Oracle® Communications Convergent Charging Controller

CAP3 GPRS Control Agent Technical Guide

Release 15.0.0

October 2023

Copyright

Contents

About This Document

Scope

The scope of this document includes all the information required to install, configure and administer the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GPRS Control Agent) application.

Audience

This guide was written primarily for system administrators and persons installing, configuring and administering the CAP3GPRS Control Agent application. However, sections of the document may be useful to anyone requiring an introduction to the application.

Prerequisites

A solid understanding of UNIX and a familiarity with IN concepts are essential prerequisites for safely using the information contained in this technical guide. Attempting to install, remove, configure or otherwise alter the described system without the appropriate background skills could cause damage to the system; including temporary or permanent incorrect operation, loss of service, and may render your system beyond recovery.

A familiarity with the CAP version 3 GPRS protocol is also required. Refer to the following document:

CAMEL Application Part (CAP) specification (3GPP TS 29.078), version 4.8.0 Release 4

This manual describes system tasks that should only be carried out by suitably trained operators.

Related Documents

The following documents are related to this document:

- *Advanced Control Services Technical Guide*
- *CAP3GPRS Compliance Protocol Conformance Statement Guide*
- *Charging Control Services Technical Guide*
- *Charging Control Services User's Guide*
- *Service Management System Technical Guide*
- *Service Management System User's Guide*
- *Service Logic Execution Environment Technical Guide*

Document Conventions

Typographical Conventions

The following terms and typographical conventions are used in the Oracle Communications Convergent Charging Controller documentation.

Formatting Convention	Type of Information
Special Bold	Items you must select, such as names of tabs. Names of database tables and fields.
Italics	Name of a document, chapter, topic or other publication. Emphasis within text.
Button	The name of a button to click or a key to press. **Example:** To close the window, either click **Close**, or press **Esc**.
Key+Key	Key combinations for which the user must press and hold down one key and then press another. Example: **Ctrl+P** or **Alt+F4**.
`Monospace`	Examples of code or standard output.
`Monospace Bold`	Text that you must enter.
variable	Used to indicate variables or text that should be replaced with an actual value.
menu option > menu option >	Used to indicate the cascading menu option to be selected. Example: **Operator Functions > Report Functions**
hypertext link	Used to indicate a hypertext link.

Specialized terms and acronyms are defined in the glossary at the end of this guide.

Chapter 1

System Overview

Overview

Introduction

This chapter provides a high-level overview of the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GPRS Control Agent). It describes the main functionality and components of the CAP3GPRS Control Agent.

This guide is not intended to advise on any specific Convergent Charging Controller network or service implications of the product.

In this chapter

This chapter contains the following topics.

What is the CAP3GPRS Control Agent?

Introduction

The CAP3GPRS Control Agent is a SLEE application that translates between CAP3 GPRS messages and internal INAP operations.

The CAP3GPRS protocol is used to charge for data usage on GSM mobile devices.

Billing Using the UATB Feature Node

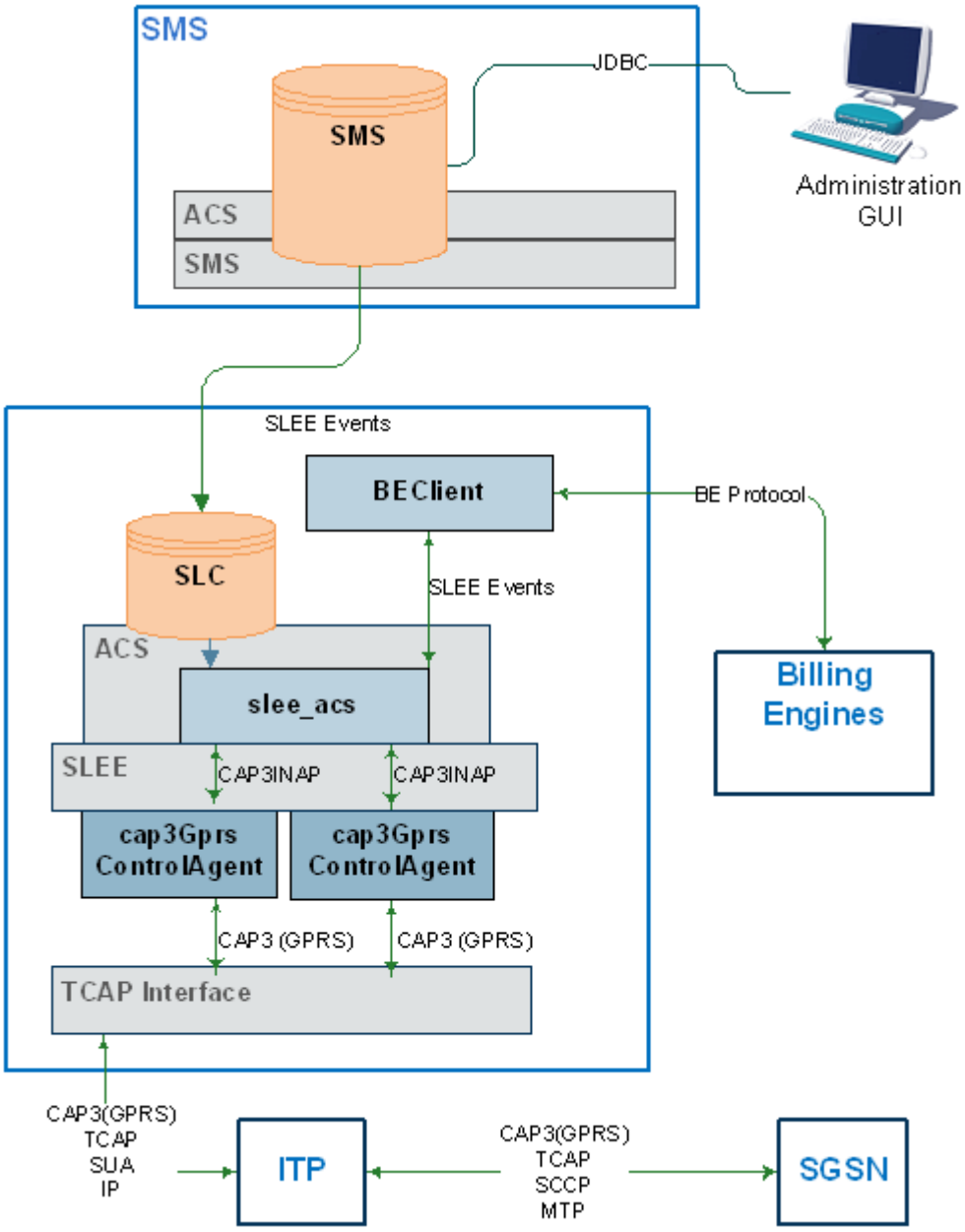
You should include the Universal Attempt Termination with Billing (UATB) feature node in control plans that are triggered by the CAP3GPRS Control Agent. You use the UATB feature node to bill for sessions.

For more information about the UATB feature node, see *Feature Nodes Reference Guide*.

Note: For volume billing, time will be converted to volume by using the multiplication factor set in *conversionFactor* (on page 14). A single GPRS request may bill for time or volume, but not both.

System Overview Diagram

This diagram shows how the CAP3GPRS Control Agent is implemented in an Convergent Charging Controller system.



For more information about Convergent Charging Controller system architecture, see the discussion on Convergent Charging Controller system architecture in the *System Administrator's Guide*.

Chapter 2

Configuration

Overview

Introduction

This chapter describes the configuration options for the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GPRS Control Agent).

In this chapter

This chapter contains the following topics.

Configuration Overview

Introduction

This section provides a high-level overview of how you configure the CAP3GPRS Control Agent.

The configuration files contain some configuration settings that are not explained in this chapter. These configuration settings are required by the application and should not be changed.

Configuration Components

This table lists the components of the CAP3GPRS Control Agent that you can configure, and the configuration file where each component is configured.

Component	Configuration File	Further Information
CAP3GPRS switch type	Configure the switch type for CAP3GPRS in the `acsCharging` section of **eserv.config** on SLC nodes.	*Switch Type Configuration for CAP3GPRS* (on page 4)
cap3gprsControlAgent	Configure the cap3gprsControlAgent binary in the `CAP3GPRS` section of **eserv.config** on SLC nodes.	*CAP3GPRS eserv.config Configuration* (on page 6)
cap3gprsControlAgent application ACSGPRS service	Configure the cap3gprsControlAgent application and the ACSGPRS service in **SLEE.cfg** on SLC nodes.	*SLEE.cfg Configuration* (on page 18) and *SLEE Technical Guide*

eserv.config Configuration

Introduction

The **eserv.config** file is a shared configuration file, from which many Oracle Communications Convergent Charging Controller applications read their configuration. Each Convergent Charging Controller machine (SMS, SLC, and VWS) has its own version of this configuration file, containing configuration relevant to that machine. The **eserv.config** file contains different sections; each application reads the sections of the file that contains data relevant to it.

The **eserv.config** file is located in the **/IN/service_packages/** directory.

The **eserv.config** file format uses hierarchical groupings, and most applications make use of this to divide the options into logical groupings.

Example Configuration File

Most applications come with an example of the section of the **eserv.config** configuration specific to that application, in a file called **eserv.config.example** in the root of the application directory. The example file for CAP3GPRS Control Agent is **/IN/service_packages/CAP3GPRS/etc/eserv.config.example**.

This example file contains commented examples of all of the parameters you can use to configure the application. You can use the example configuration for reference if you need to update the configuration for the control agent in the **eserv.config** file; for example, to add an optional parameter configuration.

Switch Type Configuration for CAP3GPRS

Introduction

The acsCharging section of **eserv.config** defines the switch types used to control the switch communication flows for the UATB feature node. You use the UATB feature node in control plans that are triggered by the CAP3GPRS Control Agent, to bill for sessions.

You must include a definition for the cap3 switch type in the acsCharging section of **eserv.config**.

Example: This example shows the default cap3 switch type configuration.

```
acsCharging = {
    swithConfiguration = [
        {
            switchType = "cap3"
            addContinue = false
            addDisconnectOrRelease = false
        }
    ]
}
```

For more information about acsCharging configuration, see *CCS Technical Guide*.

acsCharging Parameters

`switchConfiguration`

Syntax:	`switchConfiguration = [switch_parameters]`
Description:	Defines the switch type that will be used by the control agent during sessions.
Type:	Array
Optionality:	Required

Default: None

Notes: Additional switch types may be defined for other services. For more information, see *CCS Technical Guide*.

Example:

```
switchConfiguration = [
    {
        switchType = "cap3"
        addContinue = false
        addDisconnectOrRelease = false

    }
]
```

switchType

Syntax: `switchType = "str"`

Description: Sets the switch type for a UATB feature node.

Type: String

Optionality: Required

Allowed: Use `"cap3"` switch type for CAP3GPRS.

Default: None

Example: `switchType = "cap3"`

addContinue

Syntax: `addContinue = true|false`

Description: Defines whether the UATB feature node should enable send responses, add responses, and continue responses to the TCAP to enable charging for a successful subsequent reservation on the Voucher and Wallet Server.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed: true, false

Default: false

Example: `addContinue = false`

addDisconnectOrRelease

Syntax: `addDisconnectOrRelease = true|false`

Description: Sets whether the UATB feature node can release or disconnect sessions during billing scenarios; for example, if the calling party has exhausted his or her funds or the maximum call limit has been reached.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:

- true – Enable release or disconnect calls
- false – Disable release or disconnect calls

Default: false

Example: `addDisconnectOrRelease = true`

CAP3GPRS eserv.config Configuration

Introduction

The CAP3GPRS section in **eserv.config** configures the cap3gprsControlAgent binary process. See *cap3gprsControlAgent Process* (on page 27) for details.

Note: The CAP3GPRS Control Agent is configured at installation time by the post installation configuration script that runs automatically. You only need to modify this configuration if you want to change the default configuration.

Example CAP3GPRS Section

Here is an example CAP3GPRS section in the **eserv.config** configuration file.

```
CAP3GPRS = {

    sleepTimeMicroseconds = 10000
    sendContinueWithApplyCharging = true
    sendContinueWithRRGPRSE = true
    defaultCalledPartyNumber = "4839100008"
    defaultBillingType = 1
    defaultSessionTreatment = 1
    armAllEdpsAtOnce = true
    displayMessageSequences = false
    minimumInstancesForObjectCounting = 1000
    tssf = 10
    maxInactiveTimeForVolume = 600
    timeBillingInactiveTimeTolerance = 10
    activityTestResultTimeout = 10
    tcapInterfaceName = "sua_if"
    alarmOnLatency = true
    latencyInterval = 10
    releaseCauseInsufficientFunds = 26
    releaseCauseNetworkError = 38
    sendAbortForDetachEventType = false
    sendAbortForDisconnectEventType = false
    armConnectEstablishAckOnContextChangeOfPosition = true

    Services = [
        {
        serviceName = "AcsGprs"
        gprsServiceKey = 8111
        sleeServiceKey = 1
        inapServiceKey = 110
        calledPartyNumber = "4839100009"
        billingType = 1
        conversionFactor = 104857.6
        sessionTreatment = 1
        }
    ]

    NumberRules = [

        { fromNoa=4, remove=0, prepend="00" }
        { fromNoa=3, remove=0, prepend="0064" }
    ]

    Tracing = {

        OrigAddress = [
            "0064212",
```

```
            "0064213",
            "0064214"
        ]

        DestAddress = [
            "0064213",
            "0064214"
        ]

    traceDebugLevel = "cap3gprsMessageSequences"
    }

}
```

CAP3GPRS Parameters

sleepTimeMicroseconds

Syntax: `sleepTimeMicroseconds = microsecs`

Description: The maximum number of microseconds to sleep when there are no SLEE events to process.

Type: Integer

Optionality: Optional (default used if not set)

Default: 100

Example: `sleepTimeMicroseconds = 10000`

sendContinueWithApplyCharging

Syntax: `sendContinueWithApplyCharging = true|false`

Description: Whether or not to send ContinueGPRS operations with ApplyCharging INAP operations.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:

- true – Send ContinueGPRS
- false – Do not send ContinueGPRS

Default: true

Example: `sendContinueWithApplyCharging = true`

sendContinueWithRRGPRSE

Syntax: `sendContinueWithRRGPRSE = true|false`

Description: Whether or not to send ContinueGPRS messages with RequestReportGPRSEvent INAP operations.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:

- true – Send ContinueGPRS
- false – Do not send ContinueGPRS

Default: true

Example: `sendContinueWithRRGPRSE = true`

defaultCalledPartyNumber

Syntax: `defaultCalledPartyNumber = "num"`

Description: The default called party BCD number to put in the InitialDP that is used by CCS for CLI-DN charging.

Type: String

Optionality: Optional (default used if not set)

Allowed: A valid destination number.

Default: 0000

Example: `defaultCalledPartyNumber = "4839100008"`

defaultBillingType

Syntax: `defaultBillingType = int`

Description: Sets the default billing type to either time or volume.

Type: Integer

Optionality: Optional (default used if not set)

Allowed: 0 – Sets the default billing type to time.

1 – Sets the default billing type to volume.

Default: 1

Example: `defaultBillingType = 0`

defaultSessionTreatment

Syntax: `defaultSessionTreatment = int`

Description: Defines the default method used by CAP3GPRS Control Agent to handle InitalDPGPRS operations during a session.

Type: Integer

Optionality: Optional (default used if not set)

Allowed:

- 1 – Send ApplyCharging INAP operations for the duration of the session and do not arm PDP context establishment.
- 2 – Send ContiueGPRS operations, and monitor individual PDP contexts during the session, but do not monitor the session as a whole.

Default: 1

Example: `defaultSessionTreatment = 1`

armAllEdpsAtOnce

Syntax: `armAllEdpsAtOnce = true|false`

Description: Whether to arm all EDPs in a single RequestReportGPRSEvent operation, or wait for ContextEstablishmentAcknowledgement before arming disconnect.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:

- true – Arm all EDPs in a single RequestReportGPRSEvent operation.
- false – Wait for ContextEstablishmentAcknowledgement

Default: true

Example: `armAllEdpsAtOnce = true`

displayMessageSequences

Syntax: `displayMessageSequences = true|false`

Description: Whether or not to display message sequences to standard output. Message sequences are displayed as one line per message.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:
- true – Display message sequences to standard output
- false – Do not display message sequences

Default: false

Notes: Here is an example message output:

```
SLEECALLID 1234567 GPRS SCF->slee_acs:TCAP_BEGIN(InitialDP)
```

You can use grep and sed UNIX commands to collect all the output from a particular session and remove the first two fields from each line. The output would then look like this:

```
GPRS SCF->slee_acs:TCAP_BEGIN(InitialDP)
slee_acs->GPRS
SCF:TCAP_CONTINUE(ApplyCharging(releaseIfDurationExceeded)\n,RequestReport
BCSNEvent\n,Continue)
GPRS SCF-
>SGSN:TCAP_CONTINUE(RequestReportBCSNEvent(ContextEstablishmentAcknowledge
ment,disconnect)\n,ContinueGPRS)
SGSN->GPRS SCF:TCAP_END()
....
```

You can then convert this type of output to a graphical message sequence, by using a web-based sequence diagrams tool. For example:

http://www.plantuml.com/plantuml/

Example: `displayMessageSequences = true`

minimumInstancesForObjectCounting

Syntax: `minimumInstancesForObjectCounting = int`

Description: Sets the minimum number of instances allowed of a class for object counting debugging. For example, if you set the `cap3gprsObjectReport` debug flag, then the CAP3GPRS Control Agent prints a debug line every time the number of a class of objects reaches a multiple of `minimumInstancesForObjectCounting`.

You can use the reported output to help identify the source of any memory leaks. See *Finding the Cause of a Memory Leak* (on page 48) for more information.

Type: Integer

Optionality: Optional (default used if not set)

Default: 1000

Notes: If you set the `cap3gprsObjectCounts` debug flag, then the CAP3GPRS Control Agent produces one debug line every time the number of objects for a class changes.

For more information about debug and debug flags, see *System Administrator's Guide*.

Example: `minimumInstancesForObjectCounting = 900`

tssf

Syntax: `tssf = secs`

Description: Sets the tssf timer in seconds. The CAP3GPRS Control Agent runs this timer whenever it sends an operation to slee_acs that needs a response. If the timer expires before slee_acs responds, the control agent logs an error and closes the dialog with the GGSN.

Type: Integer

Optionality: Optional (default used if not set)

Default: 10

Example: `tssf = 10`

maxInactiveTimeForVolume

Syntax: `maxInactiveTimeForVolume = secs`

Description: Sets the number of seconds to wait before sending ActivityTestGPRS operations for volume billing.

Type: Integer

Optionality: Optional (default used if not set)

Default: 3600

Notes: This parameter does not apply when billing by time.

Example: `maxInactiveTimeForVolume = 600`

timeBillingInactiveTimeTolerance

Syntax: `timeBillingInactiveTimeTolerance = secs`

Description: Sets the number of seconds to add to the time taken for duration billing in ApplyCharging INAP operations. This value is used to calculate the time to wait before sending ActivityTestGPRS operations.

Type: Integer

Optionality: Optional (default used if not set)

Default: 10

Notes: This parameter does not apply when billing by volume.

Example: `timeBillingInactiveTimeTolerance = 10`

activityTestResultTimeout

Syntax: `activityTestResultTimeout = secs`

Description: How long to wait (in seconds) for an ActivityTestGPRS result.

Type: Integer

Optionality: Optional (default used if not set)

Default: 10

Example: `activityTestResultTimeout = 10`

tcapInterfaceName

Syntax: `tcapInterfaceName = "if_name"`

Description: The TCAP interface name to use for sending ActivityTestGPRS operations. The named interface must be configured in **SLEE.cfg**. For information about configuring interfaces in **SLEE.cfg**, see *SLEE Technical Guide*.

Type: String

Optionality: Required

Default: None

Notes: If this configuration entry is missing or empty, then the ActivityTestGPRS operation will not be sent.

Example: `tcapInterfaceName = "sua_if"`

Where sua_if is the name for the SIGTRAN TCAP interface.

alarmOnLatency

Syntax: `alarmOnLatency = true|false`

Description: Sets whether or not notice alarms should be generated with latency information for IDPs and ACRs (Diameter Accounting-Request Commands), for example when waiting on `slee_acs`.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:
- true – Include latency information
- false – Do not include latency information

Default: false

Example: `alarmOnLatency = true`

latencyInterval

Syntax: `latencyInterval = secs`

Description: How long (in seconds) to wait between each latency report.

Type: Integer

Optionality: Optional (default used if not set)

Default: 60

Notes: To prevent any latency reporting, set `latencyInterval` to 0 (zero).

Example: `latencyInterval = 10`

releaseCauseInsufficientFunds

Syntax: `releaseCauseInsufficientFunds = int`

Description: The cause value to send in the releaseGPRS message when a call is released due to insufficient funds.

Type: Integer

Optionality: Optional (default used if not set)

Default: 26 – 'Insufficient resources' SM cause in TS 24.008

Notes: Because the SGSN does not necessarily follow TS 24.009, another cause value may be more appropriate than 26, for a given SGSN.

Example: `releaseCauseInsufficientFunds = 26`

releaseCauseNetworkError

Syntax: `releaseCauseNetworkError = int`

Description: The cause value to send in releaseGPRS messages when a call is released due to a network error, such as a timeout, or an incorrect message sequence.

Type: Integer

Optionality: Optional (default used if not set)

Default: 38 – 'Network failure' SM cause in TS 24.008

Notes: Because the SGSN does not necessarily follow TS 24.008, another cause value may be more appropriate than 38, for a given SGSN.

Example: `releaseCauseNetworkError = 38`

`sendAbortForDetachEventType`

Syntax: `sendAbortForDetachEventType = true|false`

Description: Whether to send TCAP_ABORT or ContinueGPRS in response to a 'detach' GPRS event type in an InitialDPGPRS.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:
- true – Send TCAP_ABORT in the response.
- false – Send ContinueGPRS in the response.

Default: false

Example: `sendAbortForDetachEventType = true`

`sendAbortForDisconnectEventType`

Syntax: `sendAbortForDisconnectEventType = true|false`

Description: Whether to send TCAP_ABORT or ContinueGPRS in response to a 'disconnect' GPRS event type in an InitialDPGPRS.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:
- true – Send TCAP_ABORT in the response.
- false – Send ContinueGPRS in the response.

Default: false

Example: `sendAbortForDisconnectEventType = true`

`armConnectEstablishAckOnContextChangeOfPosition`

Syntax: `armConnectEstablishAckOnContextChangeOfPosition = true | false`

Description: Specifies whether the CAP3GPRS control agent returns applyChargingGPRS when the GPRS event type is pdp-ContextChangeOfPosition.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:
- true – Returns applyChargingGPRS
- false – Does not return applyChargingGPRS

Default: true

Notes:

Example: `armConnectEstablishAckOnContextChangeOfPosition = true`

Services Parameters

You configure a CAP3GPRS service by configuring the parameters in the CAP3GPRS, Services section of **eserv.config**. You can configure one or more services.

The following example `Services` section configures the AcsGprs service:

```
Services = [
    {
    serviceName = "AcsGprs"
    gprsServiceKey = 8111
```

```
        sleeServiceKey = 1
        inapServiceKey = 110
        calledPartyNumber = "4839100009"
        billingType = 1
        conversionFactor = 104857.6
        sessionTreatment = 1
        }
    ]
```

serviceName

Syntax `serviceName = "name"`

Description: The unique name of the service.

Type: String

Optionality: Required

Allowed: Any string

Default: Default

Example:: `serviceName = "AcsGprs"`

gprsServiceKey

Syntax: `gprsServiceKey = int`

Description: The incoming GPRS service key number.

Type: Integer

Optionality: Optional (default used if not set)

Default: 1

Example: `gprsServiceKey = 8111`

sleeServiceKey

Syntax `sleeServiceKey = int`

Description: The SLEE service key for the `serviceName` (on page 13) service. Used when sending an InitialDP to the SLEE. For more information about service keys, see *SLEE Technical Guide*.

Type: Integer

Optionality: Optional (default used if not set)

Default: 1

Example: `sleeServiceKey = 12`

inapServiceKey

Syntax: `inapServiceKey = int`

Description: Sets the INAP service key to use when sending InitialDP to the SLEE.

Type: Integer

Optionality: Optional (default used if not set)

Default: 1

Example: `inapServiceKey = 110`

calledPartyNumber

Syntax: `calledPartyNumber = "called_number"`

Description: The called party BCD number to put in the InitialDP. It is used by CCS for CLI-DN charging.

Type:	String
Optionality:	Optional (default used if not set)
Allowed:	A valid destination number.
Default:	Defaults to the value specified in the `defaultCalledPartyNumber` parameter.
Example:	`calledPartyNumber = "4839100009"`

billingType

Syntax:	`billingType = int`
Description:	Sets whether to bill based on time or volume.
Type:	Integer
Optionality:	Optional (default used if not set)
Allowed:	0 – Sets the billing type to time 1 – Sets the billing type to volume
Default:	Defaults to the value of `defaultBillingType`.
Example:	`billingType = 1`

conversionFactor

Syntax:	`conversionFactor = num`
Description:	The conversion factor to use when communicating with ACS to change deciseconds to octets. This factor is applied only when `billingType` is set to 1 (for volume billing).
Type:	Float
Optionality:	Optional (default used if not set)
Default:	104857.6 (converts one megabyte to one second)
Notes:	The billing engine charges based on the number of deciseconds used. When charging by volume, the number of "fake" deciseconds is calculated by applying the conversion factor to the number of bytes used. The actual conversion factor used is determined by the system administrator responsible for designing the service.
Example:	`conversionFactor = 104857.6`

sessionTreatment

Syntax:	`sessionTreatment = int`
Description:	Sets how the CAP3GPRS Control Agent handles an InitialDPGPRS operation for a session.
Type:	Integer
Optionality:	Optional (default used if not set)
Allowed:	• 1 – Send ApplyCharging INAP operations for the duration of the session and do not arm PDP context establishment. • 2 – Send ContiueGPRS operations to monitor individual PDP contexts during the session, but do not monitor the session as a whole.
Default:	Defaults to the value of `defaultSessionTreatment`.
Example:	`sessionTreatment = 1`

NumberRules Parameters

You can configure the number normalization rules for the CAP3GPRS Control Agent by configuring the parameters in the optional `NumberRules` section of **eserv.config**.

Example:

```
NumberRules = [
    { fromNoa=4, min=1, max=9, remove=0, prepend="" }
    { fromNoa=3, remove=0, prepend="0064", prefix="", targetNoa=4 }
    { fromNoa=1, remove=0, prepend="0064", prefix="", targetNoa=4 }
]
```

fromNoa

Syntax: `fromNoa = int`

Description: Used when attempting to match the nature of address (NoA) number contained in a message. If there is a match, the `fromNoa` part of the number rule is evaluated.

Type: Integer

Optionality: Required

Allowed:
- 2 – For unknown NoAs
- 3 – For national NoAs
- 4 – For international NoAs

Notes: If you omit `fromNoa` from the `NumberRules` parameter section, then no matching rule will be found.

Example: `fromNoa = 3`

max

Syntax: `max = num`

Description: Specifies the maximum number of digits a number may contain. To meet the max part of the number rule, the number of digits in the number must be equal to or less than the value of `max`.

Type: Integer

Optionality: Optional (default used if not set)

Default: 999

Example: `max = 9`

min

Syntax: `min = num`

Description: Specifies the minimum number of digits a number may contain. To meet the min part of the number rule, the number of digits in the number must be equal to or greater than the value of `min`.

Type: Integer

Optionality: Optional (default used if not set)

Default: 0

Notes: The value of the `min` parameter must be greater than or equal to the value of the `remove` (on page 16) parameter.

Example: `min = 5`

prefix

Syntax: `prefix = "pref"`

Description: Contains a digit or digits. Used to attempt to match the first digit or digits of a prefix number with the specified value. If the digit or digits match, the prefix part of the number rule is met.

Type: String

Optionality: Optional

Allowed: One or more decimal digits

Notes: This parameter is an element of the `NumberRules` parameter array.

Example: `prefix = "25"`

prepend

Syntax: `prepend = "digits"`

Description: Defines digits added to the beginning of a number.

Type: String

Optionality: Optional

Allowed: Any combination of decimal digits, or a null string (`""`)

Notes:

- If the `remove` and `prepend` parameters are both used in the same number rule, "*prepend*" is added to the beginning of the number after the number has been modified by the `remove` parameter.
- The `prepend` parameter is an element of the `NumberRules` parameter array.

Example: `prepend = "0"`

remove

Syntax: `remove = num`

Description: The number of digits stripped from the beginning of a number.

Type: Integer

Optionality: Required

Notes: The value of the `remove` parameter must be less than or equal to the value of the *min* (on page 15) parameter.

Example: `remove = 2`

targetNoa

Syntax `targetNoa = int`

Description: Target nature of address.

Type: Integer

Optionality: Optional

Notes:

- A value is typically specified in denormalization rules.
- The `targetNoa` parameter is an element of the `NumberRules` parameter array.

Example: `targetNoa = 4`

Tracing Parameters

The following parameters are used for tracing activities. They are all found within the `Tracing = { }` statement.

enabled

Syntax: `enabled = true|false`

Description: Switches tracing on or off.

Type: Boolean

Optionality: Optional

Allowed: true, false

Default: false

Notes: If false, then the parameters in the Tracing section are ignored.

Example:
```
enabled = false
```

OrigAddress

Syntax: `OrigAddress = ["addr","addr"...]`

Description: Lists the originating addresses to be traced.

Type: String

Optionality: Optional (default used if not set)

Allowed:
- Any valid addresses
- ""

Default: ""

Example:
```
OrigAddress = [
    "0064212",
    "0064213",
    "0064214"
]
```

DestAddress

Syntax: `DestAddress = ["addr","addr"...]`

Description: Lists the destination addresses to be traced.

Type: String

Optionality: Optional (default used if not set)

Allowed:
- Any valid address
- ""

Default: ""

Example:
```
DestAddress = [
    "0064212",
    "0064213",
]
```

traceDebugLevel

Syntax: `traceDebugLevel = "str,str"`

Description: Sets the debug tracing level.

Type: String

Optionality: Required

Allowed: For example, set `traceDebugLevel` to:
- "all" to switch on all debug for traced numbers
- "cap3gprsMessageSequences" to only record message sequences for traced numbers.

Default:	N/A
Notes:	This is a comma separated string. See *ACS Technical Guide* for more information.
Example:	`traceDebugLevel = "cap3gprsMessageSequences"`

SLEE.cfg Configuration

Introduction

The **SLEE.cfg** file is configured to enable the CAP3GPRS Control Agent to work. All necessary SLEE configuration is done at installation time by the configuration script; this section is for information only.

The SLEE configuration file is located in **/IN/service_packages/SLEE/etc/SLEE.cfg**.

See *SLEE Technical Guide* for information about configuring the SLEE.

CAP3GPRS Control Agent SLEE Configuration

During installation, the following lines are automatically added to the configuration in **SLEE.cfg**:

```
APPLICATION=cap3GprsControlAgent cap3GprsControlAgent.sh
/IN/service_packages/CAP3GPRS/bin 1 1 1000
SERVICE=ACSGPRS 1 cap3GprsControlAgent ACSGPRS
SERVICEKEY=INTEGER 8111 ACSGPRS
```

Note: **SLEE.cfg** must contain at least one SERVICE line and one SERVICEKEY line referring to the cap3GprsControlAgent APPLICATION line.

SERVICEKEY values

Here is an example SERVICEKEY entry in **SLEE.cfg**.

```
SERVICEKEY=INTEGER 0x1920000004b ACSGPRS
```

In this example the SERVICEKEY has the value of `0x1920000004b`. This number can be broken down into three parts.

1. The service key always starts with 0x1.
2. The next two digits (92) are the SCCP sub-system number from the SCCP called party address in hexadecimal digits.
3. The last digits are the `serviceKey` parameter in InitalDPGPRS operations (also in hex).

The example configuration causes all InitialDPGPRS operations sent to sub-system number 146 and with `serviceKey` 75, to use the ACSGPRS service.

Example Configuration

```
APPLICATION=cap3GprsControlAgent cap3GprsControlAgent.sh
/IN/service_packages/CAP3GPRS/bin 1 1 1000
SERVICE=ACSGPRS 1 cap3GprsControlAgent ACSGPRS
SERVICEKEY=INTEGER 0x1000000004b ACSGPRS
SERVICEKEY=INTEGER 0x1920000004b ACSGPRS
SERVICEKEY=INTEGER 0x1920000004c ACSGPRS
SERVICEKEY=INTEGER 8111 ACSGPRS
SERVICEKEY=INTEGER 8112 ACSGPRS
SERVICEKEY=INTEGER 8113 ACSGPRS
SERVICEKEY=INTEGER 8114 ACSGPRS
SERVICEKEY=INTEGER 8115 ACSGPRS
```

Incoming and Outgoing Session Data

Introduction

The CAP3GPRS Control Agent determines incoming and outgoing session data values from ACS profile tag values.

Note: Profile tags are stored in profile blocks and configured in the ACS user interface (UI). See the discussion on ACS configuration in *ACS User's Guide* for more information.

IDP Extensions Parameter

The CAP3GPRS Control Agent passes a profile block to ACS in the `InitalDP.extensions` parameter as extension type 701. The slee_acs process makes the profile tags from this profile block available to a feature node through the **Session data, Incoming Session Data** option in the feature node configuration window.

IDP Profile Tags

This table lists the information written into IDP extension type 701 by the CAP3GPRS Control Agent.

Profile Tag Number	Name
18030001	GPRS Access Point Name
18030002	GPRS Event Type
18030003	GPRS Requested QOS Delay Class
18030004	GPRS Requested QOS Reliability Class
18030005	GPRS Requested QOS Peak Throughput
18030006	GPRS Requested QOS Precedence Class
18030007	GPRS Requested QOS Mean Throughput
18030008	GPRS Long Requested QOS Priority Level
18030009	GPRS Long Requested QOS Traffic Class
18030010	GPRS Long Requested QOS Delivery Order
18030011	GPRS Long Requested QOS Delivery Of Erroneous SDU
18030012	GPRS Long Requested QOS Max SDU Size
18030013	GPRS Long Requested QOS Max Bit Rate For Up Link
18030014	GPRS Long Requested QOS Max Bit Rate For Down Link
18030015	GPRS Long Requested QOS Residual BER
18030016	GPRS Long Requested QOS SDU Error Ratio
18030017	GPRS Long Requested QOS Transfer Delay
18030018	GPRS Long Requested QOS Traffic Handling Priority
18030019	GPRS Long Req QOS Guaranteed Rate For Up Link
18030020	GPRS Long Req QOS Guaranteed Rate For Down Link
18030021	GPRS Subscribed QOS Delay Class
18030022	GPRS Subscribed QOS Reliability Class
18030023	GPRS Subscribed QOS Peak Throughput
18030024	GPRS Subscribed QOS Precedence Class
18030025	GPRS Subscribed QOS Mean Throughput

Profile Tag Number	Name
18030026	GPRS Long Subscribed QOS Priority Level
18030027	GPRS Long Subscribed QOS Traffic Class
18030028	GPRS Long Subscribed QOS Delivery Order
18030029	GPRS Long Subscribed QOS Delivery Of Erroneous SDU
18030030	GPRS Long Subscribed QOS Max SDU Size
18030031	GPRS Long Subscribed QOS Max Bit Rate For Up Link
18030032	GPRS Long Subscribed QOS Max Rate For Down Link
18030033	GPRS Long Subscribed QOS Residual BER
18030034	GPRS Long Subscribed QOS SDU Error Ratio
18030035	GPRS Long Subscribed QOS Transfer Delay
18030036	GPRS Long Subscribed QOS Traffic Handling Priority
18030037	GPRS Long Sub QOS Guaranteed Rate For Up Link
18030038	GPRS Long Sub QOS Guaranteed Rate For Down Link
18030039	GPRS Negotiated QOS Delay Class
18030040	GPRS Negotiated QOS Reliability Class
18030041	GPRS Negotiated QOS Peak Throughput
18030042	GPRS Negotiated QOS Precedence Class
18030043	GPRS Negotiated QOS Mean Throughput
18030044	GPRS Long Negotiated QOS Priority Level
18030045	GPRS Long Negotiated QOS Traffic Class
18030046	GPRS Long Negotiated QOS Delivery Order
18030047	GPRS Long Negotiated QOS Delivery Of Erroneous SDU
18030048	GPRS Long Negotiated QOS Max SDU Size
18030049	GPRS Long Negotiated QOS Max Bit Rate For Up Link
18030050	GPRS Long Negotiated QOS Max Rate For Down Link
18030051	GPRS Long Negotiated QOS Residual BER
18030052	GPRS Long Negotiated QOS SDU Error Ration
18030053	GPRS Long Negotiated QOS Transfer Delay
18030054	GPRS Long Negotiated QOSTraffic Handling Priority
18030055	GPRS Long Neg QOS Guaranteed Rate For Up Link
18030056	GPRS Long Neg QOS Rate For Down Link
18030057	GPRS Routing Area Identity Country Code
18030058	GPRS Routing Area Identity Network Code
18030059	GPRS Routing Area Identity Location Area Code
18030060	GPRS Routing Area Identity Routing Area Code
18030061	GPRS Charging ID
18030062	GPRS SGSN Capabilities
18030063	GPRS Location Information Country Code
18030064	GPRS Location Information Network Code
18030065	GPRS Location Information Location Area Code
18030066	GPRS Location Information Cell Identity

Profile Tag Number	Name
18030067	GPRS Location Information SAI Present
18030068	GPRS PDP Initiation Type
18030069	GPRS GGSN Address
18030070	GPRS Secondary PDP Context
18030071	GPRS Billing Type
18030072	GPRS SGSN Number Digits
18030073	GPRS SGSN Number NoA

ApplyChargingReport.extensions Parameter

The CAP3GPRS Control Agent passes a profile block to ACS in the `ApplyChargingReport.extensions` parameter as extension type 701. The slee_acs process makes the profile tags from this profile block available to feature node through the **Session data, Incoming Session Data** option in the feature node configuration window.

ApplyCharging Report Profile Tags

This table lists the information written into ApplyChargingReport extension type 701 by the CAP3GPRS Control Agent

Field	Description
18030201	GPRS Total Transferred Volume
18030202	GPRS 32 Bit Transferred Volume
18030203	GPRS Transferred Volume Rollover

Connect.extensions Parameter

When a feature node writes session data into the feature node configuration option for outgoing session data, the slee_acs sends a profile block in the `Connect.extensions` parameter as extension type 701.The CAP3GPRS Control Agent can then use the applicable profile tags from this profile block.

Connect Profile Tags

This table lists the information derived from the outgoing connect extension type 701 by the CAP3GPRS Control Agent.

Field	Description
18030001	GPRS Access Point Name
18030071	GPRS Billing Type

Parameter Mappings

Introduction

This section describes the mappings between INAP parameters and CAP3GPRS parameters.

GPRS Parameters Received from the SGSN

This table lists the mappings for GPRS parameters received from the SGSN.

GPRS Parameter	Mapping
InitialIDPGPRS.serviceKey	Identifies the correct service using the configuration defined in the CAP3GPRS, Services section of **eserv.config**.
InitialIDPGPRS.gPRSEventType	Put in IDP.incomingExtensions. Used by the state machine logic, for example, when selecting billing type.
InitialIDPGPRS.requestedQOS	Put in IDP.incomingExtensions.
InitialIDPGPRS.subscriberQOS	Put in IDP.incomingExtensions.
InitialIDPGPRS.negotiatedQOS	Put in IDP.incomingExtensions.
InitialIDPGPRS.routeingArealIdentity	Put in IDP.incomingExtensions.
InitialIDPGPRS.chargingID	Put in IDP.incomingExtensions.
InitialIDPGPRS.sGSNCapabilities	Put in IDP.incomingExtensions.
InitialIDPGPRS.cellGlobaIdOrServiceAreaIdOrLAI	Put in IDP.incomingExtensions.
InitialIDPGPRS.sai_Present	Put in IDP.incomingExtensions.
InitialIDPGPRS.pDPInitiationType	Put in IDP.incomingExtensions.
InitialIDPGPRS.gGSNAddress	Put in IDP.incomingExtensions.
InitialIDPGPRS.sgsnNumber	Put in IDP.incomingExtensions.
InitialIDPGPRS.secondaryPDP-context	Put in IDP.incomingExtensions.
InitialIDPGPRS.mSISDN	Put in IDP.callingpartynumber.
InitialIDPGPRS.iMSI	Put in IDP.IMSI.
InitialIDPGPRS.accessPointName	Put in IDP.incomingExtensions.
EntityReleasedGPRS.gPRSCause	Put in EventReportBCSM.routeSelectFailureSpecificInfo
ApplyChargingReportGPRS.volumeIfNoTariffSwitch	Add 4294967296 multiplied by transferredVolumeRollover. Divide by conversion factor and put in ApplyChargingReport.timeifNoTariffSwitch. Put the unmodified value of volumeIfNoTariffSwitch (without applying the conversion factor) in ApplyChargingReport.incomingExtensions.
ApplyChargingReportGPRS.transferredVolumeRollover	Add 4294967296 multiplied by transferredVolumeRollover. Divide by conversion factor and put in ApplyChargingReport.timeifNoTariffSwitch. Put the unmodified value of transferredVolumeRollover (without applying the conversion factor) in ApplyChargingReport.incomingExtensions.

GPRS Parameter	Mapping
ApplyChargingReportGPRS.timeGPRSIfNoTariffSwitch	Multiply by 10 and put in ApplyCharging.timeifNoTariffSwitch.
ApplyChargingReportGPRS.active	Used in state machine logic and put in ApplyChargingReport.callActive.
EventReportGPRS.GPRSEventType	Used in state machine logic.

GPRS Parameters Sent to SGSN

This table lists the GPRS parameters sent to SGSN, and defines how they are set.

GPRS Parameter	Definition
RequestReportGPRS.eventtypeGPRS	Set by state machine logic.
RequestReportGPRS.monitorMode	Set by state machine logic.
ApplyChargingGPRS.maxElapsedTime	Set to ApplyCharging.maxDuration divided by 10 for duration billing.
ApplyChargingGPRS.maxTransferredVolume	Set to ApplyCharging.maxDuration multiplied by conversion factor for volume billing.
connectGPRS.accessPointName	Set from Connect.outgoingExtensions.
FurnishChargingGPRS.fciGPRSBillingCharacteristics	Set from FurnishChargingInformation.fciBillingChargingCharacteristics.
SendChargingGPRS.sciGPRSBillingChargingCharacteristics	Set from SendChargingInformation.sciBillingChargingCharacteristics.

INAP Parameters Sent to slee_acs

This table lists the INAP parameters sent to slee_acs, and defines how they are set.

INAP Parameter	Definition
IDP.callingPartyNumber	Set to IDPGPRS.mSISDN.
IDP.iMSI	Set to IDPGPRS.iMSI.
IDP.calledPartyBCDNumber	Set from the calledpartyNumber defined in the CAP3GPRS, Services section of **eserv.config**.
IDP.eventTypeBCSM	Set to analyzedInformation.
IDP.bearerCapability	Set to indicated "Speech" for duration billing or "unrestricted digital information with tones and announcements" for volume billing.
IDP.extensions	The binary data that is placed in extension 701 (incoming extensions), and that contains various pieces of information, such as the transferredVolumeRollover value.
ApplyChargingReport.callActive	Set to ApplyChargingReportGPRS.active or to false if in StateOpenFinal.

INAP Parameter	Definition
ApplyChargingReport.callReleaseAtTcpExpiry	Only set if ApplyChargingReportGPRS (active = true) is received in StateOpenFinal.
ApplyChargingReport.timeifNoTariffSwitch	Set to one of: • 10 * (elapsedTime + 86400 * elapsedTimeRollover) for duration billing • (transferredVolume + 4294967296) / conversion factor for volume billing
EventReportBSCM.routeSelectFailureSpecificInfo	Set to EntityReleaseGPRS.gPRSCause.
EventReportBCSM.messageType	If EventReportBCSM.eventTypeBCSM is set to: • routeSelectFailure (RSF) then EventReportBCSM.messageType is set to request. • oDisconnect or oAnswer then EventReportBCSM.messageType is set to notification.
EventReportBCSM.receivingSideID	Set to leg 1.
EventReportBCSM.oDisconnectSpecificInfo	Set to 31.
EventReportBCSM.eventTypeBCSM	Set according to state machine logic to one of: • routeSelectFailure • oAnswer • oDisconnect.

INAP Parameters Received from slee_acs

This table lists the INAP parameters received from slee_acs and how they are mapped in GPRS.

INAP Parameter	GPRS Mapping
ApplyCharging.releaseDurationExceeded	Put into either StateOpen or StateOpenFinal as determined by the state machine.
ApplyCharging.maxDuration	For duration billing, divide by ten and put in the ApplyChargingGPRS.maxElapsedTime parameter. For volume billing, multiply by conversion factor and put in the ApplyCargingGPRS.maxTransferredVolume parameter.
Connect.extensions	Outgoing extensions (extension 701) used to change billing type to set access point name in ConnectGPRS.
EventReportBCSM.eventTypeBCSM	Used in state machine logic.
EventReportBCSM.monitormode	Used in state machine logic.
Connect.destinationRoutingAddress	Ignored.
ReleaseCall.reason	Used in ReleaseGPRS.fciGPRSBilling.
FurnishChargingInformation.fciBilling	Put in FurnishChargingGPRS.fciGPRSBilling ChargingCharacteristics

INAP Parameter	GPRS Mapping
`ChargingCharacteristics`	
`SendChargingInformation.sciBilling ChargingCharacteristics`	Put in SendChargingGPRS.sciGPRSBilling ChargingCharacteristics.

Chapter 3

Background Processes

Overview

Introduction

This chapter describes the background process that is run automatically by the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GPRS Control Agent) and lists the statistics that are logged by the control agent.

In this chapter

This chapter contains the following topics.

cap3gprsControlAgent Process

Purpose

The cap3GprsControlAgent binary is a SLEE application that translates between CAP3 GPRS messages and CAP3 INAP operations. It enables a CAP3 SGSN client to talk to a CAP3 SCF such as CCS or ACS.

Startup

The cap3gprsControlAgent binary process is started automatically by the SLEE. For more information, see *SLEE.cfg Configuration* (on page 18).

Configuration

The cap3gprsControlAgent binary process is configured by the CAP3GPRS section in **eserv.config**. For details, see *CAP3GPRS* **eserv.config** *Configuration* (on page 6).

Failure

If the cap3gprsControlAgent fails, no operations will be processed for the CAP3GPRS Control Agent.

Statistics Logged by the CAP3GPRS Control Agent

Introduction

CAP3GPRS statistics are generated by each Service Logic Controller (SLC), and then transferred at periodic intervals to the Service Management System (SMS) for permanent storage and analysis.

The statistics are logged by the smsStats functions provided by the SMS application. For more information, see *SMS Technical Guide*.

CAP3GPRS Statistics

The following statistics are logged for the CAP3GPRS Control Agent. Each statistic logged has APPLICATION_ID = ‘CAP3GPRS’ (application number 113):

- IDPGPRS_RECEIVED
- GPRS_TIMED_OUT
- IN_TIMED_OUT
- ENDED_IN_ERROR
- CHARGED_USER_DISCONNECTED
- CHARGED_TERMINATED_NO_FUNDS
- RELEASED
- NOT_CHARGED
- CHARGED
- CHARGED_ERROR
- CONTINUE_GPRS_NO_TRIGGER
- IDP_SENT
- NOT_CHARGED_CONNECT
- NOT_CHARGED_CONTINUE

For each statistic, the gprsEventType from the original InitalDPGPRS is put into the statistics detail field, SMF_STATISTICS.DETAIL, in the form of a string (for example: "ContextEstablishment").

CAP3GPRS Statistics Reports

You can generate the following statistics reports for the CAP3GPRS Control Agent:

- CAP3GPRS System Stats
- CAP3GPRS System Stats by EDP

For each report type, you can specify whether to report on all entries or only totals.
When you run the CAP3GPRS System Stats by EDP report, you specify the EDP to report on by selecting one of the following EDPs:

- attach
- attachChangeOfPosition
- ContextEstablishment
- ContextEstablishmentAcknowledgement, or
- Unsupported

You generate reports from the Report Functions window in the SMS UI. See *SMS User's Guide* for more information.

Chapter 4

CAP3GPRS Operations and Message Sequences

Overview

Introduction

This chapter lists the INAP and GPRS operations supported by the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GPRS Control Agent), and provides example message sequence charts that show the flows through the software when messages are sent or received by the CAP3GPRS Control Agent.

In this chapter

This chapter contains the following topics.

Supported CAP3 GPRS and CAP3 INAP Operations

CAP3 GPRS Supported Operations

The CAP3GPRS Control Agent supports the following CAP3 GPRS operations:

- ApplyChargingReportGPRS
- EntityReleasedGPRS
- EventReportGPRS
- InitialDPGPRS
- ApplyChargingGPRS
- RequestReportGPRSEvent
- ConnectGPRS
- ContinueGPRS
- ReleaseGPRS
- SendChargingInformationGPRS
- FurnishChargingInformationGPRS

CAP3 INAP Supported Operations

The CAP3GPRS Control Agent supports the following CAP3 INAP operations:

- ApplyCharging
- ApplyChargingReport
- Connect

- Continue
- EventReportBCSM
- FurnishChargingInformation
- InitalDP
- ReleaseCall
- RequestReportBCSMEvent
- SendChargingInformation

GPRS Context Flows

Introduction

This section details the GPRS PDP context flows that the CAP3GPRS Control Agent can produce. For all these message sequences the `IDPGPRS.gprsEventType` is 11 (ContextEstablishment).

Received ApplyCharging(releaseIfDurationExceeded)

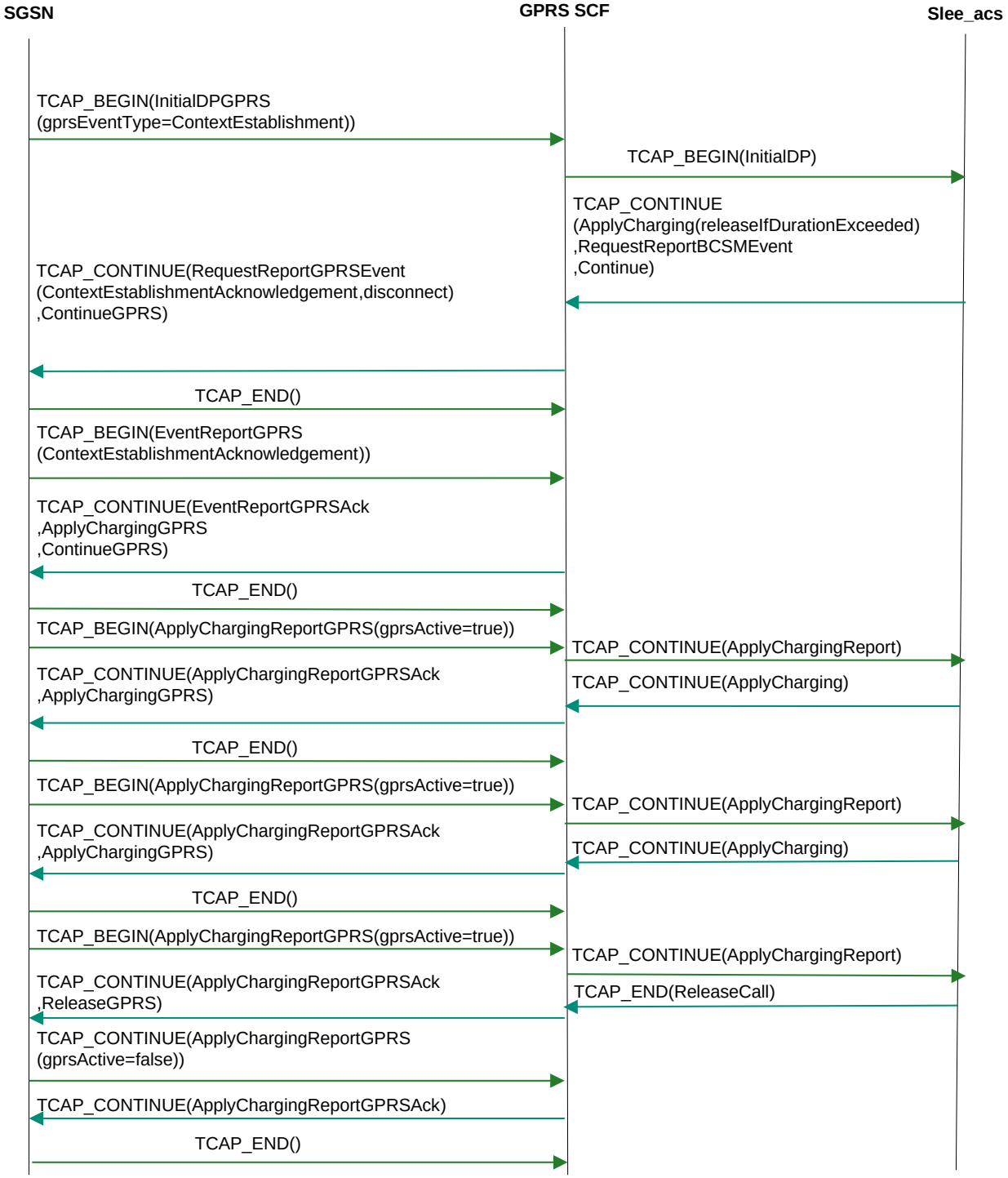
Here is an example message sequence for an ApplyCharging(releaseIfDurationExceeded) operation received by the CAP3GPRS Control Agent.

Note: Billing can be configured for either duration or volume.


SGSN
GPRS SCF
Slee_acs
TCAP_BEGIN(InitialDPGPRS
(gprsEventType=ContextEstablishment))
TCAP BEGIN(InitialDP)
TCAP_CONTINUE
(ApplyCharging(releaseIfDurationExceeded)
,RequestReportBCSMEvent
,Continue)
TCAP_CONTINUE(RequestReportGPRSEvent
(ContextEstablishmentAcknowledgement)
,ContinueGPRS)
TCAP_END()
TCAP_BEGIN(EventReportGPRS
(ContextEstablishmentAcknowledgement))
TCAP_CONTINUE(EventReportGPRSAck
,ApplyChargingGPRS
,ContinueGPRS)
TCAP_END()
TCAP_BEGIN(ApplyChargingReportGPRS
(gprsActive=true))
TCAP_END(ApplyChargingReport)
TCAP_CONTINUE(ApplyChargingReportGPRSAck
,ReleaseGPRS)
TCAP_CONTINUE(ApplyChargingReportGPRS
(gprsActive=false))
TCAP_CONTINUE(ApplyChargingReportGPRSAck)
TCAP_END()

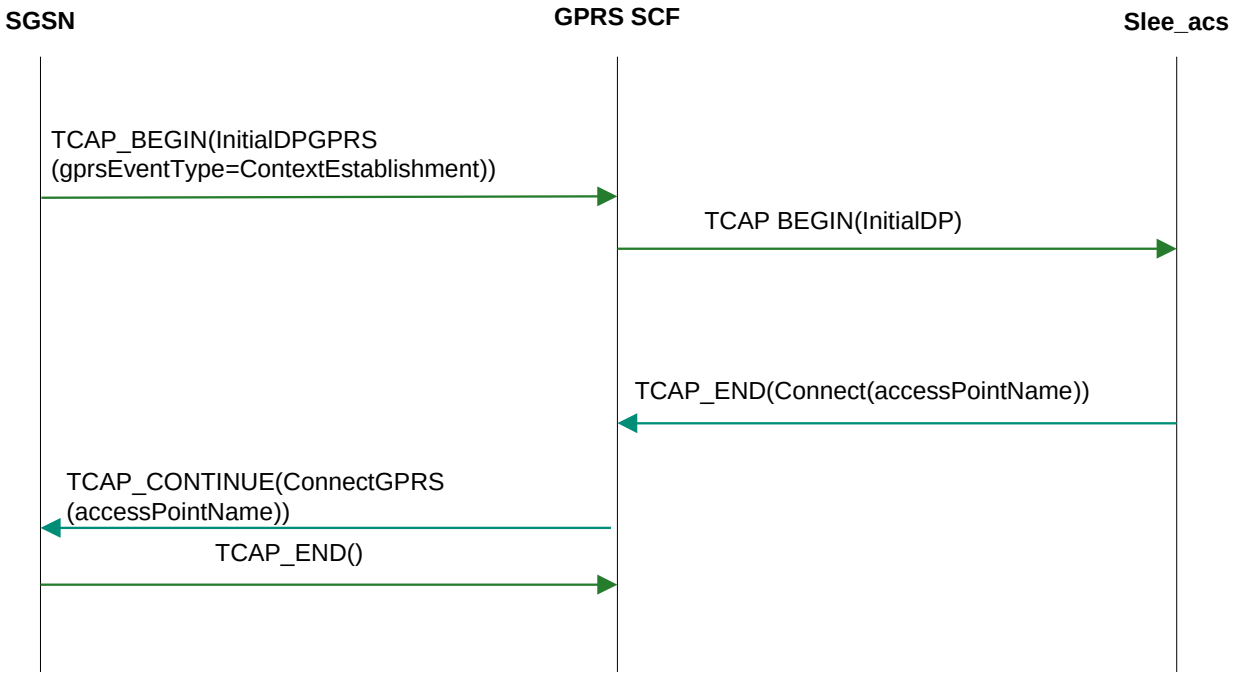
Credit Expiry Example 1

Here is an example message sequence for a credit expiry after several successful balance updates.



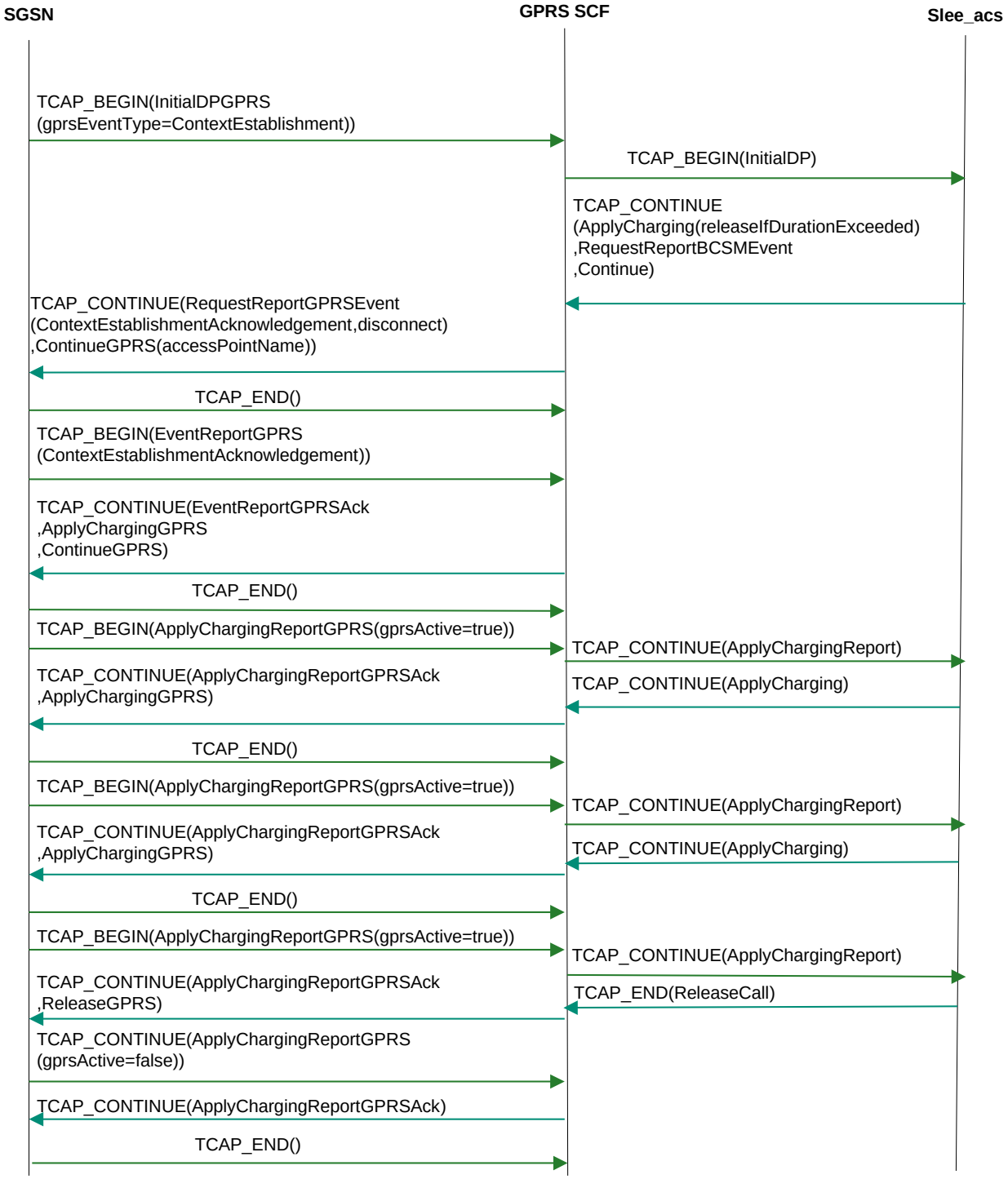
INAP Connect, New accessPointName, with No Charging

Here is an example message sequence where an INAP connect specifies a new accessPointName with no charging.



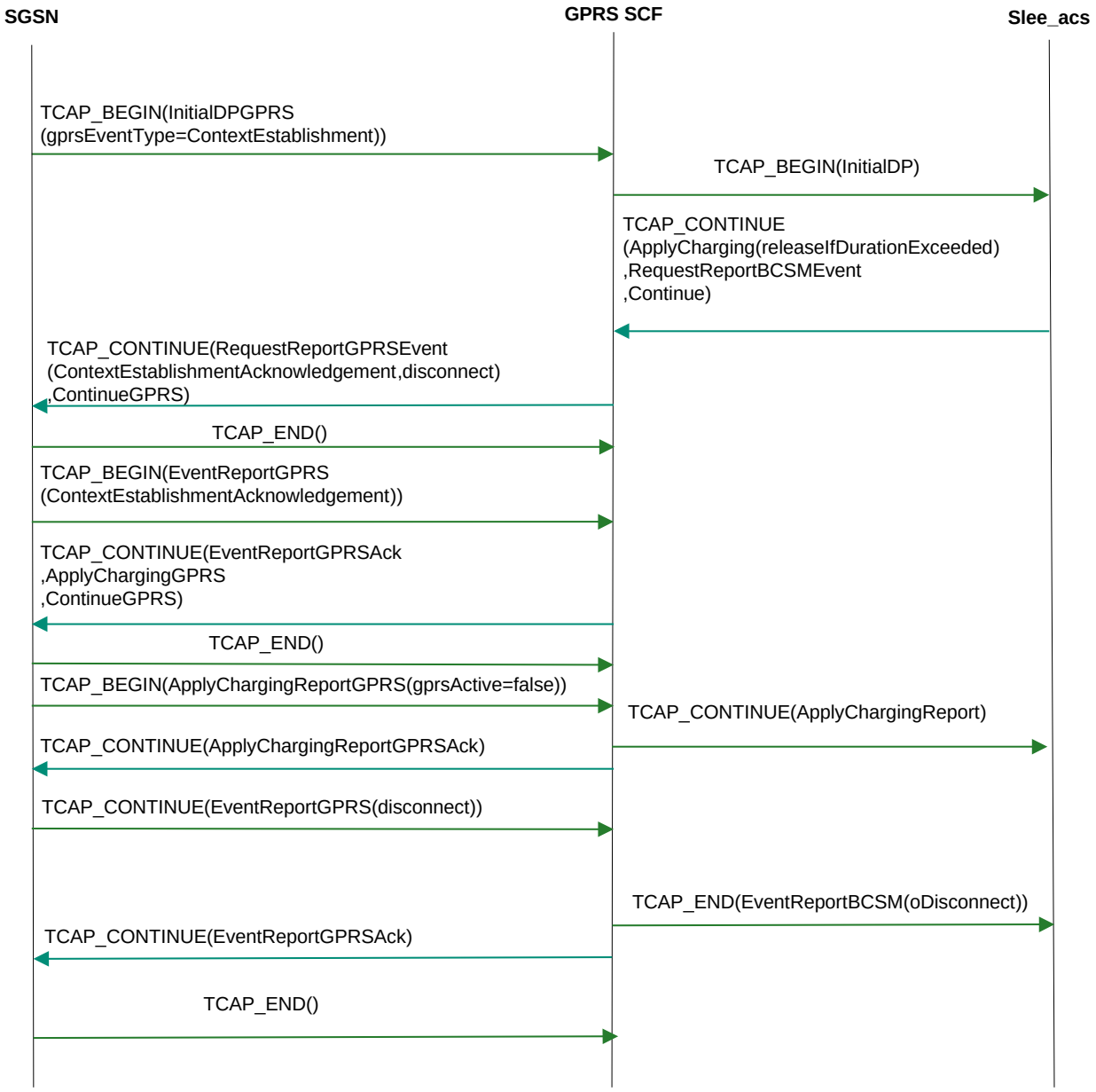
INAP Connect, New accessPointName, with Charging

Here is an example message sequence where an INAP connect specifies a new accessPointName with charging.



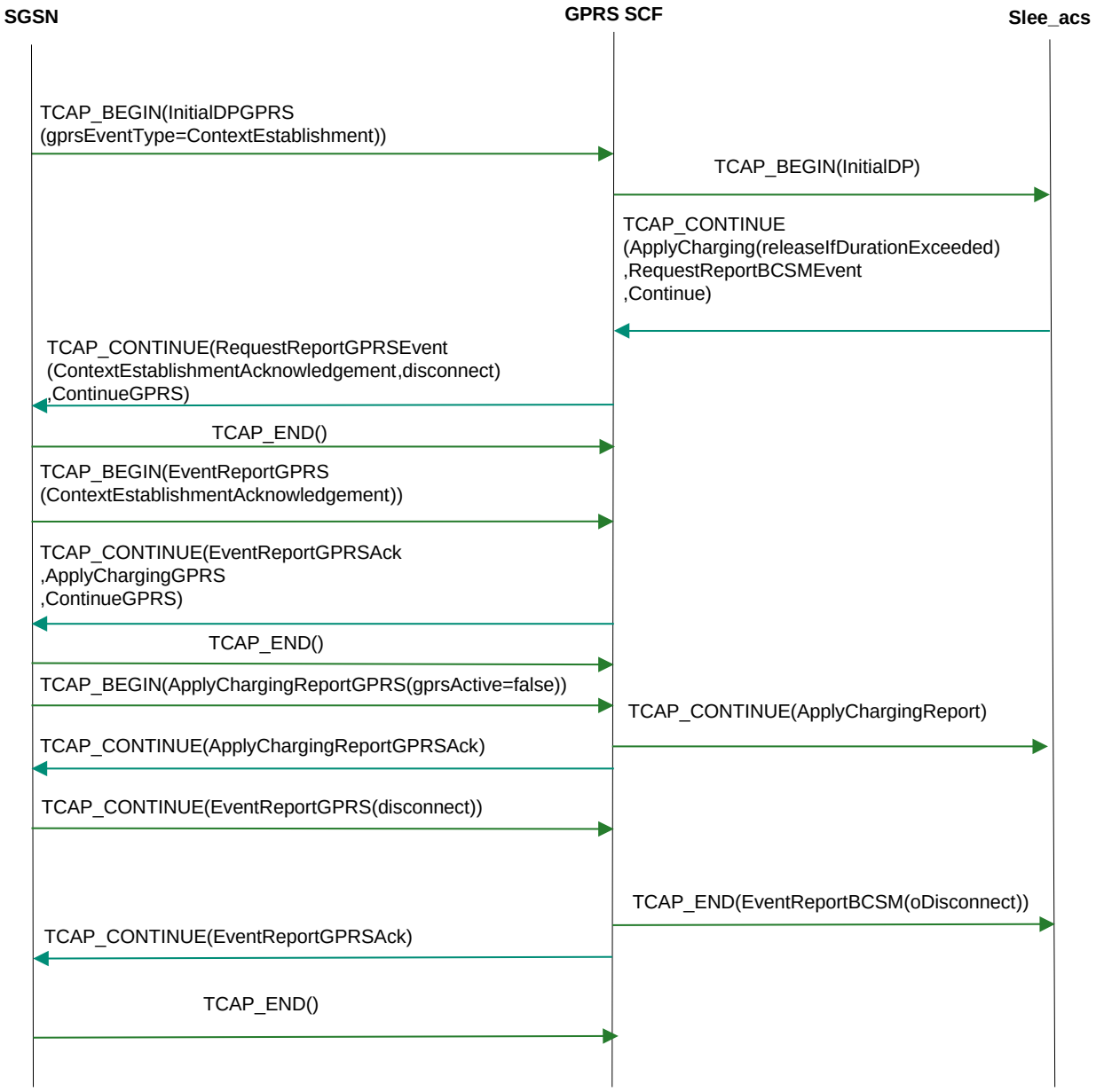
INAP Continue Received

Here is an example message sequence for an INAP Continue received.



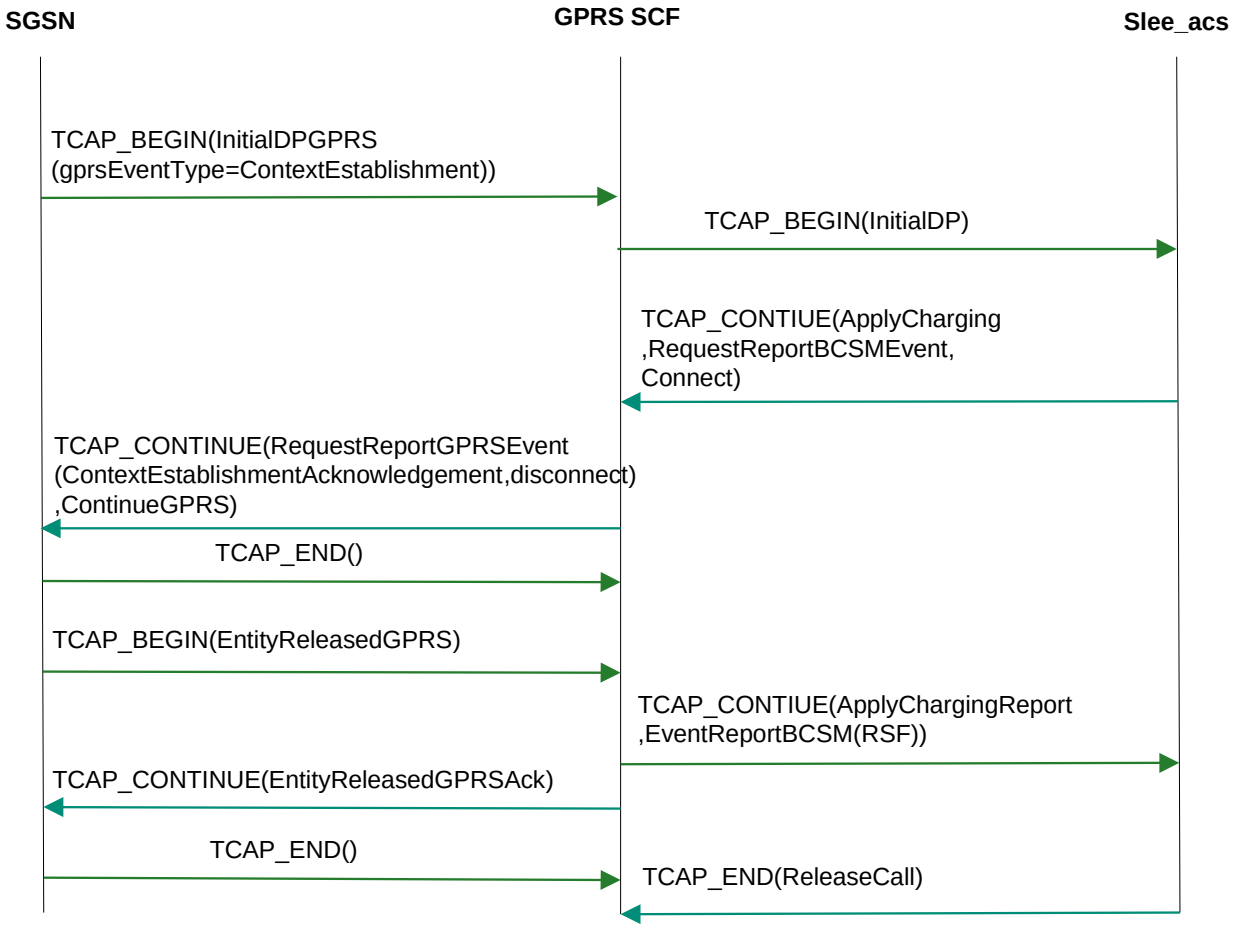
User Disconnects with Charging

Here is an example message sequence for user disconnects with charging.



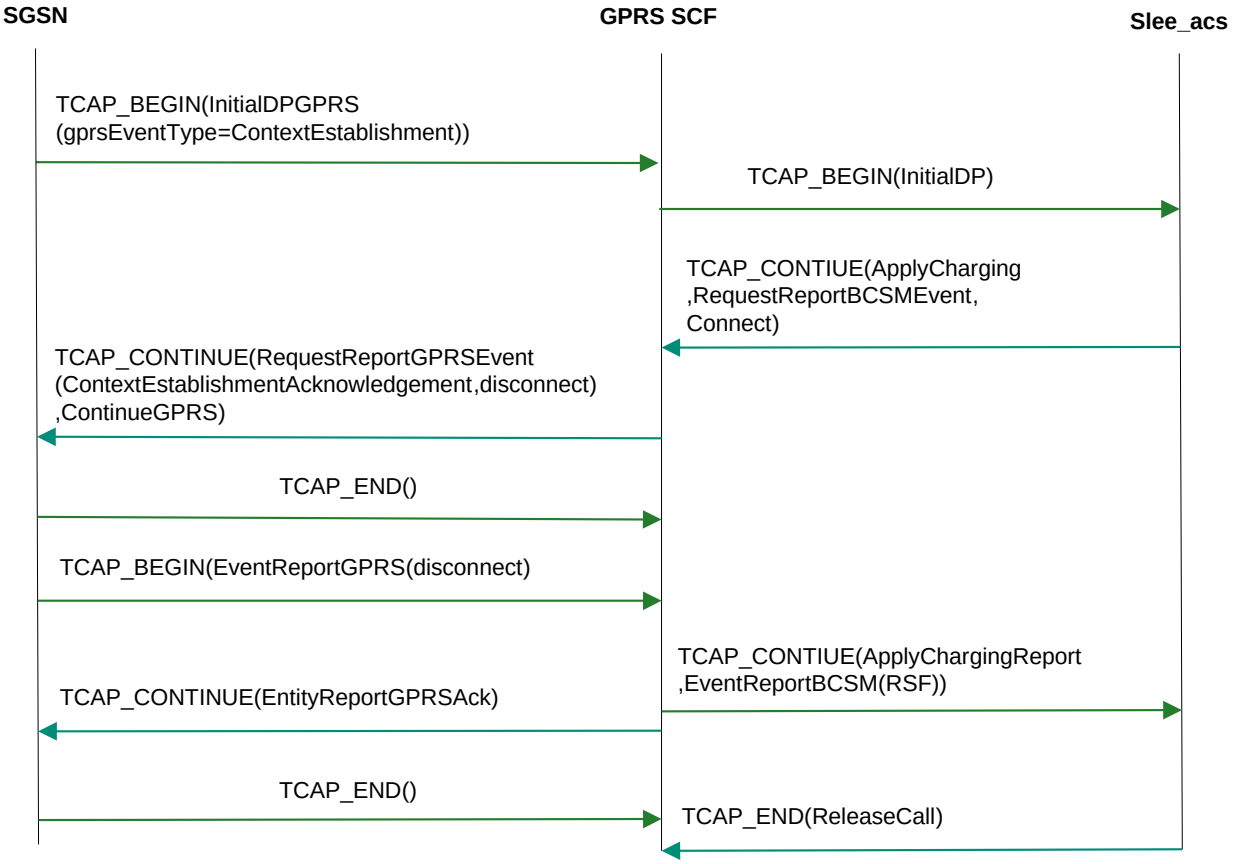
PDP Context Detached, Example 1

Here is an example message sequence for a PDP context detached before context establishment acknowledgement (first variant).



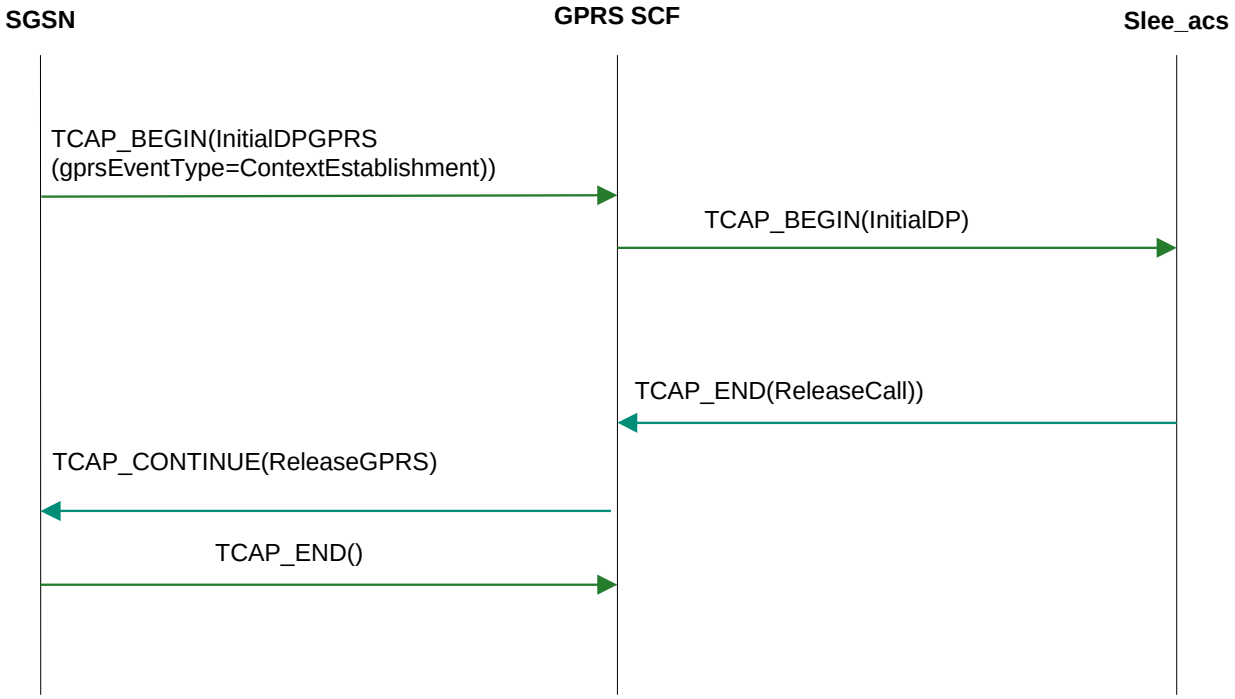
PDP Context Detached, Example 2

Here is an example message sequence for a PDP context detached before context establishment acknowledgement (second variant).



ReleaseCall Operation Received from slee_acs

Here is an example message sequence for a ReleaseCall operation received from slee_acs.



Message Flows for GPRS Sessions

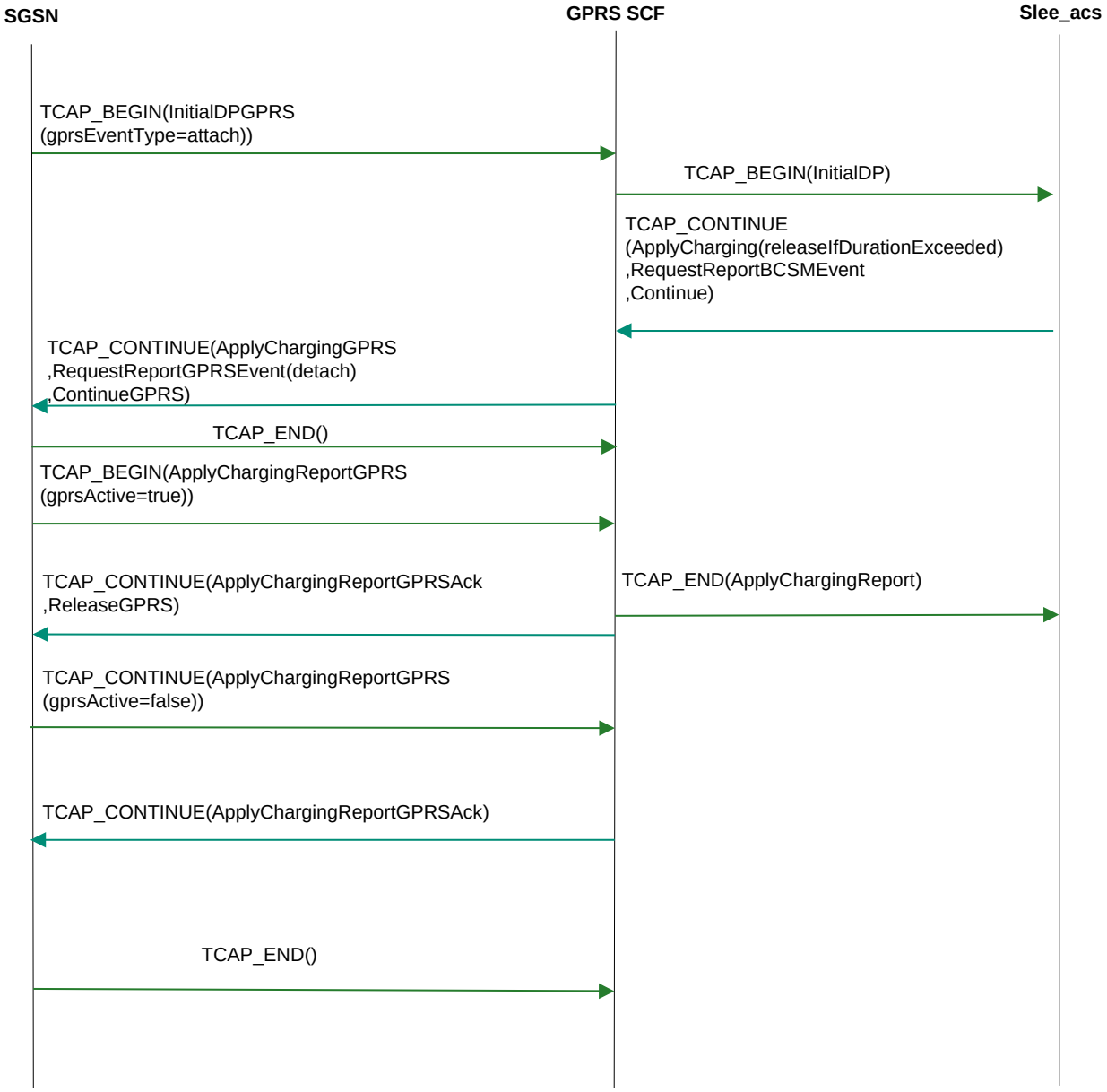
Introduction to GPRS Sessions

This section details the message sequences that the CAP3GPRS Control Agent can produce for GPRS sessions. The message sequences start with an InitalDPGPRS for the session as a whole (rather than for individual PDP contexts). The CAP3GPRS Control Agent will not arm the ContextEstablishment or ContextEstablishmentAcknowledgement EDPs for these sessions. This means that billing for individual PDP contexts must be done by using separate InitialDPGPRS operations, and therefore the control agent will only charge for the session as a whole. The IDPGPRS.gprsEventType is 1 (attach) in all cases.

Note: Volume billing is not available for GPRS sessions.

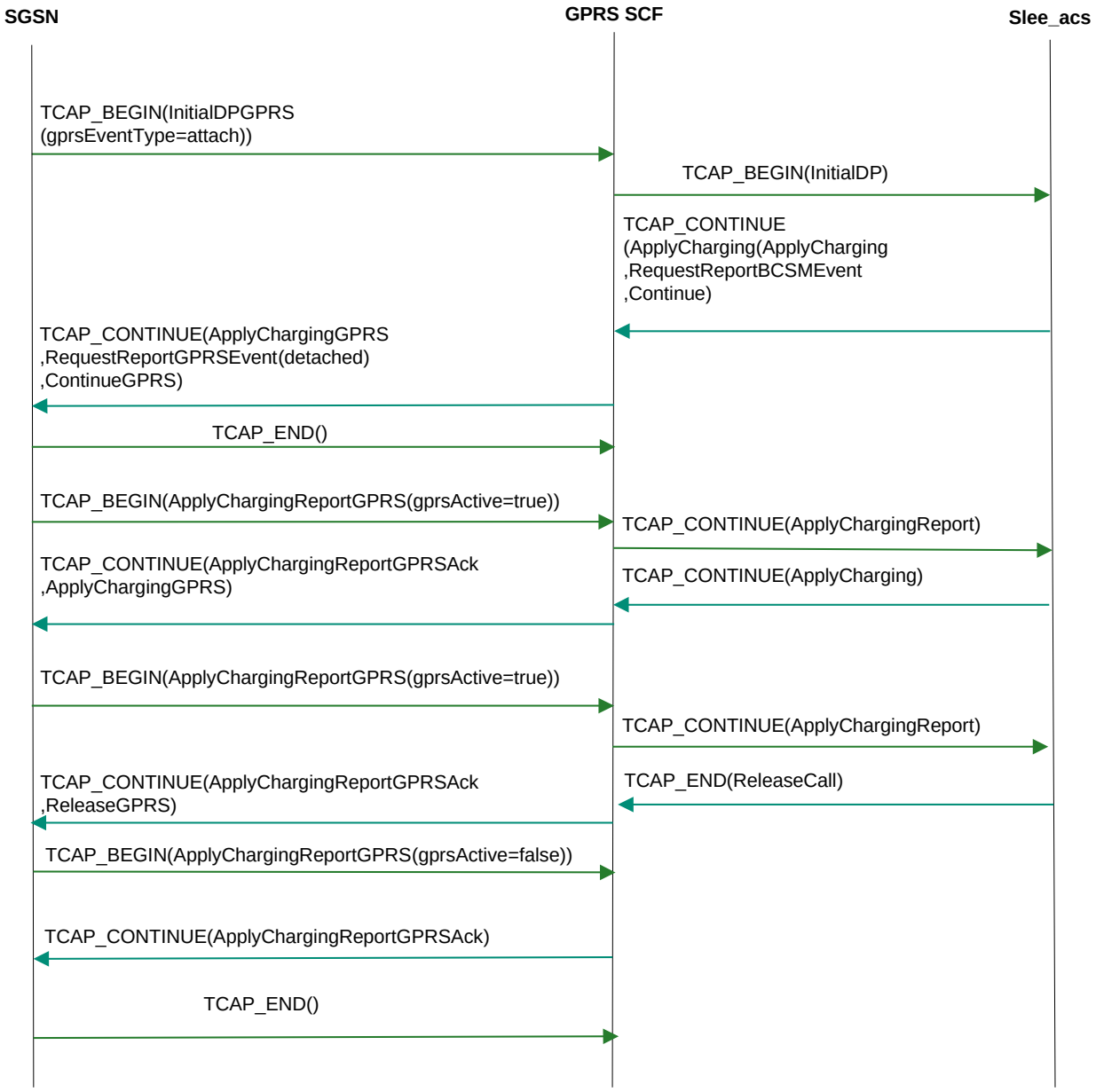
Received ApplyCharging(releaseDurationExceeded)

Here is an example message sequence for a ApplyCharging(releaseDurationExceeded) operation received by the CAP3GPRS Control Agent.



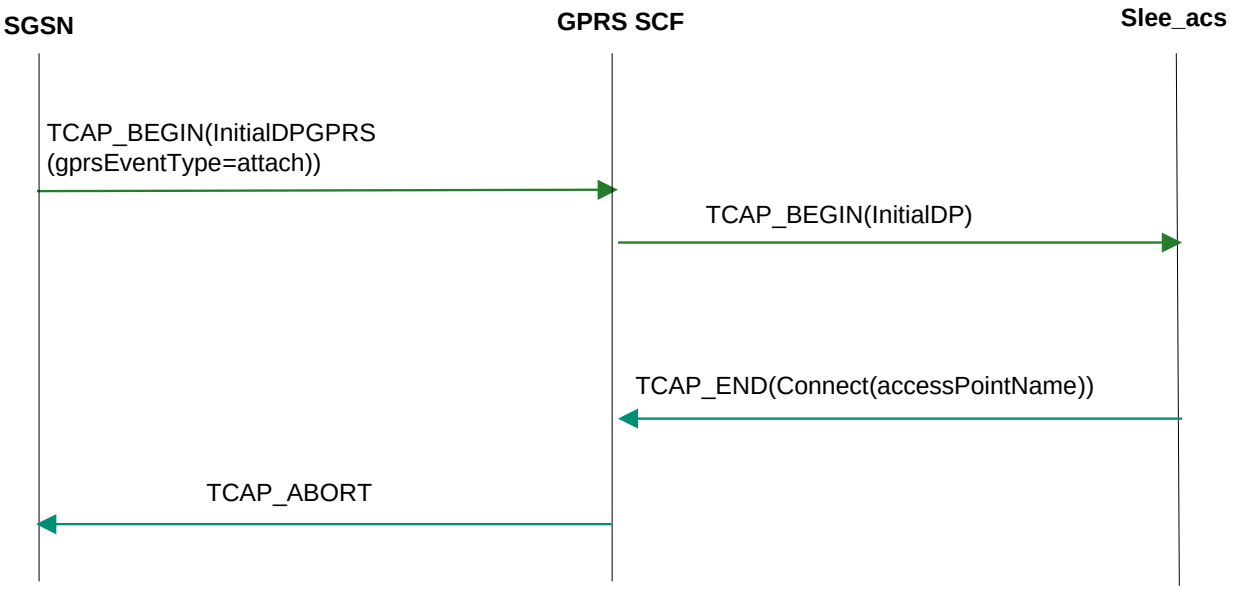
Credit Expiry, Example 2

Here is an example message sequence for credit expiry after several successful balance updates.



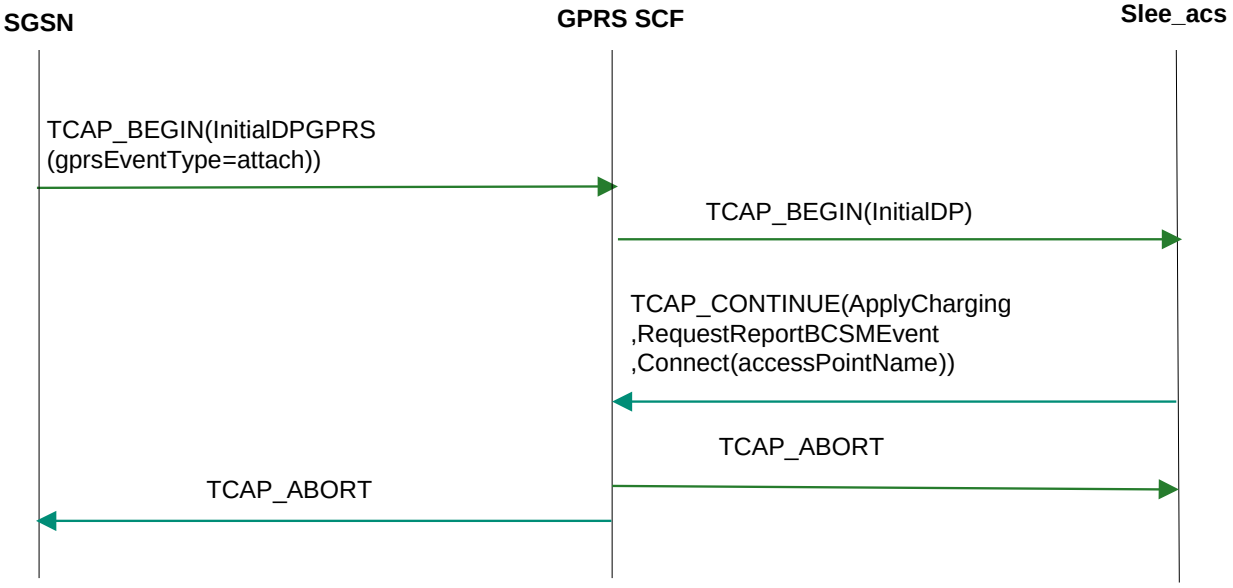
INAP Connect, New accessPointName with No Charging

Here is an example message sequence for INAP connect request that specifies new accessPointName with no charging. This shows what can happen when an error occurs during control plan processing. The CAP3GPRS protocol does not support changing accessPointName for event types other than ContextEstablishment and therefore aborts the dialog.



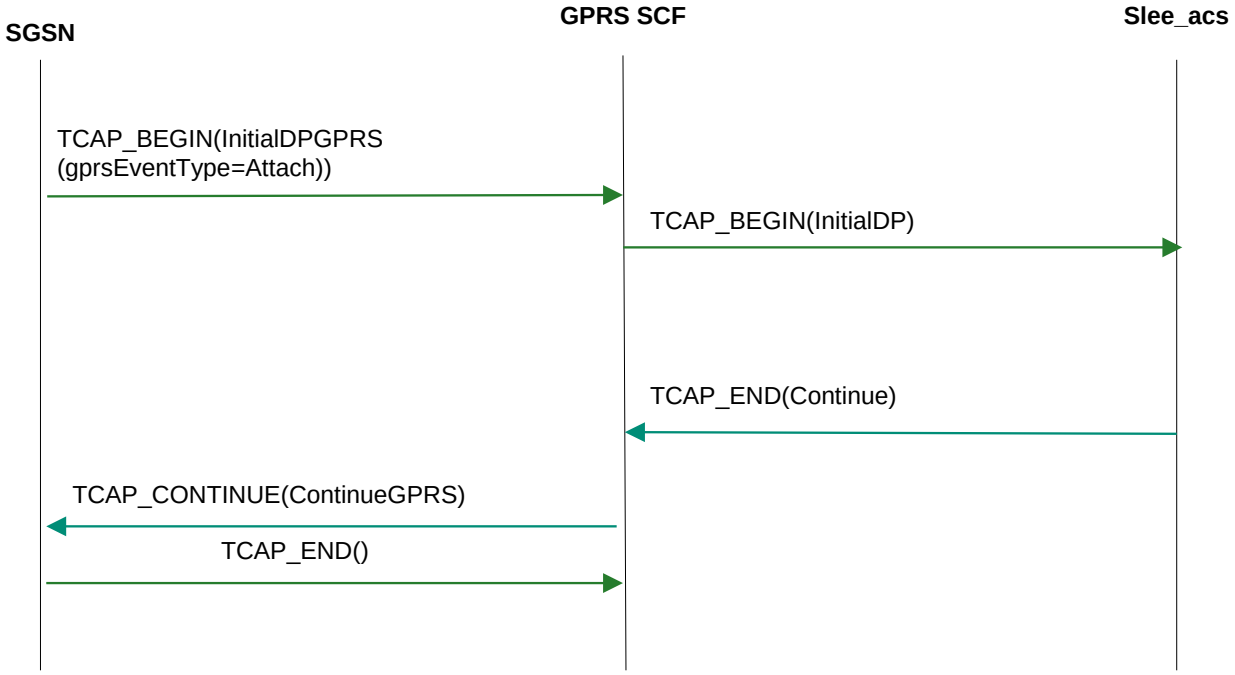
INAP Connect, New accessPointName, with Charging

Here is an example message sequence for INAP connect request that specifies new accessPointName with charging. This example shows what can happen when an error occurs during control plan processing. The CAP3GPRS protocol does not support changing accessPointName for event types other than ContextEstablishment and therefore aborts the dialog.



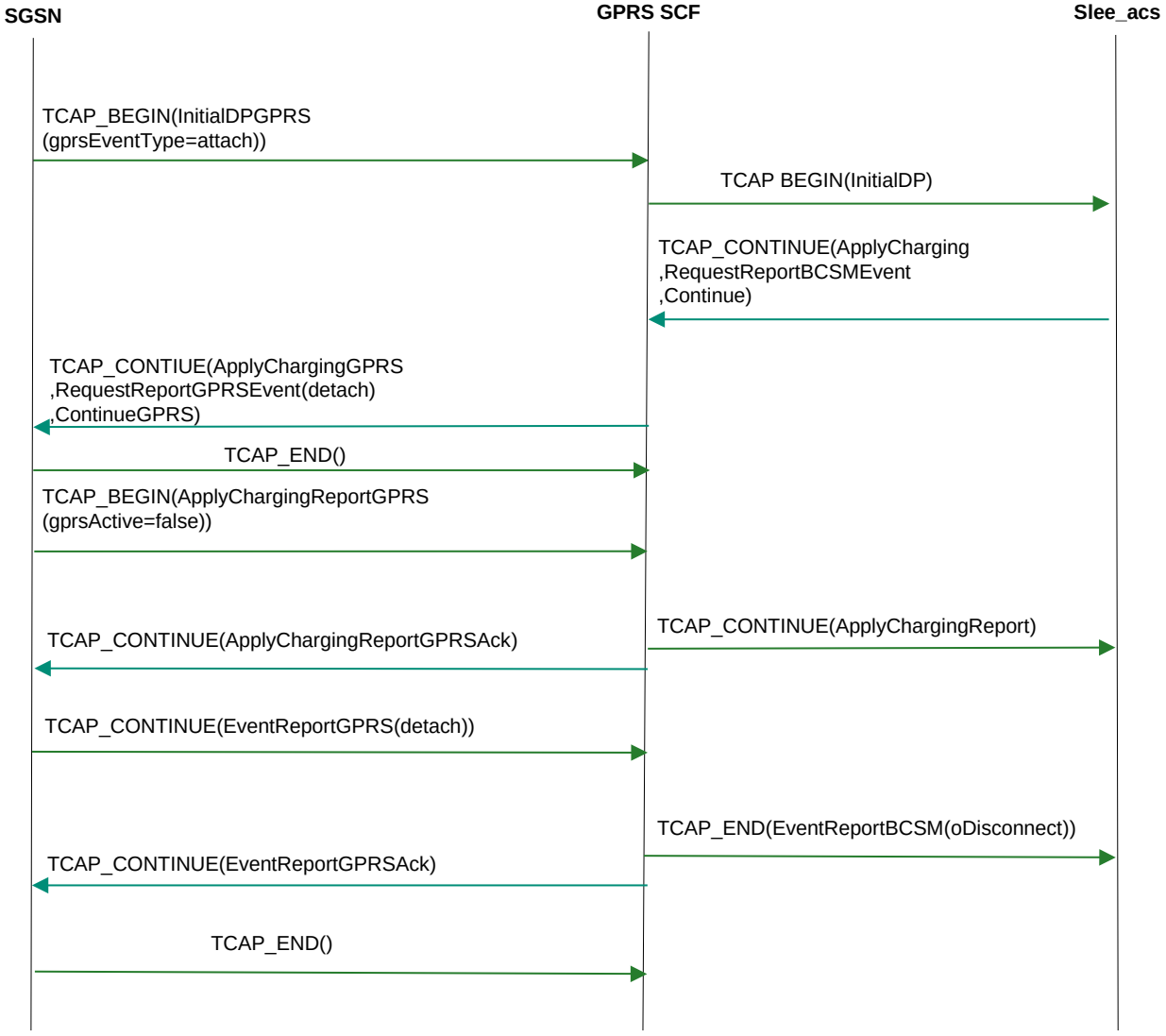
INAP Continue Received

Here is an example INAP continue received message sequence.



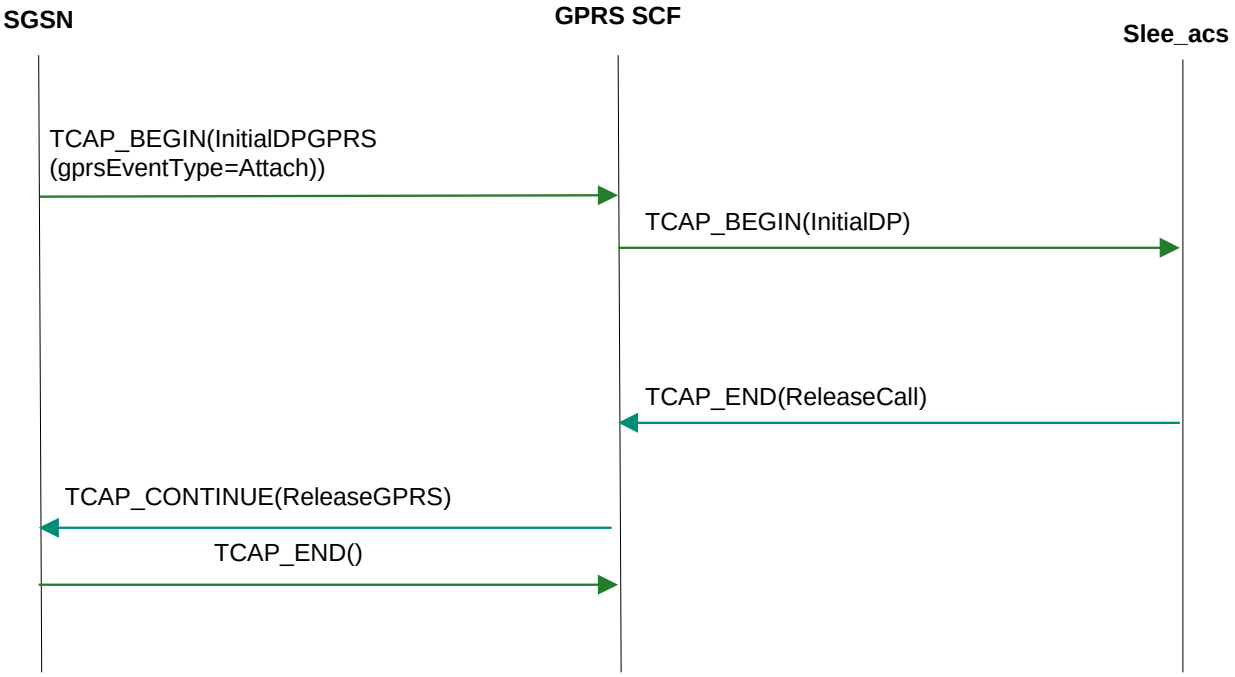
User Disconnects with Charging

Here is an example user disconnects with charging message sequence.



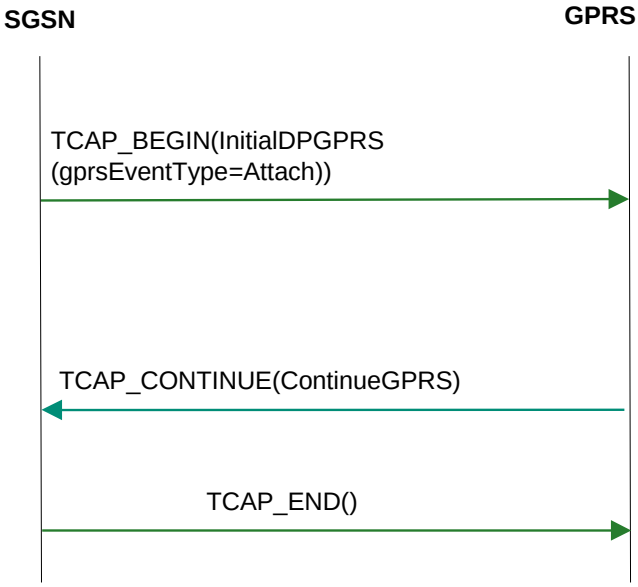
ReleaseCall Operation from slee_acs

Here is an example message sequence for a ReleaseCall operation received from slee_acs.



No Charging for Session

Here is an example message sequence for sessions with no charging.



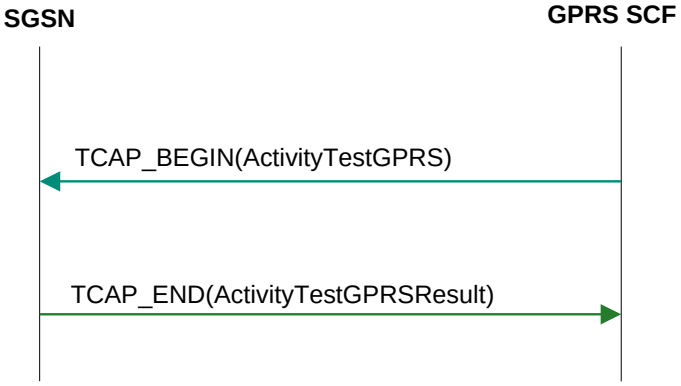
Activity Test Flows

Inactive Session Check

If a session has been inactive (no TCAP messages have been received) for longer than the maximum period configured for the operation timer, then the CAP3GPRS Control Agent sends an ActivityTestGPRS to the SGSN to determine whether the session is still open.

ActivityTestGPRS Message Sequence Example

Here is an example ActivityTestGPRS message sequence. The session is aborted if it is no longer active.



Chapter 5

Troubleshooting

Overview

Introduction

This chapter explains how to investigate message sequences and memory leaks in the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GRPS Control Agent).

In this chapter

This chapter contains the following topics.

Message Sequences and Memory Leaks

Determining the Message Sequence for a Session

This example explains how you can set the `Tracing.OrigAddress` and `Tracing.traceDebugLevel` parameters and then determine the message sequence for a session by starting a data session using a test phone.

Step	Action
1	Open the **eserv.config** configuration file, located in the **/IN/service_packages/** directory, in a text editor.

2	Set `Tracing.OrigAddress` to:
	`OrigAddress = [`
	`"num"]`
	Where *num* is your test phone number (that is, the calling number).

3	Set `Tracing.traceDebugLevel` to: `cap3gprsMessageSequences`
4	Save and close the **eserv.config** file.
5	Force the cap3gprsControlAgent to re-read its configuration by entering the following command: `-kill -HUP pid` Where *pid* is the PID for the cap3gprsControlAgent binary.
6	Start a data session using your test phone.
7	Look at the **cap3gprsControlAgent.log** file. You should see a line like this for each test session: `SLEECALLID num GPRS SCF->slee_acs:TCAP_BEGIN(InitialDP)` Where *num* is your test phone number.

Step	Action
8	Copy the lines relevant to your test session to a text file and remove the first two columns. For example, type: `grep num|sed 's/SLEECALLID[0-9]*//g' cap3gprsControlAgent.log > sequence.txt` Where *num* is your test phone number. **Tip:** All the lines relating to your test session will have the same SLEECALLID.
9	Use a web sequence diagrams tool, such as PlantUMLServer, to view the sequence chart for the message. See the documentation for the sequence diagrams tool for information on how to use it.

Finding the Cause of a Memory Leak

To find the cause of a memory leak:

Step	Action

1	Insert the following lines into the **cap3GprsControlAgent.sh** startup file immediately before the exec line:
	`DEBUG=cap3gprsObjectReport`
	`export DEBUG`

2	Ensure that a debug line is not written each time an object is created or deleted by adding the following line in the CAP3GPRS section of **eserv.config**: `minimumInstancesForObjectCounting = 1000`
3	Restart the SLEE by entering the following command as the user root: `SUPPORT/bin/slee-ctrl start` Each time the number of a particular type of object reaches a multiple of 1000, a line will be written to the log file detailing the number of this type of object. This will make any object that is leaking easy to identify.

Example cap3GprsControlAgent.sh Startup File

```
#!/usr/bin/bash
DEBUG=cap3gprsObjectReport
export DEBUG
exec /IN/service_packages/CAP3GPRS/bin/cap3GprsControlAgent >>
/IN/service_packages/CAP3GPRS/tmp/cap3GprsControlAgent.log
```

Chapter 6

About Installation and Removal

Overview

Introduction

This chapter provides information about the installed components for the Oracle Communications Convergent Charging Controller CAP version 3 GPRS Control Agent (CAP3GPRS Control Agent) and the CAP3GPRS files you can check for to ensure the control agent was successfully installed.

In this chapter

This chapter contains the following topics.

Installation and Removal Overview

Introduction

For information about the following requirements and tasks, see *Installation Guide*:

- Convergent Charging Controller system requirements
- Pre-installation tasks
- Installing and removing Convergent Charging Controller packages

CAP3GPRS Control Agent Installed Packages

When you install the CAP3GPRS Control Agent, the following packages are installed:

- The cap3gprsSms package on the SMS node
- The cap3gprsScp package on the SLC node

Checking the cap3gprsSms Installation

Check that the following directories have been created on the SMS:

/IN/service_packages/CAP3GPRS/db

/IN/service_packages/CAP3GPRS/lib

Checking the cap3gprsScp Installation

Check that the following directories have been created on the SLC:

/IN/service_packages/CAP3GPRS/bin

/IN/service_packages/CAP3GPRS/etc

/IN/service_packages/CAP3GPRS/lib

/IN/service_packages/CAP3GPRS/tmp

Check that the binary and the example configuration file for the CAP3GPRS Control Agent have been installed on the SLC:

/IN/services_packages/CAP3GPRS/bin/cap3GprsControlAgent

/IN/services_packages/CAP3GPRS/etc/eserv.config.example