Oracle® Communications Convergent Charging Controller

Location Capabilities Pack Technical Guide

Release 12.0.0

December 2017

ORACLE®

Copyright

Contents

About This Document

Scope

The scope of this document includes all the information required to install, configure and administer the Location Capabilities Pack (LCP) application.

Audience

This guide was written primarily for system administrators and persons installing, configuring and administering the LCP 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 an essential prerequisite 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.

Although it is not a prerequisite to using this guide, familiarity with the target platform would be an advantage.

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:

- *Location Capabilities Pack User's Guide*
- *IS-41 IS-848 Position Request Specification*
- *GSM 03.32 MAP Specification*

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 application. It explains the basic functionality of the system and lists the main components.

It is not intended to advise on any specific Oracle Communications Convergent Charging Controller network or service implications of the product.

In this Chapter

This chapter contains the following topics.

What is Location Capabilities Pack?

Introduction

The Oracle Communications Convergent Charging Controller Location Capabilities Pack (LCP) is a set of software components used by other applications to look up the location of mobile devices.

Components

The LCP consists of these components:

- Location Module (locApp)
- MAP ATI plug-in
- IS-41 POSREQ plug-in
- SRI-MSRN plug-in
- SRI-IMSI plug-in
- LCP ACS components (four feature nodes and one action handler)
- Database tables
- Management screens

Location Module

The key component of the LCP is the Location Module. The calling application passes the MSISDN of a mobile device to the Location Module. The Location Module looks up the current location of the mobile device and returns its locational reference to the calling application as a Cell ID/ Area ID or as a circle of uncertainty.

Plug-Ins

The Location Module uses modular communication plug-ins to communicate with Location Servers on customer networks. It provides the following plug-ins:

- MAP ATI
- IS-41 POSREQ
- SRI-MSRN
- SRI-IMSI

Provisioning and Management

Additional SMS management screens are delivered as part of the LCP to allow the provisioning and management of locational entities in the database (for example: converting data from Cell ID to circles of uncertainty in X,Y,R format).

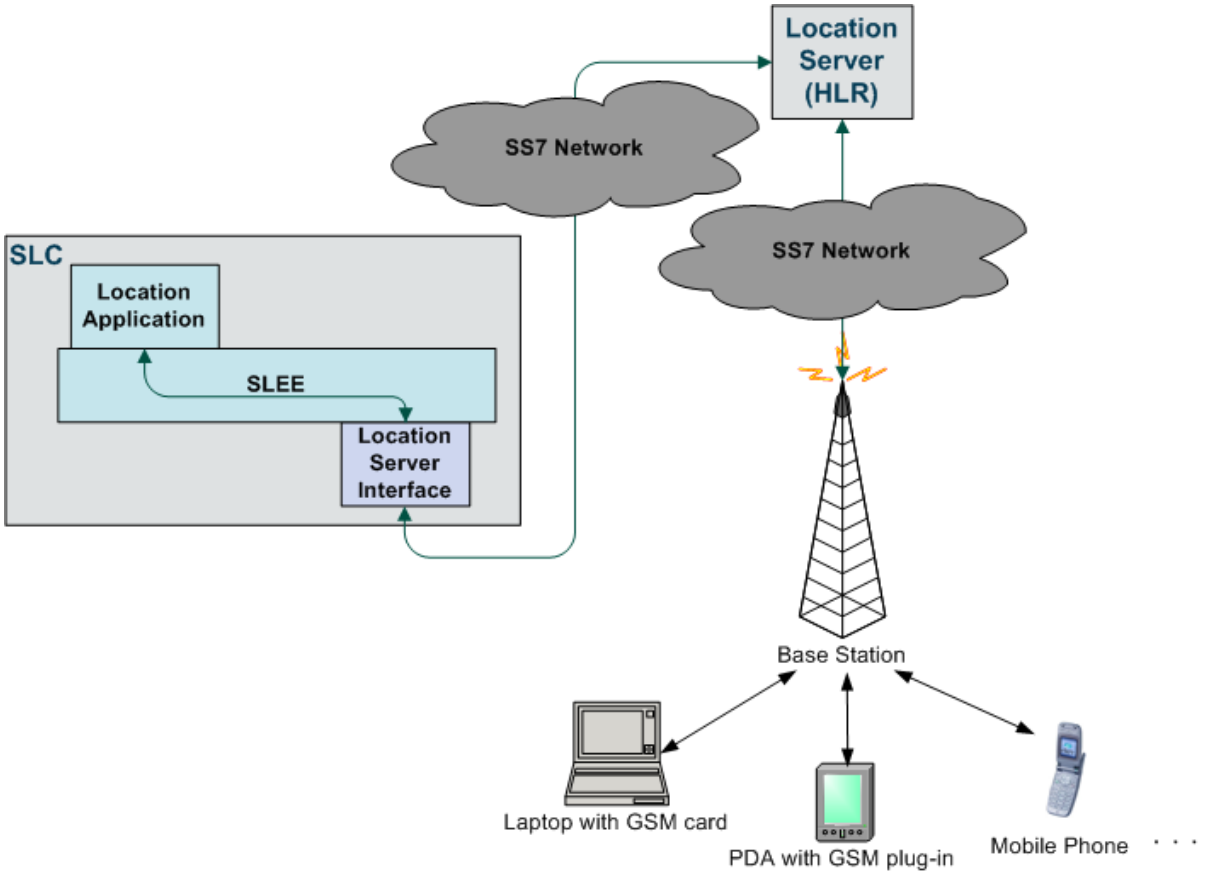
Feature nodes

The LCP also enables ACS-based customer applications to use the Location Module by providing the additional feature nodes:

- Set My Zone sets a subscriber's Home and Work zones.
- In The Zone checks if a mobile device is currently in a predefined zone.
- Store My Location stores locational data for use by other nodes.
- Store My Network ID stores the IMSI retrieved for a supplied MSISDN.

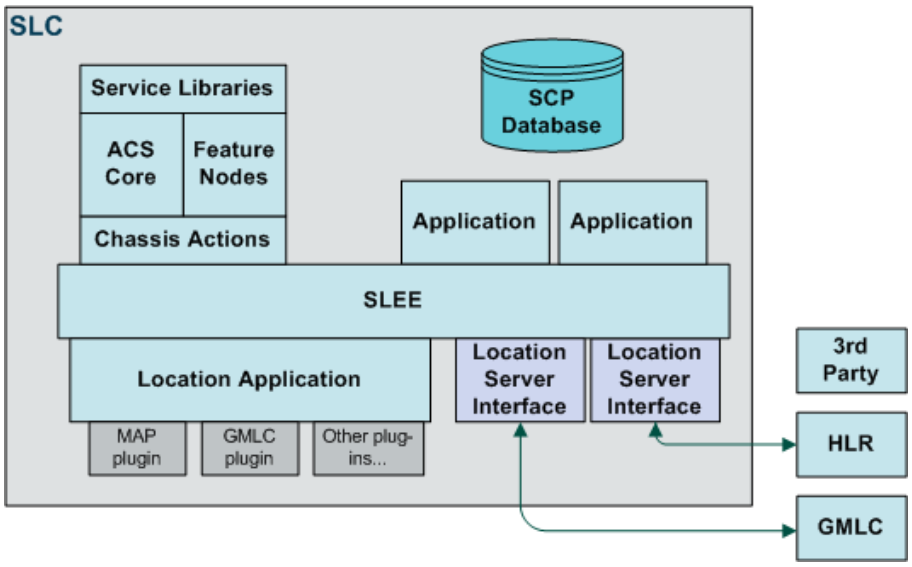
Overview Diagram

Here is an overview of how the LCP functions (interaction with the HLR). The Location Application sends location queries to the HLR.



Architectural overview

This diagram shows the architecture overview of the LCP. The Location Module is the locApp and is extendable by taking plug-ins to communicate with different Location Servers. Four feature nodes in ACS allow existing services to integrate with the LCP by dragging and dropping the nodes.



Normalization and Denormalization

Introduction

Normalization and Denormalization allow for incoming and outgoing numbers to be selected by their prefix and then have numbers stripped or added (as prefix) if necessary. Normalization can be applied to all numbers in an incoming IDP request or an MSISDN stored in a buffer. Denormalization can be applied to all numbers returned from the IN.

Denormalization rules

The LCP feature nodes will attempt to denormalize the MSISDN number stored in the **MSISDN Source to Query** profile buffer, using the standard ACS rules in the **acs.conf** file.

If no matching rule is found, the HLR query will be constructed using a default Nature of Address (NoA) value of 4.

Nature of address

The NoA (nature of address) is a classification to determine in what realm (Local, National or International) a given telephone number resides, for the purposes of routing and billing.

Details vary between different implementations of phone systems, but the following table is representative:

Dialed Digits	NOA (NOC, NON)	Definition
477 9425	1 → Subscriber	Number within Local Telephone Exchange
4 477 9425	3 → National	Number within Country Telephone Exchange
64 4 477 9425	4 → International	Number within World Telephone Exchange

Dialed Digits	NOA (NOC, NON)	Definition
477 9425	2 → UNKNOWN	Numbering Scheme rule → Subscriber
0 4 477 9425	2 → UNKNOWN	Numbering Scheme rule → National
00 64 4 477 9425	2 → UNKNOWN	Numbering Scheme rule → International

Statistics

Introduction

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

An existing statistics system (smsStats) provides functions for the collection of basic statistical events. This is provided in the Convergent Charging Controller SMS application. Refer to *Service Management System Technical Guide* for details.

Statistics gathered

The following statistics are gathered. Note that they all belong to the LCP application ID.

Statistic ID	Description
LCP_1	Total number of location requests made (and received by the locApp).
LCP_2	Number of requests via MAP ATI plug-in.
LCP_3	Number of cache hits.
LCP_4	Negative responses from Location Servers (such as HLR).
LCP_5	In The Zone macro node In-Zone hits.
LCP_6	In The Zone macro node Out-of-zone hits.
LCP_7	Set My Zone macro node “zone shape added”.
LCP_8	Set My Zone macro node “Too many zones”.
LCP_9	Location responses received (by the locApp) in <= 1 second.
LCP_10	Location responses received (by the locApp) in > 1 second and <= 2 seconds.
LCP_11	Location responses received (by the locApp) in > 2 seconds and <= 3 seconds.
LCP_12	Location responses received (by the locApp) in > 3 seconds and <= 4 seconds.
LCP_13	Location responses received (by the locApp) in > 4 seconds and <= 5 seconds.
LCP_14	Location responses received (by the locApp) in > 5 seconds.
LCP_15	Number of requests via IS41 POSREQ plug-in.
LCP_16	Number of locations stored using the ‘Store My Location’ node.
LCP_17	Number of requests set using the SRI plug-in

Chapter 2

Configuration

Overview

Introduction

This chapter explains how to configure the Oracle Communications Convergent Charging Controller application.

In this chapter

This chapter contains the following topics.

Configuration Overview

SLC configuration files

The following SLC configuration files are required for this product:

Configuration File	Description
eserv.config	The Oracle configuration file. All configuration for the LCP is under the section `LCP`.
SLEE.cfg	The SLEE configuration file. This is where to configure the SLEE to run locApp.
acs.conf	The ACS configuration file. This is where to configure ACS to load the LCP macro nodes and action handlers.

Note: The Location Capabilities Pack package installation will prompt for user input where required (including the configuration file values) and create usable start-up files and configuration files. The SLEE and ACS configuration file will also be updated automatically to include the LCP configurations.

For more details on configuring the SLEE and ACS, refer to *Service Logic Execution Environment Technical Guide* and *Advanced Control Services Technical Guide*.

Note: Configuration details are also held in the SMF database, and are configured using the SMS administration screens.

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.

Configuration File Format

To organize the configuration data within the **eserv.config** file, some sections are nested within other sections. Configuration details are opened and closed using either { } or [ ].

- Groups of parameters are enclosed with curly brackets – { }
- An array of parameters is enclosed in square brackets – [ ]
- Comments are prefaced with a # at the beginning of the line

To list things within a group or an array, elements must be separated by at least one comma or at least one line break. Any of the following formats can be used, as in this example:

```
    { name="route6", id = 3, prefixes = [ "00000148", "0000473"] }
    { name="route7", id = 4, prefixes = [ "000001049" ] }
```

or

```
    { name="route6"
        id = 3
        prefixes = [
            "00000148"
            "0000473"
        ]
    }
    { name="route7"
        id = 4
        prefixes = [
            "000001049"
        ]
    }
```

or

```
    { name="route6"
        id = 3
        prefixes = [ "00000148", "0000473" ]
    }
    { name="route7", id = 4
        prefixes = [ "000001049" ]
    }
```

eserv.config Files Delivered

Most applications come with an example **eserv.config** configuration in a file called **eserv.config.example** in the root of the application directory, for example, **/IN/service_packages/eserv.config.example**.

Editing the File

Open the configuration file on your system using a standard text editor. Do not use text editors, such as Microsoft Word, that attach control characters. These can be, for example, Microsoft DOS or Windows line termination characters (for example, ^M), which are not visible to the user, at the end of each row. This causes file errors when the application tries to read the configuration file.

Always keep a backup of your file before making any changes to it. This ensures you have a working copy to which you can return.

Example eserv.config file

Here is an example of the `LCP` section of the **eserv.config** file.

```
LCP = {
    oracleUserAndPassword=/
    timerIF=”Timer”

    actionHandlers = {
        locAppSK=15
        normalisedNumbers = false
        convertForHLR = false
    }

    atiPlugin = {
        performanceReportPeriod = 10
        timestampFormat = "%Y-%m-%d %T[usec:6]"
        generateEDR = true
        gsmScfAddress = "441234567890"
        gsmScfMapNOA = 1
        tcapIF = ”m3ua_if”
        origSSN = 0
        destSSN = 0
        cellIdPadCharacter = 'F'
        mapRequestTimeout = 15
    }

    posreqPlugin = {
        scfPC = 2443
        scfSSN = 6
        hlrSSN = 5
        mscMktId = 14656
        mscSwId = 1
        tcapIF = ”m3ua_if”
        forwardToMsc = true
    }
    atiMnpPlugin = {
        mnpDbAddress = "1,1,55555"
        gsmScfAddress = "1234"
        gsmScfMapNOA = 1
        tcapIF = "hlrIF"
        origSSN = 1234
        destSSN = 1235
    }

    sriPlugin = {
        gmscAddress = "441234567890"
        gmscMapNOA = 1
        tcapIF = "Tcap"
        origSSN = 1234
        destSSN = 1235
        destSccpNOA = 3
```

```
        mapNOA = 1
        mapVersion = 3
        hlrGt = "1,1,333"
    }

    currentLocation = {
        Map_Uncertainty_constant_C=10
        Map_Uncertainty_constant_x=0.1
        LocationNumberLength=15
    }

    locApp = {
        cacheSize = 500000
        responseDeadline = 2
        cacheExpiry = 600
        concatenateLocInfo = true
        flushPeriod = 13
        maxNum  = 10000
        destDir = "/IN/xyz-timestamp_abc/LCP/edr"
        tempDir = "/IN/xyz-timestamp_abc/LCP/tmp"
        filePrefix = ""
    }
}
```

Global parameters

Here are the global parameters for LCP. Since they are reusable across the all LCP components, they are in the top level LCP configuration section.

oracleUserAndPassword

Syntax: `oracleUserAndPassword = "user/pw"`
Description: Sets the Oracle login.
Type:
Optionality: Optional
Allowed:
Default: "/"
Notes:
Example: `oracleUserAndPassword = "/"`

timerIF

Syntax: `timerIF = "handle"`
Description: The SLEE timer interface handle.
Type: String
Optionality: Optional
Allowed:
Default: "Timer"
Notes:
Example:

actionHandlers

The `actionHandlers` section contains the LCP ACS action handler (*liblcpacschassis* (on page 32)) configuration.

locAppSK

Syntax: `locAppSK = skey`

Description: Specifies the service key of the location application.

Type: Integer

Optionality: Mandatory

Allowed:

Default: 15

Notes:

Example:

normalisedNumbers

Syntax: `normalisedNumbers = true|false`

Description: Whether to retrieve normalized or un-normalized called/calling numbers.

Type: Boolean

Optionality: Optional (default used if not set)

Allowed:

true	Use the normalized number from the ACS call context.
false	Do not use normalized numbers.

Default: false

Notes: Setting to true does not invoke the ACS denormalization rules.

Example:

convertForHLR

Syntax: `convertForHLR = true|false`

Description: If set to true, it will normalize the incoming IDP digits and then denormalize them.

Type: Boolean

Optionality: Optional

Allowed:

- true
- false

Default: false

Notes:

- If `convertForHLR` is set to true, special denormalization rules in the **acs.conf** file are used from the *LcpCustomNoA* (on page 26) section. If it is set to false, the global denormalization rules are used.
- If the `normalisedNumbers` parameter is set to false and `convertForHLR` is set to true, it will normalize the incoming number and then denormalize it. If both parameters are set to true, `convertForHLR` only denormalizes the number before sending to the HLR.

Example: `convertHLR = false`

Example

Here is an example `actionHandlers` configuration.

```
actionHandlers = {
    locAppSK=15
    normalisedNumbers = false
    convertForHLR = false
}
```

atiPlugin

The atiPlugin section contains the configuration for the locApp ATI plugin (*liblcpati* (on page 33)).

gsmScfAddress

Syntax: `gsmScfAddress = "scf_value"`
Description: Specifies what the GSM SCF Address should be set to for the ATI queries.
Type: String
Optionality: Required
Allowed:
Default: None
Notes:
Example: `gsmScfAddress = "12324"`

gsmScfMapNOA

Syntax: `gsmScfMapNOA = NOA_type_value`
Description:
Type: Integer
Optionality: Optional (default used if not set)
Allowed:
Default: 1
Notes:
Example: `gsmScfMapNOA = 1`

tcapIF

Syntax: `tcapIF = "if"`
Description: Specifies the Tcap Interface SLEE handle that the plug-in uses.
Type: String
Optionality: Required
Allowed:
Default: None
Notes:
Example: `tcapIF = "hlrIF"`

origSSN

Syntax: `origSSN = SSN`
Description: The originating SSN.
Type: Integer
Optionality: Optional (default used if not set)
Allowed:
Default: 0
Notes:
Example: `origSSN = 1234`

destSSN

Syntax: `destSSN = SSN`
Description: The destination SSN.

Type: Integer
Optionality: Optional (default used if not set)
Allowed:
Default: 0
Notes:
Example: `destSSN = 1234`

cellIdPadCharacter

Syntax: `cellIdPadCharacter = 'char'`
Description: The character to use to pad the values for the cell ID and LAI fields to the correct length.
Type: Character
Optionality: Optional (default used if not set)
Allowed:
Default: F
Notes:
Example: `cellIdPadCharacter = 'F'`

performanceReportPeriod

Syntax: `performanceReportPeriod = seconds`
Description: The period (in seconds) after which a new performance report will be generated.
Type: Integer
Optionality: Optional (default used if not set)
Allowed:
Default: `0`
Notes:

- The report must be consistently aligned with the system clock on the SLC. To achieve this, only an integer value that can be computed to be a factor of 60 or 3600, must be used.
- This period is aligned with system clock instead of the application starting time.
- No EDRs will be generated when set to default of `0` seconds.
- The first performance report after SLEE startup may differ to that which is configured for the performanceReportPeriod, depending on when the SLEE was started within the clock aligned performance period.

Example: `performanceReportPeriod = 3600`

timestampFormat

Syntax: `timestampFormat = "format"`
Description: Specifies the format of timestamps in the EDR.
Type: String
Optionality: Optional (default used if not set)
Allowed:
Default: `%Y-%m-%d %T[usec:6]`
Example timestamp output: `2013-08-25 22:12:56`

Notes:
- The strftime function will be used to format timestamps. Check the strftime man page to determine the supported format.
- In addition, microseconds can be logged in the EDR if "[usec:*XX*]" is included in the formatting string, where *XX* is an integer. Accepted values are from 1 to 6.
- Values that are less than 1 or greater than 6 will be set to 6 automatically.

Example: `timestampFormat = "%D %T"`

Example timestamp output: `02/08/13 22:12:56`

generateEDR

Syntax: `generateEDR = true|false`
Description: Determines whether an EDR record should be generated or not.
Type: Boolean
Optionality: Optional (default used if not set)
Allowed: true, false
Default: false
Notes:
Example: `generateEDR = false`

mapRequestTimeout

Syntax: `mapRequestTimeout = seconds`
Description: The number of seconds for the TC INVOKE timeout period for the MAP interrogation request that is sent to the Home Location Register (HLR).
Type: Integer
Optionality: Optional (default used if not set)
Allowed:
Default: 30 seconds
Notes:
Example: `mapRequestTimeout = 15`

Example

Here is an example `atiPlugin` configuration.

```
atiPlugin = {
    performanceReportPeriod = 10
    timestampFormat = "%Y-%m-%d %T[usec:6]"
    generateEDR = true
    gsmScfAddress = "441234567890"
    gsmScfMapNOA = 1
    tcapIF = ”m3ua_if”
    origSSN = 0
    destSSN = 0
    cellIdPadCharacter = 'F'
    mapRequestTimeout = 15
}
```

posreqPlugin

The posreqPlugin section contains the configuration for the locApp POSREQ plug-in (*liblcpposreq* (on page 35)).

scfPC

Syntax: `scfPC = addr`

Description: Specifies what the SCF SCCP addresses should be set to for the POSREQ queries.

Type: Integer

Optionality: Mandatory

Allowed:

Default:

Notes:

Example: `scfPC = 2443`

scfSSN

Syntax: `scfSSN = addr`

Description: Specifies what the SCF SCCP addresses should be set to for the POSREQ queries.

Type: Integer

Optionality: Mandatory

Allowed:

Default:

Notes:

Example: `scfSSN = 6`

hlrSSN

Syntax: `hlrSSN = addr`

Description: Specifies what the HLR SCCP addresses should be set to for the POSREQ queries.

Type: Integer

Optionality: Mandatory

Allowed:

Default:

Notes:

Example: `hlrSSN = 5`

mscMktId

Syntax: `mscMktld = id`

Description: MSC Market ID. Specifies the originating MSCID sent in the POSREQ query.

Type: Integer

Optionality: Mandatory

Allowed:

Default:

Notes:

Example: `mscMktld = 14656`

mscSwId

Syntax: `mscSwId = id`
Description: MSC Switch ID. Specifies the originating MSCID sent in the POSREQ query.
Type: Integer
Optionality: Mandatory
Allowed:
Default:
Notes:
Example: `mscSwId = 1`

tcapIF

Syntax: `tcapIF = "if"`
Description: Specifies the Tcap Interface SLEE handle that the plug-in uses.
Type: String
Optionality: Required
Allowed:
Default: None
Notes:
Example: `tcapIF = "hlrIF"`

forwardToMsc

Syntax: `forwardToMsc = true|false`
Description: Controls whether the POSREQ plug-in should:
- Allow the HLR to forward the request to the MSC (true)
- Send the request to the MSC (false)

Type: Boolean
Optionality: Optional
Allowed: true, false
Default: true
Notes:
Example: `forwardToMsc = true`

Example

Here is an example `posreqPlugin` configuration.

```
posreqPlugin = {
    scfPC = 2443
    scfSSN = 6
    hlrSSN = 5
    mscMktId = 14656
    mscSwId = 1
    tcapIF = ”m3ua_if”
    forwardToMsc = true
}
```

atimnpPlugin

The atimnpPlugin section contains the configuration for MNP USSD call-back and the ATI MNP plugin (*liblcpatimnp* (on page 34)).

mnpDbAddress

Syntax: `mnpDbAddress = "gt"`

Description: The GSM MNP DB global title, used as the destination GT in ATI messages.

Type: String

Optionality: Mandatory

Allowed:

Default: None

Notes: Values must be comma separated and in one of the following forms:

- Form a: "1,*noa*,*BCD_address_digits*"
- Form b: "2,*trans_type*,*BCD_address_digits*"
- Form c: "3,*trans_type*,*num_plan*,*BCD_address_digits*" (only BCD encoding scheme is supported)
- Form d: "4,*trans_type*,*num_plan*,*noa*,*BCD_address_digits*" (only BCD encoding scheme is supported)

Example: `mnpDbAddress = "1,1,55555"`

gsmScfAddress

Syntax: `gsmScfAddress = "scf_value"`

Description: Specifies what the GSM SCF Address should be set to for the ATI queries.

Type: String

Optionality: Required

Allowed:

Default: None

Notes:

Example: `gsmScfAddress = "12324"`

gsmScfMapNOA

Syntax: `gsmScfMapNOA = NOA_type_value`

Description:

Type: Integer

Optionality: Optional (default used if not set)

Allowed:

Default: 1

Notes:

Example: `gsmScfMapNOA = 1`

tcapIF

Syntax: `tcapIF = "if"`

Description: Specifies the Tcap Interface SLEE handle that the plug-in uses.

Type: String

Optionality: Required

Allowed:

Default: None

Notes:

Example: `tcapIF = "hlrIF"`

origSSN

Syntax: `origSSN = SSN`
Description: The originating SSN.
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: 147
Notes:
Example: `origSSN = 1234`

destSSN

Syntax: `destSSN = SSN`
Description: The destination SSN.
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: 6
Notes:
Example: `destSSN = 1235`

Example

Here is an example `atimnpPlugin` configuration.

```
atiMnpPlugin = {
    mnpDbAddress = "1,1,55555"
    gsmScfAddress = "1234"
    gsmScfMapNOA = 1
    tcapIF = "hlrIF"
    origSSN = 1234
    destSSN = 1235
}
```

sriPlugin

The `sriPlugin` section contains the configuration for the following locApp plug-ins:

- SRI-MSRN (*liblcpsrimsrn* (on page 36))
- SRI-IMSI (*liblcpsriimsi* (on page 35))

gmscAddress

Syntax: `gmscAddress = addr`
Description: Specifies the 'Gateway MSC Address' that is added to the SRI message sent by the plugin.
Type: Integer
Optionality: Mandatory
Allowed:
Default:
Notes: If you are using *liblcpsrimsrn* (on page 36), this parameter should be set to the GT of the SLC.

Example: `gmscAddress = 441234576890`

gmscMapNOA

Syntax: `gmscMapNOA = noa`
Description: .
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: 1
Notes:
Example: `gmscMapNOA = 1`

tcapIF

Syntax: `tcapIF = "if"`
Description: Specifies the Tcap Interface SLEE handle that the plug-in uses.
Type: String
Optionality: Required
Allowed:
Default: None
Notes:
Example: `tcapIF = "hlrIF"`

origSSN

Syntax: `origSSN = SSN`
Description: The originating SSN to add to the header of the SRI messages.
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: 0
Notes: Needed as LCP ignores the hss settings.
Example: `origSSN = 1234`

destSSN

Syntax: `DestSSN = SSN`
Description: The destination SSN to add to the header of the SRI messages.
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: 0
Notes: Adding the SSN to the header means it is not forced to 6 for the HLR.
Example: `destSSN = 1235`

destSccpNOA

Syntax: `destSccpNOA = noa`

Description: Override the NOA from SCCP in the IDP, and set this value instead on messages to locApp.

Type: Integer

Optionality: Optional (default used if not set)

Allowed:

Default: 4 for international, 3 otherwise

Notes:

Example: `destSccpNOA = 4`

mapNOA

Syntax: `mapNOA = noa`

Description: Override the NOA from MAP in the IDP, and set this value instead on messages to locApp.

Type: Integer

Optionality: Optional (default used if not set)

Allowed:

Default: 1 for international, 2 otherwise.

Notes:

Example: `mapNOA = 1`

mapVersion

Syntax: `mapVersion = version`

Description: The version of MAP to use in SRI messages.

Type: Integer

Optionality: Optional (default used if not set).

Allowed:

2	Use MAP 2.
3	Use MAP 3.

Default: 3

Notes:

Example: `mapVersion = 3`

hlrGt

Syntax: hlrGt = "*gt*"

Description: The HLR address.

Type: String

Optionality: Optional (default used if not set).

Allowed:

Default: Subscriber's MSISDN is used as the HLR address.

Notes: Values must be comma separated and in one of the following forms:

- Form a: "1,*noa*,*BCD_address_digits*"
- Form b: "2,*trans_type*,*BCD_address_digits*"
- Form c: "3,*trans_type*,*num_plan*,*BCD_address_digits*" (only BCD encoding scheme is supported)
- Form d: "4,*trans_type*,*num_plan*,*noa*,*BCD_address_digits*"

(only BCD encoding scheme is supported)

Example: `hlrGt = "1,1,333"`

Example

Here is an example `sriPlugin` configuration.

```
sriPlugin = {
    gmscAddress = "441234567890"
    gmscMapNOA = 1
    tcapIF = "Tcap"
    origSSN = 1234
    destSSN = 1235
    destSccpNOA = 3
    mapNOA = 1
    mapVersion = 3
    hlrGt = "1,1,333"
}
```

currentLocation

The `currentLocation` section contains the configuration for *liblcpCurrentLocProcessor* (on page 34) to process MapGeographicalInformation.

For more details on these, refer to GSM 03.32.

`Map_Uncertainty_constant_C`

Syntax: `Map_Uncertainty_constant_C = int`
Description: Used to process MapGeographicalInformation.
Type: Integer
Optionality: Optional
Allowed:
Default:
Notes:
Example: `Map_Uncertainty_constant_C = 10`

`Map_Uncertainty_constant_x`

Syntax: `Map_Uncertainty_constant_x = int`
Description: Used to process MapGeographicalInformation.
Type: Integer, floating-point
Optionality: Optional
Allowed:
Default:
Notes:
Example: `Map_Uncertainty_constant_x = 0.1`

`LocationNumberLength`

Syntax: `LocationNumberLength = len`
Description: This specifies the allowed length for an INAP Location Number. If the length does not match this value then the current location is not valid.
Type: Integer

Optionality: Optional
Allowed:
Default: 15
Notes:
Example: `LocationNumberLength = 15`

Example

Here is an example `currentLocation` configuration.

```
currentLocation = {
    Map_Uncertainty_constant_C=10
    Map_Uncertainty_constant_x=0.1
    LocationNumberLength=15
}
```

locAPP

The `locAPP` section contains the parameters controlling the *locApp* (on page 31).

cacheSize

Syntax: `cacheSize = size`
Description: Controls how many location responses the locApp caches.
Type: Integer
Optionality: Optional
Allowed:
Default:
Notes: The minimum cache space consumed is 50 MB (50*1024*1024 bytes).
If you are using the SRI plug-in (*liblcpsrimsrn* (on page 36)) you should always set this parameter to have no cache (0), as the MSRN is no longer valid after the call is connected.
Example: `cacheSize = 500000`

cacheExpiry

Syntax: `cacheExpiry = seconds`
Description: The number of seconds before a cached entry is considered “too old” and a new request must be made if the cached entry is queried.
Type: Integer
Optionality: Optional
Allowed:
Default:
Notes: If you are using the SRI plug-in (*liblcpsrimsrn* (on page 36)) you should always set this parameter to never cache (0), as the MSRN is no longer valid after the call is connected.
Example: `cacheExpiry = 600`

responseDeadline

Syntax: `responseDeadline = seconds`
Description: The number of seconds before the locApp considers a request as timed out.
Type: Integer

Optionality: Optional
Allowed:
Default: 2
Notes:
Example: `responseDeadline = 2`

concatenateLocInfo

Syntax: `concatenateLocInfo = true|false`
Description: Controls how the locApp will treat CELL_RAW type responses.
Type: Boolean
Optionality: Optional
Allowed: true, false
Default:
Notes: If the LocationNumber is present in the response, it will be pre-pended to the CellID before it is returned to the macro node.
Example: `concatenateLocInfo = true`

flushPeriod

Syntax: `flushPeriod = seconds`
Description: Determines the period (in seconds) after which a new EDR file will be created.

Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: `0`
Notes: No EDR files will be generated when set to default of `0` seconds.
Example: `flushPeriod = 60`

maxNum

Syntax: `maxNum = num`
Description: This is used to configure the maximum number of EDR records written in a single EDR file.
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: `10000`
Notes: A new EDR file will be generated according to the `flushPeriod` or `maxNum`, whichever target is reached first.
Example: `maxNum = 1000`

destDir

Syntax: `destDir = "dir"`
Description: The destination directory where the output EDR files are stored eventually.
Type: String
Optionality: Optional (default used if not set).

Allowed:

Default: "/IN/service_packages/LCP/edr"

Notes: If multiple locApp instances are running simultaneously, the `destDir` holding the final EDR files can be the same for all instances.

Example: `destDir = "/IN/xyz-timestamp_abc/LCP/edr"`

`tempDir`

Syntax: `tempDir = "dir"`

Description: The directory where the output EDR files are stored temporarily.

Type: String

Optionality: Optional (default used if not set).

Allowed:

Default: "/tmp"

Notes: EDR files leftover from previous failed/aborted instances of `locApp` can also be recovered from this directory.

Example: `tempDir = "/IN/xyz-timestamp_abc/LCP/tmp"`

`filePrefix`

Syntax: `filePrefix = "pref"`

Description: This prefix is used to specify the prefix of the name of an EDR file. The name of an EDR file consists of `filePrefix`, `PID`, and `timestamp`.

Type: String

Optionality: Optional (default used if not set).

Allowed:

Default: `""`

Notes:

Example: `filePrefix = "abc"`

Example

Here is an example of the `locAPP` configuration.

```
locApp = {
    cacheSize = 500000
    responseDeadline = 2
    cacheExpiry = 600
    concatenateLocInfo = true
    flushPeriod = 13
    maxNum  = 10000
    destDir = "/IN/xyz-timestamp_abc/LCP/edr"
    tempDir = "/IN/xyz-timestamp_abc/LCP/tmp"
    filePrefix = ""
}
```

Loading eserv.config Changes

If you change the configuration file, you must restart the appropriate parts of the service to enable the new options to take effect.

Configuring the acs.conf on the SCP

Introduction

The **acs.conf** file must be configured to enable LCP to use custom denormalization rules. All necessary configuration is done at installation time by the configuration script; this section is for information only.

The ACS configuration file is located at **/IN/service_packages/ACS/etc/acs.conf**.

See *Advanced Control Services Technical Guide* for details on ACS configuration.

Note: It is not recommended to change the values of this section. Please contact the Oracle support prior to attempting any modification to configuration data.

Example LCP section

Here's an example of the LCP configuration in the **acs.conf** file.

```
acsChassis

#  Checking for LCP denormalization rules

 useCustomLCPNotificationNoARules 1:

  LcpCustomNoA
   DenormalisationRule (000620,4,3,E,7)
   DenormalisationRule (000,4,3,E)
   DenormalisationRule (1077,2,0,0)
   DenormalisationRule (1055,2,0,0)
   DenormalisationRule (0,3,1,E,2)
   DenormalisationRule (495,2,0,E):
```

Parameters

Here are the parameters required for configuring LCP in the **acs.conf**.

`useCustomLCPNotificationNoARules`

Syntax: `useCustomLCPNotificationNoARules` *`int`*

Description: Determines whether the incoming numbers are denormalized using global denormalization rules or LCP-specific rules defined in the `LcpCustomNoA` section.

Type:

Optionality: Optional (default used if not set).

Allowed:

- 0 – Global denormalization rules
- 1 – LCP-specific rules

Default: 0

Notes: The `convertForHLR` parameter in the **eserv.config** file must be enabled for this to take effect.

Example: `useCustomLCPNotificationNoARules 1`

LcpCustomNoA

Syntax:

```
LcpCustomNoA

 DenormalisationRule 1

 ...

 DenormalisationRule n:
```

Description: Defines the custom denormalization rules available to LCP to normalize or denormalize the incoming IDP digits.

Type: Array

Optionality: Optional

Allowed:

Default:

Notes:

Example:

```
LcpCustomNoA
 DenormalisationRule (000600,1,2,E,3)
 DenormalisationRule (495,1,0,E):
```

Chapter 3

Administrative Tasks

Overview

Introduction

This chapter provides the procedures for administering the LCP application.

In this chapter

This chapter contains the following topics.

Stopping and Starting

The locApp

The locApp is a SLEE application. Therefore, it is started along with the rest of the SLEE applications and interfaces.

After installation of the package lcpScp, the SLEE start-up and configuration files (as specified at install time) will be updated to instruct the SLEE to start the locApp.

For more details on how to start the SLEE, see *Service Logic Execution Environment Technical Guide*.

The LCP ACS components

The LCP ACS components are simply shared libraries (ACS plug-in), so they are started along with slee_acs. The slee_acs startup and configuration files (as specified at lcpScp install time) are updated by the lcpScp package install scripts to allow the LCP ACS components to be integrated into ACS.

For more information about configuring slee_acs (and the macro node/action plug-in), see *Advanced Control Services Technical Guide*.

Turning DEBUG on and off

Follow these steps to turn DEBUG on and off.

Step	Action
1	Set the UNIX environment variable DEBUG to `LCP` or `all`, for example: `DEBUG=LCP; export DEBUG` **Note:** Specifying the all DEBUG section will display all sections of debug. This is the most verbose debug level, so using the LCP section is recommended when specifically checking the LCP service. To turn off debug, the DEBUG environment variable must be unset.

Step	Action
2	Restart the SLEE to make the new setting effective.

Data Encoding

ATI Cell ID/LAI encoding

According to 3GPP TS 04.08, the coding of Cell ID and LAI (without taking the country code and network code into account) are specified per local administration. As such, the Cell ID and LAI are treated as numbers. Note that if the number of digits received is less than 4, then the Cell ID or LAI will be left padded with 'F'.

Examples:

- Cell ID/LAI received is 1234: No padding is done
- Cell ID/LAI received is 12: Cell ID/LAI is padded to FF12
- Cell ID/LAI received is 12345: Cell ID/LAI is unchanged.

IS41 MSCID (Location Number) & serving Cell ID encoding

The U-CA-IS41 (cdmagw) encodes the Serving Cell ID and MSCID (Location Number) using the following encoding scheme. This has been arbitrarily created to suit the processing of such data and does not follow any set standards other than those defined by Oracle.

Depending on the *LCP.locApp.concatenateLocInfo parameter* (on page 23), the data returned from the locApp will contain one of the following encoding schemes:

The returned location data will be concatenated using the following format:

- Location Number – 9 digits (zero padded)
- Serving Cell ID – 6 digits (zero padded)

If the configuration item specifies not to concatenate (or the response does not contain the Location Number), the following format will be used:

- Serving Cell ID – 6 digits (zero padded)

Example:

- Location Number & Serving Cell ID: 014566114000300
- Serving Cell ID: 000401

SRI-MSRN encoding

No encoding of the MSRN (Mobile Station Roaming Number) is performed.

The number is an E.164 and is handled in the form that is received by the SRI plug-in.

Example:

MSRN - (16 digits max) - 00441473289900

SRI-IMSI encoding

No encoding of the IMSI (International Mobile Subscriber Identifier) is performed.

Conversions

Conversion bulk loader

The Conversion Bulk Loader enables you to load conversion data from text files. Files can contain data for updating by area ID or by cell ID, but not both. The table will allow a maximum of 1 million records.

Import file format

The conversion import files are text files with one entry per line. Whitespace (space or tab) is allowed between fields.

There are two formats possible on each line:

1. *id*,*X*,*Y*,*R*
   Insert or update an entry with this ID. If the ID already exists, it is updated with the new *X*, *Y,* and *R* values.
2. *id*
   Delete the entry with this ID.

Bulk importing using command line

The bulk loader is started by hand by the "smf_oper" user as follows:

```
lcpConversionLoader {-m M} [-c cellFile] [-a areaFile] [-o outputFile] [-u
user/password]
```

- "-m" (mode) must be specified. M =
  - "R" = replacement (insert into empty database or replace all existing rows),
  - "U" = update (update existing data).
- One or both of the "-c" and "-a" options must be specified. The "-m" mode applies to both files if both files are specified.
- If the "-o" option is not specified, output is to the standard output or standard error, whichever is appropriate.
- The "-u" option specifies the Oracle login details. This defaults to "/".

Bulk importing using screens

For details on how to bulk import conversions using the LCP maintenance screens, refer to the *Location Capabilities Pack User's Guide*.

Chapter 4

Background Processes

Overview

Introduction

This chapter explains the processes that are started automatically by Service Logic Execution Environment (SLEE).

Note: This chapter also includes some plug-ins to background processes which do not run independently.

In this chapter

This chapter contains the following topics.

locApp

Purpose

locApp is the main LCP process. In addition to its own processing, it supports plug-ins. The plug-in is chosen by the applications using LCP or the default configuration is used.

Location

This binary is located on SLCs.

Configuration

locApp accepts the following parameters.

Usage:

```
locApp [-u usr/pwd] [-t name] [-s size] [-e seconds] [-c ] [-f seconds]
```

The available parameters are:

Parameter	Default	Description
-u *usr*/*pwd*	/	Oracle username and password.
-t *name*	Timer	Name of the SLEE timer interface handle.
-s *size*		Cache size. Controls how many location responses locApp caches. The minimum cache space consumed is 50 MB (50*1024*1024 bytes). If you are using the SRI plug-in, set this parameter to **0** (no cache) because the MSRN is no longer valid after the call is connected.
-e *seconds*		Cache expiry timer. The number of seconds before a cached entry is considered too old. If the entry is queried again, a new request must be made.
-c	Disabled	Concatenate. When enabled, locApp prepends the Location Number returned in the response to the Cell ID.
-f *seconds*	0	Flush period. Determines the amount of time, in seconds, after which a new EDR file is created. • 0 – No EDR files are generated. • Positive integer – An EDR file is generated after the specified number of seconds.

The command-line parameters override the values set in the **eserv.config** file. For more information about configuring locApp through **eserv.config**, see *locAPP* (on page 22).

Startup

locApp is a SLEE application and is started during SLEE initialization. The lines in the **SLEE.cfg** that start the locApp are:

```
APPLICATION=locApp locApp.sh /IN/service_packages/LCP/bin 1 1
SERVICE=locApp 1 locApp locApp
SERVICEKEY=INTEGER 123 locApp
```

Note: The above settings are defaults and may vary.

For instructions about starting and stopping locApp, see *Service Logic Execution Environment Technical Guide*.

javaLcpConversionLoader.sh

Purpose

javaLcpConversionLoader.sh is the shell script the LCP screens use to run the *Conversions* (on page 29).

Location

This binary is located on the SMS node.

liblcpacschassis

Purpose

This slee_acs plug-in implements the chassis actions which are used by the LCP macro nodes when they need to interact with elements outside the control plan.

Location

This library is located on SLCs.

Startup

If liblcpacschassis is configured in **acs.conf**, it is made available to slee_acs when slee_acs is initialized. It is included in the `acsChassis` section of `acs.conf` in a `ChassisPlugin` entry.

```
acsChassis
 ChassisPlugin liblcpacschassis.so
```

Configuration

liblcpacschassis is configured in **eserv.config**. For more information about how to configure this library, see *actionHandlers* (on page 10).

liblcpalarms

Purpose

liblcpalarms provides the alarms definitions.

Location

This binary is located on both SLCs and SMSs.

Configuration

This binary has no specific configuration.

liblcpati

Purpose

liblcpati supports the Any Time Interrogation message on GSM networks. The ATI message is used to determine the location of a mobile subscriber.

Location

This library is located on SLCs.

Configuration

liblcpati is configured in **eserv.config**. For more information about how to configure this library, see *atiPlugin* (on page 12).

liblcpatimnp

Purpose

This is ATIMNP plug-in, which sends ATIs to a MNP DB (whose GT must be specified in **eserv.config** at *atimnpPlugin* (on page 16)) to retrieve the MSISDN and routing number of the called party.

Location

This library is located on SLCs.

Configuration

liblcpatimnp is configured in **eserv.config**. For more information about how to configure this library, see *atimnpPlugin* (on page 16).

liblcpCurrentLocProcessor

Purpose

liblcpCurrentLocProcessor processes MapGeographicalInformation.

For more details on these, refer to GSM 03.32.

Location

This library is located on SLCs.

Configuration

liblcpCurrentLocProcessor is configured in **eserv.config**. For more information about how to configure this library, see *currentLocation* (on page 21).

liblcpmacronodes

Purpose

This slee_acs plug-in provides the LCP macro nodes. There are no configuration file settings for these macro nodes, they are all configured in the Control Plan Editor node configuration screens.

For more information about:

- Macro node libraries, see *Advanced Control Services Technical Guide*
- CPE, see *Control Plan Editor User's Guide*

Location

This library is located on SLCs.

Startup

If liblcpmacronodes is configured in **acs.conf**, it is made available to slee_acs when slee_acs is initialized. It is included in the `acsChassis` section of **acs.conf** in a `MacroNodePluginFile` entry as follows:

```
acsChassis
 MacroNodePluginFile liblcpmacronodes.so
```

Configuration

This binary has no specific configuration.

liblcpposreq

Purpose

The POSREQ plug-in supports the PositionRequest message.

Location

This library is located on SLCs.

Configuration

liblcpposreq is configured in **eserv.config**. For more information about how to configure this library, see *posreqPlugin* (on page 14).

liblcpsri

Purpose

The SRI plug-in supports the Send Routing Information message.

Location

This library is located on SLCs.

Configuration

liblcpsri is configured in **eserv.config**. For more information about how to configure this library, see *sriPlugin* (on page 18).

liblcpsriimsi

Purpose

The SRI plug-n supports the Send Routing Information message and records the resulting IMSI. The plug-in extracts digits from the MSISDN parameter of the ATI Responses.

Location

This library is located on SLCs.

Configuration

liblcpsriimsi is configured in **eserv.config**. For more information about how to configure this library, see *sriPlugin* (on page 18).

liblcpsrimsrn

Purpose

The SRI plug-in supports the Send Routing Information message and records the resulting MSRN.

Location

This library is located on SLCs.

Configuration

liblcpsrimsrn is configured in **eserv.config**. For more information about how to configure this library, see *sriPlugin* (on page 18).

Chapter 5

About Installation and Removal

Overview

Introduction

This chapter provides information about the installed components for the Oracle Communications Convergent Charging Controller application described in this guide. It also lists the files installed by the application that you can check for, to ensure that the application installed successfully.

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

LCP packages

An installation of Location Capabilities Pack includes the following packages, on the:

- SMS:
  - lcpSms
- SLC:
  - lcpScp

Checking the Installation

Checklist

After the installation is complete, refer to this checklist to ensure that installation has been successful.

Step	Action	Check
1	On the SMS, check that the following tables are available: • LCP_CONVERSION • LCP_MAPPING • LCP_PLUGIN	

2 On the SMS, check that replication groups are available for the above tables from the **Replication** tab in the SMS user screens and a configuration file can be generated.

3 On the SLC, check that the following tables are available:
- LCP_CONVERSION
- LCP_MAPPING
- LCP_PLUGIN

4 On the SLC, check that the following has been inserted into **SLEE.cfg** (note that the parameters will vary depending on the installation):
```
APPLICATION=locApp locApp.sh /IN/service_packages/LCP/bin 1 1
SERVICE=locApp 1 locApp locApp
SERVICEKEY=INTEGER 13 locApp
```

5 Check that the following line is in **acs.conf** under the `acsChassis` section (note the two spaces at the beginning of the line):
```
  MacroNodePluginFile liblcpmacronodes.so
```

Location Application semaphore

As the locApp starts, it creates a file, **/IN/service_packages/LCP/tmp/lcp-semkey,** containing the semaphore key that it uses. This semaphore is removed before the SLEE starts up (the command to remove the semaphore is inserted to the SLEE startup script and run just before the binary sleeStartup is invoked).

If, for whatever reason, the semaphore is not removed and the locApp is not starting up correctly, examine content of the file and remove the semaphore by using the Unix program ipcs.

For more details on ipcs, see man -s1 ipcs.

Shared libraries

All the LCP shared libraries reside in **/IN/service_packages/LCP/lib**. They are described below.

Name	Description
liblcpati.so	The locApp ATI plug-in.
liblcpposreq.so	The locApp POSREQ plug-in.
liblcpsrimsrn.so	The locApp SRI-MSRN plug-in.
liblcpsriimsi.so	The locApp SRI-IMSI plug-in.
liblcpalarms.so	The LCP alarms library. This library contains all the alarm texts.
liblcpacschassis.so	The LCP ACS chassis action handler. Responsible for sending location queries/processing location responses from the locApp. It is also responsible for retrieving the calling party and called party number and NoA from the ACS Chassis context (as these values are not available at the Engine level).
liblcpCurrentLocProcessor.so	The library contains information on how to process the current location. Currently it understands how to process Cell ID, LAI, LocationNumber and MapGeographicInformation as the current location. GSM 24.008 states that the coding of the Cell ID and LAI is the responsibility of the administration. Currently the Current Location Processor only supports Cell ID and LAI coded as BCD digits.
liblcpmacronodes.so	The LCP ACS macro nodes. The library contains the feature nodes: • In The Zone • Set My Zone

Name	Description
	• Store My Location

Glossary of Terms

ACS

Advanced Control Services configuration platform.

ATI

Any Time Interrogation - this process is used on a GSM network to interrogate the HLR for location and or subscriber information.

CC

Country Code. Prefix identifying the country for a numeric international address.

Connection

Transport level link between two peers, providing for multiple sessions.

Convergent

Also “convergent billing”. Describes the scenario where post-paid and pre-paid calls are handed by the same service platform and the same billing system. Under strict converged billing, post-paid subscribers are essentially treated as “limited credit pre-paid”.

CPE

Control Plan Editor (previously Call Plan Editor) - software used to define the logic and data associated with a call -for example, "if the subscriber calls 0800 *nnnnnn* from a phone at location *xxx* then put the call through to *bb bbb bbbb*".

DB

Database

DP

Detection Point

FDA

First Delivery Attempt - the delivery of a short message directly to the SME rather than relaying it through the MC.

GPRS

General Packet Radio Service - employed to connect mobile cellular users to PDN (Public Data Network- for example the Internet).

GSM

Global System for Mobile communication.

It is a second generation cellular telecommunication system. Unlike first generation systems, GSM is digital and thus introduced greater enhancements such as security, capacity, quality and the ability to support integrated services.

GT

Global Title.

The GT may be defined in any of the following formats:

- Type 1: String in the form "1,<noa>,<BCD address digits>"
- Type 2: String in the form "2,<trans type><BCD address digits>"
- Type 3: String in the form "3,<trans type>,<num plan>,<BCD address digits>"
- Type 4: String in the form "4,<trans type>,<num plan>,<noa>,<BCD address digits>"

The contents of the Global Title are defined in the Q713 specification, please refer to section 3.4.2.3 for further details on defining Global Title.

HLR

The Home Location Register is a database within the HPLMN (Home Public Land Mobile Network). It provides routing information for MT calls and SMS. It is also responsible for the maintenance of user subscription information. This is distributed to the relevant VLR, or SGSN (Serving GPRS Support Node) through the attach process and mobility management procedures such as Location Area and Routing Area updates.

HPLMN

Home PLMN

IDP

INAP message: Initial DP (Initial Detection Point)

IMSI

International Mobile Subscriber Identifier. A unique identifier allocated to each mobile subscriber in a GSM and UMTS network. It consists of a MCC (Mobile Country Code), a MNC (Mobile Network Code) and a MSIN (Mobile Station Identification Number).

The IMSI is returned by the HLR query (SRI-SM) when doing FDA. This tells the MSC exactly who the subscriber is that the message is to be sent to.

IN

Intelligent Network

INAP

Intelligent Network Application Part - a protocol offering real time communication between IN elements.

Initial DP

Initial Detection Point - INAP Operation. This is the operation that is sent when the switch reaches a trigger detection point.

IS-41

Interim Standard 41 is a signaling protocol used in cellular telecommunications systems. It deals with the signalling between the MSC and other network elements for the purpose of handovers and roaming etc.

ISDN

Integrated Services Digital Network - set of protocols for connecting ISDN stations.

ISUP

ISDN User Part - part of the SS7 protocol layer and used in the setting up, management, and release of trunks that carry voice and data between calling and called parties.

ITU

International Telecommunication Union

LCP

Location Capabilities Pack - set of software components used by other applications to look up the location of mobile devices.

MAP

Mobile Application Part - a protocol which enables real time communication between nodes in a mobile cellular network. A typical usage of the protocol would be for the transfer of location information from the VLR to the HLR.

MC

Message Centre. Also known as SMSC.

MCC

Mobile Country Code. In the location information context, this is padded to three digits with leading zeros. Refer to ITU E.212 ("Land Mobile Numbering Plan") documentation for a list of codes.

MNC

Mobile Network Code. The part of an international address following the mobile country code (MCC), or at the start of a national format address. This specifies the mobile network code, that is, the operator owning the address. In the location information context, this is padded to two digits with a leading zero. Refer to ITU E.212 ("Land Mobile Numbering Plan") documentation for a list of codes.

MNP

Mobile Number Portability

MSC

Mobile Switching Centre. Also known as a switch.

MSIN

Mobile Station Identification Number.

MSISDN

Mobile Station ISDN number. Uniquely defines the mobile station as an ISDN terminal. It consists of three parts; the country code (CC), the national destination code (NDC) and the subscriber number (SN).

MSRN

Mobile Station Roaming Number

MT

Mobile Terminated

MTP

Message Transfer Part (part of the SS7 protocol stack).

NOA

Nature Of Address - a classification to determine in what realm (Local, National or International) a given phone number resides, for the purposes of routing and billing.

PLMN

Public Land Mobile Network

POSREQ

IS-41 Position Request - process used in the IS-41 network to interrogate the HLR for location and subscriber information.

SCCP

Signalling Connection Control Part (part of the SS7 protocol stack).

SCF

Service Control Function - this is the application of service logic to control functional entities in providing Intelligent Network services.

SGSN

Serving GPRS Support Node

SLC

Service Logic Controller (formerly UAS).

SLEE

Service Logic Execution Environment

SME

Short Message Entity - This is an entity which may send or receive short messages. It may be located in a fixed network, a mobile, or an SMSC.

SMS

Depending on context, can be:

- Service Management System hardware platform

- Short Message Service
- Service Management System platform
- Convergent Charging Controller Service Management System application

SN

Service Number

SRI

Send Routing Information - This process is used on a GSM network to interrogate the HLR for subscriber routing information.

SS7

A Common Channel Signalling system is used in many modern telecoms networks that provides a suite of protocols which enables circuit and non-circuit related information to be routed about and between networks. The main protocols include MTP, SCCP and ISUP.

SSN

Subsystem Number. An integer identifying applications on the SCCP layer.

For values, refer to *3GPP TS 23.003*.

USSD

Unstructured Supplementary Service Data - a feature in the GSM MAP protocol that can be used to provide subscriber functions such as Balance Query.

VLR

Visitor Location Register - contains all subscriber data required for call handling and mobility management for mobile subscribers currently located in the area controlled by the VLR.

VWS

Oracle Voucher and Wallet Server (formerly UBE).

Index

M

N

O

P

R

S

T

U

V

W