIONAL
    },
}

```

```

        elementaryMessageIDs                [29] SEQUENCE SIZE (1..
bound.&numOfMessageIDs) OF Integer4,
        variableMessage                      [30] SEQUENCE {
            elementaryMessageID              [0] Integer4,
            variableParts                     [1] SEQUENCE SIZE
(1..5) OF VariablePart {bound}
        }
}
-- Use of the text parameter is network operator/equipment vendor specific.

MidCallControlInfo ::= SEQUENCE {
    MinimumNumberOfDigits                    [0] INTEGER (1..30) DEFAULT 1,
    MaximumNumberOfDigits                    [1] INTEGER (1..30) DEFAULT
30,
    endOfReplyDigit                          [2] OCTET STRING (SIZE
(1..2)) OPTIONAL,
    cancelDigit                              [3] OCTET STRING (SIZE
(1..2)) OPTIONAL,
    startDigit                              [4] OCTET STRING (SIZE
(1..2)) OPTIONAL,
    interDigitTimeout                        [6] INTEGER (1..127) DEFAULT
10
}
--
-- - minimumNumberOfDigits                specifies the minumum number of digits
that shall be collected
-- - maximumNumberOfDigits                specifies the maximum number of digits
that shall be collected
-- - endOfReplyDigit                      specifies the digit string that
denotes the end of the digits
--                                         to be collected.
-- - cancelDigit                          specifies the digit string that
indicates that the input shall
--                                         be erased and digit
collection shall start afresh.
-- - startDigit                            specifies the digit string that
denotes the start of the digits
--                                         to be collected.
-- - interDigitTimeout                    specifies the maximum duration in
seconds between successive
--                                         digits.
--
-- endOfReplyDigit, cancelDigit and startDigit shall contain digits in the range
0..9, '*' and '#'
-- only. The collected digits string, reported to the gsmSCF, shall include the
endOfReplyDigit and
-- the startDigit, if present.
--
-- endOfReplyDigit, cancelDigit and startDigit shall be encoded as BCD digits.
Each octet shall
-- contain one BCD digit, in the 4 least significant bits of each octet.
-- The following encoding shall be used for the over-decadic digits: 1011 (*),
1100 (#).

MonitorComponent ::= ENUMERATED {
    ChargesTransitNetwork                    (0),
    ChargesSpecificPLMN                      (1)
}

```

```

    }

MonitorMode ::= ENUMERATED {
    interrupted (0),
    notifyAndContinue (1),
    transparent (2)
}
-- Indicates the event is relayed and/or processed by the SSP.
-- Transparent means that the gsmSSF or gprsSSF does not notify the gsmSCF of
the event.
-- For the use of this parameter refer to the procedure descriptions in clause
11.
-- For the RequestNotificationCharging operation, "interrupted" shall not be
used in MonitorMode.

MO-SMSCause ::= ENUMERATED {
    systemFailure (0),
    unexpectedDataValue (1),
    facilityNotSupported (2),
    sM-DeliveryFailure (3),
    releaseFromRadioInterface (4)
}
-- MO SMS error values which are reported to gsmSCF.
-- Most of these values are received from the SMSC as a response to
-- MO-ForwardSM operation.

MT-SMSCause ::= OCTET STRING (SIZE (1))
-- This variable is sent to the gsmSCF for a Short Message delivery failure
-- notification.
-- If the delivery failure is due to RP-ERROR RPDU received from the MS,
-- then MT-SMSCause shall be set to the RP-Cause component in the RP-ERROR RPDU.
-- Refer to 3G TS 24.011 [10] for the encoding of RP-Cause values.
-- Otherwise, if the delivery failure is due to internal failure in the MSC or
SGSN
-- or time-out from the MS, then MT-SMSCause shall be set to "Protocol error,
-- unspecified", as defined in 3G TS 24.011 [10].

NAOliInfo ::= OCTET STRING (SIZE (1))
-- NA Oli information takes the same value as defined in ANSI T1.113-1995 [92]
-- e.g.      '3D'H - Decimal value 61 - Cellular Service (Type 1)
--          '3E'H - Decimal value 62 - Cellular Service (Type 2)
--          '3F'H - Decimal value 63 - Cellular Service (roaming)

OCSIApplicable ::= NULL
-- Indicates that the Originating CAMEL Subscription Information, if present,
shall be
-- applied on the outgoing call leg created with a Connect operation. For the
use of this
-- parameter see 3GPP TS 23.078 [7].

OriginalCalledPartyID {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minOriginalCalledPartyIDLength ..
bound.&maxOriginalCalledPartyIDLength))
-- Indicates the original called number. Refer to ETSI EN 300 356-1 [23]
Original Called Number
-- for encoding.

```

```

PDPID ::= OCTET STRING (SIZE (1))
-- PDP Identifier is a counter used to identify a specific PDP Context within
a control
-- relationship between gprsSSF and gsmSCF.

PDPInitiationType ::= ENUMERATED {
    mSInitiated (0),
    networkInitiated (1)
}

QualityOfService ::= SEQUENCE {
    requested-QoS [0] GPRS-QoS
        OPTIONAL,
    subscribed-QoS [1] GPRS-QoS
        OPTIONAL,
    negotiated-QoS [2] GPRS-QoS
        OPTIONAL,
    ...
}
-- The procedure descriptions in clause 11 indicate which one(s) of the
-- QoS variables shall be transported.

ReceivingSideID ::= CHOICE {
    ReceivingSideID [1] LegType
}
-- used to identify LegID in operations sent from gsmSSF to gsmSCF

RedirectingPartyID {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE (
    bound.&minRedirectingPartyIDLength .. bound.&maxRedirectingPartyIDLength))
-- Indicates redirecting number.
-- Refer to ETSI EN 300 356-1 [23] Redirecting number for encoding.

ReportingConditionUnits ::= CHOICE {
    EndOfConnection [0] IMPLICIT NULL,
    ThresholdCounterValue [1] IMPLICIT INTEGER {SIZE(4)}
}

ReportingConditionValue ::= CHOICE {
    OccurrenceOfEvent [0] IMPLICIT NULL
}

RequestedInformationList {PARAMETERS-BOUND : bound} ::= SEQUENCE SIZE (1..
numOfInfoItems) OF RequestedInformation {bound}

RequestedInformationTypeList ::= SEQUENCE SIZE (1.. numOfInfoItems) OF
RequestedInformationType

RequestedInformation {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    RequestedInformationType [0] RequestedInformationType,
    requestedInformationValue [1] RequestedInformationValue
}
{bound},
...
}

RequestedInformationType ::= ENUMERATED {
    CallAttemptElapsedTime (0),
    callStopTime (1),

```

```

    callConnectedElapsedTime          (2),
    releaseCause                      (30)
}

RequestedInformationValue {PARAMETERS-BOUND : bound} ::= CHOICE {
    callAttemptElapsedTimeValue      [0] INTEGER (0..255),
    callStopTimeValue                [1] DateAndTime,
    callConnectedElapsedTimeValue    [2] Integer4,
    releaseCauseValue                [30] Cause {bound}
}

-- The callAttemptElapsedTimeValue is specified in seconds. The unit for the
-- callConnectedElapsedTimeValue is 100 milliseconds

RPCause ::= OCTET STRING (SIZE (1))
-- RP cause according to 3GPP TS 24.011 [10] or 3G TS 29.002 [11].
-- GsmSCF shall send this cause in the ReleaseSMS operation.
-- For a MO-SMS service, the MSC or SGSN shall send the RP Cause to the
originating MS.
-- It shall be used to overwrite the RP-Cause element in the RP-ERROR RPDU.
-- For a MT-SMS service, the MSC or SGSN shall send the RP Cause to the sending
SMS-GMSC.
-- It shall be used to overwrite the RP-Cause element in the RP-ERROR RPDU.

ScfID {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minScfIDLength .. bound.&maxScfIDLength))
-- defined by network operator.
-- Indicates the gsmSCF identity.

SCIBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING
(SIZE (
    bound.&minSCIBillingChargingLength .. bound.&maxSCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
-
    CAMEL-SCIBillingChargingCharacteristics})
-- Indicates AOC information to be sent to a Mobile Station
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.

SCIGPRSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET
STRING (SIZE (
    bound.&minSCIBillingChargingLength .. bound.&maxSCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
    CAMEL-SCIGPRSBillingChargingCharacteristics})
-- Indicates AOC information to be sent to a Mobile Station
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1
syntax error.

SendingSideID ::= CHOICE {sendingSideID [0] LegType}
-- used to identify LegID in operations sent from gsmSCF to gsmSSF

ServiceInteractionIndicatorsTwo ::= SEQUENCE {
    ForwardServiceInteractionInd      [0] ForwardServiceInteractionInd
        OPTIONAL,
    -- applicable to operations InitialDP, Connect and ContinueWithArgument.
    backwardServiceInteractionInd     [1] BackwardServiceInteractionInd
        OPTIONAL,
    -- applicable to operations Connect and ContinueWithArgument.

```

```

bothwayThroughConnectionInd          [2] BothwayThroughConnectionInd
        OPTIONAL,
-- applicable to ConnectToResource and EstablishTemporaryConnection
connectedNumberTreatmentInd          [4]
ConnectedNumberTreatmentInd          OPTIONAL,
-- applicable to Connect and ContinueWithArgument
nonCUGCall                            [13] NULL
        OPTIONAL,
-- applicable to Connect and ContinueWithArgument
-- indicates that no parameters for CUG shall be used for the call (i.e.
the call shall
-- be a non-CUG call).
-- If not present, it indicates one of three things:
-- a) continue with modified CUG information (when one or more of either
CUG Interlock Code
-- and Outgoing Access Indicator are present), or
-- b) continue with original CUG information (when neither CUG Interlock
Code or Outgoing
-- Access Indicator are present), i.e. no IN impact.
-- c) continue with the original non-CUG call.
holdTreatmentIndicator                [50] OCTET STRING (SIZE(1))
        OPTIONAL,
-- applicable to InitialDP, Connect and ContinueWithArgument
-- acceptHoldRequest 'xxxx xx01'B
-- rejectHoldRequest 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect call hold treatment
cwTreatmentIndicator                 [51] OCTET STRING (SIZE(1))
        OPTIONAL,
-- applicable to InitialDP, Connect and ContinueWithArgument
-- acceptCw 'xxxx xx01'B
-- rejectCw 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect call waiting treatment
ectTreatmentIndicator                [52] OCTET STRING (SIZE(1))
        OPTIONAL,
-- applicable to InitialDP, Connect and ContinueWithArgument
-- acceptEctRequest 'xxxx xx01'B
-- rejectEctRequest 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect explicit call transfer treatment
...
}

```

```
SGSNCapabilities ::= OCTET STRING (SIZE (1))
```

```

-- Indicates the SGSN capabilities. The coding of the parameter is as follows:
-- Bit      Value      Meaning
-- 0 0      0          AoC not supported by SGSN
--          1          AoC supported by SGSN
-- 1 -      -          This bit is reserved in CAP V.3
-- 2 -      -          This bit is reserved in CAP V.3
-- 3 -      -          This bit is reserved in CAP V.3
-- 4 -      -          This bit is reserved in CAP V.3
-- 5 -      -          This bit is reserved in CAP V.3
-- 6 -      -          This bit is reserved in CAP V.3
-- 7 -      -          This bit is reserved in CAP V.3

```

```

SMSEvent ::= SEQUENCE {
    eventTypeSMS          [0] EventTypeSMS,
    monitorMode          [1] MonitorMode
}

TariffSwitchInterval ::= INTEGER (1 .. 86400)
-- TariffSwitchInterval is measured in 1 second units

TimeAndTimezone {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minTimeAndTimezoneLength .. bound.&maxTimeAndTimezoneLength))
-- Indicates the time and timezone, relative to GMT. This parameter BCD
encoded.
-- The year digit indicating millenium occupies bits 0-3 of the first octet,
and the year
-- digit indicating century occupies bits 4-7 of the first octet.
-- The year digit indicating decade occupies bits 0-3 of the second octet,
whilst the digit
-- indicating the year within the decade occupies bits 4-7 of the second octet.
-- The most significant month digit occupies bits 0-3 of the third octet, and
the least
-- significant month digit occupies bits 4-7 of the third octet.
-- The most significant day digit occupies bits 0-3 of the fourth octet, and
the least
-- significant day digit occupies bits 4-7 of the fourth octet.
-- The most significant hours digit occupies bits 0-3 of the fifth octet, and
the least
-- significant hours digit occupies bits 4-7 of the fifth octet.
-- The most significant minutes digit occupies bits 0-3 of the sixth octet, and
the least
-- significant minutes digit occupies bits 4-7 of the sixth octet.
-- The most significant seconds digit occupies bits 0-3 of the seventh octet,
and the least
-- significant seconds digit occupies bits 4-7 of the seventh octet.
--
-- The timezone information occupies the eighth octet. For the encoding of
Timezone refer to
-- 3GPP TS 23.040 [6].
--
-- The BCD digits are packed and encoded as follows:
--
-- Bit      7 6 5 4 | 3 2 1 0
--      2nd digit | 1st digit      Octet 1
--      3rd digit | 4th digit      Octet 2
--      ..      | ..
--      nth digit | n-1th digit      Octet m
--
--      0000      digit 0
--      0001      digit 1
--      0010      digit 2
--      0011      digit 3
--      0100      digit 4
--      0101      digit 5
--      0110      digit 6
--      0111      digit 7
--      1000      digit 8
--      1001      digit 9

```



```

--      1010          spare
--      1011          spare
--      1100          spare
--      1101          spare
--      1110          spare
--      1101          spare
--
-- where the leftmost bit of the digit is either bit 7 or bit 3 of the octet.

TimeIfNoTariffSwitch ::= INTEGER(0..864000)
-- TimeIfNoTariffSwitch is measured in 100 millisecond intervals

TimeIfTariffSwitch ::= SEQUENCE {
    timeSinceTariffSwitch          [0] INTEGER(0..864000),
    tariffSwitchInterval          [1] INTEGER(1..864000)
                                OPTIONAL
}
-- timeSinceTariffSwitch and tariffSwitchInterval are measured in 100
millisecond intervals

TimeInformation ::= CHOICE {
    timeIfNoTariffSwitch          [0] TimeIfNoTariffSwitch,
    timeIfTariffSwitch           [1] TimeIfTariffSwitch
}
-- Indicates call duration information

TimerID ::= ENUMERATED {
    tssf                          (0)
}
-- Indicates the timer to be reset.

TimerValue ::= Integer4
-- Indicates the timer value (in seconds).

Tone ::= SEQUENCE {
    toneID                        [0] Integer4,
    duration                      [1] Integer4
                                OPTIONAL,
    ...
}
-- The duration specifies the length of the tone in seconds, value 0 indicates
infinite duration.

TPDataCodingScheme ::= OCTET STRING (SIZE (1))
-- TP Data Coding Scheme according to 3GPP TS 23.040 [6]

TPProtocolIdentifier ::= OCTET STRING (SIZE (1))
-- indicates the protocol used above the SM-Transfer Layer as specified in 3GPP
TS 23.040 [6].

TPShortMessageSpecificInfo ::= OCTET STRING (SIZE (1))
-- contains the 1st octet of the applicable TPDU or the SMS-COMMAND TPDU as
specified in
-- 3GPP TS 23.040 [6].

TPValidityPeriod ::= OCTET STRING (SIZE (1..7))

```

-- indicates the length of the validity period or the absolute time of the validity
-- period termination as specified in 3GPP TS 23.040 [6].
-- the length of ValidityPeriod is either 1 octet or 7 octets

```
TransferredVolume ::= CHOICE {
    volumeIfNoTariffSwitch          [0] INTEGER (0..4294967295),
    volumeIfTariffSwitch            [1] SEQUENCE {
        volumeSinceLastTariffSwitch [0]  INTEGER
(0..4294967295),
        volumeTariffSwitchInterval [1]  INTEGER
(0..4294967295)
    }
}
```

-- volumeIfNoTariffSwitch, volumeSinceLastTariffSwitch and
volumeTariffSwitchInterval
-- are measured in bytes.

```
TransferredVolumeRollOver ::= CHOICE {
    rO-VolumeIfNoTariffSwitch      [0] INTEGER (0..255),
    rO-VolumeIfTariffSwitch        [1] SEQUENCE {
        rO-VolumeSinceLastTariffSwitch [0]  INTEGER (0..255)
        OPTIONAL,
        rO-VolumeTariffSwitchInterval [1]  INTEGER (0..255)
        OPTIONAL
    }
}
```

-- rO-VolumeIfNoTariffSwitch, rO-VolumeSinceLastTariffSwitch and rO-
VolumeTariffSwitchInterval
-- present counters indicating the number of parameter range rollovers.

```
UnavailableNetworkResource ::= ENUMERATED {
    unavailableResources          (0),
    componentFailure              (1),
    basicCallProcessingException (2),
    resourceStatusFailure         (3),
    endUserFailure                (4)
}
```

-- Indicates the network resource that failed.

```
Units ::= INTEGER (1 .. 10000)
```

```
VariablePart {PARAMETERS-BOUND : bound} ::= CHOICE {
    integer          [0] Integer4,
    number           [1] Digits {bound},
    -- Generic digits
    time             [2] OCTET STRING
(SIZE(2)), -- HH: MM, BCD coded
    date             [3] OCTET STRING
(SIZE(4)), -- YYYYMMDD, BCD coded
    price            [4] OCTET STRING (SIZE(4))
}
```

-- Indicates the variable part of the message. Time is BCD encoded.
-- The most significant hours digit occupies bits 0-3 of the first octet, and
the least
-- significant digit occupies bits 4-7 of the first octet. The most significant
minutes digit

```

-- occupies bits 0-3 of the second octet, and the least significant digit
occupies bits 4-7
-- of the second octet.
--
-- Date is BCD encoded. The year digit indicating millenium occupies bits 0-3
of the first octet,
-- and the year digit indicating century occupies bits 4-7 of the first octet.
The year digit
-- indicating decade occupies bits 0-3 of the second octet, whilst the digit
indicating the year
-- within the decade occupies bits 4-7 of the second octet.
-- The most significant month digit occupies bits 0-3 of the third octet, and
the least
-- significant month digit occupies bits 4-7 of the third octet. The most
significant day digit
-- occupies bits 0-3 of the fourth octet, and the least significant day digit
occupies bits 4-7
-- of the fourth octet.
-- Price is BCD encoded. The digit indicating hundreds of thousands occupies
bits 0-3 of the
-- first octet, and the digit indicating tens of thousands occupies bits 4-7 of
the first octet.
-- The digit indicating thousands occupies bits 0-3 of the second octet, whilst
the digit
-- indicating hundreds occupies bits 4-7 of the second octet. The digit
indicating tens occupies
-- bits 0-3 of the third octet, and the digit indicating 0 to 9 occupies bits
4-7 of the third
-- octet. The tenths digit occupies bits 0-3 of the fourth octet, and the
hundredths digit
-- occupies bits 4-7 of the fourth octet.
--
-- For the encoding of digits in an octet, refer to the timeAndtimezone
parameter

```

```

-- The Definition of range of constants follows
numOfInfoItems INTEGER ::= 4

```

```

END

```

```

-- 5.2 Error types
CAP-errorTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-errorTypes(51) version4(3)}
-- This module contains the type definitions for the CAP Error Types.
-- Where a parameter of type CHOICE is tagged with a specific tag value, the tag
is automatically
-- replaced with an EXPLICIT tag of the same value.

```

```

DEFINITIONS IMPLICIT TAGS ::= BEGIN

```

```

IMPORTS

```

```

    ros-InformationObjects,
    datatypes,
    errorcodes
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)

```

```
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}
```

```
ERROR
```

```
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
InvokeID,  
UnavailableNetworkResource
```

```
FROM CAP-datatypes datatypes
```

```
errcode-canceled,  
errcode-cancelFailed,  
errcode-eTCFailed,  
errcode-improperCallerResponse,  
errcode-missingCustomerRecord,  
errcode-missingParameter,  
errcode-parameterOutOfRange,  
errcode-requestedInfoError,  
errcode-systemFailure,  
errcode-taskRefused,  
errcode-unavailableResource,  
errcode-unexpectedComponentSequence,  
errcode-unexpectedDataValue,  
errcode-unexpectedParameter,  
errcode-unknownLegID,  
errcode-unknownCSID,  
errcode-unknownPDPID
```

```
FROM CAP-errorcodes errorcodes
```

```
;
```

```
-- TYPE DEFINITION FOR CAP ERROR TYPES FOLLOWS
```

```
canceled ERROR ::= {  
    CODE errcode-canceled  
}
```

```
-- The operation has been canceled.
```

```
cancelFailed ERROR ::= {  
    PARAMETER SEQUENCE {  
        problem [0] ENUMERATED {  
            unknownOperation (0),  
            tooLate (1),  
            operationNotCancellable (2)  
        },  
        operation [1] InvokeID,  
        ...  
    }  
    CODE errcode-cancelFailed  
}
```

```
-- The operation failed to be canceled.
```

```
eTCFailed ERROR ::= {  
    CODE errcode-eTCFailed  
}
```

```
-- The establish temporary connection failed.
```

```
improperCallerResponse ERROR ::= {
```

```

        CODE errcode-improperCallerResponse
    }
-- The caller response was not as expected.

missingCustomerRecord ERROR ::= {
    CODE errcode-missingCustomerRecord
}
-- The Service Logic Program could not be found in the gsmSCF.

missingParameter ERROR ::= {
    CODE errcode-missingParameter
}
-- An expected optional parameter was not received.

parameterOutOfRange ERROR ::= {
    CODE errcode-parameterOutOfRange
}
-- The parameter was not as expected (e.g. missing or out of range).

requestedInfoError ERROR ::= {
    PARAMETER ENUMERATED {
        unknownRequestedInfo          (1),
        requestedInfoNotAvailable      (2)
    }
    CODE errcode-requestedInfoError
}
-- The requested information cannot be found.

systemFailure ERROR ::= {
    PARAMETER UnavailableNetworkResource
    CODE errcode-systemFailure
}
-- The operation could not be completed due to a system failure at the serving
physical entity.

taskRefused ERROR ::= {
    PARAMETER ENUMERATED {
        generic                        (0),
        unobtainable                    (1),
        congestion                       (2)
    }
    CODE errcode-taskRefused
}
-- An entity normally capable of the task requested cannot or chooses not to
perform the task at
-- this time. This includes error situations like congestion and unobtainable
address as used in
-- e.g. the connect operation.)

unavailableResource ERROR ::= {
    CODE errcode-unavailableResource
}
-- A requested resource is not available at the serving entity.

unexpectedComponentSequence ERROR ::= {
    CODE errcode-unexpectedComponentSequence
}

```

```

-- An incorrect sequence of Components was received
(e.g. "DisconnectForwardConnection"
-- followed by "PlayAnnouncement").

unexpectedDataValue ERROR ::= {
    CODE errcode-unexpectedDataValue
}
-- The data value was not as expected (e.g. route number expected but billing
number received)

unexpectedParameter ERROR ::= {
    CODE errcode-unexpectedParameter
}
-- A parameter received was not expected.

unknownLegID ERROR ::= {
    CODE errcode-unknownLegID
}
-- Leg not known to the gsmSSF.

unknownCSID ERROR ::= {
    CODE errcode-unknownCSID
}
-- Call Segment not known to the gsmSSF.

unknownPDPID ERROR ::= {
    CODE errcode-unknownPDPID
}
-- PDPID not known by the receiving entity.

END

-- 5.3      Operation codes
CAP-operationcodes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-operationcodes(53) version4(3)}

DEFINITIONS ::= BEGIN

IMPORTS

    ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

    Code
FROM Remote-Operations-Information-Objects ros-InformationObjects

;

-- the operations are grouped by the identified operation packages.

-- gsmSCF activation Package
opcode-initialDP                               Code ::= local: 0
-- gsmSCF/gsmSRF activation of assist Package
opcode-assistRequestInstructions               Code ::= local: 16
-- Assist connection establishment Package

```

```

opcode-establishTemporaryConnection           Code ::= local: 17
-- Generic disconnect resource Package
opcode-disconnectForwardConnection           Code ::= local: 18
opcode-dFCWithArgument                       Code ::= local: 86
-- Non-assisted connection establishment Package

opcode-connectToResource                     Code ::= local: 19
-- Connect Package (elementary gsmSSF function)
opcode-connect                               Code ::=
local: 20
-- Call handling Package (elementary gsmSSF function)
opcode-releaseCall                           Code ::=
local: 22
-- BCSM Event handling Package
opcode-requestReportBCSMEvent                Code ::= local: 23
opcode-eventReportBCSM                      Code ::= local: 24
-- Charging Event handling Package
opcode-requestNotificationChargingEvent      Code ::= local: 25
opcode-eventNotificationCharging            Code ::= local: 26
-- gsmSSF call processing Package
opcode-continue                              Code ::=
local: 31
opcode-continueWithArgument                  Code ::= local: 56
-- gsmSCF call initiation Package
opcode-initiateCallAttempt                   Code ::= local: 32
-- Timer Package
opcode-resetTimer                           Code ::= local: 33
-- Billing Package
opcode-furnishChargingInformation            Code ::= local: 34
-- Charging Package
opcode-applyCharging                         Code ::= local: 35
opcode-applyChargingReport                  Code ::= local: 36
opcode-playTone                              Code ::=
local: 97
-- Traffic management Package
opcode-callGap                               Code ::=
local: 41
-- Call report Package
opcode-callInformationReport                 Code ::= local: 44
opcode-callInformationRequest                Code ::= local: 45
-- Signalling control Package
opcode-sendChargingInformation               Code ::= local: 46
-- Specialized resource control Package
opcode-playAnnouncement                      Code ::= local: 47
opcode-promptAndCollectUserInformation       Code ::= local: 48
opcode-specializedResourceReport            Code ::= local: 49
-- Cancel Package
opcode-cancel                                Code ::=
local: 53
-- Activity Test Package
opcode-activityTest                          Code ::=
local: 55
-- CPH Response Package
opcode-continueWithArgument                  Code ::= local: 88
opcode-disconnectLeg                         Code ::= local: 90
opcode-moveLeg                               Code ::=
local: 93

```

```

        opcode-splitLeg                                Code ::=
local: 95
-- Exception Inform Package
        opcode-entityReleased                        Code ::= local: 96

-- Sms Activation Package
        opcode-initialDPSMS                          Code ::=
local: 60
-- Sms Billing Package
        opcode-furnishChargingInformationSMS          Code ::= local: 61
-- Sms Connect Package
        opcode-connectSMS                            Code ::= local: 62
-- Sms Event Handling Package
        opcode-requestReportSMSEvent                 Code ::= local: 63
        opcode-eventReportSMS                        Code ::= local: 64
-- Sms Processing Package
        opcode-continueSMS                            Code ::=
local: 65
-- Sms Release Package
        opcode-releaseSMS                             Code ::= local: 66
-- Sms Timer Package
        opcode-resetTimerSMS                          Code ::= local: 67

-- Gprs Activity Test Package
        opcode-activityTestGPRS                       Code ::= local: 70
-- Gprs Charging Package
        opcode-applyChargingGPRS                      Code ::= local: 71
        opcode-applyChargingReportGPRS                Code ::= local: 72
-- Gprs Cancel Package
        opcode-cancelGPRS                             Code ::= local: 73
-- Gprs Connect Package
        opcode-connectGPRS                            Code ::=
local: 74
-- Gprs Processing Package
        opcode-continueGPRS                           Code ::=
local: 75
-- Gprs Exception Information Package
        opcode-entityReleasedGPRS                     Code ::= local: 76
-- Gprs Billing Package
        opcode-furnishChargingInformationGPRS          Code ::= local: 77
-- Gprs Scf Activation Package
        opcode-initialDPGPRS                           Code ::= local: 78
-- Gprs Release Package
        opcode-releaseGPRS                             Code ::=
local: 79
-- Gprs Event Handling Package
        opcode-eventReportGPRS                        Code ::= local: 80
        opcode-requestReportGPRSEvent                 Code ::= local: 81
-- Gprs Timer Package
        opcode-resetTimerGPRS                          Code ::= local: 82
-- Gprs Charge Advice Package
        opcode-sendChargingInformationGPRS             Code ::= local: 83

```

END

-- 5.4 Error codes


```
CAP-errorcodes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-errorcodes(57) version4(3)}
```

```
DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}
```

```
Code
```

```
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
;
```

```
errcode-canceled                               Code ::= local: 0
errcode-cancelFailed                           Code ::= local: 1
errcode-eTCFailed                              Code ::= local: 3
errcode-improperCallerResponse                 Code ::= local: 4
errcode-missingCustomerRecord                  Code ::= local: 6
errcode-missingParameter                       Code ::= local: 7
errcode-parameterOutOfRange                    Code ::= local: 8
errcode-requestedInfoError                     Code ::= local: 10
errcode-systemFailure                          Code ::= local: 11
errcode-taskRefused                            Code ::=
local: 12
errcode-unavailableResource                     Code ::= local: 13
errcode-unexpectedComponentSequence            Code ::= local: 14
errcode-unexpectedDataValue                    Code ::= local: 15
errcode-unexpectedParameter                    Code ::= local: 16
errcode-unknownLegID                           Code ::= local: 17
errcode-unknownPDPID                           Code ::= local: 50
errcode-unknownCSID                            Code ::=
local: 51
```

```
END
```

```
-- 5.5      Classes
```

```
CAP-classes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1)
modules(3) cap-classes(54) version4(3)}
```

```
DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    ROS-OBJECT-CLASS,
Code
```

```
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
id-rosObject-gsmSRF,
id-rosObject-gsmSSF,
ros-InformationObjects,
gsmSSF-gsmSCF-Protocol,
gsmSCF-gsmSRF-Protocol
```

```

FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

    capSsfToScfGeneric,
    capAssistHandoffssfToScf
FROM CAP-gsmSSF-gsmSCF-pkgs-contracts-acs gsmSSF-gsmSCF-Protocol

    gsmSRF-gsmSCF-contract
FROM CAP-gsmSCF-gsmSRF-pkgs-contracts-acs gsmSCF-gsmSRF-Protocol

    CriticalityType
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1) cs2(20)
modules(0) in-cs2-datatypes(0) version1(0)}

;

gsmSSF ROS-OBJECT-CLASS ::= {
    INITIATES    {capSsfToScfGeneric|
                  capAssistHandoffssfToScf}
    RESPONDS    {capSsfToScfGeneric}
    ID          id-rosObject-gsmSSF}

gsmSRF ROS-OBJECT-CLASS ::= {
    INITIATES    {gsmSRF-gsmSCF-contract}
    ID          id-rosObject-gsmSRF}

EXTENSION ::= CLASS {
    &ExtensionType,
    &criticality          CriticalityType DEFAULT ignore,
    &id    Code}

WITH SYNTAX {
    EXTENSION-SYNTAX    &ExtensionType
    CRITICALITY        &criticality
    IDENTIFIED BY      &id}
-- Example of addition of an extension named "Some Network Specific Indicator"
of type
-- BOOLEAN, with criticality "abort" and to be identified as extension number 1
-- Example of definition using the above information object class:
--
-- SomeNetworkSpecificIndicator EXTENSION ::= {
--     EXTENSION-SYNTAX    BOOLEAN
--     CRITICALITY        abort
--     IDENTIFIED BY      local: 1
-- }

-- Example of transfer syntax, using the ExtensionField datatype as specified in
clause 5.
-- Assuming the value of the extension is set to TRUE, the extensions parameter
-- becomes a Sequence of type INTEGER ::= 1, criticality ENUMERATED ::= 1 and
value [1]
-- EXPLICIT BOOLEAN ::= TRUE.
--
-- Use of ITU-T Recommendation Q.1400 [52] defined Extension is for further
study.

```

-- In addition the extension mechanism marker is used to identify the future minor additions
-- to CAP.

```
firstExtension EXTENSION ::= {  
    EXTENSION-SYNTAX NULL  
    CRITICALITY          ignore  
    IDENTIFIED BY        local: 1}  
-- firstExtension is just an example.
```

```
SupportedExtensions EXTENSION ::= {firstExtension, ...  
-- full set of network operator extensions --  
}  
-- SupportedExtension is the full set of the network operator extensions.
```

```
PARAMETERS-BOUND ::= CLASS {  
    &minAccessPointNameLength          INTEGER,  
    &maxAccessPointNameLength          INTEGER,  
    &minAChBillingChargingLength       INTEGER,  
    &maxAChBillingChargingLength       INTEGER,  
    &minAttributesLength                INTEGER,  
    &maxAttributesLength                INTEGER,  
    &maxBearerCapabilityLength          INTEGER,  
    &minCalledPartyBCDNumberLength     INTEGER,  
    &maxCalledPartyBCDNumberLength     INTEGER,  
    &minCalledPartyNumberLength        INTEGER,  
    &maxCalledPartyNumberLength        INTEGER,  
    &minCallingPartyNumberLength       INTEGER,  
    &maxCallingPartyNumberLength       INTEGER,  
    &minCallResultLength                INTEGER,  
    &maxCallResultLength                INTEGER,  
    &minCarrierLength                  INTEGER,  
    &maxCarrierLength                  INTEGER,  
    &minCauseLength                    INTEGER,  
    &maxCauseLength                    INTEGER,  
    &minComponentLength                INTEGER,  
    &maxComponentLength                INTEGER,  
    &minDigitsLength                    INTEGER,  
    &maxDigitsLength                    INTEGER,  
    &minFCIBillingChargingDataLength   INTEGER,  
    &maxFCIBillingChargingDataLength   INTEGER,  
    &minFCIBillingChargingLength       INTEGER,  
    &maxFCIBillingChargingLength       INTEGER,  
    &minGenericNumberLength            INTEGER,  
    &maxGenericNumberLength            INTEGER,  
    &minGPRSCauseLength                 INTEGER,  
    &maxGPRSCauseLength                 INTEGER,  
    &minIPSSPCapabilitiesLength        INTEGER,  
    &maxIPSSPCapabilitiesLength        INTEGER,  
    &minLocationNumberLength           INTEGER,  
    &maxLocationNumberLength           INTEGER,  
    &minMessageContentLength           INTEGER,  
    &maxMessageContentLength           INTEGER,  
    &minOriginalCalledPartyIDLength    INTEGER,  
    &maxOriginalCalledPartyIDLength    INTEGER,  
    &minPDPAddressLength                INTEGER,  
    &maxPDPAddressLength                INTEGER,
```

&minRedirectingPartyIDLength	INTEGER,
&maxRedirectingPartyIDLength	INTEGER,
&minScfIDLength	INTEGER,
&maxScfIDLength	INTEGER,
&minSCIBillingChargingLength	INTEGER,
&maxSCIBillingChargingLength	INTEGER,
&minTimeAndTimezoneLength	INTEGER,
&maxTimeAndTimezoneLength	INTEGER,
&numOfBCSMEvents	INTEGER,
&numOfSMSEvents	INTEGER,
&numOfGPRSEvents	INTEGER,
&numOfExtensions	INTEGER,
&numOfGenericNumbers	INTEGER,
&numOfMessageIDs	INTEGER }

WITH SYNTAX {

MINIMUM-FOR-ACCESS-POINT-NAME	&minAccessPointNameLength
MAXIMUM-FOR-ACCESS-POINT-NAME	&maxAccessPointNameLength
MINIMUM-FOR-ACH-BILLING-CHARGING	
&minAchBillingChargingLength	
MAXIMUM-FOR-ACH-BILLING-CHARGING	
&maxAchBillingChargingLength	
MINIMUM-FOR-ATTRIBUTES	&minAttributesLength
MAXIMUM-FOR-ATTRIBUTES	&maxAttributesLength
MAXIMUM-FOR-BEARER-CAPABILITY	&maxBearerCapabilityLength
MINIMUM-FOR-CALLED-PARTY-BCD-NUMBER	
&minCalledPartyBCDNumberLength	
MAXIMUM-FOR-CALLED-PARTY-BCD-NUMBER	
&maxCalledPartyBCDNumberLength	
MINIMUM-FOR-CALLED-PARTY-NUMBER	
&minCalledPartyNumberLength	
MAXIMUM-FOR-CALLED-PARTY-NUMBER	
&maxCalledPartyNumberLength	
MINIMUM-FOR-CALLING-PARTY-NUMBER	
&minCallingPartyNumberLength	
MAXIMUM-FOR-CALLING-PARTY-NUMBER	
&maxCallingPartyNumberLength	
MINIMUM-FOR-CALL-RESULT	&minCallResultLength
MAXIMUM-FOR-CALL-RESULT	&maxCallResultLength
MINIMUM-FOR-CARRIER	
&minCarrierLength	
MAXIMUM-FOR-CARRIER	
&maxCarrierLength	
MINIMUM-FOR-CAUSE	&minCauseLength
MAXIMUM-FOR-CAUSE	&maxCauseLength
MINIMUM-FOR-COMPONENT	&minComponentLength
MAXIMUM-FOR-COMPONENT	&maxComponentLength
MINIMUM-FOR-DIGITS	
&minDigitsLength	
MAXIMUM-FOR-DIGITS	
&maxDigitsLength	
MINIMUM-FOR-FCI-BILLING-CHARGING-DATA	
&minFCIBillingChargingDataLength	
MAXIMUM-FOR-FCI-BILLING-CHARGING-DATA	
&maxFCIBillingChargingDataLength	
MINIMUM-FOR-FCI-BILLING-CHARGING	
&minFCIBillingChargingLength	

MAXIMUM-FOR-FCI-BILLING-CHARGING	
&maxFCIBillingChargingLength	
MINIMUM-FOR-GENERIC-NUMBER	
&minGenericNumberLength	
MAXIMUM-FOR-GENERIC-NUMBER	
&maxGenericNumberLength	
MINIMUM-FOR-GPRS-CAUSE-LENGTH	&minGPRSCauseLength
MAXIMUM-FOR-GPRS-CAUSE-LENGTH	&maxGPRSCauseLength
MINIMUM-FOR-IP-SSP-CAPABILITIES	
&minIPSSPCapabilitiesLength	
MAXIMUM-FOR-IP-SSP-CAPABILITIES	
&maxIPSSPCapabilitiesLength	
MINIMUM-FOR-LOCATION-NUMBER	
&minLocationNumberLength	
MAXIMUM-FOR-LOCATION-NUMBER	
&maxLocationNumberLength	
MINIMUM-FOR-MESSAGE-CONTENT	
&minMessageContentLength	
MAXIMUM-FOR-MESSAGE-CONTENT	
&maxMessageContentLength	
MINIMUM-FOR-ORIGINAL-CALLED-PARTY-ID	
&minOriginalCalledPartyIDLength	
MAXIMUM-FOR-ORIGINAL-CALLED-PARTY-ID	
&maxOriginalCalledPartyIDLength	
MINIMUM-FOR-PDP-ADDRESS-LENGTH	&minPDPAddressLength
MAXIMUM-FOR-PDP-ADDRESS-LENGTH	&maxPDPAddressLength
MINIMUM-FOR-REDIRECTING-ID	
&minRedirectingPartyIDLength	
MAXIMUM-FOR-REDIRECTING-ID	
&maxRedirectingPartyIDLength	
MINIMUM-FOR-GSMSCF-ID	&minScfIDLength
MAXIMUM-FOR-GSMSCF-ID	&maxScfIDLength
MINIMUM-FOR-SCI-BILLING-CHARGING	
&minSCIBillingChargingLength	
MAXIMUM-FOR-SCI-BILLING-CHARGING	
&maxSCIBillingChargingLength	
MINIMUM-FOR-TIME-AND-TIMEZONE	&minTimeAndTimezoneLength
MAXIMUM-FOR-TIME-AND-TIMEZONE	&maxTimeAndTimezoneLength
NUM-OF-BCSM-EVENT	&numOfBCSMEvents
NUM-OF-SMS-EVENTS	&numOfSMSEvents
NUM-OF-GPRS-EVENTS	
&numOfGPRSEvents	
NUM-OF-EXTENSIONS	&numOfExtensions
NUM-OF-GENERIC-NUMBERS	&numOfGenericNumbers
NUM-OF-MESSAGE-IDS	
&numOfMessageIDs	

```

cAPSpecificBoundSet PARAMETERS-BOUND ::= {
    MINIMUM-FOR-ACCESS-POINT-NAME          1
    MAXIMUM-FOR-ACCESS-POINT-NAME          100
    MINIMUM-FOR-ACH-BILLING-CHARGING        5
    MAXIMUM-FOR-ACH-BILLING-CHARGING        177
    MINIMUM-FOR-ATTRIBUTES                   2
    MAXIMUM-FOR-ATTRIBUTES                   10
    MAXIMUM-FOR-BEARER-CAPABILITY           11
    MINIMUM-FOR-CALLED-PARTY-BCD-NUMBER     1
    MAXIMUM-FOR-CALLED-PARTY-BCD-NUMBER     41

```

MINIMUM-FOR-CALLED-PARTY-NUMBER		2	
MAXIMUM-FOR-CALLED-PARTY-NUMBER		18	
MINIMUM-FOR-CALLING-PARTY-NUMBER	2		
MAXIMUM-FOR-CALLING-PARTY-NUMBER	10		
MINIMUM-FOR-CALL-RESULT		12	
MAXIMUM-FOR-CALL-RESULT		193	
MINIMUM-FOR-CARRIER			4
MAXIMUM-FOR-CARRIER			4
MINIMUM-FOR-CAUSE		2	
MAXIMUM-FOR-CAUSE		32	
MINIMUM-FOR-COMPONENT		1	
MAXIMUM-FOR-COMPONENT		2	
MINIMUM-FOR-DIGITS			2
MAXIMUM-FOR-DIGITS			16
MINIMUM-FOR-FCI-BILLING-CHARGING-DATA	1		
MAXIMUM-FOR-FCI-BILLING-CHARGING-DATA	160		
MINIMUM-FOR-FCI-BILLING-CHARGING	5		
MAXIMUM-FOR-FCI-BILLING-CHARGING	225		
MINIMUM-FOR-GENERIC-NUMBER		3	
MAXIMUM-FOR-GENERIC-NUMBER		11	
MINIMUM-FOR-GPRS-CAUSE-LENGTH	1		
MAXIMUM-FOR-GPRS-CAUSE-LENGTH	1		
MINIMUM-FOR-IP-SSP-CAPABILITIES		1	
MAXIMUM-FOR-IP-SSP-CAPABILITIES		4	
MINIMUM-FOR-LOCATION-NUMBER		2	
MAXIMUM-FOR-LOCATION-NUMBER		10	
MINIMUM-FOR-MESSAGE-CONTENT		1	
MAXIMUM-FOR-MESSAGE-CONTENT		127	
MINIMUM-FOR-ORIGINAL-CALLED-PARTY-ID	2		
MAXIMUM-FOR-ORIGINAL-CALLED-PARTY-ID	10		
MINIMUM-FOR-PDP-ADDRESS-LENGTH		1	
MAXIMUM-FOR-PDP-ADDRESS-LENGTH		63	
MINIMUM-FOR-REDIRECTING-ID		2	
MAXIMUM-FOR-REDIRECTING-ID		10	
MINIMUM-FOR-GSMSCF-ID		2	
MAXIMUM-FOR-GSMSCF-ID		10	
MINIMUM-FOR-SCI-BILLING-CHARGING	4		
MAXIMUM-FOR-SCI-BILLING-CHARGING	124		
MINIMUM-FOR-TIME-AND-TIMEZONE	8		
MAXIMUM-FOR-TIME-AND-TIMEZONE	8		
NUM-OF-BCSM-EVENT		10	
NUM-OF-SMS-EVENTS		10	
NUM-OF-GPRS-EVENTS			10
NUM-OF-EXTENSIONS		10	
NUM-OF-GENERIC-NUMBERS		5	
NUM-OF-MESSAGE-IDS			16}

END

-- 5.6 Object Identifiers (IDs)

```
CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}
```

DEFINITIONS ::= BEGIN

```

-- This module assigns object identifiers for Modules, Packages, Contracts and
ACs
-- used by CAP

-- For Modules from TC, ROS,
tc-Messages OBJECT
IDENTIFIER ::=
    {itu-t recommendation q 773 modules(2) messages(1) version3(3)}

tc-NotationExtensions OBJECT IDENTIFIER
::=
    {itu-t recommendation q 775 modules(2) notation-extension (4) version1(1)}

ros-InformationObjects OBJECT IDENTIFIER
::=
    {joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)}

-- For CAP Modules
datatypes OBJECT
IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
    cap-datatypes(52) version3(2)}

errortypes OBJECT
IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
    cap-errortypes(51) version3(2)}

operationcodes OBJECT
IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
    cap-operationcodes(53) version3(2)}

errorcodes OBJECT
IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
    cap-errorcodes(57) version3(2)}

classes OBJECT
IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
    cap-classes(54) version3(2)}

gsmSSF-gsmSCF-Operations OBJECT IDENTIFIER
::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
    cap-gsmSSF-gsmSCF-ops-args(101) version3(2)}

```

```

gsmSSF-gsmSCF-Protocol                                OBJECT IDENTIFIER
 ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-gsmSSF-gsmSCF-pkgs-contracts-acs(102) version3(2)}

gsmSCF-gsmSRF-Operations                              OBJECT IDENTIFIER
 ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-gsmSCF-gsmSRF-ops-args(103) version3(2)}

gsmSCF-gsmSRF-Protocol                                OBJECT IDENTIFIER
 ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-gsmSCF-gsmSRF-pkgs-contracts-acs(104) version3(2)}

sms-Operations                                        OBJECT
IDENTIFIER ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-SMS-ops-args(105) version4(3)}

smsSSF-gsmSCF-Protocol                                OBJECT IDENTIFIER
 ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-smsSSF-gsmSCF-pkgs-contracts-acs(106) version4(3)}

gprsSSF-gsmSCF-Operations                              OBJECT IDENTIFIER
 ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-GPRS-ops-args(107) version3(2)}

gprsSSF-gsmSCF-Protocol                                OBJECT IDENTIFIER
 ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-
network(1) modules(3)
   cap-gprsSSF-gsmSCF-pkgs-contracts-acs(108) version3(2)}

id-CAP                                                OBJECT
IDENTIFIER ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
   umts-network(1) cap4(22)}

id-CAP3                                              OBJECT
IDENTIFIER ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
   umts-network(1) cap3(20)}

id-CAPOE                                             OBJECT
IDENTIFIER ::=
   {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)

```



```

    umts-network(1) cap40E(23)}

id-CAP3OE                                OBJECT
IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
    umts-network(1) cap30E(21)}

id-ac                                    OBJECT
IDENTIFIER ::= {id-CAP          ac(3)}
id-ac3                                  OBJECT
IDENTIFIER ::= {id-CAP3        ac(3)}
id-acE                                  OBJECT
IDENTIFIER ::= {id-CAPOE       ac(3)}
id-ac3E                                  OBJECT
IDENTIFIER ::= {id-CAP30E      ac(3)}
id-as                                    OBJECT
IDENTIFIER ::= {id-CAP          as(5)}
id-asE                                  OBJECT
IDENTIFIER ::= {id-CAPOE       as(5)}
id-rosObject                            OBJECT
IDENTIFIER ::= {id-CAP          rosObject(25)}
id-contract                             OBJECT
IDENTIFIER ::= {id-CAP          contract(26)}
id-contract3                            OBJECT
IDENTIFIER ::= {id-CAP3        contract(26)}
id-contractE                             OBJECT
IDENTIFIER ::= {id-CAPOE       contract(26)}
id-package                               OBJECT
IDENTIFIER ::= {id-CAP          package(27)}
id-packageE                              OBJECT
IDENTIFIER ::= {id-CAPOE       package(27)}

-- for ac, as, rosObject, contract and package, the values are identical to ITU-
T Recommendation
-- Q.1218 [49]

-- ROS Objects

id-rosObject-gsmSCF                      OBJECT
IDENTIFIER ::= {id-rosObject 4}
id-rosObject-gsmSSF                      OBJECT
IDENTIFIER ::= {id-rosObject 5}
id-rosObject-gsmSRF                      OBJECT
IDENTIFIER ::= {id-rosObject 6}

-- Application Contexts

-- gsmSSF/gsmSCF AC
id-ac-CAP-gsmSSF-scfGenericAC            OBJECT IDENTIFIER ::= {id-
acE 4}
id-ac-CAP-gsmSSF-scfAssistHandoffAC      OBJECT IDENTIFIER ::= {id-
acE 6}
id-ac-CAP-scf-gsmSSFGenericAC            OBJECT IDENTIFIER ::= {id-
acE 8}

-- gsmSRF/gsmSCF AC

```

```

id-ac-gsmSRF-gsmSCF                                OBJECT
IDENTIFIER ::= {id-ac 14}

-- gprsSSF/gsmSCF AC
id-ac-CAP-gprsSSF-gsmSCF-AC                        OBJECT IDENTIFIER
::= {id-acE 50}
id-ac-CAP-gsmSCF-gprsSSF-AC                        OBJECT IDENTIFIER
::= {id-acE 51}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF AC
id-ac-cap3-sms-AC                                  OBJECT IDENTIFIER
::= {id-ac3E 61}
id-ac-cap4-sms-AC                                  OBJECT IDENTIFIER
::= {id-acE 61}

-- gsmSSF/gsmSCF Contracts
id-CAPSsfToScfGeneric                              OBJECT IDENTIFIER
::= {id-contractE 3}
id-CAPAssistHandoffssfToScf                       OBJECT IDENTIFIER
::= {id-contractE 5}
id-CAPScfToSsfGeneric                              OBJECT IDENTIFIER
::= {id-contractE 6}

-- gsmSRF/gsmSCF Contracts
id-contract-gsmSRF-gsmSCF                          OBJECT IDENTIFIER
::= {id-contract 13}

-- gprsSSF/gsmSCF Contracts
id-cap3GprsSsfTogsmScf                             OBJECT IDENTIFIER
::= {id-contract 14}
id-cap3GsmScfToGprsSsf                             OBJECT IDENTIFIER
::= {id-contract 15}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF Contracts
id-cap3SmsSsfTogsmScf                              OBJECT IDENTIFIER
::= {id-contract3 16}
id-cap4SmsSsfTogsmScf                              OBJECT IDENTIFIER
::= {id-contract 16}

-- gsmSSF/gsmSCF Operation Packages
id-package-scfActivation                            OBJECT IDENTIFIER
::= {id-package 11}
id-package-gsmSRF-scfActivationOfAssist            OBJECT IDENTIFIER ::= {id-
package 15}
id-package-assistConnectionEstablishment           OBJECT IDENTIFIER ::= {id-
package 16}
id-package-genericDisconnectResource               OBJECT IDENTIFIER ::= {id-
package 17}
id-package-nonAssistedConnectionEstablishment      OBJECT IDENTIFIER ::= {id-
package 18}
id-package-connect                                  OBJECT
IDENTIFIER ::= {id-package 19}
id-package-callHandling                             OBJECT IDENTIFIER
::= {id-packageE 20}
id-package-bcsmEventHandling                        OBJECT IDENTIFIER ::= {id-
package 21}

```

id-package-chargingEventHandling package 23}	OBJECT IDENTIFIER ::= {id-
id-package-ssfCallProcessing packageE 24}	OBJECT IDENTIFIER ::= {id-
id-package-scfCallInitiation package 25}	OBJECT IDENTIFIER ::= {id-
id-package-timer ::= {id-package 26}	OBJECT IDENTIFIER
id-package-billing IDENTIFIER ::= {id-package 27}	OBJECT
id-package-charging IDENTIFIER ::= {id-package 28}	OBJECT
id-package-trafficManagement package 29}	OBJECT IDENTIFIER ::= {id-
id-package-callReport ::= {id-package 32}	OBJECT IDENTIFIER
id-package-signallingControl package 33}	OBJECT IDENTIFIER ::= {id-
id-package-activityTest ::= {id-package 34}	OBJECT IDENTIFIER
id-package-cancel ::= {id-packageE 36}	OBJECT IDENTIFIER
id-package-cphResponse ::= {id-package 37}	OBJECT IDENTIFIER
id-package-exceptionInform ::= {id-package 38}	OBJECT IDENTIFIER
-- gsmSRF/gsmSCF Operation Packages	
id-package-specializedResourceControl package 42}	OBJECT IDENTIFIER ::= {id-
id-package-gsmSRF-scfCancel ::= {id-package 43}	OBJECT IDENTIFIER
-- gprsSSF/gsmSCF Operation Packages	
id-package-gprsContinue ::= {id-package 49}	OBJECT IDENTIFIER
id-package-gprsExceptionInformation package 50}	OBJECT IDENTIFIER ::= {id-
id-package-gprsSCFActivation package 51}	OBJECT IDENTIFIER ::= {id-
id-package-gprsConnect ::= {id-package 52}	OBJECT IDENTIFIER
id-package-gprsRelease ::= {id-package 53}	OBJECT IDENTIFIER
id-package-gprsEventHandling package 54}	OBJECT IDENTIFIER ::= {id-
id-package-gprsSCFTimer ::= {id-package 55}	OBJECT IDENTIFIER
id-package-gprsSCFBilling ::= {id-package 56}	OBJECT IDENTIFIER
id-package-gprsSCFCharging ::= {id-package 57}	OBJECT IDENTIFIER
id-package-gprsSCFActivityTest ::= {id-package 58}	OBJECT IDENTIFIER
id-package-gprsSCFCancel ::= {id-package 59}	OBJECT IDENTIFIER

```

id-package-gprsSCFChargeAdvice OBJECT IDENTIFIER
 ::= {id-package 60}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF Operation Packages
id-package-smsActivation OBJECT IDENTIFIER
 ::= {id-package 61}
id-package-smsConnect OBJECT IDENTIFIER
 ::= {id-package 62}
id-package-smsContinue OBJECT IDENTIFIER
 ::= {id-package 63}
id-package-smsRelease OBJECT IDENTIFIER
 ::= {id-package 64}
id-package-smsEventHandling OBJECT IDENTIFIER
 ::= {id-package 65}
id-package-smsBilling OBJECT IDENTIFIER
 ::= {id-package 66}
id-package-smsTimer OBJECT IDENTIFIER ::= {id-package 67}

-- gsmSSF/gsmSCF Abstract Syntaxes
id-as-gsmSSF-scfGenericAS OBJECT IDENTIFIER
 ::= {id-asE 4}
id-as-assistHandoff-gsmSSF-scfAS OBJECT IDENTIFIER ::= {id-
asE 6}
id-as-scf-gsmSSFGenericAS OBJECT IDENTIFIER
 ::= {id-asE 7}

-- gsmSRF/gsmSCF Abstract Syntaxes
id-as-basic-gsmSRF-gsmSCF OBJECT IDENTIFIER
 ::= {id-as 14}

-- gprsSSF/gsmSCF Abstract Syntaxes
id-as-gprsSSF-gsmSCF-AS OBJECT IDENTIFIER
 ::= {id-as 50}
id-as-gsmSCF-gprsSSF-AS OBJECT IDENTIFIER
 ::= {id-as 51}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF Abstract Syntaxes
id-as-smsSSF-gsmSCF-AS OBJECT IDENTIFIER
 ::= {id-as 61}

END

-- 5.7 User Abort Data
CAP-U-ABORT-Data {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-u-abort-data(110) version3(2)}

DEFINITIONS ::= BEGIN

id-CAP-U-ABORT-Reason OBJECT IDENTIFIER ::= {itu-t(0) identified-organization(4)
etsi(0) mobileDomain(0) umts-Network(1) as(1) cap-u-abort-reason(2) version3(2)}

cap-U-ABORT-Reason-Abstract-Syntax ABSTRACT-SYNTAX ::= {CAP-U-ABORT-REASON
IDENTIFIED BY
id-CAP-U-ABORT-Reason}

CAP-U-ABORT-REASON ::= ENUMERATED {

```

```

        no-reason-given                (1),
        application-timer-expired      (2),
        not-allowed-procedures        (3),
        abnormal-processing            (4),
        congestion                    (5),
        invalid-reference              (6),
        missing-reference              (7),
        overlapping-dialogue          (8)
    }
-- application-timer-expired shall be set when application timer (e.g. Tssf) is
expired.
-- not-allowed-procedures shall be set when received signal is not
allowed in CAP
--
--
-- operation is received from the
--
-- allowed in gsmSSF FSM.
--
-- state transition). (e.g. ReleaseCall
--
-- for End of Temporary Connection
--
--
-- abnormal-processing shall be set when abnormal procedures occur
at entity action.
-- congestion shall be set when requested resource
is unavailable due to
--
-- level.
-- invalid-reference shall be set if the received
destinationReference is unknown or
--
-- Reference the received originationReference
--
-- originationReference.
--
-- CAP defined GPRS-ReferenceNumber.
-- missing-reference shall be set when the destinationReference
or the
--
-- originationReference is absent
in the received message but is
--
-- required to be present according
to the procedures in
--
-- subclause 14.1.7.
CAP defined GPRS-ReferenceNumber.
-- overlapping-dialogue shall be used by the gprsSSF to indicate to
the gsmSCF that a
--
-- specific instance already has a
TC dialogue open. This error
--
-- cause is typically obtained when
both the gsmSCF and gprsSSF
--
-- open a new dialogue at the same
time.
-- no-reason-given shall be set when any other reasons
above do not apply
END -- of CAP-U-ABORT-Data

```

```

-- 6 Circuit Switched Call Control
-- 6.1      gsmSSF/CCF - gsmSCF Interface
-- 6.1.1    Operations and arguments
CAP-gsmSSF-gsmSCF-ops-args {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-gsmSSF-gsmSCF-ops-args(101) version4(3)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- This module contains the operations and operation arguments used for the
-- gsmSSF - gsmSCF interface, for the control of circuit switched calls.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

    errortypes,
    datatypes,
    operationcodes,
    classes,
    tc-Messages,
    ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

    CallingPartysCategory,
    HighLayerCompatibility,
    RedirectionInformation,
    ServiceKey
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
modules(0) cs1-datatypes(2) version1(0)}

    CallSegmentID {},
    MiscCallInfo
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
cs2(20) modules(0) in-cs2-datatypes (0) version1(0)}

    Ext-BasicServiceCode,
    IMEI,
    IMSI,
    ISDN-AddressString
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}

    CUG-Index,
    CUG-Interlock,
    CUG-Info,
    LocationInformation,
    MS-Classmark2,

```

```
SubscriberState,  
SupportedCamelPhases,  
SupportedCamel4Subsets  
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0)  
mobileDomain(0)  
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}
```

```
CallReferenceNumber,  
SuppressionOfAnnouncement  
FROM MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0)  
mobileDomain(0)  
gsm-Network(1) modules(3) map-CH-DataTypes(13) version6(6)}
```

```
PARAMETERS-BOUND  
FROM CAP-classes classes
```

```
opcode-activityTest,  
opcode-applyCharging,  
opcode-applyChargingReport,  
opcode-assistRequestInstructions,  
opcode-callGap,  
opcode-callInformationReport,  
opcode-callInformationRequest,  
opcode-cancel,  
opcode-connect,  
opcode-connectToResource,  
opcode-continue,  
opcode-continueWithArgument,  
opcode-disconnectForwardConnection,  
opcode-dFCWithArgument,  
opcode-disconnectLeg,  
opcode-entityReleased,  
opcode-establishTemporaryConnection,  
opcode-eventReportBCSM,  
opcode-furnishChargingInformation,  
opcode-initialDP,  
opcode-initiateCallAttempt,  
opcode-moveLeg,  
opcode-playTone,  
opcode-releaseCall,  
opcode-requestReportBCSMEvent,  
opcode-resetTimer,  
opcode-sendChargingInformation,  
opcode-splitLeg  
FROM CAP-operationcodes operationcodes
```

```
AChBillingChargingCharacteristics {},  
AdditionalCallingPartyNumber {},  
AlertingPattern,  
AssistingSSPIPRoutingAddress {},  
BCSMEvent,  
BCSM-Failure,  
BearerCapability {},  
CalledPartyNumber {},  
CalledPartyBCDNumber {},  
CallingPartyNumber {},  
CallResult {},
```

CallSegmentToCancel,
CallSegmentFailure,
Carrier,
Cause {},
CGEncountered,
ChargeNumber {},
ControlType,
CorrelationID {},
DestinationRoutingAddress {},
EventSpecificChargingInformation {},
EventSpecificInformationBCSM {},
EventTypeBCSM,
EventTypeChargingPLMN,
Extensions {},
FCIBillingChargingCharacteristics {},
GapCriteria {},
GapIndicators,
GapTreatment,
GenericNumbers {},
InvokeID,
IPRoutingAddress {},
IPSSPCapabilities {},
leg1,
LegOrCallSegment {},
LocationNumber {},
MonitorMode,
NAOLIInfo,
OCSIAplicable,
OriginalCalledPartyID {},
ReceivingSideID,
RedirectingPartyID {},
RequestedInformationList {},
RequestedInformationTypeList,
ScfID {},
SCIBillingChargingCharacteristics {},
SendingSideID,
ServiceInteractionIndicatorsTwo,
TimeAndTimezone {},
TimerID,
TimerValue

FROM CAP-datatypes datatypes

cancelFailed,
eTCFailed,
missingCustomerRecord,
missingParameter,
parameterOutOfRange,
requestedInfoError,
systemFailure,
taskRefused,
unexpectedComponentSequence,
unexpectedDataValue,
unexpectedParameter,
unknownLegID,
unknownCSID

FROM CAP-errortypes errortypes


```

;

activityTest OPERATION ::= {
    RETURN RESULT      TRUE
    CODE               opcode-activityTest}
-- Direction: gsmSCF -> gsmSSF, Timer: Tat
-- This operation is used to check for the continued existence of a relationship
-- between the gsmSCF and gsmSSF, assist gsmSSF or gsmSRF. If the relationship
is
-- still in existence, then the gsmSSF will respond. If no reply is received,
-- then the gsmSCF will assume that the gsmSSF, assist gsmSSF or grmSRF has
failed
-- in some way.

applyCharging {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT           ApplyChargingArg {bound}
    RETURN RESULT      FALSE
    ERRORS              {missingParameter |
                        unexpectedComponentSequence |
                        unexpectedParameter |
                        unexpectedDataValue |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unknownLegID |
                        unknownCSID}
    CODE               opcode-applyCharging}
-- Direction: gsmSCF -> gsmSSF, Timer: Tac
-- This operation is used for interacting from the gsmSCF with the gsmSSF
charging mechanisms.
-- The ApplyChargingReport operation provides the feedback from the gsmSSF to
the gsmSCF.

ApplyChargingArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    aChBillingChargingCharacteristics [0] AChBillingChargingCharacteristics
{bound},
    partyToCharge                      [2] SendingSideID DEFAULT
sendingSideID : leg1,
    extensions                          [3] Extensions {bound}
OPTIONAL,
    aChChargingAddress                 [50] AChChargingAddress
{bound}
DEFAULT
legID:sendingSideID:leg1,
    ...
}

-- The partyToCharge parameter indicates the party in the call to which the
ApplyCharging operation
-- shall be applied.

applyChargingReport {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT           ApplyChargingReportArg {bound}
    RETURN RESULT      FALSE
    ERRORS              {missingParameter |
                        unexpectedComponentSequence |
                        unexpectedParameter |

```

```

                unexpectedDataValue |
                parameterOutOfRange |
                systemFailure |
                taskRefused}
CODE                opcode-applyChargingReport}
-- Direction: gsmSSF -> gsmSCF, Timer: Tacr
-- This operation is used by the gsmSSF to report to the gsmSCF the occurrence
of a
-- specific charging event as requested by the gsmSCF using the ApplyCharging
operation.

ApplyChargingReportArg {PARAMETERS-BOUND : bound} ::= CallResult {bound}

assistRequestInstructions {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT                AssistRequestInstructionsArg {bound}
    RETURN RESULT           FALSE
    ERRORS                   {missingCustomerRecord |
                            missingParameter |
                            systemFailure |
                            taskRefused |
                            unexpectedComponentSequence |
                            unexpectedDataValue |
                            unexpectedParameter}
CODE                        opcode-assistRequestInstructions}
-- Direction: gsmSSF -> gsmSCF or gsmSRF -> gsmSCF, Timer: Tari
-- This operation is used when there is an assist procedure and may be
-- sent by the gsmSSF or gsmSRF to the gsmSCF. This operation is sent by the
-- assisting gsmSSF to gsmSCF, when the initiating gsmSSF has set up a
connection to
-- the gsmSRF or to the assisting gsmSSF as a result of receiving an
-- EstablishTemporaryConnection from
-- the gsmSCF.
-- Refer to clause 11 for a description of the procedures associated with this
operation.

AssistRequestInstructionsArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    correlationID                [0] CorrelationID {bound},
    iPSSPCapabilities            [2] IPSSPCapabilities {bound},
    extensions                    [3] Extensions {bound}
                                OPTIONAL,
    ...
}
-- OPTIONAL denotes network operator specific use. The value of the
correlationID may be the
-- Called Party Number supplied by the initiating gsmSSF.

callGap {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT                CallGapArg {bound}
    RETURN RESULT           FALSE
    ALWAYS RESPONDS        FALSE
CODE                        opcode-callGap}
-- Direction: gsmSCF -> gsmSSF, Timer: Tcg
-- This operation is used to request the gsmSSF to reduce the rate at which
specific service
-- requests are sent to the gsmSCF.

CallGapArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {

```

```

        gapCriteria                [0] GapCriteria {bound},
        gapIndicators              [1] GapIndicators,
        controlType                [2] ControlType
                                OPTIONAL,
        gapTreatment               [3] GapTreatment {bound}
                                OPTIONAL,
        extensions                 [4] Extensions {bound}
                                OPTIONAL,
        ...
    }
-- OPTIONAL denotes network operator optional. If gapTreatment is not present,
then the gsmSSF will
-- use a default treatment depending on network operator implementation.

callInformationReport {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          CallInformationReportArg {bound}
    RETURN RESULT     FALSE
    ALWAYS RESPONDS  FALSE
    CODE              opcode-callInformationReport}
-- Direction: gsmSSF -> gsmSCF, Timer: Tcirp
-- This operation is used to send specific call information for a single call
party to the gsmSCF as
-- requested by the gsmSCF in a previous CallInformationRequest.

CallInformationReportArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    RequestedInformationList [0] RequestedInformationList
{bound},
    extensions                [2] Extensions {bound}
                                OPTIONAL,
    legID                     [3] ReceivingSideID
                                OPTIONAL,
    ...
}

callInformationRequest {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          CallInformationRequestArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
                      parameterOutOfRange |
                      requestedInfoError |
                      systemFailure |
                      taskRefused |
                      unexpectedComponentSequence |
                      unexpectedDataValue |
                      unexpectedParameter |
                      unknownLegID}
    CODE              opcode-callInformationRequest}
-- Direction: gsmSCF -> gsmSSF, Timer: Tcirq
-- This operation is used to request the gsmSSF to record specific information
about a single
-- call party and report it to the gsmSCF (with a CallInformationReport
operation).

CallInformationRequestArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    requestedInformationTypeList [0] RequestedInformationTypeList
{bound},

```

```

        extensions                                [2] Extensions {bound}
        legID                                     OPTIONAL,
                                                [3] SendingSideID
        OPTIONAL,
        ...
    }
-- OPTIONAL denotes network operator optional.

cancel OPERATION ::= {
    ARGUMENT          CancelArg
    RETURN RESULT     FALSE
    ERRORS            {cancelFailed |
                      missingParameter |
                      taskRefused |
                      unknownCSID}
    CODE              opcode-cancel}
-- Direction: gsmSCF -> gsmSSF, or gsmSCF -> gsmSRF, Timer: Tcon
-- This operation cancels the correlated previous operation or all previous
requests. The following
-- operations can be canceled: PlayAnnouncement,
PromptAndCollectUserInformation.

CancelArg ::= CHOICE {
    invokeID                [0] InvokeID,
    allRequests             [1] NULL,
    callSegmentToCancel     [2] CallSegmentToCancel
}
-- The InvokeID has the same value as that which was used for the operation to
be cancelled.

connect {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          ConnectArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unexpectedComponentSequence |
                      unexpectedDataValue |
                      unexpectedParameter}
    CODE              opcode-connect}
-- Direction: gsmSCF-> gsmSSF, Timer: Tcon
-- This operation is used to request the gsmSSF to perform the call processing
actions
-- to route or forward a call to a specified destination.

ConnectArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    destinationRoutingAddress [0] DestinationRoutingAddress
{bound},
    alertingPattern           [1] AlertingPattern
                                OPTIONAL,
    originalCalledPartyID     [6] OriginalCalledPartyID
{bound}
    extensions                [10] Extensions {bound}
                                OPTIONAL,
    carrier                   [11] Carrier {bound}
                                OPTIONAL,
}

```

```

        callingPartysCategory          [28] CallingPartysCategory
                OPTIONAL,
        redirectingPartyID              [29] RedirectingPartyID
{bound}                                OPTIONAL,
        redirectionInformation          [30] RedirectionInformation
                OPTIONAL,
        genericNumbers                  [14] GenericNumbers
{bound}                                OPTIONAL,
        serviceInteractionIndicatorsTwo [15] ServiceInteractionIndicatorsTwo
                OPTIONAL,
        chargeNumber                    [19] ChargeNumber {bound}
                OPTIONAL,
        legToBeCreated                  [21] LegID
                OPTIONAL,
        cug-Interlock                   [31] CUG-Interlock
                OPTIONAL,
        cug-OutgoingAccess              [32] NULL
                OPTIONAL,
        suppressionOfAnnouncement       [55] SuppressionOfAnnouncement
                OPTIONAL,
        oCSIApplicable                  [56] OCSIApplicable
                OPTIONAL,
        naOliInfo                       [57] NAOliInfo
                OPTIONAL,
        bor-InterrogationRequested      [58] NULL
                OPTIONAL,
        ...
    }
-- na-Info is included at the discretion of the gsmSCF operator.

connectToResource {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          ConnectToResourceArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownCSID}
    CODE              opcode-connectToResource}
-- Direction: gsmSCF -> gsmSSF, Timer: Tctr
-- This operation is used to connect a call segment from the gsmSSF to the
-- gsmSRF.

ConnectToResourceArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    resourceAddress CHOICE {
        ipRoutingAddress [0] IPRoutingAddress
{bound},
        none [3] NULL
    },
    extensions [4] Extensions {bound}
                OPTIONAL,
    serviceInteractionIndicatorsTwo [7]
ServiceInteractionIndicatorsTwo OPTIONAL,
    callSegmentID [50] CallSegmentID {bound}
                OPTIONAL,

```

```

    ...
}

continue OPERATION ::= {
    RETURN RESULT      FALSE
    ALWAYS RESPONDS   FALSE
    CODE               opcode-continue}
-- Direction: gsmSCF -> gsmSSF, Timer: Tcuc
-- This operation is used to request the gsmSSF to proceed with call processing
at the
-- DP at which it previously suspended call processing to await gsmSCF
instructions
-- (i.e. proceed to the next point in call in the BCSM). The gsmSSF continues
call
-- processing without substituting new data from gsmSCF.

continueWithArgument {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT           ContinueWithArgumentArg {bound}
    RETURN RESULT      FALSE
    ERRORS              {missingParameter |
                        parameterOutOfRange |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownLegID}
    CODE               opcode-continueWithArgument}
-- Direction: gsmSCF -> gsmSSF, Timer: Tcwa
-- This operation is used to request the gsmSSF to proceed with call processing
at the
-- DP at which it previously suspended call processing to await gsmSCF
instructions
-- (i.e. proceed to the next point in call in the BCSM). The gsmSSF continues
call
-- processing with the modified call setup information as received from the
gsmSCF.

ContinueWithArgumentArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    legID                                     [0] LegID
                                           OPTIONAL,
    alertingPattern                          [1] AlertingPattern
                                           OPTIONAL,
    extensions                               [6] Extensions {bound}
                                           OPTIONAL,
    serviceInteractionIndicatorsTwo          [7]
ServiceInteractionIndicatorsTwo            OPTIONAL,
    callingPartysCategory                   [12] CallingPartysCategory
                                           OPTIONAL,
    genericNumbers                           [16] GenericNumbers
{bound}                                    OPTIONAL,
    cug-Interlock                            [17] CUG-Interlock
                                           OPTIONAL,
    cug-OutgoingAccess                      [18] NULL
                                           OPTIONAL,
    chargeNumber                             [50] ChargeNumber {bound}
                                           OPTIONAL,
    carrier                                  [52] Carrier {bound}
                                           OPTIONAL,

```

```

    suppressionOfAnnouncement          [55] SuppressionOfAnnouncement
            OPTIONAL,
    naOliInfo                          [56] NAOliInfo
            OPTIONAL,
    bor-InterrogationRequested         [57] NULL
            OPTIONAL,
    suppress-O-CSI                    [58] NULL
            OPTIONAL,
    continueWithArgumentArgExtension  [59] ContinueWithArgumentArgExtension
            OPTIONAL,
    ...
}

ContinueWithArgumentArgExtension ::= SEQUENCE {
    suppress-D-CSI                    [0] NULL
            OPTIONAL,
    suppress-N-CSI                    [1] NULL
            OPTIONAL,
    suppressOutgoingCallBarring       [2] NULL
            OPTIONAL,
    ...
}

disconnectForwardConnection OPERATION ::= {
    RETURN RESULT          FALSE
    ERRORS                  {systemFailure |
                            taskRefused |
                            unexpectedComponentSequence}
    CODE                    opcode-disconnectForwardConnection}
-- Direction: gsmSCF -> gsmSSF, Timer: Tdfc
-- This operation is used to disconnect a forward temporary connection or a
connection to a
-- resource. Refer to clause 11 for a description of the procedures associated
with this operation.

DisconnectForwardConnectionWithArgument {PARAMETERS-BOUND : bound} OPERATION ::=
{
    ARGUMENT                DisconnectForwardConnectionWithArgumentArg {bound}
    RETURN RESULT          FALSE
    ERRORS                  {missingParameter |
                            systemFailure |
                            taskRefused |
                            unexpectedComponentSequence |
                            unexpectedDataValue |
                            unexpectedParameter |
                            unknownLegID |
                            unknownCSID}
    CODE                    opcode-dFCWithArgument}
-- Direction gsmSCF -> gsmSSF, Timer Tdfcwa
-- This operation is used to disconnect a forward temporary connection or a
connection to a
-- resource. Refer to clause 11 for a description of the procedures associated
with this operation.

DisconnectForwardConnectionWithArgumentArg {PARAMETERS-BOUND : bound} ::=
SEQUENCE {

```

```

        callSegmentID                                [1]    CallSegmentID
{bound}
        extensions                                  [2]    Extensions {bound}
        OPTIONAL,
        ...
    }

disconnectLeg {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          DisconnectLegArg {bound}
    RETURN RESULT     TRUE
    ERRORS             {missingParameter |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownLegId}
    CODE              opcode-disconnectLeg}
-- Direction: gsmSCF -> gsmSSF, Timer Tdl
-- This operation is used by the gsmSCF to release a specific leg associated
with the call and
-- retain any other legs not specified in the DisconnectLeg. Refer to clause 11
for a description
-- of the procedures associated with this operation.

DisconnectLegArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    LegToBeReleased [0]    LegID,
    ReleaseCause    [1]    Cause {bound}
    OPTIONAL,
    extensions      [2]    Extensions {bound}
    OPTIONAL,
    ...
}

entityReleased {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          EntityReleasedArg {bound},
    RETURN RESULT     FALSE
    ALWAYS RESPONDS  FALSE
    CODE              opcode-entityReleased}
-- Direction: gsmSSF -> gsmSCF, Timer: Ter
-- This operation is used by the gsmSSF to inform the gsmSCF of an error or
exception

EntityReleasedArg {PARAMETERS-BOUND : bound} ::= CHOICE {
    CallSegmentFailure [0]    CallSegmentFailure,
    bCSM-Failure       [1]    BCSM-Failure
}

establishTemporaryConnection {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          EstablishTemporaryConnectionArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {eTCFailed |
                        missingParameter |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |

```



```

                unexpectedParameter |
                unknownCSID}
CODE                opcode-establishTemporaryConnection}
-- Direction: gsmSCF -> gsmSSF, Timer: Tetc
-- This operation is used to create a connection to a resource for a limited
period
-- of time (e.g. to play an announcement, to collect user information); it
implies
-- the use of the assist procedure. Refer to clause 11 for a description of the
-- procedures associated with this operation.

```

```

EstablishTemporaryConnectionArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    assistingSSPIPRoutingAddress [0] AssistingSSPIPRoutingAddress
bound},
    correlationID [1] CorrelationID {bound}
OPTIONAL,
    scfID [3] ScfID {bound}
OPTIONAL,
    extensions [4] Extensions {bound}
OPTIONAL,
    carrier [5] Carrier {bound}
OPTIONAL,
    serviceInteractionIndicatorsTwo [6]
ServiceInteractionIndicatorsTwo OPTIONAL,
    callSegmentID [7] CallSegmentID
bound} OPTIONAL,
    naOliInfo [50] NAOLIInfo
OPTIONAL,
    chargeNumber [51] ChargeNumber {bound}
OPTIONAL,
    ...
}

```

```

eventNotificationCharging {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT EventNotificationChargingArg { bound}
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE opcode-eventNotificationCharging}
-- Direction: gsmSSF -> gsmSCF, Timer: Tenc
-- This operation is used by the gsmSSF to report to the gsmSCF the occurrence
of a
-- specific charging event type as previously requested by the gsmSCF in a
-- RequestNotificationChargingEvent operation.

```

```

EventNotificationChargingArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    EventSpecificChargingInformation [1] EventSpecificChargingInformation{
bound},
    legID [2] LegID
OPTIONAL,
    extensions [3] Extensions {bound}
OPTIONAL,
    ...
}

```

```

eventReportBCSM {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT EventReportBCSMArg {bound}
    RETURN RESULT FALSE
}

```

```

        ALWAYS RESPONDS    FALSE
        CODE                opcode-eventReportBCSM}
-- Direction: gsmSSF -> gsmSCF, Timer: Terb
-- This operation is used to notify the gsmSCF of a call-related event (e.g.
BCSM
-- events such as O_Busy or O_No_Answer) previously requested by the gsmSCF in a
-- RequestReportBCSMEvent operation.

EventReportBCSMArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    eventTypeBCSM                [0] EventTypeBCSM,
    eventSpecificInformationBCSM [2] EventSpecificInformationBCSM
{bound} OPTIONAL,
    legID                        [3] ReceivingSideID
                                OPTIONAL,
    miscCallInfo                 [4] MiscCallInfo DEFAULT
{messageType request},
    extensions                    [5] Extensions
                                OPTIONAL,
    ...
}

furnishChargingInformation {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT        FurnishChargingInformationArg {bound}
    RETURN RESULT   FALSE
    ERRORS          {missingParameter |
                    taskRefused |
                    unexpectedComponentSequence |
                    unexpectedDataValue |
                    unexpectedParameter |
                    unknownLegId}
    CODE            opcode-furnishChargingInformation}
-- Direction: gsmSCF -> gsmSSF, Timer: Tfci
-- This operation is used to request the gsmSSF to generate, register a call
record
-- or to include some information in the default call record.
-- The registered call record is intended for off line charging of the call.

FurnishChargingInformationArg {PARAMETERS-BOUND : bound} ::=
    FCIBillingChargingCharacteristics{bound}

initialDP {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT        InitialDPArg {bound}
    RETURN RESULT   FALSE
    ERRORS          {missingCustomerRecord |
                    missingParameter |
                    parameterOutOfRange |
                    systemFailure |
                    taskRefused |
                    unexpectedComponentSequence |
                    unexpectedDataValue |
                    unexpectedParameter}
    CODE            opcode-initialDP}
-- Direction: gsmSSF -> gsmSCF, Timer: Tidp
-- This operation is used after a TDP to indicate request for service.

InitialDPArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    serviceKey                [0] ServiceKey ,

```

calledPartyNumber		[2] CalledPartyNumber {bound}
OPTIONAL,		
callingPartyNumber		[3] CallingPartyNumber
{bound}	OPTIONAL,	
callingPartysCategory		[5] CallingPartysCategory
OPTIONAL,		
cGEncountered		[7] CGEncountered
OPTIONAL,		
iPSSPCapabilities		[8] IPSSPCapabilities {bound}
OPTIONAL,		
locationNumber		[10] LocationNumber
{bound}	OPTIONAL,	
originalCalledPartyID		[12] OriginalCalledPartyID
{bound}	OPTIONAL,	
extensions		[15] Extensions
OPTIONAL,		
highLayerCompatibility		[23] HighLayerCompatibility
OPTIONAL,		
additionalCallingPartyNumber	[25] AdditionalCallingPartyNumber	
{bound}	OPTIONAL,	
bearerCapability		[27] BearerCapability {bound}
OPTIONAL,		
eventTypeBCSM		[28] EventTypeBCSM
OPTIONAL,		
redirectingPartyID		[29] RedirectingPartyID
{bound}	OPTIONAL,	
redirectionInformation		[30] RedirectionInformation
OPTIONAL,		
cause		[17] Cause {bound}
OPTIONAL,		
serviceInteractionIndicatorsTwo	[32]	
ServiceInteractionIndicatorsTwo	OPTIONAL,	
carrier		[37] Carrier {bound}
OPTIONAL,		
cug-Index		[45] CUG-Index
OPTIONAL,		
cug-Interlock		[46] CUG-Interlock
OPTIONAL,		
cug-OutgoingAccess		[47] NULL
OPTIONAL,		
iMSI		[50] IMSI
OPTIONAL,		
subscriberState		[51] SubscriberState
OPTIONAL,		
locationInformation		[52] LocationInformation
OPTIONAL,		
ext-basicServiceCode	[53] Ext-BasicServiceCode	
OPTIONAL,		
callReferenceNumber		[54] CallReferenceNumber
OPTIONAL,		
mscAddress		[55] ISDN-AddressString
OPTIONAL,		
calledPartyBCDNumber	[56] CalledPartyBCDNumber	
{bound}	OPTIONAL,	
timeAndTimezone		[57] TimeAndTimezone
{bound}	OPTIONAL,	


```

        newCallSegment                                [6]    CallSegmentID
{bound}
        callingPartyNumber                            [30]   CallingPartyNumber
{bound}
        callReferenceNumber                           [51]   CallReferenceNumber
        gsmSCFAddress                                [52]   ISDN-AddressString
        suppress-T-CSI                                [53]   NULL
        OPTIONAL,
        ...
}

```

```

InitiateCallAttemptRes {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    supportedCamelPhases                             [0]    SupportedCamelPhases
        OPTIONAL,
    supportedCamel4Subsets                           [1]    SupportedCamel4Subsets
        OPTIONAL,
    extensions                                        [2]    Extensions {bound}
        OPTIONAL,
    ...
}

```

```

moveLeg {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          MoveLegArg {bound}
    RETURN RESULT     TRUE
    ERRORS             {missingParameter |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownLegID}

    CODE              opcode-moveLeg}
-- Direction: gsmSCF -> gsmSSF, Timer: Tml
-- This operation is used by the gsmSCF to move a leg from one call segment to
another call segment
-- within the same call segment association.

```

```

MoveLegArg {PARAMETERS-BOUND : bound} ::= SEQUENCE{
    legIDToMove                                         [0]    LegID,
    extensions                                          [2]    Extensions {bound}
        OPTIONAL,
    ...
}

```

```

playTone {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          playToneArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        parameterOutOfRange |
                        systemFailure |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownLegID |
                        unknownCSID}
}

```

```

        CODE                opcode-playTone}
-- Direction: gsmSCF -> gsmSSF, Timer: Tpt
-- This operation is used to play tones to either a leg or a call segment using
-- the MSC's tone generator.

PlayToneArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    LegOrCallSegment          [0] LegOrCallSegment {bound},
    bursts                    [1] Burst,
    extensions                 [2] Extensions
                                OPTIONAL,
    ...
}

releaseCall {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT                ReleaseCallArg {bound}
    RETURN RESULT           FALSE
    ALWAYS RESPONDS        FALSE
    CODE                    opcode-releaseCall}
-- Direction: gsmSCF ->?gsmSSF, Timer: Trc
-- This operation is used to tear down an existing call at any phase of the call
for all parties
-- involved in the call.

ReleaseCallArg {PARAMETERS-BOUND : bound} ::= Cause {bound}
-- A default value of decimal 31 (normal unspecified) shall be given.

requestNotificationChargingEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT                RequestNotificationChargingEventArg {bound}
    RETURN RESULT           FALSE
    ERRORS                  {missingParameter |
                            parameterOutOfRange |
                            systemFailure |
                            taskRefused |
                            unexpectedComponentSequence |
                            unexpectedDataValue |
                            unexpectedParameter |
                            unknownLegId}
    CODE                    opcode-requestNotificationChargingEvent}
-- Direction: gsmSCF -> gsmSSF, Timer: Trnc
-- This operation is used by the gsmSCF to instruct the gsmSSF on how to manage
the charging events
-- which are received from other FEs and not under control of the service logic
instance.

RequestNotificationChargingEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    EventTypeChargingPLMN   [0] EventTypeChargingPLMN
{bound},
    monitorMode              [1] MonitorMode,
    legID                    [2] LegID
                                OPTIONAL,
    extensions                [3] Extensions {bound}
                                OPTIONAL,
    ...
}
-- This argument indicates the charging event type and the corresponding monitor
mode and LegID.

```

```

requestReportBCSMEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          RequestReportBCSMEventArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unexpectedComponentSequence |
                      unexpectedDataValue |
                      unexpectedParameter |
                      unknownLegID}
    CODE              opcode-requestReportBCSMEvent}
-- Direction: gsmSCF -> gsmSSF, Timer: Trrb
-- This operation is used to request the gsmSSF to monitor for a call-related
event
-- (e.g. BCSM events such as O_Busy or O_No_Answer) and to send a notification
-- to the gsmSCF when the event is detected.
--
-- NOTE:
-- Every EDP must be explicitly armed by the gsmSCF via a RequestReportBCSMEvent
operation.
-- No implicit arming of EDPs at the gsmSSF after reception of any operation
(different
-- from RequestReportBCSMEvent) from the gsmSCF is allowed.

RequestReportBCSMEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    bcsmEvents                [0] SEQUENCE
SIZE(1..bound.&numOfBCSMEvents) OF
    BCSMEvent {bound},
    Extensions                [2] Extensions {bound}
    OPTIONAL,
    ...
}
-- Indicates the BCSM related events for notification.

resetTimer {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          ResetTimerArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
                      parameterOutOfRange |
                      taskRefused |
                      unexpectedComponentSequence |
                      unexpectedDataValue |
                      unexpectedParameter |
                      unknownCSID}
    CODE              opcode-resetTimer}
-- Direction: gsmSCF -> gsmSSF, Timer: Trt
-- This operation is used to request the gsmSSF to refresh an application timer
in the gsmSSF.

ResetTimerArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    timerID                [0] TimerID DEFAULT
tssf,
    timervalue              [1] TimerValue,
    extensions              [2] Extensions {bound}
    OPTIONAL,

```

```

        callSegmentID                                [3]    CallSegmentID
{bound}                                           OPTIONAL,
        ...
    }

sendChargingInformation {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          SendChargingInformationArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        unexpectedComponentSequence |
                        unexpectedParameter |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unexpectedDataValue |
                        unknownLegID}

    CODE              opcode-sendChargingInformation}
-- Direction: gsmSCF -> gsmSSF, Timer: Tsci
-- This operation is used to instruct the gsmSSF on the charging information to
-- send by the gsmSSF.
-- The charging information can either be sent back by means of signalling or
-- internal
-- if the gsmSSF is located in the local exchange. In the local exchange
-- this information may be used to update the charge meter or to create a
-- standard call record.

SendChargingInformationArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    sCIBillingChargingCharacteristics [0] SCIBillingChargingCharacteristics
{bound},
    partyToCharge                      [1] SendingSideID,
    extensions                          [2] Extensions {bound}

    OPTIONAL,

    ...
}

splitLeg {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          SplitLegArg {bound}
    RETURN RESULT     TRUE
    ERRORS             {missingParameter |
                        unexpectedComponentSequence |
                        unexpectedParameter |
                        unexpectedDataValue |
                        systemFailure |
                        taskRefused |
                        unknownLegID}

    CODE              opcode-splitLeg}
-- Direction: gsmSCF -> gsmSSF, Timer Tsl
-- This operation is used by the gsmSCF to separate one joined leg from a multi-
-- way connection or
-- a single two party Call Segment.

SplitLegArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    legToBeSplit      [0]    LegID,
    newCallSegment    [1]    CallSegmentID
{bound}
    OPTIONAL,
    extensions        [2]    Extensions {bound}

    OPTIONAL,

```



```
...
}
```

```
END
```

```
-- 6.1.2    gsmSSF/gsmSCF packages, contracts and ACs
-- 6.1.2.1  gsmSSF/gsmSCF ASN.1 module
CAP-gsmSSF-gsmSCF-pkgs-contracts-accs {itu-t(0) identified-organization(4)
etsi(0) mobileDomain(0) umts-network(1) modules(3) cap-gsmSSF-gsmSCF-pkgs-
contracts-accs(102) version4(3)}
```

```
DEFINITIONS ::= BEGIN
```

```
-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the gsmSSF - gsmSCF interface, for the control
of
-- circuit switched calls.
```

```
-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.
```

```
IMPORTS
```

```
    PARAMETERS-BOUND,
    cAPSpecificBoundSet
FROM CAP-classes classes
```

```
    CONTRACT,
    OPERATION-PACKAGE,
    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
    TCMessage {}
FROM TCAPMessages tc-Messages
```

```
    APPLICATION-CONTEXT,
    dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions
```

```
    activityTest,
    applyCharging {},
    applyChargingReport {},
    assistRequestInstructions {},
    callGap {},
    callInformationReport {},
    callInformationRequest {},
    cancel,
    connect {},
    connectToResource {},
    continue,
    continueWithArgument {},
    disconnectForwardConnection,
    disconnectForwardConnectionWithArgument {},
    disconnectLeg {},
    entity Released {},
    establishTemporaryConnection {},
    eventNotificationCharging {},
```

```

eventReportBCSM {},
furnishChargingInformation {},
initialDP {},
initiateCallAttempt {},
moveLeg {},
playTone {}
releaseCall {},
requestNotificationChargingEvent {},
requestReportBCSMEvent {},
resetTimer {},
sendChargingInformation {},
splitLeg {}
FROM CAP-gsmSSF-gsmSCF-ops-args gsmSSF-gsmSCF-Operations

    playAnnouncement {},
    promptAndCollectUserInformation {},
    specializedResourceReport
FROM CAP-gsmSCF-gsmSRF-ops-args gsmSCF-gsmSRF-Operations

    specializedResourceControlPackage {}
FROM CAP-gsmSCF-gsmSRF-pkgs-contracts-acs gsmSCF-gsmSRF-Protocol

id-ac-CAP-gsmSSF-scfGenericAC,
id-ac-CAP-gsmSSF-scfAssistHandoffAC,
id-ac-CAP-scf-gsmSSFGenericAC,
id-CAPSsfToScfGeneric,
id-CAPAssistHandoffssfToScf,
id-as-gsmSSF-scfGenericAS,
id-as-assistHandoff-gsmSSF-scfAS,
id-package-scfActivation,
id-package-gsmSRF-scfActivationOfAssist,
id-package-assistConnectionEstablishment,
id-package-genericDisconnectResource,
id-package-nonAssistedConnectionEstablishment,
id-package-connect,
id-package-callHandling,
id-package-bcsmEventHandling,
id-package-chargingEventHandling,
id-package-ssfCallProcessing,
id-package-scfCallInitiation,
id-package-timer,
id-package-billing,
id-package-charging,
id-package-trafficManagement,
id-package-callReport,
id-package-signallingControl,
id-package-activityTest,
id-package-cancel,
id-package-cphResponse,
id-package-exceptionInform,
classes,
ros-InformationObjects,
tc-Messages,
tc-NotationExtensions,
gsmSSF-gsmSCF-Operations,
gsmSCF-gsmSRF-Operations,
gsmSCF-gsmSRF-Protocol

```

```

FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

;

-- Application Contexts

capssf-scfGenericAC APPLICATION-CONTEXT ::= {
    CONTRACT                capSsfToScfGeneric
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             gsmSSF-scfGenericAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gsmSSF-scfGenericAC}

capssf-scfAssistHandoffAC APPLICATION-CONTEXT ::= {
    CONTRACT                capAssistHandoffssfToScf
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             assistHandoff-gsmSSF-
scfAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gsmSSF-scfAssistHandoffAC}

capscf-ssfGenericAC APPLICATION-CONTEXT ::= {
    CONTRACT                capScfToSsfGeneric
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             scf-gsmSSFGenericAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-scf-gsmSSFGenericAC}

-- Contracts

capSsfToScfGeneric CONTRACT ::= {
-- dialogue initiated by gsmSSF with InitialDP Operation
    INITIATOR CONSUMER OF {exceptionInformPackage
{cAPSpecificBoundSet} |
                             scfActivationPackage
{cAPSpecificBoundSet}}
    RESPONDER CONSUMER OF {activityTestPackage |
                             assistConnectionEstablishmentPackage {cAPSpecificBoundSet} |
                             bcsmEventHandlingPackage
{cAPSpecificBoundSet} |
                             billingPackage
{cAPSpecificBoundSet} |
                             callHandlingPackage
{cAPSpecificBoundSet} |
                             callReportPackage
{cAPSpecificBoundSet} |
                             cancelPackage |
                             chargingEventHandlingPackage
{cAPSpecificBoundSet} |
                             chargingPackage
{cAPSpecificBoundSet} |
                             connectPackage
{cAPSpecificBoundSet} |

```

```

cphResponsePackage
{cAPSpecificBoundSet} |
genericDisconnectResourcePackage
{cAPSpecificBoundSet} |
nonAssistedConnectionEstablishmentPackage {cAPSpecificBoundSet} |
signallingControlPackage
{cAPSpecificBoundSet} |
specializedResourceControlPackage {cAPSpecificBoundSet} |
ssfCallProcessingPackage
{cAPSpecificBoundSet} |
timerPackage
{cAPSpecificBoundSet} |
trafficManagementPackage
{cAPSpecificBoundSet}
scfCallInitiationPackage
{cAPSpecificBoundSet}}
ID id-CAPSsfToScfGeneric}

capAssistHandoffssfToScf CONTRACT ::= {
-- dialogue initiated by gsmSSF with AssistRequestInstructions
INITIATOR CONSUMER OF {gsmSRF-scfActivationOfAssistPackage
{cAPSpecificBoundSet}}
RESPONDER CONSUMER OF {activityTestPackage |
callHandlingPackage
{cAPSpecificBoundSet} |
cancelPackage |
genericDisconnectResourcePackage
{cAPSpecificBoundSet} |
nonAssistedConnectionEstablishmentPackage {cAPSpecificBoundSet} |
specializedResourceControlPackage {cAPSpecificBoundSet} |
timerPackage
{cAPSpecificBoundSet}}
ID id-CAPAssistHandoffssfToScf}

capScfToSsfGeneric CONTRACT ::= {
-- dialogue initiated by gsmSCF with InitiateCallAttempt, Generic Case
INITIATOR CONSUMER OF {activityTestPackage |
assistConnectionEstablishmentPackage {cAPSpecificBoundSet} |
bcsmEventHandlingPackage
{cAPSpecificBoundSet} |
billingPackage
{cAPSpecificBoundSet} |
callHandlingPackage
{cAPSpecificBoundSet} |
callReportPackage
{cAPSpecificBoundSet} |
cancelPackage |
chargingPackage
{cAPSpecificBoundSet} |
connectPackage
{cAPSpecificBoundSet} |

```

```

cphResponsePackage
{cAPSpecificBoundSet} |
genericDisconnectResourcePackage
{cAPSpecificBoundSet} |
    nonAssistedConnectionEstablishmentPackage {cAPSpecificBoundSet} |
scfCallInitiationPackage
{cAPSpecificBoundSet} |
signallingControlPackage
{cAPSpecificBoundSet} |
    specializedResourceControlPackage {cAPSpecificBoundSet} |
ssfCallProcessingPackage
{cAPSpecificBoundSet} |
timerPackage
{cAPSpecificBoundSet}
    RESPONDER CONSUMER OF {exceptionInformPackage
{cAPSpecificBoundSet}
    ID id-capScfToSsfGeneric}

-- Operation Packages

chargingEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {requestNotificationChargingEvent {bound}}
    SUPPLIER INVOKES {eventNotificationCharging {bound}}
    ID id-package-chargingEventHandling}

scfActivationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {initialDP {bound}}
    ID id-package-scfActivation}

gsmSRF-scfActivationOfAssistPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE
::= {
    CONSUMER INVOKES {assistRequestInstructions {bound}}
    ID id-package-gsmSRF-scfActivationOfAssist}

assistConnectionEstablishmentPackage {PARAMETERS-BOUND : bound} OPERATION-
PACKAGE ::= {
    CONSUMER INVOKES {establishTemporaryConnection {bound}}
    ID id-package-assistConnectionEstablishment}

genericDisconnectResourcePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE
::= {
    CONSUMER INVOKES {disconnectForwardConnection |
disconnectForwardConnectionWithArgument
{bound}}
    ID id-package-genericDisconnectResource}

nonAssistedConnectionEstablishmentPackage {PARAMETERS-BOUND : bound} OPERATION-
PACKAGE ::= {
    CONSUMER INVOKES {connectToResource {bound}}
    ID id-package-
nonAssistedConnectionEstablishment}

connectPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {connect {bound}}

```

```

ID                                     id-package-connect}

callHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {releaseCall {bound}}
    ID                                     id-package-callHandling}

bcsmEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {requestReportBCSMEvent {bound}}
    SUPPLIER INVOKES {eventReportBCSM {bound}}
    ID                                     id-package-bcsmEventHandling}

ssfCallProcessingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {continueWithArgument {bound} | continue}
    ID                                     id-package-ssfCallProcessing}

scfCallInitiatePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {initiateCallAttempt {bound}}
    ID                                     id-package-scfCallInitiation}

timerPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {resetTimer {bound}}
    ID                                     id-package-timer}

billingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {furnishChargingInformation {bound}}
    ID                                     id-package-billing}

chargingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {applyCharging {bound}}
    SUPPLIER INVOKES {applyChargingReport {bound}}
    ID                                     id-package-charging}

trafficManagementPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {callGap {bound}}
    ID                                     id-package-trafficManagement}

callReportPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {callInformationRequest {bound}}
    SUPPLIER INVOKES {callInformationReport {bound}}
    ID                                     id-package-callReport}

signallingControlPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {sendChargingInformation {bound}}
    ID                                     id-package-signallingControl}

activityTestPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {activityTest}
    ID                                     id-package-activityTest}

cancelPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {cancel}
    ID                                     id-package-cancel}

cphResponsePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {continueWithArgument {bound} |
                    disconnectLeg {bound} |
                    moveLeg {bound} |

```

```

        splitLeg {bound}}
    ID id-package-cphResponse}

exceptionInformPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {entityReleased}
    ID id-package-exceptionInfom}

-- Abstract Syntaxes

gsmSSF-scfGenericAbstractSyntax ABSTRACT-SYNTAX ::= {
    GenericSSF-gsmSCF-PDUs
    IDENTIFIED BY id-as-gsmSSF-scfGenericAS}

GenericSSF-gsmSCF-PDUs ::= TCMMessage {{SsfToScfGenericInvokable},
    {SsfToScfGenericReturnable}}

SsfToScfGenericInvokable OPERATION ::= {
    activityTest |
    applyCharging {cAPSSpecificBoundSet} |
    applyChargingReport {cAPSSpecificBoundSet} |
    callInformationReport {cAPSSpecificBoundSet} |
    callInformationRequest {cAPSSpecificBoundSet} |
    cancel |
    connect {cAPSSpecificBoundSet} |
    continueWithArgument {cAPSSpecificBoundSet} |
    connectToResource {cAPSSpecificBoundSet} |
    disconnectForwardConnection |
    disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
    disconnectLeg {cAPSSpecificBoundSet} |
    entityReleased {cAPSSpecificBoundSet} |
    establishTemporaryConnection {cAPSSpecificBoundSet} |
    eventNotificationCharging {cAPSSpecificBoundSet} |
    eventReportBCSM {cAPSSpecificBoundSet} |
    furnishChargingInformation {cAPSSpecificBoundSet} |
    initialDP {cAPSSpecificBoundSet} |
    initiateCallAttempt {cAPSSpecificBoundSet} |
    moveLeg {cAPSSpecificBoundSet} |
    releaseCall {cAPSSpecificBoundSet} |
    requestNotificationChargingEvent {cAPSSpecificBoundSet} |
    requestReportBCSMEEvent {cAPSSpecificBoundSet} |
    resetTimer {cAPSSpecificBoundSet} |
    sendChargingInformation {cAPSSpecificBoundSet} |
    splitLeg {cAPSSpecificBoundSet} |
    playAnnouncement {cAPSSpecificBoundSet} |
    promptAndCollectUserInformation {cAPSSpecificBoundSet} |
    specializedResourceReport
}

SsfToScfGenericReturnable OPERATION ::= {
    activityTest |
    applyCharging {cAPSSpecificBoundSet} |
    applyChargingReport {cAPSSpecificBoundSet} |
    callGap {cAPSSpecificBoundSet} |
    callInformationRequest {cAPSSpecificBoundSet} |
    cancel |

```

```

connect {cAPSSpecificBoundSet} |
connectToResource {cAPSSpecificBoundSet} |
continue |
continueWithArgument {cAPSSpecificBoundSet} |
disconnectForwardConnection |
disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
disconnectLeg {cAPSSpecificBoundSet} |
entityReleased {cAPSSpecificBoundSet} |
establishTemporaryConnection {cAPSSpecificBoundSet} |
furnishChargingInformation {cAPSSpecificBoundSet} |
initialDP {cAPSSpecificBoundSet} |
initiateCallAttempt {cAPSSpecificBoundSet} |
moveLeg {cAPSSpecificBoundSet} |
releaseCall {cAPSSpecificBoundSet} |
requestReportBCSMEEvent {cAPSSpecificBoundSet} |
resetTimer {cAPSSpecificBoundSet} |
sendChargingInformation {cAPSSpecificBoundSet} |
splitLeg {cAPSSpecificBoundSet} |
playAnnouncement {cAPSSpecificBoundSet} |
promptAndCollectUserInformation {cAPSSpecificBoundSet}
}

assistHandoff-gsmSSF-scfAbstractSyntax ABSTRACT-SYNTAX ::= {
    AssistHandoffsSF-gsmSCF-PDUs
    IDENTIFIED BY id-as-assistHandoff-gsmSSF-scfAS}

AssistHandoffsSF-gsmSCF-PDUs ::= TCMMessage {{AssistHandoffssfToScfInvokable},
    {AssistHandoffssfToScfReturnable}}

AssistHandoffssfToScfInvokable OPERATION ::= {
    activityTest |
    assistRequestInstructions {cAPSSpecificBoundSet} |
    cancel |
    connectToResource {cAPSSpecificBoundSet} |
    disconnectForwardConnection |
    disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
    playAnnouncement {cAPSSpecificBoundSet} |
    promptAndCollectUserInformation {cAPSSpecificBoundSet} |
    resetTimer {cAPSSpecificBoundSet} |
    specializedResourceReport
}

AssistHandoffssfToScfReturnable OPERATION ::= {
    activityTest |
    assistRequestInstructions {cAPSSpecificBoundSet} |
    cancel |
    connectToResource {cAPSSpecificBoundSet} |
    disconnectForwardConnection |
    disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
    playAnnouncement {cAPSSpecificBoundSet} |
    promptAndCollectUserInformation {cAPSSpecificBoundSet} |
    resetTimer {cAPSSpecificBoundSet}
}

scf-gsmSSFGenericAbstractSyntax ABSTRACT-SYNTAX ::= {
    GenericSCF-gsmSSF-PDUs

```


IDENTIFIED BY id-as-scf-gsmSSFGenericAS}

GenericSCF-gsmSSF-PDUs ::= TCMMessage {{ScfToSsfGenericInvokable},
{ScfToSsfGenericReturnable}}

ScfToSsfGenericInvokable OPERATION ::= {
 activityTest |
 applyCharging {cAPSpecificBoundSet} |
 applyChargingReport {cAPSpecificBoundSet} |
 callInformationRequest {cAPSpecificBoundSet} |
 cancel |
 connect {cAPSpecificBoundSet} |
 connectToResource {cAPSpecificBoundSet} |
 continue |
 continueWithArgument {cAPSpecificBoundSet} |
 disconnectForwardConnection {cAPSpecificBoundSet} |
 disconnectForwardConnectionWithArgument {cAPSpecificBoundSet} |
 disconnectLeg {cAPSpecificBoundSet} |
 establishTemporaryConnection {cAPSpecificBoundSet} |
 furnishChargingInformation {cAPSpecificBoundSet} |
 initiateCallAttempt {cAPSpecificBoundSet} |
 moveLeg {cAPSpecificBoundSet} |
 playTone {cAPSpecificBoundSet} |
 releaseCall {cAPSpecificBoundSet} |
 requestReportBCSMEEvent {cAPSpecificBoundSet} |
 resetTimer {cAPSpecificBoundSet} |
 sendChargingInformation {cAPSpecificBoundSet} |
 splitLeg {cAPSpecificBoundSet} |
 playAnnouncement {cAPSpecificBoundSet} |
 promptAndCollectUserInformation {cAPSpecificBoundSet}
}

ScfToSsfGenericReturnable OPERATION ::= {
 activityTest |
 applyCharging {cAPSpecificBoundSet} |
 applyChargingReport {cAPSpecificBoundSet} |
 callInformationReport {cAPSpecificBoundSet} |
 callInformationRequest {cAPSpecificBoundSet} |
 cancel |
 connect {cAPSpecificBoundSet} |
 connectToResource {cAPSpecificBoundSet} |
 disconnectForwardConnection |
 disconnectForwardConnectionWithArgument {cAPSpecificBoundSet} |
 disconnectLeg {cAPSpecificBoundSet} |
 entityReleased {cAPSpecificBoundSet} |
 establishTemporaryConnection {cAPSpecificBoundSet} |
 eventReportBCSM {cAPSpecificBoundSet} |
 furnishChargingInformation {cAPSpecificBoundSet} |
 initiateCallAttempt {cAPSpecificBoundSet} |
 moveLeg {cAPSpecificBoundSet} |
 requestReportBCSMEEvent {cAPSpecificBoundSet} |
 resetTimer {cAPSpecificBoundSet} |
 sendChargingInformation {cAPSpecificBoundSet} |
 splitLeg {cAPSpecificBoundSet} |
 playAnnouncement {cAPSpecificBoundSet} |
 promptAndCollectUserInformation {cAPSpecificBoundSet} |
 specializedResourceReport

```

    }

END

-- 6.2      gsmSCF/gsmSRF interface
-- 6.2.1    gsmSCF/gsmSRF operations and arguments
CAP-gsmSCF-gsmSRF-ops-args {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-gsmSCF-gsmSRF-ops-args(103) version4(3)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- This module contains the operations and operation arguments used for the
-- gsmSRF - gsmSCF interface, for the control of circuit switched calls.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

    opcode-playAnnouncement,
    opcode-promptAndCollectUserInformation,
    opcode-specializedResourceReport
FROM CAP-operationcodes operationcodes

    CallSegmentID {}
FROM CS2-datatypes {itu-t(0) identified-organisation(4) etsi(0) inDomain(1) in-
network(1) cs2(20) modules(0) in-cs2-datatypes(0) version1(0)}

    CollectedInfo,
    Digits {},
    Extensions {},
    InformationToSend {}
FROM CAP-datatypes datatypes

    canceled,
    improperCallerResponse,
    missingParameter,
    parameterOutOfRange,
    systemFailure,
    taskRefused,
    unavailableResource,
    unexpectedComponentSequence,
    unexpectedDataValue,
    unexpectedParameter,
    unknownCSID
FROM CAP-erroratypes erroratypes

    PARAMETERS-BOUND
FROM CAP-classes classes

    ros-InformationObjects,
    operationcodes,
    datatypes,

```

```

    errortypes,
    classes
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

;

playAnnouncement {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          PlayAnnouncementArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {canceled |
                        missingParameter |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unavailableResource |
                        unknownCSID}
    LINKED             {specializedResourceReport}
    CODE               opcode-playAnnouncement}
-- Direction: gsmSCF -> gsmSRF, Timer: Tpa
-- This operation is to be used after Establish Temporary Connection (assist
procedure
-- with a second gsmSSF) or a Connect to Resource (no assist) operation. It may
be used
-- for inband interaction with a mobile station, or for interaction with an ISDN
user.
-- In the former case, the gsmSRF is usually collocated with the gsmSSF for
standard
-- tones (congestion tone...) or standard announcements.
-- In the latter case, the gsmSRF is always collocated with the gsmSSF in the
switch.
-- Any error is returned to the gsmSCF. The timer associated with this operation
must
-- be of a sufficient duration to allow its linked operation to be correctly
correlated.

PlayAnnouncementArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    informationToSend          [0] InformationToSend {bound},
    disconnectFromIPForbidden [1] BOOLEAN DEFAULT TRUE,
    requestAnnouncementCompleteNotification [2] BOOLEAN DEFAULT TRUE,
    extensions                 [3] Extensions {bound}
                                OPTIONAL,
    callSegmentID              [5] CallSegmentID {bound}
                                OPTIONAL,
    requestAnnouncementStartedNotification [51] BOOLEAN DEFAULT FALSE,
    ...
}

promptAndCollectUserInformation {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          PromptAndCollectUserInformationArg {bound}
    RESULT            ReceivedInformationArg {bound}
    ERRORS             {canceled |
                        improperCallerResponse |

```

```

missingParameter |
parameterOutOfRange |
systemFailure |
taskRefused |
unexpectedComponentSequence |
unavailableResource |
unexpectedDataValue |
unexpectedParameter |
unknownCSID}
LINKED {specializedResourceReport}
CODE opcode-promptAndCollectUserInformation}
-- Direction: gsmSCF -> gsmSRF, Timer: Tpc
-- This operation is used to interact with a user to collect information.

PromptAndCollectUserInformationArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    collectedInfo [0] CollectedInfo,
    disconnectFromIPForbidden [1] BOOLEAN DEFAULT TRUE,
    informationToSend [2] InformationToSend {bound}
        OPTIONAL,
    extensions [3] Extensions {bound}
        OPTIONAL,
    callSegmentID [4] CallSegmentID
{bound} OPTIONAL,
    requestAnnouncementStartedNotification [51] BOOLEAN DEFAULT FALSE,
    ...
}

ReceivedInformationArg {PARAMETERS-BOUND : bound} ::= CHOICE {
    digitsResponse [0] Digits {bound}
}

specializedResourceReport OPERATION ::= {
    ARGUMENT SpecializedResourceReportArg
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE opcode-specializedResourceReport}
-- Direction: gsmSRF -> gsmSCF, Timer: Tsrr
-- This operation is used as the response to a PlayAnnouncement operation when
the announcement
-- completed report indication is set.

SpecializedResourceReportArg ::= CHOICE {
    allAnnouncementsComplete [50] NULL,
    firstAnnouncementStarted [51] NULL
}

END
-- 6.2.2 gsmSRF/gsmSCF contracts, packages and ACs
-- 6.2.2.1 gsmSRF/gsmSCF ASN.1 modules
CAP-gsmSCF-gsmSRF-pkgs-contracts-ac {itu-t(0) identified-organization(4)
etsi(0) mobileDomain(0) umts-network(1) modules(3) cap-gsmSCF-gsmSRF-pkgs-
contracts-ac(104) version4(3)}

DEFINITIONS ::= BEGIN

-- This module specifies the Operation Packages, Contracts, Application Contexts

```

-- and Abstract Syntaxes used for the gsmSRF - gsmSCF interface, for the control of
-- circuit switched calls.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

PARAMETERS-BOUND,
CAPSpecificBoundSet
FROM CAP-classes classes

CONTRACT,
OPERATION-PACKAGE,
OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

TCMessage {}
FROM TCAPMessages tc-Messages

APPLICATION-CONTEXT,
dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions

playAnnouncement {},
promptAndCollectUserInformation {},
specializedResourceReport
FROM CAP-gsmSCF-gsmSRF-ops-args gsmSCF-gsmSRF-Operations

activityTest,
cancel,
assistRequestInstructions {}
FROM CAP-gsmSSF-gsmSCF-ops-args gsmSSF-gsmSCF-Operations

gsmSRF-scfActivationOfAssistPackage {}
FROM CAP-gsmSSF-gsmSCF-pkgs-contracts-acs gsmSSF-gsmSCF-Protocol

id-package-specializedResourceControl,
id-package-activityTest,
id-ac-gsmSRF-gsmSCF,
id-contract-gsmSRF-gsmSCF,
id-package-gsmSRF-scfCancel,
id-as-basic-gsmSRF-gsmSCF,
classes,
ros-InformationObjects,
tc-Messages,
tc-NotationExtensions,
gsmSCF-gsmSRF-Operations,
gsmSSF-gsmSCF-Operations,
gsmSSF-gsmSCF-Protocol
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3)r cap-object-identifiers(100) version4(3)}

;

```

-- Application Contexts

gsmSRF-gsmSCF-ac APPLICATION-CONTEXT ::= {
    CONTRACT                gsmSRF-gsmSCF-contract
    DIALOGUE MODE           structured
    TERMINATION             basic
    ABSTRACT SYNTAXES      {dialogue-abstract-syntax |
                           gsmSRF-gsmSCF-abstract-syntax}
    APPLICATION CONTEXT NAME id-ac-gsmSRF-gsmSCF}

-- Contracts

gsmSRF-gsmSCF-contract CONTRACT ::= {
    INITIATOR CONSUMER OF  {gsmSRF-scfActivationOfAssistPackage
{cAPSpecificBoundSet}}
    RESPONDER CONSUMER OF {specializedResourceControlPackage
{cAPSpecificBoundSet} |
                           activityTestPackage |
                           gsmSRF-scfCancelPackage }
    ID                     id-contract-gsmSRF-gsmSCF}

-- Operation Packages

specializedResourceControlPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE
::= {
    CONSUMER INVOKES  {playAnnouncement {bound} |
                       promptAndCollectUserInformation {bound}
    }
    SUPPLIER INVOKES {specializedResourceReport}
    ID               id-package-specializedResourceControl}

gsmSRF-scfCancelPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {cancel }
    ID               id-package-gsmSRF-scfCancel}

ActivityTestPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES {activityTest}
    ID               id-package-activityTest}

-- Abstract Syntaxes

gsmSRF-gsmSCF-abstract-syntax ABSTRACT-SYNTAX ::= {
    BASIC-gsmSRF-gsmSCF-PDUs
    IDENTIFIED BY      id-as-basic-gsmSRF-gsmSCF}

BASIC-gsmSRF-gsmSCF-PDUs ::= TCMMessage
{{GsmSRFgsmSCFInvokable},{GsmSRFgsmSCFReturnable}}

GsmSRFgsmSCFInvokable OPERATION ::= {
    activityTest |
    assistRequestInstructions {cAPSpecificBoundSet} |
    cancel |
    playAnnouncement {cAPSpecificBoundSet} |
    promptAndCollectUserInformation {cAPSpecificBoundSet} |
    specializedResourceReport
    }

```

```
GsmSRFgsmSCFReturnable OPERATION ::= {
    activityTest |
    assistRequestInstructions {CAPSpecificBoundSet} |
    cancel |
    playAnnouncement {CAPSpecificBoundSet} |
    promptAndCollectUserInformation {CAPSpecificBoundSet}
}

END
```

```

-- 7 SMS Control
-- 7.1 SMS operations and arguments
CAP-SMS-ops-args {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-SMS-ops-args(105) version4(3)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- This module contains the operations and operation arguments used for the
-- smsSSF- gsmSCF interface, for the control of MO-SMS and MT-SMS.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

    errortypes,
    datatypes,
    operationcodes,
    classes,
    ros-InformationObjects,
    tc-Messages
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

    OPERATION

FROM Remote-Operations-Information-Objects ros-InformationObjects

    ServiceKey
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
modules(0) cs1-datatypes(2) version1(0)}

    MiscCallInfo
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
cs2(20) modules(0) in-cs2-datatypes (0) version1(0)}

    IMEI,
    IMSI,
    ISDN-AddressString
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}

    GPRSMSCClass,
    LocationInformation,
    MS-Classmark2
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}

    PARAMETERS-BOUND
FROM CAP-classes classes

    opcode-connectSMS,
    opcode-continueSMS,

```



```

opcode-eventReportSMS,
opcode-furnishChargingInformationSMS,
opcode-initialDPSMS,
opcode-releaseSMS,
opcode-requestReportSMSEvent,
opcode-resetTimerSMS
FROM CAP-operationcodes operationcodes

```

```

CalledPartyBCDNumber {},
EventSpecificInformationSMS,
EventTypesSMS,
Extensions {},
FCISMSBillingChargingCharacteristics,
LocationInformationGPRS,
RPCause,
SMSEvent,
TimeAndTimezone {},
TimerID,
TimerValue,
TPDataCodingScheme,
TPProtocolIdentifier,
TPShortMessageSpecificInfo,
TPValidityPeriod
FROM CAP-datatypes datatypes

```

```

missingCustomerRecord,
missingParameter,
parameterOutOfRange,
systemFailure,
taskRefused,
unexpectedComponentSequence,
unexpectedDataValue,
unexpectedParameter
FROM CAP-errortypes errortypes

```

```

CallReferenceNumber,
FROM MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-network(1) modules(3) map-CH-DataTypes(13) version6(6)}

```

```

;

```

```

connectSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          ConnectSMSArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter}
    CODE              opcode-connectSMS}

```

```

-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Tconsms
-- This operation is used to request the smsSSF to perform the SMS processing
-- actions to route
-- or forward a short message to a specified destination.

```

```

ConnectSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    callingPartysNumber                [0] ISDN-AddressString
                                         OPTIONAL,
    destinationSubscriberNumber        [1] CalledPartyBCDNumber {bound}
                                         OPTIONAL,
    smscAddress                        [2] ISDN-AddressString
                                         OPTIONAL,
    extensions                          [10] Extensions {bound}
                                         OPTIONAL,
    ...
}

continueSMS OPERATION ::= {
    RETURN RESULT      FALSE
    ALWAYS RESPONDS   FALSE
    CODE               opcode-continueSMS}
-- Direction: gsmSCF -> smsSSF, Timer: Tcuesms
-- This operation is used to request the smsSSF to proceed with
-- Short Message processing at the DP at which it previously suspended
-- Short Message processing to await gsmSCF instructions (i.e. proceed
-- to the next Point in Association in the SMS FSM). The smsSSF
-- continues SMS processing without substituting new data from the gsmSCF.

eventReportSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          EventReportSMSArg {bound}
    RETURN RESULT     FALSE
    ALWAYS RESPONDS   FALSE
    CODE              opcode-eventReportSMS}
-- Direction: gsmSSF or gprsSSF -> gsmSCF, Timer: Terbsms
-- This operation is used to notify the gsmSCF of a Short Message related event
(FSM events
-- such as submission, delivery or failure) previously requested by the gsmSCF
in a
-- RequestReportSMSEvent operation.

EventReportSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    eventTypeSMS                [0] EventTypeSMS,
    eventSpecificInformationSMS [1] EventSpecificInformationSMS
                                         OPTIONAL,
    miscCallInfo                [2] MiscCallInfo DEFAULT
{messageType request},
    extensions                  [10] Extensions {bound}
                                         OPTIONAL,
    ...
}

furnishChargingInformationSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          FurnishChargingInformationSMSArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
taskRefused |
unexpectedComponentSequence |
unexpectedDataValue |
unexpectedParameter}
    CODE              opcode-furnishChargingInformationSMS}
-- Direction: gsmSCF ?? gsmSSF or gprsSSF, Timer: Tfcisms

```

```
-- This operation is used to request the smsSSF to generate, register a
charging record
-- or to include some information in the default SM record. The registered
charging record is
-- intended for off line charging of the Short Message.
```

```
FurnishChargingInformationSMSArg {PARAMETERS-BOUND : bound} ::=
    FCISMSBillingChargingCharacteristics {bound}
```

```
initialDPSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          InitialDPSMSArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingCustomerRecord |
                        missingParameter |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter}
    CODE              opcode-initialDPSMS}
```

```
-- Direction: gsmSSF or gprsSSF -> gsmSCF, Timer: Tidpsms
-- This operation is used after a TDP to indicate request for service.
```

```
InitialDPSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    serviceKey                [0] ServiceKey,
    destinationSubscriberNumber [1] CalledPartyBCDNumber {bound}
        OPTIONAL,
    callingPartyNumber        [2] ISDN-AddressString
        OPTIONAL,
    eventTypeSMS              [3] EventTypeSMS
        OPTIONAL,
    IMSI                      [4] IMSI
        OPTIONAL,
    locationInformationMSC     [5] LocationInformation
        OPTIONAL,
    locationInformationGPRS    [6] LocationInformationGPRS
        OPTIONAL,
    SMSCAddress               [7] ISDN-AddressString
        OPTIONAL,
    timeAndTimezone           [8] TimeAndTimezone
        OPTIONAL,
    tPShortMessageSpecificInfo [9] TPShortMessageSpecificInfo
        OPTIONAL,
    tPProtocolIdentifier       [10] TPProtocolIdentifier
        OPTIONAL,
    tPDataCodingScheme        [11] TPDataCodingScheme
        OPTIONAL,
    tPValidityPeriod          [12] TPValidityPeriod
        OPTIONAL,
    extensions                 [13] Extensions {bound}
        OPTIONAL,
    ...,
    smsReferenceNumber        [14] CallReferenceNumber
        OPTIONAL,
    mscAddress                [15] ISDN-AddressString
        OPTIONAL,
```

```

sgsn-Number [16] ISDN-AddressString
OPTIONAL,
ms-Classmark2 [17] MS-Classmark2
OPTIONAL,
gPRSMSCClass [18] GPRSMSCClass
OPTIONAL,
iMEI [19] IMEI
OPTIONAL,
calledPartyNumber [17] ISDN-AddressString
OPTIONAL
}

releaseSMS OPERATION ::= {
    ARGUMENT ReleaseSMSArg
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE opcode-releaseSMS}
-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Trelsms
-- This operation is used to prevent an attempt to submit or deliver a short
message.

ReleaseSMSArg ::= RPCause

requestReportSMSEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT RequestReportSMSEventArg {bound}
    RETURN RESULT FALSE
    ERRORS {missingParameter |
            parameterOutOfRange |
            systemFailure |
            taskRefused |
            unexpectedComponentSequence |
            unexpectedDataValue |
            unexpectedParameter}
    CODE opcode-requestReportSMSEvent}
-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Trrbsms
-- This operation is used to request the gsmSSF or gprsSSF to monitor for a
-- Short Message related event (FSM events such as submission, delivery or
failure)
-- and to send a notification to the gsmSCF when the event is detected.

RequestReportSMSEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    sMSEvents [0] SEQUENCE SIZE
(1..bound.&numOfSMSEvents) OF SMSEvent,
    extensions [10] Extensions {bound}
OPTIONAL,
...
}
-- Indicates the Short Message related events(s) for notification.

resetTimerSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT ResetTimerSMSArg {bound}
    RETURN RESULT FALSE
    ERRORS {missingParameter |
            parameterOutOfRange |
            taskRefused |
            unexpectedComponentSequence |
            unexpectedDataValue |

```

```

                                unexpectedParameter}
CODE                                opcode-resetTimerSMS}
-- Direction: gsmSCF -> smsSSF, Timer: Trtsms
-- This operation is used to request the smsSSF to refresh an application
-- timer in the smsSSF.

ResetTimerSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    timerID                                [0] TimerID DEFAULT
tssf,
    timervalue                            [1] TimerValue,
    extensions                            [2] Extensions {bound}
                                OPTIONAL,
    ...
}

END
-- 7.2      SMS contracts, packages and ACs
-- 7.2.1    SMS ASN.1 module
CAP-smsSSF-gsmSCF-pkgs-contracts-accs {itu-t(0) identified-organization(4)
etsi(0) mobileDomain(0) umts-network(1) modules(3) cap-smsSSF-gsmSCF-pkgs-
contracts-accs(106) version4(3)}

DEFINITIONS ::= BEGIN

-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the smsSSF - gsmSCF interface, for the
-- control of MO-SMS and MT-SMS.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

    PARAMETERS-BOUND,
    cAPSpecificBoundSet
FROM CAP-classes classes

    CONTRACT,
    OPERATION-PACKAGE,
    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

    TCMessage {}
FROM TCAPMessages tc-Messages

    APPLICATION-CONTEXT,
    dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions

    connectSMS {},
    continueSMS,
    eventReportsSMS {},
    furnishChargingInformationSMS {},
    initialDPSMS {},
    releaseSMS,
    requestReportSMSEvent {},
    resetTimerSMS {}

```

FROM CAP-SMS-ops-args sms-Operations

id-ac-cap3-sms-AC,
id-ac-cap4-sms-AC,
id-cap3SmsSsfTogsmScf,
id-cap4SmsSsfTogsmScf,
id-package-smsActivation,
id-package-smsConnect,
id-package-smsContinue,
id-package-smsRelease,
id-package-smsEventHandling,
id-package-smsBilling,
id-package-smsTimer,
sms-Operations,
tc-NotationExtensions,
tc-Messages,
ros-InformationObjects,
classes,
id-as-smsSSF-gsmSCF-AS

FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

;

-- Application Contexts

cap3-sms-AC APPLICATION-CONTEXT ::= {
CONTRACT cap3SMS
DIALOGUE MODE structured
ABSTRACT SYNTAXES {dialogue-abstract-syntax |
sms-AbstractSyntax}
APPLICATION CONTEXT NAME id-ac-cap3-sms-AC}

-- This application context shall be used for CAMEL support of MO-SMS.

cap4-sms-AC APPLICATION-CONTEXT ::= {
CONTRACT cap4SMS
DIALOGUE MODE structured
ABSTRACT SYNTAXES {dialogue-abstract-syntax |
sms-AbstractSyntax}
APPLICATION CONTEXT NAME id-ac-cap4-sms-AC}

-- This application context shall be used for CAMEL support of MT-SMS.

-- Contracts

cap3SMS CONTRACT ::= {
-- dialogue initiated by gprsSSF or gsmSSF with InitialDPSMS Operation
INITIATOR CONSUMER OF {smsActivationPackage {cAPSpecificBoundSet}}
RESPONDER CONSUMER OF {smsConnectPackage {cAPSpecificBoundSet} |
smsReleasePackage |
smsEventHandlingPackage
{cAPSpecificBoundSet} |
smsTimerPackage
{cAPSpecificBoundSet} |
smsBillingPackage
{cAPSpecificBoundSet} |

```

ID smsProcessingPackage}
id-cap3SmsSsfTogsmScf}

cap4SMS CONTRACT ::= {
-- dialogue initiated by gprsSSF or gsmSSF with InitialDPSMS Operation
INITIATOR CONSUMER OF {smsActivationPackage {cAPSpecificBoundSet}}
RESPONDER CONSUMER OF {smsConnectPackage {cAPSpecificBoundSet} |
smsReleasePackage |
smsEventHandlingPackage
{cAPSpecificBoundSet} |
smsTimerPackage
{cAPSpecificBoundSet} |
smsBillingPackage
{cAPSpecificBoundSet} |
ID smsProcessingPackage }
id-cap4SmsSsfTogsmScf}

-- Operation Packages

smsActivationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {initialDPSMS {bound}}
ID id-package-smsActivation}

smsConnectPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {connectSMS {bound}}
ID id-package-smsConnect}

smsProcessingPackage OPERATION-PACKAGE ::= {
CONSUMER INVOKES {continueSMS}
ID id-package-smsContinue}

smsReleasePackage OPERATION-PACKAGE ::= {
CONSUMER INVOKES {releaseSMS}
ID id-package-smsRelease}

smsEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {requestReportSMSEvent {bound}}
SUPPLIER INVOKES {eventReportSMS {bound}}
ID id-package-smsEventHandling}

smsBillingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {furnishChargingInformationSMS {bound}}
ID id-package-smsBilling}

smsTimerPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {resetTimerSMS {bound}}
ID id-package-smsTimer}

-- Abstract Syntaxes

sms-AbstractSyntax ABSTRACT-SYNTAX ::= {
Generic-sms-PDUs
IDENTIFIED BY id-as-smsSSF-gsmSCF-AS }

Generic-sms-PDUs ::= TCMessage {{SmsInvokable}, {SmsReturnable}}

```

```
SmsInvokable OPERATION ::= {  
    connectSMS {CAPSpecificBoundSet} |  
    eventReportSMS {CAPSpecificBoundSet} |  
    furnishChargingInformationSMS {CAPSpecificBoundSet} |  
    initialDPSMS {CAPSpecificBoundSet} |  
    requestReportSMSEvent {CAPSpecificBoundSet} |  
    resetTimerSMS {CAPSpecificBoundSet}}
```

```
SmsReturnable OPERATION ::= {  
    connectSMS {CAPSpecificBoundSet} |  
    continueSMS |  
    furnishChargingInformationSMS {CAPSpecificBoundSet} |  
    initialDPSMS {CAPSpecificBoundSet} |  
    releaseSMS |  
    requestReportSMSEvent {CAPSpecificBoundSet} |  
    resetTimerSMS {CAPSpecificBoundSet}}
```

END


```

-- 8 GPRS Control
-- 8.1 gsmSCF/gprsSSF operations and arguments
CAP-gprsSSF-gsmSCF-ops-args {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-GPRS-ops-args(107) version3(2)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- This module contains the operations and operation arguments used for the
-- gprsSSF - gsmSCF interface, for the control of GPRS.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

    errortypes,
    datatypes,
    operationcodes,
    classes,
    ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version3(2)}

    OPERATION

FROM Remote-Operations-Information-Objects ros-InformationObjects

    ServiceKey
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
modules(0) cs1-datatypes(2) version1(0)}

    MiscCallInfo
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
cs2(20) modules(0) in-cs2-datatypes (0) version1(0)}

    IMEI,
    IMSI,
    ISDN-AddressString
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}

    GPRSChargingID,
    GPRSMSCClass,
    GSN-Address,
    LocationInformationGPRS,
    RAIdentity
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}

    PARAMETERS-BOUND
FROM CAP-classes classes

```

```

opcode-activityTestGPRS,
opcode-applyChargingGPRS,
opcode-applyChargingReportGPRS,
opcode-cancelGPRS,
opcode-connectGPRS,
opcode-continueGPRS,
opcode-entityReleasedGPRS,
opcode-eventReportGPRS,
opcode-furnishChargingInformationGPRS,
opcode-initialDPGPRS,
opcode-releaseGPRS,
opcode-requestReportGPRSEvent,
opcode-resetTimerGPRS,
opcode-sendChargingInformationGPRS
FROM CAP-operationcodes operationcodes

```

```

AccessPointName {},
GPRSCause {},
ChargingCharacteristics,
ChargingResult,
ChargingRollOver,
EndUserAddress,

```

```

Extensions,
FCIGPRSBillingChargingCharacteristics,
GPRSEventSpecificInformation {},
GPRSEvent,
GPRSEventType,
PDPID,
PDPInitiationType,
QualityOfService,
SCIGPRSBillingChargingCharacteristics,
SGSNCapabilities,
TimeAndTimezone {},
TimerID,
TimerValue
FROM CAP-datatypes datatypes

```

```

missingCustomerRecord,
missingParameter,
parameterOutOfRange,
systemFailure,
taskRefused,
unexpectedComponentSequence,
unexpectedDataValue,
unexpectedParameter,
unknownPDPID

```

```
FROM CAP-erroratypes erroratypes
```

```
;
```

```

activityTestGPRS OPERATION ::= {
    RETURN RESULT      TRUE
    CODE               opcode-activityTestGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tatg
-- This operation is used to check for the continued existence of a relationship
between the gsmSCF

```

```

-- and gprsSSF. If the relationship is still in existence, then the gprsSSF will
respond. If no
-- reply is received, then the gsmSCF will assume that the gprsSSF has failed in
some way
-- and will take the appropriate action.

```

```

applyChargingGPRS OPERATION ::= {
    ARGUMENT          ApplyChargingGPRSArg
    RETURN RESULT    FALSE
    ERRORS            {missingParameter |
                      unexpectedComponentSequence |
                      unexpectedParameter |
                      unexpectedDataValue |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unknownPDPID}
    CODE              opcode-applyChargingGPRS}
-- Direction gsmSCF -> gprsSSF, Timer Tacg
-- This operation is used for interacting from the gsmSCF with the gprsSSF CSE-
controlled
-- GPRS session or PDP Context charging mechanism.

```

```

ApplyChargingGPRSArg ::= SEQUENCE {
    ChargingCharacteristics          [0] ChargingCharacteristics,
    tariffSwitchInterval             [1] INTEGER (1..86400)
    OPTIONAL,
    pDPID                            [2] PDPID
    OPTIONAL,
    ...
}
-- tariffSwitchInterval is measured in 1 second units.

```

```

ApplyChargingReportGPRS OPERATION ::= {
    ARGUMENT          ApplyChargingReportGPRSArg
    RETURN RESULT    TRUE
    ERRORS            {missingParameter |
                      unexpectedComponentSequence |
                      unexpectedParameter |
                      unexpectedDataValue |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unknownPDPID}
    CODE              opcode-applyChargingReportGPRS}
-- Direction gprsSSF -> gsmSCF,Timer Tacrg
-- The ApplyChargingReportGPRS operation provides the feedback from the
gprsSCF to the gsmSCF
-- CSE-controlled GPRS session charging mechanism.

```

```

ApplyChargingReportGPRSArg ::= SEQUENCE {
    ChargingResult          [0] ChargingResult,
    QualityOfService        [1] QualityOfService
    OPTIONAL,
    active                  [2] BOOLEAN DEFAULT
TRUE,

```

```

    pdpID                                     [3] PDPID
                                           OPTIONAL,
    ...,
    chargingRollOver                         [4] ChargingRollOver
                                           OPTIONAL
}

cancelGPRS OPERATION ::= {
    ARGUMENT          CancelGPRSArg
    RETURN RESULT    FALSE
    ERRORS            {missingParameter |
                      taskRefused |
                      unknownPDPID}
    CODE             opcode-cancelGPRS}
-- Direction: gsmSCF ?? gprsSSF, Timer: Tcag
-- This generic operation cancels all previous requests,
-- i.e. all EDPs and reports can be cancelled by the gsmSCF.

CancelGPRSArg ::= SEQUENCE {
    pdpID                                     [0] PDPID
                                           OPTIONAL,
    ...
}

connectGPRS {PARAMETERS-BOUND: bound} OPERATION ::= {
    ARGUMENT          ConnectGPRSArg {bound}
    RETURN RESULT    FALSE
    ERRORS            {missingParameter |
                      parameterOutOfRange |
                      unknownPDPID |
                      systemFailure |
                      taskRefused |
                      unexpectedComponentSequence |
                      unexpectedDataValue |
                      unexpectedParameter}
    CODE             opcode-connectGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tcong
-- This operation is used to modify the Access Point Name used when establishing
a PDP Context.

ConnectGPRSArg {PARAMETERS-BOUND: bound} ::= SEQUENCE {
    accessPointName [0] AccessPointName {bound},
    pdpID           [1] PDPID             OPTIONAL,
    ...
}

continueGPRS OPERATION ::= {
    ARGUMENT          ContinueGPRSArg
    RETURN RESULT    FALSE
    ERRORS            {missingParameter |
                      unknownPDPID |
                      unexpectedDataValue}
    CODE             opcode-continueGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tcueg
-- This operation is used to request the gprsSSF to proceed with processing at
the DP at

```

```

-- which it previously suspended processing to await gsmSCF instructions (i.e.,
-- proceed to
-- the next point in processing in the Attach/Detach state model or PDP Context
-- state model) substituting new data from the gsmSCF.

```

```

ContinueGPRSArg ::= SEQUENCE {
    pDPID [0] PDPID
    OPTIONAL,
    ...
}

```

```

entityReleasedGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT EntityReleasedGPRSArg {bound}
    RETURN RESULT TRUE
    ERRORS {missingParameter |
            taskRefused |
            unknownPDPID}
    CODE opcode-entityReleasedGPRS}
-- Direction: gprsSSF -> gsmSCF, Timer: Terg
-- This operation is used when the GPRS Session is detached or a PDP Context is
-- disconnected and
-- the associated event is not armed for reporting.
-- The usage of this operation is independent of the functional entity that
-- initiates the Detach
-- or PDP Context Disconnection and is independent of the cause of the Detach
-- or PDP Context
-- Disconnect.

```

```

EntityReleasedGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    gPRSCause [0] GPRSCause {bound},
    pDPID [1] PDPID
    OPTIONAL,
    ...
}

```

```

eventReportGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT EventReportGPRSArg {bound}
    RETURN RESULT TRUE
    ERRORS {unknownPDPID}
    CODE opcode-eventReportGPRS}
-- Direction gprsSSF -> gsmSCF,Timer Tereg
-- This operation is used to notify the gsmSCF of a GPRS session or PDP
-- context related
-- events (e.g. PDP context activation) previously requested by the
-- gsmSCF in a
-- RequestReportGPRSEventoperation.

```

```

EventReportGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    GPRSEventType [0] GPRSEventType,
    MiscGPRSInfo [1] MiscCallInfo DEFAULT
{messageType request},
    GPRSEventSpecificInformation [2] GPRSEventSpecificInformation
{bound} OPTIONAL,
    pDPID [3] PDPID OPTIONAL,
    ...
}

```

```

furnishChargingInformationGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          FurnishChargingInformationGPRSArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownPDPID}

    CODE              opcode-furnishChargingInformationGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tfcig
-- This operation is used to request the gprsSSF to generate, register a logical
record or to
-- include some information in the default logical GPRS record.
-- The registered logical record is intended for off line charging of the GPRS
session
-- or PDP Context.

FurnishChargingInformationGPRSArg {PARAMETERS-BOUND : bound} ::=
    FCIGPRSBillingChargingCharacteristics{bound}

initialDPGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          InitialDPGPRSArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingCustomerRecord |
                        missingParameter |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter}

    CODE              opcode-initialDPGPRS}
-- Direction gprsSSF -> gsmSCF,Timer Tidpg
-- This operation is used by the gprsSSF when a trigger is detected at a DP
in the GPRS state
-- machines to request instructions from the gsmSCF

InitialDPGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    ServiceKey                [0] ServiceKey,
    GPRSEventType             [1] GPRSEventType,
    mSISDN                    [2] ISDN-
AddressString,
    iMSI                      [3] IMSI,
    timeAndTimeZone           [4] TimeAndTimezone
{bound},
    gPRSMSCClass              [5] GPRSMSCClass
                                OPTIONAL,
    endUserAddress            [6] EndUserAddress {bound}
                                OPTIONAL,
    qualityOfService          [7] QualityOfService
                                OPTIONAL,
    accessPointName           [8] AccessPointName{bound}
                                OPTIONAL,
    routeingAreaIdentity      [9] RAIdentity
                                OPTIONAL,

```

```

chargingID [10] GPRSChargingID
                                OPTIONAL,
sGSNCapabilities [11] SGSNCapabilities
                                OPTIONAL,
locationInformationGPRS [12] LocationInformationGPRS
                                OPTIONAL,
pDPInitiationType [13] PDPInitiationType
                                OPTIONAL,
extensions [14] Extensions {bound}
                                OPTIONAL,
...,
gGSNAddress [15] GSN-Address
                                OPTIONAL,
secondaryPDP-context [16] NULL
                                OPTIONAL,
IMEI [17] IMEI
                                OPTIONAL
}
-- The RouteingAreaIdentity parameter is not used.
-- The receiving entity shall ignore RouteingAreaIdentity if received.
-- The RouteingAreaIdentity is conveyed in the LocationInformationGPRS
parameter.

releaseGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      ReleaseGPRSArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                   taskRefused |
                   unknownPDPID}
    CODE          opcode-releaseGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Trg
-- This operation is used to tear down an existing GPRS session or PDP Context
at any phase.

ReleaseGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    gprsCause [0] GPRSCause {bound},
    pDPID [1] PDPID
    OPTIONAL,
    ...
}

requestReportGPRSEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      RequestReportGPRSEventArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                   parameterOutOfRange |
                   systemFailure |
                   taskRefused |
                   unexpectedComponentSequence |
                   unexpectedDataValue |
                   unexpectedParameter |
                   unknownPDPID}
    CODE          opcode-requestReportGPRSEvent}
-- Direction: gsmSCF -> gprsSSF, Timer: Trrqe
-- This operation is used to request the gprsSSF to monitor for an event (e.g.,
GPRS events

```

-- such as attach or PDP Context activation), then send a notification back to the
-- gsmSCF when the event is detected.

```
RequestReportGPRSEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    gPRSEvent                               [0] SEQUENCE SIZE
(1..bound.&numOfGPRSEvents) OF GPRSEvent,
    pDPID                                   [1] PDPID
                                OPTIONAL,
    ...
}
```

-- Indicates the GPRS related events for notification.

```
ResetTimerGPRS OPERATION ::= {
    ARGUMENT          ResetTimerGPRSArg
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        parameterOutOfRange |
                        taskRefused |
                        unexpectedComponentSequence |
                        unexpectedDataValue |
                        unexpectedParameter |
                        unknownPDPID}
    CODE              opcode-resetTimerGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Trtg
-- This operation is used to request the gprsSSF to refresh an application
timer in the gprsSSF.
```

```
ResetTimerGPRSArg ::= SEQUENCE {
    timerID                               [0] TimerID
                                DEFAULT tssf,
    timervalue                             [1] TimerValue,
    ...
}
```

```
sendChargingInformationGPRS {PARAMETERS-BOUND: bound} OPERATION ::= {
    ARGUMENT          SendChargingInformationGPRSArg { bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                        unexpectedComponentSequence |
                        unexpectedParameter |
                        parameterOutOfRange |
                        systemFailure |
                        taskRefused |
                        unexpectedDataValue |
                        unknownPDPID}
    CODE              opcode-sendChargingInformationGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tscig
-- This operation is used to instruct the gprsSSF on the charging information
which the
-- gprsSSF shall send to the Mobile Station by means of GSM access signalling.
```

```
SendChargingInformationGPRSArg {PARAMETERS-BOUND: bound} ::= SEQUENCE {
    SCIGPRSBillingChargingCharacteristics [0]
SCIGPRSBillingChargingCharacteristics { bound},
    ...
}
```


END

CAP-GPRS-ReferenceNumber {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-dialogueInformation(111) version3(2)}
DEFINITIONS ::= BEGIN

EXPORTS

id-CAP-GPRS-ReferenceNumber,
CAP-GPRS-ReferenceNumber-Abstract-Syntax;

IMPORTS

Integer4
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-
network(1)
modules(0) cs1-datatypes(2) version1(0)}
;

id-CAP-GPRS-ReferenceNumber OBJECT IDENTIFIER ::= {itu-t(0) identified-
organization(4) etsi(0) mobileDomain(0) umts-network(1) as(1) cap-GPRS-
ReferenceNumber(5) version3(2)}

cap-GPRS-ReferenceNumber-Abstract-Syntax ABSTRACT-SYNTAX ::= {CAP-GPRS-
ReferenceNumber IDENTIFIED BY id-CAP-GPRS-ReferenceNumber}

CAP-GPRS-ReferenceNumber ::= SEQUENCE {
DestinationReference [0] Integer4
OPTIONAL,
OriginationReference [1] Integer4
OPTIONAL
}

-- This parameter is used to identify the relationship between SGSN and the
gsmSCF.

END -- of CAP-GPRS-ReferenceNumber

-- 8.2 gsmSCF/gprsSSF contracts, packages and ACs

-- 8.2.1 gprsSSF/gsmSCF ASN.1 module

CAP-gprsSSF-gsmSCF-pkgs-contracts-accs {itu-t(0) identified-organization(4)
etsi(0) mobileDomain(0) umts-network(1) modules(3) cap-gprsSSF-gsmSCF-pkgs-
contracts-accs(108) version3(2)}

DEFINITIONS ::= BEGIN

-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the gprsSSF - gsmSCF interface, for the
-- control of GPRS.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

PARAMETERS-BOUND,

```

    cAPSpecificBoundSet
FROM CAP-classes classes

    CONTRACT,
    OPERATION-PACKAGE,
    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

    TCMessage {}
FROM TCAPMessages tc-Messages

    APPLICATION-CONTEXT,
    dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions

    activityTestGPRS,
    applyChargingGPRS,
    applyChargingReportGPRS,
    cancelGPRS,
    connectGPRS {},
    continueGPRS,
    entityReleasedGPRS {},
    furnishChargingInformationGPRS {},
    initialDPGPRS {},
    releaseGPRS {},
    eventReportGPRS {},
    requestReportGPRSEvent {},
    resetTimerGPRS,
    sendChargingInformationGPRS {}
FROM CAP-gprsSSF-gsmSCF-ops-args gprsSSF-gsmSCF-Operations

    id-ac-CAP-gprsSSF-gsmSCF-AC,
    id-ac-CAP-gsmSCF-gprsSSF-AC,
    id-cap3GprsSsfToGsmScf,
    id-cap3GsmScfToGprsSsf,
    id-as-gprsSSF-gsmSCF-AS,
    id-as-gsmSCF-gprsSSF-AS,
    id-package-gprsScfActivation,
    id-package-gprsConnect,
    id-package-gprsContinue,
    id-package-gprsRelease,
    id-package-gprsEventHandling,
    id-package-gprsExceptionInformation,
    id-package-gprsTimer,
    id-package-gprsBilling,
    id-package-gprsCharging,
    id-package-gprsChargeAdvice,
    id-package-gprsActivityTest,
    id-package-gprsCancel,
    classes,
    ros-InformationObjects,
    tc-Messages,
    tc-NotationExtensions,
    gprsSSF-gsmSCF-Operations
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version3(2)}

```

;

-- Application Contexts

```
cap3-gprssf-scfAC APPLICATION-CONTEXT ::= {
    CONTRACT                cap3GprsSsfToScf
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES      {dialogue-abstract-syntax |
                           gprSSF-gsmSCFAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gprSSF-gsmSCF-AC}
```

```
cap3-gsmScf-gprSSSFAC APPLICATION-CONTEXT ::= {
    CONTRACT                cap3GsmScfToGprsSsf
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES      {dialogue-abstract-syntax |
                           gsmSCF-gprSSFAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gsmSCF-gprSSSF-AC}
```

-- Contracts

```
cap3GprsSsfToScf CONTRACT ::= {
-- dialogue initiated by gprSSF with InitialDPGPRS, ApplyChargingReportGPRS,
-- EntityReleaseGPRS and EventReportGPRS Operations
    INITIATOR CONSUMER OF {gprsScfActivationPackage
    {cAPSpecificBoundSet} |
                           gprsEventHandlingPackage
    {cAPSpecificBoundSet} |
                           gprsChargingPackage |
                           gprsExceptionInformationPackage
    {cAPSpecificBoundSet}}
    RESPONDER CONSUMER OF {gprsConnectPackage {cAPSpecificBoundSet} |
    {cAPSpecificBoundSet} |
                           gprsProcessingPackage |
                           gprsReleasePackage
    {cAPSpecificBoundSet} |
                           gprsEventHandlingPackage
    {cAPSpecificBoundSet} |
                           gprsTimerPackage |
                           gprsBillingPackage
    {cAPSpecificBoundSet} |
                           gprsChargingPackage |
                           gprsCancelPackage |
                           gprsChargeAdvicePackage
    {cAPSpecificBoundSet}}
    ID                      id-cap3GprsSsfToGsmScf}
```

```
cap3GsmScfToGprsSsf CONTRACT ::= {
-- dialogue initiated by gsmSCF with ApplyCharginGPRS, ActivityTestGPRS,
-- CancelGPRS, FurnishChargingInformationGPRS, ReleaseGPRS,
-- RequestReportGPRSEvent and SendChargingInformationGPRS Operations
    INITIATOR CONSUMER OF {gprsReleasePackage {cAPSpecificBoundSet} |
    {cAPSpecificBoundSet} |
                           gprsEventHandlingPackage
    {cAPSpecificBoundSet} |
                           gprsBillingPackage
    {cAPSpecificBoundSet} |
                           gprsChargingPackage |
```

```

gprsActivityTestPackage |
gprsCancelPackage |
gprsChargeAdvicePackage
{cAPSpecificBoundSet}
ID id-cap3GsmScfToGprsSsf

-- Operation Packages

gprsScfActivationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {initialDPGPRS {bound}}
ID id-package-gprsScfActivation}

gprsConnectPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {connectGPRS {bound}}
ID id-package-gprsConnect}

gprsProcessingPackage OPERATION-PACKAGE ::= {
CONSUMER INVOKES {continueGPRS }
ID id-package-gprsContinue}

gprsReleasePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {releaseGPRS {bound}}
ID id-package-gprsRelease}

gprsEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {requestReportGPRSEvent {bound}}
SUPPLIER INVOKES {eventReportGPRS {bound}}
ID id-package-gprsEventHandling}

gprsExceptionInformationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::=
{
CONSUMER INVOKES {entityReleasedGPRS {bound}}
ID id-package-gprsExceptionInformation}

gprsTimerPackage OPERATION-PACKAGE ::= {
CONSUMER INVOKES {resetTimerGPRS }
ID id-package-gprsTimer}

gprsBillingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {furnishChargingInformationGPRS {bound}}
ID id-package-gprsBilling}

gprsChargingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {applyChargingGPRS {bound}}
SUPPLIER INVOKES {applyChargingReportGPRS {bound}}
ID id-package-gprsCharging}

gprsChargeAdvicePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
CONSUMER INVOKES {sendChargingInformationGPRS {bound}}
ID id-package-gprsChargeAdvice}

gprsActivityTestPackage OPERATION-PACKAGE ::= {
CONSUMER INVOKES {activityTestGPRS}
ID id-package-gprsActivityTest}

gprsCancelPackage OPERATION-PACKAGE ::= {

```

```
CONSUMER INVOKES {cancelGPRS }
ID id-package-gprsCancel}
```

-- Abstract Syntaxes

```
gprsSSF-gsmSCFAbstractSyntax ABSTRACT-SYNTAX ::= {
  GenericGprsSSF-gsmSCF-PDUs
  IDENTIFIED BY id-as-gprsSSF-gsmSCF-AS}
```

```
GenericGprsSSF-gsmSCF-PDUs ::= TCMMessage {{GprsSsfToGsmScfInvokable},
  {GprsSsfToGsmScfReturnable}}
```

```
GprsSsfToGsmScfInvokable OPERATION ::= {
  activityTestGPRS |
  applyChargingGPRS |
  applyChargingReportGPRS |
  cancelGPRS |
  connectGPRS {cAPSpecificBoundSet} |
  entityReleasedGPRS {cAPSpecificBoundSet} |
  eventReportGPRS {cAPSpecificBoundSet} |
  furnishChargingInformationGPRS {cAPSpecificBoundSet} |
  initialDPGPRS {cAPSpecificBoundSet} |
  releaseGPRS {cAPSpecificBoundSet} |
  requestReportGPRSEvent {cAPSpecificBoundSet} |
  resetTimerGPRS |
  sendChargingInformationGPRS {cAPSpecificBoundSet}}
```

```
GprsSsfToGsmScfReturnable OPERATION ::= {
  activityTestGPRS |
  applyChargingGPRS |
  applyChargingReportGPRS |
  cancelGPRS |
  connectGPRS {cAPSpecificBoundSet} |
  continueGPRS |
  entityReleasedGPRS {cAPSpecificBoundSet} |
  furnishChargingInformationGPRS {cAPSpecificBoundSet} |
  initialDPGPRS {cAPSpecificBoundSet} |
  releaseGPRS {cAPSpecificBoundSet} |
  requestReportGPRSEvent {cAPSpecificBoundSet} |
  resetTimerGPRS |
  sendChargingInformationGPRS {cAPSpecificBoundSet}}
```

```
gsmSCF-gprsSSFGenericAbstractSyntax ABSTRACT-SYNTAX ::= {
  GenericGsmSCF-gprsSSF-PDUs
  IDENTIFIED BY id-as-gsmSCF-gprsSSF-AS}
```

```
GenericGsmSCF-gprsSSF-PDUs ::= TCMMessage {{GsmScfToGprsSsfInvokable},
  {GsmScfToGprsSsfReturnable}}
```

```
GsmScfToGprsSsfInvokable OPERATION ::= {
  activityTestGPRS |
  applyChargingGPRS |
  cancelGPRS |
  furnishChargingInformationGPRS {cAPSpecificBoundSet} |
  releaseGPRS {cAPSpecificBoundSet} |
  requestReportGPRSEvent {cAPSpecificBoundSet} |
```

```
sendChargingInformationGPRS {cAPSpecificBoundSet}}
```

```
GsmScfToGprsSsfReturnable OPERATION ::= {  
  activityTestGPRS |  
  applyChargingGPRS |  
  cancelGPRS |  
  furnishChargingInformationGPRS {cAPSpecificBoundSet} |  
  releaseGPRS {cAPSpecificBoundSet} |  
  requestReportGPRSEvent {cAPSpecificBoundSet} |  
  sendChargingInformationGPRS {cAPSpecificBoundSet}}
```

```
END
```

3GPP TS 29.078 5.0.0 (2002-06)

Technical Specification

**3rd Generation Partnership Project;
Technical Specification Group Core Network;
Customised Applications for Mobile network Enhanced Logic
(CAMEL) Phase 4;
CAMEL Application Part (CAP) specification
(Release 5)**



The present document has been developed within the 3rd Generation Partnership Project (3GPP™) and may be further elaborated for the purposes of 3GPP.

The present document has not been subject to any approval process by the 3GPP Organisational Partners and shall not be implemented. This Specification is provided for future development work within 3GPP only. The Organisational Partners accept no liability for any use of this Specification. Specifications and reports for implementation of the 3GPP™ system should be obtained via the 3GPP Organisational Partners' Publications Offices.

Keywords

UMTS, GSM, CAMEL, SS7

3GPP

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet

<http://www.3gpp.org>

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© 2000, 3GPP Organizational Partners (ARIB, CWTS, ETSI, T1, TTA, TTC).
All rights reserved.

Contents

Foreword.....	14
1 Scope.....	15
2 References.....	16
2.1 Specifications used for IMPORTS for CAP	19
3 Abbreviations.....	20
4 Interface specification for telecommunication services.....	22
4.1 General.....	22
4.1.1 Definition methodology	22
4.1.2 Example physical scenarios	22
4.1.3 CAP protocol architecture.....	29
4.1.4 Compatibility mechanisms used for CAP	30
4.1.4.1 Introduction	30
4.1.4.2 Definition of CAP compatibility mechanisms.....	31
4.1.4.2.1 Compatibility mechanism for interworking of CAP with ETSI CS2 Core INAP and ITU-T Q.1228 INAP.....	31
4.1.4.2.2 Procedures for major additions to CAP	31
4.1.4.2.3 Procedures for minor additions to CAP.....	31
4.1.4.2.4 Procedures for inclusion of network specific additions to CAP	31
4.1.5 Definition and usage of LegID.....	31
4.1.5.1 Definition of LegID	31
4.1.5.2 Allocation of LegID.....	31
4.2 SACF/MACF rules	32
4.2.1 Reflection of TC AC	32
4.2.2 Sequential/parallel execution of operations.....	32
5 Common CAP Types	33
5.1 Data types	33
5.2 Error types	49
5.3 Operation codes	51
5.4 Error codes.....	53
5.5 Classes	53
5.6 Object Identifiers (IDs).....	56
5.7 User Abort Data.....	59
6 Circuit Switched Call Control.....	61
6.1 gsmSSF/CCF - gsmSCF Interface	61
6.1.1 Operations and arguments.....	61
6.1.2 gsmSSF/gsmSCF packages, contracts and ACs.....	73
6.1.2.1 gsmSSF/gsmSCF ASN.1 module	73
6.2 gsmSCF/gsmSRF interface.....	79
6.2.1 gsmSCF/gsmSRF operations and arguments	79
6.2.2 gsmSRF/gsmSCF contracts, packages and ACs.....	81
6.2.2.1 gsmSRF/gsmSCF ASN.1 modules	81
7 SMS Control	84
7.1 SMS operations and arguments	84
7.1.1 Operation timers.....	87
7.2 SMS contracts, packages and ACs.....	88
7.2.1 SMS ASN.1 module.....	88
8 GPRS Control	91
8.1 gsmSCF/gprsSSF operations and arguments	91
8.1.1 Operation timers.....	96
8.2 gsmSCF/gprsSSF contracts, packages and ACs	97
8.2.1 gprsSSF/gsmSCF ASN.1 module.....	97

9	Application Entity procedures	101
10	Error procedures	102
10.1	Operation related error procedures	102
10.1.1	Canceled	102
10.1.1.1	General Description	102
10.1.1.1.1	Error description	102
10.1.1.2	Operations gsmSCF→gsmSRF	102
10.1.2	CancelFailed	103
10.1.2.1	General description	103
10.1.2.1.1	Error description	103
10.1.2.2	Operations gsmSCF→gsmSRF	103
10.1.3	ETCFailed	103
10.1.3.1	General description	103
10.1.3.1.1	Error description	103
10.1.3.2	Operations gsmSCF→gsmSSF	103
10.1.4	ImproperCallerResponse	104
10.1.4.1	General description	104
10.1.4.1.1	Error description	104
10.1.4.2	Operations gsmSCF→gsmSRF	104
10.1.5	MissingCustomerRecord	104
10.1.5.1	General description	104
10.1.5.1.1	Error description	104
10.1.5.2	Operations gsmSSF→gsmSCF	104
10.1.5.3	Operations gsmSRF→gsmSCF	105
10.1.5.4	Operations smsSSF→gsmSCF	105
10.1.5.5	Operations gprsSSF→gsmSCF	105
10.1.6	MissingParameter	106
10.1.6.1	General description	106
10.1.6.1.1	Error description	106
10.1.6.2	Operations gsmSCF→gsmSSF	106
10.1.6.3	Operations gsmSSF→gsmSCF	106
10.1.6.4	Operations gsmSCF→gsmSRF	106
10.1.6.5	Operations gsmSRF→gsmSCF	106
10.1.6.6	Operations smsSSF→gsmSCF	107
10.1.6.7	Operations gsmSCF → smsSSF	107
10.1.6.8	Operations gprsSSF→gsmSCF	107
10.1.6.9	Operations gsmSCF→gprsSSF	108
10.1.7	ParameterOutOfRange	108
10.1.7.1	General description	108
10.1.7.1.1	Error description	108
10.1.7.2	Operations gsmSCF→gsmSSF	108
10.1.7.3	Operations gsmSSF→gsmSCF	108
10.1.7.4	Operations gsmSCF→gsmSRF	108
10.1.7.5	Operations smsSSF → gsmSCF	108
10.1.7.6	Operations gsmSCF→smsSSF	108
10.1.7.7	Operations gprsSSF →gsmSCF	108
10.1.7.8	Operations gsmSCF→gprsSSF	108
10.1.8	RequestedInfoError	109
10.1.8.1	General description	109
10.1.8.1.1	Error description	109
10.1.8.2	Operations gsmSCF→gsmSSF	109
10.1.9	SystemFailure	109
10.1.9.1	General description	109
10.1.9.1.1	Error description	109
10.1.9.2	Operations gsmSCF→gsmSSF	109
10.1.9.3	Operations gsmSSF→gsmSCF	109
10.1.9.4	Operations gsmSCF→gsmSRF	109
10.1.9.5	Operations gsmSRF→gsmSCF	109
10.1.9.6	Operations smsSSF→gsmSCF	109
10.1.9.7	Operations gsmSCF→smsSSF	109
10.1.9.8	Operations gprsSSF →gsmSCF	109

10.1.9.9	Operations gsmSCF→gprsSSF	109
10.1.10	TaskRefused	110
10.1.10.1	General description	110
10.1.10.1.1	Error description	110
10.1.10.2	Operations gsmSCF→gsmSSF	110
10.1.10.3	Operations gsmSSF→gsmSCF	110
10.1.10.4	Operations gsmSCF→gsmSRF	110
10.1.10.5	Operations gsmSRF→gsmSCF	110
10.1.10.6	Operations smsSSF→gsmSCF	110
10.1.10.7	Operations gsmSCF→smsSSF	110
10.1.10.8	Operations gprsSSF →gsmSCF	110
10.1.10.9	Operations gsmSCF→gprsSSF	110
10.1.11	UnavailableResource	110
10.1.11.1	General description	110
10.1.11.1.1	Error description	110
10.1.11.2	Operations gsmSCF→gsmSRF	110
10.1.12	UnexpectedComponentSequence	111
10.1.12.1	General description	111
10.1.12.1.1	Error description	111
10.1.12.2	Operations gsmSCF→gsmSSF	111
10.1.12.3	Operations gsmSSF→gsmSCF	111
10.1.12.4	Operations gsmSCF→gsmSRF (applicable for direct gsmSCF-gsmSRF case only)	111
10.1.12.5	Operations gsmSRF→gsmSCF	111
10.1.12.6	Operations smsSSF →gsmSCF	111
10.1.12.7	Operations gsmSCF→smsSSF	111
10.1.12.8	Operations gprsSSF →gsmSCF	112
10.1.12.9	Operations gsmSCF→gprsSSF	112
10.1.13	UnexpectedDataValue	112
10.1.13.1	General description	112
10.1.13.1.1	Error description	112
10.1.13.2	Operations gsmSCF→gsmSSF	112
10.1.13.3	Operations gsmSSF→gsmSCF	112
10.1.13.4	Operations gsmSCF→gsmSRF	112
10.1.13.5	Operations gsmSRF→gsmSCF	112
10.1.13.6	Operations smsSSF→gsmSCF	112
10.1.13.7	Operations gsmSCF→smsSSF	112
10.1.13.8	Operations gprsSSF →gsmSCF	112
10.1.13.9	Operations gsmSCF→gprsSSF	112
10.1.14	UnexpectedParameter	113
10.1.14.1	General description	113
10.1.14.1.1	Error description	113
10.1.14.2	Operations gsmSCF→gsmSSF	113
10.1.14.3	Operations gsmSSF→gsmSCF	113
10.1.14.4	Operations gsmSCF→gsmSRF	113
10.1.14.5	Operations gsmSRF→gsmSCF	113
10.1.14.6	Operations smsSSF→gsmSCF	113
10.1.14.7	Operations gsmSCF→smsSSF	113
10.1.14.8	Operations gprsSSF →gsmSCF	113
10.1.14.9	Operations gsmSCF→gprsSSF	113
10.1.15	UnknownLegID	113
10.1.15.1	General description	113
10.1.15.1.1	Error description	113
10.1.15.2	Operations gsmSCF→gsmSSF	113
10.1.16	UnknownCSID	114
10.1.16.1	General description	114
10.1.16.1.1	Error description	114
10.1.16.2	Operations gsmSCF→gsmSSF	114
10.1.17	UnknownPDPID	114
10.1.17.1	General description	114
10.1.17.1.1	Error description	114
10.1.17.2	Operations gprsSSF→gsmSCF	114
10.1.17.3	Operations gsmSCF→gprsSSF	114

10.2	Entity related error procedures	114
10.2.1	Expiration of Tssf.....	114
10.2.1.1	General description.....	114
10.2.1.1.1	Error description	114
10.2.1.2	Procedures gsmSSF→gsmSCF	114
10.2.1.3	Procedures gprsSSF→gsmSCF	115
10.2.1.4	Procedures smsSSF→gsmSCF	115
10.2.2	Expiration of Tsrfs.....	115
10.2.2.1	General Description.....	115
10.2.2.1.1	Error description	115
10.2.2.2	Procedures description.....	115
11	Detailed operation procedures for circuit switched call control	116
11.1	ActivityTest procedure	116
11.1.1	General description	116
11.1.1.1	Parameters	116
11.1.2	Responding entity (gsmSSF, gsmSRF or assist gsmSSF).....	116
11.1.2.1	Normal procedure.....	116
11.1.2.2	Error handling.....	117
11.2	ApplyCharging procedure.....	117
11.2.1	General description	117
11.2.1.1	Parameters	117
11.2.2	Responding entity (gsmSSF).....	118
11.2.2.1	Normal procedure.....	118
11.2.2.2	Error handling.....	119
11.3	ApplyChargingReport procedure.....	119
11.3.1	General description	119
11.3.1.1	Parameters	119
11.3.2	Invoking entity (gsmSSF)	120
11.3.2.1	Normal procedure.....	120
11.3.2.2	Error handling.....	121
11.4	AssistRequestInstructions procedure	121
11.4.1	General description	121
11.4.1.1	Parameters	121
11.4.2	Invoking entity (gsmSSF/gsmSRF).....	121
11.4.2.1	Normal procedure.....	121
11.4.2.2	Error handling.....	121
11.5	CallGap procedure	122
11.5.1	General description	122
11.5.1.1	Parameters	122
11.5.2	Responding entity (gsmSSF).....	124
11.5.2.1	Normal procedure.....	124
11.5.2.2	Error handling.....	125
11.6	CallInformationReport procedure.....	125
11.6.1	General description	125
11.6.1.1	Parameters	125
11.6.2	Invoking entity (gsmSSF)	126
11.6.2.1	Normal procedure.....	126
11.6.2.2	Error handling.....	126
11.7	CallInformationRequest procedure.....	126
11.7.1	General description	126
11.7.1.1	Parameters	127
11.7.2	Responding entity (gsmSSF).....	127
11.7.2.1	Normal procedure.....	127
11.7.2.2	Error handling.....	127
11.8	Cancel procedure	128
11.8.1	General description	128
11.8.1.1	Parameters	128
11.8.2	Responding entity (gsmSRF)	128
11.8.2.1	Normal procedure.....	128
11.8.2.2	Error handling.....	128
11.8.3	Responding entity (gsmSSF).....	128

11.8.3.1	Normal procedure	128
11.8.3.2	Error handling.....	129
11.9	Connect procedure	129
11.9.1	General description	129
11.9.1.1	Parameters	129
11.9.2	Responding entity (gsmSSF).....	130
11.9.2.1	Normal procedure.....	130
11.9.2.2	Error handling.....	131
11.10	ConnectToResource procedure.....	131
11.10.1	General description	131
11.10.1.1	Parameters	131
11.10.2	Responding entity (gsmSSF).....	131
11.10.2.1	Normal procedure.....	131
11.10.2.2	Error handling.....	132
11.11	Continue procedure.....	132
11.11.1	General description	132
11.11.1.1	Parameters	132
11.11.2	Responding entity (gsmSSF).....	132
11.11.2.1	Normal procedure.....	132
11.11.2.2	Error handling.....	132
11.12	ContinueWithArgument Procedure	132
11.12.1	General description	132
11.12.1.1	Parameters	133
11.12.2	Responding entity (gsmSSF).....	134
11.12.2.1	Normal procedure.....	134
11.12.2.2	Error handling.....	134
11.13	DisconnectForwardConnection procedure.....	134
11.13.1	General Description.....	134
11.13.1.1	Parameters	134
11.13.2	Responding entity (gsmSSF).....	134
11.13.2.1	Normal procedure.....	134
11.13.2.2	Error handling.....	135
11.14	DisconnectForwardConnectionWithArgument procedure.....	135
11.14.1	General Description.....	135
11.14.1.1	Parameters	135
11.14.2	Responding entity (gsmSSF).....	135
11.14.2.1	Normal procedure.....	135
11.14.2.2	Error handling.....	136
11.15	DisconnectLeg procedure	136
11.15.1	General Description.....	136
11.15.1.1	Parameters	136
11.15.2	Responding entity (gsmSSF).....	136
11.15.2.1	Normal procedure.....	136
11.15.2.2	Error handling.....	136
11.16	EntityReleased procedure	137
11.16.1	General Description.....	137
11.16.1.1	Parameters	137
11.16.2	Invoking entity (gsmSSF)	137
11.16.2.1	Normal procedure.....	137
11.16.2.2	Error handling.....	137
11.17	EstablishTemporaryConnection procedure.....	138
11.17.1	General Description.....	138
11.17.1.1	Parameters	138
11.17.2	Responding entity (gsmSSF).....	139
11.17.2.1	Normal procedure.....	139
11.17.2.2	Error handling.....	139
11.18	EventNotificationCharging procedure	139
11.18.1	General Description.....	139
11.18.1.1	Parameters	139
11.18.2	Invoking entity (gsmSSF)	141
11.18.2.1	Normal procedure.....	141
11.18.2.2	Error handling.....	142

11.19	EventReportBCSM procedure	142
11.19.1	General description	142
11.19.1.1	Parameters	142
11.19.2	Invoking entity (gsmSSF)	144
11.19.2.1	Normal procedure	144
11.19.2.2	Error handling.....	144
11.20	FurnishChargingInformation procedure	144
11.20.1	General description	144
11.20.1.1	Parameters	145
11.20.2	Responding entity (gsmSSF).....	145
11.20.2.1	Normal procedure.....	145
11.20.2.2	Error handling.....	145
11.21	InitialDP procedure.....	146
11.21.1	General description	146
11.21.1.1	Parameters	146
11.21.2	Invoking entity (gsmSSF)	148
11.21.2.1	Normal procedure.....	148
11.21.2.2	Error handling.....	149
11.22	InitiateCallAttempt procedure	149
11.22.1	General Description.....	149
11.22.1.1	Parameters	149
11.22.1.1.1	Argument Parameters	149
11.22.1.1.2	Result Parameters	149
11.22.2	Responding entity (gsmSSF).....	150
11.22.2.1	Normal procedure.....	150
11.22.2.2	Error handling.....	150
11.23	MoveLeg procedure.....	150
11.23.1	General Description.....	150
11.23.1.1	Parameters	150
11.23.2	Responding entity (gsmSSF).....	150
11.23.2.1	Normal procedure.....	150
11.23.2.2	Error handling.....	151
11.24	PlayAnnouncement procedure.....	151
11.24.1	General description	151
11.24.1.1	Parameters	151
11.24.2	Responding entity (gsmSRF)	152
11.24.2.1	Normal procedure.....	152
11.24.2.2	Error handling.....	152
11.25	PlayTone procedure	153
11.25.1	General description	153
11.25.1.1	Parameters	153
11.25.2	Responding entity (gsmSSF).....	153
11.25.2.1	Normal procedure.....	153
11.25.2.2	Error handling.....	153
11.26	PromptAndCollectUserInformation procedure.....	154
11.26.1	General description	154
11.26.1.1	Parameters	154
11.26.2	Responding entity (gsmSRF)	156
11.26.2.1	Normal procedure.....	156
11.26.2.2	Error handling.....	157
11.27	ReleaseCall procedure	157
11.27.1	General description	157
11.27.1.1	Parameters	158
11.27.2	Responding entity (gsmSSF).....	158
11.27.2.1	Normal procedure.....	158
11.27.2.2	Error handling.....	158
11.28	RequestNotificationChargingEvent procedure	158
11.28.1	General Description.....	158
11.28.1.1	Parameters	158
11.28.2	Responding entity (gsmSSF).....	160
11.28.2.1	Normal procedure.....	160
11.28.2.2	Error handling.....	160

11.29	RequestReportBCSMEEvent procedure	160
11.29.1	General description	160
11.29.1.1	Parameters	161
11.29.2	Responding entity (gsmSSF).....	162
11.29.2.1	Normal procedure.....	162
11.29.2.2	Error handling.....	162
11.30	ResetTimer procedure.....	162
11.30.1	General description	162
11.30.1.1	Parameters	163
11.30.2	Responding entity (gsmSSF).....	163
11.30.2.1	Normal procedure.....	163
11.29.2.2	Error handling.....	163
11.31	SendChargingInformation procedure.....	163
11.31.1	General description	163
11.31.1.1	Parameters	163
11.31.2	Responding entity (gsmSSF).....	164
11.31.2.1	Normal procedure.....	164
11.31.2.2	Error handling.....	165
11.32	SpecializedResourceReport procedure	165
11.32.1	General description	165
11.32.1.1	Parameters	165
11.32.2	Invoking entity (gsmSRF).....	165
11.32.2.1	Normal procedure.....	165
11.32.2.2	Error handling.....	165
11.33	SplitLeg Procedure	166
11.33.1	General Description.....	166
11.33.1.1	Parameters	166
11.33.2	Responding entity (gsmSSF).....	166
11.33.2.1	Normal procedure.....	166
11.33.2.2	Error handling.....	166
12	Detailed operation procedures for SMS control	167
12.1	ConnectSMS procedure	167
12.1.1	General description	167
12.1.1.1	Parameters	167
12.1.2	Responding entity (smsSSF)	167
12.1.2.1	Normal procedure.....	167
12.1.2.2	Error handling.....	168
12.2	ContinueSMS procedure.....	168
12.2.1	General description	168
12.2.1.1	Parameters	168
12.2.2	Responding entity (smsSSF)	168
12.2.2.1	Normal procedure.....	168
12.2.2.2	Error handling.....	168
12.3	EventReportSMS procedure	168
12.3.1	General description	168
12.3.1.1	Parameters	169
12.3.2	Invoking entity (smsSSF).....	169
12.3.2.1	Normal procedure.....	169
12.3.2.2	Error handling.....	169
12.4	FurnishChargingInformationSMS procedure	169
12.4.1	General description	169
12.4.1.1	Parameters	169
12.4.2	Responding entity (smsSSF)	170
12.4.2.1	Normal procedure.....	170
12.4.2.2	Error handling.....	170
12.5	InitialDPSMS procedure.....	170
12.5.1	General description	170
12.5.1.1	Parameters	170
12.5.2	Invoking entity (smsSSF).....	171
12.5.2.1	Normal procedure.....	171
12.5.2.2	Error handling.....	172

12.6	ReleaseSMS procedure	172
12.6.1	General description	172
12.6.1.1	Parameters	172
12.6.2	Responding entity (smsSSF)	172
12.6.2.1	Normal procedure	172
12.6.2.2	Error handling.....	173
12.7	RequestReportSMSEvent procedure	173
12.7.1	General description	173
12.7.1.1	Parameters	173
12.7.2	Responding entity (smsSSF)	173
12.7.2.1	Normal procedure	173
12.7.2.2	Error handling.....	173
12.8	ResetTimerSMS procedure.....	173
12.8.1	General description	173
12.8.1.1	Parameters	174
12.8.2	Responding entity (smsSSF)	174
12.8.2.1	Normal procedure	174
12.8.2.2	Error handling.....	174
13	Detailed operation procedures for GPRS control	175
13.1	ActivityTestGPRS procedure	175
13.1.1	General description	175
13.1.1.1	Parameters	175
13.1.2	Responding entity (gprsSSF)	175
13.1.2.1	Normal procedure	175
13.1.2.2	Error handling.....	175
13.2	ApplyChargingGPRS procedure.....	176
13.2.1	General description	176
13.2.1.1	Parameters	176
13.2.2	Responding entity (gprsSSF)	176
13.2.2.1	Normal procedure	176
13.2.2.2	Error handling.....	176
13.3	ApplyChargingReportGPRS procedure.....	177
13.3.1	General description	177
13.3.1.1	Parameters	177
13.3.2	Invoking entity (gprsSSF)	179
13.3.2.1	Normal procedure	179
13.3.2.2	Error handling.....	179
13.4	CancelGPRS procedure	179
13.4.1	General description	179
13.4.1.1	Parameters	179
13.4.2	Responding entity (gprsSSF)	180
13.4.2.1	Normal procedure	180
13.4.2.2	Error handling.....	180
13.5	ConnectGPRS procedure	180
13.5.1	General description	180
13.5.1.1	Parameters	180
13.5.2	Responding entity (gprsSSF)	180
13.5.2.1	Normal procedure	180
13.5.2.2	Error handling.....	181
13.6	ContinueGPRS procedure.....	181
13.6.1	General description	181
13.6.1.1	Parameters	181
13.6.2	Responding entity (gprsSSF)	181
13.6.2.1	Normal procedure	181
13.6.2.2	Error handling.....	181
13.7	EntityReleasedGPRS procedure	182
13.7.1	General description	182
13.7.1.1	Parameters	182
13.7.2	Invoking entity (gprsSSF)	182
13.7.2.1	Normal procedure	182
13.7.2.2	Error handling.....	182

13.8	EventReportGPRS procedure	182
13.8.1	General description	182
13.8.1.1	Parameters	183
13.8.2	Invoking entity (gprsSSF)	183
13.8.2.1	Normal procedure	183
13.8.2.2	Error handling.....	184
13.9	FurnishChargingInformationGPRS procedure	184
13.9.1	General description	184
13.9.1.1	Parameters	184
13.9.2	Responding entity (gprsSSF)	184
13.9.2.1	Normal procedure	184
13.9.2.2	Error handling.....	185
13.10	InitialDPGPRS procedure.....	185
13.10.1	General description	185
13.10.1.1	Parameters	185
13.10.2	Invoking entity (gprsSSF)	186
13.10.2.1	Normal procedure	186
13.10.2.2	Error handling.....	186
13.11	ReleaseGPRS procedure	187
13.11.1	General description	187
13.11.1.1	Parameters	187
13.11.2	Responding entity (gprsSSF)	187
13.11.2.1	Normal procedure	187
13.11.2.2	Error handling.....	187
13.12	RequestReportGPRSEvent procedure	187
13.12.1	General description	187
13.12.1.1	Parameters	188
13.12.2	Responding entity (gprsSSF)	188
13.12.2.1	Normal procedure	188
13.12.2.2	Error handling.....	188
13.13	ResetTimerGPRS procedure.....	188
13.13.1	General description	188
13.13.1.1	Parameters	188
13.13.2	Responding entity (gprsSSF)	189
13.13.2.1	Normal procedure	189
13.13.2.2	Error handling.....	189
13.14	SendChargingInformationGPRS Procedure	189
13.14.1	General description	189
13.14.1.1	Parameters	189
13.14.2	Responding Entity (gprsSSF).....	190
13.14.2.1	Normal Procedure.....	190
13.14.2.2	Error handling.....	190
14	Services assumed from lower layers.....	191
14.1	Services assumed from TC	191
14.1.1	Common procedures	191
14.1.1.1	Normal procedures	191
14.1.1.2	Abnormal procedures	192
14.1.1.3	Dialogue handling	193
14.1.1.3.1	Dialogue establishment.....	193
14.1.1.3.2	Dialogue continuation.....	194
14.1.1.3.3	Dialogue termination	194
14.1.1.3.4	User abort.....	194
14.1.1.3.5	Provider abort	195
14.1.1.3.6	Mapping to TC dialogue primitives	195
14.1.1.3.7	Default mapping to TC dialogue parameters	196
14.1.1.4	Component handling.....	196
14.1.1.4.1	Procedures for CAP Operations.....	196
14.1.1.4.2	Mapping to TC component primitives	198
14.1.1.4.3	Default mapping to TC component parameters	199
14.1.2	gsmSSF-gsmSCF interfaces.....	200
14.1.2.1	Normal procedures	200

14.1.2.1.1	gsmSSF-to-gsmSCF messages.....	200
14.1.2.1.2	gsmSCF-to-gsmSSF messages.....	201
14.1.2.1.3	smsSSF -to-gsmSCF SMS related messages	201
14.1.2.1.4	gsmSCF-to-smsSSF SMS related messages	201
14.1.2.1.5	Use of dialogue handling services.....	201
14.1.2.2	Abnormal procedures	202
14.1.2.2.1	gsmSCF-to-gsmSSF/gsmSRF messages	202
14.1.2.2.2	gsmSSF/gsmSRF/ -to-gsmSCF messages.....	202
14.1.2.2.3	gsmSCF-to-smsSSF SMS related messages	202
14.1.2.2.4	smsSSF-to-gsmSCF SMS related messages	202
14.1.2.2.5	Use of dialogue handling services.....	203
14.1.2.3	Dialogue handling	203
14.1.2.3.1	Dialogue establishment.....	203
14.1.2.3.2	Dialogue continuation.....	203
14.1.2.3.3	Dialogue termination	203
14.1.2.3.4	User abort.....	203
14.1.2.3.5	Provider abort	203
14.1.2.3.6	Mapping to TC dialogue primitives	203
14.1.2.4	Component Handling.....	204
14.1.2.4.1	Procedures for CAP Operations.....	204
14.1.2.4.2	Mapping to TC component parameters.....	204
14.1.3	gsmSCF-gsmSRF interface	204
14.1.3.1	Normal procedures	204
14.1.3.1.1	gsmSCF-to/from-gsmSRF messages	204
14.1.3.1.2	Abnormal procedures.....	205
14.1.3.1.3	Dialogue handling.....	205
14.1.3.1.3.1	Dialogue establishment	205
14.1.3.1.3.2	Dialogue continuation	205
14.1.3.1.3.3	Dialogue termination.....	205
14.1.3.1.3.4	User abort.....	205
14.1.3.1.3.5	Provider abort.....	205
14.1.3.1.3.6	Mapping to TC dialogue primitives	205
14.1.4.2	Component handling.....	206
14.1.4.2.1	Procedures for CAP Operations.....	206
14.1.4.2.2	Mapping to TC component parameters.....	206
14.1.4	gprsSSF-gsmSCF interface	206
14.1.4.1	Normal procedures	206
14.1.4.1.1	TC-dialogues and relationships.....	206
14.1.4.1.2	Use of the GPRS Reference	206
14.1.4.1.3	gprsSSF-to-gsmSCF messages.....	206
14.1.4.1.4	gsmSCF-to-gprsSSF messages.....	208
14.1.4.2	Abnormal procedures	208
14.1.4.2.1	gsmSCF-to-gprsSSF messages.....	208
14.1.4.2.2	gprsSSF-to-gsmSCF messages.....	208
14.1.4.2.3	Default GPRS Handling.....	209
14.2	Services assumed from SCCP.....	209
14.2.1	Normal procedures	209
14.2.2	Service functions from SCCP	210
14.2.2.1	SCCP connectionless services	210
14.2.2.1.1	Sub-System Number (SSN)	210
14.2.2.1.2	Addressing	210
14.2.2.1.3	Sequence control.....	214
14.2.2.1.4	Return on error.....	214
14.2.2.1.5	Segmentation / reassembly.....	214
14.2.2.1.6	Congestion control.....	214
14.2.2.2	SCCP connection oriented services	214
14.2.2.3	SCCP management.....	215

Annex A (normative): Mapping between CAP and ISUP	216
A.1 InitialDP operation.....	216
A.2 ContinueWithArgument operation.....	217
A.3 Connect operation.....	218
A.4 AssistRequestInstructions operation.....	221
A.5 ConnectToResource operation.....	222
A.6 EstablishTemporaryConnection operation.....	223
A.7 ReleaseCall operation	224
A.8 InitiateCallAttempt operation	225
Annex B (normative): Mapping between CAP and MAP and mapping between CAP and SM-CP....	226
B.1 Mapping between CAP and MAP.....	227
B.1.1 MO-SMS	227
B.1.1.1 ConnectSMS operation	227
B.1.1.2 EventReportSMS operation.....	227
B.1.2 MT-SMS	227
B.1.2.1 InitialDPSMS operation (MT-SMS)	227
B.1.2.2 ReleaseSMS operation (MT-SMS)	228
B.2 Mapping between CAP and SM-CP	228
B.2.1 MO-SMS	228
B.2.1.1 InitialDPSMS operation	228
B.2.1.2 ReleaseSMS operation	228
B.2.2 MT-SMS	228
B.2.2.1 ConnectSMS operation	228
B.2.2.2 EentReportSMS operation.....	229
Annex C (informative): Change history.....	230

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The present document identifies the 3G system specifications for Rel-5.

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document specifies the CAMEL Application Part (CAP) supporting the fourth phase of the network feature Customized Applications for Mobile network Enhanced Logic. CAP is based on a sub-set of the ETSI Core INAP CS-2 as specified by ETSI EN 301 140-1 [26]. Descriptions and definitions provided by ETSI EN 301 140-1 [26] are directly referenced by this standard in the case no additions or clarifications are needed for the use in the CAP.

2 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 TS 09.12: "Application of ISDN User Part (ISUP) version 2 for the ISDN-Public Land Mobile Network (PLMN) signalling interface; Part 1: Protocol specification".
- [2] 3GPP TS 22.024: "Description of Charge Advice Information (CAI)".
- [3] 3GPP TS 22.078: "Customised Applications for Mobile network Enhanced Logic (CAMEL); Service description, Stage 1".
- [4] 3GPP TS 22.115: "Service Aspects Charging and Billing".
- [5] 3GPP TS 23.003: "Numbering, addressing and identification".
- [6] 3GPP TS 23.040: "Technical realization of the Short Message Service (SMS); Point-to-Point (PP)".
- [7] 3GPP TS 23.078: " Customised Applications for Mobile network Enhanced Logic (CAMEL) Phase 3 - Stage 2".
- [8] 3GPP TS 23.079: "Support of Optimal Routeing (SOR); Technical realization".
- [9] 3GPP TS 24.008: "Digital cellular telecommunications system (Phase 2+); Mobile radio interface layer 3 specification (3GPP TS 24.008)".
- [10] 3GPP TS 24.011: "Point-toPoint (PP) Short Message Service (SMS); support on mobile radio interface".
- [11] 3GPP TS 29.002: "Digital cellular telecommunications system (Phase 2+); Mobile Application Part (MAP) specification (3GPP TS 29.002)".
- [12] 3GPP TS 29.060: "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp Interface".
- [13] 3GPP TS 32.205: "Telecommunication Management; Charging and billing; 3G call and event data for the Circuit Switched (CS) domain".
- [14] 3GPP TS 32.215: "Telecommunication Management; Charging and billing; 3G call and event data for the Packet Switched (PS) domain".
- [15] – [20] Spare
- [21] ETSI ES 201 296: "Integrated Services Digital Network (ISDN); Signalling System No.7; ISDN User Part (ISUP); Signalling aspects of charging".
- [22] ETSI ETS 300 287-1: "Integrated Services Digital Network (ISDN); Signalling System No.7; Transaction Capabilities (TC) version 2; Part 1: Protocol specification [ITU-T Recommendations Q.771 to Q.775 (1993), modified]".
- [23] ETSI EN 300 356-1: "Integrated Services Digital Network (ISDN); Signalling System No.7; ISDN User Part (ISUP) version 3 for the international interface; Part 1: Basic services [ITU-T Recommendations Q.761 to Q.764 (1997), modified]".

- [24] ETSI ETS 300 374-1: "Intelligent Network (IN); Intelligent Network Capability Set 1 (CS1); Core Intelligent Network Application Protocol (INAP); Part 1: Protocol specification".
- [25] ETSI EN 300 403-1: "Integrated Services Digital Network (ISDN); Digital Subscriber Signalling System No. one (DSS1) protocol; Signalling network layer for circuit-mode basic call control; Part 1: Protocol specification [ITU-T Recommendation Q.931 (1993), modified]".
- [26] ETSI EN 301 140-5: "Intelligent Network (IN); Intelligent Network Application Protocol (INAP); Capability Set 2 (CS2); Part 1: Protocol Specification".
- [27] – [40] Spare
- [41] ITU-T Recommendation Q.71: "ISDN circuit mode switched bearer services".
- [42] ITU-T Recommendation Q.713: "Specifications of Signalling System No.7; SCCP formats and codes".
- [43] ITU-T Recommendation Q.714: "Specifications of Signalling System No.7; Signalling Connection Control Part procedures".
- [44] ITU-T Recommendation Q.762: "General function of messages and signals of the ISDN user part of signalling system no.7".
- [45] ITU-T Recommendation Q.763: "Formats and codes of the ISDN user part of Signalling System No.7".
- [46] ITU-T Recommendation Q.773: "Specifications of Signalling System No.7; Transaction capabilities formats and encoding".
- [47] ITU-T Recommendation Q.850: "Usage of cause and location in the digital subscriber signalling system no.1 and the signalling system no.7 ISDN user part".
- [48] ITU-T Recommendation Q.932: "Digital subscriber Signalling System No.1 (DSS 1) - Generic procedures for the control of ISDN supplementary services".
- [49] ITU-T Recommendation Q.1218: "Interface Recommendation for Intelligent Network CS-1".
- [50] ITU-T Recommendation Q.1224: "Distributed functional plane for intelligent network CS2".
- [51] ITU-T Recommendation Q.1228: "Interface ITU-T Recommendation for intelligent network CS2".
- [52] ITU-T Recommendation Q.1400: "Architecture framework for the development of signalling and organization, administration and maintenance protocols using OSI principles".
- [53] ITU-T Recommendation X.680: "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation".
- [54] ITU-T Recommendation X.681: "Information technology – Abstract Syntax Notation One (ASN.1): Information object specification".
- [55] ITU-T Recommendation X.682: "Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications".
- [56] ITU-T Recommendation X.683: "Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification".
- [57] ITU-T Recommendation X.690: "ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)".
- [58] ITU-T Recommendation X.880: "Data networks and open system communication - Open System Interconnection - Service definitions - Remote operations: Concepts, model and notation".
- [59] ITU-T Recommendation X.881: "Data networks and open system communication - Open System Interconnection - Service definitions - Remote operations: OSI Realizations – Remote Operations Service Element (ROSE) service definition".

- [60] ITU-T Recommendation X.882: "Data networks and open system communication - Open System Interconnection - Service definitions - Remote operations: OSI Realizations – Remote Operations Service Element (ROSE) protocol specification".
- [61] – [80] Spare
- [81] ISO 9545 (1989): "Information technology - Open Systems Interconnection - Application Layer structure".
- [82] – [90] Spare
- [91] ANSI T1.112-1996: "American National Standards for Telecommunications- Signalling System Number 7 (SS7) - Signalling Connection Control Part (SCCP)".
- [92] ANSI T1.113-1995: "American National Standards for Telecommunications- Signalling System Number 7 (SS7) - ISDN User Part".

2.1 Specifications used for IMPORTS for CAP

The following table lists the modules from which CAP V4 imports. For each module, the table indicates in which formal specification this module can be found.

Table 2-1: Module IMPORTS specifications

Module Name	Specification	Ref
CS1-DataTypes {itu-titu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3) cs1-datatypes(2) version1(0)}	ETS 300 374-1	[24]
CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3) in-cs2-datatypes (0) version1(0)}	EN 301 140-1	[26]
MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}	3GPP TS 29.002	[11]
MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}	3GPP TS 29.002	[11]
MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) gsm-Network(1) modules(3) map-CH-DataTypes(13) version6(6)}	3GPP TS 29.002	[11]
MAP-ExtensionDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) gsm-Network(1) modules(3) map-ExtensionDataTypes(21) version6(6)}	3GPP TS 29.002	[11]
TCAPMessages {itu-t recommendation q 773 modules(2) messages(1) version3(3)}	ITU-T Q.773	[46]
Remote-Operations-Information-Objects {joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)}	ITU-T X.880	[58]
TC-Notation-Extensions {itu-t recommendation q 775 modules(2) notation-extension (4) version1(1)}	ETS 300 287-1	[22]

3 Abbreviations

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

AC	Application Context
AE	Application Entity
AEI	Application Entity Invocation
APDU	Application Protocol Data Unit
ASE	Application Service Element
ASN.1	Abstract Syntax Notation One
BCSM	Basic Call State Model
CAP	CAMEL Application Part
CCF	Call Control Function
CCITT	International Telegraph and Telephone Consultative Committee
CS1	Capability Set 1
CS2	Capability Set 2
CS	Call Segment
CSA	Call Segment Association
CSI	CAMEL Subscription Information
DP	Detection Point
DSS1	Digital Subscriber Signalling System No. One
EDP	Event Detection Point
EDP-N	Event Detection Point - Notification
EDP-R	Event Detection Point - Request
FE	Functional Entity
FEAM	Functional Entity Access Manager
ffs	for further study
FSM	Finite State Model
gprsSSF	GPRS Service Switching Function
gsmSCF	GSM Service Control Function
gsmSRF	GSM Specialized Resource Function
gsmSSF	GSM Service Switching Function
GT	Global Title
ID	IDentifier
IN	Intelligent Network
INAP	Intelligent Network Application Protocol
IP	Intelligent Peripheral
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
ITU-T	International Telecommunication Union – Telecommunication Standardization Sector
LE	Local Exchange
MACF	Multiple Association Control Function
MO	Mobile Originated
MS	Mobile Station
MSC	Mobile services Switching Centre
MT	Mobile Terminated
MTP	Message Transfer Part
NA	North American
O-BCSM	Originating BCSM
PDU	Protocol Data Unit
PE	Physical Entity
PIA	Point In Association
PIC	Point In Call
PLMN	Public Land Mobile Network
PSTN	Public Switched Telecommunication Network
ROS	Remote Operations Service
ROSE	ROS Element
SACF	Single Association Control Function
SAO	Single Association Object
SCCP	Signalling Connection Control Part

SCP	Service Control Point
SDL	System Description Language
SL	Service Logic
SLP	Service Logic Program
SLPI	Service Logic Program Instance
SM	Short Message
SM-CP	Short Message Control Protocol
SMS	Short Message Service
SMSC	Short Message Service Centre
smsSSF	Short Message Service Service Switching Function
SMF	Service Management Function
SRME	gsmSRF Management Entity
SRSM	gsmSRF Call State Model
SS7	Signalling System no. 7
smsSSF	SMS Service Switching Function
SSME	gsmSSF Management Entity
SSN	Sub-System Number
SSP	Service Switching Point
T-BCSM	Terminating BCSM
TC	Transaction Capabilities
TCAP	Transaction Capabilities Application Part
TDP	Trigger Detection Point
TDP-R	Trigger Detection Point - Request

4 Interface specification for telecommunication services

4.1 General

4.1.1 Definition methodology

The definition of the protocol can be split into three sections:

- the definition of the Single Association Control Function (SACF)/Multiple Association Control Function (MACF) rules for the protocol;
- the definition of the operations transferred between entities;
- the definition of the actions taken at each entity.

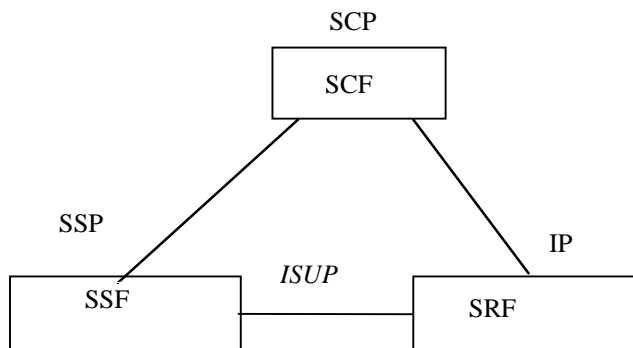
The SACF/MACF rules are defined in prose. The operation definitions are in Abstract Syntax Notation One (ASN.1) For details on ASN.1 refer to ITU-T Recommendation X.680 [53] and ITU-T Recommendation X.681 [54]. The actions are defined in terms of state transition diagrams. Further guidance on the actions to be performed on receipt of an operation can be gained from the description of the relevant information flow in ITU-T Recommendation Q.1224 [50].

The CAMEL Application Part (CAP) is a ROS Element (ROSE) user protocol. Refer to ITU-T Recommendation X.880 [58], ITU-T Recommendation X.881 [59] and ITU-T Recommendation X.882 [60] for Remote Operations. The ROSE protocol is contained within the component sublayer of Transaction Capabilities Application Part (TCAP) (see ETS 300 287-1 [22]) and Digital Subscriber Signalling System No One (DSS1) (see ITU-T Recommendation Q.932 [48]). At present, the ROSE Application Protocol Data Units (APDUs) are conveyed in transaction sublayer messages in Signalling System no. 7 (SS7) and in the EN 300 403-1 [25] REGISTER, FACILITY and call control messages in DSS1. Other supporting protocols may be added at a later date.

The CAP (as a ROSE user) and the ROSE protocol are specified in ASN.1, as defined in ITU-T Recommendation X.680 [53], ITU-T Recommendation X.681 [54], ITU-T Recommendation X.682 [55] and ITU-T Recommendation X.683 [56]. The encoding of the resulting Protocol Data Units (PDUs) shall be done in accordance with the Basic Encoding Rules as defined in ITU-T Recommendation X.690 [57].

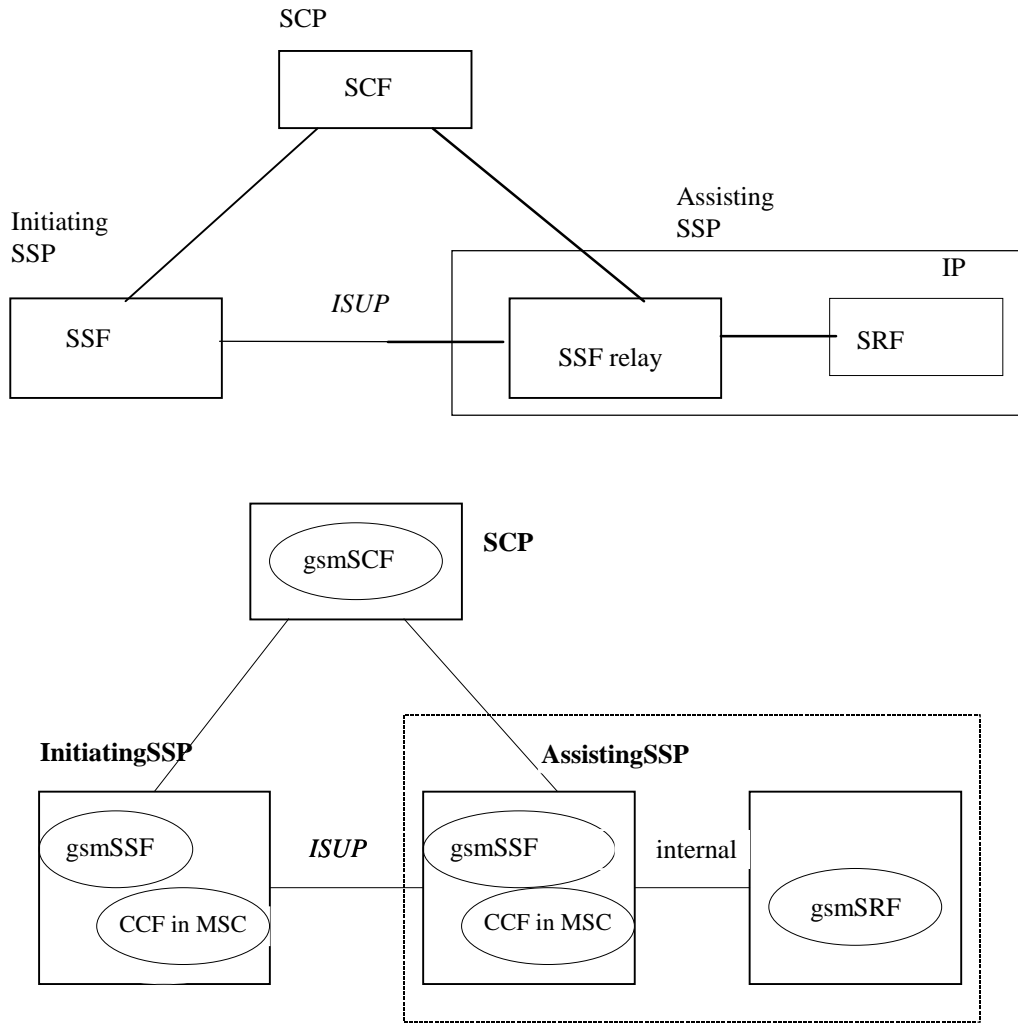
4.1.2 Example physical scenarios

The reader is referred to Intelligent Network Capability Set 1 (CS1) Core INAP [24] for details of the example physical scenarios.



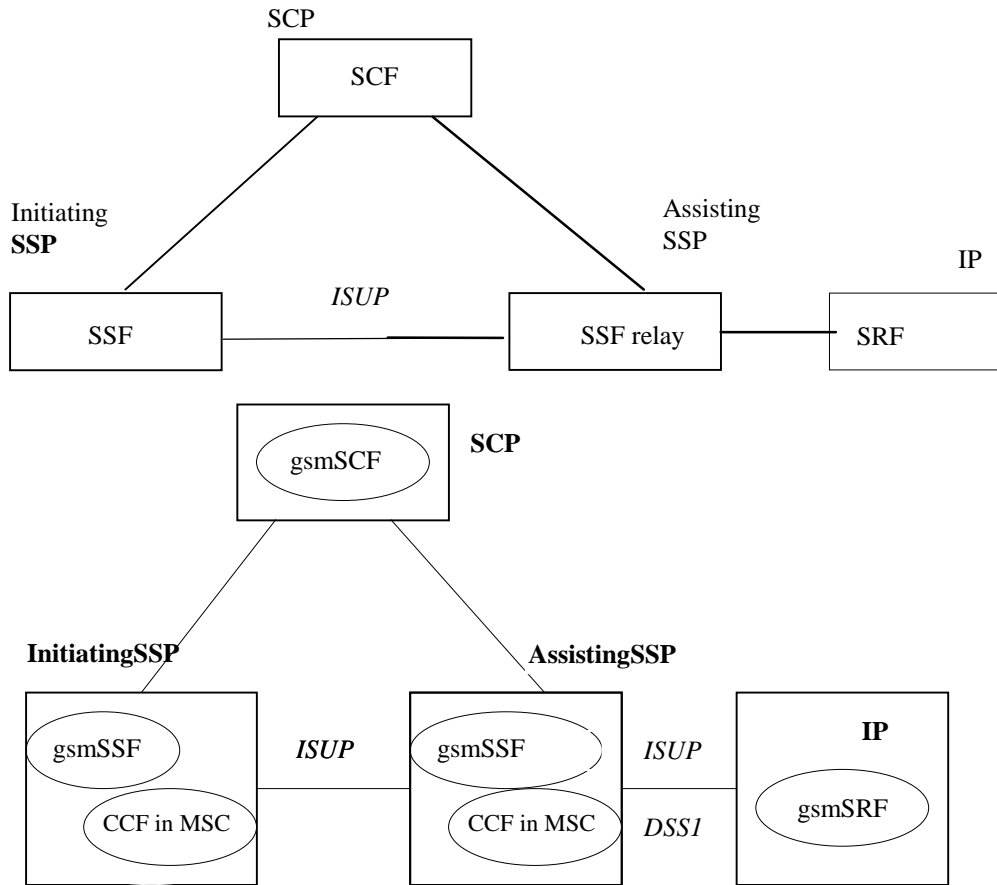
Scenario 1, Direct Path to IP (Ref. CS1 cases b) & d)).

Figure 4-1: Scenarios



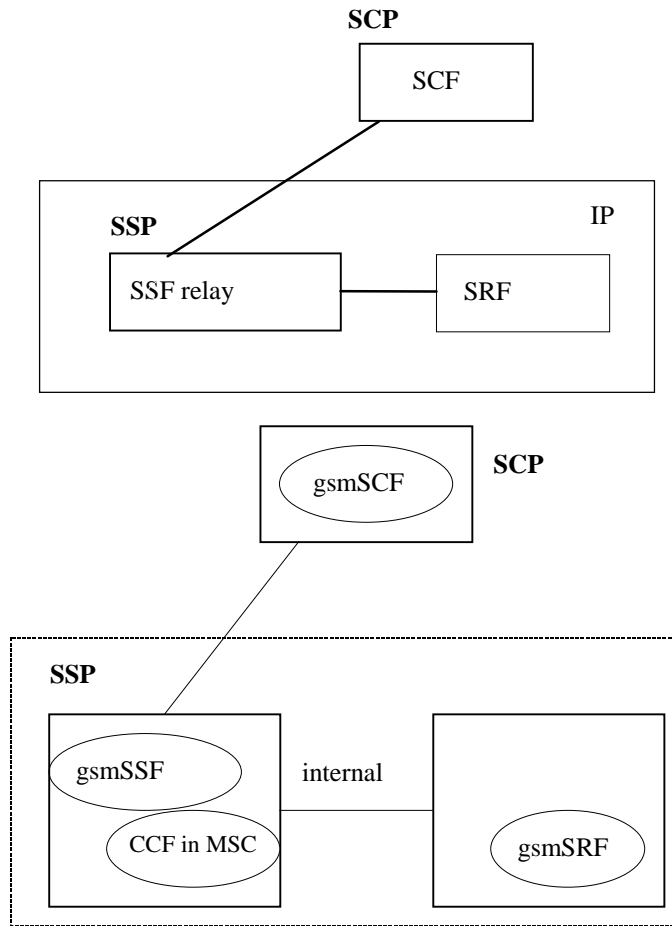
Scenario 2a. Connection to IP via an assisting gsmSSF with relay function; IP co-located with assisting gsmSSF (Ref. CS1 case c).

Figure 4-1 (continued): Scenarios



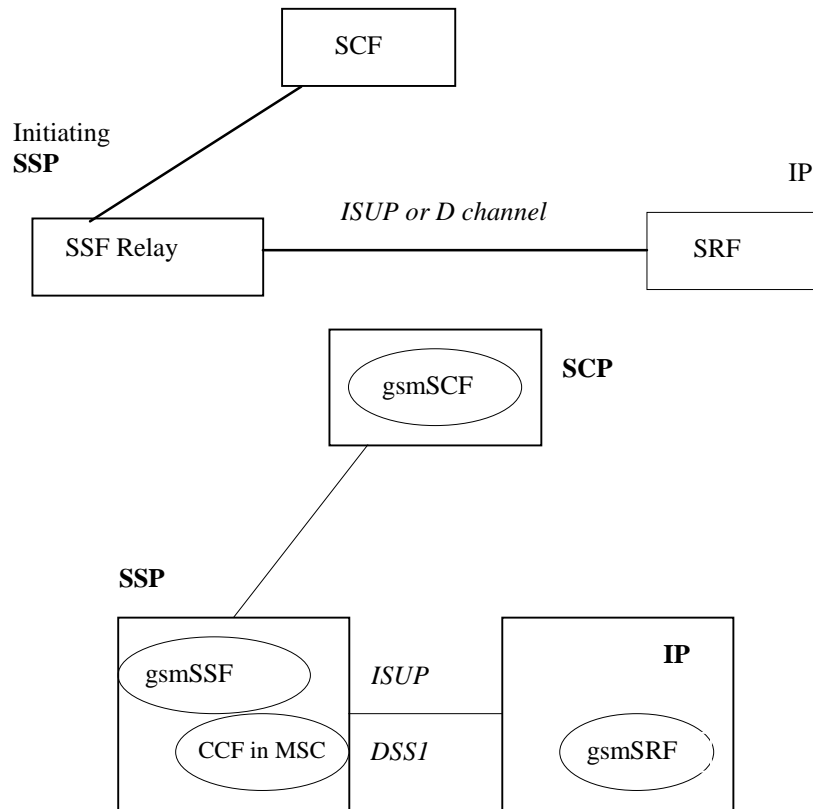
Scenario 2b: Connection to IP via an assisting gsmSSF with relay function; IP not co-located with assisting gsmSSF (Ref CS1 case c)).

Figure 4-1 (continued): Scenarios



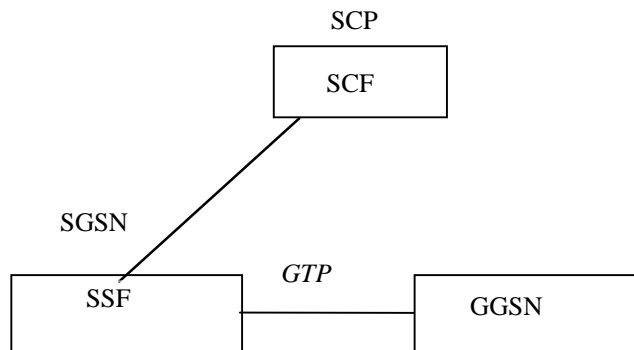
Scenario 3, Connection to IP with relay function; IP co-located with gsmSSF (Ref CS1 case a)).

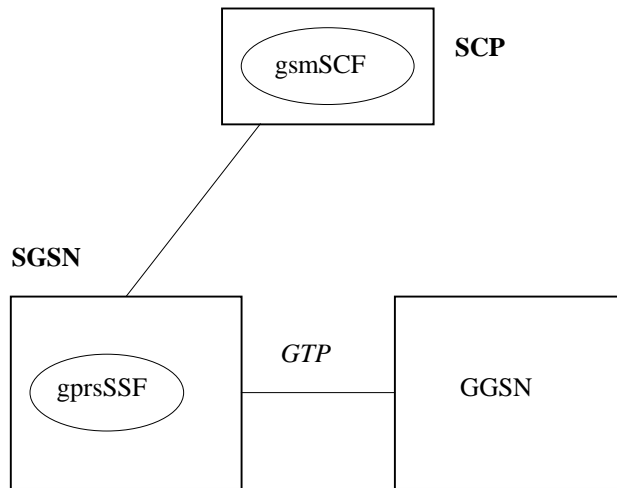
Figure 4-1 (continued): Scenarios



Scenario 4, Connection to IP with relay function; IP not co-located with gsmSSF (Ref CS1 case a)).

Figure 4-1 (continued): Scenarios





Scenario 5, GPRS interworking. No connection to IP.

Figure 4-1 (concluded): Scenarios

The following table summarises the scenarios and corresponding interface connections that shall be supported by the CAP protocol. The following terms used in the table are defined as follows:

- Basic:** Fully defined in CAP and may be used between any two network operators supporting CAP.
- Bilateral:** Additional clarifications of CAP capabilities between network operators and/or equipment vendors are necessary in order for CAP to be used between any two network operators supporting CAP.
- Direct:** This refers to the case where CAP Operations are exchanged between the gsmSRF and the gsmSCF via a transaction-level relationship established directly between the gsmSRF and the gsmSCF.
- Relay:** This refers to the case where CAP Operations are exchanged between the gsmSRF and the gsmSCF via two transaction-layer relationships. These relationships are:
- gsmSCF to/from gsmSSF;
 - gsmSSF to/from gsmSRF.

The gsmSSF sends operations it receives from the gsmSCF to the gsmSRF, and operations it receives from the gsmSRF to the gsmSCF. This is done without unpacking (and thus processing) of the relayed operations.

The gsmSSF function referred to in the table is always located in an MSC or GMSC.

The gprsSSF function referred to in the table is always located in an SGSN.

Table 4-1

Scenario	Interface Support				
	GsmSSF to/from gsmSCF	gsmSSF to/from gsmSRF	gsmSSF to/from assisting gsmSSF	gsmSRF to/from gsmSCF	assisting gsmSSF to/from gsmSCF
<p>Scenario 1</p> <p>gsmSRF in IP connected to gsmSSF in MSC or GMSC via ISUP and accessed by gsmSCF through direct Signalling System No.7 Connection</p>	See Note 1	See Note 2	-	See Notes 3 and 6. For gsmSRF in VPLMN see Note 4; For gsmSRF in HPLMN see note 5	-
<p>Scenario 2a</p> <p>assisting gsmSSF in MSC or GMSC connected to gsmSSF in MSC or GMSC via ISUP. Assisting gsmSSF is accessed by gsmSCF through direct Signalling System No.7 Connection.</p> <p>gsmSRF is co-located with assisting gsmSSF and accessed (by gsmSCF) by relay via assisting gsmSSF over an internal nodal interface</p>	See Note 1 For gsmSRF in VPLMN see Notes 4 and 6; For gsmSRF in HPLMN see note 5 and 6	-	See Note 2	-	See Note 3
<p>Scenario 2b</p> <p>assisting gsmSSF in MSC or GMSC connected to gsmSSF in MSC or GMSC via ISUP. Assisting gsmSSF is accessed by gsmSCF through direct Signalling System No.7 Connection</p> <p>gsmSRF is in IP connected to assisting gsmSSF and accessed (by gsmSCF) by relay through ISUP or DSS1 via assisting gsmSSF</p>	See Note 1 See Notes 4 and 6	See Notes 4 and 6	See Note 2	-	See Note 3
<p>Scenario 3</p> <p>gsmSRF is co-located with a gsmSSF in an MSC or GMSC and accessed by relay via gsmSSF over an internal nodal interface</p>	For gsmSRF in VPLMN see Notes 4; For gsmSRF in HPLMN see notes 5 and 6	-	-	-	-
<p>Scenario 4</p> <p>gsmSRF in IP connected to gsmSSF and accessed by gsmSCF by relay through ISUP or DSS1 via gsmSSF</p>	See Notes 4 and 6	See Notes 4 and 6	-	-	-

NOTE 1: Basic for establishment of interface when CorrelationID and SCFiD are transferred in the AssistingSSPIPRoutingAddress. Bilateral when CorrelationID and SCFiD are transferred by other means than in the AssistingSSPIPRoutingAddress.

NOTE 2: Basic for establishment of interface when CorrelationID and SCFiD are transferred in the Called Party Number. Bilateral when CorrelationID and SCFiD are transferred by other means than in the Called Party Number.

NOTE 3: Basic when the full Called Party Number received in VPLMN or HPLMN is transferred on its own in the AssistRequestInstructions operation CorrelationID parameter to a gsmSCF in HPLMN.

Bilateral when CorrelationID is extracted from Called Party Number in HPLMN or VPLMN and transferred on its own in AssistRequestInstructions CorrelationID field to a gsmSCF in HPLMN.

NOTE 4: Bilateral for the playing of announcements via elementaryMessageIDs and variableMessages, playing of tones and the collection of DTMF digits.

NOTE 5: Basic for the playing of announcements via elementaryMessageIDs and variableMessages, playing of tones and the collection of DTMF digits.

NOTE 6: Bilateral for the playing of announcements via text to speech translation, translation of DTMF digits via speech to caller and the translation of voice to digits.

4.1.3 CAP protocol architecture

Many of the terms used in the present subclause are based on the OSI application layer structure as defined in ISO 9545 (1989) [81].

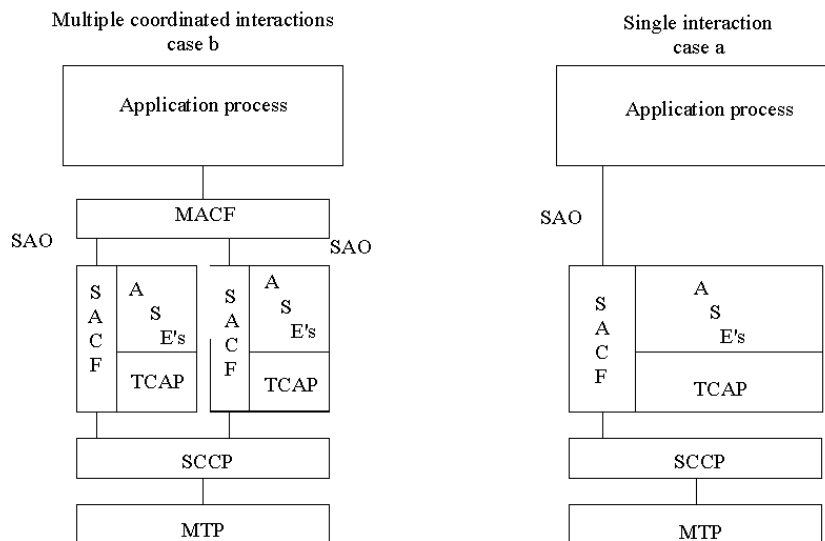
The CAP protocol architecture is illustrated in figure 4-2.

A Physical Entity (PE) has either single interactions (case a) or multiple co-ordinated interactions (case b) with another PE.

In case (a), SACF provides a co-ordination function in using Application Service Elements (ASEs), which includes the ordering of operations supported by ASE(s), in the order of received primitives. The Single Association Object (SAO) represents the SACF plus a set of ASEs to be used over a single interaction between a pair of PEs.

In case (b), MACF provides a co-ordinating function among several SAO's, each of which interacts with an SAO in a remote PE.

Each ASE supports one or more operations. Description of each operation is tied with the action of corresponding Functional Entity (FE) modelling. For FE modelling, refer to ITU-T Recommendation Q.1224 [50] and clause 11 of the present document. Each operation is specified using the OPERATION macro described in figure 4-3.



NOTE: CAP is the collection of all specifications in ASEs.

Figure 4-2: CAP protocol architecture

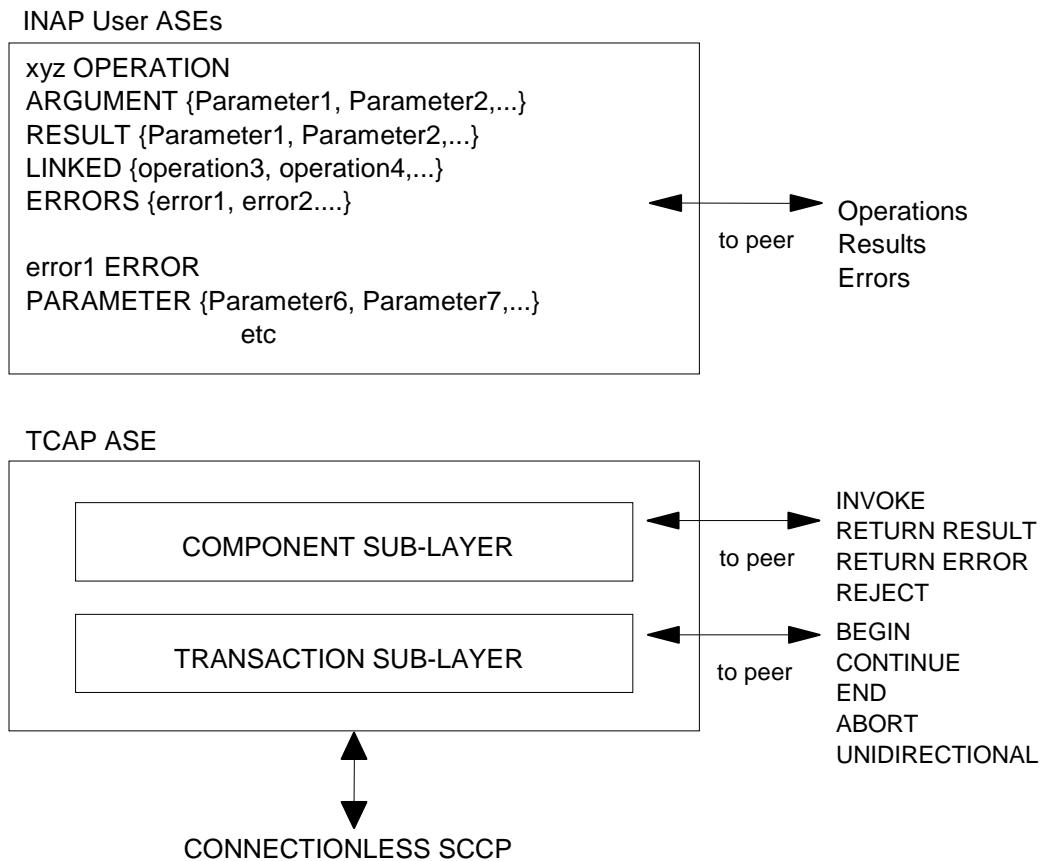


Figure 4-3: Operation description

4.1.4 Compatibility mechanisms used for CAP

4.1.4.1 Introduction

The present subclause specifies the compatibility mechanisms that shall be used for CAP.

Two major categories of compatibility are handled by these mechanisms:

- compatibility with the ITU-T Recommendation Q.1228 [51] version of CS2 INAP and the specification EN 301 140-1 version of CS2 INAP [26];
- compatibility with future versions of CAP.

The second category has three subcategories of compatibility dealt with within the present subclause:

- **Minor changes to CAP in future versions of CAP:**
A minor change can be defined as a change of a functionality which is not essential for the requested IN service. Where it is a modification of an existing function, it is acceptable that the addressed function is executed in either the older or the modified variant. If the change is purely additional, then it is acceptable that it is not executed at all and that the peer Application Entity (AE) need not know about the effects of the change. For minor changes, a new Application Context is not required.
- **Major changes to CAP in future versions of CAP:**
A major change can be defined as a change of a functionality which is essential for the requested IN service. Where it is a modification of an existing function, both application entities shall have a shared knowledge about the addressed functional variant. If the change is purely additional and one of the application entities does not support the additional functionality, then the requested IN service will not be provided. For major changes, a new Application Context is required.

- Network-specific changes to CAP:
These additions may be of either the major or minor type for a service. No new Application Context is expected to be defined for this type of change. At the time of definition of these network-specific changes to CAP, the additions would not be expected to be included in identical form in future versions of CAP.

4.1.4.2 Definition of CAP compatibility mechanisms

4.1.4.2.1 Compatibility mechanism for interworking of CAP with ETSI CS2 Core INAP and ITU-T Q.1228 INAP

On receipt of an operation according to ITU-T Recommendation Q.1228 [51] or an operation according to ETSI EN 301 140-1 [26], which is not part of the CAP or is part of the CAP but which contains parameters which are not part of the CAP:

- the gsmSSF, gsmSCF, assisting gsmSSF and gsmSRF shall apply the normal error handling for unknown operations or parameters, i.e. the normal error handling procedures as specified in clause 10 shall be followed.

Tagging of CAP additions to ITU-T Recommendation Q.1228 [51] and ETSI EN 301 140-1 [26] are specified from 50 to 59.

4.1.4.2.2 Procedures for major additions to CAP

In order to support the introduction of major functional changes, the protocol allows a synchronisation between the two applications with regard to which functionality is to be performed. This synchronisation takes place before the new function is invoked in either application entity, in order to avoid complicated fall-back procedures.

4.1.4.2.3 Procedures for minor additions to CAP

The extension mechanism marker shall be used for future standardised minor additions to CAP. This mechanism implements extensions by including an "extensions marker" in the type definition. The extensions are expressed by optional fields that are placed after the marker. When an entity receives unrecognised parameters that occur after the extension marker, they are ignored. Refer to ITU-T Recommendation X.680 [53] for further details on the extension mechanism.

4.1.4.2.4 Procedures for inclusion of network specific additions to CAP

This mechanism is based on the ability to explicitly declare fields of any type via the Macro facility in ASN.1 at the outermost level of a type definition. It works by defining an "ExtensionField" that is placed within the type definition. This extension field is defined as a set of extensions, where an extension can contain any type. Each extension is associated with an identification that unambiguously identifies the extension. Refer to ITU-T Recommendation Q.1400 [52] for a definition of this mechanism.

4.1.5 Definition and usage of LegID

4.1.5.1 Definition of LegID

In CAP V4, two types of LegID may be exchanged between the gsmSCF and the gsmSSF. These are:

- Sending Side LegID; and
- Receiving Side LegID.

Sending Side LegID is always used in operations sent from the gsmSCF to the gsmSSF, and Receiving Side LegID is always used in operations sent from the gsmSSF to the gsmSCF.

4.1.5.2 Allocation of LegID

For all operations containing a LegID:

- LegID = 1 shall always refer to the Calling Party, more specifically that party in the call present when InitialDP is sent to the gsmSCF;
- LegID = 2 shall always refer to a Called Party, more specifically a party in the call created as a result of the Connect, Continue or ContinueWithArgument operation.
- LegID > 2 shall always refer to a Called Party, more specifically a party in the call created as a result of the InitiateCallAttempt operation.

4.2 SACF/MACF rules

4.2.1 Reflection of TC AC

TC AC negotiation rules require that the proposed AC, if acceptable, is reflected in the first backwards message.

NOTE: If the gsmSSF, gprsSSF or smsSSF provides an AC which is not acceptable to the gsmSCF, then an alternate AC shall not be returned. If the AC presented to the gsmSCF is not acceptable, then this is most probably due to an error in subscriber data provisioning or an error at the gsmSSF, gprsSSF or smsSSF.

4.2.2 Sequential/parallel execution of operations

In some cases it may be necessary to distinguish whether operations should be performed sequentially or in parallel (synchronised). Operations which may be synchronised are:

- charging operations; may be synchronised with any other operation(s).

The method of indicating to the receiving entity that operations should be synchronised is to transmit these operations in a single TC message. If one of the operations identified above shall not be executed until the execution of another operation has progressed to a certain extent or has finished, then the sending PE shall control this by sending the operations in two separate TC messages.

This method does not imply that all operations sent in a single TC message shall be executed simultaneously, but that where it could make sense to do so (in the situations identified above) the operations should be synchronised.

In the case of inconsistency between the above-mentioned generic rules and the FE-specific rules, as specified in clause 9, the FE-specific rules take precedence over the generic rules.

5 Common CAP Types

5.1 Data types

```

CAP-datatypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-datatypes(52) version4(3)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS

    Duration,
    Integer4,
    Interval,
    LegID,
    ServiceKey
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
modules(0) cs1-datatypes(2) version1(0)}

    BothwayThroughConnectionInd,
    CallSegmentID {},
    CriticalityType,
    MiscCallInfo
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
cs2(20) modules(0) in-cs2-datatypes(0) version1(0)}

    IMSI,
    ISDN-AddressString,
    Ext-BasicServiceCode,
    NAEA-CIC
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}

    Ext-QoS-Subscribed,
    GeographicalInformation,
    GSN-Address,
    LocationInformation,
    LSAIdentity,
    QoS-Subscribed,
    SubscriberState
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}

    CallReferenceNumber,
    SuppressionOfAnnouncement
FROM MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-CH-DataTypes(13) version6(6)}

    tc-Messages,
    classes
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

    TCInvokeIdSet
FROM TCAPMessages tc-Messages

    EXTENSION,
    PARAMETERS-BOUND,
    SupportedExtensions
FROM CAP-classes classes

    ExtensionContainer
FROM MAP-ExtensionDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-ExtensionDataTypes(21) version6(6)}

;

AccessPointName {PARAMETERS-BOUND: bound} ::= OCTET STRING (SIZE(
    bound.&minAccessPointNameLength .. bound.&maxAccessPointNameLength))
-- Indicates the AccessPointName, refer to 3GPP TS 24.008 [9] for the encoding.
-- It shall be coded as in the value part defined in 3GPP TS 24.008,
-- i.e. the 3GPP TS 24.008 IEI and 3GPP TS 24.008 octet length indicator
-- shall not be included.

AChBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE
    (bound.&minAChBillingChargingLength .. bound.&maxAChBillingChargingLength))
    (CONSTRAINED BY {-- shall be the result of the BER-encoded value of the type --
    CAMEL-AChBillingChargingCharacteristics {bound}})
-- The AChBillingChargingCharacteristics parameter specifies the charging related information
-- to be provided by the gsmSSF and the conditions on which this information has to be reported
-- back to the gsmSCF with the ApplyChargingReport operation. The value of the

```

```
-- AChBillingChargingCharacteristics of type OCTET STRING carries a value of the ASN.1 data type:
-- CAMEL-AChBillingChargingCharacteristics. The normal encoding rules are used to encode this
-- value.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.
```

```
AChChargingAddress {PARAMETERS-BOUND : bound} ::= CHOICE {
  legID          [2] LegID,
  srfConnection [50] CallSegmentID {bound}
}
```

```
AdditionalCallingPartyNumber {PARAMETERS-BOUND : bound} ::= Digits {bound}
-- Indicates the Additional Calling Party Number.
```

```
AlertingPattern ::= OCTET STRING (SIZE(3))
-- Indicates a specific pattern that is used to alert a subscriber
-- (e.g. distinctive ringing, tones, etc.).
-- The encoding of the last octet of this parameter is as defined in 3GPP TS 29.002 [11].
-- Only the trailing OCTET is used, the remaining OCTETS shall be sent as NULL (zero)
-- The receiving side shall ignore the leading two OCTETS.
```

```
AOCBeforeAnswer ::= SEQUENCE {
  aOCInitial          [0] CAI-GSM0224,
  aOCSubsequent      [1] AOCSubsequent          OPTIONAL
}
```

```
AOCGPRS ::= SEQUENCE {
  aOCInitial          [0] CAI-GSM0224,
  aOCSubsequent      [1] AOCSubsequent          OPTIONAL
}
```

```
AOCSubsequent ::= SEQUENCE {
  cAI-GSM0224        [0] CAI-GSM0224 ,
  tariffSwitchInterval [1] INTEGER (1..86400)    OPTIONAL
}
-- tariffSwitchInterval is measured in 1 second units
```

```
AppendFreeFormatData ::= ENUMERATED {
  overwrite (0),
  append (1)
}
```

```
ApplicationTimer ::= INTEGER (0..2047)
-- Used by the gsmSCF to set a timer in the gsmSSF. The timer is in seconds.
```

```
AssistingSSPIPRoutingAddress {PARAMETERS-BOUND : bound} ::= Digits {bound}
-- Indicates the destination address of the gsmSRF for the assist procedure.
```

```
AudibleIndicator ::= CHOICE {
  Tone          BOOLEAN,
  burstList    [1] Burstlist
}
```

```
BackwardServiceInteractionInd ::= SEQUENCE {
  conferenceTreatmentIndicator [1] OCTET STRING (SIZE(1))    OPTIONAL,
  -- acceptConferenceRequest 'xxxx xx01'B
  -- rejectConferenceRequest 'xxxx xx10'B
  -- if absent from Connect or ContinueWithArgument,
  -- then CAMEL service does not affect conference treatment
  callCompletionTreatmentIndicator [2] OCTET STRING (SIZE(1))    OPTIONAL,
  -- acceptCallCompletionServiceRequest 'xxxx xx01'B,
  -- rejectCallCompletionServiceRequest 'xxxx xx10'B
  -- if absent from Connect or ContinueWithArgument,
  -- then CAMEL service does not affect call completion treatment
  ...
}
```

```
BasicGapCriteria {PARAMETERS-BOUND : bound} ::= CHOICE {
  calledAddressValue [0] Digits {bound},
  gapOnService       [2] GapOnService,
  calledAddressAndService [29] SEQUENCE {
    calledAddressValue [0] Digits {bound},
    serviceKey         [1] ServiceKey,
    ...
  },
  callingAddressAndService [30] SEQUENCE {
    callingAddressValue [0] Digits {bound},
    serviceKey         [1] ServiceKey,
    ...
  }
}
```

```
-- Both calledAddressValue and callingAddressValue can be
-- incomplete numbers, in the sense that a limited amount of digits can be given.
-- For the handling of numbers starting with the same digit string refer to the detailed
-- procedure of the CallGap operation
```

```
BCSMEvent ::= SEQUENCE {
  eventTypeBCSM [0] EventTypeBCSM,
  monitorMode  [1] MonitorMode,
```



```

    legID                [2] LegID                OPTIONAL,
    dpSpecificCriteria    [30] DpSpecificCriteria   OPTIONAL,
    automaticRearm        [50] NULL                OPTIONAL
  }
-- Indicates the BCSM Event information for monitoring.

BCSM-Failure ::= SEQUENCE {
  LegID                [0] LegID                OPTIONAL,
  cause                [2] Cause {bound}         OPTIONAL,
  ...
}

BearerCapability {PARAMETERS-BOUND : bound} ::= CHOICE {
  bearerCap            [0] OCTET STRING (SIZE(2..bound.&maxBearerCapabilityLength))
}
-- Indicates the type of bearer capability connection to the user. For bearerCap, the ISUP User
-- Service Information, ETSI EN 300 356-1 [23]
-- encoding shall be used.

Burst ::= SEQUENCE {
  numberOfBursts        [0] INTEGER (1..3) DEFAULT 1,
  burstInterval          [1] INTEGER (1..20)                OPTIONAL,
  numberOfTonesInBurst  [2] INTEGER (1..3) DEFAULT 3,
  toneDuration          [3] INTEGER (1..20) DEFAULT 2,
  toneInterval          [4] INTEGER (1..20) DEFAULT 2
}
-- burstInterval, toneDurartion and toneInterval are measured in 100 millisecond units

BurstList ::= SEQUENCE {
  warningPeriod          [0] INTEGER (1..1200) DEFAULT 30,
  bursts                 [1] Burst
}
-- warningPeriod is measured in 1 second units.

CAI-GSM0224 ::= SEQUENCE {
  e1                    [0] INTEGER (0..8191)                OPTIONAL,
  e2                    [1] INTEGER (0..8191)                OPTIONAL,
  e3                    [2] INTEGER (0..8191)                OPTIONAL,
  e4                    [3] INTEGER (0..8191)                OPTIONAL,
  e5                    [4] INTEGER (0..8191)                OPTIONAL,
  e6                    [5] INTEGER (0..8191)                OPTIONAL,
  e7                    [6] INTEGER (0..8191)                OPTIONAL
}
-- Indicates Charge Advice Information to the Mobile Station. For information regarding
-- parameter usage, refer to 3GPP TS 22.024 [2].

CalledPartyBCDNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
  bound.&minCalledPartyBCDNumberLength .. bound.&maxCalledPartyBCDNumberLength))
-- Indicates the Called Party Number, including service selection information.
-- Refer to 3GPP TS 24.008 [9] for encoding.
-- This data type carries only the "type of number", "numbering plan
-- identification" and "number digit" fields defined in 3GPP TS 24.008 [9];
-- it does not carry the "called party BCD number IEI" or "length of called
-- party BCD number contents".

CalledPartyNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
  bound.&minCalledPartyNumberLength .. bound.&maxCalledPartyNumberLength))
-- Indicates the Called Party Number. Refer to ETS EN 300 356-1 [23] for encoding.

-- A CalledPartyNumber may contain national-specific values of the Nature Of Address
-- indicator. The filling-in of the national-specific Nature Of Address indicator
-- values shall be done in accordance with the national ISUP of the gsmSSF country, e.g.
-- ANSI T1.113-1995 [92].
-- In terms of ETS EN 300 356-1 [23], the Destination Address Field is not present if the
-- destination address length is set to zero. This is the case e.g. when the ANSI
-- ISUP Nature Of Address indicator indicates no number present, operator requested
-- (1110100) or no number present, cut-through call to carrier (1110101).
-- See also see 3GPP TS 23.078 [7].

CallingPartyNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
  bound.&minCallingPartyNumberLength .. bound.&maxCallingPartyNumberLength))
-- Indicates the Calling Party Number. Refer to ETSI EN 300 356-1 [23] for encoding.

CallResult {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
  bound.&minCallResultLength .. bound.&maxCallResultLength))
  (CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
  CAMEL-CallResult {bound}})
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.

-- This parameter provides the gsmSCF with the charging related information previously requested
-- using the ApplyCharging operation. This shall include the partyToCharge parameter as
-- received in the related ApplyCharging operation to correlate the result to the request

CallSegmentFailure {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  callSegmentID        [0] CallSegmentID {bound}           OPTIONAL,
  cause                [2] Cause {bound}                   OPTIONAL,
  ...
}

```

```

    }
CallSegmentToCancel ::= SEQUENCE {
    invokeID                [0] Invoke ID                OPTIONAL,
    callSegmentID           [1] CallSegmentID {bound}         OPTIONAL,
    ...
}

CAMEL-AChBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= CHOICE {
    timeDurationCharging    [0] SEQUENCE {
        maxCallPeriodDuration [0] INTEGER (1..864000),
        releaseIfdurationExceeded [1] BOOLEAN DEFAULT FALSE,
        tariffsSwitchInterval [2] INTEGER (1..86400)           OPTIONAL,
        audibleIndicator      [3] AudibleIndicator DEFAULT {tone FALSE},
        extensions            [4] Extensions {bound}           OPTIONAL,
        ...
    }
}
-- tariffsSwitchInterval is measured in 1 second units.
-- maxCallPeriodDuration is measured in 100 millisecond units

CAMEL-CallResult {PARAMETERS-BOUND : bound} ::= CHOICE {
    TimeDurationChargingResult [0] SEQUENCE {
        partyToCharge          [0] ReceivingSideID,
        timeInformation        [1] TimeInformation,
        callActive             [2] BOOLEAN DEFAULT TRUE,
        callReleasedAtTcpExpiry [3] NULL                    OPTIONAL,
        extensions             [4] Extensions {bound}         OPTIONAL,
        aChChargingAddress     [5] AChChargingAddress {bound}
                                DEFAULT legID:receivingSideID:leg1,
        ...
    }
}

CAMEL-FCIBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= CHOICE{
    fCIBCCAMELsequence1     [0] SEQUENCE {
        freeFormatData       [0] OCTET STRING (SIZE(
            bound.&minFCIBillingChargingDataLength .. bound.&maxFCIBillingChargingDataLength)),
        partyToCharge        [1] SendingSideID DEFAULT sendingSideID : leg1,
        appendFreeFormatData [2] AppendFreeFormatData DEFAULT overwrite
    }
}

CAMEL-FCIGPRSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= SEQUENCE{
    fCIBCCAMELsequence1     [0] SEQUENCE {
        freeFormatData       [0] OCTET STRING (SIZE(
            bound.&minFCIBillingChargingDataLength .. bound.&maxFCIBillingChargingDataLength)),
        pDPID                [1] PDPID OPTIONAL,
        appendFreeFormatData [2] AppendFreeFormatData DEFAULT overwrite,
        ...
    }
}

CAMEL-FCISMSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= CHOICE{
    fCIBCCAMELsequence1     [0] SEQUENCE {
        freeFormatData       [0] OCTET STRING (SIZE(
            bound.&minFCIBillingChargingDataLength .. bound.&maxFCIBillingChargingDataLength)),
        appendFreeFormatData [1] AppendFreeFormatData DEFAULT overwrite
    }
}

CAMEL-SCIBillingChargingCharacteristics ::= CHOICE {
    aOCBeforeAnswer         [0] AOCBeforeAnswer,
    aOCAfterAnswer         [1] AOCSubsequent
}

CAMEL-SCIGPRSBillingChargingCharacteristics ::= SEQUENCE {
    aOCGPRS                 [0] AOCGPRS,
    pDPID                   [1] PDPID                OPTIONAL,
    ...
}

Carrier {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minCarrierLength .. bound.&maxCarrierLength))
-- This parameter is used for North America (na) only.
-- It contains the carrier selection field (first octet) followed by Carrier ID
-- information (North America (na)).

-- The Carrier selection is one octet and is encoded as:
-- 00000000 No indication
-- 00000001 Selected carrier identification code (CIC) pre subscribed and not
--          input by calling party
-- 00000010 Selected carrier identification code (CIC) pre subscribed and input by
--          calling party
-- 00000011 Selected carrier identification code (CIC) pre subscribed, no
--          indication of whether input by calling party (undetermined)
-- 00000100 Selected carrier identification code (CIC) not pre subscribed and
--          input by calling party

```

```

-- 00000101
-- to Spare
-- 11111110
-- 11111111 Reserved

-- Refer to ANSI T1.113-1995 [92] for encoding of na carrier ID information (3 octets).

Cause {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minCauseLength .. bound.&maxCauseLength))
-- Indicates the cause for interface related information.
-- Refer to ETSI EN 300 356-1 [23] Cause parameter for encoding.
-- For the use of cause and location values refer to ITU-T Recommendation Q.850 [47]
-- Shall always include the cause value and shall also include the diagnostics field,
-- if available.

CGEncountered ::= ENUMERATED {
    noCGencountered (0),
    manualCGencountered (1),
    scpOverload (2)
}
-- Indicates the type of automatic call gapping encountered, if any.

ChargeNumber {PARAMETERS-BOUND : bound} ::= LocationNumber {bound}
-- Information sent in either direction indicating the chargeable number for the call and
-- consisting of the odd/even indicator, nature of address indicator, numbering plan indicator,
-- and address signals.

-- Uses the LocationNumber format which is based on the Location Number format as defined
-- in ITU-T Recommendation Q.763 [45].
-- For example, the ChargeNumber may be a third party number to which a call is billed for
-- the 3rd party billing service. In this case, the calling party may request operator assistance
-- to charge the call to, for example, their home number.

-- For NA, this parameter uniquely identifies the chargeable number for a call sent into a North
-- American long distance carrier. It transports the ChargeNumber Parameter Field as defined in
-- ANSI T1.113-1995 [92]. This provides
-- - 1 octet for the nature of address indicator field, plus
-- - 1 octet for a numbering plan field, plus
-- - up to 5 octets for the address signal (up to 10 digits)

-- The Charge Number in ANSI T1.113-1995 [92] normally contains a 10 digit national number within
-- the North American Numbering Plan (NANP); longer (e.g. international) charge numbers are not
-- supported in ANSI T1.113-1995 [92].

ChargeUnitsComponent ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE {
    monitorComponent [0] MonitorComponent,
    reportingConditionUnits [1] ReportingConditionUnits
    }

ChargeUnitsTotal ::= SEQUENCE {
    reportingConditionUnits [0] ReportingConditionUnits
    }

ChargingCharacteristics ::= CHOICE {
    MaxTransferredVolume [0] INTEGER (1..4294967295),
    maxElapsedTime [1] INTEGER (1..86400)
    }
-- maxTransferredVolume is measured in number of bytes
-- maxElapsedTime is measured in seconds

ChargingResult ::= CHOICE {
    transferredVolume [0] TransferredVolume,
    elapsedTime [1] ElapsedTime
    }

ChargingRollOver ::= CHOICE {
    transferredVolumeRollOver [0] TransferredVolumeRollOver,
    elapsedTimeRollOver [1] ElapsedTimeRollOver
    }
-- transferredVolumeRollOver shall be reported if ApplyChargingReportGPRS reports volume and
-- a roll-over has occurred in one or more volume counters. Otherwise, it shall be absent.
-- elapsedTimeRollOver shall be reported if ApplyChargingReportGPRS reports duration and
-- a roll-over has occurred in one or more duration counters. Otherwise, it shall be absent.

CollectedDigits ::= SEQUENCE {
    MinimumNbOfDigits [0] INTEGER (1..30) DEFAULT 1,
    maximumNbOfDigits [1] INTEGER (1..30),
    endOfReplyDigit [2] OCTET STRING (SIZE (1..2)) OPTIONAL,
    cancelDigit [3] OCTET STRING (SIZE (1..2)) OPTIONAL,
    startDigit [4] OCTET STRING (SIZE (1..2)) OPTIONAL,
    firstDigitTimeout [5] INTEGER (1..127) OPTIONAL,
    interDigitTimeout [6] INTEGER (1..127) OPTIONAL,
    errorTreatment [7] ErrorTreatment DEFAULT stdErrorAndInfo,
    interruptableAnnInd [8] BOOLEAN DEFAULT TRUE,
    voiceInformation [9] BOOLEAN DEFAULT FALSE,
    voiceBack [10] BOOLEAN DEFAULT FALSE
    }

```

```

}
-- The use of voiceBack and the support of voice recognition via voiceInformation
-- is network operator specific.
-- The endOfReplyDigit, cancelDigit, and startDigit parameters have been
-- designated as OCTET STRING, and are to be encoded as BCD, one digit per octet
-- only, contained in the four least significant bits of each OCTET. The following encoding shall
-- be applied for the non-decimal characters:
-- 1011 (*), 1100 (#).
-- The usage is service dependent.
-- firstDigitTimeOut and interDigitTimeOut are measured in seconds.

CollectedInfo ::= CHOICE {
    collectedDigits [0] CollectedDigits
}

ConnectedNumberTreatmentInd ::= ENUMERATED {
    noINImpact (0),
    presentationRestricted (1),
    presentCalledINNumber (2),
    presentCallINNumberRestricted (3)
}
-- This parameter is used to suppress or to display the connected number.

ControlType ::= ENUMERATED {
    sCPOverloaded (0),
    manuallyInitiated (1)
}

CompoundCriteria {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    BasicGapCriteria [0] BasicGapCriteria {bound},
    scfID [1] ScfID {bound}
}
OPTIONAL

CorrelationID {PARAMETERS-BOUND : bound} ::= Digits {bound}
-- used by gsmSCF for correlation with a previous operation.

DateAndTime ::= OCTET STRING (SIZE(7))
-- DateAndTime is BCD encoded. The year digit indicating millenium occupies bits
-- 0-3 of the first octet, and the year digit indicating century occupies bits
-- 4-7 of the first octet.
-- The year digit indicating decade occupies bits 0-3 of the second octet,
-- whilst the digit indicating the year within the decade occupies bits 4-7 of
-- the second octet.
-- The most significant month digit occupies bits 0-3 of the third octet,
-- and the least significant month digit occupies bits 4-7 of the third octet.
-- The most significant day digit occupies bits 0-3 of the fourth octet,
-- and the least significant day digit occupies bits 4-7 of the fourth octet.
-- The most significant hours digit occupies bits 0-3 of the fifth octet,
-- and the least significant digit occupies bits 4-7 of the fifth octet.
-- The most significant minutes digit occupies bits 0-3 of the sixth octet,
-- and the least significant digit occupies bits 4-7 of the sixth octet.
-- The most significant seconds digit occupies bits 0-3 of the seventh octet,
-- and the least significant digit occupies bits 4-7 of the seventh octet.
-- For the encoding of digits in an octet, refer to the timeAndtimezone parameter.

DestinationRoutingAddress {PARAMETERS-BOUND : bound} ::= SEQUENCE SIZE(1) OF
    CalledPartyNumber {bound}
-- Indicates the Called Party Number.

Digits {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minDigitsLength .. bound.&maxDigitsLength))
-- Indicates the address signalling digits.
-- Refer to ETSI EN 300 356-1 [23] Generic Number & Generic Digits parameters for encoding.
-- The coding of the subfields "NumberQualifier" in Generic Number and "TypeOfDigits" in
-- Generic Digits are irrelevant to the CAP;
-- the ASN.1 tags are sufficient to identify the parameter.
-- The ISUP format does not allow to exclude these subfields,
-- therefore the value is network operator specific.
--
-- The following parameters shall use Generic Number:
-- - AdditionalCallingPartyNumber for InitialDP
-- - AssistingSSPIPRoutingAddress for EstablishTemporaryConnection
-- - CorrelationID for AssistRequestInstructions
-- - CalledAddressValue for all occurrences, CallingAddressValue for all occurrences.
--
-- The following parameters shall use Generic Digits:
-- - CorrelationID in EstablishTemporaryConnection
-- - number in VariablePart
-- - digitsResponse in ReceivedInformationArg
-- - midCallEvents in oMidCallSpecificInfo and tMidCallSpecificInfo
--
-- In the digitsResponse and midCallevents, the digits may also include the '*', '#',
-- a, b, c and d digits by using the IA5 character encoding scheme. If the BCD even or
-- BCD odd encoding scheme is used, then the following encoding shall be applied for the
-- non-decimal characters: 1011 (*), 1100 (#).

-- Note that when CorrelationID is transported in Generic Digits, then the digits shall
-- always be BCD encoded.

```

```

DpSpecificCriteria ::= CHOICE {
    applicationTimer          [1] ApplicationTimer,
    midCallControlInfo       [2] MidCallControlInfo
}
-- The gsmSCF may set a timer in the gsmSSF for the No_Answer event.
-- If the user does not answer the call within the allotted time,
-- then the gsmSSF reports the event to the gsmSCF.
-- The gsmSCF may define a criterion for the detection of DTMF digits during a call.

ElapsedTime ::= CHOICE {
    TimeGPRSIfNoTariffSwitch [0] INTEGER (0..86400),
    timeGPRSIfTariffSwitch   [1] SEQUENCE {
        timeGPRSSinceLastTariffSwitch [0] INTEGER (0..86400),
        timeGPRSTariffSwitchInterval [1] INTEGER (0..86400)
    }
}
-- timeGPRSIfNoTariffSwitch is measured in seconds
-- timeGPRSSinceLastTariffSwitch and timeGPRSTariffSwitchInterval are measured in seconds

ElapsedTimeRollOver ::= CHOICE {
    rO-TimeGPRSIfNoTariffSwitch [0] INTEGER (0..255),
    rO-TimeGPRSIfTariffSwitch   [1] SEQUENCE {
        rO-TimeGPRSSinceLastTariffSwitch [0] INTEGER (0..255)
        rO-TimeGPRSTariffSwitchInterval [1] INTEGER (0..255)
    }
}
-- rO-TimeGPRSIfNoTariffSwitch, rO-TimeGPRSSinceLastTariffSwitch and
-- rO-TimeGPRSTariffSwitchInterval
-- present counters indicating the number of parameter range rollovers.

EndUserAddress {PARAMETERS-BOUND: bound} ::= SEQUENCE {
    pdPTypeOrganization [0] OCTET STRING (SIZE(1)),
    pdPTypeNumber       [1] OCTET STRING (SIZE(1)),
    pdPAddress          [2] OCTET STRING (SIZE(
        bound.&minPDPAddressLength .. bound.&maxPDPAddressLength))
}
-- Indicates the EndUserAddress, refer to 3GPP TS 29.060 [12] for the encoding.
-- The pdPTypeOrganization shall use the least significant 4 bits of the octet encoded.
-- The sender of this parameter shall set the most significant 4 bits of the octet to 1.
-- The receiver of this parameter shall ignore the most significant 4 bits of this octet.

ErrorTreatment ::= ENUMERATED {
    stdErrorAndInfo (0),
    help (1),
    repeatPrompt (2)
}
-- stdErrorAndInfo means returning the "ImproperCallerResponse" error in the event of an error
-- condition during collection of user info.

EvaluesAndTsw ::= SEQUENCE {
    values [0] CAI-GSM0224,
    tariffSwitchInterval [1] TariffSwitchInterval
}

EvaluesComponent ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE {
    monitorComponent [0] MonitorComponent,
    reportingConditionValue [1] ReportingConditionValue
}

EvaluesTotal ::= SEQUENCE {
    ReportingConditionValue [0] ReportingConditionValue
}

EventSpecificChargingInformation {PARAMETERS-BOUND : bound} ::= CHOICE {
    infoChargeUnitsTotal [0] InfoChargeUnitsTotal,
    infoChargeUnitsComponents [1] InfoChargeUnitsComponents {bound},
    infoValueTotal [2] InfoValueTotal,
    infoValueComponents [3] InfoValueComponents {bound}
}

EventSpecificInformationBCSM {PARAMETERS-BOUND : bound} ::= CHOICE {
    routeSelectFailureSpecificInfo [2] SEQUENCE {
        failureCause [0] Cause {bound}
    }
    ;
    oCalledPartyBusySpecificInfo [3] SEQUENCE {
        busyCause [0] Cause {bound}
    }
    ;
    oNoAnswerSpecificInfo [4] SEQUENCE {
        -- no specific info defined --
    }
    ;
    oAnswerSpecificInfo [5] SEQUENCE {
        destinationAddress [50] CalledPartyNumber {bound}
    }
}

```

```

        or-Call                [51] NULL                OPTIONAL,
        forwardedCall          [52] NULL                OPTIONAL,
        ...
    };
oMidCallSpecificInfo          [6] SEQUENCE {
    midCallEvents              [1] CHOICE {
        dtmfdigitsCompleted    [3] Digits,
        dtmfdigitsTimeOut      [4] Digits
    }
    ...
}
oDisconnectSpecificInfo      [7] SEQUENCE {
    releaseCause               [0] Cause {bound}
    ...
}
tBusySpecificInfo            [8] SEQUENCE {
    busyCause                  [0] Cause {bound}
    callForwarded              [50] NULL
    routeNotPermitted          [51] NULL
    forwardingDestinationNumber [52] CalledPartyNumber {bound}
    ...
}
tNoAnswerSpecificInfo        [9] SEQUENCE {
    callForwarded              [50] NULL
    forwardingDestinationNumber [52] CalledPartyNumber {bound}
    ...
}
tAnswerSpecificInfo          [10] SEQUENCE {
    destinationAddress          [50] CalledPartyNumber {bound}
    or-Call                    [51] NULL
    forwardedCall               [52] NULL
    ...
}
tMidCallSpecificInfo         [11] SEQUENCE {
    midCallEvents              [1] CHOICE {
        dtmfdigitsCompleted    [3] Digits,
        dtmfdigitsTimeOut      [4] Digits
    }
    ...
}
tDisconnectSpecificInfo      [12] SEQUENCE {
    releaseCause               [0] Cause {bound}
    ...
}
oTermSeizedSpecificInfo      [13] SEQUENCE {
    locationInformation          [50] LocationInformation
    ...
}
callAcceptedSpecificInfo     [20] SEQUENCE {
    locationInformation          [50] LocationInformation
    ...
}
oAbandonSpecificInfo         [21] SEQUENCE {
    routeNotPermitted          [50] NULL
    ...
}
oChangeOfPositionSpecificInfo [50] SEQUENCE {
    locationInformation          [50] LocationInformation
    ...
}
tChangeOfPositionSpecificInfo [51] SEQUENCE {
    locationInformation          [50] LocationInformation
    ...
}
}
-- Indicates the call related information specific to the event.
EventSpecificInformationSMS ::= CHOICE {
    o-smsFailureSpecificInfo   [0] SEQUENCE {
        failureCause           [0] MO-SMSCause
        ...
    };
    o-smsSubmissionSpecificInfo [1] SEQUENCE {
        -- no specific info defined-
        ...
    };
    t-smsFailureSpecificInfo   [2] SEQUENCE {
        failureCause           [0] MT-SMSCause
        ...
    };
    t-smsDeliverySpecificInfo  [3] SEQUENCE {
        -- no specific info defined-
        ...
    };
}

```

```

EventTypeBCSM ::= ENUMERATED {
    collectedInfo                (2),
    analyzedInformation          (3),
    routeSelectFailure           (4),
    oCalledPartyBusy             (5),
    oNoAnswer                    (6),
    oAnswer                      (7),
    oMidCall                     (8),
    oDisconnect                  (9),
    oAbandon                     (10),
    termAttemptAuthorized        (12),
    tBusy                        (13),
    tNoAnswer                    (14),
    tAnswer                      (15),
    tMidCall                     (16),
    tDisconnect                  (17),
    tAbandon                     (18),
    oTermSeized                  (19),
    callAccepted                 (27),
    oChangeOfPosition            (50),
    tChangeOfPosition            (51)
}
-- Indicates the BCSM detection point event.
-- Values collectedInfo, analyzedInformation and termAttemptAuthorized may be used
-- for TDPs only.

EventTypeChargingPLMN {PARAMETERS-BOUND : bound} ::= CHOICE {
    ChargeUnitsTotal             [0] ChargeUnitsTotal,
    ChargeUnitsComponent         [1] ChargeUnitsComponent {bound},
    EvaluesTotal                 [2] EvaluesTotal,
    EvaluesComponent             [3] EvaluesComponent {bound}
}
-- This parameter indicates the charging event type.

EventTypesSMS ::= ENUMERATED {
    sms-CollectedInfo           (1),
    o-smsFailure                 (2),
    o-smsSubmission              (3),
    sms-DeliveryRequested        (11),
    t-smsFailure                 (12),
    t-smsDelivery                (13)
}
-- Values sms-CollectedInfo and sms-DeliveryRequested may be used for TDPs only.

Extensions {PARAMETERS-BOUND : bound} ::= SEQUENCE SIZE (1..bound.&numOfExtensions) OF
ExtensionField
ExtensionField ::= SEQUENCE {
    Type                        EXTENSION.&id ({SupportedExtensions {bound}}),
                                -- shall identify the value of an EXTENSION type
    criticality                 CriticalityType DEFAULT ignore,
    value                       [1] EXTENSION.&ExtensionType ({SupportedExtensions}{@type}),
    ...
}
-- This parameter indicates an extension of an argument data type.
-- Its content is network operator specific

FCIBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minFCIBillingChargingLength .. bound.&maxFCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type --
    CAMEL-FCIBillingChargingCharacteristics {bound}})
-- This parameter indicates the billing and/or charging characteristics.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.

FCIGPRSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minFCIBillingChargingLength .. bound.&maxFCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
    CAMEL-FCIGPRSBillingChargingCharacteristics {bound}})
-- This parameter indicates the GPRS billing and/or charging characteristics.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.

FCISMSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minFCIBillingChargingLength .. bound.&maxFCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
    CAMEL-FCISMSBillingChargingCharacteristics {bound}})
-- This parameter indicates the SMS billing and/or charging characteristics.
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.

ForwardServiceInteractionInd ::= SEQUENCE {
    conferenceTreatmentIndicator [1] OCTET STRING (SIZE(1))          OPTIONAL,
    -- acceptConferenceRequest    'xxxx xx01'B
    -- rejectConferenceRequest    'xxxx xx10'B
    -- if absent from Connect or ContinueWithArgument,
    -- then CAMEL service does not affect conference treatment
    callDiversionTreatmentIndicator [2] OCTET STRING (SIZE(1))      OPTIONAL,
    -- callDiversionAllowed       'xxxx xx01'B
    -- callDiversionNotAllowed    'xxxx xx10'B
}

```

```

-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect call diversion treatment
callingPartyRestrictionIndicator [4] OCTET STRING (SIZE(1)) OPTIONAL,
-- noINImpact 'xxxx xx01'B
-- presentationRestricted 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect calling party restriction treatment
...
}

GapCriteria {PARAMETERS-BOUND : bound} ::= CHOICE {
  basicGapCriteria BasicGapCriteria {bound},
  compoundGapCriteria CompoundCriteria {bound}
}

GapIndicators ::= SEQUENCE {
  duration [0] Duration,
  gapInterval [1] Interval,
  ...
}
-- Indicates the call gapping characteristics.
-- No call gapping when gapInterval equals 0.

GapOnService ::= SEQUENCE {
  serviceKey [0] ServiceKey,
  ...
}

GapTreatment {PARAMETERS-BOUND : bound} ::= CHOICE {
  informationToSend [0] InformationToSend {bound},
  releaseCause [1] Cause {bound}
}
-- The default value for Cause is the same as in ISUP.

GenericNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
  bound.&minGenericNumberLength .. bound.&maxGenericNumberLength))
-- Indicates a generic number. Refer to ETSI EN 300 356-1 [23] Generic number for encoding.

GenericNumbers {PARAMETERS-BOUND : bound} ::= SET SIZE(1..bound.&numOfGenericNumbers) OF
GenericNumber {bound}

GPRS-QoS ::= CHOICE {
  short-QoS-format [0] QoS-Subscribed,
  long-QoS-format [1] Ext-QoS-Subscribed
}
-- Short-QoS-format shall be sent for QoS in pre GSM release 99 format.
-- Long-QoS-format shall be sent for QoS in GSM release 99 (and beyond) format.
-- Which of the two QoS formats shall be sent is determined by which QoS
-- format is available in the SGSN at the time of sending.
-- Refer to 3GPP TS 29.002 [11] for encoding details of QoS-Subscribed and
-- Ext-QoS-Subscribed.

GPRSCause {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE
  (bound.&minGPRSCauseLength .. bound.&maxGPRSCauseLength))
-- Shall only include the cause value.

-- 00000000 Unspecified
-- All other values shall be interpreted as "Unspecified".
--
-- This parameter indicates the cause for CAP interface related information.
-- The GPRSCause mapping to/from GTP cause values specified in the 3GPP TS 29.060 [12] and
-- to/from 3GPP TS 24.008 [9] GMM cause and SM cause values are outside scope of this document.

GPRSEvent ::= SEQUENCE {
  gPRSEventType [0] GPRSEventType,
  monitorMode [1] MonitorMode
}
-- Indicates the GPRS event information for monitoring.

GPRSEventSpecificInformation {PARAMETERS-BOUND : bound} ::= CHOICE {
  attachChangeOfPositionSpecificInformation [0] SEQUENCE {
    locationInformationGPRS [0] LocationInformationGPRS OPTIONAL,
    ...
  },
  pdp-ContextchangeOfPositionSpecificInformation [1] SEQUENCE {
    accessPointName [0] AccessPointName {bound} OPTIONAL,
    chargingID [1] GPRSChargingID OPTIONAL,
    locationInformationGPRS [2] LocationInformationGPRS OPTIONAL,
    endUserAddress [3] EndUserAddress {bound} OPTIONAL,
    qualityOfService [4] QualityOfService OPTIONAL,
    timeAndTimeZone [5] TimeAndTimezone {bound} OPTIONAL,
    ...
    gGSNAddress [6] GSN-Address OPTIONAL
  },
}

```



```

detachSpecificInformation          [2] SEQUENCE {
    initiatingEntity                [0] InitiatingEntity          OPTIONAL,
    ...,
    routeingAreaUpdate             [1] NULL                      OPTIONAL
},

disconnectSpecificInformation      [3] SEQUENCE {
    initiatingEntity                [0] InitiatingEntity          OPTIONAL,
    ...,
    routeingAreaUpdate             [1] NULL                      OPTIONAL
},

pDPContextEstablishmentSpecificInformation
    [4] SEQUENCE {
        accessPointName            [0] AccessPointName {bound}    OPTIONAL,
        endUserAddress             [1] EndUserAddress {bound}    OPTIONAL,
        qualityOfService           [2] QualityOfService        OPTIONAL,
        locationInformationGPRS     [3] LocationInformationGPRS  OPTIONAL,
        timeAndTimeZone            [4] TimeAndTimezone {bound}    OPTIONAL,
        pdpInitiationType         [5] PDPInitiationType        OPTIONAL,
        ...,
        secondaryPDP-context       [6] NULL                      OPTIONAL
    },

pDPContextEstablishmentAcknowledgementSpecificInformation
    [5] SEQUENCE {
        accessPointName            [0] AccessPointName {bound}    OPTIONAL,
        chargingID                 [1] GPRSChargingID          OPTIONAL,
        endUserAddress             [2] EndUserAddress {bound}    OPTIONAL,
        qualityOfService           [3] QualityOfService        OPTIONAL,
        locationInformationGPRS     [4] LocationInformationGPRS  OPTIONAL,
        timeAndTimeZone            [5] TimeAndTimezone {bound}    OPTIONAL,
        ...,
        gGSNAddress                [6] GSN-Address              OPTIONAL
    }
}

GPRSEventType ::= ENUMERATED {
    attach                          (1),
    attachChangeOfPosition          (2),
    detached                        (3),
    pdp-ContextEstablishment        (11),
    pdp-ContextEstablishmentAcknowledgement (12),
    disonnect                      (13),
    pdp-ContextChangeOfPosition     (14)
}

InbandInfo {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    messageID                       [0] MessageID {bound},
    numberOfRepetitions              [1] INTEGER (1..127)          OPTIONAL,
    duration                         [2] INTEGER (0..32767)        OPTIONAL,
    interval                         [3] INTEGER (0..32767)        OPTIONAL,
    ...
}
-- Interval is the time in seconds between each repeated announcement. Duration is the total
-- amount of time in seconds, including repetitions and intervals.
-- The end of announcement is either the end of duration or numberOfRepetitions,
-- whatever comes first.
-- duration with value 0 indicates infinite duration

InfoChargeUnitsComponents ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE {
    reportingConditionUnits         [0] ReportingConditionUnits,
    units                           [1] Units,
    monitorComponent                [2] MonitorComponent
}

InfoChargeUnitsTotal ::= SEQUENCE {
    ReportingConditionUnits         [0] ReportingConditionUnits,
    Units                           [1] Units
}

InfoValueComponents ::= SEQUENCE SIZE
    (bound.&minComponentLength .. bound.&maxComponentLength) OF SEQUENCE {
    reportingConditionEvaluate      [0] ReportingConditionEvaluate,
    valuesAndTsw                   [1] ValuesAndTsw,
    monitorComponent               [2] MonitorComponent
}

InfoValueTotal ::= SEQUENCE {
    ReportingConditionEvaluate      [0] ReportingConditionEvaluate,
    ValuesAndTsw                   [1] ValuesAndTsw
}

InformationToSend {PARAMETERS-BOUND : bound} ::= CHOICE {

```

```

    inbandInfo          [0] InbandInfo {bound},
    tone                [1] Tone
  }

InitiatingEntity ::= ENUMERATED {
    mobileStation      (0),
    sgsn                (1),
    hlr                 (2),
    ggsn                (3)
}

InvokeID ::= TCInvokeIdSet

IPRoutingAddress {PARAMETERS-BOUND : bound} ::= CalledPartyNumber {bound}
-- Indicates the routeing address for the IP.

IPSSPCapabilities {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minIPSSPCapabilitiesLength .. bound.&maxIPSSPCapabilitiesLength))
-- Indicates the gsmSRF resources available. The parameter has two parts, a standard and a
-- bilateral part. The standard part indicates capabilities defined as optional in CAP V.2
-- that shall be recognised (but not necessarily supported) by a CAP V.2 gsmSCF. The bilateral
-- part contains further information that is not specified in this standard, but which is set
-- according to bilateral agreements between network operators and/or equipment vendors.
-- The last octet of the standard part is indicated by bit 7 being set to 0, otherwise Bit 7 of
-- a standard part octet is set to 1 indicating that the standard part continues in the following
-- octet. Coding is as follows:

-- Octet 1          Standard Part for CAP V.3
-- Bit Value       Meaning
-- 0 0             IPRoutingAddress not supported
-- 0 1             IPRoutingAddress supported
-- 1 0             VoiceBack not supported
-- 1 1             VoiceBack supported
-- 2 0             VoiceInformation not supported, via speech recognition
-- 2 1             VoiceInformation supported, via speech recognition
-- 3 0             VoiceInformation not supported, via voice recognition
-- 3 1             VoiceInformation supported, via voice recognition
-- 4 0             Generation of voice announcements from Text not supported
-- 4 1             Generation of voice announcements from Text supported
-- 5 -             Reserved
-- 6 -             Reserved
-- 7 0             End of standard part
-- 7 1             This value is reserved in CAP V.3
--
-- Octets 2 to 4   Bilateral Part: Network operator/equipment vendor specific

LegOrCallSegment ::= CHOICE {
    callSegmentID      [0] CallSegmentID {bound},
    legID              [1] LegID
}

LegType ::= OCTET STRING (SIZE(1))
leg1 LegType ::= '01'H
leg2 LegType ::= '02'H

LocationNumber {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE (
    bound.&minLocationNumberLength .. bound.&maxLocationNumberLength))
-- Indicates the Location Number for the calling party.
-- Refer to ETSI EN 300 356-1 [23] for encoding.

MessageID {PARAMETERS-BOUND : bound} ::= CHOICE {
    elementaryMessageID [0] Integer4,
    text                [1] SEQUENCE {
        messageContent [0] IA5String (SIZE(
            bound.&minMessageContentLength .. bound.&maxMessageContentLength)),
        attributes      [1] OCTET STRING (SIZE(
            bound.&minAttributesLength .. bound.&maxAttributesLength)) OPTIONAL
    },
    elementaryMessageIDs [29] SEQUENCE SIZE (1.. bound.&numOfMessageIDs) OF Integer4,
    variableMessage      [30] SEQUENCE {
        elementaryMessageID [0] Integer4,
        variableParts        [1] SEQUENCE SIZE (1..5) OF VariablePart {bound}
    }
}
-- Use of the text parameter is network operator/equipment vendor specific.

MidCallControlInfo ::= SEQUENCE {
    MinimumNumberOfDigits [0] INTEGER (1..30) DEFAULT 1,
    MaximumNumberOfDigits [1] INTEGER (1..30) DEFAULT 30,
    endOfReplyDigit       [2] OCTET STRING (SIZE (1..2)) OPTIONAL,
    cancelDigit           [3] OCTET STRING (SIZE (1..2)) OPTIONAL,
    startDigit            [4] OCTET STRING (SIZE (1..2)) OPTIONAL,
    interDigitTimeout     [6] INTEGER (1..127) DEFAULT 10
}
--
-- - minimumNumberOfDigits specifies the minumum number of digits that shall be collected
-- - maximumNumberOfDigits specifies the maximum number of digits that shall be collected

```

```

-- - endOfReplyDigit          specifies the digit string that denotes the end of the digits
--                             to be collected.
-- - cancelDigit              specifies the digit string that indicates that the input shall
--                             be erased and digit collection shall start afresh.
-- - startDigit               specifies the digit string that denotes the start of the digits
--                             to be collected.
-- - interDigitTimeout        specifies the maximum duration in seconds between successive
--                             digits.
--
-- endOfReplyDigit, cancelDigit and startDigit shall contain digits in the range 0..9, '*' and '#'
-- only. The collected digits string, reported to the gsmSCF, shall include the endOfReplyDigit and
-- the startDigit, if present.
--
-- endOfReplyDigit, cancelDigit and startDigit shall be encoded as BCD digits. Each octet shall
-- contain one BCD digit, in the 4 least significant bits of each octet.
-- The following encoding shall be used for the over-decadic digits: 1011 (*), 1100 (#).

MonitorComponent ::= ENUMERATED {
    ChargesTransitNetwork          (0),
    ChargesSpecificPLMN           (1)
}

MonitorMode ::= ENUMERATED {
    interrupted                    (0),
    notifyAndContinue              (1),
    transparent                     (2)
}
-- Indicates the event is relayed and/or processed by the SSP.
-- Transparent means that the gsmSSF or gprsSSF does not notify the gsmSCF of the event.
-- For the use of this parameter refer to the procedure descriptions in clause 11.
-- For the RequestNotificationCharging operation, "interrupted" shall not be used in MonitorMode.

MO-SMSCause ::= ENUMERATED {
    systemFailure                  (0),
    unexpectedDataValue            (1),
    facilityNotSupported           (2),
    sM-DeliveryFailure             (3),
    releaseFromRadioInterface      (4)
}
-- MO SMS error values which are reported to gsmSCF.
-- Most of these values are received from the SMSC as a response to
-- MO-ForwardSM operation.

MT-SMSCause ::= OCTET STRING (SIZE (1))
-- This variable is sent to the gsmSCF for a Short Message delivery failure
-- notification.
-- If the delivery failure is due to RP-ERROR RPDU received from the MS,
-- then MT-SMSCause shall be set to the RP-Cause component in the RP-ERROR RPDU.
-- Refer to 3G TS 24.011 [10] for the encoding of RP-Cause values.
-- Otherwise, if the delivery failure is due to internal failure in the MSC or SGSN
-- or time-out from the MS, then MT-SMSCause shall be set to "Protocol error,
-- unspecified", as defined in 3G TS 24.011 [10].

NAOliInfo ::= OCTET STRING (SIZE (1))
-- NA Oli information takes the same value as defined in ANSI T1.113-1995 [92]
-- e.g.   '3D'H - Decimal value 61 - Cellular Service (Type 1)
--        '3E'H - Decimal value 62 - Cellular Service (Type 2)
--        '3F'H - Decimal value 63 - Cellular Service (roaming)

OCSIApplicable ::= NULL
-- Indicates that the Originating CAMEL Subscription Information, if present, shall be
-- applied on the outgoing call leg created with a Connect operation. For the use of this
-- parameter see 3GPP TS 23.078 [7].

OriginalCalledPartyID {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minOriginalCalledPartyIDLength .. bound.&maxOriginalCalledPartyIDLength))
-- Indicates the original called number. Refer to ETSI EN 300 356-1 [23] Original Called Number
-- for encoding.

PDPID ::= OCTET STRING (SIZE (1))
-- PDP Identifier is a counter used to identify a specific PDP Context within a control
-- relationship between gprsSSF and gsmSCF.

PDPInitiationType ::= ENUMERATED {
    mSInitiated                    (0),
    networkInitiated               (1)
}

QualityOfService ::= SEQUENCE {
    requested-QoS                   [0] GPRS-QoS                OPTIONAL,
    subscribed-QoS                  [1] GPRS-QoS                OPTIONAL,
    negotiated-QoS                  [2] GPRS-QoS                OPTIONAL,
    ...

```

```

}
-- The procedure descriptions in clause 11 indicate which one(s) of the
-- QoS variables shall be transported.

ReceivingSideID ::= CHOICE {
    ReceivingSideID                [1] LegType
}
-- used to identify LegID in operations sent from gsmSSF to gsmSCF

RedirectingPartyID {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE (
    bound.&minRedirectingPartyIDLength .. bound.&maxRedirectingPartyIDLength))
-- Indicates redirecting number.
-- Refer to ETSI EN 300 356-1 [23] Redirecting number for encoding.

ReportingConditionUnits ::= CHOICE {
    EndOfConnection                [0] IMPLICIT NULL,
    ThresholdCountervalue          [1] IMPLICIT INTEGER {SIZE(4)}
}

ReportingConditionEvaluate ::= CHOICE {
    OccurrenceOfEvent              [0] IMPLICIT NULL
}

RequestedInformationList {PARAMETERS-BOUND : bound} ::= SEQUENCE SIZE (1.. numOfInfoItems) OF
RequestedInformation {bound}

RequestedInformationTypeList ::= SEQUENCE SIZE (1.. numOfInfoItems) OF RequestedInformationType

RequestedInformation {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    RequestedInformationType        [0] RequestedInformationType,
    requestedInformationValue       [1] RequestedInformationValue {bound},
    ...
}

RequestedInformationType ::= ENUMERATED {
    CallAttemptElapsedTime         (0),
    callStopTime                   (1),
    callConnectedElapsedTime       (2),
    releaseCause                   (30)
}

RequestedInformationValue {PARAMETERS-BOUND : bound} ::= CHOICE {
    callAttemptElapsedTimeValue    [0] INTEGER (0..255),
    callStopTimeValue              [1] DateAndTime,
    callConnectedElapsedTimeValue  [2] Integer4,
    releaseCauseValue              [30] Cause {bound}
}
-- The callAttemptElapsedTimeValue is specified in seconds. The unit for the
-- callConnectedElapsedTimeValue is 100 milliseconds

RPCause ::= OCTET STRING (SIZE (1))
-- RP cause according to 3GPP TS 24.011 [10] or 3G TS 29.002 [11].
-- GsmSCF shall send this cause in the ReleaseSMS operation.
-- For a MO-SMS service, the MSC or SGSN shall send the RP Cause to the originating MS.
-- It shall be used to overwrite the RP-Cause element in the RP-ERROR RPDU.
-- For a MT-SMS service, the MSC or SGSN shall send the RP Cause to the sending SMS-GMSC.
-- It shall be used to overwrite the RP-Cause element in the RP-ERROR RPDU.

ScfID {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minScfIDLength .. bound.&maxScfIDLength))
-- defined by network operator.
-- Indicates the gsmSCF identity.

SCIBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE (
    bound.&minSCIBillingChargingLength .. bound.&maxSCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type --
    CAMEL-SCIBillingChargingCharacteristics})
-- Indicates AOC information to be sent to a Mobile Station
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.

SCIGPRSBillingChargingCharacteristics {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE (
    bound.&minSCIBillingChargingLength .. bound.&maxSCIBillingChargingLength))
(CONSTRAINED BY {-- shall be the result of the BER-encoded value of type -
    CAMEL-SCIGPRSBillingChargingCharacteristics})
-- Indicates AOC information to be sent to a Mobile Station
-- The violation of the UserDefinedConstraint shall be handled as an ASN.1 syntax error.

SendingSideID ::= CHOICE {sendingSideID [0] LegType}
-- used to identify LegID in operations sent from gsmSCF to gsmSSF

ServiceInteractionIndicatorsTwo ::= SEQUENCE {
    ForwardServiceInteractionInd    [0] ForwardServiceInteractionInd                OPTIONAL,
    -- applicable to operations InitialDP, Connect and ContinueWithArgument.
    backwardServiceInteractionInd    [1] BackwardServiceInteractionInd              OPTIONAL,
    -- applicable to operations Connect and ContinueWithArgument.
    bothwayThroughConnectionInd     [2] BothwayThroughConnectionInd                OPTIONAL,
    -- applicable to ConnectToResource and EstablishTemporaryConnection

```

```

connectedNumberTreatmentInd      [4] ConnectedNumberTreatmentInd      OPTIONAL,
-- applicable to Connect and ContinueWithArgument
nonCUGCall                        [13] NULL                          OPTIONAL,
-- applicable to Connect and ContinueWithArgument
-- indicates that no parameters for CUG shall be used for the call (i.e. the call shall
-- be a non-CUG call).
-- If not present, it indicates one of three things:
-- a) continue with modified CUG information (when one or more of either CUG Interlock Code
-- and Outgoing Access Indicator are present), or
-- b) continue with original CUG information (when neither CUG Interlock Code or Outgoing
-- Access Indicator are present), i.e. no IN impact.
-- c) continue with the original non-CUG call.
holdTreatmentIndicator            [50] OCTET STRING (SIZE(1))      OPTIONAL,
-- applicable to InitialDP, Connect and ContinueWithArgument
-- acceptHoldRequest 'xxxx xx01'B
-- rejectHoldRequest 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect call hold treatment
cwTreatmentIndicator             [51] OCTET STRING (SIZE(1))      OPTIONAL,
-- applicable to InitialDP, Connect and ContinueWithArgument
-- acceptCw 'xxxx xx01'B
-- rejectCw 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect call waiting treatment
ectTreatmentIndicator            [52] OCTET STRING (SIZE(1))      OPTIONAL,
-- applicable to InitialDP, Connect and ContinueWithArgument
-- acceptEctRequest 'xxxx xx01'B
-- rejectEctRequest 'xxxx xx10'B
-- if absent from Connect or ContinueWithArgument,
-- then CAMEL service does not affect explicit call transfer treatment
...
}

```

```
SGSNCapabilities ::= OCTET STRING (SIZE (1))
```

```

-- Indicates the SGSN capabilities. The coding of the parameter is as follows:
-- Bit Value      Meaning
-- 0 0            AoC not supported by SGSN
-- 1 1            AoC supported by SGSN
-- 2 -            This bit is reserved in CAP V.3
-- 3 -            This bit is reserved in CAP V.3
-- 4 -            This bit is reserved in CAP V.3
-- 5 -            This bit is reserved in CAP V.3
-- 6 -            This bit is reserved in CAP V.3
-- 7 -            This bit is reserved in CAP V.3

```

```

SMSEvent ::= SEQUENCE {
    eventTypeSMS      [0] EventTypeSMS,
    monitorMode      [1] MonitorMode
}

```

```

TariffSwitchInterval ::= INTEGER (1 .. 86400)
-- TariffSwitchInterval is measured in 1 second units

```

```

TimeAndTimezone {PARAMETERS-BOUND : bound} ::= OCTET STRING (SIZE(
    bound.&minTimeAndTimezoneLength .. bound.&maxTimeAndTimezoneLength))
-- Indicates the time and timezone, relative to GMT. This parameter BCD encoded.
-- The year digit indicating millenium occupies bits 0-3 of the first octet, and the year
-- digit indicating century occupies bits 4-7 of the first octet.
-- The year digit indicating decade occupies bits 0-3 of the second octet, whilst the digit
-- indicating the year within the decade occupies bits 4-7 of the second octet.
-- The most significant month digit occupies bits 0-3 of the third octet, and the least
-- significant month digit occupies bits 4-7 of the third octet.
-- The most significant day digit occupies bits 0-3 of the fourth octet, and the least
-- significant day digit occupies bits 4-7 of the fourth octet.
-- The most significant hours digit occupies bits 0-3 of the fifth octet, and the least
-- significant hours digit occupies bits 4-7 of the fifth octet.
-- The most significant minutes digit occupies bits 0-3 of the sixth octet, and the least
-- significant minutes digit occupies bits 4-7 of the sixth octet.
-- The most significant seconds digit occupies bits 0-3 of the seventh octet, and the least
-- significant seconds digit occupies bits 4-7 of the seventh octet.

```

```

-- The timezone information occupies the eighth octet. For the encoding of Timezone refer to
-- 3GPP TS 23.040 [6].

```

```

-- The BCD digits are packed and encoded as follows:

```

```

-- Bit 7 6 5 4 | 3 2 1 0
-- 2nd digit | 1st digit      Octet 1
-- 3rd digit | 4th digit      Octet 2
-- ..
-- nth digit | n-lth digit    Octet m
--
-- 0000      digit 0
-- 0001      digit 1
-- 0010      digit 2
-- 0011      digit 3
-- 0100      digit 4

```

```

--      0101      digit 5
--      0110      digit 6
--      0111      digit 7
--      1000      digit 8
--      1001      digit 9
--      1010      spare
--      1011      spare
--      1100      spare
--      1101      spare
--      1110      spare
--      1101      spare
--
-- where the leftmost bit of the digit is either bit 7 or bit 3 of the octet.

TimeIfNoTariffSwitch ::= INTEGER(0..864000)
-- TimeIfNoTariffSwitch is measured in 100 millisecond intervals

TimeIfTariffSwitch ::= SEQUENCE {
    timeSinceTariffSwitch      [0] INTEGER(0..864000),
    tariffSwitchInterval      [1] INTEGER(1..864000)           OPTIONAL
}
-- timeSinceTariffSwitch and tariffSwitchInterval are measured in 100 millisecond intervals

TimeInformation ::= CHOICE {
    timeIfNoTariffSwitch      [0] TimeIfNoTariffSwitch,
    timeIfTariffSwitch        [1] TimeIfTariffSwitch
}
-- Indicates call duration information

TimerID ::= ENUMERATED {
    tssf                      (0)
}
-- Indicates the timer to be reset.

TimerValue ::= Integer4
-- Indicates the timer value (in seconds).

Tone ::= SEQUENCE {
    toneID                    [0] Integer4,
    duration                   [1] Integer4                   OPTIONAL,
    ...
}
-- The duration specifies the length of the tone in seconds, value 0 indicates infinite duration.

TPDataCodingScheme ::= OCTET STRING (SIZE (1))
-- TP Data Coding Scheme according to 3GPP TS 23.040 [6]

TPProtocolIdentifier ::= OCTET STRING (SIZE (1))
-- indicates the protocol used above the SM-Transfer Layer as specified in 3GPP TS 23.040 [6].

TPShortMessageSpecificInfo ::= OCTET STRING (SIZE (1))
-- contains the 1st octet of the applicable TPDU or the SMS-COMMAND TPDU as specified in
-- 3GPP TS 23.040 [6].

TPValidityPeriod ::= OCTET STRING (SIZE (1..7))
-- indicates the length of the validity period or the absolute time of the validity
-- period termination as specified in 3GPP TS 23.040 [6].
-- the length of ValidityPeriod is either 1 octet or 7 octets

TransferredVolume ::= CHOICE {
    volumeIfNoTariffSwitch      [0] INTEGER (0..4294967295),
    volumeIfTariffSwitch        [1] SEQUENCE {
        volumeSinceLastTariffSwitch      [0] INTEGER (0..4294967295),
        volumeTariffSwitchInterval      [1] INTEGER (0..4294967295)           OPTIONAL
    }
}
-- volumeIfNoTariffSwitch, volumeSinceLastTariffSwitch and volumeTariffSwitchInterval
-- are measured in bytes.

TransferredVolumeRollOver ::= CHOICE {
    rO-VolumeIfNoTariffSwitch    [0] INTEGER (0..255),
    rO-VolumeIfTariffSwitch       [1] SEQUENCE {
        rO-VolumeSinceLastTariffSwitch    [0] INTEGER (0..255)           OPTIONAL,
        rO-VolumeTariffSwitchInterval     [1] INTEGER (0..255)           OPTIONAL
    }
}
-- rO-VolumeIfNoTariffSwitch, rO-VolumeSinceLastTariffSwitch and rO-VolumeTariffSwitchInterval
-- present counters indicating the number of parameter range rollovers.

UnavailableNetworkResource ::= ENUMERATED {
    unavailableResources          (0),
    componentFailure              (1),
    basicCallProcessingException  (2),
    resourceStatusFailure         (3),
    endUserFailure                (4)
}
-- Indicates the network resource that failed.

Units ::= INTEGER (1 .. 10000)

```

```

VariablePart {PARAMETERS-BOUND : bound} ::= CHOICE {
    integer          [0] Integer4,
    number           [1] Digits {bound}, -- Generic digits
    time             [2] OCTET STRING (SIZE(2)), -- HH: MM, BCD coded
    date             [3] OCTET STRING (SIZE(4)), -- YYYYMMDD, BCD coded
    price            [4] OCTET STRING (SIZE(4))
}
-- Indicates the variable part of the message. Time is BCD encoded.
-- The most significant hours digit occupies bits 0-3 of the first octet, and the least
-- significant digit occupies bits 4-7 of the first octet. The most significant minutes digit
-- occupies bits 0-3 of the second octet, and the least significant digit occupies bits 4-7
-- of the second octet.
--
-- Date is BCD encoded. The year digit indicating millenium occupies bits 0-3 of the first octet,
-- and the year digit indicating century occupies bits 4-7 of the first octet. The year digit
-- indicating decade occupies bits 0-3 of the second octet, whilst the digit indicating the year
-- within the decade occupies bits 4-7 of the second octet.
-- The most significant month digit occupies bits 0-3 of the third octet, and the least
-- significant month digit occupies bits 4-7 of the third octet. The most significant day digit
-- occupies bits 0-3 of the fourth octet, and the least significant day digit occupies bits 4-7
-- of the fourth octet.
-- Price is BCD encoded. The digit indicating hundreds of thousands occupies bits 0-3 of the
-- first octet, and the digit indicating tens of thousands occupies bits 4-7 of the first octet.
-- The digit indicating thousands occupies bits 0-3 of the second octet, whilst the digit
-- indicating hundreds occupies bits 4-7 of the second octet. The digit indicating tens occupies
-- bits 0-3 of the third octet, and the digit indicating 0 to 9 occupies bits 4-7 of the third
-- octet. The tenths digit occupies bits 0-3 of the fourth octet, and the hundredths digit
-- occupies bits 4-7 of the fourth octet.
--
-- For the encoding of digits in an octet, refer to the timeAndtimezone parameter

-- The Definition of range of constants follows
numOfInfoItems INTEGER ::= 4

END

```

5.2 Error types

```

CAP-errortypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-errortypes(51) version4(3)}
-- This module contains the type definitions for the CAP Error Types.
-- Where a parameter of type CHOICE is tagged with a specific tag value, the tag is automatically
-- replaced with an EXPLICIT tag of the same value.

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS

    ros-InformationObjects,
    datatypes,
    errorcodes

FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

ERROR

FROM Remote-Operations-Information-Objects ros-InformationObjects

    InvokeID,
    UnavailableNetworkResource

FROM CAP-datatypes datatypes

    errcode-canceled,
    errcode-cancelFailed,
    errcode-eTCFailed,
    errcode-improperCallerResponse,
    errcode-missingCustomerRecord,
    errcode-missingParameter,
    errcode-parameterOutOfRange,
    errcode-requestedInfoError,
    errcode-systemFailure,
    errcode-taskRefused,
    errcode-unavailableResource,
    errcode-unexpectedComponentSequence,
    errcode-unexpectedDataValue,
    errcode-unexpectedParameter,
    errcode-unknownLegID,
    errcode-unknownCSID,
    errcode-unknownPDPID

```

```

FROM CAP-errorcodes errorcodes
;

-- TYPE DEFINITION FOR CAP ERROR TYPES FOLLOWS

canceled ERROR ::= {
    CODE    errcode-canceled
}
-- The operation has been canceled.

cancelFailed ERROR ::= {
    PARAMETER SEQUENCE {
        problem [0] ENUMERATED {
            unknownOperation (0),
            tooLate (1),
            operationNotCancellable (2)
        },
        operation [1] InvokeID,
        ...
    }
    CODE    errcode-cancelFailed
}
-- The operation failed to be canceled.

eTCFailed ERROR ::= {
    CODE    errcode-eTCFailed
}
-- The establish temporary connection failed.

improperCallerResponse ERROR ::= {
    CODE    errcode-improperCallerResponse
}
-- The caller response was not as expected.

missingCustomerRecord ERROR ::= {
    CODE    errcode-missingCustomerRecord
}
-- The Service Logic Program could not be found in the gsmSCF.

missingParameter ERROR ::= {
    CODE    errcode-missingParameter
}
-- An expected optional parameter was not received.

parameterOutOfRange ERROR ::= {
    CODE    errcode-parameterOutOfRange
}
-- The parameter was not as expected (e.g. missing or out of range).

requestedInfoError ERROR ::= {
    PARAMETER ENUMERATED {
        unknownRequestedInfo (1),
        requestedInfoNotAvailable (2)
    }
    CODE    errcode-requestedInfoError
}
-- The requested information cannot be found.

systemFailure ERROR ::= {
    PARAMETER UnavailableNetworkResource
    CODE    errcode-systemFailure
}
-- The operation could not be completed due to a system failure at the serving physical entity.

taskRefused ERROR ::= {
    PARAMETER ENUMERATED {
        generic (0),
        unobtainable (1),
        congestion (2)
    }
    CODE    errcode-taskRefused
}
-- An entity normally capable of the task requested cannot or chooses not to perform the task at
-- this time. This includes error situations like congestion and unobtainable address as used in
-- e.g. the connect operation.)

unavailableResource ERROR ::= {

```



```

        CODE    errcode-unavailableResource
    }
-- A requested resource is not available at the serving entity.

unexpectedComponentSequence ERROR ::= {
    CODE    errcode-unexpectedComponentSequence
}
-- An incorrect sequence of Components was received (e.g."DisconnectForwardConnection"
-- followed by "PlayAnnouncement").

unexpectedDataValue ERROR ::= {
    CODE    errcode-unexpectedDataValue
}
-- The data value was not as expected (e.g. route number expected but billing number received)

unexpectedParameter ERROR ::= {
    CODE    errcode-unexpectedParameter
}
-- A parameter received was not expected.

unknownLegID ERROR ::= {
    CODE    errcode-unknownLegID
}
-- Leg not known to the gsmSSF.

unknownCSID ERROR ::= {
    CODE    errcode-unknownCSID
}
-- Call Segment not known to the gsmSSF.

unknownPDPID ERROR ::= {
    CODE    errcode-unknownPDPID
}
-- PDPID not known by the receiving entity.

END

```

5.3 Operation codes

```

CAP-operationcodes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-operationcodes(53) version4(3)}

```

```

DEFINITIONS ::= BEGIN

```

```

IMPORTS

```

```

    ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

```

```

    Code
FROM Remote-Operations-Information-Objects ros-InformationObjects

```

```

;

```

```

-- the operations are grouped by the identified operation packages.

```

```

-- gsmSCF activation Package
    opcode-initialDP                               Code ::= local: 0
-- gsmSCF/gsmSRF activation of assist Package
    opcode-assistRequestInstructions               Code ::= local: 16
-- Assist connection establishment Package
    opcode-establishTemporaryConnection           Code ::= local: 17
-- Generic disconnect resource Package
    opcode-disconnectForwardConnection            Code ::= local: 18
    opcode-dFCWithArgument                         Code ::= local: 86
-- Non-assisted connection establishment Package
    opcode-connectToResource                       Code ::= local: 19
-- Connect Package (elementary gsmSSF function)
    opcode-connect                                 Code ::= local: 20
-- Call handling Package (elementary gsmSSF function)
    opcode-releaseCall                             Code ::= local: 22
-- BCSM Event handling Package
    opcode-requestReportBCSMEvent                 Code ::= local: 23
    opcode-eventReportBCSM                        Code ::= local: 24

```

```

-- Charging Event handling Package
  opcode-requestNotificationChargingEvent      Code ::= local: 25
  opcode-eventNotificationCharging            Code ::= local: 26
-- gsmSSF call processing Package
  opcode-continue                             Code ::= local: 31
  opcode-continueWithArgument                 Code ::= local: 56
-- gsmSCF call initiation Package
  opcode-initiateCallAttempt                  Code ::= local: 32
-- Timer Package
  opcode-resetTimer                           Code ::= local: 33
-- Billing Package
  opcode-furnishChargingInformation           Code ::= local: 34
-- Charging Package
  opcode-applyCharging                        Code ::= local: 35
  opcode-applyChargingReport                  Code ::= local: 36
  opcode-playTone                             Code ::= local: 97
-- Traffic management Package
  opcode-callGap                              Code ::= local: 41
-- Call report Package
  opcode-callInformationReport                 Code ::= local: 44
  opcode-callInformationRequest                Code ::= local: 45
-- Signalling control Package
  opcode-sendChargingInformation              Code ::= local: 46
-- Specialized resource control Package
  opcode-playAnnouncement                     Code ::= local: 47
  opcode-promptAndCollectUserInformation      Code ::= local: 48
  opcode-specializedResourceReport            Code ::= local: 49
-- Cancel Package
  opcode-cancel                               Code ::= local: 53
-- Activity Test Package
  opcode-activityTest                         Code ::= local: 55
-- CPH Response Package
  opcode-continueWithArgument                 Code ::= local: 88
  opcode-disconnectLeg                        Code ::= local: 90
  opcode-moveLeg                              Code ::= local: 93
  opcode-splitLeg                             Code ::= local: 95
-- Exception Inform Package
  opcode-entityReleased                       Code ::= local: 96

-- Sms Activation Package
  opcode-initialDPSMS                         Code ::= local: 60
-- Sms Billing Package
  opcode-furnishChargingInformationSMS         Code ::= local: 61
-- Sms Connect Package
  opcode-connectSMS                           Code ::= local: 62
-- Sms Event Handling Package
  opcode-requestReportSMSEvent                Code ::= local: 63
  opcode-eventReportSMS                       Code ::= local: 64
-- Sms Processing Package
  opcode-continueSMS                          Code ::= local: 65
-- Sms Release Package
  opcode-releaseSMS                           Code ::= local: 66
-- Sms Timer Package
  opcode-resetTimerSMS                        Code ::= local: 67

-- Gprs Activity Test Package
  opcode-activityTestGPRS                     Code ::= local: 70
-- Gprs Charging Package
  opcode-applyChargingGPRS                    Code ::= local: 71
  opcode-applyChargingReportGPRS              Code ::= local: 72
-- Gprs Cancel Package
  opcode-cancelGPRS                           Code ::= local: 73
-- Gprs Connect Package
  opcode-connectGPRS                          Code ::= local: 74
-- Gprs Processing Package
  opcode-continueGPRS                         Code ::= local: 75
-- Gprs Exception Information Package
  opcode-entityReleasedGPRS                   Code ::= local: 76
-- Gprs Billing Package
  opcode-furnishChargingInformationGPRS        Code ::= local: 77
-- Gprs Scf Activation Package
  opcode-initialDPGPRS                         Code ::= local: 78
-- Gprs Release Package
  opcode-releaseGPRS                           Code ::= local: 79
-- Gprs Event Handling Package
  opcode-eventReportGPRS                      Code ::= local: 80
  opcode-requestReportGPRSEvent               Code ::= local: 81
-- Gprs Timer Package

```

```

opcode-resetTimerGPRS                               Code ::= local: 82
-- Gprs Charge Advice Package
opcode-sendChargingInformationGPRS                   Code ::= local: 83

```

END

5.4 Error codes

```

CAP-errorcodes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-errorcodes(57) version4(3)}

```

DEFINITIONS ::= BEGIN

IMPORTS

```

ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

```

```

Code
FROM Remote-Operations-Information-Objects ros-InformationObjects

```

;

```

errcode-canceled                               Code ::= local: 0
errcode-cancelFailed                           Code ::= local: 1
errcode-eTCFailed                              Code ::= local: 3
errcode-improperCallerResponse                 Code ::= local: 4
errcode-missingCustomerRecord                  Code ::= local: 6
errcode-missingParameter                       Code ::= local: 7
errcode-parameterOutOfRange                    Code ::= local: 8
errcode-requestedInfoError                     Code ::= local: 10
errcode-systemFailure                          Code ::= local: 11
errcode-taskRefused                            Code ::= local: 12
errcode-unavailableResource                     Code ::= local: 13
errcode-unexpectedComponentSequence            Code ::= local: 14
errcode-unexpectedDataValue                    Code ::= local: 15
errcode-unexpectedParameter                    Code ::= local: 16
errcode-unknownLegID                           Code ::= local: 17
errcode-unknownPDPID                           Code ::= local: 50
errcode-unknownCSID                            Code ::= local: 51

```

END

5.5 Classes

```

CAP-classes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-classes(54) version4(3)}

```

DEFINITIONS ::= BEGIN

IMPORTS

```

ROS-OBJECT-CLASS,
Code
FROM Remote-Operations-Information-Objects ros-InformationObjects

```

```

id-rosObject-gsmSRF,
id-rosObject-gsmSSF,
ros-InformationObjects,
gsmSSF-gsmSCF-Protocol,
gsmSCF-gsmSRF-Protocol
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

```

```

capSsfToScfGeneric,
capAssistHandoffssfToScf
FROM CAP-gsmSSF-gsmSCF-pkgs-contracts-acs gsmSSF-gsmSCF-Protocol

```

```

gsmSRF-gsmSCF-contract
FROM CAP-gsmSCF-gsmSRF-pkgs-contracts-acs gsmSCF-gsmSRF-Protocol

```

```

    CriticalityType
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1) cs2(20)
modules(0) in-cs2-datatypes(0) version1(0)}

;

gsmSSF ROS-OBJECT-CLASS ::= {
    INITIATES    {capSsfToScfGeneric|
                  capAssistHandoffssfToScf}
    RESPONDS    {capSsfToScfGeneric}
    ID          id-rosObject-gsmSSF}

gsmSRF ROS-OBJECT-CLASS ::= {
    INITIATES    {gsmSRF-gsmSCF-contract}
    ID          id-rosObject-gsmSRF}

EXTENSION ::= CLASS {
    &ExtensionType,
    &criticality          CriticalityType DEFAULT ignore,
    &id Code}

WITH SYNTAX {
    EXTENSION-SYNTAX    &ExtensionType
    CRITICALITY        &criticality
    IDENTIFIED BY      &id}
-- Example of addition of an extension named "Some Network Specific Indicator" of type
-- BOOLEAN, with criticality "abort" and to be identified as extension number 1
-- Example of definition using the above information object class:
--
-- SomeNetworkSpecificIndicator EXTENSION ::= {
--     EXTENSION-SYNTAX    BOOLEAN
--     CRITICALITY        abort
--     IDENTIFIED BY      local: 1
-- }

-- Example of transfer syntax, using the ExtensionField datatype as specified in clause 5.
-- Assuming the value of the extension is set to TRUE, the extensions parameter
-- becomes a Sequence of type INTEGER ::= 1, criticality ENUMERATED ::= 1 and value [1]
-- EXPLICIT BOOLEAN ::= TRUE.
--
-- Use of ITU-T Recommendation Q.1400 [52] defined Extension is for further study.
-- In addition the extension mechanism marker is used to identify the future minor additions
-- to CAP.

firstExtension EXTENSION ::= {
    EXTENSION-SYNTAX    NULL
    CRITICALITY        ignore
    IDENTIFIED BY      local: 1}
-- firstExtension is just an example.

SupportedExtensions EXTENSION ::= {firstExtension, ...
}
-- SupportedExtension is the full set of the network operator extensions.

PARAMETERS-BOUND ::= CLASS {
    &minAccessPointNameLength          INTEGER,
    &maxAccessPointNameLength          INTEGER,
    &minAchBillingChargingLength        INTEGER,
    &maxAchBillingChargingLength        INTEGER,
    &minAttributesLength                INTEGER,
    &maxAttributesLength                INTEGER,
    &maxBearerCapabilityLength          INTEGER,
    &minCalledPartyBCDNumberLength      INTEGER,
    &maxCalledPartyBCDNumberLength      INTEGER,
    &minCalledPartyNumberLength         INTEGER,
    &maxCalledPartyNumberLength         INTEGER,
    &minCallingPartyNumberLength        INTEGER,
    &maxCallingPartyNumberLength        INTEGER,
    &minCallResultLength                INTEGER,
    &maxCallResultLength                INTEGER,
    &minCarrierLength                  INTEGER,
    &maxCarrierLength                  INTEGER,
    &minCauseLength                    INTEGER,
    &maxCauseLength                    INTEGER,
    &minComponentLength                INTEGER,
    &maxComponentLength                INTEGER,
    &minDigitsLength                  INTEGER,

```

```

&maxDigitsLength                INTEGER,
&minFCIBillingChargingDataLength INTEGER,
&maxFCIBillingChargingDataLength INTEGER,
&minFCIBillingChargingLength    INTEGER,
&maxFCIBillingChargingLength    INTEGER,
&minGenericNumberLength         INTEGER,
&maxGenericNumberLength         INTEGER,
&minGPRSCauseLength             INTEGER,
&maxGPRSCauseLength             INTEGER,
&minIPSSPCapabilitiesLength     INTEGER,
&maxIPSSPCapabilitiesLength     INTEGER,
&minLocationNumberLength        INTEGER,
&maxLocationNumberLength        INTEGER,
&minMessageContentLength        INTEGER,
&maxMessageContentLength        INTEGER,
&minOriginalCalledPartyIDLength INTEGER,
&maxOriginalCalledPartyIDLength INTEGER,
&minPDPAddressLength            INTEGER,
&maxPDPAddressLength            INTEGER,
&minRedirectingPartyIDLength    INTEGER,
&maxRedirectingPartyIDLength    INTEGER,
&minScfIDLength                 INTEGER,
&maxScfIDLength                 INTEGER,
&minSCIBillingChargingLength    INTEGER,
&maxSCIBillingChargingLength    INTEGER,
&minTimeAndTimezoneLength       INTEGER,
&maxTimeAndTimezoneLength       INTEGER,
&numOfBCSMEvents                INTEGER,
&numOfSMSEvents                 INTEGER,
&numOfGPRSEvents                INTEGER,
&numOfExtensions                 INTEGER,
&numOfGenericNumbers             INTEGER,
&numOfMessageIDs                 INTEGER}

```

```

WITH SYNTAX {
MINIMUM-FOR-ACCESS-POINT-NAME    &minAccessPointNameLength
MAXIMUM-FOR-ACCESS-POINT-NAME    &maxAccessPointNameLength
MINIMUM-FOR-ACH-BILLING-CHARGING &minAChBillingChargingLength
MAXIMUM-FOR-ACH-BILLING-CHARGING &maxAChBillingChargingLength
MINIMUM-FOR-ATTRIBUTES           &minAttributesLength
MAXIMUM-FOR-ATTRIBUTES           &maxAttributesLength
MAXIMUM-FOR-BEARER-CAPABILITY     &maxBearerCapabilityLength
MINIMUM-FOR-CALLED-PARTY-BCD-NUMBER &minCalledPartyBCDNumberLength
MAXIMUM-FOR-CALLED-PARTY-BCD-NUMBER &maxCalledPartyBCDNumberLength
MINIMUM-FOR-CALLED-PARTY-NUMBER   &minCalledPartyNumberLength
MAXIMUM-FOR-CALLED-PARTY-NUMBER   &maxCalledPartyNumberLength
MINIMUM-FOR-CALLING-PARTY-NUMBER  &minCallingPartyNumberLength
MAXIMUM-FOR-CALLING-PARTY-NUMBER  &maxCallingPartyNumberLength
MINIMUM-FOR-CALL-RESULT           &minCallResultLength
MAXIMUM-FOR-CALL-RESULT           &maxCallResultLength
MINIMUM-FOR-CARRIER              &minCarrierLength
MAXIMUM-FOR-CARRIER              &maxCarrierLength
MINIMUM-FOR-CAUSE                 &minCauseLength
MAXIMUM-FOR-CAUSE                 &maxCauseLength
MINIMUM-FOR-COMPONENT             &minComponentLength
MAXIMUM-FOR-COMPONENT             &maxComponentLength
MINIMUM-FOR-DIGITS                &minDigitsLength
MAXIMUM-FOR-DIGITS                &maxDigitsLength
MINIMUM-FOR-FCI-BILLING-CHARGING-DATA &minFCIBillingChargingDataLength
MAXIMUM-FOR-FCI-BILLING-CHARGING-DATA &maxFCIBillingChargingDataLength
MINIMUM-FOR-FCI-BILLING-CHARGING &minFCIBillingChargingLength
MAXIMUM-FOR-FCI-BILLING-CHARGING &maxFCIBillingChargingLength
MINIMUM-FOR-GENERIC-NUMBER        &minGenericNumberLength
MAXIMUM-FOR-GENERIC-NUMBER        &maxGenericNumberLength
MINIMUM-FOR-GPRS-CAUSE-LENGTH     &minGPRSCauseLength
MAXIMUM-FOR-GPRS-CAUSE-LENGTH     &maxGPRSCauseLength
MINIMUM-FOR-IP-SSP-CAPABILITIES   &minIPSSPCapabilitiesLength
MAXIMUM-FOR-IP-SSP-CAPABILITIES   &maxIPSSPCapabilitiesLength
MINIMUM-FOR-LOCATION-NUMBER         &minLocationNumberLength
MAXIMUM-FOR-LOCATION-NUMBER         &maxLocationNumberLength
MINIMUM-FOR-MESSAGE-CONTENT       &minMessageContentLength
MAXIMUM-FOR-MESSAGE-CONTENT       &maxMessageContentLength
MINIMUM-FOR-ORIGINAL-CALLED-PARTY-ID &minOriginalCalledPartyIDLength
MAXIMUM-FOR-ORIGINAL-CALLED-PARTY-ID &maxOriginalCalledPartyIDLength
MINIMUM-FOR-PDP-ADDRESS-LENGTH    &minPDPAddressLength
MAXIMUM-FOR-PDP-ADDRESS-LENGTH    &maxPDPAddressLength
MINIMUM-FOR-REDIRECTING-ID        &minRedirectingPartyIDLength
MAXIMUM-FOR-REDIRECTING-ID        &maxRedirectingPartyIDLength

```

```

MINIMUM-FOR-GSMSCF-ID &minScfIDLength
MAXIMUM-FOR-GSMSCF-ID &maxScfIDLength
MINIMUM-FOR-SCI-BILLING-CHARGING &minSCIBillingChargingLength
MAXIMUM-FOR-SCI-BILLING-CHARGING &maxSCIBillingChargingLength
MINIMUM-FOR-TIME-AND-TIMEZONE &minTimeAndTimezoneLength
MAXIMUM-FOR-TIME-AND-TIMEZONE &maxTimeAndTimezoneLength
NUM-OF-BCSM-EVENT &numOfBCSMEvents
NUM-OF-SMS-EVENTS &numOfSMSEvents
NUM-OF-GPRS-EVENTS &numOfGPRSEvents
NUM-OF-EXTENSIONS &numOfExtensions
NUM-OF-GENERIC-NUMBERS &numOfGenericNumbers
NUM-OF-MESSAGE-IDS &numOfMessageIDs}

cAPSpecificBoundSet PARAMETERS-BOUND ::= {
    MINIMUM-FOR-ACCESS-POINT-NAME 1
    MAXIMUM-FOR-ACCESS-POINT-NAME 100
    MINIMUM-FOR-ACH-BILLING-CHARGING 5
    MAXIMUM-FOR-ACH-BILLING-CHARGING 177
    MINIMUM-FOR-ATTRIBUTES 2
    MAXIMUM-FOR-ATTRIBUTES 10
    MAXIMUM-FOR-BEARER-CAPABILITY 11
    MINIMUM-FOR-CALLED-PARTY-BCD-NUMBER 1
    MAXIMUM-FOR-CALLED-PARTY-BCD-NUMBER 41
    MINIMUM-FOR-CALLED-PARTY-NUMBER 2
    MAXIMUM-FOR-CALLED-PARTY-NUMBER 18
    MINIMUM-FOR-CALLING-PARTY-NUMBER 2
    MAXIMUM-FOR-CALLING-PARTY-NUMBER 10
    MINIMUM-FOR-CALL-RESULT 12
    MAXIMUM-FOR-CALL-RESULT 193
    MINIMUM-FOR-CARRIER 4
    MAXIMUM-FOR-CARRIER 4
    MINIMUM-FOR-CAUSE 2
    MAXIMUM-FOR-CAUSE 32
    MINIMUM-FOR-COMPONENT 1
    MAXIMUM-FOR-COMPONENT 2
    MINIMUM-FOR-DIGITS 2
    MAXIMUM-FOR-DIGITS 16
    MINIMUM-FOR-FCI-BILLING-CHARGING-DATA 1
    MAXIMUM-FOR-FCI-BILLING-CHARGING-DATA 160
    MINIMUM-FOR-FCI-BILLING-CHARGING 5
    MAXIMUM-FOR-FCI-BILLING-CHARGING 225
    MINIMUM-FOR-GENERIC-NUMBER 3
    MAXIMUM-FOR-GENERIC-NUMBER 11
    MINIMUM-FOR-GPRS-CAUSE-LENGTH 1
    MAXIMUM-FOR-GPRS-CAUSE-LENGTH 1
    MINIMUM-FOR-IP-SSP-CAPABILITIES 1
    MAXIMUM-FOR-IP-SSP-CAPABILITIES 4
    MINIMUM-FOR-LOCATION-NUMBER 2
    MAXIMUM-FOR-LOCATION-NUMBER 10
    MINIMUM-FOR-MESSAGE-CONTENT 1
    MAXIMUM-FOR-MESSAGE-CONTENT 127
    MINIMUM-FOR-ORIGINAL-CALLED-PARTY-ID 2
    MAXIMUM-FOR-ORIGINAL-CALLED-PARTY-ID 10
    MINIMUM-FOR-PDP-ADDRESS-LENGTH 1
    MAXIMUM-FOR-PDP-ADDRESS-LENGTH 63
    MINIMUM-FOR-REDIRECTING-ID 2
    MAXIMUM-FOR-REDIRECTING-ID 10
    MINIMUM-FOR-GSMSCF-ID 2
    MAXIMUM-FOR-GSMSCF-ID 10
    MINIMUM-FOR-SCI-BILLING-CHARGING 4
    MAXIMUM-FOR-SCI-BILLING-CHARGING 124
    MINIMUM-FOR-TIME-AND-TIMEZONE 8
    MAXIMUM-FOR-TIME-AND-TIMEZONE 8
    NUM-OF-BCSM-EVENT 10
    NUM-OF-SMS-EVENTS 10
    NUM-OF-GPRS-EVENTS 10
    NUM-OF-EXTENSIONS 10
    NUM-OF-GENERIC-NUMBERS 5
    NUM-OF-MESSAGE-IDS 16}

```

END

5.6 Object Identifiers (IDs)

CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)}

```
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

DEFINITIONS ::= BEGIN

-- This module assigns object identifiers for Modules, Packages, Contracts and ACs
-- used by CAP

-- For Modules from TC, ROS,
tc-Messages OBJECT IDENTIFIER ::=
  {itu-t recommendation q 773 modules(2) messages(1) version3(3)}

tc-NotationExtensions OBJECT IDENTIFIER ::=
  {itu-t recommendation q 775 modules(2) notation-extension (4) version1(1)}

ros-InformationObjects OBJECT IDENTIFIER ::=
  {joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)}

-- For CAP Modules
datatypes OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-datatypes(52) version3(2)}

errortypes OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-errortypes(51) version3(2)}

operationcodes OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-operationcodes(53) version3(2)}

errorcodes OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-errorcodes(57) version3(2)}

classes OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-classes(54) version3(2)}

gsmSSF-gsmSCF-Operations OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-gsmSSF-gsmSCF-ops-args(101) version3(2)}

gsmSSF-gsmSCF-Protocol OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-gsmSSF-gsmSCF-pkgs-contracts-ac(102) version3(2)}

gsmSCF-gsmSRF-Operations OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-gsmSCF-gsmSRF-ops-args(103) version3(2)}

gsmSCF-gsmSRF-Protocol OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-gsmSCF-gsmSRF-pkgs-contracts-ac(104) version3(2)}

sms-Operations OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-SMS-ops-args(105) version4(3)}

smsSSF-gsmSCF-Protocol OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-smsSSF-gsmSCF-pkgs-contracts-ac(106) version4(3)}

gprsSSF-gsmSCF-Operations OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-GPRS-ops-args(107) version3(2)}

gprsSSF-gsmSCF-Protocol OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1) modules(3)
  cap-gprsSSF-gsmSCF-pkgs-contracts-ac(108) version3(2)}

id-CAP OBJECT IDENTIFIER ::=
  {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
  umts-network(1) cap4(22)}

id-CAP3 OBJECT IDENTIFIER ::=
```

```

    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
      umts-network(1) cap3(20)}

id-CAPOE                                OBJECT IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
      umts-network(1) cap40E(23)}

id-CAP3OE                                OBJECT IDENTIFIER ::=
    {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
      umts-network(1) cap30E(21)}

id-ac                                     OBJECT IDENTIFIER ::= {id-CAP      ac(3)}
id-ac3                                   OBJECT IDENTIFIER ::= {id-CAP3     ac(3)}
id-acE                                   OBJECT IDENTIFIER ::= {id-CAPOE    ac(3)}
id-ac3E                                  OBJECT IDENTIFIER ::= {id-CAP30E   ac(3)}
id-as                                     OBJECT IDENTIFIER ::= {id-CAP      as(5)}
id-asE                                   OBJECT IDENTIFIER ::= {id-CAPOE    as(5)}
id-rosObject                             OBJECT IDENTIFIER ::= {id-CAP      rosObject(25)}
id-contract                              OBJECT IDENTIFIER ::= {id-CAP      contract(26)}
id-contract3                             OBJECT IDENTIFIER ::= {id-CAP3     contract(26)}
id-contractE                             OBJECT IDENTIFIER ::= {id-CAPOE    contract(26)}
id-package                               OBJECT IDENTIFIER ::= {id-CAP      package(27)}
id-packageE                              OBJECT IDENTIFIER ::= {id-CAPOE    package(27)}

-- for ac, as, rosObject, contract and package, the values are identical to ITU-T Recommendation
-- Q.1218 [49]

-- ROS Objects

id-rosObject-gsmSCF                      OBJECT IDENTIFIER ::= {id-rosObject 4}
id-rosObject-gsmSSF                      OBJECT IDENTIFIER ::= {id-rosObject 5}
id-rosObject-gsmSRF                      OBJECT IDENTIFIER ::= {id-rosObject 6}

-- Application Contexts

-- gsmSSF/gsmSCF AC
id-ac-CAP-gsmSSF-scfGenericAC            OBJECT IDENTIFIER ::= {id-acE 4}
id-ac-CAP-gsmSSF-scfAssistHandoffAC     OBJECT IDENTIFIER ::= {id-acE 6}
id-ac-CAP-scf-gsmSSFGenericAC            OBJECT IDENTIFIER ::= {id-acE 8}

-- gsmSRF/gsmSCF AC
id-ac-gsmSRF-gsmSCF                     OBJECT IDENTIFIER ::= {id-ac 14}

-- gprsSSF/gsmSCF AC
id-ac-CAP-gprsSSF-gsmSCF-AC              OBJECT IDENTIFIER ::= {id-acE 50}
id-ac-CAP-gsmSCF-gprsSSF-AC              OBJECT IDENTIFIER ::= {id-acE 51}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF AC
id-ac-cap3-sms-AC                         OBJECT IDENTIFIER ::= {id-ac3E 61}
id-ac-cap4-sms-AC                         OBJECT IDENTIFIER ::= {id-acE 61}

-- gsmSSF/gsmSCF Contracts
id-CAPSSfToScfGeneric                    OBJECT IDENTIFIER ::= {id-contractE 3}
id-CAPAssistHandoffSsfToScf              OBJECT IDENTIFIER ::= {id-contractE 5}
id-CAPScfToSsfGeneric                    OBJECT IDENTIFIER ::= {id-contractE 6}

-- gsmSRF/gsmSCF Contracts
id-contract-gsmSRF-gsmSCF                OBJECT IDENTIFIER ::= {id-contract 13}

-- gprsSSF/gsmSCF Contracts
id-cap3GprsSsfToGsmScf                   OBJECT IDENTIFIER ::= {id-contract 14}
id-cap3GsmScfToGprsSsf                   OBJECT IDENTIFIER ::= {id-contract 15}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF Contracts
id-cap3SmsSsfToGsmScf                    OBJECT IDENTIFIER ::= {id-contract3 16}
id-cap4SmsSsfToGsmScf                    OBJECT IDENTIFIER ::= {id-contract 16}

-- gsmSSF/gsmSCF Operation Packages
id-package-scfActivation                   OBJECT IDENTIFIER ::= {id-package 11}
id-package-gsmSRF-scfActivationOfAssist   OBJECT IDENTIFIER ::= {id-package 15}
id-package-assistConnectionEstablishment OBJECT IDENTIFIER ::= {id-package 16}
id-package-genericDisconnectResource      OBJECT IDENTIFIER ::= {id-package 17}
id-package-nonAssistedConnectionEstablishment OBJECT IDENTIFIER ::= {id-package 18}
id-package-connect                        OBJECT IDENTIFIER ::= {id-package 19}
id-package-callHandling                   OBJECT IDENTIFIER ::= {id-packageE 20}
id-package-bcsmEventHandling              OBJECT IDENTIFIER ::= {id-package 21}
id-package-chargingEventHandling          OBJECT IDENTIFIER ::= {id-package 23}
id-package-ssfCallProcessing              OBJECT IDENTIFIER ::= {id-packageE 24}
id-package-scfCallInitiation              OBJECT IDENTIFIER ::= {id-package 25}

```



```

id-package-timer OBJECT IDENTIFIER ::= {id-package 26}
id-package-billing OBJECT IDENTIFIER ::= {id-package 27}
id-package-charging OBJECT IDENTIFIER ::= {id-package 28}
id-package-trafficManagement OBJECT IDENTIFIER ::= {id-package 29}
id-package-callReport OBJECT IDENTIFIER ::= {id-package 32}
id-package-signallingControl OBJECT IDENTIFIER ::= {id-package 33}
id-package-activityTest OBJECT IDENTIFIER ::= {id-package 34}
id-package-cancel OBJECT IDENTIFIER ::= {id-packageE 36}
id-package-cphResponse OBJECT IDENTIFIER ::= {id-package 37}
id-package-exceptionInform OBJECT IDENTIFIER ::= {id-package 38}

-- gsmSRF/gsmSCF Operation Packages
id-package-specializedResourceControl OBJECT IDENTIFIER ::= {id-package 42}
id-package-gsmSRF-scfCancel OBJECT IDENTIFIER ::= {id-package 43}

-- gprsSSF/gsmSCF Operation Packages
id-package-gprsContinue OBJECT IDENTIFIER ::= {id-package 49}
id-package-gprsExceptionInformation OBJECT IDENTIFIER ::= {id-package 50}
id-package-gprsSCFActivation OBJECT IDENTIFIER ::= {id-package 51}
id-package-gprsConnect OBJECT IDENTIFIER ::= {id-package 52}
id-package-gprsRelease OBJECT IDENTIFIER ::= {id-package 53}
id-package-gprsEventHandling OBJECT IDENTIFIER ::= {id-package 54}
id-package-gprsSCFTimer OBJECT IDENTIFIER ::= {id-package 55}
id-package-gprsSCFBilling OBJECT IDENTIFIER ::= {id-package 56}
id-package-gprsSCFCharging OBJECT IDENTIFIER ::= {id-package 57}
id-package-gprsSCFActivityTest OBJECT IDENTIFIER ::= {id-package 58}
id-package-gprsSCFCancel OBJECT IDENTIFIER ::= {id-package 59}
id-package-gprsSCFChargeAdvice OBJECT IDENTIFIER ::= {id-package 60}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF Operation Packages
id-package-smsActivation OBJECT IDENTIFIER ::= {id-package 61}
id-package-smsConnect OBJECT IDENTIFIER ::= {id-package 62}
id-package-smsContinue OBJECT IDENTIFIER ::= {id-package 63}
id-package-smsRelease OBJECT IDENTIFIER ::= {id-package 64}
id-package-smsEventHandling OBJECT IDENTIFIER ::= {id-package 65}
id-package-smsBilling OBJECT IDENTIFIER ::= {id-package 66}
id-package-smsTimer OBJECT IDENTIFIER ::= {id-package 67}

-- gsmSSF/gsmSCF Abstract Syntaxes
id-as-gsmSSF-scfGenericAS OBJECT IDENTIFIER ::= {id-asE 4}
id-as-assistHandoff-gsmSSF-scfAS OBJECT IDENTIFIER ::= {id-asE 6}
id-as-scf-gsmSSFGenericAS OBJECT IDENTIFIER ::= {id-asE 7}

-- gsmSRF/gsmSCF Abstract Syntaxes
id-as-basic-gsmSRF-gsmSCF OBJECT IDENTIFIER ::= {id-as 14}

-- gprsSSF/gsmSCF Abstract Syntaxes
id-as-gprsSSF-gsmSCF-AS OBJECT IDENTIFIER ::= {id-as 50}
id-as-gsmSCF-gprsSSF-AS OBJECT IDENTIFIER ::= {id-as 51}

-- gprsSSF/gsmSCF or gsmSSF/gsmSCF Abstract Syntaxes
id-as-smsSSF-gsmSCF-AS OBJECT IDENTIFIER ::= {id-as 61}

END

```

5.7 User Abort Data

```

CAP-U-ABORT-Data {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-u-abort-data(110) version3(2)}

```

```

DEFINITIONS ::= BEGIN

```

```

id-CAP-U-ABORT-Reason OBJECT IDENTIFIER ::= {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0) umts-Network(1) as(1) cap-u-abort-reason(2) version3(2)}

```

```

cap-U-ABORT-Reason-Abstract-Syntax ABSTRACT-SYNTAX ::= {CAP-U-ABORT-REASON IDENTIFIED BY
id-CAP-U-ABORT-Reason}

```

```

CAP-U-ABORT-REASON ::= ENUMERATED {
    no-reason-given (1),
    application-timer-expired (2),
    not-allowed-procedures (3),
    abnormal-processing (4),
    congestion (5),
    invalid-reference (6),
    missing-reference (7),
    overlapping-dialogue (8)
}

```

```
-- application-timer-expired shall be set when application timer (e.g. Tssf) is expired.
-- not-allowed-procedures shall be set when received signal is not allowed in CAP
-- procedures.
-- For example, when a class 4 operation is received from the
-- gsmSCF and the operation is not allowed in gsmSSF FSM.
-- (gsmSSF FSM cannot continue state transition). (e.g. ReleaseCall
-- operation received in Waiting for End of Temporary Connection
-- state.)
-- abnormal-processing shall be set when abnormal procedures occur at entity action.
-- congestion shall be set when requested resource is unavailable due to
-- congestion at TC user (CAP) level.
-- invalid-reference shall be set if the received destinationReference is unknown or
-- for a known destination Reference the received originationReference
-- does not match with the stored originationReference.
-- This abort reason is used for CAP defined GPRS-ReferenceNumber.
-- missing-reference shall be set when the destinationReference or the
-- originationReference is absent in the received message but is
-- required to be present according to the procedures in
-- subclause 14.1.7.
-- This abort reason is used for CAP defined GPRS-ReferenceNumber.
-- overlapping-dialogue shall be used by the gprsSSF to indicate to the gsmSCF that a
-- specific instance already has a TC dialogue open. This error
-- cause is typically obtained when both the gsmSCF and gprsSSF
-- open a new dialogue at the same time.
-- no-reason-given shall be set when any other reasons above do not apply
END -- of CAP-U-ABORT-Data
```

6 Circuit Switched Call Control

6.1 gsmSSF/CCF - gsmSCF Interface

6.1.1 Operations and arguments

```

CAP-gsmSSF-gsmSCF-ops-args {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-gsmSSF-gsmSCF-ops-args(101) version4(3)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- This module contains the operations and operation arguments used for the
-- gsmSSF - gsmSCF interface, for the control of circuit switched calls.

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

IMPORTS

    errortypes,
    datatypes,
    operationcodes,
    classes,
    tc-Messages,
    ros-InformationObjects
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

    CallingPartysCategory,
    HighLayerCompatibility,
    RedirectionInformation,
    ServiceKey
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
modules(0) cs1-datatypes(2) version1(0)}

    CallSegmentID {},
    MiscCallInfo
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
cs2(20) modules(0) in-cs2-datatypes (0) version1(0)}

    Ext-BasicServiceCode,
    IMEI,
    IMSI,
    ISDN-AddressString
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}

    CUG-Index,
    CUG-Interlock,
    CUG-Info,
    LocationInformation,
    MS-Classmark2,
    SubscriberState,
    SupportedCamelPhases,
    SupportedCamel4Subsets
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}

    CallReferenceNumber,
    SuppressionOfAnnouncement
FROM MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-CH-DataTypes(13) version6(6)}

    PARAMETERS-BOUND
FROM CAP-classes classes

    opcode-activityTest,
    opcode-applyCharging,
    opcode-applyChargingReport,

```

```

opcode-assistRequestInstructions,
opcode-callGap,
opcode-callInformationReport,
opcode-callInformationRequest,
opcode-cancel,
opcode-connect,
opcode-connectToResource,
opcode-continue,
opcode-continueWithArgument,
opcode-disconnectForwardConnection,
opcode-dFCWithArgument,
opcode-disconnectLeg,
opcode-entityReleased,
opcode-establishTemporaryConnection,
opcode-eventReportBCSM,
opcode-furnishChargingInformation,
opcode-initialDP,
opcode-initiateCallAttempt,
opcode-moveLeg,
opcode-playTone,
opcode-releaseCall,
opcode-requestReportBCSMEvent,
opcode-resetTimer,
opcode-sendChargingInformation,
opcode-splitLeg

```

FROM CAP-operationcodes operationcodes

```

AChBillingChargingCharacteristics {},
AdditionalCallingPartyNumber {},
AlertingPattern,
AssistingSSPIPRoutingAddress {},
BCSMEvent,
BCSM-Failure,
BearerCapability {},
CalledPartyNumber {},
CalledPartyBCDNumber {},
CallingPartyNumber {},
CallResult {},
CallSegmentToCancel,
CallSegmentFailure,
Carrier,
Cause {},
CGEncountered,
ChargeNumber {},
ControlType,
CorrelationID {},
DestinationRoutingAddress {},
EventSpecificChargingInformation {},
EventSpecificInformationBCSM {},
EventTypeBCSM,
EventTypeChargingPLMN,
Extensions {},
FCIBillingChargingCharacteristics {},
GapCriteria {},
GapIndicators,
GapTreatment,
GenericNumbers {},
InvokeID,
IPRoutingAddress {},
IPSSPCapabilities {},
leg1,
LegOrCallSegment {},
LocationNumber {},
MonitorMode,
NAoliInfo,
OCSIAplicable,
OriginalCalledPartyID {},
ReceivingSideID,
RedirectingPartyID {},
RequestedInformationList {},
RequestedInformationTypeList,
ScfID {},
SCIBillingChargingCharacteristics {},
SendingSideID,
ServiceInteractionIndicatorsTwo,
TimeAndTimezone {},
TimerID,
TimerValue

```

```
FROM CAP-datatypes datatypes
```

```
cancelFailed,
eTCFailed,
missingCustomerRecord,
missingParameter,
parameterOutOfRange,
requestedInfoError,
systemFailure,
taskRefused,
unexpectedComponentSequence,
unexpectedDataValue,
unexpectedParameter,
unknownLegID,
unknownCSID
```

```
FROM CAP-erroratypes erroratypes
```

```
;
```

```
activityTest OPERATION ::= {
```

```
RETURN RESULT TRUE
CODE opcode-activityTest
```

```
-- Direction: gsmSCF -> gsmSSF, Timer: Tat
```

```
-- This operation is used to check for the continued existence of a relationship
-- between the gsmSCF and gsmSSF, assist gsmSSF or gsmSRF. If the relationship is
-- still in existence, then the gsmSSF will respond. If no reply is received,
-- then the gsmSCF will assume that the gsmSSF, assist gsmSSF or gsmSRF has failed
-- in some way.
```

```
applyCharging {PARAMETERS-BOUND : bound} OPERATION ::= {
```

```
ARGUMENT ApplyChargingArg {bound}
RETURN RESULT FALSE
ERRORS {missingParameter |
unexpectedComponentSequence |
unexpectedParameter |
unexpectedDataValue |
parameterOutOfRange |
systemFailure |
taskRefused |
unknownLegID |
unknownCSID}
```

```
CODE opcode-applyCharging}
```

```
-- Direction: gsmSCF -> gsmSSF, Timer: Tac
```

```
-- This operation is used for interacting from the gsmSCF with the gsmSSF charging mechanisms.
-- The ApplyChargingReport operation provides the feedback from the gsmSSF to the gsmSCF.
```

```
ApplyChargingArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
```

```
aChBillingChargingCharacteristics [0] AChBillingChargingCharacteristics {bound},
partyToCharge [2] SendingSideID DEFAULT sendingSideID : leg1,
extensions [3] Extensions {bound} OPTIONAL,
aChChargingAddress [50] AChChargingAddress {bound}
DEFAULT legID:sendingSideID:leg1,
```

```
...
}
```

```
-- The partyToCharge parameter indicates the party in the call to which the ApplyCharging operation
-- shall be applied.
```

```
applyChargingReport {PARAMETERS-BOUND : bound} OPERATION ::= {
```

```
ARGUMENT ApplyChargingReportArg {bound}
RETURN RESULT FALSE
ERRORS {missingParameter |
unexpectedComponentSequence |
unexpectedParameter |
unexpectedDataValue |
parameterOutOfRange |
systemFailure |
taskRefused}
```

```
CODE opcode-applyChargingReport}
```

```
-- Direction: gsmSSF -> gsmSCF, Timer: Tacr
```

```
-- This operation is used by the gsmSSF to report to the gsmSCF the occurrence of a
-- specific charging event as requested by the gsmSCF using the ApplyCharging operation.
```

```
ApplyChargingReportArg {PARAMETERS-BOUND : bound} ::= CallResult {bound}
```

```
assistRequestInstructions {PARAMETERS-BOUND : bound} OPERATION ::= {
```

```
ARGUMENT AssistRequestInstructionsArg {bound}
RETURN RESULT FALSE
```

```

ERRORS          {missingCustomerRecord |
                 missingParameter |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter}
CODE            opcode-assistRequestInstructions}
-- Direction:  gsmSSF -> gsmSCF or gsmSRF -> gsmSCF, Timer: Tari
-- This operation is used when there is an assist procedure and may be
-- sent by the gsmSSF or gsmSRF to the gsmSCF. This operation is sent by the
-- assisting gsmSSF to gsmSCF, when the initiating gsmSSF has set up a connection to
-- the gsmSRF or to the assisting gsmSSF as a result of receiving an
-- EstablishTemporaryConnection from
-- the gsmSCF.
-- Refer to clause 11 for a description of the procedures associated with this operation.

AssistRequestInstructionsArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    correlationID          [0] CorrelationID {bound},
    iPSSPCapabilities     [2] IPSSPCapabilities {bound},
    extensions             [3] Extensions {bound}                OPTIONAL,
    ...
}
-- OPTIONAL denotes network operator specific use. The value of the correlationID may be the
-- Called Party Number supplied by the initiating gsmSSF.

callGap {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          CallGapArg {bound}
    RETURN RESULT     FALSE
    ALWAYS RESPONDS  FALSE
    CODE              opcode-callGap}
-- Direction:  gsmSCF -> gsmSSF, Timer: Tcg
-- This operation is used to request the gsmSSF to reduce the rate at which specific service
-- requests are sent to the gsmSCF.

CallGapArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    gapCriteria         [0] GapCriteria {bound},
    gapIndicators       [1] GapIndicators,
    controlType         [2] ControlType                OPTIONAL,
    gapTreatment        [3] GapTreatment {bound}       OPTIONAL,
    extensions          [4] Extensions {bound}        OPTIONAL,
    ...
}
-- OPTIONAL denotes network operator optional. If gapTreatment is not present, then the gsmSSF will
-- use a default treatment depending on network operator implementation.

callInformationReport {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          CallInformationReportArg {bound}
    RETURN RESULT     FALSE
    ALWAYS RESPONDS  FALSE
    CODE              opcode-callInformationReport}
-- Direction:  gsmSSF -> gsmSCF, Timer: Tcirp
-- This operation is used to send specific call information for a single call party to the gsmSCF as
-- requested by the gsmSCF in a previous CallInformationRequest.

CallInformationReportArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    RequestedInformationList [0] RequestedInformationList {bound},
    extensions               [2] Extensions {bound}                OPTIONAL,
    legID                   [3] ReceivingSideID                   OPTIONAL,
    ...
}

callInformationRequest {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          CallInformationRequestArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
                     parameterOutOfRange |
                     requestedInfoError |
                     systemFailure |
                     taskRefused |
                     unexpectedComponentSequence |
                     unexpectedDataValue |
                     unexpectedParameter |
                     unknownLegID}
    CODE              opcode-callInformationRequest}
-- Direction:  gsmSCF -> gsmSSF, Timer: Tcirq

```

-- This operation is used to request the gsmSSF to record specific information about a single
 -- call party and report it to the gsmSCF (with a CallInformationReport operation).

```
CallInformationRequestArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  requestedInformationTypeList [0] RequestedInformationTypeList {bound},
  extensions [2] Extensions {bound} OPTIONAL,
  legID [3] SendingSideID OPTIONAL,
  ...
}
```

-- OPTIONAL denotes network operator optional.

```
cancel OPERATION ::= {
  ARGUMENT CancelArg
  RETURN RESULT FALSE
  ERRORS {cancelFailed |
  missingParameter |
  taskRefused |
  unknownCSID}
  CODE opcode-cancel}
-- Direction: gsmSCF -> gsmSSF, or gsmSCF -> gsmSRF, Timer: Tcan
-- This operation cancels the correlated previous operation or all previous requests. The following
-- operations can be canceled: PlayAnnouncement, PromptAndCollectUserInformation.
```

```
CancelArg ::= CHOICE {
  invokeID [0] InvokeID,
  allRequests [1] NULL,
  callSegmentToCancel [2] CallSegmentToCancel
}
-- The InvokeID has the same value as that which was used for the operation to be cancelled.
```

```
connect {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT ConnectArg {bound}
  RETURN RESULT FALSE
  ERRORS {missingParameter |
  parameterOutOfRange |
  systemFailure |
  taskRefused |
  unexpectedComponentSequence |
  unexpectedDataValue |
  unexpectedParameter}
  CODE opcode-connect}
-- Direction: gsmSCF-> gsmSSF, Timer: Tcon
-- This operation is used to request the gsmSSF to perform the call processing actions
-- to route or forward a call to a specified destination.
```

```
ConnectArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  destinationRoutingAddress [0] DestinationRoutingAddress {bound},
  alertingPattern [1] AlertingPattern OPTIONAL,
  originalCalledPartyID [6] OriginalCalledPartyID {bound} OPTIONAL,
  extensions [10] Extensions {bound} OPTIONAL,
  carrier [11] Carrier {bound} OPTIONAL,
  callingPartysCategory [28] CallingPartysCategory OPTIONAL,
  redirectingPartyID [29] RedirectingPartyID {bound} OPTIONAL,
  redirectionInformation [30] RedirectionInformation OPTIONAL,
  genericNumbers [14] GenericNumbers {bound} OPTIONAL,
  serviceInteractionIndicatorsTwo [15] ServiceInteractionIndicatorsTwo OPTIONAL,
  chargeNumber [19] ChargeNumber {bound} OPTIONAL,
  legToBeCreated [21] LegID OPTIONAL,
  cug-Interlock [31] CUG-Interlock OPTIONAL,
  cug-OutgoingAccess [32] NULL OPTIONAL,
  suppressionOfAnnouncement [55] SuppressionOfAnnouncement OPTIONAL,
  oCSIApplicable [56] OCSIApplicable OPTIONAL,
  naOliInfo [57] NAOliInfo OPTIONAL,
  bor-InterrogationRequested [58] NULL OPTIONAL,
  ...
}
-- na-Info is included at the discretion of the gsmSCF operator.
```

```
connectToResource {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT ConnectToResourceArg {bound}
  RETURN RESULT FALSE
  ERRORS {missingParameter |
  systemFailure |
  taskRefused |
  unexpectedComponentSequence |
  unexpectedDataValue |
  unexpectedParameter |
  unknownCSID}
```

```

CODE          opcode-connectToResource}
-- Direction: gsmSCF -> gsmSSF, Timer: Tctr
-- This operation is used to connect a call segment from the gsmSSF to the
-- gsmSRF.

ConnectToResourceArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    resourceAddress CHOICE {
        ipRoutingAddress          [0] IPRoutingAddress {bound},
        none                      [3] NULL
    },
    extensions                    [4] Extensions {bound}                OPTIONAL,
    serviceInteractionIndicatorsTwo [7] ServiceInteractionIndicatorsTwo  OPTIONAL,
    callSegmentID                 [50] CallSegmentID {bound}            OPTIONAL,
    ...
}

continue OPERATION ::= {
    RETURN RESULT    FALSE
    ALWAYS RESPONDS FALSE
    CODE            opcode-continue}
-- Direction: gsmSCF -> gsmSSF, Timer: Tcue
-- This operation is used to request the gsmSSF to proceed with call processing at the
-- DP at which it previously suspended call processing to await gsmSCF instructions
-- (i.e. proceed to the next point in call in the BCSM). The gsmSSF continues call
-- processing without substituting new data from gsmSCF.

continueWithArgument {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT    ContinueWithArgumentArg {bound}
    RETURN RESULT    FALSE
    ERRORS        {missingParameter |
                  parameterOutOfRange |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |
                  unknownLegID}
    CODE          opcode-continueWithArgument}
-- Direction: gsmSCF -> gsmSSF, Timer: Tcwa
-- This operation is used to request the gsmSSF to proceed with call processing at the
-- DP at which it previously suspended call processing to await gsmSCF instructions
-- (i.e. proceed to the next point in call in the BCSM). The gsmSSF continues call
-- processing with the modified call setup information as received from the gsmSCF.

ContinueWithArgumentArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    legID                [0] LegID                OPTIONAL,
    alertingPattern      [1] AlertingPattern      OPTIONAL,
    extensions           [6] Extensions {bound}   OPTIONAL,
    serviceInteractionIndicatorsTwo [7] ServiceInteractionIndicatorsTwo  OPTIONAL,
    callingPartysCategory [12] CallingPartysCategory  OPTIONAL,
    genericNumbers       [16] GenericNumbers {bound}  OPTIONAL,
    cug-Interlock       [17] CUG-Interlock        OPTIONAL,
    cug-OutgoingAccess  [18] NULL                OPTIONAL,
    chargeNumber        [50] ChargeNumber {bound}   OPTIONAL,
    carrier             [52] Carrier {bound}      OPTIONAL,
    suppressionOfAnnouncement [55] SuppressionOfAnnouncement  OPTIONAL,
    naOliInfo          [56] NAOliInfo            OPTIONAL,
    bor-InterrogationRequested [57] NULL          OPTIONAL,
    suppress-O-CSI     [58] NULL                OPTIONAL,
    continueWithArgumentArgExtension [59] ContinueWithArgumentArgExtension  OPTIONAL,
    ...
}

ContinueWithArgumentArgExtension ::= SEQUENCE {
    suppress-D-CSI      [0] NULL                OPTIONAL,
    suppress-N-CSI     [1] NULL                OPTIONAL,
    suppressOutgoingCallBarring [2] NULL        OPTIONAL,
    ...
}

disconnectForwardConnection OPERATION ::= {
    RETURN RESULT    FALSE
    ERRORS        {systemFailure |
                  taskRefused |
                  unexpectedComponentSequence}
    CODE          opcode-disconnectForwardConnection}
-- Direction: gsmSCF -> gsmSSF, Timer: Tdfc
-- This operation is used to disconnect a forward temporary connection or a connection to a
-- resource. Refer to clause 11 for a description of the procedures associated with this operation.

```



```

DisconnectForwardConnectionWithArgument {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      DisconnectForwardConnectionWithArgumentArg {bound}
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter |
                 unknownLegID |
                 unknownCSID}
  CODE          opcode-dFCWithArgument}
-- Direction: gsmSCF -> gsmSSF, Timer Tdfcwa
-- This operation is used to disconnect a forward temporary connection or a connection to a
-- resource. Refer to clause 11 for a description of the procedures associated with this operation.

DisconnectForwardConnectionWithArgumentArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  callSegmentID      [1] CallSegmentID {bound}          OPTIONAL,
  extensions          [2] Extensions {bound}            OPTIONAL,
  ...
}

disconnectLeg {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      DisconnectLegArg {bound}
  RETURN RESULT TRUE
  ERRORS        {missingParameter |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter |
                 unknownLegId}
  CODE          opcode-disconnectLeg}
-- Direction: gsmSCF -> gsmSSF, Timer Td1
-- This operation is used by the gsmSCF to release a specific leg associated with the call and
-- retain any other legs not specified in the DisconnectLeg. Refer to clause 11 for a description
-- of the procedures associated with this operation.

DisconnectLegArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  LegToBeReleased   [0] LegID,
  ReleaseCause      [1] Cause {bound}                  OPTIONAL,
  extensions         [2] Extensions {bound}            OPTIONAL,
  ...
}

entityReleased {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      EntityReleasedArg {bound},
  RETURN RESULT FALSE
  ALWAYS RESPONDS FALSE
  CODE          opcode-entityReleased}
-- Direction: gsmSSF -> gsmSCF, Timer: Ter
-- This operation is used by the gsmSSF to inform the gsmSCF of an error or exception

EntityReleasedArg {PARAMETERS-BOUND : bound} ::= CHOICE {
  CallSegmentFailure [0] CallSegmentFailure,
  bCSM-Failure       [1] BCSM-Failure
}

establishTemporaryConnection {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      EstablishTemporaryConnectionArg {bound}
  RETURN RESULT FALSE
  ERRORS        {eTCFailed |
                 missingParameter |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter |
                 unknownCSID}
  CODE          opcode-establishTemporaryConnection}
-- Direction: gsmSCF -> gsmSSF, Timer: Tetc
-- This operation is used to create a connection to a resource for a limited period
-- of time (e.g. to play an announcement, to collect user information); it implies
-- the use of the assist procedure. Refer to clause 11 for a description of the
-- procedures associated with this operation.

```

```

EstablishTemporaryConnectionArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    assistingSSPIPRoutingAddress [0] AssistingSSPIPRoutingAddress {bound},
    correlationID [1] CorrelationID {bound} OPTIONAL,
    scfID [3] ScfID {bound} OPTIONAL,
    extensions [4] Extensions {bound} OPTIONAL,
    carrier [5] Carrier {bound} OPTIONAL,
    serviceInteractionIndicatorsTwo [6] ServiceInteractionIndicatorsTwo OPTIONAL,
    callSegmentID [7] CallSegmentID {bound} OPTIONAL,
    naOliInfo [50] NAOliInfo OPTIONAL,
    chargeNumber [51] ChargeNumber {bound} OPTIONAL,
    ...
}

eventNotificationCharging {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT EventNotificationChargingArg { bound}
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE opcode-eventNotificationCharging}
-- Direction: gsmSSF -> gsmSCF, Timer: Tenc
-- This operation is used by the gsmSSF to report to the gsmSCF the occurrence of a
-- specific charging event type as previously requested by the gsmSCF in a
-- RequestNotificationChargingEvent operation.

EventNotificationChargingArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    EventSpecificChargingInformation [1] EventSpecificChargingInformation{ bound},
    legID [2] LegID OPTIONAL,
    extensions [3] Extensions {bound} OPTIONAL,
    ...
}

eventReportBCSM {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT EventReportBCSMArg {bound}
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE opcode-eventReportBCSM}
-- Direction: gsmSSF -> gsmSCF, Timer: Terb
-- This operation is used to notify the gsmSCF of a call-related event (e.g. BCSM
-- events such as O_Busy or O_No_Answer) previously requested by the gsmSCF in a
-- RequestReportBCSMEvent operation.

EventReportBCSMArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    eventTypeBCSM [0] EventTypeBCSM,
    eventSpecificInformationBCSM [2] EventSpecificInformationBCSM {bound} OPTIONAL,
    legID [3] ReceivingSideID OPTIONAL,
    miscCallInfo [4] MiscCallInfo DEFAULT {messageType request},
    extensions [5] Extensions OPTIONAL,
    ...
}

furnishChargingInformation {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT FurnishChargingInformationArg {bound}
    RETURN RESULT FALSE
    ERRORS {missingParameter |
            taskRefused |
            unexpectedComponentSequence |
            unexpectedDataValue |
            unexpectedParameter |
            unknownLegId}
    CODE opcode-furnishChargingInformation}
-- Direction: gsmSCF -> gsmSSF, Timer: T_fci
-- This operation is used to request the gsmSSF to generate, register a call record
-- or to include some information in the default call record.
-- The registered call record is intended for off line charging of the call.

FurnishChargingInformationArg {PARAMETERS-BOUND : bound} ::=
    FCIBillingChargingCharacteristics{bound}

initialDP {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT InitialDPArg {bound}
    RETURN RESULT FALSE
    ERRORS {missingCustomerRecord |
            missingParameter |
            parameterOutOfRange |
            systemFailure |
            taskRefused |
            unexpectedComponentSequence |
            unexpectedDataValue |
            unexpectedParameter}

```

```

CODE          opcode-initialDP}
-- Direction: gsmSSF -> gsmSCF, Timer: Tidp
-- This operation is used after a TDP to indicate request for service.

InitialDPArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  serviceKey          [0] ServiceKey ,
  calledPartyNumber  [2] CalledPartyNumber {bound}          OPTIONAL,
  callingPartyNumber [3] CallingPartyNumber {bound}         OPTIONAL,
  callingPartysCategory [5] CallingPartysCategory           OPTIONAL,
  cGEncountered      [7] CGEncountered                      OPTIONAL,
  ipSSPCapabilities [8] IPSSPCapabilities {bound}          OPTIONAL,
  locationNumber     [10] LocationNumber {bound}            OPTIONAL,
  originalCalledPartyID [12] OriginalCalledPartyID {bound}  OPTIONAL,
  extensions         [15] Extensions                       OPTIONAL,
  highLayerCompatibility [23] HighLayerCompatibility        OPTIONAL,
  additionalCallingPartyNumber [25] AdditionalCallingPartyNumber {bound} OPTIONAL,
  bearerCapability   [27] BearerCapability {bound}          OPTIONAL,
  eventTypeBCSM      [28] EventTypeBCSM                    OPTIONAL,
  redirectingPartyID [29] RedirectingPartyID {bound}        OPTIONAL,
  redirectionInformation [30] RedirectionInformation         OPTIONAL,
  cause              [17] Cause {bound}                    OPTIONAL,
  serviceInteractionIndicatorsTwo [32] ServiceInteractionIndicatorsTwo OPTIONAL,
  carrier            [37] Carrier {bound}                  OPTIONAL,
  cug-Index         [45] CUG-Index                         OPTIONAL,
  cug-Interlock     [46] CUG-Interlock                     OPTIONAL,
  cug-OutgoingAccess [47] NULL                             OPTIONAL,
  iMSI              [50] IMSI                              OPTIONAL,
  subscriberState   [51] SubscriberState                   OPTIONAL,
  locationInformation [52] LocationInformation              OPTIONAL,
  ext-basicServiceCode [53] Ext-BasicServiceCode           OPTIONAL,
  callReferenceNumber [54] CallReferenceNumber              OPTIONAL,
  mscAddress        [55] ISDN-AddressString                 OPTIONAL,
  calledPartyBCDNumber [56] CalledPartyBCDNumber {bound}   OPTIONAL,
  timeAndTimezone   [57] TimeAndTimezone {bound}          OPTIONAL,
  callForwardingSS-Pending [58] NULL                       OPTIONAL,
  initialDPArgExtension [59] InitialDPArgExtension          OPTIONAL,
  ...
}

InitialDPArgExtension ::= SEQUENCE {
  gsmcAddress        [0] ISDN-AddressString                 OPTIONAL,
  forwardingDestinationNumber [1] CalledPartyNumber {bound} OPTIONAL,
  ms-Classmark2     [2] MS-Classmark2                      OPTIONAL,
  iMEI              [3] IMEI                              OPTIONAL,
  supportedCamelPhases [4] SupportedCamelPhases            OPTIONAL,
  supportedCamel4Subsets [5] SupportedCamel4Subsets        OPTIONAL,
  ...
}
-- If ipSSPCapabilities is not present then this denotes that a colocated gsmSRF is not
-- supported by the gsmSSF. If present, then the gsmSSF supports a colocated gsmSRF capable
-- of playing announcements via elementaryMessageIDs and variableMessages, the playing of
-- tones and the collection of DTMF digits. Other supported capabilities are explicitly
-- detailed in the IPSSPCapabilities parameter itself.
-- Carrier is included at the discretion of the gsmSSF operator.

InitiateCallAttempt {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      InitiateCallAttemptArg {bound}
  RESULT        InitiateCallAttemptRes {bound}
  ERRORS        {missingParameter |
                 parameterOutOfRange |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter |
                 unknownCSId}
  CODE          opcode-initiateCallAttempt}
-- Direction: gsmSCF -> gsmSSF, Timer Tica
-- This operation is used to instruct the gsmSSF to create a new call to a call party using the
-- address information provided by the gsmSCF.

InitiateCallAttemptArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  DestinationRoutingAddress [0] DestinationRoutingAddress {bound},
  extensions                 [4] Extensions {bound}          OPTIONAL,
  legToBeCreated            [5] LegID                         OPTIONAL,
  newCallSegment            [6] CallSegmentID {bound}        OPTIONAL,
  callingPartyNumber        [30] CallingPartyNumber {bound}  OPTIONAL,
  callReferenceNumber        [51] CallReferenceNumber         OPTIONAL,

```

```

    gsmSCFAddress          [52] ISDN-AddressString          OPTIONAL,
    suppress-T-CSI         [53] NULL                      OPTIONAL,
    ...
}

InitiateCallAttemptRes {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    supportedCamelPhases   [0] SupportedCamelPhases       OPTIONAL,
    supportedCamel4Subsets [1] SupportedCamel4Subsets     OPTIONAL,
    extensions             [2] Extensions {bound}        OPTIONAL,
    ...
}

moveLeg {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      MoveLegArg {bound}
    RETURN RESULT TRUE
    ERRORS        {missingParameter |
                  systemFailure |
                  taskRefused |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |
                  unknownLegID}
    CODE          opcode-moveLeg}
-- Direction: gsmSCF -> gsmSSF, Timer: Tm1
-- This operation is used by the gsmSCF to move a leg from one call segment to another call segment
-- within the same call segment association.

MoveLegArg {PARAMETERS-BOUND : bound} ::= SEQUENCE{
    legIDToMove          [0] LegID,
    extensions           [2] Extensions {bound}          OPTIONAL,
    ...
}

playTone {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      playToneArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  parameterOutOfRange |
                  systemFailure |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |
                  unknownLegID |
                  unknownCSID}
    CODE          opcode-playTone}
-- Direction: gsmSCF -> gsmSSF, Timer: Tpt
-- This operation is used to play tones to either a leg or a call segment using
-- the MSC's tone generator.

PlayToneArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    LegOrCallSegment     [0] LegOrCallSegment {bound},
    bursts               [1] Burst,
    extensions           [2] Extensions                 OPTIONAL,
    ...
}

releaseCall {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      ReleaseCallArg {bound}
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE          opcode-releaseCall}
-- Direction: gsmSCF -> gsmSSF, Timer: Trc
-- This operation is used to tear down an existing call at any phase of the call for all parties
-- involved in the call.

ReleaseCallArg {PARAMETERS-BOUND : bound} ::= Cause {bound}
-- A default value of decimal 31 (normal unspecified) shall be given.

requestNotificationChargingEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      RequestNotificationChargingEventArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  parameterOutOfRange |
                  systemFailure |
                  taskRefused |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |

```

```

        unknownLegID}
    CODE          opcode-requestNotificationChargingEvent}
-- Direction: gsmSCF -> gsmSSF, Timer: Trnc
-- This operation is used by the gsmSCF to instruct the gsmSSF on how to manage the charging events
-- which are received from other FEs and not under control of the service logic instance.

RequestNotificationChargingEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    EventTypeChargingPLMN          [0] EventTypeChargingPLMN {bound},
    monitorMode                    [1] MonitorMode,
    legID                           [2] LegID                                OPTIONAL,
    extensions                      [3] Extensions {bound}                OPTIONAL,
    ...
}
-- This argument indicates the charging event type and the corresponding monitor mode and LegID.

requestReportBCSMEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      RequestReportBCSMEventArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  parameterOutOfRange |
                  systemFailure |
                  taskRefused |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |
                  unknownLegID}
    CODE          opcode-requestReportBCSMEvent}
-- Direction: gsmSCF -> gsmSSF, Timer: Trrb
-- This operation is used to request the gsmSSF to monitor for a call-related event
-- (e.g. BCSM events such as O_Busy or O_No_Answer) and to send a notification
-- to the gsmSCF when the event is detected.
--
-- NOTE:
-- Every EDP must be explicitly armed by the gsmSCF via a RequestReportBCSMEvent operation.
-- No implicit arming of EDPs at the gsmSSF after reception of any operation (different
-- from RequestReportBCSMEvent) from the gsmSCF is allowed.

RequestReportBCSMEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    bcsmevents          [0] SEQUENCE SIZE(1..bound.&numOfBCSMEvents) OF
                        BCSMEvent {bound},
    Extensions          [2] Extensions {bound}                                OPTIONAL,
    ...
}
-- Indicates the BCSM related events for notification.

resetTimer {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      ResetTimerArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  parameterOutOfRange |
                  taskRefused |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |
                  unknownCSID}
    CODE          opcode-resetTimer}
-- Direction: gsmSCF -> gsmSSF, Timer: Trt
-- This operation is used to request the gsmSSF to refresh an application timer in the gsmSSF.

ResetTimerArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    timerID          [0] TimerID DEFAULT tssf,
    timervalue       [1] TimerValue,
    extensions       [2] Extensions {bound}                                OPTIONAL,
    callSegmentID    [3] CallSegmentID {bound}                            OPTIONAL,
    ...
}

sendChargingInformation {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      SendChargingInformationArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  unexpectedComponentSequence |
                  unexpectedParameter |
                  parameterOutOfRange |
                  systemFailure |
                  taskRefused |
                  unexpectedDataValue |
                  unknownLegID}
}

```

```

CODE          opcode-sendChargingInformation}
-- Direction: gsmSCF -> gsmSSF, Timer: Tsci
-- This operation is used to instruct the gsmSSF on the charging information to send by the gsmSSF.
-- The charging information can either be sent back by means of signalling or internal
-- if the gsmSSF is located in the local exchange. In the local exchange
-- this information may be used to update the charge meter or to create a standard call record.

SendChargingInformationArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    sCIBillingChargingCharacteristics [0] SCIBillingChargingCharacteristics {bound},
    partyToCharge                      [1] SendingSideID,
    extensions                          [2] Extensions {bound} OPTIONAL,
    ...
}

splitLeg {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          SplitLegArg {bound}
    RETURN RESULT    TRUE
    ERRORS            {missingParameter |
                    unexpectedComponentSequence |
                    unexpectedParameter |
                    unexpectedDataValue |
                    systemFailure |
                    taskRefused |
                    unknownLegID}
    CODE              opcode-splitLeg}
-- Direction: gsmSCF -> gsmSSF, Timer Ts1
-- This operation is used by the gsmSCF to separate one joined leg from a multi-way connection or
-- a single two party Call Segment.

SplitLegArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    legToBeSplit      [0] LegID,
    newCallSegment    [1] CallSegmentID {bound} OPTIONAL,
    extensions         [2] Extensions {bound} OPTIONAL,
    ...
}

END

```

The following value ranges apply for operation specific timers in CAP:

short:	1 s - 10 s
medium:	1 s - 60 s
long:	1 s - 30 minutes

Table 6-1 lists all operation timers and the value range for each timer. The definitive value for each operation timer may be network specific and has to be defined by the network operator.

Table 6-1: Timer value ranges

Operation Name	Timer	value range
ActivityTest	T _{at}	Short
ApplyCharging	T _{ac}	Short
ApplyChargingReport	T _{acr}	Short
AssistRequestInstructions	T _{ari}	Short
CallInformationReport	T _{cirp}	Short
CallInformationRequest	T _{cirq}	Short
Cancel	T _{can}	Short
CallGap	T _{cg}	Short
Connect	T _{con}	Short
ConnectToResource	T _{ctr}	Short
Continue	T _{cue}	Short
ContinueWithArgument	T _{cwa}	Short
DisconnectForwardConnectionWithArgument	T _{dfcwa}	Short
DisconnectLeg	T _{dl}	Short
EntityReleased	T _{er}	Short
DisconnectForwardConnection	T _{dfc}	Short
EstablishTemporaryConnection	T _{etc}	Medium
EventNotificationCharging	T _{enc}	Short
EventReportBCSM	T _{erb}	Short
FurnishChargingInformation	T _{fci}	Short
InitialDP	T _{idp}	Short
InitiateCallAttempt	T _{ica}	Short
MoveLeg	T _{ml}	Short
PlayTone	T _{pt}	Short
ReleaseCall	T _{rc}	Short
RequestNotificationChargingEvent	T _{rnc}	Short
RequestReportBCSMEvent	T _{rrb}	Short
ResetTimer	T _{rt}	Short
SendChargingInformation	T _{sci}	Short
SplitLeg	T _{sl}	Short

6.1.2 gsmSSF/gsmSCF packages, contracts and ACs

6.1.2.1 gsmSSF/gsmSCF ASN.1 module

```
CAP-gsmSSF-gsmSCF-pkgs-contracts-acs {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-gsmSSF-gsmSCF-pkgs-contracts-acs(102) version4(3)}
```

```
DEFINITIONS ::= BEGIN
```

```
-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the gsmSSF - gsmSCF interface, for the control of
-- circuit switched calls.
```

```
-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.
```

```
IMPORTS
```

```
PARAMETERS-BOUND,
cAPSpecificBoundSet
FROM CAP-classes classes
```

```
CONTRACT,
```

```

OPERATION-PACKAGE,
OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

TCMessage {}
FROM TCAPMessages tc-Messages

APPLICATION-CONTEXT,
dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions

activityTest,
applyCharging {},
applyChargingReport {},
assistRequestInstructions {},
callGap {},
callInformationReport {},
callInformationRequest {},
cancel,
connect {},
connectToResource {},
continue,
continueWithArgument {},
disconnectForwardConnection,
disconnectForwardConnectionWithArgument {},
disconnectLeg {},
entity Released {},
establishTemporaryConnection {},
eventNotificationCharging {},
eventReportBCSM {},
furnishChargingInformation {},
initialDP {},
initiateCallAttempt {},
moveLeg {},
playTone {}
releaseCall {},
requestNotificationChargingEvent {},
requestReportBCSMEvent {},
resetTimer {},
sendChargingInformation {},
splitLeg {}
FROM CAP-gsmSSF-gsmSCF-ops-args gsmSSF-gsmSCF-Operations

playAnnouncement {},
promptAndCollectUserInformation {},
specializedResourceReport
FROM CAP-gsmSCF-gsmSRF-ops-args gsmSCF-gsmSRF-Operations

specializedResourceControlPackage {}
FROM CAP-gsmSCF-gsmSRF-pkgs-contracts-acs gsmSCF-gsmSRF-Protocol

id-ac-CAP-gsmSSF-scfGenericAC,
id-ac-CAP-gsmSSF-scfAssistHandoffAC,
id-ac-CAP-scf-gsmSSFGenericAC,
id-CAPSsfToScfGeneric,
id-CAPAssistHandoffssfToScf,
id-as-gsmSSF-scfGenericAS,
id-as-assistHandoff-gsmSSF-scfAS,
id-package-scfActivation,
id-package-gsmSRF-scfActivationOfAssist,
id-package-assistConnectionEstablishment,
id-package-genericDisconnectResource,
id-package-nonAssistedConnectionEstablishment,
id-package-connect,
id-package-callHandling,
id-package-bcsmEventHandling,
id-package-chargingEventHandling,
id-package-ssfCallProcessing,
id-package-scfCallInitiation,
id-package-timer,
id-package-billing,
id-package-charging,
id-package-trafficManagement,
id-package-callReport,
id-package-signallingControl,
id-package-activityTest,
id-package-cancel,
id-package-cphResponse,

```



```

    id-package-exceptionInform,
    classes,
    ros-InformationObjects,
    tc-Messages,
    tc-NotationExtensions,
    gsmSSF-gsmSCF-Operations,
    gsmSCF-gsmSRF-Operations,
    gsmSCF-gsmSRF-Protocol
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

;

-- Application Contexts

capssf-scfGenericAC APPLICATION-CONTEXT ::= {
    CONTRACT                capSsfToScfGeneric
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             gsmSSF-scfGenericAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gsmSSF-scfGenericAC}

capssf-scfAssistHandoffAC APPLICATION-CONTEXT ::= {
    CONTRACT                capAssistHandoffssfToScf
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             assistHandoff-gsmSSF-scfAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gsmSSF-scfAssistHandoffAC}

capscf-ssfGenericAC APPLICATION-CONTEXT ::= {
    CONTRACT                capScfToSsfGeneric
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             scf-gsmSSFGenericAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-scf-gsmSSFGenericAC}

-- Contracts

capSsfToScfGeneric CONTRACT ::= {
-- dialogue initiated by gsmSSF with InitialDP Operation
    INITIATOR CONSUMER OF  {exceptionInformPackage {cAPSpecificBoundSet} |
                             scfActivationPackage {cAPSpecificBoundSet}}
    RESPONDER CONSUMER OF  {activityTestPackage |
                             assistConnectionEstablishmentPackage {cAPSpecificBoundSet} |
                             bcsmeventHandlingPackage {cAPSpecificBoundSet} |
                             billingPackage {cAPSpecificBoundSet} |
                             callHandlingPackage {cAPSpecificBoundSet} |
                             callReportPackage {cAPSpecificBoundSet} |
                             cancelPackage |
                             chargingEventHandlingPackage {cAPSpecificBoundSet} |
                             chargingPackage {cAPSpecificBoundSet} |
                             connectPackage {cAPSpecificBoundSet} |
                             cphResponsePackage {cAPSpecificBoundSet} |
                             genericDisconnectResourcePackage {cAPSpecificBoundSet} |
                             nonAssistedConnectionEstablishmentPackage {cAPSpecificBoundSet} |
                             signallingControlPackage {cAPSpecificBoundSet} |
                             specializedResourceControlPackage {cAPSpecificBoundSet} |
                             ssfCallProcessingPackage {cAPSpecificBoundSet} |
                             timerPackage {cAPSpecificBoundSet} |
                             trafficManagementPackage {cAPSpecificBoundSet}
                             scfCallInitiationPackage {cAPSpecificBoundSet}}
    ID                      id-CAPSsfToScfGeneric}

capAssistHandoffssfToScf CONTRACT ::= {
-- dialogue initiated by gsmSSF with AssistRequestInstructions
    INITIATOR CONSUMER OF  {gsmSRF-scfActivationOfAssistPackage {cAPSpecificBoundSet}}
    RESPONDER CONSUMER OF  {activityTestPackage |
                             callHandlingPackage {cAPSpecificBoundSet} |
                             cancelPackage |
                             genericDisconnectResourcePackage {cAPSpecificBoundSet} |
                             nonAssistedConnectionEstablishmentPackage {cAPSpecificBoundSet} |
                             specializedResourceControlPackage {cAPSpecificBoundSet} |
                             timerPackage {cAPSpecificBoundSet}}
    ID                      id-CAPAssistHandoffssfToScf}

capScfToSsfGeneric CONTRACT ::= {
-- dialogue initiated by gsmSCF with InitiateCallAttempt, Generic Case

```

```

INITIATOR CONSUMER OF      {activityTestPackage |
                           assistConnectionEstablishmentPackage {cAPSpecificBoundSet} |
                           bcsmEventHandlingPackage {cAPSpecificBoundSet} |
                           billingPackage {cAPSpecificBoundSet} |
                           callHandlingPackage {cAPSpecificBoundSet} |
                           callReportPackage {cAPSpecificBoundSet} |
                           cancelPackage |
                           chargingPackage {cAPSpecificBoundSet} |
                           connectPackage {cAPSpecificBoundSet} |
                           cphResponsePackage {cAPSpecificBoundSet} |
                           genericDisconnectResourcePackage {cAPSpecificBoundSet} |
                           nonAssistedConnectionEstablishmentPackage {cAPSpecificBoundSet} |
                           scfCallInitiationPackage {cAPSpecificBoundSet} |
                           signallingControlPackage {cAPSpecificBoundSet} |
                           specializedResourceControlPackage {cAPSpecificBoundSet} |
                           ssfCallProcessingPackage {cAPSpecificBoundSet} |
                           timerPackage {cAPSpecificBoundSet}}
RESPONDER CONSUMER OF     {exceptionInformPackage {cAPSpecificBoundSet}}
ID                         id-capScfToSsfGeneric}

```

-- Operation Packages

```

chargingEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {requestNotificationChargingEvent {bound}}
  SUPPLIER INVOKES  {eventNotificationCharging {bound}}
  ID                id-package-chargingEventHandling}

scfActivationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {initialDP {bound}}
  ID                id-package-scfActivation}

gsmSRF-scfActivationOfAssistPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {assistRequestInstructions {bound}}
  ID                id-package-gsmSRF-scfActivationOfAssist}

assistConnectionEstablishmentPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {establishTemporaryConnection {bound}}
  ID                id-package-assistConnectionEstablishment}

genericDisconnectResourcePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {disconnectForwardConnection |
                    disconnectForwardConnectionWithArgument {bound}}
  ID                id-package-genericDisconnectResource}

nonAssistedConnectionEstablishmentPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {connectToResource {bound}}
  ID                id-package-nonAssistedConnectionEstablishment}

connectPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {connect {bound}}
  ID                id-package-connect}

callHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {releaseCall {bound}}
  ID                id-package-callHandling}

bcsmEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {requestReportBCSMEvent {bound}}
  SUPPLIER INVOKES  {eventReportBCSM {bound}}
  ID                id-package-bcsmEventHandling}

ssfCallProcessingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {continueWithArgument {bound} | continue}
  ID                id-package-ssfCallProcessing}

scfCallInitiatePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {initiateCallAttempt {bound}}
  ID                id-package-scfCallInitiation}

timerPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {resetTimer {bound}}
  ID                id-package-timer}

billingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES  {furnishChargingInformation {bound}}
  ID                id-package-billing}

```

```

chargingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {applyCharging {bound}}
  SUPPLIER INVOKES    {applyChargingReport {bound}}
  ID                  id-package-charging}

trafficManagementPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {callGap {bound}}
  ID                  id-package-trafficManagement}

callReportPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {callInformationRequest {bound}}
  SUPPLIER INVOKES    {callInformationReport {bound}}
  ID                  id-package-callReport}

signallingControlPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {sendChargingInformation {bound}}
  ID                  id-package-signallingControl}

activityTestPackage OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {activityTest}
  ID                  id-package-activityTest}

cancelPackage OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {cancel}
  ID                  id-package-cancel}

cphResponsePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {continueWithArgument {bound} |
                      disconnectLeg {bound} |
                      moveLeg {bound} |
                      splitLeg {bound}}
  ID                  id-package-cphResponse}

exceptionInformPackage OPERATION-PACKAGE ::= {
  CONSUMER INVOKES    {entityReleased}
  ID                  id-package-exceptionInfom}

-- Abstract Syntaxes

gsmSSF-scfGenericAbstractSyntax ABSTRACT-SYNTAX ::= {
  GenericSSF-gsmSCF-PDUs
  IDENTIFIED BY      id-as-gsmSSF-scfGenericAS}

GenericSSF-gsmSCF-PDUs ::= TCMessgae {{SsfToScfGenericInvokable},
                                       {SsfToScfGenericReturnable}}

SsfToScfGenericInvokable OPERATION ::= {
  activityTest |
  applyCharging {cAPSpecificBoundSet} |
  applyChargingReport {cAPSpecificBoundSet} |
  callInformationReport {cAPSpecificBoundSet} |
  callInformationRequest {cAPSpecificBoundSet} |
  cancel |
  connect {cAPSpecificBoundSet} |
  continueWithArgument {cAPSpecificBoundSet} |
  connectToResource {cAPSpecificBoundSet} |
  disconnectForwardConnection |
  disconnectForwardConnectionWithArgument {cAPSpecificBoundSet} |
  disconnectLeg {cAPSpecificBoundSet} |
  entityReleased {cAPSpecificBoundSet} |
  establishTemporaryConnection {cAPSpecificBoundSet} |
  eventNotificationCharging {cAPSpecificBoundSet} |
  eventReportBCSM {cAPSpecificBoundSet} |
  furnishChargingInformation {cAPSpecificBoundSet} |
  initialDP {cAPSpecificBoundSet} |
  initiateCallAttempt {cAPSpecificBoundSet} |
  moveLeg {cAPSpecificBoundSet} |
  releaseCall {cAPSpecificBoundSet} |
  requestNotificationChargingEvent {cAPSpecificBoundSet} |
  requestReportBCSMEvent {cAPSpecificBoundSet} |
  resetTimer {cAPSpecificBoundSet} |
  sendChargingInformation {cAPSpecificBoundSet} |
  splitLeg {cAPSpecificBoundSet} |
  playAnnouncement {cAPSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSpecificBoundSet} |
  specializedResourceReport
}

```

```

SsfToSsfGenericReturnable OPERATION ::= {
  activityTest |
  applyCharging {cAPSSpecificBoundSet} |
  applyChargingReport {cAPSSpecificBoundSet} |
  callGap {cAPSSpecificBoundSet} |
  callInformationRequest {cAPSSpecificBoundSet} |
  cancel |
  connect {cAPSSpecificBoundSet} |
  connectToResource {cAPSSpecificBoundSet} |
  continue |
  continueWithArgument {cAPSSpecificBoundSet} |
  disconnectForwardConnection |
  disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
  disconnectLeg {cAPSSpecificBoundSet} |
  entityReleased {cAPSSpecificBoundSet} |
  establishTemporaryConnection {cAPSSpecificBoundSet} |
  furnishChargingInformation {cAPSSpecificBoundSet} |
  initialDP {cAPSSpecificBoundSet} |
  initiateCallAttempt {cAPSSpecificBoundSet} |
  moveLeg {cAPSSpecificBoundSet} |
  releaseCall {cAPSSpecificBoundSet} |
  requestReportBCSMEEvent {cAPSSpecificBoundSet} |
  resetTimer {cAPSSpecificBoundSet} |
  sendChargingInformation {cAPSSpecificBoundSet} |
  splitLeg {cAPSSpecificBoundSet} |
  playAnnouncement {cAPSSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSSpecificBoundSet}
}

assistHandoff-gsmSSF-scfAbstractSyntax ABSTRACT-SYNTAX ::= {
  AssistHandoffsSF-gsmSCF-PDUs
  IDENTIFIED BY id-as-assistHandoff-gsmSSF-scfAS}

AssistHandoffsSF-gsmSCF-PDUs ::= TCMMessage {{AssistHandoffssfToSsfInvokable},
  {AssistHandoffssfToSsfReturnable}}

AssistHandoffssfToSsfInvokable OPERATION ::= {
  activityTest |
  assistRequestInstructions {cAPSSpecificBoundSet} |
  cancel |
  connectToResource {cAPSSpecificBoundSet} |
  disconnectForwardConnection |
  disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
  playAnnouncement {cAPSSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSSpecificBoundSet} |
  resetTimer {cAPSSpecificBoundSet} |
  specializedResourceReport
}

AssistHandoffssfToSsfReturnable OPERATION ::= {
  activityTest |
  assistRequestInstructions {cAPSSpecificBoundSet} |
  cancel |
  connectToResource {cAPSSpecificBoundSet} |
  disconnectForwardConnection |
  disconnectForwardConnectionWithArgument {cAPSSpecificBoundSet} |
  playAnnouncement {cAPSSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSSpecificBoundSet} |
  resetTimer {cAPSSpecificBoundSet}
}

scf-gsmSSFGenericAbstractSyntax ABSTRACT-SYNTAX ::= {
  GenericSCF-gsmSSF-PDUs
  IDENTIFIED BY id-as-scf-gsmSSFGenericAS}

GenericSCF-gsmSSF-PDUs ::= TCMMessage {{ScfToSsfGenericInvokable}, {ScfToSsfGenericReturnable}}

ScfToSsfGenericInvokable OPERATION ::= {
  activityTest |
  applyCharging {cAPSSpecificBoundSet} |
  applyChargingReport {cAPSSpecificBoundSet} |
  callInformationRequest {cAPSSpecificBoundSet} |
  cancel |
  connect {cAPSSpecificBoundSet} |
  connectToResource {cAPSSpecificBoundSet} |
  continue |
  continueWithArgument {cAPSSpecificBoundSet} |

```

```

disconnectForwardConnection {cAPSpecificBoundSet} |
disconnectForwardConnectionWithArgument {cAPSpecificBoundSet} |
disconnectLeg {cAPSpecificBoundSet} |
establishTemporaryConnection {cAPSpecificBoundSet} |
furnishChargingInformation {cAPSpecificBoundSet} |
initiateCallAttempt {cAPSpecificBoundSet} |
moveLeg {cAPSpecificBoundSet} |
playTone {cAPSpecificBoundSet} |
releaseCall {cAPSpecificBoundSet} |
requestReportBCSMEEvent {cAPSpecificBoundSet} |
resetTimer {cAPSpecificBoundSet} |
sendChargingInformation {cAPSpecificBoundSet} |
splitLeg {cAPSpecificBoundSet} |
playAnnouncement {cAPSpecificBoundSet} |
promptAndCollectUserInformation {cAPSpecificBoundSet}
}

```

```

ScfToSsfGenericReturnable OPERATION ::= {
  activityTest |
  applyCharging {cAPSpecificBoundSet} |
  applyChargingReport {cAPSpecificBoundSet} |
  callInformationReport {cAPSpecificBoundSet} |
  callInformationRequest {cAPSpecificBoundSet} |
  cancel |
  connect {cAPSpecificBoundSet} |
  connectToResource {cAPSpecificBoundSet} |
  disconnectForwardConnection |
  disconnectForwardConnectionWithArgument {cAPSpecificBoundSet} |
  disconnectLeg {cAPSpecificBoundSet} |
  entityReleased {cAPSpecificBoundSet} |
  establishTemporaryConnection {cAPSpecificBoundSet} |
  eventReportBCSM {cAPSpecificBoundSet} |
  furnishChargingInformation {cAPSpecificBoundSet} |
  initiateCallAttempt {cAPSpecificBoundSet} |
  moveLeg {cAPSpecificBoundSet} |
  requestReportBCSMEEvent {cAPSpecificBoundSet} |
  resetTimer {cAPSpecificBoundSet} |
  sendChargingInformation {cAPSpecificBoundSet} |
  splitLeg {cAPSpecificBoundSet} |
  playAnnouncement {cAPSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSpecificBoundSet} |
  specializedResourceReport
}

```

END

6.2 gsmSCF/gsmSRF interface

6.2.1 gsmSCF/gsmSRF operations and arguments

```

CAP-gsmSCF-gsmSRF-ops-args {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-gsmSCF-gsmSRF-ops-args(103) version4(3)}

```

```

DEFINITIONS IMPLICIT TAGS ::= BEGIN

```

```

-- This module contains the operations and operation arguments used for the
-- gsmSRF - gsmSCF interface, for the control of circuit switched calls.

```

```

-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.

```

```

IMPORTS

```

```

  OPERATION

```

```

FROM Remote-Operations-Information-Objects ros-InformationObjects

```

```

  opcode-playAnnouncement,
  opcode-promptAndCollectUserInformation,
  opcode-specializedResourceReport

```

```

FROM CAP-operationcodes operationcodes

```

```

  CallSegmentID {}

```

```

FROM CS2-datatypes {itu-t(0) identified-organisation(4) etsi(0) inDomain(1) in-network(1) cs2(20)
modules(0) in-cs2-datatypes(0) version1(0)}

```

```

    CollectedInfo,
    Digits {},
    Extensions {},
    InformationToSend {}
FROM CAP-datatypes datatypes

    canceled,
    improperCallerResponse,
    missingParameter,
    parameterOutOfRange,
    systemFailure,
    taskRefused,
    unavailableResource,
    unexpectedComponentSequence,
    unexpectedDataValue,
    unexpectedParameter,
    unknownCSID
FROM CAP-erroratypes erroratypes

    PARAMETERS-BOUND
FROM CAP-classes classes

    ros-InformationObjects,
    operationcodes,
    datatypes,
    erroratypes,
    classes
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

;

playAnnouncement {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      PlayAnnouncementArg {bound}
    RETURN RESULT  FALSE
    ERRORS        {canceled |
                  missingParameter |
                  parameterOutOfRange |
                  systemFailure |
                  taskRefused |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter |
                  unavailableResource |
                  unknownCSID}
    LINKED        {specializedResourceReport}
    CODE          opcode-playAnnouncement}
-- Direction: gsmSCF -> gsmSRF, Timer: Tpa
-- This operation is to be used after Establish Temporary Connection (assist procedure
-- with a second gsmSSF) or a Connect to Resource (no assist) operation. It may be used
-- for inband interaction with a mobile station, or for interaction with an ISDN user.
-- In the former case, the gsmSRF is usually collocated with the gsmSSF for standard
-- tones (congestion tone...) or standard announcements.
-- In the latter case, the gsmSRF is always collocated with the gsmSSF in the switch.
-- Any error is returned to the gsmSCF. The timer associated with this operation must
-- be of a sufficient duration to allow its linked operation to be correctly correlated.

PlayAnnouncementArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    informationToSend          [0] InformationToSend {bound},
    disconnectFromIPForbidden [1] BOOLEAN DEFAULT TRUE,
    requestAnnouncementCompleteNotification [2] BOOLEAN DEFAULT TRUE,
    extensions                 [3] Extensions {bound} OPTIONAL,
    callSegmentID              [5] CallSegmentID {bound} OPTIONAL,
    requestAnnouncementStartedNotification [51] BOOLEAN DEFAULT FALSE,
    ...
}

promptAndCollectUserInformation {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      PromptAndCollectUserInformationArg {bound}
    RESULT        ReceivedInformationArg {bound}
    ERRORS        {canceled |
                  improperCallerResponse |
                  missingParameter |
                  parameterOutOfRange |
                  systemFailure |
                  taskRefused |
                  unexpectedComponentSequence |

```

```

        unavailableResource |
        unexpectedDataValue |
        unexpectedParameter |
        unknownCSID}
LINKED      {specializedResourceReport}
CODE       opcode-promptAndCollectUserInformation}
-- Direction: gsmSCF -> gsmSRF, Timer: Tpc
-- This operation is used to interact with a user to collect information.

PromptAndCollectUserInformationArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    collectedInfo                [0] CollectedInfo,
    disconnectFromIPForbidden    [1] BOOLEAN DEFAULT TRUE,
    informationToSend             [2] InformationToSend {bound}           OPTIONAL,
    extensions                    [3] Extensions {bound}                OPTIONAL,
    callSegmentID                [4] CallSegmentID {bound}              OPTIONAL,
    requestAnnouncementStartedNotification [51] BOOLEAN DEFAULT FALSE,
    ...
}

ReceivedInformationArg {PARAMETERS-BOUND : bound} ::= CHOICE {
    digitsResponse                [0] Digits {bound}
}

specializedResourceReport OPERATION ::= {
    ARGUMENT      SpecializedResourceReportArg
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE         opcode-specializedResourceReport}
-- Direction: gsmSRF -> gsmSCF, Timer: Tsrr
-- This operation is used as the response to a PlayAnnouncement operation when the announcement
-- completed report indication is set.

SpecializedResourceReportArg ::= CHOICE {
    allAnnouncementsComplete    [50] NULL,
    firstAnnouncementStarted    [51] NULL
}

END

```

The following value ranges apply for operation specific timers in CAP:

```

short:      1 s - 10 s
medium:    1 s - 60 s
long:      1 s - 30 minutes

```

Table 6-2 lists all operation timers and the value range for each timer. The definitive value for each operation timer may be network specific and has to be defined by the network operator.

Table 6-2: Operation timers and their value range

Operation Name	Timer	Value range
PlayAnnouncement	T _{pa}	Long
PromptAndCollectUserInformation	T _{pc}	Long
SpecializedResourceReport	T _{srr}	Short

6.2.2 gsmSRF/gsmSCF contracts, packages and ACs

6.2.2.1 gsmSRF/gsmSCF ASN.1 modules

```

CAP-gsmSCF-gsmSRF-pkgs-contracts-ac {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-gsmSCF-gsmSRF-pkgs-contracts-ac(104) version4(3)}

```

```

DEFINITIONS ::= BEGIN

```

```

-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the gsmSRF - gsmSCF interface, for the control of
-- circuit switched calls.

```

```

-- The table in subclause 2.1 lists the specifications that contain the modules

```

```

-- that are used by CAP.
IMPORTS

    PARAMETERS-BOUND,
    cAPSpecificBoundSet
FROM CAP-classes classes

    CONTRACT,
    OPERATION-PACKAGE,
    OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects

    TCMessages {}
FROM TCAPMessages tc-Messages

    APPLICATION-CONTEXT,
    dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions

    playAnnouncement {},
    promptAndCollectUserInformation {},
    specializedResourceReport
FROM CAP-gsmSCF-gsmSRF-ops-args gsmSCF-gsmSRF-Operations

    activityTest,
    cancel,
    assistRequestInstructions {}
FROM CAP-gsmSSF-gsmSCF-ops-args gsmSSF-gsmSCF-Operations

    gsmSRF-scfActivationOfAssistPackage {}
FROM CAP-gsmSSF-gsmSCF-pkgs-contracts-acg gsmSSF-gsmSCF-Protocol

    id-package-specializedResourceControl,
    id-package-activityTest,
    id-ac-gsmSRF-gsmSCF,
    id-contract-gsmSRF-gsmSCF,
    id-package-gsmSRF-scfCancel,
    id-as-basic-gsmSRF-gsmSCF,
    classes,
    ros-InformationObjects,
    tc-Messages,
    tc-NotationExtensions,
    gsmSCF-gsmSRF-Operations,
    gsmSSF-gsmSCF-Operations,
    gsmSSF-gsmSCF-Protocol
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}

;

-- Application Contexts

gsmSRF-gsmSCF-ac APPLICATION-CONTEXT ::= {
    CONTRACT                gsmSRF-gsmSCF-contract
    DIALOGUE MODE           structured
    TERMINATION              basic
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             gsmSRF-gsmSCF-abstract-syntax}
    APPLICATION CONTEXT NAME id-ac-gsmSRF-gsmSCF}

-- Contracts

gsmSRF-gsmSCF-contract CONTRACT ::= {
    INITIATOR CONSUMER OF   {gsmSRF-scfActivationOfAssistPackage {cAPSpecificBoundSet}}
    RESPONDER CONSUMER OF   {specializedResourceControlPackage {cAPSpecificBoundSet} |
                             activityTestPackage |
                             gsmSRF-scfCancelPackage }
    ID                      id-contract-gsmSRF-gsmSCF}

-- Operation Packages

specializedResourceControlPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES       {playAnnouncement {bound} |
                             promptAndCollectUserInformation {bound}}
    SUPPLIER INVOKES       {specializedResourceReport}
    ID                     id-package-specializedResourceControl}

gsmSRF-scfCancelPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES       {cancel }
    ID                     id-package-gsmSRF-scfCancel}

ActivityTestPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES       {activityTest}
    ID                     id-package-activityTest}

-- Abstract Syntaxes

```



```
gsmSRF-gsmSCF-abstract-syntax ABSTRACT-SYNTAX ::= {
  BASIC-gsmSRF-gsmSCF-PDUs
  IDENTIFIED BY id-as-basic-gsmSRF-gsmSCF}

BASIC-gsmSRF-gsmSCF-PDUs ::= TCMMessage {{GsmSRFgsmSCFInvokable},{GsmSRFgsmSCFReturnable}}

GsmSRFgsmSCFInvokable OPERATION ::= {
  activityTest |
  assistRequestInstructions {cAPSpecificBoundSet} |
  cancel |
  playAnnouncement {cAPSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSpecificBoundSet} |
  specializedResourceReport
}

GsmSRFgsmSCFReturnable OPERATION ::= {
  activityTest |
  assistRequestInstructions {cAPSpecificBoundSet} |
  cancel |
  playAnnouncement {cAPSpecificBoundSet} |
  promptAndCollectUserInformation {cAPSpecificBoundSet}
}

END
```

7 SMS Control

This clause defines the protocol used for CAMEL control of MO SMS and MT SMS. CAMEL control of MO SMS uses version 3 of the application context, and CAMEL control of MT SMS uses version 4 of the application context.

7.1 SMS operations and arguments

```
CAP-SMS-ops-args {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0) umts-network(1)
modules(3) cap-SMS-ops-args(105) version4(3)}
```

```
DEFINITIONS IMPLICIT TAGS ::= BEGIN
```

```
-- This module contains the operations and operation arguments used for the
-- smsSSF- gsmSCF interface, for the control of MO-SMS and MT-SMS.
```

```
-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.
```

```
IMPORTS
```

```
    errortypes,
    datatypes,
    operationcodes,
    classes,
    ros-InformationObjects,
    tc-Messages
```

```
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}
```

```
OPERATION
```

```
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
ServiceKey
```

```
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
modules(0) cs1-datatypes(2) version1(0)}
```

```
MiscCallInfo
```

```
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
cs2(20) modules(0) in-cs2-datatypes (0) version1(0)}
```

```
IMEI,
IMSI,
ISDN-AddressString
```

```
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}
```

```
GPRSMSClass,
LocationInformation,
MS-Classmark2
```

```
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}
```

```
PARAMETERS-BOUND
```

```
FROM CAP-classes classes
```

```
opcode-connectSMS,
opcode-continueSMS,
opcode-eventReportSMS,
opcode-furnishChargingInformationSMS,
opcode-initialDPSMS,
opcode-releaseSMS,
opcode-requestReportSMSEvent,
opcode-resetTimerSMS
```

```
FROM CAP-operationcodes operationcodes
```

```
CalledPartyBCDNumber {},
EventSpecificInformationSMS,
EventTypesSMS,
Extensions {},
FCISMSBillingChargingCharacteristics,
LocationInformationGPRS,
RPCause,
SMSEvent,
```

```

    TimeAndTimezone {},
    TimerID,
    TimerValue,
    TPDataCodingScheme,
    TPProtocolIdentifier,
    TPShortMessageSpecificInfo,
    TPValidityPeriod
FROM CAP-datatypes datatypes

    missingCustomerRecord,
    missingParameter,
    parameterOutOfRange,
    systemFailure,
    taskRefused,
    unexpectedComponentSequence,
    unexpectedDataValue,
    unexpectedParameter
FROM CAP-erroratypes erroratypes

    CallReferenceNumber,
FROM MAP-CH-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-network(1) modules(3) map-CH-DataTypes(13) version6(6)}

;

connectSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          ConnectSMSArg {bound}
    RETURN RESULT     FALSE
    ERRORS            {missingParameter |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unexpectedComponentSequence |
                      unexpectedDataValue |
                      unexpectedParameter}
    CODE              opcode-connectSMS}
-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Tconsms
-- This operation is used to request the smsSSF to perform the SMS processing
-- actions to route
-- or forward a short message to a specified destination.

ConnectSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    callingPartyNumber          [0] ISDN-AddressString          OPTIONAL,
    destinationSubscriberNumber [1] CalledPartyBCDNumber {bound} OPTIONAL,
    smscAddress                 [2] ISDN-AddressString          OPTIONAL,
    extensions                   [10] Extensions {bound}        OPTIONAL,
    ...
}

continueSMS OPERATION ::= {
    RETURN RESULT     FALSE
    ALWAYS RESPONDS  FALSE
    CODE              opcode-continueSMS}
-- Direction: gsmSCF -> smsSSF, Timer: Tcuesms
-- This operation is used to request the smsSSF to proceed with
-- Short Message processing at the DP at which it previously suspended
-- Short Message processing to await gsmSCF instructions (i.e. proceed
-- to the next Point in Association in the SMS FSM). The smsSSF
-- continues SMS processing without substituting new data from the gsmSCF.

eventReportSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT          EventReportSMSArg {bound}
    RETURN RESULT     FALSE
    ALWAYS RESPONDS  FALSE
    CODE              opcode-eventReportSMS}
-- Direction: gsmSSF or gprsSSF -> gsmSCF, Timer: Terbsms
-- This operation is used to notify the gsmSCF of a Short Message related event (FSM events
-- such as submission, delivery or failure) previously requested by the gsmSCF in a
-- RequestReportSMSEvent operation.

EventReportSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    eventTypeSMS          [0] EventTypeSMS,
    eventSpecificInformationSMS [1] EventSpecificInformationSMS    OPTIONAL,
    miscCallInfo          [2] MiscCallInfo DEFAULT {messageType request},
    extensions            [10] Extensions {bound}                  OPTIONAL,
    ...
}

```

```
furnishChargingInformationSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      FurnishChargingInformationSMSArg {bound}
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter}
  CODE          opcode-furnishChargingInformationSMS}
-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Tfcisms
-- This operation is used to request the smsSSF to generate, register a charging record
-- or to include some information in the default SM record. The registered charging record is
-- intended for off line charging of the Short Message.
```

```
FurnishChargingInformationSMSArg {PARAMETERS-BOUND : bound} ::=
  FCISMSBillingChargingCharacteristics {bound}
```

```
initialDPSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      InitialDPSMSArg {bound}
  RETURN RESULT FALSE
  ERRORS        {missingCustomerRecord |
                 missingParameter |
                 parameterOutOfRange |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter}
  CODE          opcode-initialDPSMS}
-- Direction: gsmSSF or gprsSSF -> gsmSCF, Timer: Tidpsms
-- This operation is used after a TDP to indicate request for service.
```

```
InitialDPSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  serviceKey                [0] ServiceKey,
  destinationSubscriberNumber [1] CalledPartyBCDNumber {bound} OPTIONAL,
  callingPartyNumber        [2] ISDN-AddressString OPTIONAL,
  eventTypeSMS              [3] EventTypeSMS OPTIONAL,
  IMSI                      [4] IMSI OPTIONAL,
  locationInformationMSC     [5] LocationInformation OPTIONAL,
  locationInformationGPRS    [6] LocationInformationGPRS OPTIONAL,
  smSCAddress               [7] ISDN-AddressString OPTIONAL,
  timeAndTimezone           [8] TimeAndTimezone {bound} OPTIONAL,
  tPShortMessageSpecificInfo [9] TPShortMessageSpecificInfo OPTIONAL,
  tPProtocolIdentifier       [10] TPProtocolIdentifier OPTIONAL,
  tPDataCodingScheme        [11] TPDataCodingScheme OPTIONAL,
  tPValidityPeriod          [12] TPValidityPeriod OPTIONAL,
  extensions                 [13] Extensions {bound} OPTIONAL,
  . . .
  smsReferenceNumber        [14] CallReferenceNumber OPTIONAL,
  mscAddress                [15] ISDN-AddressString OPTIONAL,
  sgsn-Number               [16] ISDN-AddressString OPTIONAL,
  ms-Classmark2             [17] MS-Classmark2 OPTIONAL,
  gPRSMSCClass              [18] GPRSMSCClass OPTIONAL,
  IMEI                      [19] IMEI OPTIONAL,
  calledPartyNumber         [17] ISDN-AddressString OPTIONAL
}
```

```
releaseSMS OPERATION ::= {
  ARGUMENT      ReleaseSMSArg
  RETURN RESULT FALSE
  ALWAYS RESPONDS FALSE
  CODE          opcode-releasesMS}
-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Trelsms
-- This operation is used to prevent an attempt to submit or deliver a short message.
```

```
ReleaseSMSArg ::= RPCause
```

```
requestReportSMSEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      RequestReportSMSEventArg {bound}
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 parameterOutOfRange |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
```

```

                unexpectedParameter}
CODE            opcode-requestReportSMSEvent}
-- Direction: gsmSCF -> gsmSSF or gprsSSF, Timer: Trrbsms
-- This operation is used to request the gsmSSF or gprsSSF to monitor for a
-- Short Message related event (FSM events such as submission, delivery or failure)
-- and to send a notification to the gsmSCF when the event is detected.

RequestReportSMSEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    sMSEvents          [0] SEQUENCE SIZE (1..bound.&numOfSMSEvents) OF SMSEvent,
    extensions         [10] Extensions {bound} OPTIONAL,
    ...
}
-- Indicates the Short Message related events(s) for notification.

resetTimerSMS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT           ResetTimerSMSArg {bound}
    RETURN RESULT     FALSE
    ERRORS             {missingParameter |
                    parameterOutOfRange |
                    taskRefused |
                    unexpectedComponentSequence |
                    unexpectedDataValue |
                    unexpectedParameter}
    CODE              opcode-resetTimerSMS}
-- Direction: gsmSCF -> smsSSF, Timer: Trtsms
-- This operation is used to request the smsSSF to refresh an application
-- timer in the smsSSF.

ResetTimerSMSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
    timerID            [0] TimerID DEFAULT tssf,
    timervalue         [1] TimerValue,
    extensions         [2] Extensions {bound} OPTIONAL,
    ...
}

END

```

7.1.1 Operation timers

The following value ranges apply for operation specific timers in CAP:

short: 1 to 20 seconds;
medium: 1 to 60 seconds;
long: 1 second to 30 minutes.

Table 7-1 lists all operation timers and the value range for each timer. The definitive value for each operation timer may be network specific and has to be defined by the network operator.

Table 7-1: Operation timers and their value range

Operation Name	Timer	Value range
ConnectSMS	T _{consms}	Short
ContinueSMS	T _{cuesms}	Short
EventReportSMS	T _{erbsms}	Short
FurnishChargingInformationSMS	T _{fcisms}	Short
InitialDPSMS	T _{idpsms}	Short
ReleaseSMS	T _{relsms}	Short
RequestReportSMSEvent	T _{rrbsms}	Short
ResetTimerSMS	T _{rtsms}	Short

7.2 SMS contracts, packages and ACs

7.2.1 SMS ASN.1 module

```
CAP-smsSSF-gsmSCF-pkgs-contracts-acs {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-smsSSF-gsmSCF-pkgs-contracts-acs(106) version4(3)}
```

```
DEFINITIONS ::= BEGIN
```

```
-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the smsSSF - gsmSCF interface, for the
-- control of MO-SMS and MT-SMS.
```

```
-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.
```

```
IMPORTS
```

```
PARAMETERS-BOUND,
cAPSpecificBoundSet
FROM CAP-classes classes
```

```
CONTRACT,
OPERATION-PACKAGE,
OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
TCMessage {}
FROM TCAPMessages tc-Messages
```

```
APPLICATION-CONTEXT,
dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions
```

```
connectSMS{},
continueSMS,
eventReportSMS{},
furnishChargingInformationSMS{},
initialDPSMS{},
releaseSMS,
requestReportSMSEvent{},
resetTimerSMS{}
FROM CAP-SMS-ops-args sms-Operations
```

```
id-ac-cap3-sms-AC,
id-ac-cap4-sms-AC,
id-cap3SmsSsfTogsmScf,
id-cap4SmsSsfTogsmScf,
id-package-smsActivation,
id-package-smsConnect,
id-package-smsContinue,
id-package-smsRelease,
id-package-smsEventHandling,
id-package-smsBilling,
id-package-smsTimer,
sms-Operations,
tc-NotationExtensions,
tc-Messages,
ros-InformationObjects,
classes,
id-as-smsSSF-gsmSCF-AS
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version4(3)}
```

```
;
```

```
-- Application Contexts
```

```
cap3-sms-AC APPLICATION-CONTEXT ::= {
CONTRACT cap3SMS
DIALOGUE MODE structured
ABSTRACT SYNTAXES {dialogue-abstract-syntax |
sms-AbstractSyntax}
APPLICATION CONTEXT NAME id-ac-cap3-sms-AC}
-- This application context shall be used for CAMEL support of MO-SMS.
```

```
cap4-sms-AC APPLICATION-CONTEXT ::= {
```

```

CONTRACT                                cap4SMS
DIALOGUE MODE                            structured
ABSTRACT SYNTAXES                        {dialogue-abstract-syntax |
                                        sms-AbstractSyntax}
APPLICATION CONTEXT NAME                  id-ac-cap4-sms-AC}
-- This application context shall be used for CAMEL support of MT-SMS.

-- Contracts

cap3SMS CONTRACT ::= {
-- dialogue initiated by gprsSSF or gsmSSF with InitialDPSMS Operation
  INITIATOR CONSUMER OF                  {smsActivationPackage {cAPSpecificBoundSet}}
  RESPONDER CONSUMER OF                  {smsConnectPackage {cAPSpecificBoundSet} |
                                        smsReleasePackage |
                                        smsEventHandlingPackage {cAPSpecificBoundSet} |
                                        smsTimerPackage {cAPSpecificBoundSet} |
                                        smsBillingPackage {cAPSpecificBoundSet} |
                                        smsProcessingPackage}
  ID                                      id-cap3SmsSsfTogsmScf}

cap4SMS CONTRACT ::= {
-- dialogue initiated by gprsSSF or gsmSSF with InitialDPSMS Operation
  INITIATOR CONSUMER OF                  {smsActivationPackage {cAPSpecificBoundSet}}
  RESPONDER CONSUMER OF                  {smsConnectPackage {cAPSpecificBoundSet} |
                                        smsReleasePackage |
                                        smsEventHandlingPackage {cAPSpecificBoundSet} |
                                        smsTimerPackage {cAPSpecificBoundSet} |
                                        smsBillingPackage {cAPSpecificBoundSet} |
                                        smsProcessingPackage}
  ID                                      id-cap4SmsSsfTogsmScf}

-- Operation Packages

smsActivationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {initialDPSMS {bound}}
  ID                                       id-package-smsActivation}

smsConnectPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {connectSMS {bound}}
  ID                                       id-package-smsConnect}

smsProcessingPackage OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {continueSMS}
  ID                                       id-package-smsContinue}

smsReleasePackage OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {releaseSMS}
  ID                                       id-package-smsRelease}

smsEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {requestReportSMSEvent {bound}}
  SUPPLIER INVOKES                        {eventReportSMS {bound}}
  ID                                       id-package-smsEventHandling}

smsBillingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {furnishChargingInformationSMS {bound}}
  ID                                       id-package-smsBilling}

smsTimerPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
  CONSUMER INVOKES                        {resetTimerSMS {bound}}
  ID                                       id-package-smsTimer}

-- Abstract Syntaxes

sms-AbstractSyntax ABSTRACT-SYNTAX ::= {
  Generic-sms-PDUs
  IDENTIFIED BY id-as-smsSSF-gsmSCF-AS }

Generic-sms-PDUs ::= TCMessage {{SmsInvokable}, {SmsReturnable}}

SmsInvokable OPERATION ::= {
  connectSMS {cAPSpecificBoundSet} |
  eventReportSMS {cAPSpecificBoundSet} |
  furnishChargingInformationSMS {cAPSpecificBoundSet} |
  initialDPSMS {cAPSpecificBoundSet} |
  requestReportSMSEvent {cAPSpecificBoundSet} |
  resetTimerSMS {cAPSpecificBoundSet}}

```

```
SmsReturnable OPERATION ::= {  
  connectSMS {cAPSpecificBoundSet} |  
  continueSMS |  
  furnishChargingInformationSMS {cAPSpecificBoundSet} |  
  initialDPSMS {cAPSpecificBoundSet} |  
  releaseSMS |  
  requestReportSMSEvent {cAPSpecificBoundSet} |  
  resetTimerSMS {cAPSpecificBoundSet}}
```

END

8 GPRS Control

8.1 gsmSCF/gprsSSF operations and arguments

```
CAP-gprsSSF-gsmSCF-ops-args {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-GPRS-ops-args(107) version3(2)}
```

```
DEFINITIONS IMPLICIT TAGS ::= BEGIN
```

```
-- This module contains the operations and operation arguments used for the
-- gprsSSF - gsmSCF interface, for the control of GPRS.
```

```
-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.
```

```
IMPORTS
```

```
    errortypes,
    datatypes,
    operationcodes,
    classes,
    ros-InformationObjects
```

```
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version3(2)}
```

```
OPERATION
```

```
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
ServiceKey
```

```
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
modules(0) cs1-datatypes(2) version1(0)}
```

```
MiscCallInfo
```

```
FROM CS2-datatypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
cs2(20) modules(0) in-cs2-datatypes (0) version1(0)}
```

```
IMEI,
IMSI,
ISDN-AddressString
```

```
FROM MAP-CommonDataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-CommonDataTypes(18) version6(6)}
```

```
GPRSChargingID,
GPRSMSCClass,
GSN-Address,
LocationInformationGPRS,
RAIdentity
```

```
FROM MAP-MS-DataTypes {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
gsm-Network(1) modules(3) map-MS-DataTypes(11) version6(6)}
```

```
PARAMETERS-BOUND
```

```
FROM CAP-classes classes
```

```
opcode-activityTestGPRS,
opcode-applyChargingGPRS,
opcode-applyChargingReportGPRS,
opcode-cancelGPRS,
opcode-connectGPRS,
opcode-continueGPRS,
opcode-entityReleasedGPRS,
opcode-eventReportGPRS,
opcode-furnishChargingInformationGPRS,
opcode-initialDPGPRS,
opcode-releaseGPRS,
opcode-requestReportGPRSEvent,
opcode-resetTimerGPRS,
opcode-sendChargingInformationGPRS
```

```
FROM CAP-operationcodes operationcodes
```

```
AccessPointName {},
GPRSCause {},
ChargingCharacteristics,
ChargingResult,
```

```

    ChargingRollOver,
    EndUserAddress,

    Extensions,
    FCIGPRSBillingChargingCharacteristics,
    GPRSEventSpecificInformation {},
    GPRSEvent,
    GPRSEventType,
    PDPID,
    PDPInitiationType,
    QualityOfService,
    SCIGPRSBillingChargingCharacteristics,
    SGSNCapabilities,
    TimeAndTimezone {},
    TimerID,
    TimerValue
FROM CAP-datatypes datatypes

    missingCustomerRecord,
    missingParameter,
    parameterOutOfRange,
    systemFailure,
    taskRefused,
    unexpectedComponentSequence,
    unexpectedDataValue,
    unexpectedParameter,
    unknownPDPID
FROM CAP-errortypes errortypes

;

activityTestGPRS OPERATION ::= {
    RETURN RESULT    TRUE
    CODE             opcode-activityTestGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: T_atg
-- This operation is used to check for the continued existence of a relationship between the gsmSCF
-- and gprsSSF. If the relationship is still in existence, then the gprsSSF will respond. If no
-- reply is received, then the gsmSCF will assume that the gprsSSF has failed in some way
-- and will take the appropriate action.

applyChargingGPRS OPERATION ::= {
    ARGUMENT         ApplyChargingGPRSArg
    RETURN RESULT    FALSE
    ERRORS            {missingParameter |
                      unexpectedComponentSequence |
                      unexpectedParameter |
                      unexpectedDataValue |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unknownPDPID}
    CODE             opcode-applyChargingGPRS}
-- Direction gsmSCF -> gprsSSF, Timer Tacg
-- This operation is used for interacting from the gsmSCF with the gprsSSF CSE-controlled
-- GPRS session or PDP Context charging mechanism.

ApplyChargingGPRSArg ::= SEQUENCE {
    ChargingCharacteristics          [0] ChargingCharacteristics,
    tariffSwitchInterval             [1] INTEGER (1..86400)             OPTIONAL,
    pDPIID                           [2] PDPID                        OPTIONAL,
    ...
}
-- tariffSwitchInterval is measured in 1 second units.

ApplyChargingReportGPRS OPERATION ::= {
    ARGUMENT         ApplyChargingReportGPRSArg
    RETURN RESULT    TRUE
    ERRORS            {missingParameter |
                      unexpectedComponentSequence |
                      unexpectedParameter |
                      unexpectedDataValue |
                      parameterOutOfRange |
                      systemFailure |
                      taskRefused |
                      unknownPDPID}
    CODE             opcode-applyChargingReportGPRS}
-- Direction gprsSSF -> gsmSCF, Timer Tacrg

```

```

-- The ApplyChargingReportGPRS operation provides the feedback from the gprsSCF to the gsmSCF
-- CSE-controlled GPRS session charging mechanism.

ApplyChargingReportGPRSArg ::= SEQUENCE {
    ChargingResult          [0] ChargingResult,
    QualityOfService        [1] QualityOfService,
    active                   [2] BOOLEAN DEFAULT TRUE,
    pDPID                   [3] PDPID,
    ...,
    chargingRollOver        [4] ChargingRollOver
}

cancelGPRS OPERATION ::= {
    ARGUMENT      CancelGPRSArg
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  taskRefused |
                  unknownPDPID}
    CODE          opcode-cancelGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tcag
-- This generic operation cancels all previous requests,
-- i.e. all EDPs and reports can be cancelled by the gsmSCF.

CancelGPRSArg ::= SEQUENCE {
    pDPID          [0] PDPID,
    ...
}

connectGPRS {PARAMETERS-BOUND: bound} OPERATION ::= {
    ARGUMENT      ConnectGPRSArg {bound}
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  parameterOutOfRange |
                  unknownPDPID |
                  systemFailure |
                  taskRefused |
                  unexpectedComponentSequence |
                  unexpectedDataValue |
                  unexpectedParameter}
    CODE          opcode-connectGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tcong
-- This operation is used to modify the Access Point Name used when establishing a PDP Context.

ConnectGPRSArg {PARAMETERS-BOUND: bound} ::= SEQUENCE {
    accessPointName [0] AccessPointName {bound},
    pdpID           [1] PDPID,
    ...
}

continueGPRS OPERATION ::= {
    ARGUMENT      ContinueGPRSArg
    RETURN RESULT FALSE
    ERRORS        {missingParameter |
                  unknownPDPID |
                  unexpectedDataValue}
    CODE          opcode-continueGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tcueg
-- This operation is used to request the gprsSSF to proceed with processing at the DP at
-- which it previously suspended processing to await gsmSCF instructions (i.e., proceed to
-- the next point in processing in the Attach/Detach state model or PDP Context
-- state model) substituting new data from the gsmSCF.

ContinueGPRSArg ::= SEQUENCE {
    pDPID          [0] PDPID,
    ...
}

entityReleasedGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
    ARGUMENT      EntityReleasedGPRSArg {bound}
    RETURN RESULT TRUE
    ERRORS        {missingParameter |
                  taskRefused |
                  unknownPDPID}
    CODE          opcode-entityReleasedGPRS}
-- Direction: gprsSSF -> gsmSCF, Timer: Terg
-- This operation is used when the GPRS Session is detached or a PDP Context is diconnected and
-- the associated event is not armed for reporting.

```

-- The usage of this operation is independent of the functional entity that initiates the Detach
 -- or PDP Context Disconnection and is independent of the cause of the Detach or PDP Context
 -- Disconnect.

```
EntityReleasedGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  gPRSCause          [0] GPRSCause {bound},
  pDPID              [1] PDPID
  ...
}
```

```
eventReportGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT          EventReportGPRSArg {bound}
  RETURN RESULT     TRUE
  ERRORS            {unknownPDPID}
  CODE              opcode-eventReportGPRS}
-- Direction gprsSSF -> gsmSCF,Timer Tereg
-- This operation is used to notify the gsmSCF of a GPRS session or PDP context related
-- events (e.g. PDP context activation) previously requested by the gsmSCF in a
-- RequestReportGPRSEventoperation.
```

```
EventReportGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  GPRSEventType     [0] GPRSEventType,
  MiscGPRSInfo      [1] MiscCallInfo DEFAULT {messageType request},
  GPRSEventSpecificInformation [2] GPRSEventSpecificInformation {bound} OPTIONAL,
  pDPID             [3] PDPID OPTIONAL,
  ...
}
```

```
furnishChargingInformationGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT          FurnishChargingInformationGPRSArg {bound}
  RETURN RESULT     FALSE
  ERRORS            {missingParameter |
                    taskRefused |
                    unexpectedComponentSequence |
                    unexpectedDataValue |
                    unexpectedParameter |
                    unknownPDPID}
  CODE              opcode-furnishChargingInformationGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Tfcig
-- This operation is used to request the gprsSSF to generate, register a logical record or to
-- include some information in the default logical GPRS record.
-- The registered logical record is intended for off line charging of the GPRS session
-- or PDP Context.
```

```
FurnishChargingInformationGPRSArg {PARAMETERS-BOUND : bound} ::=
  FCIGPRSBillingChargingCharacteristics{bound}
```

```
initialDPGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT          InitialDPGPRSArg {bound}
  RETURN RESULT     FALSE
  ERRORS            {missingCustomerRecord |
                    missingParameter |
                    parameterOutOfRange |
                    systemFailure |
                    taskRefused |
                    unexpectedComponentSequence |
                    unexpectedDataValue |
                    unexpectedParameter}
  CODE              opcode-initialDPGPRS}
-- Direction gprsSSF -> gsmSCF,Timer Tidpg
-- This operation is used by the gprsSSF when a trigger is detected at a DP in the GPRS state
-- machines to request instructions from the gsmSCF
```

```
InitialDPGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  ServiceKey        [0] ServiceKey,
  GPRSEventType     [1] GPRSEventType,
  mSISDN            [2] ISDN-AddressString,
  IMSI              [3] IMSI,
  timeAndTimeZone   [4] TimeAndTimezone {bound},
  gPRSMSCClass      [5] GPRSMSCClass
  ...
  endUserAddress    [6] EndUserAddress {bound}
  qualityOfService  [7] QualityOfService
  accessPointName   [8] AccessPointName{bound}
  routingAreaIdentity [9] RAIdentity
  chargingID        [10] GPRSChargingID
  sGSNCapabilities [11] sGSNCapabilities
  locationInformationGPRS [12] LocationInformationGPRS
  pDPInitiationType [13] PDPInitiationType
```

```

    extensions                [14] Extensions {bound}                OPTIONAL,
    ...,
    gGSNAddress               [15] GSN-Address                    OPTIONAL,
    secondaryPDP-context      [16] NULL                                                            OPTIONAL,
    iMEI                       [17] IMEI                                                                OPTIONAL
  }
-- The RoutingAreaIdentity parameter is not used.
-- The receiving entity shall ignore RoutingAreaIdentity if received.
-- The RoutingAreaIdentity is conveyed in the LocationInformationGPRS parameter.

releaseGPRS {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      ReleaseGPRSArg {bound}
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 taskRefused |
                 unknownPDPID}
  CODE          opcode-releaseGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Trg
-- This operation is used to tear down an existing GPRS session or PDP Context at any phase.

ReleaseGPRSArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  gprsCause      [0] GPRSCause {bound},
  pDPID          [1] PDPID                                OPTIONAL,
  ...
}

requestReportGPRSEvent {PARAMETERS-BOUND : bound} OPERATION ::= {
  ARGUMENT      RequestReportGPRSEventArg {bound}
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 parameterOutOfRange |
                 systemFailure |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter |
                 unknownPDPID}
  CODE          opcode-requestReportGPRSEvent}
-- Direction: gsmSCF -> gprsSSF, Timer: Trrqe
-- This operation is used to request the gprsSSF to monitor for an event (e.g., GPRS events
-- such as attach or PDP Context activation), then send a notification back to the
-- gsmSCF when the event is detected.

RequestReportGPRSEventArg {PARAMETERS-BOUND : bound} ::= SEQUENCE {
  gPRSEvent      [0] SEQUENCE SIZE (1..bound.&numOfGPRSEvents) OF GPRSEvent,
  pDPID          [1] PDPID                                OPTIONAL,
  ...
}
-- Indicates the GPRS related events for notification.

ResetTimerGPRS OPERATION ::= {
  ARGUMENT      ResetTimerGPRSArg
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 parameterOutOfRange |
                 taskRefused |
                 unexpectedComponentSequence |
                 unexpectedDataValue |
                 unexpectedParameter |
                 unknownPDPID}
  CODE          opcode-resetTimerGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: Trtg
-- This operation is used to request the gprsSSF to refresh an application timer in the gprsSSF.

ResetTimerGPRSArg ::= SEQUENCE {
  timerID        [0] TimerID                                DEFAULT tssf,
  timerValue     [1] TimerValue,
  ...
}

sendChargingInformationGPRS {PARAMETERS-BOUND: bound} OPERATION ::= {
  ARGUMENT      SendChargingInformationGPRSArg { bound}
  RETURN RESULT FALSE
  ERRORS        {missingParameter |
                 unexpectedComponentSequence |
                 unexpectedParameter |
                 parameterOutOfRange |

```

```

        systemFailure |
        taskRefused |
        unexpectedDataValue |
        unknownPDPID}
    CODE
        opcode-sendChargingInformationGPRS}
-- Direction: gsmSCF -> gprsSSF, Timer: T_scig
-- This operation is used to instruct the gprsSSF on the charging information which the
-- gprsSSF shall send to the Mobile Station by means of GSM access signalling.

SendChargingInformationGPRSArg {PARAMETERS-BOUND: bound} ::= SEQUENCE {
    SCIGPRSBillingChargingCharacteristics [0] SCIGPRSBillingChargingCharacteristics { bound},
    ...
}

END

CAP-GPRS-ReferenceNumber {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-dialogueInformation(111) version3(2)}
DEFINITIONS ::= BEGIN

EXPORTS
    id-CAP-GPRS-ReferenceNumber,
    cAP-GPRS-ReferenceNumber-Abstract-Syntax;

IMPORTS

    Integer4
FROM CS1-DataTypes {itu-t(0) identified-organization(4) etsi(0) inDomain(1) in-network(1)
modules(0) cs1-datatypes(2) version1(0)}
;

id-CAP-GPRS-ReferenceNumber OBJECT IDENTIFIER ::= {itu-t(0) identified-organization(4) etsi(0)
mobileDomain(0) umts-network(1) as(1) cap-GPRS-ReferenceNumber(5) version3(2)}

cAP-GPRS-ReferenceNumber-Abstract-Syntax ABSTRACT-SYNTAX ::= {CAP-GPRS-ReferenceNumber IDENTIFIED BY
id-CAP-GPRS-ReferenceNumber}

CAP-GPRS-ReferenceNumber ::= SEQUENCE {
    DestinationReference [0] Integer4 OPTIONAL,
    OriginationReference [1] Integer4 OPTIONAL
}
-- This parameter is used to identify the relationship between SGSN and the gsmSCF.

END -- of CAP-GPRS-ReferenceNumber

```

8.1.1 Operation timers

The following value ranges apply for operation specific timers in CAP:

short:	1 to 20 seconds;
medium:	1 to 60 seconds;
long:	1 second to 30 minutes

Table 8-1 lists all operation timers and the value range for each timer. The definitive value for each operation timer may be network specific and has to be defined by the network operator.

Table 8-1: Operation timers and their value range

Operation Name	Timer	value
ActivityTestGPRST	T _{atg}	Short
ApplyChargingGPRS	T _{acg}	Short
ApplyChargingReportGPRS	T _{acrg}	Short
CancelGPRS	T _{cag}	Short
ConnectGPRS	T _{cong}	Short
ContinueGPRS	T _{cueg}	Short
EntityReleasedGPRS	T _{erg}	Short
EventReportGPRS	T _{ereg}	Short
FurnishChargingInformationGPRS	T _{fcig}	Short
InitialDPGPRS	T _{idpg}	Short
ReleaseGPRS	T _{rg}	Short
RequestReportGPRSEvent	T _{rrqe}	Short
ResetTimerGPRS	T _{rtg}	Short
SendChargingInformationGPRS	T _{scig}	Short

8.2 gsmSCF/gprsSSF contracts, packages and ACs

8.2.1 gprsSSF/gsmSCF ASN.1 module

```
CAP-gprsSSF-gsmSCF-pkgs-contracts-acs {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-gprsSSF-gsmSCF-pkgs-contracts-acs(108) version3(2)}
```

```
DEFINITIONS ::= BEGIN
```

```
-- This module specifies the Operation Packages, Contracts, Application Contexts
-- and Abstract Syntaxes used for the gprsSSF - gsmSCF interface, for the
-- control of GPRS.
```

```
-- The table in subclause 2.1 lists the specifications that contain the modules
-- that are used by CAP.
```

```
IMPORTS
```

```
PARAMETERS-BOUND,
CAPSpecificBoundSet
FROM CAP-classes classes
```

```
CONTRACT,
OPERATION-PACKAGE,
OPERATION
FROM Remote-Operations-Information-Objects ros-InformationObjects
```

```
TCMessage {}
FROM TCAPMessages tc-Messages
```

```
APPLICATION-CONTEXT,
dialogue-abstract-syntax
FROM TC-Notation-Extensions tc-NotationExtensions
```

```
activityTestGPRS,
applyChargingGPRS,
applyChargingReportGPRS,
cancelGPRS,
connectGPRS {},
continueGPRS,
entityReleasedGPRS {},
furnishChargingInformationGPRS {},
initialDPGPRS {},
releaseGPRS {},
eventReportGPRS {},
requestReportGPRSEvent {},
resetTimerGPRS,
```

```

    sendChargingInformationGPRS {}
FROM CAP-gprSSSF-gsmSCF-ops-args gprSSSF-gsmSCF-Operations

    id-ac-CAP-gprSSSF-gsmSCF-AC,
    id-ac-CAP-gsmSCF-gprSSSF-AC,
    id-cap3GprSSsfToGsmScf,
    id-cap3GsmScfToGprSSsf,
    id-as-gprSSSF-gsmSCF-AS,
    id-as-gsmSCF-gprSSSF-AS,
    id-package-gprSScfActivation,
    id-package-gprSSconnect,
    id-package-gprSScontinue,
    id-package-gprSSrelease,
    id-package-gprSSeventHandling,
    id-package-gprSSexceptionInformation,
    id-package-gprSStimer,
    id-package-gprSSbilling,
    id-package-gprSScharging,
    id-package-gprSSchargeAdvice,
    id-package-gprSSactivityTest,
    id-package-gprSScancel,
    classes,
    ros-InformationObjects,
    tc-Messages,
    tc-NotationExtensions,
    gprSSSF-gsmSCF-Operations
FROM CAP-object-identifiers {itu-t(0) identified-organization(4) etsi(0) mobileDomain(0)
umts-network(1) modules(3) cap-object-identifiers(100) version3(2)}

;

-- Application Contexts

cap3-gprSSsf-scfAC APPLICATION-CONTEXT ::= {
    CONTRACT                cap3GprSSsfToScf
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             gprSSSF-gsmSCFAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gprSSSF-gsmSCF-AC}

cap3-gsmScf-gprSSsfAC APPLICATION-CONTEXT ::= {
    CONTRACT                cap3GsmScfToGprSSsf
    DIALOGUE MODE           structured
    ABSTRACT SYNTAXES       {dialogue-abstract-syntax |
                             gsmSCF-gprSSSFAbstractSyntax}
    APPLICATION CONTEXT NAME id-ac-CAP-gsmSCF-gprSSSF-AC}

-- Contracts

cap3GprSSsfToScf CONTRACT ::= {
-- dialogue initiated by gprSSSF with InitialDPGPRS, ApplyChargingReportGPRS,
-- EntityReleaseGPRS and EventReportGPRS Operations
    INITIATOR CONSUMER OF  {gprSScfActivationPackage {cAPSSpecificBoundSet} |
                             gprSSeventHandlingPackage {cAPSSpecificBoundSet} |
                             gprSSchargingPackage |
                             gprSSexceptionInformationPackage {cAPSSpecificBoundSet}}
    RESPONDER CONSUMER OF  {gprSSconnectPackage {cAPSSpecificBoundSet} |
                             gprSSprocessingPackage |
                             gprSSreleasePackage {cAPSSpecificBoundSet} |
                             gprSSeventHandlingPackage {cAPSSpecificBoundSet} |
                             gprSStimerPackage |
                             gprSSbillingPackage {cAPSSpecificBoundSet} |
                             gprSSchargingPackage |
                             gprSScancelPackage |
                             gprSSchargeAdvicePackage {cAPSSpecificBoundSet}}
    ID                      id-cap3GprSSsfToGsmScf}

cap3GsmScfToGprSSsf CONTRACT ::= {
-- dialogue initiated by gsmSCF with ApplyCharginGPRS, ActivityTestGPRS,
-- CancelGPRS, FurnishChargingInformationGPRS, ReleaseGPRS,
-- RequestReportGPRSEvent and SendChargingInformationGPRS Operations
    INITIATOR CONSUMER OF  {gprSSreleasePackage {cAPSSpecificBoundSet} |
                             gprSSeventHandlingPackage {cAPSSpecificBoundSet} |
                             gprSSbillingPackage {cAPSSpecificBoundSet} |
                             gprSSchargingPackage |
                             gprSSactivityTestPackage |
                             gprSScancelPackage |

```



```

        gprsChargeAdvicePackage {cAPSpecificBoundSet}}
    ID                               id-cap3GsmScfToGprsSsf}

-- Operation Packages

gprsScfActivationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {initialDPGPRS {bound}}
    ID                  id-package-gprsScfActivation}

gprsConnectPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {connectGPRS {bound}}
    ID                  id-package-gprsConnect}

gprsProcessingPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {continueGPRS }
    ID                  id-package-gprsContinue}

gprsReleasePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {releaseGPRS {bound}}
    ID                  id-package-gprsRelease}

gprsEventHandlingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {requestReportGPRSEvent {bound}}
    SUPPLIER INVOKES    {eventReportGPRS {bound}}
    ID                  id-package-gprsEventHandling}

gprsExceptionInformationPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {entityReleasedGPRS {bound}}
    ID                  id-package-gprsExceptionInformation}

gprsTimerPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {resetTimerGPRS }
    ID                  id-package-gprsTimer}

gprsBillingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {furnishChargingInformationGPRS {bound}}
    ID                  id-package-gprsBilling}

gprsChargingPackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {applyChargingGPRS {bound}}
    SUPPLIER INVOKES    {applyChargingReportGPRS {bound}}
    ID                  id-package-gprsCharging}

gprsChargeAdvicePackage {PARAMETERS-BOUND : bound} OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {sendChargingInformationGPRS {bound}}
    ID                  id-package-gprsChargeAdvice}

gprsActivityTestPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {activityTestGPRS}
    ID                  id-package-gprsActivityTest}

gprsCancelPackage OPERATION-PACKAGE ::= {
    CONSUMER INVOKES    {cancelGPRS }
    ID                  id-package-gprsCancel}

-- Abstract Syntaxes

gprsSSF-gsmSCFAbstractSyntax ABSTRACT-SYNTAX ::= {
    GenericGprsSSF-gsmSCF-PDUs
    IDENTIFIED BY    id-as-gprsSSF-gsmSCF-AS}

GenericGprsSSF-gsmSCF-PDUs ::= TCMessgae {{GprsSsfToGsmScfInvokable},
    {GprsSsfToGsmScfReturnable}}

GprsSsfToGsmScfInvokable OPERATION ::= {
    activityTestGPRS |
    applyChargingGPRS |
    applyChargingReportGPRS |
    cancelGPRS |
    connectGPRS {cAPSpecificBoundSet} |
    entityReleasedGPRS {cAPSpecificBoundSet} |
    eventReportGPRS {cAPSpecificBoundSet} |
    furnishChargingInformationGPRS {cAPSpecificBoundSet} |
    initialDPGPRS {cAPSpecificBoundSet} |
    releaseGPRS {cAPSpecificBoundSet} |
    requestReportGPRSEvent {cAPSpecificBoundSet} |

```

```
    resetTimerGPRS |
    sendChargingInformationGPRS {cAPSpecificBoundSet}}

GprsSsfToGsmScfReturnable OPERATION ::= {
    activityTestGPRS |
    applyChargingGPRS |
    applyChargingReportGPRS |
    cancelGPRS |
    connectGPRS {cAPSpecificBoundSet} |
    continueGPRS |
    entityReleasedGPRS {cAPSpecificBoundSet} |
    furnishChargingInformationGPRS {cAPSpecificBoundSet} |
    initialDPGPRS {cAPSpecificBoundSet} |
    releaseGPRS {cAPSpecificBoundSet} |
    requestReportGPRSEvent {cAPSpecificBoundSet} |
    resetTimerGPRS |
    sendChargingInformationGPRS {cAPSpecificBoundSet}}

gsmSCF-gprsSSFGenericAbstractSyntax ABSTRACT-SYNTAX ::= {
    GenericGsmSCF-gprsSSF-PDUs
    IDENTIFIED BY id-as-gsmSCF-gprsSSF-AS}

GenericGsmSCF-gprsSSF-PDUs ::= TCMMessage {{GsmScfToGprsSsfInvokable}, {GsmScfToGprsSsfReturnable}}

GsmScfToGprsSsfInvokable OPERATION ::= {
    activityTestGPRS |
    applyChargingGPRS |
    cancelGPRS |
    furnishChargingInformationGPRS {cAPSpecificBoundSet} |
    releaseGPRS {cAPSpecificBoundSet} |
    requestReportGPRSEvent {cAPSpecificBoundSet} |
    sendChargingInformationGPRS {cAPSpecificBoundSet}}

GsmScfToGprsSsfReturnable OPERATION ::= {
    activityTestGPRS |
    applyChargingGPRS |
    cancelGPRS |
    furnishChargingInformationGPRS {cAPSpecificBoundSet} |
    releaseGPRS {cAPSpecificBoundSet} |
    requestReportGPRSEvent {cAPSpecificBoundSet} |
    sendChargingInformationGPRS {cAPSpecificBoundSet}}

END
```

9 Application Entity procedures

The description of the application entity procedures for CAMEL Phase 3 can be found in 3GPP TS 23.078 [7].

10 Error procedures

The present clause specifies the generic error procedures for CAP. The error procedure descriptions are divided in two subclauses, subclause 10.1 listing the errors related to CAP Operations and subclause 10.2 listing the errors related to error conditions in the different FEs which are not directly related to the CAP Operations.

The gsmSSF FSM states, smsSSF FSM states and gprsSSF FSM states which are referred to in the present clause are described in 3GPP TS 23.078 [7]. The CAP Operations PlayAnnouncement, PromptAndCollectUserInformation and SpecializedResourceReport refer to states in the gsmSRF SRSM which are described in ETSI ETS 300 374-1 [24] as well as to states in 3GPP TS 23.078 [7].

10.1 Operation related error procedures

The following subclauses specify the generic error handling for the CAP Operation related errors. The errors are defined as Operation Errors in clauses 6, 7 and 8. Errors which have a specific procedure for a CAP Operation are described in clause 11 with the detailed procedure of the related CAP Operation.

The TC services which are used for reporting CAP Operation Errors are described in subclause 14.1. All Errors which can be detected by the ASN.1 decoder, already may have been detected during the decoding of the TC message and indicated by the TC error indication "MistypedParameter" in the TC-U-Reject.

10.1.1 Canceled

10.1.1.1 General Description

10.1.1.1.1 Error description

The gsmSRF uses this Error to indicate to the gsmSCF that the cancellation of a specific CAP Operation, as requested by the gsmSCF, has been successful. The only CAP Operations which may be cancelled are PlayAnnouncement and PromptAndCollectUserInformation.

10.1.1.2 Operations gsmSCF→gsmSRF

Procedures at responding entity (gsmSRF)

A) Receiving Cancel

Precondition: SRSM FSM state "User Interaction".
Postcondition: SRSM FSM state "User Interaction".

If execution of the indicated CAP Operation has not yet started, then the indicated CAP Operation shall be removed from the buffer. If execution of the indicated CAP Operation is in progress, then the execution of that CAP Operation shall be terminated. If the indicated CAP Operation was already executed, then then this causes a failure ("CancelFailed").

B) Sending Error "Canceled"

Precondition: SRSM FSM state "User Interaction".
Postcondition: SRSM FSM state "User Interaction".

After returning the Error "Canceled", the gsmSRF FSM remains in the same state. The execution of the indicated PlayAnnouncement or PromptAndCollectUserInformation CAP Operation is aborted. The gsmSRF shall remain connected and the next PlayAnnouncement or PromptAndCollectUserInformation CAP Operation, if available, shall be executed.

10.1.2 CancelFailed

10.1.2.1 General description

10.1.2.1.1 Error description

The gsmSRF uses this Error to indicate to the gsmSCF that the cancellation of a specific CAP Operation, as requested by the gsmSCF, was not successful. Possible failure reasons are:

- 0 "unknownOperation", when the InvokeID of the CAP Operation to cancel is not known to the gsmSRF (this may also happen when the CAP Operation has already been completed);
- 1 "tooLate", when the invokeID is known but the execution of the CAP Operation is in a state in which it cannot be canceled anymore. For instance the announcement is finished but the SpecializedResourceReport has not been sent to the gsmSCF yet. The conditions for the occurrence of failure reason "tooLate" are implementation dependent;
- 2 "operationNotCancellable", when the invokeID points to a CAP Operation that the gsmSCF is not allowed to cancel.

10.1.2.2 Operations gsmSCF→gsmSRF

Procedures at responding entity (gsmSRF)

A) Receiving Cancel. However, the indicated CAP Operation is not known, or has already been executed or may not be cancelled. This causes a failure, "CancelFailed".

Precondition: SRSM FSM state "User Interaction".

Postcondition: SRSM FSM state "User Interaction" or
SRSM FSM state "Idle".

B) Sending Error "CancelFailed"

Precondition: SRSM FSM state "User Interaction" or
SRSM FSM state "Idle".

Postcondition: SRSM FSM state "User Interaction" or
SRSM FSM state "Idle".

After returning the Error "CancelFailed" the gsmSRF FSM remains in the same state.

10.1.3 ETCFailed

10.1.3.1 General description

10.1.3.1.1 Error description

The gsmSSF uses this Error to indicate to the gsmSCF that the establishment of a temporary connection to an assisting gsmSSF or gsmSRF was not successful (e.g., receiving a "Backwards Release" after sending an ISUP IAM).

10.1.3.2 Operations gsmSCF→gsmSSF

Procedures at responding entity (gsmSSF)

A gsmSSF receives EstablishTemporaryConnection from a gsmSCF but the establishment of the connection fails, which results in returning the Error "ETCFailed" to the gsmSCF.

Precondition: gsmSSF FSM state "Waiting_for_Instructions".

Postcondition: gsmSSF FSM state "Waiting_for_Instructions".

No further error treatment.

10.1.4 ImproperCallerResponse

10.1.4.1 General description

10.1.4.1.1 Error description

The gsmSRF uses this Error to indicate to the gsmSCF that the format of the user input has been checked by the gsmSRF, but does not correspond to the required format as it was defined in the initiating CAP Operation.

10.1.4.2 Operations gsmSCF→gsmSRF

Procedures at responding entity (gsmSRF)

A) gsmSRF receives PromptAndCollectUserInformation

Precondition: SRSM FSM state "Connected"; or
SRSM FSM state "User Interaction".

Postcondition: SRSM FSM state "User Interaction".

B) format of the user input is not correct, gsmSRF returns ImproperCallerResponse to gsmSCF

Precondition: SRSM FSM state "User Interaction".

Postcondition: SRSM FSM state "User Interaction".

The gsmSRF shall wait for a new CAP Operation from the gsmSCF. This may be a new PromptAndCollectUserInformation or a PlayAnnouncement.

10.1.5 MissingCustomerRecord

10.1.5.1 General description

10.1.5.1.1 Error description

The gsmSCF uses this Error to indicate to the gsmSSF, gprsSSF, smsSSF or gsmSRF, that the SLP could not be found in the gsmSCF, because the required customer record does not exist, or the requested SLPI, indicated by the correlationID in "AssistRequestInstructions" does not exist anymore.

10.1.5.2 Operations gsmSSF→gsmSCF

Procedures at invoking entity (gsmSSF)

A) Sending CAP Operation

Precondition: gsmSSF FSM state "Trigger processing" or
gsmSSF FSM state "Waiting_for_Instructions"; in the assisting gsmSSF case.

Postcondition: gsmSSF FSM state "Waiting_for_Instructions".
gsmSSF FSM state "Waiting_for_Instructions"; in the assisting gsmSSF case.

B) gsmSSF receives Error "MissingCustomerRecord"

Precondition: gsmSSF FSM state "Waiting_for_Instructions" or
gsmSSF FSM state "Waiting_for_Instructions"; in the assisting gsmSSF case.

Postcondition: gsmSSF FSM state "Idle" or
gsmSSF FSM state "Idle"; in the assisting gsmSSF case.

The GMSC or VMSC shall handle the call in accordance with the Default Call Handling parameter of the valid CSI.

10.1.5.3 Operations gsmSRF→gsmSCF

Procedures at invoking entity (gsmSRF)

A) Sending CAP Operation

Precondition: SRSM FSM state "Connected".

Postcondition: SRSM FSM state "Connected".

B) gsmSRF receives Error "MissingCustomerRecord"

Precondition: SRSM FSM state "Connected".

Postcondition: SRSM FSM state "Idle".

The gsmSRF shall initiate a Disconnect.

10.1.5.4 Operations smsSSF→gsmSCF

Procedures at invoking entity (smsSSF)

A) Sending CAP Operation

Precondition: smsSSF FSM state "Waiting_for_Instructions".

Postcondition: smsSSF FSM state "Waiting_for_Instructions".

B) smsSSF receives Error "MissingCustomerRecord"

Precondition: smsSSF FSM state "Waiting_for_Instructions".

Postcondition: smsSSF FSM state "Idle".

The VMSC or SGSN shall handle the Short Message in accordance with the Default SMS Handling parameter of the valid CSI.

10.1.5.5 Operations gprsSSF→gsmSCF

Procedures at invoking entity (gprsSSF)

A) Sending CAP Operation

Precondition: gprsSSF FSM state "Waiting_for_Instructions".

Postcondition: gprsSSF FSM state "Waiting_for_Instructions".

B) gprsSSF receives Error "MissingCustomerRecord"

Precondition: gprsSSF FSM state "Waiting_for_Instructions".

Postcondition: gprsSSF FSM state "Idle".

The SGSN shall handle the GPRS Session or PDP Context in accordance with the Default GPRS Handling parameter of the valid CSI.

10.1.6 MissingParameter

10.1.6.1 General description

10.1.6.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this Error to indicate that there is an error in the received CAP Operation argument. The responding entity cannot start the execution of the requested CAP Operation because the argument is incorrect: an expected optional parameter which is essential for the application is not included in the CAP Operation argument.

10.1.6.2 Operations gsmSCF→gsmSSF

Procedures at responding entity (gsmSSF)

Precondition: (1) gsmSSF FSM appropriate state
 (2) gsmSSF FSM Call associated CAP Operation received, appropriate event occurred
 (3) gsmSSME FSM appropriate state
 (4) gsmSSME FSM Non call associated CAP Operation received, appropriate event

Postcondition: (1) No gsmSSF FSM transition
 (2) gsmSSME FSM transition to the initial state (i.e., before receiving the erroneous CAP Operation)

The gsmSSF detects the error in the received CAP Operation. The Error parameter "MissingParameter" is returned to inform the gsmSCF of this situation.

10.1.6.3 Operations gsmSSF→gsmSCF

Procedures at invoking entity (gsmSSF)

A) Sending CAP Operation

Precondition: gsmSSF FSM appropriate state

Postcondition: gsmSSF FSM appropriate state as result of the transfer of the CAP Operation

B) gsmSSF receives Error "MissingParameter"

Precondition: gsmSSF FSM appropriate state as result of the transfer of any of the CAP Operation

Postcondition: gsmSSF FSM state "Idle"

After receiving this Error, the gsmSSF FSM shall return to the state "Idle". The GMSC or VMSC shall handle the call in accordance with the Default Call Handling parameter of the valid CSI. In the case of an assisting gsmSSF, the temporary connection shall be released by the assisting gsmSSF.

10.1.6.4 Operations gsmSCF→gsmSRF

Procedures at responding entity (gsmSRF)

Precondition: SRSM FSM state "Connected"; or
 SRSM FSM state "User Interaction".

Postcondition: SRSM FSM state "User Interaction".

The SRSM detects that a required parameter is not present in the CAP Operation argument. The Error parameter "MissingParameter" is returned to inform the gsmSCF of this situation.

10.1.6.5 Operations gsmSRF→gsmSCF

Procedures at invoking entity (gsmSRF)

A) Sending CAP Operation

Precondition: SRSM FSM state "Connected".

Postcondition: SRSM FSM state "Connected".

B) Receiving Error

Precondition: SRSM FSM state "Connected".

Postcondition: SRSM FSM state "Idle".

The gsmSCF detects the error in the received CAP Operation. The Error parameter "MissingParameter" is used to inform the gsmSRF of this situation. The SL and maintenance functions are informed. The gsmSCF might try another gsmSRF, route the call or release the call (SL dependent).

10.1.6.6 Operations smsSSF→gsmSCF

Procedures at invoking entity (smsSSF)

A) Sending CAP Operation

Precondition: smsSSF FSM state "WaitingForInstructions".

Postcondition: smsSSF FSM state "WaitingForInstructions".

B) Receiving Error

Precondition: smsSSF FSM state "WaitingForInstructions".

Postcondition: smsSSF FSM state "Idle".

After receiving this error, the smsSSF FSM shall transit to the state "Idle". The SGSN or VMSC shall handle the Short Message in accordance with the Default SMS Handling parameter of the valid CSI.

10.1.6.7 Operations gsmSCF → smsSSF

Procedures at responding entity (smsSSF)

precondition:

(1) smsSSF appropriate state.

(2) smsSSF SMS associated CAP Operation received, appropriate event occurred.

postcondition:

(1) smsSSF transition to the same state.

The smsSSF detects the error in the received CAP Operation. The Error parameter "MissingParameter" is returned to inform the gsmSCF of this situation.

10.1.6.8 Operations gprsSSF→gsmSCF

Procedures at invoking entity (gprsSSF)

A) Sending CAP Operation

Precondition: gprsSSF FSM state "Waiting_for_Instructions".

Postcondition: gprsSSF FSM state "Waiting_for_Instructions".

B) Receiving Error

Precondition: gprsSSF FSM state "Waiting_for_Instructions".

Postcondition: gprsSSF FSM state "Idle".

After receiving this error, the gprsSSF FSM shall transit to the state "Idle". The SGSN shall handle the GPRS Session or PDP Context in accordance with the Default GPRS Handling parameter of the valid CSI.

10.1.6.9 Operations gsmSCF→gprsSSF

Procedures at responding entity (gprsSSF)

precondition:

- (1) gprsSSF appropriate state.
- (2) gprsSSF CAP Operation received, appropriate event occurred.

postcondition:

- (1) gprsSSF transition to the same state.

The gprsSSF detects the error in the received CAP Operation. The Error parameter "MissingParameter" is returned to inform the gsmSCF of this situation.

10.1.7 ParameterOutOfRange

10.1.7.1 General description

10.1.7.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this Error to indicate that it cannot start the execution of the requested CAP Operation because an error in a parameter of the CAP Operation argument is detected: a parameter value is out of range.

10.1.7.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.7.3 Operations gsmSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.7.4 Operations gsmSCF→gsmSRF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.7.5 Operations smsSSF → gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.7.6 Operations gsmSCF→smsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.7.7 Operations gprsSSF →gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.7.8 Operations gsmSCF→gprsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.8 RequestedInfoError

10.1.8.1 General description

10.1.8.1.1 Error description

The gsmSSF uses this Error to indicate to the gsmSCF that the information requested in the CallInformationRequest CAP Operation is not known to the gsmSSF or is not available.

10.1.8.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9 SystemFailure

10.1.9.1 General description

10.1.9.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this error to indicate that it is not able to execute a specific task as requested by a CAP Operation, and recovery is not expected to be completed within the current call instance.

10.1.9.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.3 Operations gsmSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.4 Operations gsmSCF→gsmSRF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.5 Operations gsmSRF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.6 Operations smsSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.7 Operations gsmSCF→smsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.8 Operations gprsSSF →gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.9.9 Operations gsmSCF→gprsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10 TaskRefused

10.1.10.1 General description

10.1.10.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this Error to indicate that it is not able to execute a specific task as requested by a CAP Operation, and recovery is expected to be completed within the current call instance.

10.1.10.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.3 Operations gsmSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.4 Operations gsmSCF→gsmSRF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.5 Operations gsmSRF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.6 Operations smsSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.7 Operations gsmSCF→smsSSF

SMS Related

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.8 Operations gprsSSF →gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.10.9 Operations gsmSCF→gprsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.11 UnavailableResource

10.1.11.1 General description

10.1.11.1.1 Error description

The gsmSRF uses this Error to indicate to the gsmSCF that it is not able to perform its function (i.e., play a certain announcement and/or collect specific user information), and cannot be replaced. A reattempt is not possible.

10.1.11.2 Operations gsmSCF→gsmSRF

Procedures at responding entity (gsmSRF)

A) gsmSRF receiving PlayAnnouncement or PromptAndCollectUserInformation

Precondition: SRSM FSM state "Connected"; if initial PlayAnnouncement or PromptAndCollectUserInformation; or
SRSM FSM state "User Interaction"; if not initial PlayAnnouncement or PromptAndCollectUserInformation.

B) gsmSRF is not able to perform its function (and cannot be replaced). The gsmSRF sends the Error "UnavailableResource".

Precondition: SRSM FSM state "User Interaction".

Postcondition: SRSM FSM state "User Interaction".

10.1.12 UnexpectedComponentSequence

10.1.12.1 General description

10.1.12.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this Error to indicate that it cannot start the execution of the requested CAP Operation because a SACF or MACF rule is violated, or the CAP Operation cannot be executed in the current state of the FSM.

10.1.12.2 Operations gsmSCF→gsmSSF

In this case the gsmSSF detects the erroneous situation, sends the Error "UnexpectedComponentSequence" and remains in the same state.

10.1.12.3 Operations gsmSSF→gsmSCF

If the CAP Operation is sent by an "initiating" gsmSSF in the context of an existing relationship, then the gsmSCF returns the Error parameter. On receiving the error the gsmSSF FSM shall transit to the state "Idle". The VMSC or GMSC shall handle the call in accordance with the Default Call Handling parameter of the valid CSI.

10.1.12.4 Operations gsmSCF→gsmSRF (applicable for direct gsmSCF-gsmSRF case only)

In this case the gsmSRF detects the erroneous situation, sends the Error "UnexpectedComponentSequence" and remains in the same state.

10.1.12.5 Operations gsmSRF→gsmSCF

In this case, an error occurs if the gsmSRF has already an established relationship with the gsmSCF and sends the CAP Operation AssistRequestInstructions. The gsmSCF detects the erroneous situation, informs SL and maintenance functions and returns the error parameter. On receiving the Error parameter, the gsmSRF FSM shall transit to the state "Idle" and releases the temporary connection.

10.1.12.6 Operations smsSSF →gsmSCF

If the CAP Operation is sent by the smsSSF in the context of an existing relationship, then the gsmSCF returns the Error parameter. On receiving the error, the smsSSF FSM shall transit to the state "Idle". The SGSN or VMSC shall handle the Short Message in accordance with the Default SMS Handling parameter of the valid CSI.

10.1.12.7 Operations gsmSCF→smsSSF

In this case the smsSSF detects the erroneous situation, sends the Error "UnexpectedComponentSequence" and remains in the same state.

10.1.12.8 Operations gprsSSF →gsmSCF

If the CAP Operation is sent by the gprsSSF in the context of an existing relationship, then the gsmSCF returns the Error parameter. On receiving the error, the gprsSSF FSM shall transit to the state "Idle". The SGSN shall handle the GPRS Session or PDP Context in accordance with the Default GPRS Handling parameter of the valid CSI.

10.1.12.9 Operations gsmSCF→gprsSSF

In this case the gprsSSF detects the erroneous situation, sends the Error "UnexpectedComponentSequence" and remains in the same state.

10.1.13 UnexpectedDataValue

10.1.13.1 General description

10.1.13.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this Error to indicate that it cannot execute the requested CAP Operation because a parameter has an unexpected data value.

Note that this error does not overlap with "ParameterOutOfRange".

EXAMPLE: startTime DateAndTime ::= -- value indicating January 32 1993, 12: 15: 01

 The responding entity does not expect this value and responds with Error "UnexpectedDataValue".

10.1.13.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.3 Operations gsmSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.4 Operations gsmSCF→gsmSRF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.5 Operations gsmSRF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.6 Operations smsSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.7 Operations gsmSCF→smsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.8 Operations gprsSSF →gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.13.9 Operations gsmSCF→gprsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14 UnexpectedParameter

10.1.14.1 General description

10.1.14.1.1 Error description

The gsmSCF, gsmSSF, gsmSRF, smsSSF or gprsSSF uses this Error to indicate that there is an error in the received CAP Operation argument. A valid but unexpected parameter was present in the CAP Operation argument. The presence of this parameter is not consistent with the presence of the other parameters. The responding entity cannot start the execution of the CAP Operation.

10.1.14.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.3 Operations gsmSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.4 Operations gsmSCF→gsmSRF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.5 Operations gsmSRF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.6 Operations smsSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.7 Operations gsmSCF→smsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.8 Operations gprsSSF →gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.14.9 Operations gsmSCF→gprsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.15 UnknownLegID

10.1.15.1 General description

10.1.15.1.1 Error description

The gsmSSF uses this error to indicate to the gsmSCF that a specific leg, indicated by the LegID parameter value in the CAP Operation, is unknown to the gsmSSF.

10.1.15.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.16 UnknownCSID

10.1.16.1 General description

10.1.16.1.1 Error description

The gsmSSF uses this error to indicate to the gsmSCF that a specific Call Segment, indicated by the callSegmentID parameter value in the CAP Operation, is unknown to the gsmSSF.

10.1.16.2 Operations gsmSCF→gsmSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.17 UnknownPDPID

10.1.17.1 General description

10.1.17.1.1 Error description

The gsmSCF or gprsSSF uses this error to indicate that a specific PDP Context, indicated by the PDPID parameter value in the CAP Operation argument, is unknown to the receiving entity.

10.1.17.2 Operations gprsSSF→gsmSCF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.1.17.3 Operations gsmSCF→gprsSSF

Refer to subclause 10.1.6 MissingParameter for the appropriate error procedures.

10.2 Entity related error procedures

The following subclauses specify the error handling procedures for the Functional Entity related errors. Since the error situations are not originated by the reception of a CAP Operation, the invoking entity is denoted here as the entity at which the error situation is detected. The responding entity is the entity which receives the error report.

The TC services used for reporting errors are described in clause 14.

10.2.1 Expiration of Tssf

10.2.1.1 General description

10.2.1.1.1 Error description

A timeout has occurred in the gsmSSF, gprsSSF, smsSSF or assisting gsmSSF on the response from the gsmSCF.

10.2.1.2 Procedures gsmSSF→gsmSCF

Procedure at the invoking entity (gsmSSF or assisting gsmSSF)

Timeout occurs in gsmSSF on Tssf

Precondition:	gsmSSF FSM state	"Waiting_for_Instructions"; or
	gsmSSF FSM state	"Waiting_for_end_of_User_Interaction"; or
	gsmSSF FSM state	"Waiting_for_end_of_Temporary_Connection".

Postcondition:	gsmSSF FSM state	"Idle".
----------------	------------------	---------

The gsmSSF shall abort the TS dialogue and shall transit to the state "Idle". The GMSC or VMSC shall handle the call in accordance with the Default Call Handling parameter of the valid CSI.

The assisting gsmSSF shall abort the TC dialogue and shall transit to the state "Idle". The assisting gsmSSF shall release the temporary connection.

10.2.1.3 Procedures gprsSSF→gsmSCF

Procedure at the invoking entity (gprsSSF)

Timeout occurs in gprsSSF on Tssf.

Precondition: gprsSSF FSM state "Waiting_for_Instructions".

Postcondition: gprsSSF FSM state "Idle".

The gprsSSF shall abort the TC dialogue and and the GPRS dialogue and shall transit to the state "Idle". The SGSN shall handle the PDP Context in accordance with the Default GPRS Handling parameter of the valid CSI.

10.2.1.4 Procedures smsSSF→gsmSCF

Procedure at the invoking entity (smsSSF)

Timeout occurs in smsSSF on Tssf.

Precondition: smsSSF FSM state "Waiting_for_Instructions".

Postcondition: smsSSF FSM state "Idle".

The smsSSF shall abort the TC dialogue and shall transit to the state "Idle". The MSC or SGSN shall handle the Short Message in accordance with the Default SMS Handling parameter of the valid CSI.

10.2.2 Expiration of Tsrfr

10.2.2.1 General Description

10.2.2.1.1 Error description

A timeout has occurred in the gsmSRF on the response from the gsmSCF. The procedures for handling this error are described in ETSI ETS 300 374-1 [24].

10.2.2.2 Procedures description

The procedures for handling this error are described in ETSI ETS 300 374-1 [24].

11 Detailed operation procedures for circuit switched call control

NOTE: The detailed operation procedures in the present clause which cross reference the gsmSCF FSMs for the pre- and postconditions are for information only.

NOTE: Where a parameter for a circuit switched call control Operation is marked OPTIONAL in ASN.1, the reader is referred to the conditions for presence for this parameter, specified in the respective Information Flow in 3GPP TS 23.078 [7].

11.1 ActivityTest procedure

11.1.1 General description

The gsmSCF uses this operation to check for the continued existence of a relationship between the gsmSCF and the gsmSSF, between the gsmSCF and the gsmSRF or between the gsmSCF and the assist gsmSSF. If the relationship is still in existence, then the receiving entity will respond. If the ActivityTest operation timer expires, then the gsmSCF will assume that the receiving entity has failed in some way and will take appropriate action.

11.1.1.1 Parameters

None.

11.1.2 Responding entity (gsmSSF, gsmSRF or assist gsmSSF)

11.1.2.1 Normal procedure

gsmSSF preconditions:

- (1) A relationship exists between the gsmSCF and the gsmSSF.
- (2) The gsmSSME FSM is in the state "Idle Management" or in the state "Non-call Associated Treatment".

gsmSRF or assist gsmSSF preconditions:

- (1) A relationship exists between the gsmSCF and the gsmSRF or assist gsmSSF.
- (2) The gsmSSME FSM is in the state "Idle Management".

gsmSSF postconditions:

- (1) The gsmSSME FSM stays in, or transits to the state "Non-call Associated Treatment".
- (2) If the TC Dialogue ID is active and there is a gsmSSF using the dialogue, then the gsmSSME FSM shall send a Return Result "ActivityTest" to the gsmSCF. If there are no other management activities (e.g. Call Gapping), then the gsmSSME FSM shall transit to the state "Idle Management". Otherwise, the gsmSSME FSM remains in the state "Non-call Associated Treatment".

If the TC Dialogue ID is not active, then the TC in the gsmSSF will issue a P-Abort. The gsmSSME FSM will in that case not receive the "ActivityTest" indication and thus will not be able to reply.

gsmSRF or assist gsmSSF postconditions:

- (1) The gsmSSME FSM transits to the state "Non-call Associated Treatment".
- (2) If the TC Dialogue ID is active and there is a gsmSRF or assist gsmSSF using the dialogue, then the gsmSSME FSM shall send a Return Result "ActivityTest" to the gsmSCF. The gsmSSME FSM then returns to the state "Idle Management".

If the TC Dialogue ID is not active, then the TC in the gsmSRF or assist gsmSSF will issue a P-Abort. The gsmSSME FSM will in that case not receive the ActivityTest indication and thus will not be able to reply.

11.1.2.2 Error handling

Operation related error handling is not applicable, due to class 3 operation.

11.2 ApplyCharging procedure

11.2.1 General description

The gsmSCF uses this operation for interacting with the gsmSSF function: "CSE control of call duration". The ApplyChargingReport operation provides the feedback from the gsmSSF to the gsmSCF.

The charging scenarios supported by this operation are those given in 3GPP TS 22.078 [3] for "CSE control of call duration".

11.2.1.1 Parameters

- aChBillingChargingCharacteristics:
This parameter specifies a list of parameters required for "CSE control of call duration":

The list may contain the following parameters:

- timeDurationCharging:

This list contains the following parameters:

- maxCallPeriodDuration:
This parameter specifies the period of time for which a call may progress before an ApplyChargingReport shall be sent to the gsmSCF.
- releaseIfdurationExceeded:
This parameter specifies the action to be taken at the gsmSSF when the duration specified above has been reached. If the parameter is present and the call duration has been reached, then the gsmSSF shall release the leg.
- audibleIndicator:
This parameter indicates to the gsmSSF that an audible indicator may be played to the served subscriber. This audible indicator may either be a predefined sequence of tones or a sequence defined by the gsmSCF. This parameter shall consist of either tone or burstlist as described below:
 - tone:
This parameter indicates that a warning tone shall be played when the pre-defined warning tone timer expires.
 - burstlist:
This parameter indicates that a variable sequence of tones shall be played when the gsmSCF-defined warning tone timer expires. This parameter may consist of the following parameters:
 - warningPeriod:
This parameter indicates the time, before the Max Call Period Duration timer expires when the playing of the burstlist shall start.
 - burst:
This parameter indicates the number of bursts that form the burstlist.
 - burstInterval:
This parameter indicates the time interval between the successive burst in the burstlist.

- toneInBurst:
This parameter indicates the number of tones to be played in each burst.
- toneDuration:
This parameter indicates the time duration that the tone shall be played for.
- toneInterval:
This parameter indicates the time interval between successive tones in a burst
- tariffSwitchInterval (for the CSE control of call duration):
This parameter indicates the time duration until the next tariff switch for the CSE control of call duration. The measurement of the elapsed tariff switch period shall start immediately after successful execution of this operation.
- partyToCharge:
This parameter indicates the party in the call.
- aChChargingAddress:
This parameter identifies the call party to which the ApplyCharging operation applies. That is the leg or srfConnection. If not present, then the ApplyCharging operation applies to the default legID 1.

This parameter is a choice of one of the following parameters:

- legID:
This parameter indicates that the "CSE control of call duration" is associated to the specified leg.
- or
- srfConnection:
This parameter indicates that the "CSE control of call duration" is associated to the Temporary Connection or to the connection to a gsmSRF. The connection is related to the specified Call Segment indicated by the srfConnection parameter.

11.2.2 Responding entity (gsmSSF)

11.2.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between the gsmSCF and the gsmSSF.
- (2) The gsmSSF FSM is in one of the following states:
"Waiting_for_Instructions"; or
"Waiting_for_end_of_User_Interaction"; or
"Waiting_for_end_of_Temporary_Connection"; or
"Monitoring".

gsmSSF postconditions:

- (1) No gsmSSF FSM state transition.

On receipt of this operation, the gsmSSF sets the charging data using the information elements included in the operation and acts accordingly.

If the aChChargingAddress indicates a legID, then:

The "CSE control of call duration" is associated to the specified leg. If Answer has not already been received on the incoming or outgoing connection (leg) to the Call Party, then the gsmSSF shall start monitoring for the Answer event upon receipt of the ApplyCharging operation. Upon subsequent detection of the Answer event on the outgoing connection charging is started. If the Answer event has occurred on the incoming or outgoing connection already when the ApplyCharging operation is received, then charging starts immediately.

Upon release of the incoming or outgoing connection to the Call Party any indication of Answer event occurred on the incoming or outgoing connection shall be cleared, i.e. set to "Answer event not occurred".

If the aChChargingAddress indicates a srfConnection, then:

The "CSE control of call duration" is associated to the Temporary Connection or to the connection to a gsmSRF. If Answer has not already been received on the Temporary Connection, then the gsmSSF will start monitoring for the Answer event upon receipt of the ApplyCharging operation; or, if the gsmSRF is not yet connected, then the gsmSSF will start monitoring for a connection to a gsmSRF. Upon subsequent detection of the Answer event on the Temporary Connection or the subsequent connection to a gsmSRF, charging shall be started. If the Answer event has been received from an Temporary Connection already or if the gsmSRF is already connected when the ApplyCharging operation is received, then charging shall start immediately.

Upon release of the Temporary Connection any indication of Answer event receipt on the outgoing connection shall be cleared i.e. set to "Answer event not received". Upon end of the connection to an gsmSRF, any indication of the connection to the gsmSRF shall be cleared.

11.2.2.2 Error handling

"TaskRefused": In addition to the generic error handling noted below, this error shall be indicated when:

- a previously received call period duration is pending for this leg or srfConnection;
- a tariffSwitchInterval for the "CSE control of call duration" is indicated when a previously received tariffSwitchInterval for the "CSE control of call duration" is pending for the Called Party, the Temporary Connection or the connection to a gsmSRF.

Generic error handling for the operation related errors is described in clause 10 and the TC services used for reporting operation errors are described in clause 14.

11.3 ApplyChargingReport procedure

11.3.1 General description

The gsmSSF uses this operation to report charging related information to the gsmSCF as requested by the gsmSCF using the "ApplyCharging" operation.

If Answer is detected by the gsmSSF, then timing of duration shall be started as requested by the ApplyCharging operation. It shall be started independently for a connection to a Called Party, a Temporary Connection and a gsmSRF connection.

The charging report shall be generated as specified in the 3GPP TS 23.078 [7].

The tariff switch mentioned in the following subclause are the tariff switch for "CSE control of call duration" for the connection to which the ApplyChargingReport procedure applies.

11.3.1.1 Parameters

- callResult:
This parameter provides the gsmSCF with the charging related information previously requested using the ApplyCharging operation. The "CallResult" is a list and may contain the following parameters:
 - timeDurationChargingResult:
This parameter is a list, and can contain the following parameters:
 - timeInformation:
This parameter is a choice of the following parameters:
 - timeIfNoTariffSwitch:
This parameter shall be present if no tariff switch has occurred since the reception of the first ApplyCharging operation for the connection to the Called Party, Temporary Connection or gsmSRF connection, otherwise it shall be absent.
If Answer was detected for the connection to the Called Party, the Temporary Connection or the gsmSRF connection, then the elapsed time since detection of Answer shall be reported. If answer was not detected, then it shall be set to "0".

- timeIfTariffSwitch:
This parameter shall be present if a tariff switch has occurred since the reception of the first ApplyCharging operation for the connection to the Called Party, Temporary Connection or gsmSRF connection, otherwise it shall be absent.
The parameter may contain the following information:
 - timeSinceLastTariffSwitch:
If Answer was detected for the connection to the Called Party, the Temporary Connection or the gsmSRF connection, then the elapsed time since detection of Answer or the last tariff switch (whichever of these events was last detected) shall be reported. If Answer was not detected, then it shall be set to "0".
 - TariffSwitchInterval (for the "CSE control of call duration"):
This parameter is present only if a tariff switch has occurred since the detection of Answer for the connection to the Called Party, the temporary connection or the gsmSRF connection in the reported call period.
The time interval between either the detection of the Answer event or the previous tariff switch (whichever of these events was last detected) and the last tariff switch is reported.
- partyToCharge:
The "partyToCharge" parameter as received in the related ApplyCharging operation or deduced from the default value. This parameter is used by the gsmSCF, to correlate the result to the request.
- aChChargingAddress:
The "aChChargingAddress" parameter as received in the related ApplyCharging operation or deduced from the default value This parameter is used by the gsmSCF, to correlate the result to the request.
- callActive:
This parameter indicates whether the leg, the Temporary Connection or the connection to a gsmSRF is still active or has been released.
- callReleasedAtTcpExpiry:
This parameter indicates that the gsmSSF has released the call and terminated the dialogue.
It shall be present when ApplyChargingReport is sent due to Tcp expiry and the gsmSSF has released the call (because ReleaseIfExceeded was present in ApplyCharging) and terminated the dialogue.
In all other instances, this parameter shall be absent.

11.3.2 Invoking entity (gsmSSF)

11.3.2.1 Normal procedure

gsmSSF preconditions:

- (1) A relationship exists between the gsmSSF and the gsmSCF.
- (2) A charging event has been detected that was requested by the gsmSCF via an ApplyCharging operation or a Called Party, Temporary Connection or gsmSRF disconnection event has occurred.

gsmSSF postconditions:

- (1) If release of the call has occurred because the allowed call duration has been reached, then:
 - All armed EDPs shall be disarmed;
 - ApplyChargingReport shall be sent to gsmSCF followed by any pending CallInformationReports, if applicable;
 - The gsmSSF FSM shall transit to the state "Idle".
- (2) If release of the call has occurred but not because the allowed call duration has been reached, then:
 - If there are any armed EDPs or pending reports, then the gsmSSF FSM shall remain in the same state, else;
 - The gsmSSF FSM shall transit to the state "Idle".

(3) else:

- no gsmSSF FSM state transition.

11.3.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services used for reporting operation errors are described in clause 14.

11.4 AssistRequestInstructions procedure

11.4.1 General description

This operation may be used by an assist gsmSSF or by a gsmSRF. The operation is sent to the gsmSCF when the assisting gsmSSF or gsmSRF receives an indication from an initiating gsmSSF indicating an assist procedure.

11.4.1.1 Parameters

- correlationID:
This parameter is used by the gsmSCF to associate the "AssistRequestInstructions" from the assisting gsmSSF or by a gsmSRF with the request from the initiating gsmSSF. The value of the "correlationID" may be extracted from the digits received from the initiating gsmSSF.
- iPSSPCapabilities:
Indicates which gsmSRF resources are attached, available and supported within:
 - the VMSC or GMSC where the gsmSSF resides; or
 - the IP where the gsmSRF resides.

11.4.2 Invoking entity (gsmSSF/gsmSRF)

11.4.2.1 Normal procedure

gsmSSF preconditions:

- (1) An assist indication is detected by the assisting gsmSSF.

gsmSSF postconditions:

- (1) The assisting gsmSSF FSM is in the state "Waiting_for_Instructions".

On receipt of an assist indication from the initiating gsmSSF, the assisting gsmSSF or gsmSRF shall ensure that the required resources are available to invoke an "AssistRequestInstructions" operation in the assist gsmSSF or gsmSRF and shall indicate to the initiating gsmSSF that the call is accepted. The "AssistRequestInstructions" operation is invoked by the assist gsmSSF or gsmSRF after the call, which initiated the assist indication, is accepted.

11.4.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.5 CallGap procedure

11.5.1 General description

The gsmSCF uses this operation to request the gsmSSF to reduce the rate at which specific service requests are sent to the gsmSCF. This operation may be sent only within a dialogue that has been opened by the gsmSSF by an InitialDP operation.

11.5.1.1 Parameters

- gapCriteria:
This parameter identifies the criteria for a call to be subject to call gapping. It is a choice of the following parameters: "basicGapCriteria" or "compoundGapCriteria":
 - basicGapCriteria:
This parameter consists of the following parameters:
 - calledAddressValue:
This parameter indicates that call gapping shall be applied when the leading digits of the dialled number of a call attempt match those specified in "calledAddressValue". The called address is the one received from the current call control.
 - gapOnService:
This parameter indicates that call gapping shall be applied when the Service Key of a Service initiation attempt matches the Service Key specified in "gapOnService".
 - calledAddressAndService:
This parameter indicates that call gapping shall be applied when the Service Key of a Service initiation attempt matches the Service Key defined in "calledAddressAndService" and the leading digits of the dialled number of a that call attempt match the dialled digits defined in "calledAddressAndService". The called address is the one received from the current call control.
 - callingAddressAndService:
This parameter indicates that call gapping shall be applied when the Service Key of a Service initiation attempt matches the Service Key defined in "callingAddressAndService" and the leading digits of the calling party number of that call attempt match the calling party number defined in "callingAddressAndService". In the case of call forwarding the calling address to be gapped shall be the redirecting number which is placed in the Initial DP operation.
 - compoundGapCriteria:
This parameter consists of the following parameters:
 - basicGapCriteria:
This parameter is as described above.
 - scfID:
This parameter identifies a gsmSCF. The scfID shall convey the necessary gsmSCF address information (e.g. Global Title) in the network to the requesting gsmSSF. Refer to ITU-T Recommendation Q.713 [42] "calling party address" parameter. The network operator shall decide about the actual mapping of this parameter on the used signalling system.
This parameter indicates the address of the gsmSCF which requests the call gapping.
If scfID is used in an operation that crosses an inter-network boundary, then its encoding must be understood in both networks; this requires bilateral agreement on the encoding. If this parameter is not available, then the call gapping is not dedicated to a specific gsmSCF.
This parameter is restricted to include a fixed GT address string.

NOTE: In the case where the GT addresses more than one SCP (e.g. a mated pair), then if one of these physical SCPs enters an overload conditions and issues a CallGap instruction, then the CallGap is applied to all these SCPs.
- gapIndicators:
This parameter indicates the call gapping characteristics.

- duration:
This parameter specifies the total time interval during which call gapping for the specified gap criteria will be active.

A duration of 0 indicates that call gapping shall be removed.

A duration of -2 indicates a network specific duration.

Other values indicate duration in seconds. A duration of -1 shall not be used.
- gapInterval:
This parameter specifies the minimum time between calls being allowed through.

An interval of 0 indicates that calls meeting the gap criteria shall not be rejected.

An interval of -1 indicates that all calls meeting the gap criteria shall be rejected.

Other values indicate the gap interval in milliseconds.
- controlType:
This parameter indicates the reason for activating call gapping.

The "controlType" value "sCPOverloaded" indicates that an automatic congestion detection and control mechanism in the SCP has detected a congestion situation.

The "controlType" value "manuallyInitiated" indicates that the service, the network management centre or service management centre, or both has detected a congestion situation, or any other situation that requires manually initiated controls.

NOTE: The controlType "manuallyInitiated" shall have priority over "sCPOverloaded" call gap. Also non-CAMEL controlled traffic control mechanism may apply to an exchange with the gsmSSF functionality. The non-CAMEL controlled traffic control may also have some influence to the CAMEL call. It is therefore recommended to take measures to co-ordinate several traffic control mechanisms. The non-CAMEL controlled traffic control and co-ordination of several traffic control mechanisms are out of the scope of CAP.
- gapTreatment:
This parameter indicates how calls that were stopped by the call gapping mechanism shall be treated.
- informationToSend:

This parameter indicates an announcement or a tone to be sent to the calling party. At the end of information sending, the call shall be released.
- inbandInfo:
This parameter specifies the inband information to be sent.
 - messageID:
This parameter indicates the message(s) to be sent, it can be one of the following:
 - elementaryMessageID:
This parameter indicates a single announcement.
 - duration:
This parameter indicates the maximum time duration in seconds that the message shall be played or repeated. A value of "0" indicates endless repetition.
- tone:
This parameter specifies a tone to be sent to the end-user.
 - toneID:
This parameter indicates the tone to be sent.
 - duration:
This parameter indicates the time duration in seconds of the tone. A value of "0" indicates infinite duration.

- releaseCause:
If the call shall be released, then this parameter indicates a specific cause value to be sent in the release message. See ETSI EN 300 356-1 [23].

11.5.2 Responding entity (gsmSSF)

11.5.2.1 Normal procedure

gsmSSF preconditions:

- (1) Call gapping for gapCriteria is not active; or
Call gapping for gapCriteria is active.
- (2) The gsmSSF FSM is in any state except "Idle" and except "Wait_For_Request".

gsmSSF postconditions:

- (1) The gsmSSME FSM is in the state "Active".
- (2) Call gapping for gapCriteria is activated; or
Call gapping for gapCriteria is renewed; or
Call gapping for gapCriteria is removed.
- (3) No gsmSSF FSM state transition.

If there is not already an existing gsmSSME FSM for the gap criteria and gsmSCFAddress provided, then a new gsmSSME FSM shall be created. If no gsmSCFAddress is provided, then this refers in general to the gsmSSME FSM without a gsmSCFAddress. This gsmSSME FSM enters the state "Active" and initialises call gapping for the specified CAMEL calls. The parameters "gapIndicators", "controlType", "gapTreatment" and "gsmSCFAddress" for the indicated gap criteria shall be set as provided by the "CallGap" operation.

In the case that both "manuallyInitiated" and "sCPOverloaded" call gapping are active for the same "gapCriteria", the manuallyInitiated call gapping shall prevail over the "sCPOverloaded" call gapping. More specifically, the following rules shall be applied in the gsmSSF to manage the priority of different control Types associated with the same "gapCriteria":

- If a gsmSSME FSM already exists for the "gapCriteria" and the gsmSCFAddress provided, then:
 - (1) If the (new) "controlType" equals an existing "controlType", then the new parameters (i.e., "gapIndicators" and "gapTreatment") shall overwrite the existing parameters.
 - (2) If the (new) "controlType" is different from the existing "controlType", then the new parameters (i.e., "controlType", "gapIndicators" and "gapTreatment") shall be appended to the appropriate gsmSSME FSM (in addition to the existing parameters). The gsmSSME FSM remains in the state "Active".

If the gsmSSF meets a TDP, then it shall check if call gapping was initiated for the same gsmSCF as the one currently assigned to this TDP or if call gapping exists with no provided gsmSCFAddress. If neither call gapping was initiated nor exists, then an "InitialDP" operation may be sent.

The gsmSSF shall check if call gapping was initiated either for the "serviceKey" or for the "calledAddressValue" assigned to this TDP. If this is not the case, then an "InitialDP" operation may be sent. If call gapping was initiated for "calledAddressAndService" or "callingAddressAndService" and the "serviceKey" matches, then a check on the "calledAddressValue" and "callingAddressValue" for active call gapping shall be performed. Otherwise, an "InitialDP" operation may be sent.

If a call to a controlled number matches one "gapCriteria" only, then the corresponding control is applied. If both "manuallyInitiated" call gapping and "sCPOverload" call gapping controls are active, then only the "manuallyInitiated" call gapping shall be applied.

If a call matches several active "basicGapCriteria", then the treatment as specified in the CallGap associated with the gapCriteria with the highest priority shall be applied, with the priority being from high to low:

1. calledAddressAndService or calledAddressValue;

2. callingAddressAndService;
3. gapOnService.

For example, a call with called number 123456 and ServiceKey = NP matches two CallGaps, one with gapCriteria "CalledAddressValue=123" and another one with gapCriteria "gapOnService=NP". Then the call shall be subject to the control of the service request CallGap with "CalledAddressValue=123".

In the case that multiple call gapping procedures are active with the same gap criteria, the "manuallyInitiated" call gapping shall prevail over the "sCPOverloaded" call gapping.

If a call to a controlled called number or from a controlled calling number matches several active "basicGapCriteria" of the same type (in this context "calledAddressAndService" and "calledAddressValue" are seen as one type), then only the "gapCriteria" associated with the longest called party number shall be used, and the corresponding control shall be applied. For example, the codes 1234 and 12345 are under control. Then the call with 123456 is subject to the control on 12345.

If a call to a controlled called number matches calledAddressAndService and calledAddressValue with the same number length, then calledAddressAndService has priority. Furthermore, if both "manuallyInitiated" and "sCPOverloaded" "controlTypes" are active for this "gapCriteria", then the "manuallyInitiated" control shall be applied.

If call gapping is performed on a call for a particular service and triggering of this service is allowed, then no other gap criteria shall be applied this service instance.

Active GapCriteria with assigned scfID shall have higher priority than the other GapCriteria. In the case that an entry with scfID matching the current call exists, all other criteria without scfID shall not be evaluated.

The matching entries with scfID shall be evaluated in accordance with the priority rules for the basic criteria listed above.

If call gapping shall be applied and there is no gap interval active, then an "InitialDP" operation may be sent including the "cGEncountered" parameter in accordance with the specified controlType. A new gap interval shall be initiated as indicated by "gapInterval".

If a gap interval is active, then no "InitialDP" operation shall be sent and the call shall be treated as defined by Default Call Handling of the valid CSI and "gapTreatment".

If the indicated duration equals "0", then the call gap process shall be stopped.

If call gapping proceeds, then the gsmSSME FSM remains in the state "Active". Otherwise, the gsmSSME FSM transits to the state "idle".

11.5.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.6 CallInformationReport procedure

11.6.1 General description

The gsmSSF uses this operation to send specific call information for a single call party to the gsmSCF as requested by the gsmSCF in previous "CallInformationRequest" operation. The report shall be sent at the end of a call party connection which is indicated by one of the events specified below.

11.6.1.1 Parameters

- requestedInformationList:
The gsmSSF sends the appropriate types and values to the gsmSCF in accordance with the requested information.

- legID:
This parameter indicates the party in the call for which the information has been collected. When absent, it indicates the "outgoing" leg, this can be a leg created by Connect, Continue or ContinueWithArgument.

11.6.2 Invoking entity (gsmSSF)

11.6.2.1 Normal procedure

gsmSSF preconditions:

- (1) The indicated or default party is released from the call or call setup towards the indicated or default party is terminated prematurely.
- (2) Requested call information has been collected.
- (3) "CallInformationReport" is pending due to a previously received "CallInformationRequest" operation.
- (4) A control or a monitor relationship exists between the gsmSCF and the gsmSSF.

gsmSSF postconditions:

- (1) If there are no armed EDPs or pending reports, then the gsmSSF FSM shall transit to the state "Idle"; otherwise, the gsmSSF FSM shall remain in the same state.

If the gsmSSF FSM executes a state transition caused by one of the following events, then:

- release for the indicated or default leg;
- abandon for the indicated or default leg;
- Called party Busy or Not Reachable for the indicated or default leg;
- gsmSSF No_Answer timer expiration for the indicated or default leg;
- route select failure for the indicated or default leg;
- release of call initiated by the gsmSCF (ReleaseCall),

and "CallInformationRequest" is pending for the indicated or default legs, then one "CallInformationReport" operation shall be sent to the gsmSCF, containing all information requested for that leg.

If a "CallInformationReport" has been sent to the gsmSCF, then no "CallInformationReport" is pending on that leg, i.e. if a further "CallInformationReport" on that leg is required, for example in the case of follow-on call, then this has to be explicitly requested by the gsmSCF.

If an event causing the "CallInformationReport" is also detected by an armed EDP-R, then immediately after "CallInformationReport" the corresponding "EventReportBCSM" shall be sent.

If an event causing the "CallInformationReport" is also detected by an armed EDP-N, then immediately before "CallInformationReport" the corresponding "EventReportBCSM" shall be sent.

11.6.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.7 CallInformationRequest procedure

11.7.1 General description

The gsmSCF uses this operation to request the gsmSSF to record specific information about a single call party and report it to the gsmSCF using the "CallInformationReport" operation.

11.7.1.1 Parameters

- requestedInformationTypeList:
This parameter specifies a list of specific items of information which is requested. The list may contain the following parameters:
 - callAttemptElapsedTime:
This parameter indicates the duration between the end of CAP processing of operations initiating call setup ("Connect", "Continue" or "ContinueWithArgument") and the received answer indication from the called party. If the callAttemptElapsedTime is requested for the calling party, then a value of "0" shall be reported by the gsmSSF.

In the case of unsuccessful call setup, the network event indicating the unsuccessful call setup stops the measurement of "callAttemptElapsedTime".
 - callStopTime:
This parameter indicates the time stamp when the connection is released.
 - callConnectedElapsedTime:
This parameter indicates the duration between the received answer indication from the called party and the release of that connection. For the calling party it indicates the duration between the sending of InitialDP and the release of that party.
 - releaseCause:
This parameter indicates the release cause for the call.
 - legID:
This parameter indicates the party in the call for which the information shall be collected and at the end of connection of which the report shall be sent. When absent, it shall apply to the "outgoing" leg, this can be a leg created by Connect, Continue or ContinueWithArgument.

11.7.2 Responding entity (gsmSSF)

11.7.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between gsmSSF and gsmSCF.
- (2) The gsmSSF FSM is in the state "Waiting_for_Instructions".

gsmSSF postconditions:

- (1) Requested call information is retained by the gsmSSF.
- (2) No gsmSSF FSM state transition.

The gsmSSF allocates a record for the indicated or default party and stores the requested information if already available and prepares the recording of information items, that will become available later like for example "callStopTimeValue".

Call information may be requested for any call party (identified by a legID).

11.7.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.8 Cancel procedure

11.8.1 General description

The gsmSCF uses this operation to request the gsmSRF or gsmSSF to cancel a correlated previous operation.

The Cancel operation may be used for the following purposes:

- (1) Cancel may be sent to the gsmSRF to cancel the "PlayAnnouncement" or the "PromptAndCollectUserInformation" operation. The cancellation of the operation is indicated via a respective error indication, "Canceled", to the invoking entity of the cancelled "PlayAnnouncement" or "PromptAndCollectUserInformation" operation.
- (2) Cancel may be sent to the gsmSSF to disarm all armed EDPs and cancel all pending reports. This enables the gsmSSF FSM to transit to the state "Idle". If Cancel is used for this purpose, then the Cancel operation does not specify any specific operation to be cancelled.

11.8.1.1 Parameters

- invokeID:
This parameter specifies which operation invocation shall be cancelled, i.e. PromptAndCollectUserInformation or PlayAnnouncement.
- callSegmentID:
This parameter specifies the Call Segment to which the cancellation of the user interaction operation shall apply.
- allRequests:
This parameter indicates that all armed EDP shall be disarmed and that any pending "ApplyChargingReport" and "CallInformationReport" shall be cancelled.

11.8.2 Responding entity (gsmSRF)

In the case of Cancel(invokeID), the gsmSRF is the responding entity.

11.8.2.1 Normal procedure

gsmSRF preconditions:

- (1) A PlayAnnouncement or PromptAndCollectUserInformation operation has been received and the gsmSRF FSM is in the state "User Interaction".

gsmSRF postconditions:

- (1) The execution of the PlayAnnouncement or PromptAndCollectUserInformation operation has been aborted and the gsmSRF remains in the state "User Interaction".

11.8.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.8.3 Responding entity (gsmSSF)

In the case of Cancel(allRequests), the gsmSSF is the responding entity.

11.8.3.1 Normal procedure

gsmSSF preconditions:

- (1) The gsmSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gsmSSF postconditions:

- (1) All armed EDPs are disarmed and all pending reports are cancelled.
- (2) If the gsmSSF FSM was in the state "Monitoring", then it shall transit to the state "Idle"; or
If the gsmSSF FSM was in the state "Waiting_for_Instructions", then it shall remain in that state.
A subsequent call-processing operation will result in a gsmSSF FSM state transition to the state "Idle". If the call is in the active state, then the gsmSSF shall treat it autonomously as a normal (non-CAMEL) call.

11.8.3.2 Error handling

Sending of return error on cancel is not applicable in the cancel "allRequests" case. Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.9 Connect procedure

11.9.1 General description

The gsmSCF uses this operation request the gsmSSF to perform the call processing actions to route a call to a specific destination.

In general all parameters which are provided in a Connect operation to the gsmSSF shall replace the corresponding signalling parameter in the CCF in O-BCSM, in accordance with ETSI ES 201 296 [21] and shall be used for subsequent call processing. The CCF of the T-BCSM shall send corresponding signalling parameters to new call leg without using them in subsequent call processing. Parameters which are not provided by the Connect operation shall retain their value (if already assigned) in the CCF for subsequent call processing.

11.9.1.1 Parameters

- destinationRoutingAddress:
This parameter contains the called party numbers towards which the call shall be routed.
- alertingPattern:
This parameter indicates the type of alerting to be applied. It is defined in 3GPP TS 29.002 [11].
- serviceInteractionIndicatorsTwo:
This parameter contains indicators to resolve interactions between CAMEL based services and network based services.
- callingPartysCategory:
This parameter indicates the type of calling party (e.g., operator, pay phone, ordinary subscriber).
- originalCalledPartyID:
If the call is forwarded by the gsmSCF, then this parameter carries the dialled digits.
- redirectingPartyID:
This parameter indicates the last directory number the call was redirected from.
- redirectionInformation:
This parameter contains forwarding related information, such as redirecting counter.
- genericNumbers:
This parameter allows the gsmSCF to set the Generic Number parameter used in the network. It is used for transfer of Additional Calling Party Number.
- suppressionOfAnnouncement:
This parameter indicates that announcements and tones which are played in the exchange at non-successful call set-up attempts shall be suppressed.
- oCSIApplicable:
This parameter indicates to the GMSC/gsmSSF or VMSC/gsmSSF that the Originating CAMEL Subscription Information, if present, shall be applied on the outgoing call leg created with the Connect operation. For the use of this parameter see 3GPP TS 23.078 [7].

- Carrier:
This parameter indicates carrier information. It consists of the carrier selection field followed by the Carrier ID information to be used by gsmSSF for routing a call to a carrier.

It contains the following embedded parameters:
 - carrierSelectionField:
This parameter indicates how the selected carrier is provided (e.g. pre-subscribed).
 - carrierID:
This parameter indicates the carrier to use for the call. It contains the digits of the carrier identification code.
- naOliInfo:
This parameter contains originating line information which identifies the charged party number type to the carrier.
- ChargeNumber:
This parameter contains the number that identifies the entity to be charged for the call. It identifies the chargeable number for the usage of a carrier (applicable on a call sent into a North American long distance carrier). For a definition of this parameter refer to ANSI T1.113-1995 [92].
- cug-Interlock:
This parameter uniquely identifies a CUG within a network.
- cug-OutgoingAccess:
This parameter indicates if the calling user has subscribed to the outgoing access inter-CUG accessibility subscription option.
- bor-InterrogationRequested:
This parameter indicates that Basic Optimal Routing is requested for the call.
- legToBeCreated:
This parameter indicates the leg to be connected.

11.9.2 Responding entity (gsmSSF)

11.9.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between the gsmSSF and the gsmSCF.
- (2) Basic call processing has been suspended at a DP.
- (3) The gsmSSF FSM is in the state "Waiting_for_Instructions".

gsmSSF postconditions:

- (1) The gsmSSF performs the call processing actions to route the call to the specified destination.
- (2) In the O-BCSM, call processing resumes at PIC Analyze_Information.
- (3) Tssf is stopped.

On receipt of this operation, the gsmSSF performs the following actions:

- If no EDPs have been armed and neither a CallInformationReport nor an ApplyChargingReport has been requested, then the gsmSSF transits to the state "Idle". Otherwise, the gsmSSF transits to the state "Monitoring".

The gsmSSF shall not perform any implicit arming or disarming of DPs.

Statistic counter(s) are not affected.

11.9.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.10 ConnectToResource procedure

11.10.1 General description

The gsmSCF uses this operation to connect a call segment from the gsmSSF to a specialized resource. After successful connection to the gsmSRF, the interaction with the parties in the call segment can take place. The gsmSSF relays all operations for the gsmSRF and all responses from the gsmSRF.

11.10.1.1 Parameters

- resourceAddress:
This parameter identifies the physical location of the gsmSRF. It may be one of the following parameters.
- iPRoutingAddress:
This parameter indicates the routeing address to set up a connection towards the gsmSRF.
- none:
This parameter indicates that the call segment shall be connected to a predefined gsmSRF.
- serviceInteractionIndicatorsTwo:
This parameter contains an indicator that is used for the control of the through connection between the calling party and the the gsmSRF.

NOTE The call segment to which the ConnectToResource applies does not have to contain a leg to the calling party.

The assisting gsmSSF shall always assume that Bothway Throughconnection is required, and hence will ignore this parameter if received.

- callSegmentId:
This parameter indicates the call segment to which the Connect To Resource procedure applies.

11.10.2 Responding entity (gsmSSF)

11.10.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between the gsmSCF and the gsmSSF.
- (2) The gsmSSF FSM is in the state "Waiting_for_Instructions".

gsmSSF postconditions:

- (1) The call segment is connected to the gsmSRF.
- (2) A control relationship between the gsmSCF and the gsmSRF is established.
- (3) The gsmSSF FSM transits to the state "Waiting_for_end_of_User_Interaction".
- (4) The gsmSSF loads Tssf with the default value and starts Tssf.

NOTE: The successful connection to the gsmSRF causes a gsmSRF FSM state transition from the state "Idle" to the state "Connected".

11.10.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.11 Continue procedure

11.11.1 General description

The gsmSCF uses this operation to request the gsmSSF to proceed with call processing at the DP at which it previously suspended call processing to await gsmSCF instructions. The gsmSSF continues call processing without substituting new data from the gsmSCF.

11.11.1.1 Parameters

None.

11.11.2 Responding entity (gsmSSF)

11.11.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between the gsmSSF and the gsmSCF.
- (2) Basic call processing has been suspended at any DP.
- (3) The gsmSSF FSM is in the state "Waiting_for_Instructions".

gsmSSF postconditions:

- (1) BCSM: If all required resumptions have been received, then basic call processing continues; otherwise, the only action is to decrement the resumption counter. For details refer to 3GPP TS 23.078 [7].
- (2) If all resumptions have not been received, then the gsmSSF FSM remains in the same state; or

If at least one EDP was armed, or a "CallInformationReport" or "ApplyChargingReport" was requested and no user interaction is ongoing, then the gsmSSF transits to the state "Monitoring"; or

If no EDPs were armed and neither the "CallInformationReport" nor the "ApplyChargingReport" was requested, then the gsmSSF transits to the state "Idle".

11.11.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.12 ContinueWithArgument Procedure

11.12.1 General description

The gsmSCF uses this operation to request the gsmSSF to proceed with call processing at the DP at which it previously suspended call processing to await gsmSCF instructions. It is also used to provide additional service related information to a User (Called Party or Calling Party) whilst the call processing proceeds.

In general all parameters which are provided in a ContinueWithArgument operation to the gsmSSF shall replace the corresponding signalling parameter in the CCF, in accordance with ETSI ES 201 296 [21]; shall be used for subsequent call processing. Parameters which are not provided by the ContinueWithArgument operation shall retain their value (if already assigned) in the CCF for subsequent call processing.

11.12.1.1 Parameters

- alertingPattern:
This parameter indicates the type of alerting to be applied. It is defined in 3GPP TS 29.002 [11].
- serviceInteractionIndicatorsTwo:
This parameter contains indicators that are used to resolve interactions between CAMEL based services and network based services.
- callingPartysCategory:
This parameter indicates the type of calling party (e.g., operator, pay phone, ordinary subscriber).
- genericNumbers:
This parameter allows the gsmSCF to set the Generic Number parameter used in the network. It is used for transfer of Additional Calling Party Number.
- suppressionOfAnnouncement:
This parameter indicates that announcements and tones which are played in the exchange at non-successful call set-up attempts shall be suppressed.
- carrier:
This parameter indicates carrier information. It consists of the carrier selection field followed by the Carrier ID information to be used by gsmSSF for routeing a call to a carrier.

It contains the following embedded parameter:

- carrierSelectionField:
This parameter indicates how the selected carrier is provided (e.g. pre-subscribed).
- carrierID:
This parameter indicates the carrier to use for the call. It contains the digits of the carrier identification code.
- naOliInfo:
This parameter contains originating line information which identifies the charged party number type to the carrier.
- chargeNumber:
This parameter contains the number that identifies the entity to be charged for the call. It identifies the chargeable number for the usage of a carrier (applicable on a call sent into a North American long distance carrier). For a definition of this parameter refer to ANSI T1.113-1995 [92].
- cug-Interlock:
This parameter uniquely identifies a CUG within a network.
- cug-OutgoingAccess:
This parameter indicates if the calling user has subscribed to the outgoing access inter-CUG accessibility subscription option.
- bor-InterrogationRequested:
This parameter indicates that Basic Optimal Routeing is requested for the call.
- suppress-O-CSI:
This parameter indicates that O-CSI shall be suppressed for the forwarding leg or deflecting leg.
- legID:
This parameter indicates the leg to which the ContinueWithArgument shall apply.
- suppress-D-CSI:
This parameter indicates that D-CSI shall be suppressed for the leg.
- suppress-N-CSI:
This parameter indicates that N-CSI shall be suppressed for the leg.
- suppressOutgoingCallBarring:
This parameter indicates that outgoing call barrings shall be suppressed for the leg.

11.12.2 Responding entity (gsmSSF)

11.12.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between the gsmSSF and the gsmSCF.
- (2) Basic call processing has been suspended at a DP.
- (3) The gsmSSF FSM is in the state "Waiting_for_Instructions".

gsmSSF postconditions:

- (1) BCSM: Basic call processing continues with modified information.
- (2) If there are armed EDPs or pending reports, then the gsmSSF FSM transits to the state "Monitoring"; otherwise the gsmSSF FSM transits to the state "Idle".

11.12.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.13 DisconnectForwardConnection procedure

11.13.1 General Description

The gsmSCF uses this operation for the following two purposes:

- (1) To clear a connection to a gsmSRF:

The operation is used to explicitly disconnect a connection to a resource (gsmSRF) established previously with a "ConnectToResource" or an "EstablishTemporaryConnection" operation. It is used for a forward disconnection from the gsmSSF.

An alternative solution is the backward disconnect from the gsmSRF, controlled by the "DisconnectFromIPForbidden" parameter in the "PlayAnnouncement" and "PromptAndCollectUserInformation" operations.

- (2) To clear a connection to an assisting gsmSSF:

The operation is sent to the non-assisting gsmSSF of a pair of gsmSSFs involved in an assist procedure. It is used to disconnect the temporary connection between the initiating gsmSSF and the assisting gsmSSF and the connection between the assisting gsmSSF and its associated gsmSRF.

The DisconnectForwardConnection operation shall not be used when the CallSegmentID is required.

11.13.1.1 Parameters

None.

11.13.2 Responding entity (gsmSSF)

11.13.2.1 Normal procedure

gsmSSF preconditions:

- (1) The basic call processing has been suspended at a DP. The gsmSSF FSM in the initiating gsmSSF is in the state "Waiting_for_end_of_User_Interaction" or in the state "Waiting_for_end_of_Temporary_Connection".

gsmSSF postconditions:

- (1) The connection to the gsmSRF or assisting gsmSSF is released.
- (2) The gsmSSF FSM is in the state "Waiting_for_Instructions".

The receipt of "DisconnectForwardConnection" results in a disconnection of the assisting gsmSSF or the PE containing the gsmSRF from the call. It does not result in a release of the connection between the gsmSSF and the end-user.

On receipt of this operation, the gsmSSF shall perform the following actions:

- The initiating gsmSSF releases the connection to the assisting gsmSSF or the gsmSRF.
- The gsmSSF loads Tssf with the default and restarts Tssf.
- The gsmSSF FSM transits to the state "Waiting_for_Instructions".

NOTE: The successful disconnection to the gsmSRF causes a gsmSRF FSM state transition to the state "Idle". A current order (e.g. "PlayAnnouncement" or "PromptAndCollectUserInformation") is cancelled and any queued order (e.g. "PlayAnnouncement" or "PromptAndCollectUserInformation") is discarded.

11.13.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.14 DisconnectForwardConnectionWithArgument procedure

11.14.1 General Description

The gsmSCF uses this operation to disconnect a connection to a resource (gsmSRF) established previously with a "ConnectToResource" or an "EstablishTemporaryConnection" operation.

11.14.1.1 Parameters

- callSegmentID:
This parameter indicates the Call Segment to be disconnected from the resource.

11.14.2 Responding entity (gsmSSF)

11.14.2.1 Normal procedure

gsmSSF preconditions:

- (1) The basic call processing has been suspended at a DP. The CS_gsmSSF FSM in the initiating gsmSSF is in the state "Waiting_for_end_of_User_Interaction" or in the state "Waiting_for_end_of_Temporary_Connection".

gsmSSF postconditions:

- (1) The connection to the gsmSRF or assisting gsmSSF is released.
- (2) The CS_gsmSSF FSM transits to the state "Waiting_for_Instructions".

The receipt of "DisconnectForwardConnectionWithArgument" results in disconnecting the PE containing the gsmSRF from the specified Call Segment. It does not result in a release of the connection between the gsmSSF and the end-user.

On receipt of this operation, the gsmSSF shall perform the following actions:

- The initiating gsmSSF releases the connection to the assisting gsmSSF or the gsmSRF.
- The gsmSSF loads Tssf with the default value and restarts Tssf.

- The gsmSSF FSM transits to the state "Waiting_for_Instructions".

NOTE: The successful disconnection from the gsmSRF causes the gsmSRF to transit to the state "Idle". A current order (e.g. "PlayAnnouncement" or "PromptAndCollectUserInformation") is cancelled and any queued order (e.g. "PlayAnnouncement" or "PromptAndCollectUserInformation") is discarded.

11.14.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.15 DisconnectLeg procedure

11.15.1 General Description

The gsmSCF uses this operation to request the gsmSSF to release a specific leg associated with the call. Any other leg(s) not specified in the Disconnect Leg operation are retained.

11.15.1.1 Parameters

- legToBeReleased:
This parameter indicates the call leg to be released.
- releaseCause:
This parameter may be used by the MSC for generating specific tones to the party to be released or to fill in the "cause" parameter in the release message.

11.15.2 Responding entity (gsmSSF)

11.15.2.1 Normal procedure

gsmSSF preconditions:

- 1) A control relationship exists between the gsmSCF and the gsmSSF.

gsmSSF postconditions:

- 1) The gsmSSF performs the call processing actions to release the indicated party.
- 2) Any armed EDPs on that leg shall be disarmed; any pending reports for that leg shall be sent to the gsmSCF.
- 3) If the released leg was the last leg within the Call Segment, then the CS_gsmSSF FSM for that Call Segment returns to the state "Idle".
- 4) If the leg was the last leg within the call, then the CSA_gsmSSF FSM returns to the state "Idle".
- 5) If the CS_gsmSSF FSM for the Call Segment concerned has not returned to the state "Idle", then it transits to the state "Waiting_for_Instructions". The remaining BCSM instances within the Call Segment shall transit to the O_Mid_Call DP or to the T_Mid_Call DP, unless already suspended at a DP. Note that no Mid_Call EDP will be reported for this case.
- 6) A Return Result shall be sent to the gsmSCF immediately after successful execution of this operation.

11.15.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10, and the TC services which are used for reporting operation errors are described in clause 14.

11.16 EntityReleased procedure

11.16.1 General Description

The CSA uses this operation to inform the gsmSCF about the release of a logical entity (Call Segment or BCSM) caused by exception or errors. The CSA shall use this operation if this information can not be conveyed in a TC_ABORT or TC_END TC message because the TC dialogue has to be maintained for other logical entities (Call Segment or BCSM) in this CSA which are not affected by this error or exception. The CSA shall not use this operation if the last Call Segment in the CSA was released.

11.16.1.1 Parameters

This operation consists of a choice of the following parameters:

- CSFailure:
Indicates that a Call Segment was released. It contains of the following parameters:
 - callSegmentID:
This parameter identifies the released Call Segment.
 - cause:
This parameter indicates the cause for releasing this Call Segment. The cause may be used by the gsmSCF to decide the further handling of the call.
- BCSMFailure:
This parameter indicates that a leg was released. It consists of the following parameters:
 - legID:
This parameter identifies the released leg.
 - cause:
This parameter indicates the cause for releasing this BCSM. The cause may be used by the gsmSCF to decide the further handling of the call.

11.16.2 Invoking entity (gsmSSF)

11.16.2.1 Normal procedure

CSA_gsmSSF preconditions:

- (1) Any CSA_gsmSSF FSM state except "Idle".

CSA_gsmSSF postconditions:

- (1) gsmSSF postconditions:
 - (1) If the released entity was a BCSM (leg), then only the appropriate resources are released. If the released entity was a Call Segment, then the related CS_gsmSSF FSM returns to the state "Idle".

11.16.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10, and the TC services which are used for reporting operation errors are described in clause 14.

11.17 EstablishTemporaryConnection procedure

11.17.1 General Description

The gsmSCF uses this operation to create a connection between an initiating gsmSSF and an assisting gsmSSF as part of a service assist procedure. It can also be used to create a connection between a gsmSSF and a gsmSRF, for the case where the gsmSRF exists in a separately addressable PE.

The assistingSSPIPRoutingAddress shall contain routeing digits, a correlationID and an scfID when a temporary connection shall be established between PLMNs and no bilateral agreement exists between the involved network operators to transfer correlationID and SCFiD as separate parameters.

11.17.1.1 Parameters

- assistingSSPIPRoutingAddress:
This parameter indicates the destination address of the gsmSRF for assist procedure. The "assistingSSPIPRoutingAddress" may contain embedded within it, a "correlationID" and "scfID", but only if "correlationID" and "scfID" are not specified separately.
- correlationID:
This parameter is used by the gsmSCF to associate the "AssistRequestInstructions" from the assisting gsmSSF (or the gsmSRF) with the Request from the initiating gsmSSF. The "correlationID" shall be used only if the correlation id is not embedded in the "assistingSSPIPRoutingAddress". The network operator has to decide about the actual mapping of this parameter on the used signalling system.
- scfID:
This parameter contains the gsmSCF identifier and enables the assisting gsmSSF to identify which gsmSCF the AssistRequestInstructions shall be sent to.
The "scfID" shall be used only if the gsmSCF id is not embedded in the "assistingSSPIPRoutingAddress". The network operator has to decide about the actual mapping of this parameter on the used signalling system.
When ScfID is used in an operation, which may cross an internetwork boundary, its encoding must be understood in both networks; this requires bilateral agreement on the encoding.
- serviceInteractionIndicatorsTwo:
This parameter contains an indicator that is used for the control of the through connection to the Calling Party.
- Carrier:
This parameter contains carrier information. It consists of the carrier selection field followed by the Carrier ID information to be used by gsmSSF for routeing a call to a carrier.

It contains the following embedded parameter:

- carrierSelectionField:
This parameter indicates how the selected carrier is provided (e.g. pre-subscribed).
- carrierID:
This parameter indicates the carrier to use for the call. It contains the digits of the carrier identification code.
- naOliInfo:
This parameter contains originating line information which identifies the charged party number type to the carrier.
- chargeNumber:
This parameter contains the number that identifies the entity to be charged for the call. It identifies the chargeable number for the usage of a carrier (applicable on a call sent into a North American long distance carrier). For a definition of this parameter refer to ANSI T1.113-1995 [92].
- callSegmentID:
This parameter indicates the Call Segment to which the temporary connection shall be established.

11.17.2 Responding entity (gsmSSF)

11.17.2.1 Normal procedure

gsmSSF preconditions:

- (1) The gsmSSF FSM is in the state "Waiting_for_Instructions".
- (2) The gsmSSF is not an assisting gsmSSF.

gsmSSF postconditions:

- (1) The gsmSSF performs the call processing actions to route the call to the assisting gsmSSF or gsmSRF in accordance with the "assistingSSPIPRoutingAddress" requested by the gsmSCF.
- (2) The gsmSSF FSM transits to the state "Waiting_for_end_of_Temporary_Connection".
- (3) The gsmSSF loads Tssf with the default value and starts Tssf.

On receipt of this operation , the gsmSSF shall perform the following actions:

- Route the call to assisting gsmSSF or gsmSRF using "assistingSSPIPRoutingAddress";

11.17.2.2 Error handling

Until the connection setup has been accepted (refer to ITU-T Recommendation Q.71 [41]) by the assisting gsmSSF or the gsmSRF, all received failure indications from the network on the ETC establishment shall be reported to the gsmSCF as EstablishTemporaryConnection error ETCFailed (e.g., busy, congestion).

NOTE The operation timer for EstablishTemporaryConnection shall be longer than the maximum allowed time for the signalling procedures to accept the connection.

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.18 EventNotificationCharging procedure

11.18.1 General Description

The gsmSSF uses this operation to report to the gsmSCF the occurrence of a specific charging event type as requested by the gsmSCF using the RequestNotificationChargingEvent operation. As several charging events may occur during a connection configuration, the possibility exists for the EventNotificationCharging operation to be invoked on multiple occasions. For each connection configuration, the EventNotificationCharging operation may be used several times.

11.18.1.1 Parameters

- eventSpecificChargingInformation:

This parameter contains charging related information specific to the event. This parameter is a choice between four lists of information.

- InfoChargeUnitsTotal:

This parameter contains the accumulated charge units which have been monitored. This parameter is a list of the following information.

- reportingConditionUnits:

This parameter contains the reporting condition. This parameter shall be the copy of the corresponding parameter received in Request Notification Charging. This parameter shall contain one of the following information.

- endOfConnection:

This parameter indicates that the charge units is being reported at the end of connection.

- thresholdCounterValue:

This parameter indicates that the charge units is being reported after exceeding or equalling a threshold.

- chargeUnits:

This parameter contains the accumulated charge units which have been monitored.

- infoChargeUnitsComponents:

In this parameter the gsmSSF shall specify the charge units of various charging components in the PLMN which have been monitored. This is a list of charging components, accumulated charge units of those components along with the corresponding reporting condition.

- reportingConditionUnits:

This parameter contains the reporting condition. This parameter shall be the copy of the corresponding parameter received in Request Notification Charging. This parameter shall contain one of the following information.

- endOfConnection:

This parameter indicates that the charge units is being reported at the end of connection.

- thresholdCounterValue:

This parameter indicates that the charge units is being reported after exceeding or equalling the threshold.

- chargeUnits:

This parameter contains the accumulated charge units which have been monitored.

- monitorComponent:

This parameter contains the specific charging components which have been monitored.

- chargesSpecificPLMN:

This component specifies the PLMN specific charges at the serving MSC determined by the operator based on radio resource utilization, roaming etc.

- chargesTransitNetwork:

This component specifies charges at the serving MSC to be levied for utilization of transit network resources.

- infoEvalueTotal:

This parameter contains the net applicable e-values which have been monitored. This is a list of following information.

- reportingConditionEvalue:

This parameter contains the reporting condition. This parameter shall be the copy of the corresponding parameter received in Request Notification Charging. This parameter shall contain the following information.

- occurrenceOfEvent:

This parameter shall be the event when the e-values are reported. This may be the first determination of applicable evalues for the call while routing or the occurrence of network tariff switch and the determination of new applicable evalues.

- evaluesAndTsw:

This parameter contains the e-values and the duration for which these e-values are valid. This parameter is a list of following information.

- cAIElements:

This is a set of Charge Advice Information elements, as defined in 3GPP TS 22.024 [2].

- tariffSwitchInterval:

This parameter contains the time duration for which the above mentioned e-values are valid.

- infoEvaluateComponents:

In this parameter the gsmSSF shall specify the evalues and the tariff switch interval of various charging components in the PLMN which have been monitored. It contains the list of the charging component, applicable e-values for those components which have been monitored along with the corresponding reporting condition.

- reportingConditionEvaluate:

This parameter contains the reporting condition. This parameter shall be the copy of the corresponding parameter received in Request Notification Charging. This parameter shall contain the following information.

- occurrenceOfEvent:

This parameter shall be the event when the e-values are reported. This may be the first determination of applicable evalues for the call while routing or the occurrence of network tariff switch and the determination of new applicable evalues.

- evaluesAndTsw:

This parameter contains the e-values and the duration for which these e-values are valid. This parameter is a list of following information.

- cAIElements:

This is a set of Charge Advice Information elements, as defined in 3GPP TS 22.024 [2].

- tariffSwitchInterval:

This parameter contains the time duration for which the above mentioned e-values are valid.

- monitorComponent:

This parameter contains the specific charging component which has been monitored.

- chargesSpecificPLMN:

This component specifies the PLMN specific charges at the serving MSC determined by the operator based on radio resource utilization, roaming etc.

- chargesTransitNetwork:

This component specifies charges at the serving MSC to be levied for utilization of transit network resources.

- legID:

This parameter indicates the leg ID for which the event is monitored.

11.18.2 Invoking entity (gsmSSF)

11.18.2.1 Normal procedure

gsmSSF preconditions:

- (1) A relationship exist between the gsmSCF and the gsmSSF.
- (2) A charging event has been detected that is requested by the gsmSCF.

gsmSSF postconditions:

- (1) No gsmSSF FSM state transition.

The gsmSSF FSM is in any state except "Idle". This operation is invoked if a charging event has been detected that is requested by the gsmSCF.

11.18.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.19 EventReportBCSM procedure

11.19.1 General description

The gsmSSF uses this operation to notify the gsmSCF of a call related event previously requested by the gsmSCF in a "RequestReportBCSMEvent" operation.

11.19.1.1 Parameters

- eventTypeBCSM:
This parameter specifies the type of event that is reported.
- eventSpecificInformationBCSM:
This parameter indicates the call related information specific to the event.

For Route_Select_Failure it shall contain the "FailureCause", if available.

For O_Busy it shall contain the "BusyCause", if available.

- If the busy event is triggered by an ISUP release message, then the BusyCause is a copy of the ISUP release cause, for example: Subscriber absent, 20 or User busy, 17.
- If the busy event is triggered by a MAP error, for example: Absent subscriber, received from the HLR, then the MAP cause is mapped to the corresponding ISUP release cause.

NOTE 1: If no BusyCause is received, then the gsmSCF shall assume busy.

For T_Busy it may contain the following parameters, if available.

- CallForwarded:
This parameter indicates that the busy event is triggered by call forwarding at the GMSC or VMSC.
- ForwardingDestinationNumber:
This parameter indicates the forwarding destination.
- RouteNotPermitted:
This parameter indicates that the busy event is triggered because call forwarding was not invoked in this GMSC due to the rules of Basic Optimal Routeing.
- BusyCause:
 - If the busy event is triggered by an ISUP release message, then the BusyCause is a copy of the ISUP release cause, for example: Subscriber absent, 20 or User busy, 17.
 - If the busy event is triggered by a MAP error, for example: Absent subscriber, received from the HLR, then the MAP cause is mapped to the corresponding ISUP release cause.

- If the busy event is triggered by call forwarding invocation in the GMSC or VMSC, then the BusyCause will refer to the type of the call forwarding service in accordance with the mapping table in 3GPP TS 23.078 [7].

NOTE 2: If no BusyCause is received, then the gsmSCF shall assume busy.

- If the busy event is triggered by call forwarding at the GMSC, then the BusyCause reflects the forwarding reason (Subscriber Absent, 20 or User busy, 17). The eventSpecificInformationBCSM shall in that case also contain the CallForwarded indication.

For O_No_Answer it shall be empty.

For T_No_Answer it may contain the CallForwarded indication and the ForwardingDestinationNumber.

- If the No_Answer event is triggered by an ISUP release message or expiry of the CAMEL timer TNRY, then the eventSpecificInformationBCSM shall be empty.
- If the No_Answer event is triggered by call forwarding at the GMSC or VMSC, then the eventSpecificInformationBCSM shall contain the CallForwarded indication and the ForwardingDestinationNumber.

For O_Answer or T_Answer it shall contain the following information, if available:

- The destination address for the call;
- The OR indicator, in the case that the call was subject to Basic Optimal Routeing, as specified in 3GPP TS 23.079 [8];
- The forwarding indicator, in the case that the Call Forwarding Supplementary Service was invoked.

For O_Mid_Call and T_Mid_Call it shall contain the detected digit string, in accordance with the criterion defined in the RequestReportBCSMEvent operation.

For Call_Accepted, O_Term_Seized, O_Change_Of_Position and T_Change_Of_Position it shall contain the following information:

- locationInformation:
This parameter indicates the location of the MS.

For O_Disconnect and T_Disconnect it shall contain the "releaseCause", if available.

For O_Abandon" it may contain the following parameter, if available.

- routeNotPermitted:
This parameter indicates that the O-Abandon event is triggered because call set up shall not be invoked in this MSC due to the rules of Basic Optimal Routeing.
- legID:
This parameters indicates the party in the call for which the event is reported. The gsmSSF shall use the option "receivingSideID" only.
- receivingSideID:
If not included, then the following defaults are assumed:
"legID" = 1 for the events O_Abandon and T_Abandon,
"legID" = 2 for the events Route_Select_Failure, O_Busy, O_No_Answer, O_Answer, T_Busy, O_Term_Seized, Call_Accepted, T_No_Answer and T_Answer.

The "legID" parameter shall always be included for the events O_Disconnect and T_Disconnect.

- miscCallInfo:
This parameter indicates Detection Point (DP) related information.
- messageType:
This parameter indicates whether the message is a request, i.e. resulting from a "RequestReportBCSMEvent"

with monitorMode = interrupted, or a notification, i.e. resulting from a "RequestReportBCSMEvent" with "monitorMode" = "notifyAndContinue".

11.19.2 Invoking entity (gsmSSF)

11.19.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship or a monitoring relationship exists between the gsmSSF and the gsmSCF.
- (2) For the O_Disconnect DP, T_Disconnect DP, O_Answer DP and T_Answer DP, the gsmSSF FSM is in the state "Monitoring" or in the state "Waiting_for_Instructions". For the O_Abandon DP and T_Abandon DP, the gsmSSF FSM is in any state, except "Idle".
- (3) The BCSM proceeds to an EDP that is armed.

gsmSSF postconditions:

- (1) If the message type was notification and there are still armed EDPs or pending reports, then the gsmSSF FSM stays in the state "Monitoring".
- (2) If the message type was notification and there are neither any armed EDPs nor pending reports, then the gsmSSF FSM transits to the state "Idle".
- (3) If the message type was request, then the gsmSSF FSM transits to the state "Waiting_for_Instructions". Call processing is interrupted.

11.19.2.2 Error handling

If the message type is "request" and the Tssf timer expires, then the gsmSSF shall abort the TC dialogue and shall instruct the MSC to treat the call in accordance with the Default Call Handling of the valid CSI.

Operation related error handling is not applicable, due to class 4 operation.

11.20 FurnishChargingInformation procedure

11.20.1 General description

The gsmSCF uses this operation to send charging related information to a logical call record. This logical call record is CAMEL specific. The first FurnishChargingInformation operation of a call leg leads to the generation of a logical call record. The handling of subsequent FurnishChargingInformation operations for a call leg depends on the presence and value of the append free format data parameter in the FurnishChargingInformation operation. For details see 3GPP TS 23.078 [7].

If a FurnishChargingInformation operation is received for the called party when the gsmSSF FSM is in the state "Monitoring", or is suspended in one of the following DPs, then the charging information shall be included in the logical call record for the leg that has been or is to be established:

- Collected_Info;
- Analysed_Information;
- O_Answer;
- Terminating_Attempt_Authorised; or
- T_Answer.

If a FurnishChargingInformation operation is received for the called party when the gsmSSF is suspended in any other DP, then the charging information shall be included in the logical call record created for the last failed or disconnected called party.

11.20.1.1 Parameters

- fCIBillingChargingCharacteristics:
This parameter contains the following parameters;
- fCIBCCAMELsequence1:
This parameter contains the following parameters;
 - freeFormatData:
This parameter contains free-format billing and/or charging characteristics;
 - partyToCharge:
This parameter indicates the party to bill and/or charge;
 - appendFreeFormatData:
This parameter indicates whether previous FurnishChargingInformation free format data is appended or overwritten. Refer to 3GPP TS 23.078 [7] for details on the usage of this parameter.

11.20.2 Responding entity (gsmSSF)

11.20.2.1 Normal procedure

gsmSSF preconditions:

- (1) The gsmSSF FSM is in the state
"Waiting_for_Instructions"; or
"Waiting_for_end_of_User_Interaction"; or
"Waiting_for_end_of_Temporary_Connection"; or
"Monitoring".

gsmSSF postconditions:

- (1) No gsmSSF FSM state transition.

On receipt of this operation the gsmSSF performs actions to create the Logical call record if necessary, and writes the free-format information carried in the operation into the Logical call record. A FurnishChargingInformation operation will create a Logical Call Data Record (CDR) if such a record does not already exist for the indicated leg. Refer to sect. 11.26.1 for the handling in the case of successive FurnishChargingInformation operations for a call leg.

The Logical CDRs will be associated for a given call into one or more physical CDRs, as specified in 3GPP TS 32.205 [13].

A logical call record is output to a physical CDR when a disconnection event is propagated to the call leg associated with it, or when a Connect operation to create a connection to a Follow-on Called Party is received. Successive FurnishChargingInformation operations indicating the calling leg (leg1) may overwrite data from previously received FurnishChargingInformation operation(s) indicating that leg during the entire call or call attempt. Successive FurnishChargingInformation operations indicating a called leg (leg2 or higher) may overwrite any previously received data from FurnishChargingInformation operation(s) indicating that called leg until the called leg representing that particular called party number is released from the call. When a new called party is created as a result of a follow-on call, and a FurnishChargingInformation operation indicating the called leg is received, then a new Logical call record shall be created for that call leg, for that portion of the call. From then on, any subsequent FurnishChargingInformation operations for that called party may overwrite the data from any previous FurnishChargingInformation operation(s) for the called leg presenting that particular called party number. Logical call records that have been output already are not affected.

No Logical call record is output at the end of a user interaction.

11.20.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.21 InitialDP procedure

11.21.1 General description

The gsmSSF uses this operation after detection of a TDP-R in the BCSM, to request the gsmSCF for instructions to complete the call.

11.21.1.1 Parameters

- serviceKey:
This parameter indicates to the gsmSCF the requested IN service. It is used to address the required application/SLP within the gsmSCF; this parameter is not for SCP addressing.
- calledPartyNumber:
This parameter contains the number used to identify the called party in the forward direction, i.e. see ETSI EN 300 356-1 [23]. This parameter shall be sent only in the Mobile Terminating, Mobile Forwarding and mobile originating on unsuccessful TDP cases.
- callingPartyNumber:
This parameter carries the calling party number to identify the calling party or the origin of the call. See ETSI EN 300 356-1 [23] Calling Party Number signalling information.
- callingPartysCategory:
Indicates the type of calling party (e.g. operator, pay phone, ordinary subscriber). See ETSI EN 300 356-1 [23] Calling Party Category signalling information.
- locationNumber:
This parameter is used to convey the geographical area address for mobility services, see ITU-T Recommendation Q.762 [44]. It is used when "callingPartyNumber" does not contain any information about the geographical location of the calling party (e.g., origin dependent routing when the calling party is a mobile subscriber).
- originalCalledPartyID:
If the call has met call forwarding on the route to the gsmSSF, then this parameter carries the dialled digits. Refer to EN 300 356-1[23] Original Called Number signalling information.
- highlayerCompatibility:
This parameter indicates the type of the high layer compatibility, which will be used to determine the ISDN - teleservice of a connected ISDN terminal. The highlayerCompatibility can also be transported by ISUP (e.g. within the ATP (see ITU-T Recommendation Q.763 [45]) parameter).
- additionalCallingPartyNumber:
The calling party number provided by the access signalling system of the calling user, e.g. provided by a PBX.
- bearerCapability:
This parameter indicates the type of the bearer capability connection or the transmission medium requirements to the user. It is a network option to select which of the two parameters to be used:
 - bearerCap:
This parameter contains the value of the ISUP User Service Information parameter.

The parameter "bearerCapability" shall be included in the "InitialDP" operation only in the case the ISUP User Service Information parameter is available at the gsmSSF.

If User Service Information and User Service Information Prime are available at the gsmSSF, then the "bearerCap" shall contain the value of the User Service Information Prime parameter.
- eventTypeBCSM:
This parameter indicates the armed BCSM DP event, resulting in the "InitialDP" operation.
- redirectingPartyID:
This parameter indicates the last directory number the call was redirected from.

- redirectionInformation:
This parameter contains forwarding related information, such as redirecting counter.
See ITU-T Recommendation Q.763 [45] Redirection Information signalling information.
- iPSSPCapabilities:
This parameter indicates which gsmSRF resources supported within the VMSC or GMSC the gsmSSF resides in are attached and available.
- serviceInteractionIndicatorsTwo:
This parameter contains indicators that are used to resolve interactions between CAMEL based services and network based services.
- iMSI:
This parameter contains the IMSI of the mobile subscriber for which the service is invoked.
- subscriberState:
This parameter indicates the the state of the mobile subscriber for which the service is invoked. The possible states are "busy", "idle" and "not reachable".
- locationInformation:
This parameter indicates the location of the MS and the age of the information defining the location.
- ext-BasicServiceCode:
This parameter indicates the Basic Service Code.
- callReferenceNumber:
This parameter contains the call reference number assigned to the call by the CCF.
- mscAddress:
This parameter contains the mscId assigned to the MSC.
- gmscAddress:
This parameter contains the gmscId assigned to the GMSC.
- calledPartyBCDNumber:
This parameter contains the number used to identify the called party in the forward direction. It may also include service selection information, including * and # characters.
- time&Timezone:
This parameter contains the time that the gsmSSF was triggered, and the time zone that the invoking gsmSSF resides in.
- callForwardingSS-Pending:
This parameter indicates that a forwarded-to-number was received and that the call will be forwarded due to the Call Forwarding supplementary service in the GMSC or in the VMSC, unless otherwise instructed by the gsmSCF.
- carrier:
This parameter contains carrier information. It consists of the carrier selection field followed by the Carrier ID information associated with the calling subscriber of a mobile originating call, the called subscriber of a mobile terminating call or the forwarding subscriber of a mobile forwarded call.

It contains the following embedded parameter:

- carrierSelectionField:
This parameter indicates how the selected carrier is provided (e.g. pre-subscribed).
- carrierID:
This parameter indicates the carrier to use for the call. It contains the digits of the carrier identification code.
- cug-Index:
This parameter is used to select a CUG for an outgoing call at the user, or to indicate an incoming CUG call to the user.
- cug-Interlock:
This parameter uniquely identifies a CUG within a network.

- **cug-OutgoingAccess:**
This parameter indicates if the calling user has subscribed to the outgoing access inter-CUG accessibility subscription option.
- **cGEncountered:**
This parameter indicates the type of call gapping the related call has been subjected to, if any.
- **cause:**
This parameter indicates the release cause which triggered the event:

For Route_Select_Failure" it shall contain the "FailureCause", if available.

For T_Busy it may contain the following parameters, if available.
 - If the busy event is triggered by an ISUP release message, then the BusyCause shall a copy of the ISUP release cause, for example: Subscriber absent, 20 or User busy, 17.
 - If the busy event is triggered by a MAP error, for example: Absent subscriber, received from the HLR, then the MAP cause is mapped to the corresponding ISUP release cause.
 - If the busy event is triggered by call forwarding invocation in the GMSC or VMSC, then the BusyCause shall refer to the type of the call forwarding service in accordance with the mapping table in 3GPP TS 23.078 [7].
- **forwardingDestinationNumber:**
This parameter contains the forwarding destination.
- **ms-Classmark2:**
This parameter contains the MS Classmark 2 of the mobile subscriber for which the service is invoked.
- **iMEI:**
This parameter contains the IMEI (with software version) of the mobile subscriber for which the service is invoked.
- **supportedCamelPhases:**
This parameter indicates the CAMEL Phases supported in the GMSC or VMSC which sends this operation.
- **supportedCamel4Subsets:**
This parameter contains the supported CAMEL phase 4 subsets.

11.21.2 Invoking entity (gsmSSF)

11.21.2.1 Normal procedure

gsmSSF preconditions:

- (1) An event fulfilling the criteria for the DP being executed has been detected.
- (2) Call gapping and SS7 overload are not in effect for the call.

gsmSSF postconditions:

- (1) If the DP was armed as a TDP-R and trigger conditions, if present, are fulfilled, then a control relationship between the gsmSCF and the gsmSSF is established. The gsmSSF transits to the State "Waiting_for_Instructions".

The address of the gsmSCF shall be fetched from the valid CSI. The gsmSSF shall provide all available parameters to the gsmSCF.

If no triggering takes place, because trigger conditions were not fulfilled, then the gsmSSF shall proceed with call handling without CAMEL Service.

The gsmSSF application timer Tssf is loaded and started when the gsmSSF sends "InitialDP" for requesting instructions from the gsmSCF. It is used to prevent excessive call suspension time.

11.21.2.2 Error handling

If the gsmSCF is not accessible, then the call proceeds in accordance with the Default Call Handling parameter in the CSI.

When Tssf expires, then the gsmSSF shall abort the interaction with the gsmSCF by means of an abort to TC and shall call continue the call in accordance with the Default Call Handling parameter in the valid CSI.

If the calling party abandons after the sending of "InitialDP", then the gsmSSF shall abort the interaction with the gsmSCF by means of an abort to TC.

NOTE TC will wait until the first response message from the gsmSCF has been received before it sends an abort to the gsmSCF (see also clause 14).

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.22 InitiateCallAttempt procedure

11.22.1 General Description

The gsmSCF uses this operation to request the gsmSSF to create a new call leg to one call party using the address information provided by the gsmSCF (e.g. wake-up call). The gsmSCF shall subsequently arm O_Answer as an EDP-R and the call failure events (Route_Select_Failure, O_Busy and O_No_Answer) as EDP-Rs and/or EDP-Ns, in order to enable the gsmSCF to treat this call appropriately when any of these events is encountered. InitiateCallAttempt can also be used to create an additional call party in a new Call Segment within an existing Call Segment Association.

11.22.1.1 Parameters

11.22.1.1.1 Argument Parameters

- destinationRouteingAddress:
This parameter contains the called party number towards which the call shall be routed.
- callingPartyNumber:
This parameter identifies which number shall be regarded as the calling party for the created call.
- legToBeCreated:
This parameter indicates the LegID to be assigned to the newly created party.
- newCallSegment:
This parameter indicates the Call Segment ID to be assigned to the newly created Call Segment.
- callReferenceNumber:
This parameter contains the call reference number assigned to the call by the gsmSCF.
- gsmSCFAddress:
This parameter indicates the address of the gsmSCF initiating the operation.
- suppress-T-CSI:
This parameter indicates that the T-CSI for the served subscriber shall be suppressed for this call leg.

11.22.1.1.2 Result Parameters

- supportedCamelPhases:
This parameter indicates the CAMEL Phases supported in the gsmSSF which receives this operation.
- supportedCamel4Subsets:
This parameter contains the supported CAMEL phase 4 subsets.

11.22.2 Responding entity (gsmSSF)

11.22.2.1 Normal procedure

gsmSSF preconditions:

None.

gsmSSF postconditions:

- 1) A new O-BCSM has been created; call processing is suspended.
- 2) A Return Result is sent to the gsmSCF.
- 3) The CS_gsmSSF FSMtransits from the state "Idle" to the state "Waiting_for_Instructions".

All subsequent operations are treated in accordance with their normal procedures.

11.22.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10, and the TC services which are used for reporting operation errors are described in clause 14.

11.23 MoveLeg procedure

11.23.1 General Description

The gsmSCF uses this operation to request the gsmSSF to move the leg from its current Call Segment to the initial Call Segment (CS ID = 1).

11.23.1.1 Parameters

- legIDToMove:
This parameter indicates the leg that shall be moved.

11.23.2 Responding entity (gsmSSF)

11.23.2.1 Normal procedure

gsmSSF preconditions:

- 1) A control relationship exists between the gsmSCF and the gsmSSF.
- 2) The corresponding BCSM is in the alerting, active or mid-call phase.
- 3) The CS_gsmSSF FSM for each Call Segment involved is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gsmSSF postconditions:

- 1) The gsmSSF performs the appropriate call processing actions.
- 2) The CS_gsmSSF FSM for the initial Call Segment transits to the state "Waiting_for_Instructions". The BCSM instances within the initial Call Segment transit to the O_Mid_Call DP or to the T_Mid_Call DP, if not already suspended. Note that no Mid_Call EDP will be reported for this case.
- 3) The CS_gsmSSF process for the source Call Segment is terminated.
- 4) A Return Result is sent to the gsmSCF immediately after successful execution of this operation.

11.23.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10, and the TC services which are used for reporting operation errors are described in clause 14.

11.24 PlayAnnouncement procedure

11.24.1 General description

The gsmSCF uses this operation for inband interaction with a CS user.

11.24.1.1 Parameters

- informationToSend:
This parameter indicates an announcement a tone to be sent to the end-user by the gsmSRF.
- inbandInfo:
This parameter specifies the inband information to be sent.
 - messageID:
This parameter indicates the message(s) to be sen; this may be one of the following:
 - elementaryMessageID:
This parameter indicates a single announcement.
 - text:
This parameter indicates a text to be sent. The text shall be transformed to inband information (speech) by the gsmSRF. This parameter consists of two parameters; messageContent and attributes. The attributes of the text parameter may consist of items such as language.
 - elementaryMessageIDs:
This parameter specifies a sequence of announcements.
 - variableMessage:
This parameter specifies an announcement with one or more variable parts.
 - numberOfRepetitions:
This parameter indicates the maximum number of times the message shall be sent to the end-user.
 - duration:
This parameter indicates the maximum time duration in seconds that the message shall be played or repeated. A value of "0" indicates endless repetition.
 - interval:
This parameter indicates the time interval in seconds between successive messages, i.e. the time between the end of the announcement and the start of the repetition of this announcement. This parameter may be used only when "numberOfRepetitions" is > 1.
- tone:
This parameter specifies a tone to be sent to the end-user.
 - toneID:
This parameter indicates the tone to be sent.
 - duration:
This parameter indicates the time duration in seconds of the tone to be sent. A value of "0" indicates infinite duration.
- disconnectFromIPForbidden:
This parameter indicates whether the gsmSRF may initiate a disconnection from the gsmSSF after the interaction has been completed.

If this parameter is TRUE, then the gsmSRF shall not initiate a disconnection. If this parameter is FALSE, then the gsmSRF may initiate a disconnection.

- requestAnnouncementCompleteNotification:
This parameter indicates whether or not a "SpecializedResourceReport" shall be sent to the gsmSCF when all information has been sent.
- requestAnnouncementStartedNotification:
This parameter indicates whether or not a "SpecializedResourceReport" shall be sent to the gsmSCF when the first announcement or tone has started.
- callSegmentID:
This parameter indicates the Call Segment to which the user interaction shall apply.

11.24.2 Responding entity (gsmSRF)

11.24.2.1 Normal procedure

gsmSRF preconditions:

- (1) The SRSM-FSM is in the state "Connected"; if the gsmSRF received previously an operation from the gsmSCF, then the SRSM-FSM is in the state "User Interaction".
- (2) When the first announcement or tone has started and "RequestAnnouncementStartedNotification" is TRUE, then the SRSM shall send a "SpecializedResourceReport" operation, containing the "FirstAnnouncementStarted" parameter, to the gsmSCF.

gsmSRF postconditions:

- (1) The gsmSRF sends the information to the user as indicated by "informationToSend".
- (2) The SRSM-FSM transits to the state "User Interaction", or remains in the same state.
- (3) If all information has been sent and "RequestAnnouncementCompleteNotification" is TRUE, then the SRSM shall send a "SpecializedResourceReport" operation, containing the "AllAnnouncementsComplete" parameter, to the gsmSCF.
- (4) If all information has been sent and "disconnectFromIPForbidden" is FALSE, then the SRSM disconnects the gsmSRF from the user.

The announcement sent to the end-user is ended in the following conditions:

- if neither "duration" nor "numberOfRepetitions" is specified, then the network specific announcement ending conditions shall apply; or
- if "numberOfRepetitions" is specified, when all repetitions have been sent; or
- if duration is specified, when the duration has expired. The announcement is repeated until this condition is met; or
- if "duration" and "numberOfRepetitions" is specified, when one of both conditions is satisfied (whichever comes first).

11.24.2.2 Error handling

If a Cancel operation is received before or during the processing of the operation, then that operation shall be cancelled immediately and the error "Canceled" shall be reported to the gsmSCF.

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.25 PlayTone procedure

11.25.1 General description

The gsmSCF uses this operation to instruct the gsmSSF to play tones to a leg or a Call Segment using the MSC's tone generator.

If a Call Segment is indicated, then the tones shall be played to all active legs in that Call Segment. If a leg is indicated, then the tones shall be played to that leg only.

11.25.1.1 Parameters

- legOrCallSegment:
This parameter indicates the leg or Call Segment to which the PlayTone operation shall apply.
- bursts:
This parameter indicates the variable sequence of tones to be played and consists of the following parameters:
 - numberOfBursts:
This parameter indicates the number of bursts that form the burstlist.
 - burstInterval:
This parameter indicates the time interval between successive bursts in a sequence of bursts.
 - numberOfTonesInBurst:
This parameter indicates the number of tones to be played in each burst.
 - toneDuration:
This parameter indicates the time duration of a single tone in a burst.
 - toneInterval:
This parameter indicates the time interval between successive tones in a burst.

11.25.2 Responding entity (gsmSSF)

11.25.2.1 Normal procedure

gsmSSF preconditions:

The gsmSSF FSM is in one of the following states:

"Monitoring"; or
"Waiting_for_Instructions"; or
"Waiting_for_End_of_Temporary_Connection".

If a Call Segment is indicated, then at least one of the legs in that Call Segment is in the active phase.

gsmSSF postconditions:

- (1) No gsmSSF FSM state transition.

11.25.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services used for reporting operation errors are described in clause 14.

11.26 PromptAndCollectUserInformation procedure

11.26.1 General description

The gsmSCF uses this operation to interact with a call party in order to collect information.

11.26.1.1 Parameters

- collectedInfo:

- collectedDigits:

- minimumNbOfDigits:

This parameter specifies the minimum number of valid digits to be collected.

- maximumNbOfDigits:

This parameter specifies the maximum number of valid digits to be collected. The following applies: "maximumNbOfDigits" \geq "minimumNbOfDigits".

- endOfReplyDigit:

This parameter indicates the digit string used to signal the end of input.

In the case that the "maximumNbOfDigits" > "minimumNbOfDigits" the following applies:

If "endOfReplyDigit" is not present, then the end of input is indicated:

- when the inter-digit timer expires; or
- when the number of valid digits received equals the "maximumNbOfDigits".

If "endOfReplyDigit" is present, then the end of input is indicated:

- when the inter-digit timer expires; or
- when the end of reply digit is received; or
- when the number of valid digits received equals the "maximumNbOfDigits".

When the end of input is reached, the collected digits are sent from gsmSRF to the gsmSCF, including the "endOfReplyDigit" if received by the gsmSRF. In the case the number of valid digits received is less than the "minimumNbOfDigits" when the inter-digit timer expires or when the end of reply digit is received, the input is considered to be erroneous.

- cancelDigit:

This parameter indicates the cancel digit string that may be entered by the user to request a retry. All digits already received by the gsmSRF are discarded and the PromptAndCollectUserInformation procedure is performed again, thus e.g. the same announcement to request user information is given to the user and information is collected. If this parameter is not present, then the user is not able to request a retry.

- startDigit:

This parameter indicates the start digit string that indicates the start of the valid digits to be collected. The digits that are received by the gsmSRF before this start digit is received, are discarded and are not considered to be valid. The start digit string itself is considered to be valid digits.

If this parameter is not present, then all received digits are considered to be valid.

When the end of input is reached, the collected digits are sent from gsmSRF to the gsmSCF, including the "startDigit" if received by the gsmSRF.

- firstDigitTimeOut:

If this parameter is present, then the first digit shall be received by the gsmSRF before first-digit timer expiration. If the first digit is not received before first-digit timer expiration, then the input is considered to be erroneous. After receipt of the first valid or invalid input digit, the first-digit timer shall be stopped.

If this parameter is not present, then the gsmSRF shall use a default value for the first-digit timer.

If "startDigit" is present, then the first-digit timer shall be stopped after the start digit is received.

- interDigitTimeOut:

If this parameter is present, then any subsequent valid or invalid digit shall be received by the gsmSRF before the inter-digit timer expires. As a result of receiving a digit, the inter-digit timer is reset and restarted.

If a subsequent valid or invalid digit is not received before the inter-digit timer expires and the number of received valid digits is less than "minimumNbOfDigits", then the input is considered to be unsuccessful.

If a subsequent valid or invalid digit is not received before the inter-digit timer expires and the number of received valid digits is greater than "minimumNbOfDigits" and smaller than or equal to "maximumNbOfDigits", then the input is considered to be successful.

If the "interDigitTimeOut" is not present, then the gsmSRF shall use a default value for the inter-digit timer.

- errorTreatment:

This parameter defines what specific action shall be taken by the gsmSRF in the event of error conditions occurring.

- interruptableAnnInd:

If this parameter is TRUE, then the announcement shall be interrupted after the first valid or invalid digit is received by the gsmSRF. If the announcement is interrupted, then a first-digit timer shall not apply anymore. However, if the announcement has not been interrupted, then the first-digit timer shall be started after the announcement has been finished.

If this parameter is FALSE, then the announcement shall not be interrupted after the first digit is received by the gsmSRF. The received digits during the announcement are discarded and considered to be invalid. All other specified parameters ("minimumNbOfDigits", "maximumNbOfDigits", "endOfReplyDigit", etc.) do not apply before the announcement has been finished. The first-digit timer shall be started after the announcement has been finished.

- voiceInformation:

If this parameter is FALSE, then all valid or invalid digits shall be entered by DTMF.

If this parameter is TRUE, then the calling user is required to provide all valid or invalid information by speech. The gsmSRF shall perform voice recognition and shall translate the provided information into digits. The end of reply digit(s), if required, shall be provided by speech.

- voiceBack:

If this parameter is FALSE, then no voice back information shall be given by the gsmSRF.

If this parameter is TRUE, then the valid input digits received by the gsmSRF shall be announced back to the calling user immediately after the end of input is received. The invalid input digits will not be announced back to the calling user. The end of reply digit(s) shall not be voiced back to the calling user.

- disconnectFromIPForbidden:

This parameter indicates whether the gsmSRF may initiate a disconnection from the gsmSSF after the interaction has been completed.

If this parameter is TRUE, then the gsmSRF shall not initiate a disconnection. If this parameter is FALSE, then the gsmSRF may initiate a disconnection.

- informationToSend:

This parameter indicates an announcement or tone to be sent to the end-user by the gsmSRF.

- inbandInfo:
This parameter specifies the inband information to be sent.
- messageID:
This parameter indicates the message(s) to be sent;, this may be one of the following:
 - elementaryMessageID:
This parameter indicates a single announcement.
 - text:
This parameter indicates a text to be sent. The text shall be transformed to inband information (speech) by the gsmSRF. The attributes of text may consist of items such as language.
 - elementaryMessageIDs:
This parameter specifies a sequence of announcements.
 - variableMessage:
This parameter specifies an announcement with one or more variable parts.
- numberOfRepetitions:
This parameter indicates the maximum number of times the message shall be sent to the end-user.
- duration:
This parameter indicates the maximum time duration in seconds that the message shall be played or repeated. A value of "0" indicates endless repetition.
- interval:
This parameter indicates the time interval between successive messages, i.e. the time between the end of the announcement and the start of the repetition of this announcement. This parameter may be used only when "numberOfRepetitions " > 1.
- tone:
This parameter specifies a tone to be sent to the end-user.
 - toneID:
This parameter indicates the tone to be sent.
 - duration:
This parameter indicates the time duration in seconds of the tone. A value of "0" indicates infinite duration.
- requestAnnouncementStartedNotification:
This parameter indicate whether or not a "SpecializedResourceReport" shall be sent to the gsmSCF when the first announcement or tone has started.
- callSegmentID:
This parameter indicates the Call Segment to which the user interaction shall apply.

Result Parameter:

- digitsResponse:
This parameter contains the information collected from the end-user.

11.26.2 Responding entity (gsmSRF)

11.26.2.1 Normal procedure

gsmSRF preconditions:

- (1) The SRSM-FSM is in the state "Connected"; if the gsmSRF received previously an operation from the gsmSCF, then the SRSM-FSM is in the state "User Interaction".
- (2) If the first announcement or tone has started and "RequestAnnouncementStartedNotification" is TRUE, then the SRSM sends a "SpecializedResourceReport" operation, containing the "FirstAnnouncementStarted" parameter, to the gsmSCF.

gsmSRF postconditions:

- (1) The gsmSRF has sent the information to the end-user as indicated by "informationToSend".
- (2) The collected information from the end-user is sent to the gsmSCF as RETURN RESULT of the "PromptAndCollectUserInformation".
- (3) If the "disconnectFromIPForbidden" is FALSE, then the gsmSRF initiates a bearer channel disconnect to the gsmSSF and the SRSF FSM transits to the state "Idle".
- (4) Otherwise, the SRSF FSM transits to the state "User Interaction" or remains in the same state.

The announcement sent to the end-user is ended in the following conditions:

- if neither "duration" nor "numberOfRepetitions" is specified, then the network specific announcement ending conditions shall apply; or
- if "numberOfRepetitions" is specified, when all repetitions have been sent; or
- if duration is specified, when the duration has expired. The announcement is repeated until this condition is met; or
- if "duration" and "numberOfRepetitions" is specified, when one of both conditions is satisfied (whichever comes first).

If the parameter "interruptableAnnInd" is not FALSE and the end-user has responded with a digit during the sending of the announcement, then the above conditions are overruled. In that case, the announcement shall be ended immediately.

The parameter "errorTreatment" specifies how the gsmSRF shall treat an error. The value "reportErrorToSCF" means that the error shall be reported to the gsmSCF by means of Return Error with "ImproperCallerResponse". The value "help" indicates that no error shall be reported to gsmSCF but assistance shall be given to the end-user in the form of a network dependent default announcement (which may be dependent on the context, i.e. the sent message). The value "repeatPrompt" indicates that no error shall be reported to the gsmSCF but the prompt shall be repeated to the end-user. The error handling procedures related to "help" and "repeatPrompt" shall be done only once per "PromptAndCollectUserInformation" operation.

NOTE Note on processing "endOfInput"

The receipt of any "endOfInput" condition (e.g. endOfReplyDigit, cancelDigit, firstDigitTimeout, interDigitTimeout) terminates immediately the ongoing input. In other words when e.g. an endOfReplyDigit is received, then the receipt of a subsequent cancelDigit will not be processed anymore.

11.26.2.2 Error handling

If a Cancel operation is received before or during the processing of the operation, then the operation shall be cancelled immediately and the error "Canceled" shall be reported to the gsmSCF.

Generic error handling for the operation related errors are described in clause 10, the TC services which are used for reporting operation errors are described in clause 14.

If any of the parameter restrictions are violated (e.g. "minimumNbOfDigits" > "maximumNbOfDigits"), then an operation error has occurred.

11.27 ReleaseCall procedure

11.27.1 General description

The gsmSCF uses this operation to tear down a call at any phase. This operation may not be sent to an assisting gsmSSF.

11.27.1.1 Parameters

- releaseCause:
This parameter gives an indication to the gsmSSF about the reason of releasing this specific call. This may be used by gsmSSF for generating specific tones to the different parties in the call or to fill in the "cause" in the release message.

11.27.2 Responding entity (gsmSSF)

11.27.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between gsmSCF and gsmSSF.
- (2) The gsmSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gsmSSF postconditions:

- (1) The gsmSSF FSM transits to the state "Idle" after sending any pending "CallInformationReport" or "ApplyChargingReport". All armed EDPs shall be disarmed. All connections and resources related to the call shall be released.

11.27.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.28 RequestNotificationChargingEvent procedure

11.28.1 General Description

The gsmSCF uses this operation to instruct the gsmSSF how to manage the charging events which are received from other Functional Entities (FEs) not under the control of the ServiceLogic (SL) instance.

11.28.1.1 Parameters

- eventTypeChargingPLMN:

This parameter indicates the charging event type, ie. the type of charging information to be monitored by the serving MSC and to be reported to the CSE. This parameter is a choice between four lists of information.

- chargeUnitsTotal:

This parameter indicates that the accumulated charge units for the call are to be monitored. This parameter is a list of following information.

- reportingConditionUnits:

This parameter indicates the charge units are to be reported on the occurrence of one of the following conditions:

- endOfConnection:

This parameter indicates that the charge units are to be reported at the disconnection of the established connection configuration.

- thresholdCountervalue:

This parameter indicates that the charge units are to be reported when the accumulated charge units equal or exceed the specified threshold value.

- chargeUnitsComponents:

In this parameter CSE shall specify the units of various charging components in the PLMN to be monitored. It contains the list of the charging components to be monitored and the corresponding reporting condition.

- monitorComponent:

This parameter specifies the PLMN charging component to be monitored. It is a choice of

- chargesSpecificPLMN:

This component specifies the PLMN specific charges at the serving MSC determined by the operator based on radio resource utilization, roaming etc.

- chargesTransitNetwork:

This component specifies charges at the serving MSC to be levied for utilization of transit network resources.

- reportingConditionUnits:

This parameter indicates the charge units are to be reported on the occurrence of one of the following conditions

- endOfConnection:

This parameter indicates that the charge units are to be reported at the disconnection of the established connection configuration.

- thresholdCountervalue:

This parameter indicates that the charge units are to be reported when the accumulated charge units equal or exceed the specified threshold value.

- evaluesTotal:

This parameter specifies that the net applicable evalues are to be reported. The serving MSC shall report the currently applicable e-values. It shall also notify the new e-values to CSE whenever applicable e-values change due to event such as tariff switch. It is a list of following information.

- reportingConditionEvalue:

This parameter indicates that the e-values are to be reported on the occurrence of the following conditions

- occurrenceOfEvent:

This indicates that the e-values are to be reported at the occurrence of charging event which may be the first determination of applicable evalues for the call while routing or the occurrence of network tariff switch and the determination of new applicable e-values.

- evaluesComponents:

In this parameter CSE may request the evalues corresponding to the various charging components in the PLMN. It contains the list of the component to be monitored and the corresponding reporting condition

- monitorComponent:

This parameter specifies the PLMN charging component to be monitored. It is a choice of following parameters:

- chargesSpecificPLMN:

This component specifies the PLMN specific charges at the serving MSC determined by the operator based on radio resource utilization, roaming etc.

- chargesTransitNetwork:

This component specifies charges at the serving MSC to be levied for utilization of transit network resources.

- reportingConditionEvaluate:

This parameter indicates that the e-values are to be reported on the occurrence of any of the following conditions:

- occurrenceOfEvent:

This indicates that the e-values are to be reported at the occurrence of charging event which may be the first determination of applicable e-values for the call while routing or the occurrence of network tariff switch and the determination of new applicable e-values.

- monitorMode:

This parameter indicates the monitorMode applicable for the corresponding "eventTypeCharging" parameter. Monitor mode may be "notifyAndContinue" or "transparent".

The monitor mode "interrupted" is not applicable for the RequestNotificationCharging operation.

- legID:

This parameter indicates the leg ID for which the charging event is to be monitored.

11.28.2 Responding entity (gsmSSF)

11.28.2.1 Normal procedure

gsmSSF preconditions:

- (1) The gsmSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".
- (2) A relationship exist between the gsmSCF and the gsmSSF.

gsmSSF postconditions:

- (1) No gsmSSF FSM state transition.

On receipt of this operation the gsmSSF performs actions to cope with the interactions concerning the charging according the information elements included in the operation.

In the case that multiple RequestNotificationChargingEvent operations are received for the same connection configuration with the same "eventTypeCharging" and "legID", only the latest received "monitorMode" shall apply.

11.28.2.2 Error handling

Generic error handling for the operation related errors is described in Clause 10 and the TCAP services which are used for reporting operation errors are described in Clause 14.

11.29 RequestReportBCSMEvent procedure

11.29.1 General description

The gsmSCF uses this operation to request the gsmSSF to monitor for a call-related event (e.g., BCSM events such as O_Busy or O_No_Answer) and to send a notification to the gsmSCF when the event is detected.

The monitoring of more than one event may be requested with a single "RequestReportBCSMEvent" operation, but each of these requested events will be reported in a separate "EventReportBCSM" operation.

NOTE: If the RequestReportBCSMEvent requests arming of the current DP from which the call processing was suspended, then the next occurrence of the DP encountered during BCSM processing will be detected (i.e. not the current one from which the call was suspended).

The DP arming principle is as follows:

- The DPs O_Disconnect and T_Disconnect can be armed for any or all legs depending on the direction for which events have to be captured. As an example, the O_Disconnect DP can be armed for leg1 and leg2; in this case, if a release request is received from the A-party, then it will be detected by the O_Disconnect DP armed for leg1, while a release request from the B-party will be detected by the O_Disconnect DP armed for leg2.
- The O_Abandon DP can be armed only for leg1 in the O-BCSM and the T_Abandon DP can be armed only for leg1 in the T-BCSM.

Table 11-1: DP Arming Table for O-BCSM:

O-BCSM	leg1	Not leg 1	Default_leg_ID
O_Term_Seized DP	-	X	2
Route_Select_Failure DP	-	X	2
O_Busy DP	-	X	2
O_No_Answer DP	-	X	2
O_Answer DP	-	X	2
O_Disconnect DP	X	X	- (note 1)
O_Abandon DP	X	-	1
O_Mid_Call	X	-	1
O_Change_Of_Position	X	-	1
Note 1: The "legID" parameter shall be included Nomenclature: X = Arming Applicable - = Arming not Applicable			

Table 11-2: DP Arming Table for T-BCSM:

T-BCSM	leg2	leg1	Default Leg ID
Call_Accepted DP	X	-	2
T_Busy DP	X	-	2
T_No_Answer DP	X	-	2
T_Answer DP	X	-	2
T_Disconnect DP	X	X	- (note 1)
T_Abandon DP	-	X (note 2)	1
T_Mid_Call	X	-	2
T_Change_Of_Position	X	-	2
Note 1: The "legID" parameter shall be included Note 2: T_Abandon can be armed for leg1 only. Nomenclature: X = Arming Applicable - = Arming not Applicable			

11.29.1.1 Parameters

- bcsmEvents:
This parameter specifies the event or events of which a report is requested.
- eventTypeBCSM:
This parameter specifies the type of event of which a report is requested.
- monitorMode:
This parameter indicates how the event shall be reported. If the "monitorMode" is "interrupted", then the event shall be reported as a request; if the "monitorMode" is "notifyAndContinue", then the event shall be reported as a notification; if the "monitorMode" is "transparent", then the event shall not be reported.
- legID:
This parameter indicates the party in the call for which the event shall be reported. The gsmSCF shall use the option "sendingSideID" only.

- sendingSideID:

If not included, then the following defaults are assumed for LegID:

"legID" = 1 for the events O_Abandon, T_Abandon and O_Mid_Call,

"legID" = 2 for the events Route_Select_Failure, O_Busy, O_No_Answer, O_Answer, T_Busy, O_Term_Seized, Call_Accepted, T_No_Answer, T_Answer and T_Mid_Call.

The "legID" parameter shall always be included for the events O_Disconnect and T_Disconnect.

- dPSpecificCriteria:

This parameter contains information specific to the EDP that shall be armed.

- applicationTimer:

This parameter indicates the No_Answer timer value for the No_Answer event. If the called party does not answer the call within the allotted time, then the gsmSSF shall report the event to the gsmSCF. This timer shall be shorter than the network No_Answer timer.

- midCallControlInfo:

This parameter defines the criterion for the detection and reporting of mid-call digits. If this parameter is absent, then the first digit entered shall be reported.

- automaticRearm:

This parameter indicates that the gsmSSF shall rearm the DP whenever it is encountered.

11.29.2 Responding entity (gsmSSF)

11.29.2.1 Normal procedure

gsmSSF preconditions:

- (1) A control relationship exists between the gsmSSF and the gsmSCF.
- (2) The gsmSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

NOTE: In the state "monitoring" only requests to disarm detection points (with MonitorMode set to "Transparent") or to send notifications of events (with MonitorMode set to "NotifyAndContinue") shall be accepted by the gsmSSF.

gsmSSF postconditions:

- (1) The requested EDPs are armed or disarmed as indicated.
- (2) Previously requested events are monitored until ended by a transparent monitor mode, until the end of the call, until the EDPs are detected or until the corresponding leg is released.
- (3) The gsmSSF FSM remains in the same state, unless all EDPs have been disarmed and no CallInformationReport or ApplyChargingReport has been requested; in the latter case, the gsmSSF FSM transits to the state "Idle".

11.29.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.30 ResetTimer procedure

11.30.1 General description

The gsmSCF uses this operation to refresh the Tssf application timer, in order to avoid the Tssf time-out at the gsmSSF.

11.30.1.1 Parameters

- timerID:
This parameter indicates which timer shall be reset. The only permissible value of this parameter is "tssf".
- timerValue:
This parameter specifies the value to which the timer shall be set.
- callSegmentID:
This parameter indicates the Call Segment in the gsmSSF for which the timer shall be reset.

11.30.2 Responding entity (gsmSSF)

11.30.2.1 Normal procedure

gsmSSF preconditions:

- (1) Basic call processing has been suspended at a DP.
- (2) The gsmSSF FSM is in the state "Waiting_for_Instructions", the state "Waiting_for_end_of_User_Interaction" or in the state "Waiting_for_end_of_Temporary_Connection".

gsmSSF postconditions:

- (1) The Tssf timer is loaded with the value received from the gsmSCF and is restarted.
- (2) No gsmSSF FSM state transition.

11.29.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.31 SendChargingInformation procedure

11.31.1 General description

The gsmSCF uses this operation to instruct the gsmSSF on the Advice of Charge information to be sent by the gsmSSF. The SendChargingInformation operation may be invoked on multiple occasions.

The SendChargingInformation operation may be used for MO and MT calls in the VMSC. In the case of an MT call, the CSE provided e-parameters are not used by Mobile Station if a call forwarding or follow-on call occurs.

11.31.1.1 Parameters

- sCIBillingChargingCharacteristics:
This parameter is a choice between two lists of information.

The first list shall be sent only if there is neither an active call leg, nor a Temporary Connection, nor a connection to a gsmSRF. It contains the following parameters:

- aOCBeforeAnswer:
This is a list of the following information:
 - aOCInitial:
This is a set of Charge Advice Information (CAI) elements, as defined in 3GPP TS 22.024 [2] These CAI elements shall be sent by the gsmSSF to the MS when Answer is detected and a tariff switch for the "CSE control of e-parameters" has not yet occurred.
 - aOCSubsequent:
This list may contain the following information:

- cAIElements:
This is a set of Charge Advice Information (CAI) elements, as defined in 3GPP TS 22.024 [2]. These CAI elements shall be sent to the MS when Answer is detected and a tariff switch for the "CSE control of e-parameters" has occurred previously, or when Answer has previously been detected and a tariff switch for the "CSE control of e-parameters" occurs.
- tariffSwitchInterval (for the "CSE control of e-parameters"):
This parameter indicates the time duration until the next tariff switch for the "CSE control of e-parameters". The measurement of the elapsed tariff switch period shall start immediately after successful execution of this operation.

The second list in the choice shall be sent only if there is an active call leg or a Temporary Connection or a connection to a gsmSRF. It contains the following parameters:

- aOCAfterAnswer:
This list may contain the following information:
 - cAIElements:
This is a set of Charge Advice Information (CAI) elements, as defined in 3GPP TS 22.024 [2]. These CAI elements shall be sent to the MS by the gsmSSF when Answer is detected and a tariff switch for the "CSE control of e-parameters" has occurred previously, or when Answer has previously been detected and a tariff switch for the "CSE control of e-parameters" occurs.
 - tariffSwitchInterval (for the "CSE control of e-parameters"):
This parameter indicates the time duration until the next tariff switch for the "CSE control of e-parameters". The measurement of the elapsed tariff switch period shall start immediately after successful execution of this operation.
- legID:
This parameter indicates where the charging information shall be sent. For Mobile Originated calls, only leg 1 shall be used. For Mobile Terminated calls in the VMSC, only the leg of the CAMEL subscriber shall be used.

11.31.2 Responding entity (gsmSSF)

11.31.2.1 Normal procedure

gsmSSF preconditions:

- (1) The gsmSSF FSM is in one of the following states:

"Waiting_for_Instructions"; or
"Waiting_for_end_of_User_Interaction"; or
"Waiting_for_end_of_Temporary_Connection"; or
"Monitoring".

gsmSSF postconditions:

- (1) No gsmSSF FSM state transition.

On receipt of this operation the gsmSSF performs actions to send the Advice of Charge information to the indicated Call Party MS.

If Advice of Charge shall be provided to a MS in conjunction with "CSE control of call duration", then the following sequence of operations may be sent by the gsmSCF to the gsmSSF in the following order, in the same TC-CONTINUE TC message:

one or more ApplyCharging operations; SendChargingInformation operation.

These operations shall be processed sequentially by the gsmSSF, in the order that they are sent by the gsmSCF. In this case, the parameters TariffSwitchInterval (for the "CSE control of call duration") may be present in any of the ApplyCharging operations and the parameter TariffSwitchInterval (for the "CSE control of e-parameters") may be present in the SendChargingInformation operation.

The TariffSwitchInterval information received with the ApplyCharging operation shall set or overwrite the tariff switch timer for the "CSE control of call duration" in the gsmSSF and this duration timer shall run from the time of successful operation execution.

The TariffSwitchInterval information received in the SendChargingInformation operation shall set or overwrite the tariff switch timer for the "CSE control of e-parameters" in the gsmSSF and this duration timer shall run from the time of successful operation execution.

11.31.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

11.32 SpecializedResourceReport procedure

11.32.1 General description

The gsmSRF uses this operation as a response to a "PlayAnnouncement" operation or a "PromptAndCollectUserInformation" operation. This operation shall be used only when the "requestAnnouncementCompleteNotification" parameter is TRUE or the "requestAnnouncementStartedNotification" parameter is TRUE in the "PlayAnnouncement" operation or the "PromptAndCollectUserInformation".

11.32.1.1 Parameters

- allAnnouncementsComplete:
This parameter indicates that all the announcements and tones are complete.
- firstAnnouncementStarted:
This parameter indicates that the first announcement or tone has started.

11.32.2 Invoking entity (gsmSRF)

11.32.2.1 Normal procedure

gsmSRF preconditions:

- (1) The gsmSRF FSM is in the state "User_Interaction".
- (2) Either:
 - (i) The sending of announcements and tones as defined in the "PlayAnnouncement" operation or the "PromptAndCollectUserInformation" operation has started and the parameter "RequestAnnouncementStartedNotification" in the "PlayAnnouncement" or "PromptAndCollectUserInformation" operation is TRUE; or
 - (ii) A "PlayAnnouncement" operation has been executed, all announcements and tones have been sent and the parameter "RequestAnnouncementCompleteNotification" in the "PlayAnnouncement" operation is TRUE .

gsmSRF postconditions:

- (1) No gsmSRF FSM state transition.
- (2) If the "DisconnectFromIPForbidden" parameter is FALSE, then the gsmSRF initiates a bearer channel disconnect sequence to the gsmSSF using the applicable bearer channel signalling system after sending the "SpecializedResourceReport" operation to the gsmSCF. The gsmSRF transits to the state "Idle".

11.32.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.33 SplitLeg Procedure

11.33.1 General Description

The gsmSCF uses this operation to request the gsmSSF to separate one party from the source Call Segment and place it in a new target Call Segment.

11.33.1.1 Parameters

- legToBeSplit:
This parameter indicates the party in the call to be split from the source Call Segment.
- newCallSegment:
This parameter indicates the CallSegmentID to be assigned to the newly-created Call Segment.

11.33.2 Responding entity (gsmSSF)

11.33.2.1 Normal procedure

gsmSSF preconditions:

- 1) A control relationship exists between the gsmSCF and the gsmSSF.
- 2) The initial Call Segment is either the source Call Segment or the target Call Segment.
- 3) The BCSM for the leg to be split is in the state O_Active, T_Active, O_Mid_Call or T_Mid_Call.

gsmSSF postconditions:

- 1) The gsmSSF performs the necessary actions to separate the specified leg from its original Call Segment and place it in a new target Call Segment.
- 2) The CS_gsmSSF FSM for the new Call Segment transits to the state "Waiting_for_Instructions".
- 3) The CS_gsmSSF FSM for the source Call Segment transits to the state "Waiting_for_Instructions".
- 4) The remaining BCSM instances within the source Call Segment transit to the O_Mid_Call DP or to the T_Mid_Call DP, unless already suspended at a DP. Note that no Mid_Call EDP will be reported for this case.
- 5) A Return Result shall be sent to the gsmSCF immediately after successful execution of this operation.

11.33.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10, and the TC services which are used for reporting operation errors are described in clause 14.

12 Detailed operation procedures for SMS control

NOTE: For Short Message processing in a Circuit Switched network, the CAMEL interaction with SMS is done through the MSC/smsSSF.
For Short Message processing in a Packet Switched network, the CAMEL interaction with SMS is done through the SGSN/smsSSF.

Any reference to the smsSSF applies to both the smsSSF co-located with the MSC and the smsSSF co-located with the SGSN.

NOTE Where a parameter for an SMS control Operation is marked OPTIONAL in ASN.1, the reader is referred to the conditions for presence for this parameter, specified in the respective Information Flow in 3GPP TS 23.078 [7].

12.1 ConnectSMS procedure

12.1.1 General description

The gsmSCF uses this operation to request the smsSSF to continue Short Message processing, such as routing a Short Message to a specific destination or delivering a Short Message to the served subscriber, with modified information.

12.1.1.1 Parameters

- callingPartysNumber:
This parameter allows the gsmSCF to set the Calling Party Number parameter used in the network. It is used for showing the sending party's id to the receiving party.
- destinationSubscriberNumber:
This parameter contains the receiving party number to who the Short Message shall be routed by the SMSC.
- smscAddress:
This parameter contains the Short Message Service Centre address towards which the Short Message shall be routed.

12.1.2 Responding entity (smsSSF)

12.1.2.1 Normal procedure

smsSSF preconditions:

- (1) Mobile originating Short Message submission or Mobile terminating Short Message delivery attempt has been initiated.
- (2) Short Message processing has been suspended at the DP SMS_Collected_Info or at DP SMS_Delivery_Requested.
- (3) The smsSSF FSM is in the state "Waiting_for_Instructions".

smsSSF postconditions:

- (1) The smsSSF performs the Short Message processing actions to route the Short Message to the specified destination or to deliver the Short Message to the served subscriber.
- (2) Tssf is stopped.

On receipt of this operation, the smsSSF performs the following actions:

- if the callingPartysNumber, destinationSubscriberNumber or smscAddress are supplied, then these values shall be used for subsequent processing;

- if no EDPs have been armed, then the smsSSF transits to the state "Idle". Otherwise, the smsSSF transits to the state "Monitoring".

The smsSSF shall not perform any implicit arming or disarming of DPs.

Statistic counter(s) are not affected.

12.1.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

12.2 ContinueSMS procedure

12.2.1 General description

The gsmSCF uses this operation to request the smsSSF to proceed with processing at the DP at which it previously suspended processing to await gsmSCF instructions. The smsSSF continues processing without substituting new data from the gsmSCF.

12.2.1.1 Parameters

None.

12.2.2 Responding entity (smsSSF)

12.2.2.1 Normal procedure

smsSSF preconditions:

- (1) processing has been suspended at any DP.
- (2) The smsSSF FSM is in the state "Waiting_for_Instructions".

smsSSF postconditions:

- (1) Tssf is stopped.
- (2) processing continues.
- (3) The smsSSF FSM is in one of the following states:
 - State "Monitoring" because at least one EDP was armed; or
 - State "Idle" because no EDPs were armed.

12.2.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

12.3 EventReportSMS procedure

12.3.1 General description

The smsSSF uses this operation to notify the gsmSCF of a short message related event previously requested by the gsmSCF in a RequestReporSMSEvent operation.

12.3.1.1 Parameters

- eventTypeSMS:
This parameter identifies the type of event that is reported.
- eventSpecificInformationSMS:
This parameter contains the Short Message related information specific to the event.
 - For O_SMS_Failure it shall contain the O-SMSCause, if available.
 - For T_SMS_Failure it shall contain the T-SMSCause, if available.
 - For O_SMS_Submitted and for T_SMS_Delivery it shall be empty.
- miscCallInfo:
This parameter contains DP related information.
 - messageType:
This parameter indicates whether the message is a request, i.e. resulting from a RequestReportSMSEvent with "monitorMode" = "interrupted", or a notification, i.e. resulting from a RequestReportSMSEvent with "monitorMode" = "notifyAndContinue".

12.3.2 Invoking entity (smsSSF)

12.3.2.1 Normal procedure

smsSSF preconditions:

- (1) The smsSSF FSM is in the state "Monitoring";
- (2) The smsSSF FSM proceeds to an EDP that is armed.

smsSSF postconditions:

- (1) The smsSSF has applied implicit disarming of EDPs.
- (2) If message type was notification and there are no more EDPs armed, then the smsSSF FSM transits to the state "Idle". Otherwise, the smsSSF transits to the state "Waiting_for_Instructions".
- (3) If the smsSSF has transitted to the state "Waiting_for_Instructions", then Tssf shall be loaded with the default value and shall be started.

12.3.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

12.4 FurnishChargingInformationSMS procedure

12.4.1 General description

The gsmSCF uses this operation to send charging related information to a Logical SMS record. This Logical SMS record is CAMEL specific. The first FurnishChargingInformationSMS operation leads to the generation of a Logical SMS record. Receipt of subsequent FurnishChargingInformationSMS operations shall overwrite or append the contents of the Logical SMS record.

12.4.1.1 Parameters

- fCISMSBillingChargingCharacteristics:
This parameter contains the following parameters;

- fcIBCCCAMELsequence1:
This parameter contains the following parameters;
- freeFormatData:
This parameter contains free-format billing and/or charging characteristics.
- appendFreeFormatData:
This parameter indicates whether previous free-format data shall be appended or overwritten. See 3GPP TS 23.078 [7] for details.

12.4.2 Responding entity (smsSSF)

12.4.2.1 Normal procedure

smsSSF preconditions:

- (1) The smsSSF FSM is in the state "Waiting_for_Instructions".

smsSSF postconditions:

- (1) Tssf is loaded with the default value and is restarted.
- (2) No smsSSF FSM state transition.

On receipt of this operation, the smsSSF performs actions to create the Logical SMS record, if a Logical SMS record does not already exist, and writes the free-format information carried in the operation into the Logical SMS record. Subsequent FurnishChargingInformationSMS operations received, by default, will overwrite the free-format data previously written in the Logical SMS record, as specified in 3GPP TS 23.078 [7]. It is also possible to append free format data with subsequent FurnishChargingInformationSMS operations.

The Logical SMS records will be associated for a given Short Message submission or Short Message delivery with one or more physical CDRs, as specified in 3GPP TS 32.205 [13], 3GPP TS 32.215 [14] and 3GPP TS 22.115 [4].

12.4.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

12.5 InitialDPSMS procedure

12.5.1 General description

The smsSSF uses this operation after detection of a TDP-R in the smsSF FSM, to request the gsmSCF for instructions to complete the Short Message submission to the SMSC or the Short Message delivery to the served subscriber.

12.5.1.1 Parameters

- destinationSubscriberNumber:
This parameter carries the ISDN number of the entity receiving the short message or the MSISDN of the destination subscriber, in an MO-SMS procedure.
- callingPartyNumber:
In an MO-SMS procedure, this parameter carries the MSISDN of the subscriber. In an MT-SMS procedure, this parameter carries the address of the submitter of the short message.
- eventType:
This parameter indicates the armed smSSF FSM DP, resulting in the InitialDPSMS operation.
- iMSI:
IMSI of the mobile subscriber for whom the CAMEL service is invoked.

- locationInformationInMSC:
This parameter indicates the location of the MSC of the served subscriber. This parameter shall be included only if the InitialDP operation is sent from the MSC.
- locationInformationInSGSN:
This parameter indicates the location of the SGSN of the served subscriber. This parameter shall be included only if the InitialDPSMS operation is sent from the SGSN.
- serviceKey:
This parameter indicates to the gsmSCF the requested IN service. It is used to address the required application/SLP within the gsmSCF; it is not for gsmSCF addressing.
- timeAndTimeZone:
This parameter contains the time that the smsSSF was triggered, and the time zone that the invoking smsSSF resides in.
- tPDataCodingScheme:
This parameter indicates the data coding scheme of the TP-User-Data element within the TPDU. It may indicate a message class. The message class may indicate e.g. the originator of the Short Message.
- tPShortMessageSpecificInfo:
This parameter contains the 1st octet of the TPDU. Refer to 3GPP TS 23.040 [6] for a description of the various TPDU's.
- tPProtocolIdentifier:
This parameter indicates the protocol used above the SM-Transfer Layer.
- tPValidityPeriod:
This parameter indicates the length of the validity period or the absolute time of the validity period termination.
- sMSCAddress:
This parameter defines the address of the SMSC to which the Short Message is intended to be submitted.
- smsReferenceNumber:
This parameter contains the SMS Reference Number assigned to the Short Message by the MSC or SGSN.
- mscAddress:
This parameter contains the E.164 address of the MSC. It shall be present if the SMS processing takes place in the MSC; otherwise shall be absent.
- sgsn-Number:
This parameter contains the Global Title of the SGSN. It shall be present if the SMS processing takes place in the SGSN; otherwise it shall be absent.
- ms-Classmark2:
This parameter contains the MS Classmark 2 of the mobile subscriber for which the service is invoked.
- gPRMSMClass:
This parameter contains the GPRS MS capabilities of the mobile subscriber for which the CAMEL service is invoked.
- iMEI:
This parameter contains the IMEI (with software version) of the mobile subscriber for which the service is invoked.
- calledPartyNumber:
This parameter indicates the served subscriber in an MT-SMS procedure.

12.5.2 Invoking entity (smsSSF)

12.5.2.1 Normal procedure

smsSSF preconditions:

- (1) A Short Message submission attempt or a Short Message delivery attempt has been initiated.

- (2) An event has been detected at a DP.
- (3) For MT-SMS, the event fulfilled the criteria for the DP being executed.

smsSSF postconditions:

- (1) A control relationship has been established and the smsSSF waits for instructions from the gsmSCF.

The address of the gsmSCF the InitialDPSMS operation shall be sent to, shall be fetched from the MO-SMS-CSI or the MT-SMS-CSI. The smsSSF shall provide all available parameters to the gsmSCF.

A control relationship is established with the gsmSCF. The smsSSF application timer Tssf is laded with the deafult value and is started. It is used to prevent excessive Short Message submission or delivery suspension time.

12.5.2.2 Error handling

If the gsmSCF is not accessible, then the smsSSF instructs the MSC or SGSN to proceed with the Short Message processing in accordance with the Default SMS Handling parameter of the MO-SMS-CSI or MT-SMS-CSI.

If Tssf expires, then the smsSSF aborts the interaction with the gsmSCF by means of an abort to TC and shall instruct the MSC or SGSN to proceed with the Short Message processing in accordance with the Default SMS Handling parameter of the MO-SMS-CSI CSI or MT-SMS-CSI.

In the case of an MO-SMS Service, if the sending mobile party abandons after the sending of InitialDPSMS, then the smsSSF shall abort the interaction with the gsmSCF by means of an abort to TC.

NOTE TC will wait until the first response message from the gsmSCF has been received before it sends an abort to the gsmSCF (see also clause 14).

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

12.6 ReleaseSMS procedure

12.6.1 General description

The gsmSCF uses this operation to tear down a Short Message submission attempt or Short Message delivery attempt. The operation may be sent within a control relationship only; it is not allowed in a monitor relationship.

12.6.1.1 Parameters

- rPCause:
This parameter gives an indication to the smsSSF about the reason of releasing this specific Short Message. For a MO-SMS Service, the rPCause shall be sent to the served subscriber in the RP-ERROR RPDU. For a MT-SMS Service, the rPCause shall be sent to the SMS-GMSC in the RP-ERROR RPDU.

12.6.2 Responding entity (smsSSF)

12.6.2.1 Normal procedure

smsSSF preconditions:

- (1) The smsSSF FSM is in the state "Waiting_for_Instructions".
- (2) The FSM is in DP SMS_Collected_Info or in DP SMS_Delivery_Requested.

smsSSF postconditions:

- (1) The smsSSF FSM transits to the state "Idle". All armed EDPs shall be disarmed. All resources in the MSC or SGSN related to the Short Message shall be released.

12.6.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

12.7 RequestReportSMSEvent procedure

12.7.1 General description

The gsmSCF uses this operation to request the smsSSF to monitor for a Short Message related event (FSM events such as failure, delivery or submission) and to send a notification to the gsmSCF when the event is detected.

The monitoring of more than one event may be requested with a single RequestReportSMSEvent operation, but each of these requested events will be reported in a separate EventReportSMS operation.

12.7.1.1 Parameters

- smsEvents:
This parameter indicates the event or events of which a report is requested.
- eventTypeSMS:
This parameter indicates the type of event of which a report is requested. Values SMSCollectedInfo and SMSDeliveryRequested are not valid for the RequestReportSMSEvent operation.
- monitorMode:
This parameter indicates how the event shall be reported. When the "monitorMode" is "interrupted", the event shall be reported as a request, if the "monitorMode" is "notifyAndContinue", the event shall be reported as a notification, if the "monitorMode" is "transparent", then the event shall not be reported.

12.7.2 Responding entity (smsSSF)

12.7.2.1 Normal procedure

smsSSF preconditions:

- (1) A control relationship exists between the smsSSF and the gsmSCF.
- (2) The smsSSF FSM is in the state "Waiting_for_Instructions".

smsSSF postconditions:

- (1) Tssf is loaded with the default value and is restarted.
- (2) The requested EDPs have been armed or disarmed as indicated.
- (3) Armed events are monitored until ended by a transparent monitor mode, until the occurrence of the event or until the implicit disarming of the event.
- (4) No smsSSF FSM state transition.

12.7.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

12.8 ResetTimerSMS procedure

12.8.1 General description

The gsmSCF uses this operation to refresh the Tssf application timer, in order to prevent Tssf expiry at the smsSSF.

12.8.1.1 Parameters

- timerID:
This parameter indicates which timer shall be reset. The only permissible value of this parameter is "tssf".
- timerValue:
This parameter defines the value to which the timer shall be set.

12.8.2 Responding entity (smsSSF)

12.8.2.1 Normal procedure

smsSSF preconditions:

- (1) Short Message processing has been suspended at a DP.
- (2) The smsSSF FSM is in the state "Waiting_for_Instructions".

smsSSF postconditions:

- (1) The Tssf timer is loaded with the value received from the gsmSCF and is restarted.
- (2) No smsSSF FSM state transition.

12.8.2.2 Error handling

Generic error handling for the operation related errors are described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13 Detailed operation procedures for GPRS control

NOTE Where a parameter for a GPRS control Operation is marked OPTIONAL in ASN.1, the reader is referred to the conditions for presence for this parameter, specified in the respective Information Flow in 3GPP TS 23.078 [7].

13.1 ActivityTestGPRS procedure

13.1.1 General description

The gsmSCF uses this operation to check for the continued existence of a relationship between the gsmSCF and the gprsSSF. If the relationship is still in existence, then the identified instance of gprsSSF will respond. If the ActivityTestGPRS operation timer expires, then the gsmSCF will assume that the receiving entity has failed in some way and will take appropriate action. This operation opens a new SS7 dialogue between gsmSCF and gprsSSF.

13.1.1.1 Parameters

None.

13.1.2 Responding entity (gprsSSF)

11.2.2.1 Normal procedure

gprsSSF preconditions:

- (1) A relationship exists between the gsmSCF and the gprsSSF.
- (2) The gprsSSME FSM is in the state "Idle Management".
- (3) No active TC dialogue exists at the moment of receiving ActivityTestGPRS.

gprsSSF postconditions:

- (1) The gprsSSME FSM transits to the state "Non-call Associated Treatment".
- (2) If there is a gprsSSF using the GPRS-Reference Number, then the gprsSSME FSM sends a Return Result "ActivityTestGPRS" to the gsmSCF. The gprsSSME FSM then returns to the state "Idle Management".

If there is no gprsSSF using the GPRS-ReferenceNumber, then the gprsSSME FSM will issue a U-Abort. The gprsSSME FSM then returns to the state "Idle Management".

- (3) The temporary TC dialogue is closed.

If at the time of receiving ActivityTestGPRS there is an active TC dialogue for this GPRS Dialogue, then the gprsSSME FSM issues a U-Abort with Abort reason "overlapping-dialogue".

13.1.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services used for reporting operation errors are described in clause 14.

13.2 ApplyChargingGPRS procedure

13.2.1 General description

The gsmSCF uses this operation for interacting with the gprsSSF function "CSE control of GPRS session or PDP Context duration and volume". The ApplyChargingGPRSReport operation provides the feedback from the gprsSSF to the gsmSCF. The charging scenarios supported by this operation are those given in 3GPP TS 22.078 [3] for "CSE control of GPRS session and PDP Context duration and volume".

If this procedure is used within a PDP Context dialogue, then the charging instruction shall pertain to the PDP Context only. Data volume threshold and duration threshold may be defined separately.

If this procedure is used within a GPRS Session dialogue, then the charging instruction may pertain to the GPRS Session or to a single PDP Context. Charging for a PDP Context may be on duration and/or volume. Charging for a GPRS Session may be on duration only.

NOTE: Charging for a PDP Context on duration and volume requires two ApplyChargingGPRS operations.

13.2.1.1 Parameters

- chargingCharacteristics:
This parameter is a choice between parameters required for "CSE control of a GPRS session or a PDP Context duration or volume":
 - maxTransferredVolume:
This parameter specifies the maximum volume that may be transferred before a ApplyChargingReportGPRS shall be sent to the gsmSCF.
 - maxElapsedTime:
This parameter specifies the maximum period of time before a ApplyChargingReportGPRS shall be sent to the gsmSCF.
- tariffSwitchInterval:
This parameter indicates the time duration until the next tariff switch. The measurement of the elapsed tariff switch period shall start immediately after successful execution of this operation.
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, to which the charging instruction applies.

13.2.2 Responding entity (gprsSSF)

13.2.2.1 Normal procedure

gprsSSF preconditions:

- (1) A control relationship exists between the gsmSCF and the GPRS Session or PDP Context to which the operation applies.
- (2) The gprsSSF FSM is in one of the following states: "Waiting_for_Instructions" or "Monitoring".

gprsSSF postconditions:

- (1) No gprsSSF FSM state transition.

On receipt of this operation, the gprsSSF shall set the charging data using the information elements included in the operation.

13.2.2.2 Error handling

"TaskRefused": In addition to the generic error handling noted below, this error shall be indicated when:

- a previously received GPRS Session or PDP Context period or volume duration is pending;
- a tariffSwitchInterval is indicated when a previously received tariffSwitchInterval is pending.

Generic error handling for the operation related errors is described in clause 10 and the TC services used for reporting operation errors are described in clause 14.

13.3 ApplyChargingReportGPRS procedure

13.3.1 General description

The gprsSSF uses this operation to report charging related information to the gsmSCF as requested by the gsmSCF using the ApplyChargingGPRS operation.

Timing of duration and measuring of transferred data (if applicable) shall be started when either an Attach, PDP Context Establishment Acknowledgement or an Inter SGSN Routeing Area Update acceptance is detected by the gprsSSF.

A report shall be made when a PDP Context Disconnect, a Detach or a Change in QoS is detected by the gprsSSF or when the gprsSSF detects that the transferred volume or elapsed time duration indicated in the parameter "transferredVolume" or "elapsedTime" (received in ApplyChargingGPRS operation) has been reached. ApplyChargingReportGPRS shall be sent only on chargeable QoS changes.

13.3.1.1 Parameters

- chargingResult:
This parameter provides the gsmSCF with the charging related information previously requested using the ApplyChargingGPRS operation. The "ChargingResult" is a choice, and can contain either of the following parameters:
 - transferredVolume:
This is a choice of the following parameters:
 - volumeIfNoTariffSwitch:
This parameter shall be present if no tariff switch has occurred for the PDP Context, otherwise it shall be absent. If present, then the volume transferred since the detection of the event that triggered volume count shall be reported.
 - volumeIfTariffSwitch:
This parameter shall be present if a tariff switch has occurred for the PDP Context, otherwise it shall be absent. This parameter may contain the following information:
 - volumeSinceLastTariffSwitch:
The volume since the detection of the event that triggered volume count or the last tariffSwitch (whichever of these events was last detected) shall be reported.
 - VolumeTariffSwitchInterval:
This parameter shall be present only if a tariff switch was detected after the event that triggered volume count for the PDP Context in the current volume count period. If present, the volume between either the detection the event that triggered volume count or the previous tariff switch (whichever of these events was last detected) and the last tariff switch shall be reported.
- elapsedTime:
This parameter is a choice of the following parameters:
 - timeGPRSIfNoTariffSwitch:
This parameter shall be present if no tariff switch has occurred for the GPRS Session or the PDP Context, otherwise it shall be absent. If present, then the elapsed time since the detection of the event that triggered time count shall be reported.

- timeGPRSIfTariffSwitch:
This parameter shall be present if a tariff switch has occurred for the GPRS Session or the PDP Context, otherwise it shall be absent. This parameter may contain the following information:
 - timeGPRSSinceLastTariffSwitch:
The time since the event that triggered time count or the last tariffSwitch shall be reported.
 - timeGPRSTariffSwitchInterval:
This parameter shall be present only if a tariff switch was detected after the event that triggered time count for the GPRS Session or PDP Context in the current time count period. If present, then the time between either the detection the event that triggered time count or the previous tariff switch (whichever of these events was last detected) and the last tariff switch shall be reported.
- qualityOfService:
This parameter identifies the QoS which was negotiated between the user, the SGSN and the GGSN. This parameter shall be present only if the sending of the ApplyChargingReportGPRS operation was triggered by a change in Quality of Service.
- active:
This parameter indicates whether the GPRS Session or PDP Context is still active
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, for which the charging report is valid.
- chargingRollOver:
This parameter indicates possible rollovers of the "ChargingResult" parameter due to the limited value ranges of the associated charging parameters. The "chargingRollOver" parameter is a choice and may contain either of the following parameters:
 - transferredVolumeRollOver:
This parameter is a choice of the following parameters:
 - rO-VolumeIfNoTariffSwitch:
This parameter indicates how many times the volumeIfNoTariffSwitch parameter of the chargingResult has rolled over. If no rollover has happened, then rO-VolumeIfNoTariffSwitch may be absent.
 - rO-VolumeIfTariffSwitch:
This parameter is present if at least one of the parameters below is present. It may contain the following information:
 - rO-VolumeSinceLastTariffSwitch:
This parameter indicates how many times the volumeSinceLastTariffSwitch parameter of the chargingResult has rolled over. If no rollover has happened, then rO-VolumeSinceLastTariffSwitch may be absent.
 - rO-VolumeTariffSwitchInterval:
This parameter indicates how many times the VolumeTariffSwitchInterval parameter of the chargingResult has rolled over. If no rollover has happened, then rO-VolumeTariffSwitchInterval may be absent.
- elapsedTimeRollOver:
This parameter is a choice of the following parameters:
 - rO-TimeGPRSIfNoTariffSwitch:
This parameter indicates how many times the timeGPRSIfNoTariffSwitch parameter of the chargingResult has rolled over. If no rollover has happened, then rO-TimeGPRSIfNoTariffSwitch may be absent.
 - rO-TimeGPRSIfTariffSwitch:
This parameter shall be present if at least one of the parameters below is present. It may contain the following information:
 - rO-TimeGPRSSinceLastTariffSwitch:
This parameter indicates how many times the timeGPRSSinceLastTariffSwitch parameter of the

chargingResult has rolled over. If no rollover has happened, then rO-TimeGPRSSinceLastTariffSwitch may be absent.

- rO-TimeGPRSTariffSwitchInterval:
This parameter indicates how many times the timeGPRSTariffSwitchInterval parameter of the chargingResult has rolled over. If no rollover has happened, then rO-TimeGPRSTariffSwitchInterval may be absent.

13.3.2 Invoking entity (gprsSSF)

13.3.2.1 Normal procedure

gprsSSF preconditions:

- (1) A relationship exists between the gsmSCF and the GPRS Session or PDP Context.
- (2) A charging event has been detected that was requested by the gsmSCF via an ApplyChargingGPRS operation.

gprsSSF postconditions:

- (1) If termination of the GPRS Session or a PDP Context has occurred, then:
 - If the sending of ApplyChargingReportGPRS is directly followed by the reporting of an EDP-R, then the gprsSSF FSM shall transit to the state "Waiting_for_Instructions"; else
 - If there are any armed EDPs or pending reports, then the gprsSSF FSM shall remain in the same state; else
 - If there are no armed EDPs or pending reports, then the gprsSSF FSM shall transit to the state "Idle".
- (2) If the sending of ApplyChargingReportGPRS is due to a timer or counter expiry, then:
 - If there are any armed EDPs or pending reports, then the gprsSSF FSM shall remain in the same state; else
 - If there are no armed EDPs or pending reports, then the gprsSSF FSM shall transit to the state "Idle".
- (3) If the sending of ApplyChargingReportGPRS is due to a change in QoS of a PDP Context, then:
 - The gprsSSF FSM shall remain in the same state.

13.3.2.2 Error handling

If the operation timer expires, then the gprsSSF shall abort the TC dialogue, terminate the GPRS dialogue and instruct the SGSN to handle the GPRS session or PDP context in accordance with the default GPRS handling parameter of the valid CSI.

Generic error handling for the operation related errors is described in clause 10 and the TC services used for reporting operation errors are described in clause 14.

13.4 CancelGPRS procedure

13.4.1 General description

The gsmSCF uses this operation to request the gprsSSF to disarm all pending EDPs and to cancel all pending reports for a GPRS Session or for a specific PDP Context. This enables the gprsSSF FSM to transit to the state "Idle". This procedure can not be used to cancel a previous operation.

13.4.1.1 Parameters

- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, for which the armed EDPs shall be disarmed and the pending reports shall be cancelled.

13.4.2 Responding entity (gprsSSF)

13.4.2.1 Normal procedure

gprsSSF preconditions:

- (1) The gprsSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gprsSSF postconditions:

- (1) All pending ApplyChargingReportGPRS are cancelled and all pending EDPs are disarmed. If a PDPID is included in the operation, then the cancelling of the pending reports and the disarming of the armed events applies to the indicated PDP Context only.
- (2) If the gprsSSF FSM was in the state "Monitoring and there are no more armed EDPs or pending reports, then the gprsSSF FSM shall transit to the state "Idle". If the gprsSSF FSM was in the state "Monitoring" it shall transit to the state "Idle" if there are no other PDP Contexts pending; or

If the gprsSSF FSM was in the state "Waiting_for_Instructions", it shall remain in that state. If there are no more armed EDPs or pending reports, then a subsequent GPRS Session or PDP Context processing operation will result in the gprsSSF FSM to transit the state "Idle".

The GPRS Session or PDP Context to which the CancelGPRS operation applies, if in active state, shall further be treated by the gprsSSF autonomously as a normal (non-CAMEL) GPRS Session or PDP Context.

13.4.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.5 ConnectGPRS procedure

13.5.1 General description

The gsmSCF uses this operation to provide an APN to the gprsSSF, to be used for establishing a PDP Context.

13.5.1.1 Parameters

- accessPointName:
This parameter contains the Access Point Name towards which the PDP Context shall be established. Refer to 3GPP TS 29.060 [12] for details on the Access Point Name.
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, for which the Access Point Name shall be used.

13.5.2 Responding entity (gprsSSF)

13.5.2.1 Normal procedure

gprsSSF preconditions:

- (1) A control relationship exists between the the gsmSCF and the PDP Context.
- (2) The GPRS PDP Context FSM is suspended at DP PDP_Context_Establishment.
- (3) The gprsSSF FSM is in the state "Waiting_for_Instructions".

gprsSSF postconditions:

- (1) The gprsSSF performs the actions to establish the PDP Context using the given Access Point Name.
- (2) The gprsSSF stops Tssf.
- (3) If no EDPs are armed, then the gprsSSF FSM transits to the state "Idle". Otherwise the gprsSSF FSM transits to the state "Monitoring".

The gsmSSF shall not perform any implicit arming or disarming of DPs.

13.5.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.6 ContinueGPRS procedure

13.6.1 General description

The gsmSCF uses this operation to request the gprsSSF to proceed with GPRS Session or PDP Context processing at the DP at which it previously suspended processing to await gsmSCF instructions. The gprsSSF continues processing without substituting new data from the gsmSCF.

13.6.1.1 Parameters

- pDPID:
This parameter identifies the PDP Context within the control relationship for which the processing shall continue.

13.6.2 Responding entity (gprsSSF)

13.6.2.1 Normal procedure

gprsSSF preconditions:

- (1) A control relationship exists between the the gsmSCF and the GPRS Session or PDP Context.
- (2) The GPRS Session or PDP Context processing has been suspended at any DP.
- (3) The gprsSSF FSM is in the state "Waiting_for_Instructions".

gsmSSF postconditions:

- (1) GPRS Session or PDP Context processing continues.
- (2) The gprsSSF FSM is in one of the following states:
 - "Monitoring" because at least one EDP was armed or an ApplyChargingReportGPRS was requested; or
 - "Idle" because no EDPs were armed and no ApplyChargingReportGPRS was requested.

13.6.2.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

13.7 EntityReleasedGPRS procedure

13.7.1 General description

The gprsSSF uses this operation to inform the gsmSCF that the GPRS Session is detached or a PDP Context is disconnected. It shall be used only when the associated event detection point (ie. for GPRS Session Detach: DP "Detach" and for PDP Context Disconnect: DP "PDP Context Disconnection") is at that moment not armed for reporting.

This operation shall be used irrespectively of the functional entity that initiated the Detach or PDP Context Disconnect and irrespectively of the cause for the Detach or PDP Context Disconnect.

When a PDP Context is terminated, then the gprsSSF shall send all pending reports of that PDP Context to the gsmSCF. When a GPRS Session is terminated, then the gprsSSF shall send all outstanding reports of the GPRS Session to the gsmSCF.

13.7.1.1 Parameters

- gPRSCause:
This parameter gives an indication to the gsmSCF about the reason for discontinuing the PDP Context or GPRS Session. This may be used by gsmSCF if a FurnishChargingInformationGPRS operation needs to be sent to the gprsSSF.
- pDPID:
This parameter identifies the PDP Context within the GPRS Session dialogue, which has terminated.

13.7.2 Invoking entity (gprsSSF)

13.7.2.1 Normal procedure

gprsSSF preconditions:

- (1) The gprsSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gprsSSF postconditions:

- (1) All armed EDPs for the indicated PDP Context or GPRS Session shall be disarmed. All connections and resources related to the indicated PDP Context or GPRS Session shall be released.
If there are no more armed EDPs or pending reports, then the gprsSSF FSM transits to the state "Idle"; otherwise the gprsSSF FSM remains in the same state.

13.7.2.2 Error handling

If the operation timer expires, then the gprsSSF shall abort the TC dialogue, terminate the GPRS dialogue and instruct the SGSN to handle the GPRS session or PDP context in accordance with the default GPRS handling parameter of the valid CSI.

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.8 EventReportGPRS procedure

13.8.1 General description

The gprsSSF uses this operation to notify the gsmSCF of a GPRS Session or PDP Context event previously requested by the gsmSCF in a RequestReportGPRSEvent operation.

13.8.1.1 Parameters

- gPRSEventType:
This parameter specifies the type of event that is reported.
- gPRSEventSpecificInformation:
This parameter indicates the GPRS Session or PDP Context related information specific to the event.
 - For Change of Position GPRS Session it shall contain the "locationInformationGPRS", if available.
 - For Change of Position PDP Context it shall contain the "accessPointName", "chargingID", "locationInformationGPRS", "endUserAddress", Quality of Service, "timeAndTimeZone" and "gGSNAddress" and "secondaryPDP-context", if available.
 - For Detach and PDP Context Disconnect it shall contain the "initiatingEntity" and, conditionally, "routeingAreaUpdate". The "initiatingEntity" indicates the entity that initiated the Detach or PDP Context Disconnect. The "routeingAreaUpdate" indicates that the Detach or PDP Context Disconnect is due to inter-SGSN routeing area update.
 - In the case of inter-SGSN routeing area update, the gsmSCF may ignore the "initiatingEntity".
 - For PDP Context Establishment it shall contain the "accessPointName", "endUserAddress", "pDPInitiationType", Quality of Service, "locationInformationGPRS", "timeAndTimeZone" and "secondaryPDP-context", if available.
 - The Quality of Service shall contain the Requested QoS and the Subscribed QoS.
 - For PDP Context Establishment Acknowledgement it shall contain the "accessPointName", "chargingID", "endUserAddress", Quality of Service, "locationInformationGPRS", "timeAndTimeZone" and "gGSNAddress", if available.
 - The Quality of Service shall contain the Requested QoS, the Subscribed QoS and the Negotiated QoS.
- All optional gPRSEventSpecificInformation parameters shall be sent in accordance with 3GPP TS 23.078 [7] subclause 6.6.1.4 and 3GPP TS 22.078 [3] annex "GPRS Information provided to the CSE".
- miscGPRSInfo:
This parameter contains DP related information.
- messageType:
This parameter indicates whether the message is a request, i.e. resulting from a RequestReportGPRSEvent with "monitorMode" = "interrupted", or a notification, i.e. resulting from a RequestReportGPRSEvent with "monitorMode" = "notifyAndContinue".
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, for which the event is reported.

13.8.2 Invoking entity (gprsSSF)

13.8.2.1 Normal procedure

gprsSSF preconditions:

- (1) The gprsSSF FSM is in the state "Monitoring" or in the state "WaitingForInstructions".
- (2) The GPRS Session or PDP Context FSM proceeds to an EDP that is armed.

gprsSSF postconditions:

- (1) If the message type was notification and there are still armed EDPs that can be met or there any pending reports, then the gprsSSF FSM shall remain in the state "Monitoring".
- (2) If the message type was notification and there no more EDPs armed that can be met and there are no pending reports, then the gprsSSF FSM shall transit to the state "Idle".
- (3) If the message type was request, then the gprsSSF FSM shall transit to the state "Waiting_for_Instructions". GPRS Session or PDP Context processing is interrupted.

If an EDP-R is met that causes the release of a GPRS Session or PDP Context, then all EDPs related to the GPRS Session or PDP Context shall be disarmed.

13.8.2.2 Error handling

If the operation timer expires, then the gprsSSF shall abort the TC dialogue, terminate the GPRS dialogue and instruct the SGSN to handle the GPRS session or PDP context in accordance with the default GPRS handling parameter of the valid CSI.

If the message type is "request" and the Tssf timer expires, then the gprsSSF shall abort the TC dialogue and shall instruct the SGSN to treat the GPRS Session or PDP Context in accordance with the default GPRS handling of the valid CSI.

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.9 FurnishChargingInformationGPRS procedure

13.9.1 General description

The gsmSCF uses this operation to send charging related information to a Logical GPRS record. This Logical GPRS record is CAMEL specific. The first FurnishChargingInformationGPRS operation results in the generation of a Logical GPRS record. Receipt of subsequent FurnishChargingInformationGPRS operations shall overwrite or append the contents of the Logical GPRS record.

13.9.1.1 Parameters

- fCIGPRSBillingChargingCharacteristics:
This parameter contains the following parameters;
- fCIBCCCAMELsequence1:
This parameter contains the following parameters;
 - freeFormatData:
This parameter contains free-format billing and/or charging characteristics.
 - appendFreeFormatData:
This parameter indicates whether previous FCI free format data is appended or overwritten. Refer to 3GPP TS 23.078 [7] for details of this mechanism.
 - pDPID:
This parameter indicates the PDP Context's Logical GPRS record to which the free format data belongs.

13.9.2 Responding entity (gprsSSF)

13.9.2.1 Normal procedure

gprsSSF preconditions:

- (1) The gprsSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gprsSSF postconditions:

- (1) No gprsSSF FSM state transition.

On receipt of this operation the gprsSSF performs actions to create the Logical GPRS record, if necessary, and writes the free-format information carried in the operation into the GPRS record. A FurnishChargingInformationGPRS operation will create a Logical GPRS Data Record (CDR) if such a record does not already exist. Subsequent FurnishChargingInformationGPRS operations received, will either overwrite or append the data previously written in the free-format CDR field depending on the presence and value of the parameter "appendFreeFormatData".

The Logical GPRS records will be associated for a given GPRS Session or PDP Context with one or more physical CDRs, as specified in 3GPP TS 32.205 [14] and 3GPP TS 22.115 [4].

13.9.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.10 InitialDPGPRS procedure

13.10.1 General description

The gprsSSF uses this operation after detection of a TDP-R in the GPRS Session or PDP Context state machine, to request the gsmSCF for instructions to complete the GPRS Session or PDP Context.

For a GPRS Session, the "Attach" and "Change of Position Session" TDPs may result in the InitialDPGPRS Procedure.

For a PDP Context, the "PDP Context Establishment", the "PDP Context Establishment Acknowledgement" and the "Change of Position Context" TDPs may result in the InitialDPGPRS Procedure.

If a PDP Context related TDP is met and there is at that moment a GPRS dialogue for the GPRS Session, then the gprsSSF shall not initiate the InitialDPGPRS Procedure for that PDP Context.

If the "PDP Context Establishment Acknowledgement" event occurs and this event is armed as a TDP, and there is at that moment a GPRS dialogue for the PDP Context, then the gprsSSF shall not initiate a new InitialDPGPRS Procedure for that PDP Context.

13.10.1.1 Parameters

- serviceKey:
This parameter indicates to the gsmSCF the requested IN service. It is used to address the required application/SLP within the gsmSCF; it is not used for SCP addressing.
- gPRSEventType:
This parameter indicates the armed GPRS Attach/Detach FSM or PDP Context FSM DP event, resulting in the InitialDPGPRS operation.
- mSISDN:
This parameter contains the MSISDN of the mobile subscriber for which the CAMEL service is invoked.
- iMSI:
This parameter contains the IMSI of the mobile subscriber for which the CAMEL service is invoked.
- timeAndTimezone:
This parameter contains the time that the gprsSSF is triggered, and the time zone that the invoking gprsSSF resides in.
- gPRSMSCClass:
This parameter contains the MS capabilities of the mobile subscriber for which the CAMEL service is invoked.
- endUserAddress:
This parameter identifies the PDP type, PDP type organisation and the actual PDP address.
- qualityOfService:
This parameter contains the Quality of Service.
If the InitialDPGPRS operation is sent as a result of the "PDP Context Establishment" TDP, then the Quality of Service parameter shall contain the Requested QoS and the Subscribed QoS.
If the InitialDPGPRS operation is sent as a result of the "PDP Context Establishment Acknowledgement" TDP or the Change of Position Context TDP, then the Quality of Service parameter shall contain the Requested QoS, the Subscribed QoS and the Negotiated QoS.

- **accessPointName:**
This parameter contains the requested address that the MS for which the CAMEL service is invoked for wants to connect to.
- **routingAreaIdentity:**
This parameter contains the location information of the MS for which the CAMEL service is invoked.
- **chargingID:**
This parameter contains the charging ID that, together with the gGSNAddress, uniquely identifies the PDP Context for the MS for which the CAMEL service is invoked.
- **sGSNcapabilities:**
This parameter specifies the capabilities which the SGSN node can provide for the CAMEL service control.
- **locationInformationInSGSN:**
This parameter indicates the location of the sending MS.
- **pDPInitiationType:**
This parameter indicates whether a PDP Context was established as a result of a network-initiated request or as a result of a subscriber request.
- **gGSNAddress:**
This parameter refers to the IP address of the GGSN where the PDP Context terminates. It is used together with the chargingID for uniquely identification of the PDP Context for which the CAMEL service is invoked from.
- **secondaryPDP-context:**
This parameter indicates that the PDP Context is requested as a secondary PDP Context.
- **iMEI:**
This parameter contains the IMEI (with software version) of the mobile subscriber for which the service is invoked.

13.10.2 Invoking entity (gprsSSF)

13.10.2.1 Normal procedure

gprsSSF preconditions:

- (1) An event has been met that is armed as TDP.
- (2) There is no GPRS dialogue active for that PDP Context or for the GPRS Session.

gprsSSF postconditions:

- (1) A control relationship is established between the gsmSCF and the GPRS Session or the PDP Context.
- (2) The gprsSSF FSM is in the state "Waiting_for_Instructions".

The address of the gsmSCF that the InitialDPGPRS operation shall be sent to, shall be fetched from the valid CSI. The gprsSSF shall provide all available parameters to the gsmSCF.

The gprsSSF application timer Tssf shall be loaded and started when the gprsSSF sends InitialDPGPRS for requesting instructions from the gsmSCF. It is used to prevent excessive GPRS Session or PDP Context duration or volume usage.

13.10.2.2 Error handling

If the gsmSCF is not accessible, then the gprsSSF instructs the SGSN to handle the GPRS Session or PDP Context in accordance with the Default GPRS Handling parameter of the valid CSI.

If Tssf expires, then the gprsSSF shall abort the interaction with the gsmSCF by means of an abort to TC and shall instruct the SGSN to handle the GPRS Session or PDP Context in accordance with the Default GPRS Handling parameter of the valid CSI.

If the MS abandons the establishment of a GPRS Session or PDP Context after the sending of InitialDPGPRS, then the gprsSSF shall abort the interaction with the gsmSCF by means of an abort to TC.

NOTE TC will wait until the first response message from the gsmSCF has been received before it sends an abort to the gsmSCF (see also clause 14).

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.11 ReleaseGPRS procedure

13.11.1 General description

The gsmSCF uses this operation to tear down by the gsmSCF an existing GPRS Session or PDP Context at any phase.

13.11.1.1 Parameters

- gPRSCause:
This parameter gives an indication to the gprsSSF about the reason of releasing the GPRS Session or a specific PDP Context. This may be used by gprsSSF for generating specific indications to the MS or to fill in the "cause" in the release message.
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, which shall be released.

13.11.2 Responding entity (gprsSSF)

13.11.2.1 Normal procedure

gprsSSF preconditions:

- (1) A control relationship exists between gsmSCF and the GPRS Session or PDP Context. More specifically, in order to tear down an individual PDP Context, an EDP-R must be armed for that PDP Context. In order to make a SCP controlled detach an EDP-R must be armed for the GPRS Session.
- (2) The gprsSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gprsSSF postconditions:

- (1) All pending reports for the GPRS Session or the PDP Context shall be reported to the gsmSCF.
- (2) All connections and resources related to the GPRS Session or the PDP Context shall be released.
- (3) All armed EDPs for the GPRS Session or the PDP Context shall be disarmed.
- (4) If there are no more pending reports or armed events, then the gprsSSF FSM transits to the state "Idle".

If there are any pending reports or armed events, then the gprsSSF FSM remains in the same state.

13.11.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.12 RequestReportGPRSEvent procedure

13.12.1 General description

The gsmSCF uses this operation to request the gprsSSF to monitor for a GPRS Session or PDP Context related event (e.g., events such as PDP Context establishment or detach) and to send a notification to the gsmSCF when the event is detected.

The monitoring of more than one event may be requested with a single RequestReportGPRSEvent operation, but each of these requested events will be reported in a separate EventReportGPRS operation.

13.12.1.1 Parameters

- gPRSEvent:
This parameter specifies the event or events of which a report is requested.
- gPRSEventType:
This parameter specifies the type of event of which a report is requested.
- monitorMode:
This parameter indicates how the event shall be reported. If the "monitorMode" is "interrupted", then the event shall be reported as a request; if the "monitorMode" is "notifyAndContinue", then the event shall be reported as a notification; if the "monitorMode" is "transparent", then the event shall not be reported.
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, for which the event reporting is requested

13.12.2 Responding entity (gprsSSF)

13.12.2.1 Normal procedure

gprsSSF preconditions:

- (1) A control relationship exists between the gsmSCF and the GPRS Session or PDP Context.
- (2) The gprsSSF FSM is in the state "Waiting_for_Instructions" or the state "Monitoring".

In the state "monitoring" only requests to disarm detection points (with MonitorMode set to "Transparent") or send notifications of events (with MonitorMode set to "NotifyAndContinue") shall be accepted.

gprsSSF postconditions:

- (1) The requested EDPs have been armed or disarmed as indicated.
- (2) Previously requested events are monitored until ended by a transparent monitor mode, until the end of the GPRS Session or PDP Context or until the EDPs are detected.
- (3) If there are no armed events or pending reports, then the gsmSSF FSM shall transit to the state "Idle". Otherwise it shall remain in the same state.

13.12.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.13 ResetTimerGPRS procedure

13.13.1 General description

The gsmSCF uses this operation to refresh the Tssf application timer, in order to avoid the Tssf time-out at the gprsSSF.

13.13.1.1 Parameters

- timerID:
This parameter indicates which timer shall be reset. The only permissible value for this parameter is "Tssf".
- timerValue:
This parameter specifies the value to which the timer shall be set.

13.13.2 Responding entity (gprsSSF)

13.13.2.1 Normal procedure

gprsSSF preconditions:

- (1) GPRS Session or PDP Context processing has been suspended at a DP.
- (2) The gprsSSF FSM is in the state "Waiting_for_Instructions".

gprsSSF postconditions:

- (1) The Tssf timer is loaded with the value received from the gsmSCF and is restarted.
- (2) No gprsSSF FSM state transition.

13.13.2.2 Error handling

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

13.14 SendChargingInformationGPRS Procedure

13.14.1 General description

The gsmSCF uses this operation to instruct the gprsSSF on the Advice of Charge information to be sent to the MS, provided that the SGSN supports Advice of Charge. The operation may be invoked on multiple occasions.

13.14.1.1 Parameters

- sCIGPRSBillingChargingCharacteristics:
This parameter contains the Advice of Charge information:
 - aOCGPRS:
This parameter specifies the Advice of Charge information that shall be sent to the MS. It may contain one or more of the following parameters:
 - aOCInitial:
This is a set of Charge Advice Information elements, as defined in 3GPP TS 22.024 [2]. These CAI elements shall be sent by the gprsSSF to the MS when an Activate PDP Context Accept or Attach Accept is sent to MS and a tariff switch has not yet occurred. It may also be sent at any other time e.g. upon change of QoS or RAI.
 - aOCSubsequent:
This parameter may contain the following information:
 - cAIElements:
This is a set of Charge Advice Information (CAI) elements, as defined in 3GPP TS 22.024 [2]. These CAI elements shall be sent to the MS when an Activate PDP Context Accept or Attach Accept is detected and a tariff switch has occurred previously, or when Activate PDP Context Accept or Attach Accept has previously been detected and a tariff switch occurs.
 - tariffSwitchInterval:
This parameter indicates to the gprsSSF the time duration until the next tariff switch. The measurement of the elapsed tariff switch period shall start immediately after successful execution of this operation.
- pDPID:
This parameter identifies the PDP Context, within the GPRS Session dialogue, for which the Advice-of-Charge instruction applies.

13.14.2 Responding Entity (gprsSSF)

13.14.2.1 Normal Procedure

gprsSSF preconditions:

- (1) A relationship exist between the gsmSCF and the GPRS Session or PDP Context.
- (2) The gprsSSF FSM is in the state "Waiting_for_Instructions" or in the state "Monitoring".

gprsSSF postconditions:

- (1) No gprsSSF FSM state transition.

On receipt of this operation, the gprsSSF performs actions to send the Advice of Charge information to the MS, provided that the SGSN supports Advice of Charge.

If Advice of Charge is to be provided to an MS in conjunction with "CSE control of GPRS Session or PDP Context duration or volume", then the following sequence of operations shall be sent from the gsmSCF to the gprsSSF in the following order and in the same TC TC-CONTINUE or TC-BEGIN component:

ApplyChargingGPRS; SendChargingInformationGPRS.

These operations shall be processed sequentially by the gprsSSF, in the order that they are sent by the gsmSCF. In this, case parameter TariffSwitchInterval may be present either in the ApplyChargingGPRS operation or in the SendChargingInformationGPRS operation, but not in both operations. It is recommended that this parameter be transported in the ApplyGPRSCharging operation.

The TariffSwitchInterval information received with either one of these operations shall set the same tariff switch timer in the gprsSSF. This duration timer shall run from the time of successful operation execution.

13.14.2.2 Error handling

"TaskRefused": In addition to the generic error handling noted below, this error shall be indicated when:

- a tariffSwitchInterval is indicated when a previously received tariffSwitchInterval is pending.

Generic error handling for the operation related errors is described in clause 10 and the TC services which are used for reporting operation errors are described in clause 14.

14 Services assumed from lower layers

14.1 Services assumed from TC

The SS7 application layer protocol defined in this 3GPP TS, is a protocol to provide communication between a pair of application processes. In the SS7 environment this is represented as communication between a pair of application-entities (AEs) using the TC. The function of an AE is provided by a set of application-service-elements (ASEs). The interaction between AEs is described in terms of their use of the services provided by the ASEs.

If AC are to be used for FE differentiation within a physical node, then the version of TC used must support the dialogue portion of TC (ie ETSI ETS 300 287-1 [22]).

This requirement applies to all interfaces, not just those used for internetworking.

Table 14-1 defines which versions of TC are the minimum versions required to support the defined CAP interfaces:

Table 14-1: Minimum TC requirements for CAP interfaces

Interface	CAP
gsmSSF - gsmSCF	White Book
gsmSRF - gsmSCF	White Book
assist gsmSSF - gsmSCF	White Book
smsSSF - gsmSCF	White Book
gprsSSF - gsmSCF	White Book

14.1.1 Common procedures

The present subclause defines the procedures and mapping which apply between CAP and TC to be used in the absence of specific procedures and mapping instructions for the specific CAP interfaces as defined in subsequent subclauses.

14.1.1.1 Normal procedures

The present subclause describes the procedures and TC primitives that shall be used for transmitting messages between AEs under normal operation.

The CAP, as TC-user, uses only the structured dialogue facility provided by TC. The following situations can occur when a message is sent between two PE:

- a dialogue shall be established: the TC-user issues a TC-BEGIN request primitive.
- a dialogue shall be maintained: the TC-user issues a TC-CONTINUE request primitive.
- a dialogue shall no longer be maintained: the TC-user issues a TC-END request primitive with either basic end or with pre-arranged end depending on the following conditions:
 - Basic End
 - In the case the dialogue is established, operations, leading to a termination of the relationship, can be transmitted by the FE with a TC-END request primitive (basic) in the case the FE is not interested in the reception of any ERROR or REJECT components for these sent operations. Once the FE dialogue resources have been released, any ERROR or REJECT components received for these operations will be discarded by TC as described in ETSI ETS 300 287-1 [22].
 - In the case that the dialogue is established and the FE has received an operation, leading to the termination of the relationship, does not wish to continue dialogue and there is no operation to be sent, a TC-END request primitive (basic) with zero components can be sent from the FE.
- Pre-arranged End

- Where an entity is interested in possible ERROR or REJECT messages on response to sent operations leading to a termination of the relationship, the dialogue is ended with a TC-END request primitive (pre-arranged end) after the last associated operation timer expires. The receiving entity can end the dialogue with a TC-END request primitive (pre-arranged end) after successful processing of these operations (i.e. the relationship is terminated).
- in general, the use of prearranged end shall be limited to the case for both communicating entities clearly recognizable that peer entity applies prearranged end. In all other cases, basic end shall be used.

14.1.1.2 Abnormal procedures

The present subclause describes the procedures and TC primitives that shall be used for reporting abnormal situations between AEs. The error cases are defined in clause 10.

The following primitives shall be used to report abnormal situations:

- operation errors, as defined in the CAP, are reported with TC-U-ERROR request primitive.
- rejection of a TC component by the TC-user shall be reported with TC-U-REJECT request primitive.
- when the FE detecting error or rejecting operation decides the termination of TC dialogue, TC-END request primitive (basic) with error or reject can be used for the termination of TC dialogue.
- when the gsmSSF or the gsmSRF detecting error or rejecting operation recognizes the possibility to continue dialogue, TC-CONTINUE request primitive with error or reject can be used for the continuation of TC dialogue.
- a dialogue shall be aborted by the TC-user with a TC-U-ABORT request primitive.
- on expiration of application timer Tssf or Tsrif, dialogue shall be terminated by means of by TC-U-ABORT primitive with an Abort reason, regardless of TC dialogue is established or not.

For abnormal situations detected by TC the same rules shall apply for reception of TC-R-REJECT indication as for transmission of TC-U-REJECT request and for transmission of TC-P-ABORT indication as for transmission of TC-U-ABORT request primitive.

The following rules shall be applied to terminate the TC dialogue under abnormal situations:

- in the case that abort condition is detected and TC dialogue is established, TC dialogue is terminated by TC-U-ABORT primitive with an Abort reason.
- in the case that abort condition is detected and TC dialogue is not established, TC dialogue is locally terminated by TC-U-ABORT primitive. (in the case such as application time out).

In error situations prearranged end shall not be used to terminate the TC dialogue. In the case any AE encounters an error situation the peer entity shall be explicitly notified of the error, if possible. If from any entity's point of view the error encountered requires the relationship to be ended, then it shall close the dialogue via a TC-END request primitive with basic end or via a TC-U-ABORT request primitive, depending on whether any pending ERROR or REJECT component is to be sent or not.

In the case an entity receives a TC-END indication primitive and after all components have been considered, the FSM is not in a state to terminate the relationship, an appropriate internal error should be provided.

In cases when a dialogue needs to be closed by the initiating entity before its establishment has been completed (before the first TC indication primitive to the TC-BEGIN request primitive has been received from the responding entity), the TC-user shall issue a TC-END request primitive with prearranged end or a TC-U-ABORT request primitive. The result of these primitives will be only local, any subsequent TC indication received for this dialogue will be handled in accordance with the abnormal procedures as specified in ETSI ETS 300 287-1 [22]).

When the gsmSSF, gprsSSF or smsSSF receives multiple Operation components in a single TC Message and there is an error in the processing of one of these Operations, then the gsmSSF FSM, gprsSSF FSM or smsSSF FSM shall process the error and shall discard all Operation components in that TC Message of which the processing has not yet started.

14.1.1.3 Dialogue handling

14.1.1.3.1 Dialogue establishment

The establishment of a CAP dialogue involves two application processes as described in clause 1, one that is the dialogue-initiator and one that is the dialogue-responder.

This procedure is driven by the following signals:

- A TC-BEGIN request primitive from the dialogue-initiator.
- A TC-BEGIN indication primitive occurring at the responding side
- The first TC-CONTINUE indication primitive occurring at the initiating side or under specific conditions:
 - A TC-END indication primitive occurring at the initiating side
 - A TC-U-ABORT indication primitive occurring at the initiating side
 - A TC-P-ABORT indication primitive occurring at the initiating side

Sending of a TC-BEGIN request

Before issuing a TC-BEGIN request primitive, TC-USER shall store the AC-name and if present the user-information parameter.

TC-USER shall request the invocation of the associated operations using the TC-INVOKE service. See subclause 14.1.1.4.1 for a description of the invocation procedure.

After processing of the last invocation request, TC-USER shall issue a TC-BEGIN request primitive.

The initiator TC-USER then waits for a TC indication primitive and will not issue any other requests, except a TC-U-ABORT request or a TC-END request with the release method parameter set to "pre-arranged release".

Receipt of a TC-BEGIN indication

On receipt of a TC-BEGIN indication primitive, responder TC-USER shall:

- Analyse the application-context-name included in the primitive. If it is supported, then process any other indication primitives received from TC as described in subclause 14.1.1.4.1.
- If the application-context-name included in the primitive is not supported, then issue a TC-U-ABORT request primitive.

Receipt of the first TC-CONTINUE indication

On receipt of the first TC-CONTINUE indication primitive for a dialogue, TC-USER shall check the value of the application-context-name parameter. If this value matches the one used in the TC-BEGIN request primitive, then TC-USER shall process the following TC component handling indication primitives as described in subclause 14.1.1.4.1, otherwise it shall issue a TC-U-ABORT request primitive.

Receipt of a TC-END indication

On receipt of a TC-END indication primitive in the dialogue initiated state, TC-USER shall check the value of the application-context-name parameter. If this value match the one used in the TC-BEGIN request primitive, then the TC-USER shall process the following TC component handling indication primitives as described in subclause 14.1.1.4.1.

Receipt of a TC-U-ABORT indication

Receipt of a TC-U-ABORT indication primitive is described as part of user abort procedure (see 14.1.1.3.4.)

Receipt of a TC-P-ABORT indication

Receipt of a TC-P-ABORT indication primitive is described as part of provider abort procedure (see 14.1.1.3.5.)

14.1.1.3.2 Dialogue continuation

Once established the dialogue is said to be in a continuation phase.

Both application processes can request the transfer of CAP APDUs until one of them requests the termination of the dialogue.

Sending entity

TC-USER shall process any component handling request primitives as described in subclause 14.1.1.4.1.

After processing the last component handling request primitive, TC-USER shall issue a TC-CONTINUE request primitive.

Receiving entity

On receipt of a TC-CONTINUE indication primitive TC-USER shall accept zero, one or several TC component handling indication primitives and process them as described in subclause 14.1.1.4.1.

14.1.1.3.3 Dialogue termination

Both the dialogue-initiator and the dialogue-responder have the ability to request the termination of a dialogue after it has been established when no dialogue is to be established or when a dialogue is no longer to be maintained according to the rules as stated in subclauses 14.1.2.1.1 and 14.1.2.1.2.

The dialogue termination procedure is driven by the following events:

- A TC-END request primitive
- A TC-END indication primitive

Sending of TC-END request

When the dialogue shall no longer be maintained, TC-USER shall process any component handling request primitives as described in subclause 14.1.1.4.1

After processing the last component handling request primitive (if any), TC-USER shall issue a TC-END request primitive with the release method parameter set to "basic end" or "prearranged release", in accordance with the rules as stated in subclauses 14.1.2.1.1 and 14.1.2.1.2.

When no dialogue is to be established, refer to subclauses 14.1.1.3.1.

Receipt of a TC-END indication

On receipt of a TC-END indication primitive, the TC-USER shall accept any component handling indication primitives and process them as described in subclause 14.1.1.4.1.

After processing the last component handling primitive all dialogue related resources are released.

14.1.1.3.4 User abort

Both the dialogue-initiator and the dialogue-responder have the ability to abort a dialogue at any time.

The user abort procedure is driven by one of the following events:

- A TC-U-ABORT request primitive
- A TC-U-ABORT indication primitive

Sending of TC-U-ABORT request

After issuing a TC-U-ABORT request primitive, all dialogue related resources are released.

Receipt of a TC-U-ABORT indication

On receipt of a TC-U-ABORT indication all dialogue related resources are released.

14.1.1.3.5 Provider abort

TC has the ability to abort a dialogue at both the dialogue-initiator side and the dialogue-responder side.

The provider abort procedure is driven by the following event:

- A TC-P-ABORT indication primitive

Receipt of a TC-P-ABORT indication

On receipt of a TC-P-ABORT indication, all dialogue related resources are released.

14.1.1.3.6 Mapping to TC dialogue primitives

The TC-UNI service is not used by CAP.

The mapping of parameters onto the TC Dialogue services is as follows:

The use of parameters of the TC-BEGIN service is as defined in subclause 14.1.1.3.7 with the following qualifications:

- The Destination Address parameter of the TC-BEGIN service shall be set to the CAP address of the AE which is to respond to the TC-BEGIN service.

NOTE 1: The address used in this parameter may be mapped by SCCP address translation to one of a number of alternative AEs.

- The AC Name parameter of the TC-BEGIN service shall be set according to the specific interface being used between the initiating AE and the responding AE.
- The Originating Address parameter of the TC-BEGIN service shall be set to the unambiguous CAP address of the AE initiating the TC-BEGIN service.

The use of parameters of the TC-CONTINUE service is as defined in subclause 14.1.1.3.7 with the following qualifications:

- The AC Name parameter of the TC-CONTINUE service shall be set to the value of the AC Name parameter of the TC-BEGIN service for the same TC Dialogue ID parameter value.
- If present, the Originating Address parameter of the TC-CONTINUE service shall be set to the unambiguous CAP address of the AE initiating the TC-CONTINUE service. This parameter is present only in the first TC-CONTINUE service after a TC-BEGIN service with the same TC Dialogue ID parameter value.

The use of parameters of the TC-END service is as defined in subclause 14.1.1.3.7 with the following qualifications:

- The AC Name parameter of the TC-END service shall be set to the value of the AC Name parameter of the TC-BEGIN service for the same TC Dialogue ID parameter value. This parameter is present only if the TC-END service is used immediately after the TC-BEGIN service.

The use of parameters of the TC-U-ABORT service is as defined in subclause 14.1.1.3.7 with the following qualifications:

- The Abort Reason parameter of the TC-U-ABORT service shall be used as specified in ETSI ETS 300 287-1 [22].
- The AC Name parameter of the TC-U-ABORT service shall be set to the value used in the TC-BEGIN service.

NOTE 2: This parameter is present only if the TC-U-ABORT is the immediate response to a TC-BEGIN indication.

The use of parameters of the TC-P-ABORT service is as defined in subclause 14.1.1.3.7 with the following qualifications:

- The P-Abort parameter of the TC-P-ABORT service is set by TC to indicate the reason why TC aborted the dialogue. It shall take the values as defined in ETSI ETS 300 287-1 [22].

14.1.1.3.7 Default mapping to TC dialogue parameters

Dialogue Id

The value of this parameter is associated with the CAP invocation in an implementation dependent manner. This parameter uniquely identifies a specific TC dialogue to a remote CAP AE for a CAP AE.

Application-context-name

The application-context-name parameter is set in accordance with the set of Operations which need to be supported by the TC dialogue. The defined AC Names can be found in clauses 6 to 8.

User information

This parameter may be used by both initiating and responding application processes. This parameter shall be used for the CAP-GPRS-ReferenceNumber as defined in 14.1.7. For interfaces other than the gprsSSF-gsmSCF interface and for SMS related messages (as in subclauses 14.1.3, 14.1.4 and 14.1.5) the receiving side may ignore this parameter if received. The User Information parameter shall be encoded in accordance with the definition provided in ITU-T Recommendation Q.773 [46], subclause 3.2, and the definition of EXTERNAL type provided in ITU-T Recommendation X.690 [57], with the restriction that:

- a size (1..10) constraint of SEQUENCE OF EXTERNAL;
- an Object Identifier shall always be present to identify the user information and the entity which sent it;
- a single-ASN-1-type is used for encoding.

For the use of CAP defined TC-U-Abort reason, see the ASN.1 notation in the subclause 5.7.

For the use of CAP defined CAP-GPRS-ReferenceNumber, see subclause 14.1.7. For the abstract syntax of CAP defined CAP-GPRS-ReferenceNumber, see the ASN.1 notation in the subclause 8.1.

Component present

This parameter is used by TC-USER as described in ETSI ETS 300 287-1 [22].

Termination

The value of the release method parameter of the TC-END request primitive is set by TC-USER according to the rules as stated in subclauses 14.1.2.1.1 and 14.1.2.1.2.

Quality of service

The quality of service of TC request primitives is set by the TC-USER to the following value:

- Sequencing requested;
- return option, this parameter is set by TC-USER in an implementation dependent manner.

14.1.1.4 Component handling

14.1.1.4.1 Procedures for CAP Operations

The present subclause describes the procedures for CAP Operations.

Operation invocation

TC-USER shall build an operation argument from the parameters received and request the invocation of the associated operation using the TC-INVOKE procedure. If a linked ID parameter is inserted in the primitive, then this indicates a child operation and implies that the operation is linked to a parent operation.

Operation invocation receipt

On receipt of a TC-INVOKE indication primitive, TC-USER shall

- If the operation code does not correspond to an operation supported by the application-context, then request the transfer of a reject component using the TC-U-REJECT request primitive, with the appropriate problem code (unrecognized operation);
- If a linked ID is included, then perform the following checks: If the operation referred to by the linked ID does not allow linked operations or if the operation code does not correspond to a permitted linked operation, or if the parent operation invocation is not active, then issue a TC-U-REJECT request primitive with the appropriate problem code (linked response unexpected or unexpected linked operation);
- If the type of the argument is not the one defined for the operation, then request the transfer of a reject component using the TC-U-REJECT request primitive, with the appropriate problem code (mistyped parameter);
- if the operation cannot be invoked because the CAP related dialogue is about to be released, requests the transfer of the reject component using the TC-U-REJECT request primitive with the problem code (Initiating Release);
- if sufficient CAP related resources are not available to perform the requested operation, request the transfer of a reject component using the TC-U-REJECT request primitive with the problem code (Resource Limitation);
- Otherwise, accept the TC-INVOKE indication primitive. If the operation is to be user confirmed, then TC-USER waits for the corresponding response.

Operation Response

For user confirmed operations, TC-USER shall:

- If no error indication is included in the response to a class 1 or 3 operation, then construct a result information element from the parameters received and request its transfer using the TC-RESULT-L service.
- If an error indication is included in the response to a class 1 or 2 operation, then construct an error parameter from the parameters received and request its transfer using the TC-U-ERROR request primitive.

Receipt of a response

On receipt of a TC-RESULT-NL indication, TC-USER shall:

- Request the transfer of a reject component using the TC-U-REJECT request primitive, with the appropriate problem code (mistyped parameter).

On receipt of a TC-RESULT-L indication, TC-USER shall:

- if the type of the result parameter is not the one defined for the result of this operation, request the transfer of a reject component using the TC-U-REJECT request primitive, with the appropriate problem code (mistyped parameter);
- otherwise, accept the TC-RESULT-L indication primitive.

On receipt of a TC-U-ERROR indication, TC-USER shall:

- if the error code is not defined for the TC-USER or is not one associated with the operation referred to by the invoke ID, request the transfer of a reject component using the TC-U-REJECT request primitive, with the appropriate problem code (unrecognized error or unexpected error);
- if the type of the error parameter is not the one defined for this error, request the transfer of a reject component using the TC-U-REJECT request primitive, with the appropriate problem code (mistyped parameter);
- Otherwise, accept the TC-U-ERROR indication primitive.

On receipt of a TC-U-REJECT indication primitive which affects a pending operation, TC-USER shall:

- accept the TC-U-REJECT indication primitive.

On receipt of a TC-L-REJECT indicating "return result problem, return error unexpected", TC-USER shall inform the application process.

On receipt of a TC-L-REJECT indicating "return error problem, return error unexpected", TC-USER shall inform the application process.

This event occurs when the local TC detects a protocol error in an incoming component which affects an operation.

When the problem code indicates a general problem, it is considered that the event cannot be related to an active operation even if the invoke Id is provided by TC. This is because it is unclear whether the invoke Id refers to a local or remote invocation. The behaviour of TC-USER in such a case is described in the subclause headed "other events".

On receipt of a TC-L-CANCEL indication, the TC-USER shall:

- if the associated operation is a class 1 operation, inform the application process;
- if the associated operation is a class 2 operation and no linked operations are defined for this operation, ignore the primitive;
- if the associated operation is a class 2 operation and has linked operations but none of them has been invoked, inform the application process;
- if the associated operation is a class 2 operation and a linked operation invocation has already been received in response to this operation, ignore the primitive;
- if the associated operation is a class 3 operation, inform the application process;
- if the associated operation is a class 4 operation, ignore the primitive;

Other events

This subclause describes the behaviour of TC-USER on receipt of a component handling indication primitive which cannot be related to any operation or which does not affect a pending one.

On receipt of a TC-U-REJECT indication primitive which does not affect an active operation (i.e. indicating a return result or return error problem), it is up to the application process to abort, continue or terminate the dialogue, if not already terminated by the sending application process according to the rules as stated in subclause 14.1.2.1.2. This is also applicable for invoke problems related to a class 4 linked operation.

On receipt of a TC-R-REJECT indication (i.e. when a protocol error has been detected by the peer TC entity) which does not affect an active operation, it is up to the application process to abort, continue or terminate the dialogue, if not already terminated by the sending application process according to the rules as stated in subclause 14.1.2.1.2.

On receipt of a TC-L-REJECT indication primitive (i.e. when a protocol error has been detected by the local TC entity) which cannot be related to an active operation, it is up to the application process to continue, or to terminate the dialogue and implicitly trigger the transmission of the reject component or to abort the dialogue.

On receipt of a TC-NOTICE indication primitive, which informs the TC-USER that a message cannot be delivered by the Network Layer, it is for the application process to decide whether to terminate the dialogue or retry.

This primitive can occur only if the Return Option has been set (see subclause 14.1.1.3.6).

14.1.1.4.2 Mapping to TC component primitives

The mapping of parameters onto the TC Component services is as follows:

The TC-U-CANCEL service is not used.

The TC-RESULT-NL service is not used.

The use of parameters of the TC-INVOKE service is as defined in subclause 14.1.1.4.3 with the following qualifications:

- The Operation parameter of the TC-INVOKE service shall contain the *operation.&operationCode* value of the CAP Operation to be invoked. The operation must be one of the valid operations supported by the AC for the TC dialogue and must be invocable by the local AE.
- The Parameters parameter of the TC-INVOKE service shall contain a value of the *operation.&ArgumentType* value for the operation being invoked, as specified by the Operation parameter.

The use of parameters of the TC-RESULT-L service is as defined in subclause 14.1.1.4.3 with the following qualifications:

- The Invoke Id parameter of the TC-RESULT-L service shall be set to the value of the Invoke Id parameter of the TC-INVOKE service from the remote AE to which a result is being sent.
- The Operation parameter of the TC-RESULT-L service be set to the value of the Operation parameter of the TC-INVOKE service from the remote AE which contains the same Invoke Id Parameter value.
- The Parameters parameter of the TC-RESULT-L service shall contain the *operation.&ResultType* value for the operation result, as specified by the Operation parameter.

The use of parameters of the TC-U-ERROR service is as defined in subclause 14.1.1.4.3 with the following qualifications:

- The Invoke Id parameter of the TC-U-ERROR service shall be set to the value of the Invoke Id parameter of the TC-INVOKE service from the remote AE to which an error is being sent.
- The Error parameter of the TC-U-ERROR service shall be set to the value of the *error.&errorCode* of the error to be sent. It must be one of the errors which is expected for the invoked operation as defined in the *operation.&Errors* specification.
- The Parameters parameter of the TC-U-ERROR service shall be set to the value of the *error.&ParameterType* of the error to be sent, as identified by the Error parameter.

The use of parameters of the TC-U-REJECT service is as defined in subclause 14.1.1.4.3 with the following qualifications:

- The Invoke Id parameter of the TC-U-REJECT service shall be set to the Invoke Id Parameter of the TC component service from the remote AE which is being rejected.

The use of parameters of the TC-L-CANCEL service is as defined in subclause 14.1.1.4.3.

14.1.1.4.3 Default mapping to TC component parameters

Invoke Id

This parameter is set by the sending application process. It represents the unique identity of an instance of an operation which is invoked by an AE within a specific TC dialogue. The TC dialogue is identified by the Dialogue Id parameter.

Linked Id

This parameter is set by the sending application process. It represents the Invoke Id of an operation which was received from the remote AE for a specific TC dialogue to which the operation being invoked by the local AE is to be linked. This parameter is present only if the original operation invoked by the remote AE is defined as having linked operations. The type of local operation invoked must be the same type as one of the operations defined as being linked.

Dialogue Id

The value of this parameter is associated with the CAP invocation in an implementation dependent manner. It represents the identity of the established TC dialogue which will carry the component services between the local AE and the remote AE.

Class

The value of this parameter is set according to the type of the operation to be invoked according to the operation definitions in clauses 6 through 8.

Time out

The value of this parameter is set according to the type of operation invoked.

Last component

This parameter is used as described in ETSI ETS 300 287-1 [22].

Problem code

This parameter is used as described in subclause 14.1.1.4.1.

Abort reason

This parameter is used by TC-USER, and attributes and coding are specified by network operator.

14.1.2 gsmSSF-gsmSCF interfaces

14.1.2.1 Normal procedures

14.1.2.1.1 gsmSSF-to-gsmSCF messages

The present subclause defines the normal procedures for TC messages from the gsmSSF to the gsmSCF.

gsmSSF FSM related messages

A dialogue shall be established when the gsmSSF FSM transits from the state "Idle" to the state "Waiting_for_Instructions". The InitialDP operation shall be transmitted in the same message.

The CAP Operation InitialDP shall be sent with a TC-BEGIN request primitive.

For all other operations sent from the gsmSSF FSM, the dialogue shall be maintained except for the following cases.

When the gsmSSF FSM executes a non-error case state transition to the state "Idle" and there is one or more pending operation and TC dialogue is established, TC dialogue can be terminated by TC-END primitive with component(s). When the gsmSSF sends the last EventReportBCSM, ApplyChargingReport or CallInformationReport the dialogue may be ended from the gsmSSF by a TC-END request primitive with basic end.

In the case that there is no pending operation and TC dialogue is established, TC dialogue can be terminated by TC-END primitive with zero component or prearranged end. When the gsmSSF FSM makes a non-error case state transition to the state "Idle" and there is no operation to be sent, the dialogue is ended by means of a TC-END request primitive (basic) with zero components, or the dialogue is locally ended by means of a TC-END request primitive with prearranged end.

In the case where a call release is initiated by any other entity than an gsmSCF, the gsmSSF can end a dialogue with a TC-END request primitive with zero component or prearranged end if a TC dialogue is established and the gsmSSF has no pending call information requests (or pending requests which should be treated in the same way, see subclause 14.1.1.1) nor any armed EDP.

When the gsmSSF has sent the last EventReportBCSM, ApplyChargingReport or CallInformationReport the dialogue may be ended from the gsmSCF by a TC-END request primitive with basic end.

Assisting gsmSSF FSM related messages

A dialogue shall be established when the assisting gsmSSF FSM transits from the state "Idle" to the state "Waiting_for_Instructions". The AssistRequestInstructions operation shall be transmitted with a TC-BEGIN request primitive.

For all other operations sent from the assisting gsmSSF FSM, the dialogue shall be maintained except for the following cases.

When the assisting gsmSSF FSM makes a non-error case state transition to the state "Idle" and there is one or more pending operation and TC dialogue is established, TC dialogue can be terminated by TC-END primitive with component(s).

In the case that there is no pending operation and TC dialogue is established, TC dialogue can be terminated by TC-END primitive with zero component or prearranged end. When the assisting gsmSSF FSM makes a non-error case state transition to the state "Idle" and there is no operation to be sent, the dialogue is ended by means of a TC-END request primitive (basic) with zero components, or the dialogue is locally ended by means of a TC-END request primitive with prearranged end.

gsmSSME FSM related messages

The following procedures shall be followed:

- The dialogue shall be maintained when the ActivityTest Return Result is sent.

14.1.2.1.2 gsmSCF-to-gsmSSF messages

The present subclause defines the normal procedures for TC messages from the gsmSCF to the gsmSSF.

SCSM-FSM related messages

A dialogue shall be established when the SCSM-FSM receives of InitialDP operation for TDP-R or AssistRequestInstructions operation.

For subsequent operations sent from the SCSM-FSM, the dialogue shall be maintained except for the following cases, i.e. all other operations are sent after a dialogue was established from the gsmSSF (the gsmSCF has previously received a TC-BEGIN indication primitive with an InitialDP operation or an AssistRequestInstructions operation).

The dialogue shall no longer be maintained when the prearranged end condition is met in the gsmSCF. When the gsmSCF does not expect any messages other than possibly REJECT or ERROR messages for the operations sent and when the last associated operation timer expires, the dialogue is locally ended by means of a TC-END request primitive with prearranged end.

Alternatively, the sending of operations, leading to the termination of the relationship, by means of a TC-END request primitive (basic end) is possible.

SCME-FSM related messages

The operations sent from the SCME-FSM shall be issued according to the following procedures:

- The dialogue shall be maintained when the ActivityTest operation is sent.
- For sending one or more CallGap operations, the SCME FSM shall use an existing SCSM FSM associated dialogue which was initiated by a gsmSSF FSM (i.e. established for the transmission of the InitialDP operation). The dialogue shall be maintained.

14.1.2.1.3 smsSSF -to-gsmSCF SMS related messages

A dialogue shall be established when the smsSSF has finalised trigger processing and transits to the state "Waiting_for_Instructions". The relevant CAP Operation, which can be the InitialDPSMS operation only, shall be transmitted in the same message.

For all other operations sent from the smsSSF, the dialogue shall be maintained.

The dialogue shall no longer be maintained when the prearranged end condition is met in the smsSSF. When the smsSSF makes a state transition to the state "Idle", the dialogue is locally ended by means of a TC-END request primitive with prearranged end.

When the smsSSF has sent the last EventReportSMS operation the dialogue may be ended from the gsmSCF by a TC-END request primitive with basic end. If the smsSSF decides to apply basic end, then it shall send TC-END with zero components.

14.1.2.1.4 gsmSCF-to-smsSSF SMS related messages

All operations are sent after a dialogue was established from the smsSSF (the gsmSCF has previously received a TC-BEGIN indication primitive with an InitialSMSEvent operation).

The dialogue shall no longer be maintained when the prearranged end condition is met in the gsmSCF. When the gsmSCF does not expect any messages other than possibly REJECT or ERROR messages for the operations sent and when the last associated operation timer expires, the dialogue is locally ended by means of a TC-END request primitive with prearranged end.

Alternatively, the sending of operations, leading to the termination of the control relationship, by means of a TC-END request primitive (basic end) is possible.

14.1.2.1.5 Use of dialogue handling services

Dialogue handling services are used to trigger the sending of the APDUs associated with the operations involved in the CAP packages.

Component grouping is performed under the control of the application-process through an appropriate usage of the TC-BEGIN, TC-CONTINUE and TC-END service.

14.1.2.2 Abnormal procedures

The following procedures also apply to the gsmSCF-gsmSRF interfaces.

14.1.2.2.1 gsmSCF-to-gsmSSF/gsmSRF messages

Considering that gsmSSF and gsmSRF do not have the logic to recover from error cases detected on the gsmSCF-gsmSSF/gsmSRF interface, the following shall apply:

- Operation errors and rejection of TC components shall be transmitted to the gsmSSF and, respectively, the gsmSRF with a TC-END request primitive, basic end.

If, in violation of the above procedure, an ERROR or REJECT component is received with a TC-CONTINUE indication primitive, then the gsmSSF and, respectively, the gsmSRF shall abort the dialogue with a TC-U-ABORT request primitive.

14.1.2.2.2 gsmSSF/gsmSRF/ -to-gsmSCF messages

Operation errors and rejection of TC components shall be transmitted to the gsmSCF according to the following rules:

- The dialogue shall be maintained when the preceding message, which contained the erroneous component, indicated that the dialogue shall be maintained. I.e. the error or reject shall be transmitted with a TC-CONTINUE request primitive if the erroneous component was received with a TC-CONTINUE indication primitive.
On receipt of an ERROR or REJECT component the gsmSCF decides on further processing. It may either continue, explicitly end or abort the dialogue.
- In all other situations the dialogue shall no longer be maintained. I.e. the error or reject shall be transmitted with a TC-END request primitive, basic end, if the erroneous component was received with a TC-BEGIN indication primitive.
- on expiration of application timer Tssf or Tsrif, dialogue shall be terminated by means of by TC-U-ABORT primitive with an Abort reason, regardless of TC dialogue is established or not.

If the error processing in the gsmSSF or gsmSRF leads to the case where the gsmSSF or gsmSRF is not able to process further gsmSCF operations while the dialogue is to be maintained, then the gsmSSF or gsmSRF aborts the dialogue with a TC-END request primitive with basic end or a TC-U-ABORT request primitive, depending on whether any pending ERROR or REJECT component is to be sent or not.

The gsmSSF can end a dialogue with a TC-U-ABORT request primitive in the case that call release is initiated by any other entity then the gsmSCF and the gsmSSF has no pending call information requests (or pending requests which should be treated in the same way, i.e., ApplyCharging nor any armed EDP to notify the gsmSCF of the call release (for alternative way, see subclause 14.1.2.1.1).

14.1.2.2.3 gsmSCF-to-smsSSF SMS related messages

Considering that the smsSSF does not have the logic to recover from error cases detected on the gsmSCF-smsSSF interface, the following shall apply:

- operation errors and rejection of TC components shall be transmitted to the smsSSF with a TC-END request primitive, basic end.

If, in violation of the above procedure, an ERROR or REJECT component is received with a TC-CONTINUE indication primitive, then the smsSSF shall abort the dialogue with a TC-U-ABORT request primitive.

14.1.2.2.4 smsSSF-to-gsmSCF SMS related messages

Operation errors and rejection of TC components shall be transmitted to the gsmSCF according to the following rules:

- the dialogue shall be maintained when the preceding message, which contained the erroneous component, indicated that the dialogue shall be maintained. I.e. the error or reject shall be transmitted with a TC-CONTINUE request primitive if the erroneous component was received with a TC-CONTINUE indication primitive;
- on receipt of an ERROR or REJECT component the gsmSCF decides on further processing. It may either continue, explicitly end or abort the dialogue;

If the error processing in the smsSSF leads to the case where the smsSSF is not able to process further gsmSCF operations while the dialogue is to be maintained, then the smsSSF aborts the dialogue with a TC-U-ABORT request primitive.

The smsSSF aborts a dialogue with a TC-U-ABORT request primitive if release is initiated by any other entity than the gsmSCF and the smsSSF has no armed EDPs to notify the gsmSCF.

14.1.2.2.5 Use of dialogue handling services

On receipt of a TC-U-REJECT.ind in the FE, this primitive should be ignored. It is up to the application process to abort, continue or terminate the dialogue, if not already terminated by the sending application process according to the rules as stated in subclause 14.1.1.2. This is also applicable for invoke problems related to a class 4 linked operation.

A TC-U-REJECT.req should be sent followed by a TC-CONTINUE.req.

On receipt of a TC-R-REJECT.ind in the FE, this primitive should be ignored. It is up to the application process to abort, continue or terminate the dialogue, if not already terminated by the sending application process according to the rules as stated in subclause 14.1.1.2. This is also applicable for invoke problems related to a class 4 linked operation.

On receipt of a TC-L-REJECT indication primitive (i.e. when a protocol error has been detected by the local TC entity) which cannot be related to an active operation, it is up to the application process to continue or to terminate the dialogue and implicitly trigger the transmission of the reject component or to abort the dialogue.

On receipt of a TC-NOTICE indication the TC-USER is informed that a message cannot be delivered by the Network Layer. It occurs if the Return Option has been set (see subclause 14.1.1.3.7). It is for the application process to decide whether to terminate the dialogue or retry.

The application-process is the sole user of the TC-P-ABORT service and TC-NOTICE service.

The receipt of a TC-U-ABORT-Ind or TC-P-ABORT-Ind on a dialogue terminates all request processing.

14.1.2.3 Dialogue handling

14.1.2.3.1 Dialogue establishment

14.1.2.3.2 Dialogue continuation

14.1.2.3.3 Dialogue termination

14.1.2.3.4 User abort

14.1.2.3.5 Provider abort

14.1.2.3.6 Mapping to TC dialogue primitives

The gsmSSF-gsmSCF IN services can be mapped onto TC services. The present subclause defines the mapping of the gsmSSF-gsmSCF IN services onto the services of the TC dialogue handling services defined in ETSI ETS 300 287-1 [22].

- a) The TC-BEGIN service is used to invoke the operations of the gsmSCF-gsmSSF connection packages as defined in clause 6.

- b) The TC-CONTINUE service is used to report the success of the operations invoked in a TC-BEGIN service and to invoke or respond to any other operations.
- c) The TC-U-ABORT service is used to report the failure of operations of the connection packages as defined in clause 6.

The mapping of the parameters onto the TC-BEGIN primitive is defined in subclause 14.1.1.3.6 with the following qualifications:

- The AC Name parameter shall take the value of the application-context-name field of the cap3-sms-AC object if the initiating AE is a gsmSSF.

The mapping of the parameters onto the TC-CONTINUE primitive is defined in subclause 14.1.1.3.6.

The mapping of the parameters onto the TC-U-ABORT primitive is defined in subclause 14.1.1.3.6 with the following qualifications:

- The Application-Context-Name parameter shall be used as specified in ETSI ETS 300 287-1 [22]. When the responding AE refuses a dialogue because the application-context-name it receives is not supported, this parameter shall have the value of the application-context-name field of the cap3-sms-AC object if the responding AE is a gsmSCF.

The use of the parameters of the TC-END service is defined in subclause 14.1.1.3.6.

14.1.2.4 Component Handling

14.1.2.4.1 Procedures for CAP Operations

The CAP ASEs are users of the TC component handling services except for the TC-L-REJECT and TC-L-CANCEL services which are used by the application-process. Receipt of a TC-L-REJECT-Ind leads the application-process to abandon the dialogue (i.e. it issues a TC-U-ABORT-Request primitive).

The TC-U-CANCEL service is never used.

14.1.2.4.2 Mapping to TC component parameters

The gsmSSF-gsmSCF IN ASE services are mapped onto the TC component handling services. The mapping of operations and errors onto TC services is defined in subclause 14.1.1.4.2 with the following qualifications:

The timeout parameter of the TC-INVOKE-Req primitives is set according to clause 6.

14.1.3 gsmSCF-gsmSRF interface

14.1.3.1 Normal procedures

14.1.3.1.1 gsmSCF-to/from-gsmSRF messages

A dialogue is established when the gsmSRF sends an AssistRequestInstructions operation to the gsmSCF. For all other operations sent to/from the gsmSRF, the dialogue shall be maintained.

In the case that there is no pending operation and TC dialogue is established, TC dialogue can be terminated by TC-END primitive with zero component. When the SCSM makes a non-error case state transition to end-user interaction and there is no operation to be sent, the dialogue is ended by means of a TC-END request primitive (basic) with zero components.

The dialogue shall no longer be maintained when sending the SpecialisedResourceReport operation for PlayAnnouncement with disconnection from the gsmSRF set to true or Return Result of the PromptAndCollectUserInformation with disconnection from the gsmSRF set to true with disconnection from the gsmSRF set to true. The dialogues is ended by means of a TC-END request primitive with basic end, and the one of above operations is transmitted with the same request.

Regardless of whether pending operation exists or not, when the SRSM-FSM is informed of the disconnection of bearer connection (in the case of gsmSCF initiated disconnection or call abandon from call party) and dialogue is established, the dialogue is ended by means of a TC-END request primitive (basic) with zero components or TC-END request primitive (prearranged end).

The dialogue shall no longer be maintained when the prearranged end condition is met in the gsmSRF. When the SRSM-FSM is informed the disconnection of bearer connection and TC dialogue is not established, TC dialogue is locally terminated by TC-END primitive with prearranged end.

When the gsmSCF does not expect any messages other than possibly REJECT or ERROR messages for the operations sent and when the last associated operation timer expires, the dialogue is locally ended by means of a TC-END request primitive with prearranged end. Alternatively, the sending of operations, leading to the termination of the relationship, by means of a TC-END request primitive (basic end) is possible.

In the relay case, the gsmSRF-gsmSCF relationship uses the gsmSSF-gsmSCF TC dialogue. This is possible, because begin and end of the gsmSRF-gsmSCF relationship are embedded in the gsmSSF-gsmSCF relationship. gsmSRF-gsmSCF information shall be exchanged with TC-CONTINUE request primitives.

14.1.3.1.2 Abnormal procedures

14.1.3.1.3 Dialogue handling

- 14.1.3.1.3.1 Dialogue establishment
- 14.1.3.1.3.2 Dialogue continuation
- 14.1.3.1.3.3 Dialogue termination
- 14.1.3.1.3.4 User abort
- 14.1.3.1.3.5 Provider abort
- 14.1.3.1.3.6 Mapping to TC dialogue primitives

The gsmSCF-gsmSRF IN services can be mapped onto TC services. The present subclause defines the mapping of the gsmSCF-gsmSRF IN services onto the services of the TC dialogue handling services defined in ETSI ETS 300 287-1 [22].

- a) The TC-BEGIN service is used to invoke the operations of the gsmSRF-gsmSCF connection packages as defined in clause 6.
- b) The TC-CONTINUE service is used to report the success of the operations invoked in a TC-BEGIN service and to invoke or respond to any other operations.
- c) The TC-U-ABORT service is used to report the failure of operation of the gsmSCF-gsmSRF operations packages as defined in clause 6.

The mapping of parameters onto the TC Dialogue services is as defined in subclause 14.1.1.3.6 with the following qualifications:

The mapping of the parameters onto the TC-BEGIN primitive is defined in subclause 14.1.1.3.6 with the following qualifications:

- The AC Name parameter shall take the value of the application-context-name field of the gsmSRF-gsmSCF-ac object.

14.1.4.2 Component handling

14.1.4.2.1 Procedures for CAP Operations

14.1.4.2.2 Mapping to TC component parameters

The mapping of parameters for the TC component services is defined in subclause 14.1.1.4.2 with the following qualifications.

The Timeout Parameter of the TC-INVOKE service is set according to clauses 6.

14.1.4 gprsSSF-gsmSCF interface

14.1.4.1 Normal procedures

14.1.4.1.1 TC-dialogues and relationships

The GPRS dialogue can consist of multiple consecutive TC-dialogues. A GPRS dialogue is identified by a GPRS-ReferenceNumber consisting of the originationReference and the destinationReference. One GPRS-Reference is assigned by the SGSN and shall be unique within this SGSN. The other GPRS-Reference is assigned by the gsmSCF and shall be unique within this gsmSCF.

The *TC*-dialogues are closed and (re)opened whenever necessary.

14.1.4.1.2 Use of the GPRS Reference

For the use of CAP defined GPRS-ReferenceNumber, see also the ASN.1 notation in the subclause 8.1.

When the gprsSSF sends the first operation for a new GPRS dialogue (InitialDPGPRS), the gprsSSF shall include a GPRS Reference Number in the TC message. This GPRS Reference Number shall consist of the *SGSN Process Id* as originationReference, which is internally allocated by the gprsSSF. This number is used by the gprsSSF to associate an incoming TC message with an internal GPRS Process.

When the gsmSCF has received the InitialDPGPRS operation, it shall store the SGSN Process ID and allocate an *SCF Process Id* which is used by the gsmSCF to associate an incoming TC message with an internal gsmSCF Process.

The gsmSCF shall include the GPRS Reference Number in the first TC-CONTINUE message, *SGSN Process Id* in destinationReference and *SCF Process Id* in originationReference, returned to the gprsSSF.

When the gprsSSF receives the first TC message from the gsmSCF for this GPRS dialogue, the gprsSSF shall store the gsmSCF Process Id together with the SGSN Process Id.

From here onwards all the TC messages that open a new TC dialogue shall include the GPRS Reference Number consisting of the originationReference and the destinationReference to associate the internal process in the origination entity and the destination entity, respectively, until the end of the relationship between these processes.

For any TC-CONTINUE in the existing TC dialogue, transporting the GPRS Reference Number is not needed except for the first response after the InitialDPGPRS operation.

14.1.4.1.3 gprsSSF-to-gsmSCF messages

The present subclause defines the normal procedures for TC messages from the gprsSSF to the gsmSCF.

gprsSSF FSM related messages

A GPRS dialogue and a TC dialogue shall be established when the gprsSSF FSM transits from the state "Idle" to the state "Waiting_for_Instructions". The InitialDPGPRS operation shall be transmitted in the same TC message, i.e. TC-BEGIN. It shall contain the GPRS-Reference as assigned by the SGSN in the originationReference. The gprsSSF may initiate the subsequent TC dialogues for this GPRS dialogue with the following operations:

- ApplyChargingReportGPRS

- EntityReleasedGPRS
- EventReportGPRS

For the establishment of a new TC dialogue within the context of the current GPRS dialogue, the gprsSSF may apply one of the following mechanisms:

- (1) the gprsSSF shall memorise the gsmSCF address used in the first response message to the InitialDPGPRS and use it to open the new TC dialogue;
- (2) the gprsSSF shall use the gsmSCF address from GPRS-CSI to open the new TC dialogue.

The gsmSCF shall memorise the gprsSSF address received along with the InitialDPGPRS and use it for the opening of new TC dialogues within the context of the current GPRS dialogue.

The gsmSCF may open subsequent TC dialogues with the following CAP Operations:

- ActivityTestGPRS;
- ApplyChargingGPRS;
- CancelGPRS;
- FurnishChargingInformationGPRS;
- ReleaseGPRS;
- RequestReportGPRSEvent;
- SendChargingInformationGPRS.

The CAP Operation that opens a TC dialogue shall be sent with a TC-BEGIN request primitive. This message shall contain the GPRS-ReferenceNumber assigned by the sender of this message in the originationReference. If the operation opens a subsequent TC dialogue, then this message shall contain also the previously received destinationReference. If an operation opens a GPRS dialogue, then the TC message reply shall contain the originationReference as assigned by the sender, i.e. the gsmSCF.

The TC dialogue shall be closed for the idle periods, i.e. when the gprsSSF FSM transits from the state "Waiting_for_Instructions" to the state "Idle", if the gprsSSF FSM is in the state "Monitoring" and has received all replies or time-outs for the operations sent, after standalone operations of the gsmSCF in Monitoring state if gprsSSF FSM does not transit to the state "Idle" (ActivityTestGPRS, ApplyChargingGPRS, CancelGPRS, FurnishChargingInformationGPRS, RequestReportGPRSEvent, SendChargingInformationGPRS), or at the end of a GPRS dialogue.

Each TC dialogue shall be terminated by the gprsSSF using TC-END (basic end). The following operations can cause the end of the GPRS dialogue:

- ContinueGPRS;
- ConnectGPRS;
- ApplyChargingReportGPRS result;
- EntityReleasedGPRS result;
- EventReportGPRS (EDP-N) result;
- CancelGPRS;
- ReleaseGPRS;
- RequestReportGPRSEvent (disarming of DPs).

When the gprsSSF FSM makes a non-error case state transition to the state "Idle" and there is one or more pending operation and TC dialogue is established, TC dialogue may be terminated by TC-END primitive with zero component(s) after all pending operations have been sent. When the gprsSSF sends the last EventReportGPRS, EntityreleasedGPRS or ApplyChargingReportGPRS, then after reception of the result or error, the GPRS dialogue may be ended from the gprsSSF by a TC-END request primitive with basic end.

In the case that there is no pending operation, result nor error, and TC dialogue is established, TC dialogue shall be terminated by a TC-END primitive with zero component.

In the case where a PDP Context release or detach is initiated by any other entity than an gsmSCF, the gprsSSF shall end a GPRS dialogue with the EntityReleasedGPRS operation if the gprsSSF has no armed DP to report nor pending ApplyChargingReportGPRS which should be reported.

In the case of overlapping dialogues for the same GPRS dialogue the gsmSCF opened TC dialogue is aborted by the gprsSSF with the abort reason overlapping-dialogue as specified in subclause 5.7. This abort reason is used to indicate to the gsmSCF that a specific instance already has a TC dialogue open. It is typically obtained when both the gsmSCF and gprsSSF open a new dialogue at the same time. While the gprsSSF waits for a response to an operation sent in TC-BEGIN it may receive an operation from the gsmSCF in TC-BEGIN. In such cases the dialogue opened by the gprsSSF is maintained and the dialogue opened by the gsmSCF is aborted with this abort reason.

gprsSSME FSM related messages

The following procedures shall be followed:

- The TC dialogue shall be terminated by a TC-END primitive with zero components after the ActivityTestGPRS Return Result is sent.

14.1.4.1.4 gsmSCF-to-gprsSSF messages

The present subclause defines the normal procedures for TC messages from the gsmSCF to the gprsSSF.

In the case of overlapping dialogues for the same relationship the gsmSCF opened dialogue is closed by the gprsSSF as specified in subclause 5.7. The gsmSCF shall first respond normally to the operations sent by the gprsSSF, and then decide on the further actions.

SCME-FSM related messages

The operations sent from the SCME-FSM shall be issued according to the following procedures:

- A new subsequent TC dialogue is established when the ActivityTestGPRS operation is sent.

14.1.4.2 Abnormal procedures

14.1.4.2.1 gsmSCF-to-gprsSSF messages

The present subclause defines the abnormal procedures for TC messages from the gsmSCF to the gprsSSF.

Considering that gprsSSF do not have the logic to recover from error cases detected on the gsmSCF-gprsSSF interface, the following shall apply:

- Operation errors and rejection of TC components shall be transmitted to the gprsSSF with a TC-END request primitive, basic end.
- The GPRS dialogue shall be closed.

If, in violation of the above procedure, an ERROR or REJECT component is received with a TC-CONTINUE indication primitive, then the gprsSSF shall abort the dialogue with a TC-U-ABORT request primitive.

14.1.4.2.2 gprsSSF-to-gsmSCF messages

The present subclause defines the abnormal procedures for TC messages from the gprsSSF to the gsmSCF.

Operation errors and rejection of TC components shall be transmitted to the gsmSCF according to the following rules:

- The TC dialogue shall be maintained when the preceding message, which contained the erroneous component, indicated that the dialogue shall be maintained. I.e. the error or reject shall be transmitted with a TC-CONTINUE request primitive.
On receipt of an ERROR or REJECT component the gsmSCF decides on further processing. It may either continue, explicitly end or abort the TC dialogue. If the TC dialogue is closed due to such error, then the GPRS dialogue shall also be closed.

- on expiration of application timer Tssf, the TC dialogue shall be terminated by means of by TC-U-ABORT primitive with an Abort reason. The GPRS dialogue shall be closed.

If the error processing in the gprsSSF leads to the case where the gprsSSF is not able to process further gsmSCF operations while the TC dialogue is to be maintained, then the gprsSSF aborts the TC dialogue with a TC-END request primitive with basic end or a TC-U-ABORT request primitive, depending on whether any pending ERROR or REJECT component is to be sent or not.

The gprsSSF can end a TC dialogue with a TC-U-ABORT request primitive in the following case:

- Any entity other than the gsmSCF initiates closure of the GPRS dialogue, and
- The gprsSSF has no pending reports, and
- The gprsSSF has no armed EDP to notify the gsmSCF that the GPRS dialogue has been closed.

For an alternative method, see subclause 14.1.7.1.1.

14.1.4.2.3 Default GPRS Handling

If a TC dialogue is closed due to unrecoverable TC/protocol error (does not apply to the overlapping TC dialogues), or aborted by the gsmSCF, or at the Tssf expiry, then the gprsSSF shall check the applicable Default GPRS Handling parameter of the GPRS-CSI. In this context the applicable Default GPRS Handling is the one that corresponds the TDP that opened the GPRS dialogue. The same default handling shall apply to all state models that are controlled by the particular GPRS dialogue.

14.2 Services assumed from SCCP

The present subclause describes the services required from the SCCP that may be used by the CAMEL applications for the CAMEL Application Part (CAP) used between the gmsSSF, assisting gmsSSF, gmsSRF, gprsSSF, and gsmSCF.

The following SCCP revisions are supported by CAP version 3:

- Signalling Connection Control Part, Signalling System no. 7 CCITT ("Blue Book SCCP")
- Signalling Connection Control Part, Signalling System no. 7 ITU-T Recommendation Q.711 to Q.716 ("White Book SCCP")

NOTE: Support of White Book SCCP at the receiving side shall be mandated from 00:01hrs, 1st July 2002(UTC).

- ANSI T1.112-1996 [91]: "American National Standards for Telecommunications– Signalling System Number 7 (SS7) – Signalling Connection Control Part (SCCP)".

When CAP uses White Book SCCP to send a message, and SCCP segments the message into one or more XUDT messages, then the transmission of this message may fail.

Failure will occur when the destination SCCP, or any intermediate SCCP, does not support White Book SCCP.

Support of ANSI T1.112-1996 [91] SCCP applies to PLMNs in North America only. Interworking between a PLMN in North America and a PLMN outside North America will involve a STP to translate between ANSI SCCP and ITU-T/CCITT SCCP.

14.2.1 Normal procedures

The SCCP forms the link between the TC and the MTP and provides (in conjunction with the MTP) the network services for the CAMEL applications. The network services provided allow the signalling messages sent by the application to the lower layers to be successfully delivered to the peer application.

14.2.2 Service functions from SCCP

14.2.2.1 SCCP connectionless services

The services described are those given in the SCCP ITU-T recommendations Q.711 to Q.716 should be consulted to identify possible interworking and compatibility issues between the different SCCP versions.

The following Connection-less services are expected from the SCCP:

- a) Network Addressing to enable signalling connections between SCCP users;
- b) Sequence Control to enable the SCCP users to invoke "sequence guaranteed" or "sequence not guaranteed" options for a given stream of messages to the same destination;
- c) Segmentation/reassembly of large user messages (for "White Book SCCP" only);
- d) Return Option to enable the SCCP users to invoke "discard message on error" or "return message on error" for a given message not able to be delivered by the SCCP to the destination SCCP user, due to routing or segmentation/re-assembly failure;
- e) Congestion control.

The primitives used for the above services are given below.

The N-UNITDATA request and N-UNITDATA indication primitives are used to send and receive data. The parameters of these primitives include the Called and Calling Addresses, Sequence Control, Return Option and User Data with the addressing parameters always mandatory.

The N-NOTICE indication primitive is used to return undelivered data if return option is set and a routing/segmentation error occurs.

14.2.2.1.1 Sub-System Number (SSN)

The use of SSN is a network operator option and values for intra-PLMN usage are network specific. A CAP SSN has been reserved for inter-PLMN use, as defined in 3GPP TS 23.003 [5].

14.2.2.1.2 Addressing

The addressing elements consist of information contained within the Calling and the Called Party Addresses which are sent by the application to the lower layers.

The application expects the SCCP to route messages by either (a) the use of the Destination Point Code (DPC) plus the Subsystem Number (SSN), or (b) the use of the GT plus optionally the SSN. The application also specifies to the lower layer whether to route the message on the DPC or the GT.

Method (a) above may be used when the application is aware of the destination point code and the destination SSN located at that point code to which the message is to be delivered. Within a national network different SSNs, according to ITU-T Recommendation Q.713 [42], may be allocated for the different network specific applications, e.g. a SSN may be allocated for a gsmSCF functionality.

Method (b) above may be used when a message is to be delivered to a SCCP-user which can be identified by the combination of the elements within the GT. An example of the use of this method is when messages have to be delivered between different networks. This method may be used since the originating network is unaware of the point code and SSN's allocations within the destination network. The network that determines the end-node to which the message is to be delivered has to perform a GT Translation to derive the destination Point Code and the SSN. If optionally the original address contained the SSN, then this may be used as the destination SSN, or the translation may, if required, provide an appropriate new SSN.

When GT is used for addressing, the CAMEL application expects that the SCCP supports the following elements as defined in ITU-T Recommendation Q.713 [42]:

Address Indicator:

The application will set this indicator to indicate one or any combination of the elements "signalling point code, GT, subsystem number" in the address information octets.

GT Indicator:

This indicator specifies the method employed for the formatting of the address information. There are four values (1 to 4), for example, the value 4 indicates that the format includes the numbering plan, the nature of the address indicator and the translation type. The format with the indicator value 4 is always used for internetwork connections.

Translation Type:

The Translation Types are defined within ITU-T Recommendation Q.713 [42].

Numbering Plan:

- 1) The *proposed* "generic numbering plan" is described within the ITU-T Recommendation Q.713 [42]. This numbering plan identifies the SCCP nodes or SCCP subsystems unambiguously such that messages may be efficiently routed within one or more networks, and is particularly useful when used in the Calling Address for the sending of a response message back to the originating node. This is achieved by having an international and a national part in the generic numbering plan. For response messages the responding node analyses the international part of the generic numbering plan to determine the gateway node to which the response is to be routed. Having routed to the gateway node, the national part (which was populated within the originating network) is analysed to determine the originating node within the originating network.
- 2) A numbering plan which would define particular nodes based specifically on services is outside the scope of CAMEL.
- 3) The SCCP caters for a number of other numbering plans (e.g. ISDN, Mobile etc. numbering plans). The whole range catered for is shown in [?]. These may be used by CAMEL applications if deemed suitable.

Encoding Scheme:

This identifies the encoding scheme employed by the application and is generally BCD encoded with odd or even number of digits.

GT Address Information:

These are the actual address digits supplied by the application and may be BCD digits or encoded as indicated by the encoding scheme.

The network provider must ensure that any change of GT value during translation preserves any CAP specific information contained in the initial GT value.

This requirement applies to all interfaces, not just those used for internetworking.

If *route on SSN* is to be supported from the originating node, then an ITU-T non-zero internationally standardised SSN is required for international internetworking.

In the absence of a ITU-T standardised non-zero SSN for CAP services, the use of *route on GT* is mandatory from the origin node to the network containing the destination node.

When the SCCP of CCITT Signalling System No. 7 is used, the format and coding of address parameters carried by the SCCP for that purpose shall comply with ITU-T Recommendation Q.713 [42] with the following restrictions:

- 1) Intra-PLMN addressing

For communication between entities within the same PLMN, the use of SCCP addressing is network specific, and method (a) and (b) are both applicable.

- 2) Inter-PLMN addressing

method (b) with the mandatory SSN is applicable only with the following format:

- i) Called Party Address

- SSN indicator = a standardised SSN shall be used. The SSN used shall be that specified for CAP in 3GPP TS 23.003 [5];
- Point Code indicator = 0;
- Global title indicator = 0100 (Global title includes translation type, numbering plan, encoding scheme and nature of address indicator);
- Translation type = 0 (Not used);
- Routing indicator = 0 (Routeing on global title);

The format is also described in the table 14-2 below (for NP=1, NAI=4):

Table 14-2: Called Party Address format

8	7	6	5	4	3	2	1	
0	RI = 0	GTI = 4 (0100)				SSNI = 1	PCI = 0	Octet 1
SSN = a value for CAP as specified in 3GPP TS 23.003 [5]								Octet 2
Translation type = 0								Octet 3
Numbering plan = 1 (E.164)				Encoding scheme = 1 or 2				Octet 4
0	Nature of address indicator = 4 (International)							Octet 5
Country code digit 2 (if present)				Country code digit 1				Octet 6
National Destination Code (NDC) Digit 1				Country code digit 3 (if present)				Octet 7
NDC digit 3 (if present)				NDC digit 2 (if present)				Octet 8
NDC digit 5 (if present)				NDC digit 4 (if present)				Octet 9
Equipment identification digit 2				Equipment identification digit 1				Octet 10
...			
filler = 0 (if needed)				Equipment identification digit m				Octet n

Note Country code, National Destination Code, and SN(equipment id) are provided as example, so each digit may differ for each Inter-PLMN addressing case. (e.g., there is a case where only CC digit 1 shall be used). See ITU-T Recommendation Q.713 [42] for translation rules.

ii) Calling Party Address

- SSN indicator = a standardised SSN shall be used. The SSN used shall be that specified for CAP in 3GPP TS 23.003 [5];
- Point code indicator = 0;
- Global title indicator = 0100 (Global title includes translation type, numbering plan, encoding scheme and nature of address indicator);
- Translation type = 0 (Not used);
- Routing indicator = 0 (Routeing on Global Title).

The format is also described in the table 12-3 below (for NP=1, NAI=4):

Table 14-3: Calling Party Address format

8	7	6	5	4	3	2	1	
0	RI = 0	GTI = 4				SSNI = 1	PCI = 0	Octet 1
SSN = a value for CAP as specified in 3GPP TS 23.003 [5]								Octet 2
Translation type = 0								Octet 3
Numbering plan = 1 (E.164)				Encoding scheme = 1 or 2				Octet 4
0	Nature of address indicator = 4 (International)						Octet 5	
Country code digit 2 (if present)				Country code digit 1				Octet 6
National Destination Code (NDC) Digit 1				Country code digit 3 (if present)				Octet 7
NDC digit 3 (if present)				NDC digit 2 (if present)				Octet 8
NDC digit 5 (if present)				NDC digit 4 (if present)				Octet 9
Equipment identification digit 2				Equipment identification digit 1				Octet 10
...			
Filler = 0 (if needed)				Equipment identification digit m				Octet n

Note Country code, National Destination Code, and SN(equipment id) are provided as example, so each digit may differ for each Inter-PLMN addressing case. (e.g., there is a case where only CC digit 1 shall be used). See ITU-T Recommendation Q.713 [42] for translation rules.

When the SCCP of ANSI Signalling System No. 7 is used, the format and coding of address parameters carried by the SCCP for the purpose of signalling transfer shall comply with ANSI Recommendation T1.112-1996 [91] with the following restrictions:

1) Intra-PLMN addressing

For communication between entities within the same PLMN, the use of SCCP addressing is network specific.

2) Inter-PLMN addressing

a) Called Party Address

- SSN indicator = a standardised SSN shall be used. The SSN used shall be that specified for CAP in 3GPP TS 23.003 [5];
- Point Code indicator = 0;
- Global title indicator = 0010 (Global title includes translation type);
- the Translation Type (TT) field shall be coded according to the content of the address information as follows:
 - TT = 9 (decimal), if IMSI is included; or
 - TT = 14 (decimal), if MSISDN is included; or
 - TT = 10 (decimal), if a Network Element address is included. (If TT=10, then Number Portability is not applicable, if TT=14, then Number Portability is applicable)
- Routing indicator = 0 (Routeing on global title);

b) Calling Party Address

- SSN indicator = a standardised SSN shall be used. The SSN used shall be that specified for CAP in 3GPP TS 23.003 [5];
- Point code indicator = 0;
- Global title indicator = 0010 (Global title includes translation type);
- the Translation Type (TT) field shall be coded according to the content of the address information as follows:
 - TT = 9 (decimal), if IMSI is included; or
 - TT = 14 (decimal), if MSISDN is included; or

TT = 10 (decimal), if a Network Element address is included. (If TT=10, then Number Portability is not applicable, if TT=14, then Number Portability is applicable)

- Routing indicator = 0 (Routeing on Global Title).

14.2.2.1.3 Sequence control

The application will specify whether SCCP protocol class 0 or 1 is required. Class 0 provides a basic connection-less service where the sequence of message delivery is not guaranteed. Class 1 connection-less service provides a guaranteed sequence delivery of messages (with the same called address) for a given stream of messages. Class 1 shall be requested by any application that can send more than 1 TC message to its peer (consecutive TR-CONTINUE) before receiving a response from its peer (TR-CONTINUE or TR-END).

On receipt of a TC-RESULT-NL indication, the TC-USER shall request the transfer of a reject component using TC-U-REJECT request primitive, with the appropriate problem code (mistyped parameter).

The return option may be used if requested by the application (Network Operator to determine).

14.2.2.1.4 Return on error

Return on Error mechanism may be required by the CAMEL applications such that the application is aware of messages that have not been delivered to the destination by the SCCP. The return option allows the return of the message that was not delivered due to routeing or segmentation/re-assembly failure back to the issuing user. This return option may be required in all segments of a long message or only in the first segment by the CAMEL applications.

If the return option is invoked by the application and the message is not delivered, then the SCCP specifies the "return reason" as specified in ITU-T Recommendation Q.713 [42]. The N-NOTICE primitive is used to return the undelivered message to the originating user.

14.2.2.1.5 Segmentation / reassembly

The application expects that since the SCCP can send up to 260 octets of user data (including the address information and TC-message) in a UDT message (248 octets in a XUDT message performing segmentation and congestion control), segmentation is available for long user messages.

Also the SCCP is expected to perform the reassembly function on received segmented messages and deliver the reassembled message to the user.

However, it should be noted that even though the theoretical maximum size of SCCP-user data and addresses that can be segmented by the SCCP is 3 968 octets, the SCCP-user would limit the length to about 2 560 octets to allow for the largest known addresses. Note that the application must also allow for the octets used for the TC-message in the 2 560 octets.

The CAMEL application does not expect the SCCP to segment the long message into more than 16 segments.

14.2.2.1.6 Congestion control

To help control of possible congestion that might occur in the lower layers the application may assign a value to indicate the importance of the message. The use of this parameter requires the use of SCCP (1997) ITU-T Recommendations.

Also there exist other congestion control mechanisms as indicated below in SCCP Management.

These congestion control methods are network operator option in the case of intra-PLMN network signalling, and shall not be used in the case of inter-PLMN network signalling.

14.2.2.2 SCCP connection oriented services

The use by CAMEL applications for the Connection-oriented services is outside the scope of CAMEL.

14.2.2.3 SCCP management

The subsystems used within the CAMEL application expect the SCCP to provide management procedures to maintain network performance by re-routeing in the event of failure of a subsystem, and in the case of network congestion by use of the congestion handling procedure. These procedures have appropriate interactions with the SCCP user as described in ITU-T Recommendations Q.713 [42] and ITU-T Recommendation Q.714 [43].

To achieve the above the SCCP is expected to perform the following procedures:

- Signalling point status management (which include the signalling point prohibited, signalling point allowed, signalling point congested, and local MTP availability sub procedures).
- Subsystem status management (which include the subsystem prohibited, subsystem allowed, and subsystem status test sub procedures).
- Co-ordinated state change (a procedure which allows a duplicated subsystem to be withdrawn from service without affecting the performance of the network).

These SCCP management procedures are network operator option in the case of intra-PLMN network signalling, and shall not be used in the case of inter-PLMN network signalling.

Annex A (normative): Mapping between CAP and ISUP

A.1 InitialDP operation

Table A.1

ISUP message IAM (Note 1)	CAP Operation InitialDP
Called party number	CalledPartyNumber
Calling party number	CallingPartyNumber
Calling party's category	CallingPartysCategory
Location number	LocationNumber
Original called number	OriginalCalledPartyID
User teleservice information (1 st priority)	HighLayerCompatibility
High layer compatibility IE contained in access transport (2 nd priority) (Note 2)	
Generic number "additional calling party number"	AdditionalCallingPartyNumber
User service information prime (1 st priority)	BearerCapability
User service information (2 nd priority)	
Redirecting number	RedirectingPartyID
Redirection information	RedirectionInformation
Call diversion treatment indicators	ServiceInteractionIndicatorsTwo.Call diversion treatment indicators
Conference treatment indicators	ServiceInteractionIndicatorsTwo.Conference treatment indicators

NOTE 1: Optional parameters may be absent, i.e. they are only mapped only if these parameters are available at the DP.

NOTE 2: If two high layer compatibility information elements are contained in the access transport parameter, then the second information element, carrying the preferred HLC, is mapped to the CAP highLayerCompatibility parameter.

A.2 ContinueWithArgument operation

Table A.2 illustrates the mapping of parameters received in the ContinueWithArgument operation to parameters sent in the ISUP IAM to the succeeding exchange. Parameters which were received in the ISUP IAM and are not replaced by parameters of the ContinueWithArgument operation are treated according to the normal procedures.

On sending of the ISUP IAM the awaiting address complete timer is started. If the timer expires, then the call is released in both directions and an appropriate indication is returned to the calling subscriber.

Table A.2

CAP Operation ContinueWithArgument (Note 1)	ISUP message IAM
originalCalledPartyID	Original called number
callingPartysCategory	Calling party's category
redirectingPartyID	Redirecting number
redirectionInformation	Redirection information
genericNumbers	Generic number (Note 2)
serviceInteractionIndicatorTwo	See Table A.4
cug-Interlock	Closed user group interlock code
cug-OutgoingAccess	Optional forward call indicators (Note 3)

NOTE 1: Optional parameters may be absent, i.e. they are mapped only if received.

NOTE 2: The set of generic numbers received in the genericNumbers parameter is mapped to the appropriate number of Generic Number parameters in the ISUP IAM. This shall be performed irrespective of the value of the screening indicator in the ISUP calling party number.

NOTE 3: The cug-OutgoingAccess is mapped to the Closed User Group indicator which is carried in bits A & B of the Optional forward call indicators.

A.3 Connect operation

On receipt of a Connect operation from the gsmSCF the called party number used for routing is derived from the destinationRoutingAddress (see Table A.3). If the triggering of the CAMEL service was made for a mobile terminating or forwarded call, then an ISUP ACM shall be sent to the preceding exchange. The encoding of the backward call indicators in the ISUP ACM is specified in 3GPP TS 09.12 [1].

Table A.3 illustrates the mapping of parameters received in the Connect operation to parameters sent in the ISUP IAM to the succeeding exchange. Parameters which were received in the ISUP IAM and are not replaced by parameters of the Connect operation are treated according to the normal procedures.

On sending of the ISUP IAM the awaiting address complete timer is started. If the timer expires, then the call is released in both directions and an appropriate indication is returned to the calling subscriber.

Table A.3

CAP Operation Connect (Note 1)	ISUP message IAM
DestinationRoutingAddress	Called party number
OriginalCalledPartyID	Original called number
callingPartysCategory	Calling party's category
redirectingPartyID	Redirecting number
redirectionInformation	Redirection information
GenericNumbers	Generic number (Note 2)
ServiceInteractionIndicatorTwo	See Table A.4
Cug-Interlock	Closed user group interlock code
cug-OutgoingAccess	Optional forward call indicators (Note 3)

NOTE 1: Optional parameters may be absent, i.e. they are mapped only if received.

NOTE 2: The set of generic numbers received in the genericNumbers parameter is mapped to the appropriate number of Generic Number parameters in the ISUP IAM. This shall be performed irrespective of the value of the screening indicator in the ISUP calling party number.

NOTE 3: The cug-OutgoingAccess is mapped to the Closed User Group indicator which is carried in bits A & B of the Optional forward call indicators.

Table A.4 – Mapping of the CAP Connect and ContinueWithArgument operation serviceInteractionIndicatorsTwo to ISUP.

Table A.4

CAP ServiceInteractionIndicators	ISUP parameter in	
	ACM/CPG/CON/ANM/REL	IAM
Call to be diverted indicator – call diversion allowed – call diversion not allowed	---	Call diversion treatment indicators parameter Call to be diverted indicator – no indication – call diversion allowed – call diversion not allowed
Conference at DLE accept. ind. – accept conference request – reject conference request	---	Conference treatment indicators parameter Conference acceptance ind. – no indication – accept conference request – reject conference request
Calling party restriction indicator - no IN impact - presentation restricted	---	Calling party number address presentation restricted indicator - no impact - presentation restricted
Conference at OLE accept. ind. – accept conference request – reject conference request	ACM/CPG/CON/ANM: Conference treatment indicators parameter Conference acceptance ind. – no indication – accept conference request – reject conference request	---
Call completion treatment indicator - accept CCBS service request - reject CCBS service request	REL, busy cause Diagnostig field – CCBS possible – CCBS not possible	---
Connected number treatment indicator – no IN impact – presentation restricted – present called IN number – present called IN number restricted	Note 3	

NOTE 3:

If "no IN impact" was received in the CAP serviceInteractionIndicatorsTwo (connected number treatment indicator), then a connected number parameter and a generic number parameter "additional connected number" are passed on unchanged.

If "presentation restricted" was received in the CAP serviceInteractionIndicatorsTwo, then:

- a) If a connected number parameter has been received in the ISUP ANM or ISUP CON, then the address presentation restricted indicator is set to "presentation restricted".
- b) If a generic number parameter "additional connected number" has been received in the ISUP ANM or ISUP CON, then the address presentation restricted indicator is set to "presentation restricted".
- c) If a redirection number parameter has been received, then a redirection number restriction parameter is sent in the ISUP ANM with bits AB set to "presentation restricted".

If "present called IN number" was received in the CAP serviceInteractionIndicatorsTwo, then:

- a) If a connected number parameter has been received in the ISUP ANM or ISUP CON, then the connected number parameter is modified as follows:

nature of address indicator and numbering plan indicator are encoded as received in the called party number of the ISUP IAM,

address presentation restricted indicator: 00 (presentation allowed),

address signals: as received in the called party number and possible subsequent number parameters, until the ISUP ACM was sent.

- b) A generic number parameter "additional connected number" is deleted from the message, if applicable,
- c) A redirection number parameter is deleted from the relevant messages, if applicable.

If "present called IN number restricted" was received in the CAP serviceInteractionIndicatorsTwo, then:

- a) If a connected number parameter has been received in the ISUP ANM or ISUP CON, then the connected number parameter is modified as follows:
 - nature of address indicator and numbering plan indicator are encoded as received in the called party number of the ISUP IAM,
 - address presentation restricted indicator:01 (presentation restricted),
 - address signals:as received in the called party number and possible subsequent number parameters, until the ISUP ACM was sent.
- b) A generic number parameter "additional connected number" is deleted from the message, if applicable,
- c) A redirection number parameter is deleted from the relevant messages, if applicable.

A.4 AssistRequestInstructions operation

If an ISUP IAM is received at an assisting SSP containing an assisting gsmSSF or at an IP containing a gsmSRF, then an AssistRequestInstructions operation is sent to the gsmSCF. The correlationID parameter in the AssistRequestInstructions operation can contain:

- a) the CorrelationID digits extracted from the ISUP IAM Called Party Number,
- b) the whole Called Party Number received in the ISUP IAM (CorrelationID digits extracted at gsmSCF),
- c) the contents of the ISUP IAM CorrelationID parameter.

In the case where the gsmSCF and the assisting gsmSSF are both in the HPLMN and ISUP 97 is supported then any of these mechanisms may be used.

In the case where the gsmSCF and the assisting gsmSSF are both in the HPLMN and ISUP 97 is not supported then mechanisms a) and b) may be used.

In the case where the gsmSCF is in the HPLMN and the assisting gsmSSF is in the VPLMN then only mechanism b) may be used when an all-ISUP 97 signalling path cannot be guaranteed. Mechanism a) may be used if bilateral agreements on the format of the information transferred in the ISUP IAM Called Party Number are defined between the HPLMN and the VPLMN.

In the case where the gsmSCF is in the HPLMN and the assisting gsmSSF is in the VPLMN then mechanism c) may only be used if an all-ISUP 97 signalling path can be guaranteed between the HPLMN and the VPLMN.

A.5 ConnectToResource operation

On receipt of a ConnectToResource operation from the gsmSCF, the the IP is connected to the incoming call, to facilitate User Interactive dialogue with the user.

If the User Interactive dialogue is to be performed at a forwarding MSC or GMSC, then an ISUP ACM shall be sent to the preceding exchange. The encoding of the backward call indicators in the ISUP ACM is specified in 3GPP TS 09.12 [1], with the Optional Backward Call Indicators indicating "in-band information or an appropriate pattern is now available".

If the User Interactive dialogue is to be performed at a forwarding MSC or GMSC, then when the IP indicates through-connection and the ConnectToResource operation indicates that a bothway throughconnection is required, an ISUP ANM shall be sent to the preceding exchange if answer has not previously been sent. As a network operator/equipment vendor option an ISUP CPG message may be sent if ISUP ANM has already been sent.

A.6 EstablishTemporaryConnection operation

On receipt of an EstablishTemporaryConnection operation from the gsmSCF then if the triggering of the CAMEL service was made for a mobile terminating or forwarded call, an ISUP ACM shall be sent to the preceding exchange. The encoding of the backward call indicators in the ISUP ACM is specified in 3GPP TS 09.12 [1]. In addition, an ISUP IAM shall be sent to the succeeding exchange.

Table A.5 illustrates the mapping of parameters received in the EstablishTemporaryConnection operation to parameters sent in the ISUP IAM to the succeeding exchange. On sending of the ISUP IAM the awaiting address complete timer is started. If the timer expires, then the call is released in both directions and an appropriate indication is returned to the calling subscriber.

Table A.5

CAP Operation EstablishTemporaryConnection (Note 1)	ISUP message IAM
AssistingSSPIPRoutingAddress	Called party number
CorrelationID	Correlation id (note 1)
Scfld	GsmSCF id (note 1)

NOTE 1: These optional parameters may be absent, i.e. they are mapped only if received. If they are received and cannot be mapped, then an error is sent to the gsmSCF as detailed in clause 11.

NOTE 2: The AssistingSSPIPRoutingAddress parameter may also include a Hex B digit, in order to delineate the boundary between digits used for routing and digits forming part of the SCFiD and/or CorrelationID.

Except for the Called Party Number the remaining mandatory ISUP IAM parameters are set as follows:

a) Nature of connection indicators

Satellite indicator: set as in an Originating MSC,

Continuity check indicator: set as in Originating MSC,

Echo control device indicator: set as in Originating MSC

b) Forward Call Indicators

National/international call indicator: set as in Originating MSC,

End-to-end method indicator: 00 (no end-to-end method available),

Interworking indicator: 0 (no interworking encountered),

End-to-end information indicator: 0 (no end-to-end information available),

ISDN User Part indicator: 1 (ISDN User Part used all the way),

ISDN User Part preference indicator: 00 (ISDN User Part preferred all the way),

ISDN access indicator: 0 (originating access non-ISDN),

SCCP method indicator: 00 (no indication)

c) Calling Party's Category

00001010 (ordinary subscriber)

d) Transmission Medium Requirement

00000011 (3.1 kHz audio)

The ISUP IAM optional parameter Propagation Delay Counter is set as in an Originating MSC.

A.7 ReleaseCall operation

Upon receipt of the ReleaseCall operation, the GMSC/gsmSSF (VMSC/gsmSSF) sends REL messages in both directions. The cause indicators parameter contains the releaseCallArg parameter of the ReleaseCall operation.

A.8 InitiateCallAttempt operation

On receipt of an InitiateCallAttempt operation, the GMSC/gsmSSF or VMSC/gsmSSF prepares to set up a new call leg in an existing call configuration, or to set up a new call, to the destination defined by the DestinationRoutingAddress parameter. The GMSC/gsmSSF or VMSC/gsmSSF does not send the ISUP IAM until it has received a ContinueWithArgument operation, which contains additional information needed to populate the parameters of the ISUP IAM.

Table A.6 illustrates the mapping of parameters received in the InitiateCallAttempt operation to parameters sent in the ISUP IAM to the succeeding exchange.

Table A.6

CAP Operation InitiateCallAttempt (Note 1)	ISUP message IAM
DestinationRoutingAddress	Called party number
CallingPartyNumber	Calling party number

NOTE 1: Optional parameters may be absent, i.e. they are mapped only if received.

Annex B (normative): Mapping between CAP and MAP and mapping between CAP and SM-CP

The Short Message signalling between the SMS-GMSC and the MSC or SGSN, for the transport of SM RP-DUs, is specified in 3G TS 29.002 [11]. The Short Message signalling between the MSC or SGSN and the MS, for the transport of SM RP-DUs, is specified in 3G TS 24.011 [10].

B.1 Mapping between CAP and MAP

B.1.1 MO-SMS

B.1.1.1 ConnectSMS operation

The Short Message TPDU is carried by a Short Message RPDU. The RPDU between the MSC or SGSN and the SMS-IW MSC is conveyed in the MAP MO-ForwardSM OPERATION. The MSC or SGSN shall obtain information from CAP OPERATION ConnectSMS and place it in MAP OPERATION MO-ForwardSM.

Table B.1

CAP ConnectSMS	MAP MO-ForwardSM ARGUMENT
CallingPartysNumber	sm-RP-OA; CHOICE set to msisdn
DestinationSubscriberNumber	sm-RP-UI; TP-Destination-Address
SMSCAddress	sm-RP-DA; CHOICE set to serviceCentreAddressDA

NOTE The encoding of "SignalInfo" in MO-ForwardSM is specified in 3G TS 23.040 [6].

B.1.1.2 EventReportSMS operation

The MSC or SGSN may receive error information from the SMS-IW MSC, via the MAP OPERATION MO-ForwardSM ERROR. The MSC or SGSN shall obtain the error information from MAP OPERATION MO-ForwardSM and place it in the CAP OPERATION EventReportSMS.

Table B.2

CAP EventReportSMS	MAP MO-ForwardSM ERROR
MO_SMS Cause	SM-DeliveryFailure; SM-DeliveryFailureCause; SM-EnumeratedDeliveryFailureCause

B.1.2 MT-SMS

B.1.2.1 InitialDPSMS operation (MT-SMS)

The Short Message TPDU is carried by a Short Message RPDU. The RPDU between the SMS-GMSC and the MSC or SGSN is conveyed in the MAP MT-ForwardSM OPERATION ARGUMENT. The MSC or SGSN shall obtain information from the MT-ForwardSM MAP Message and place it in CAP InitialDPSMS.

Table B.3

CAP InitialDPSMS	MAP MT-ForwardSM ARGUMENT
CallingPartysNumber	TP-Originating-Address from SMS-Deliver-TPDU. SMS-Deliver-TPDU is contained in sm-RP-UI.
SMSCAddress	sm-RP-OA
TPShortMessageSpecificInfo	First OCTET of the TPDU. The TPDU is contained in sm-RP-UI.
TPProtocolIdentifier	TP-Protocol-Identifier of the TPDU. The TPDU is contained in sm-RP-UI.
TPDataCodingScheme	TP-Data-Coding-Scheme of the TPDU. The TPDU is contained in sm-RP-UI.
TPValidityPeriod	TP-Validity-Period of the TPDU. The TPDU is contained in sm-RP-UI.

NOTE The encoding of "SignalInfo" in MT-ForwardSM is specified in 3G TS 23.040 [6].

B.1.2.2 ReleaseSMS operation (MT-SMS)

When the MSC or SGSN receives the CAP ReleaseSMS operation, it shall return a RP-ERROR RPDU to the SMS-GMSC. The RP-ERROR RPDU between the MSC or SGSN and the SMS-GMSC is conveyed in the MAP MT-ForwardSM OPERATION ERROR.

Table B.4

CAP ReleaseSMS	MAP MT-ForwardSM ERROR
RPCause	SM-DeliveryFailure; sm-DeliveryFailureCause; sm-EnumeratedDeliveryFailureCause

B.2 Mapping between CAP and SM-CP

B.2.1 MO-SMS

B.2.1.1 InitialDPSMS operation

The Short Message TPDU is carried by a Short Message RPDU. The RPDU between the MS and the MSC or SGSN is conveyed in the RP-DATA Relay Message. The MSC or SGSN shall obtain information from the RP-DATA Message and place it in CAP InitialDP OPERARTION.

Table B.5

CAP InitialDPSMS ARGUMENT	SM-CP RP-DATA Message
DestinationSubscriberNumber	RP-User Data; TPDU; TP-Destination-Address
SMSCAddress	RP-User Data; RP-Destination Address
TPShortMessageSpecificInfo	RP-User Data; TPDU; 1 st octet
TPProtocolIdentifier	RP-User Data; TPDU; TP-Protocol-Identifier
TPDataCodingScheme	RP-User Data; TPDU; TP-Data-Coding-Scheme
TPValidityPeriod	RP-User Data; TPDU; TP-Validity-Period-Format

B.2.1.2 ReleaseSMS operation

When the MSC or SGSN receives the CAP ReleaseSMS operation, it shall return a RP-ERROR RPDU to the MS. The RP-ERROR RPDU between the MSC or SGSN and the MS is conveyed in the RP-ERROR Relay Message. The MSC or SGSN shall obtain the RPCause from the CAP ReleaseSMS OPERATION and place it in the RP-ERROR Message.

Table B.6

CAP ReleaseSMS ARGUMENT	SM-CP RP-ERROR Message
RPCause	RP-Cause

B.2.2 MT-SMS

B.2.2.1 ConnectSMS operation

When the MSC or SGSN receives the ConnectSMS operation, it shall use the received information to replace data in the Short Message SMS-DELIVER TPDU. The SMS-DELIVER TPDU is carried in a RP-MT-DATA RPDU. The RP-MT-DATA RPDU between the MSC or SGSN and the MS is conveyed by the SM-CP CP-Data Message.

Table B.7

CAP ConnectSMS	SM-CP RP-DATA Message
CallingPartysNumber	RP-User Data; TPDU; TP-Originating-Address

B.2.2.2 EentReportSMS operation

When the MSC or SGSN receives the RP-ERROR RPDU from the MS, it shall send the RP-Cause to the gsmSCF. The RP-ERROR RPDU between the MS and the MSC or SGSN is conveyed in the RP-ERROR Relay Message. The MSC or SGSN shall obtain the RPCause from the RP-ERROR Message and place it in CAP EventReportSMS.

Table B.8

CAP EventReportSMS	SM-CP RP-ERROR Message
EventSpecificInformation	RP-Cause

Annex C (informative): Change history

Change history					
TSG CN#	Version	CR	Tdoc CN	New Version	Subject/Comment
June 2002	4.4.0	244r1	NP-020206	5.0.0	First release of TS 29.078 Rel-5 (CAMEL Phase 4)

CHANGE REQUEST

⌘ **29.078** CR **244** ⌘ rev **1** ⌘ Current version: **4.4.0** ⌘

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title: ⌘ Composite changes for CAMEL Phase 4

Source: ⌘ CN2

Work item code: ⌘ CAMEL4

Date: ⌘ 28 February 2002

Category: ⌘ **B**

Release: ⌘ Rel-5

Use one of the following categories:

Use one of the following releases:

F (correction)

2 (GSM Phase 2)

A (corresponds to a correction in an earlier release)

R96 (Release 1996)

B (addition of feature),

R97 (Release 1997)

C (functional modification of feature)

R98 (Release 1998)

D (editorial modification)

R99 (Release 1999)

REL-4 (Release 4)

REL-5 (Release 5)

Reason for change: ⌘ **Introduction of CAMEL Phase 4 (Rel-5).**

The present document specifies the stage 3 description for the fourth phase (see 3GPP TS 22.078) of "Customized Applications for Mobile network Enhanced Logic" (CAMEL) feature which provides the mechanisms to support services of operators which are not covered by standardized GSM services even when roaming outside the HPLMN.

The fourth phase of CAMEL feature supports, in addition to the third phase of CAMEL:

- Interactions with Optimal Routeing;
- Call Party Handling;
- DTMF Mid call procedure for Mobile Originated and Mobile Terminating calls;
- Inclusion of flexible tone injection;
- Charging Notification to CSE;
- Provision of location information of called subscriber;
- Provide location information during ongoing call;
- CAMEL control over MT SMS;
- Notification of GPRS mobility management to CSE;
- Inclusion of ODB data in Any Time Modification;
- Enhancement of Any Time Interrogation and Provide Subscriber Information for PS Domain.
- Reporting of IMEI (with SW version) to the CSE
- Support of CAMEL4 subsets

In order to realise the technical specification stage 3, CN2 has worked to create the draft specification. All the approved proposals to CAMEL Phase 4, as well as most of the approved CRs to CAMEL Phase 3, are included in the present CAMEL Phase 4 draft specification.

Approval of the attached draft version of TS 29.078 for Rel-5 shall result in the formal release of TS 29.078 V5.0.0.

Summary of change: ⌘ This CR contains all the changes that are required for the update of CAMEL Phase 3 to CAMEL Phase 4.

Consequences if not approved: ⌘ CAMEL Phase 4 will not work.

Clauses affected: ⌘ All

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

Other comments: ⌘ - The content of the CR is contained in a separate document file within N2-020627.zip.
- This document was submitted for information at CN#15 Plenary. CN2 requests the approval at the CN#16 Plenary.
- All the individual change requests that have been implemented in TS 29.078 for WI CAMEL4 are listed in a separate file, "29078-500 Change History".

Attached to the present document is the draft CAMEL Phase 4 3GPP TS 29.078.