

LXR.CCN

**Multi-application Connector
for CCN/CSI**

A contribution to the
ELLiXXiR™
e-Customs Community Initiative

DEVELOPER'S GUIDE



** Limited. Please do not disclose without prior consent of Vivansa. **

This document provides description of the interfaces exposed by the LXR.CCN connector.

Keywords

eCustoms, Service Oriented Architecture (SOA), Multi-Annual Strategic Plan (MASP), CCN, CCNC SI, CSI

Classification

*** Limited ***

Document history

<u>Version</u>	<u>Release date</u>	<u>Description</u>
0.01	01-Dec-2008	Preliminary version
0.10	12-Apr-2009	Added QoS capabilities
1.00	17-Aug-2009	Aligned to LXR.CCN Enterprise Edition, v2.0

Reviewed by

<u>Version</u>	<u>Release date</u>	<u>Name</u>	<u>Description</u>
0.01	8-Dec-2008	Said Eloudrhiri	Quality checked
0.10	16-Apr-2009	Said Eloudrhiri	Quality checked
1.00	18-Aug-2009	Said Eloudrhiri	Quality checked

Approved by

<u>Version</u>	<u>Release date</u>	<u>Name</u>	<u>Description</u>
1.00	20-Aug-2009	Vivien Monti	

** Limited. Please do not disclose without prior consent of Vivansa. **

Table of contents

1	INTRODUCTION	4
2	TARGET AUDIENCE.....	5
3	CONVENTIONS	6
3.1	NOTICE.....	6
4	WEB SERVICE INTERFACES	7
4.1	WSDL LOCATION	7
4.2	WEB SERVICE STANDARDS	8
4.3	SUBMISSION INTERFACE	8
4.4	CALLBACK INTERFACE	10

** Limited. Please do not disclose without prior consent of Vivansa. **

1 Introduction

The purpose of this document is to provide the user of LXR.CCN with the information needed to send and receive messages to/from the connector.

This developer's guide is applicable to the following licensed versions of LXR.CCN:

- LXR.CCN Core, Enterprise Edition, v2.0
- LXR.CCN Advanced, Enterprise Edition, v2.0

** Limited. Please do not disclose without prior consent of Vivansa. **

2 Target audience

This document addresses to developers and integrators of applications interacting with the LXR.CCN connector.

** Limited. Please do not disclose without prior consent of Vivansa. **

3 Conventions

This section provides the naming and graphic conventions, the directory paths and the definition of the key words used in this document.

3.1 Notice



Title of the warning or the error

This is a warning or an error message.



Title of the note

This is a note.

** Limited. Please do not disclose without prior consent of Vivansa. **

4 Web Service interfaces

This section provides a description of the Web Service interfaces provided by the LXR.CCN Connector to send and to receive messages:

- [Submission interface](#)
- [Callback](#)

4.1 WSDL location

In the source code, the WSDL are located in ServiceContracts/lxr/v10/. They are as follows:

CcnCsiAdapterSubmission-UsernameToken.wsdl	Interface to submit messages to LXR.CCN, with usernameToken/passwordDigest security requirement.
CcnCsiAdapterSubmission.wsdl	Interface to submit messages to LXR.CCN, no security requirement.
CcnCsiAdapterCallback-UsernameToken.wsdl	Interface that applications need to implement to be able to receive incoming messages, with usernameToken/passwordDigest security requirement.
CcnCsiAdapterCallback.wsdl	Interface that applications need to implement to be able to receive incoming messages, no security requirement.
CcnCsiAdapterAdministration-UsernameToken.wsdl	Management interface of the LXR.CCN backend, with usernameToken/passwordDigest security requirement.
CcnCsiAdapterAdministration.wsdl	Management interface of the LXR.CCN backend, no security requirement.
CcnCsiAdapterAdminDispatcher-UsernameToken.wsdl	Unified Administration interface for all LXR.CCN backend instances, with usernameToken/passwordDigest security requirement.
CcnCsiAdapterAdminDispatcher.wsdl	Unified Administration interface for all LXR.CCN backend instances, no security requirement.
BackendCalls.wsdl GatewayRouter.wsdl SplitJoins.wsdl	Used internally by the Admin Dispatcher.
lxr-common-types.xsd	Element types common to all Ellixxir

** Limited. Please do not disclose without prior consent of Vivansa. **

	products.
lxrcen-types.xsd	Element types specific to LXR.CCN but shared by the various WSDL interfaces.

4.2 Web service standards

All WSDL interfaces from the LXR.CCN are defined as SOAP 1.2. It is not possible to address these services directly in SOAP 1.1 because the WSDL namespaces are different.



Support of SOAP 1.1

If SOAP 1.1 access is necessary, one can use a wrapper, for example in ALSB, to present a SOAP 1.1 façade.

The services and the provided java clients are implemented with Oracle WLS 10 JAX-RPC 1.1 including WS-Security. The WSDL with security that are provided use the usernameToken method with passwordDigest. Note that the passwordDigest is not available on the default domain setups of WLS 10. This activation is described in the Installation Guide.

4.3 Submission interface

The Submission interface provides the operation implemented by the LXR.CCN Connector to send messages to the CCN/CSI Gateway.

4.3.1 Send

Here below, we provide a sample SOAP request used to send a message:

```
<v10:SendRequest>
  <Partner>FR</Partner>
  <MessageDescriptor>
    <!--Optional:-->
    <MessageId>BE054963349759676</MessageId>
    <!--Optional:-->
    <CorrelationId>08BE03456234568934856</CorrelationId>
    <MessageType>CD801A-MSG.emcs</MessageType>
    <!--Optional:-->
    <ExchangeType>REQUEST</ExchangeType>
  </MessageDescriptor>
  <!--Optional:-->
  <DataDescriptor>
    <Data>VGvzdCBtZXNzYWdl</Data>
  </DataDescriptor>
```


** Limited. Please do not disclose without prior consent of Vivansa. **

```
<QualityOfService>
  <!--Optional:-->
  <Priority>1</Priority>
  <!--Optional:-->
  <Integrity>true</Integrity>
  <!--Optional:-->
  <Confidentiality>true</Confidentiality>
  <!--Optional:-->
  <Compression>true</Compression>
  <!--Optional:-->
  <CompressionId>LZW</CompressionId>
</QualityOfService>
</v10:SendRequest>
```

4.3.1.1 Parameters

- **Partner:** Country code of the country to which the message should be sent.
- **MessageID:** ID of the message, if generated by the application.
- **CorrelationId:** Logical ID linking several messages into a single conversation.
- **MessageType:** CCN-defined message type. This should correspond with a message type define in the LXR.CCN Connector configuration.
- **Data:** Actual content of the message, encoded in base64 to allow binary data.
- **QualityOfService:** This overrides the CCN/CSI default QoS:
 - **Priority:** lower numbers will be processed first in the destination queue.
 - **Integrity:** Check the integrity of the messages
 - **Confidentiality:** Encrypt the communication with the CCN/CSI gateway.
 - **Compression:** Whether the communication with the CCN/CSI gateway is compressed.
 - **CompressionId:** Compression algorithm to use. Only valid if Compression is true. The only compression algorithm currently supported is LZW.

4.3.1.2 Business rules

- **MessageId:** optional
 - If none is provided, a unique ID will be generated by the LXR.CCN Connector for this outgoing message.
 - If it is present in the request, it must be unique, i.e. no message must exist yet for this ID in the LXR.CCN Connector outgoing message database table.
 - Maximum 24 characters long.
- **CorrelationId:** mandatory, maximum 24 characters long
- **Partner:** mandatory and must be a valid ISO country code (2 characters)

** Limited. Please do not disclose without prior consent of Vivansa. **

- `MessageType`: mandatory, maximum 32 characters long
- `Data`: mandatory
- There must exist a entry matching this `Partner` and this `MessageType` in the LXR.CCN Connector configuration (see the administration guide)

4.3.1.3 Response

Here below, we provide a sample of response.

```
<v10:SendResponse xmlns:v10="http://www.ellixir.org/lxr/ccn/submission/wsd/v10">
  <MessageDescriptor>
    <MessageId>2df4820133154b3eb3fef9e</MessageId>
    <MessageType>CD801A-MSG.emcs</MessageType>
    <applicationType>EMCS</applicationType>
  </MessageDescriptor>
</v10:SendResponse>
```

- *MessageId*: ID of the message that was sent
- *MessageType*: Repeats the `MessageType` from the request
- *applicationType*: the application name is deduced by the LXR.CCN Connector from the `MessageType`.

4.4 Callback interface

The Callback interface describes the operations that must to be implemented by partner applications to process messages sent by the LXR.CCN Connector.

4.4.1 Receive

The Receive operation is invoked by the LXR.CCN Connector to transmit messages received from the CCN/CSI Gateway to the target application.

The Receive operation must be implemented by the partner application.

Here below, we provide a sample request sent by the LXR.CCN Connector:

```
<v101:ReceiveRequest
xmlns:v101="http://www.vivansa.com/xsize/gateway/ccncsi/cb/wsd/v10">
  <OriginDescriptor>
    <Partner>BE</Partner>
  </OriginDescriptor>
  <MessageDescriptor>
```

** Limited. Please do not disclose without prior consent of Vivansa. **

```

    <MessageType>CD801A-MSG.emcs</MessageType>
    <applicationType>EMCS</applicationType>
</MessageDescriptor>
<DataDescriptor>
    <Data>VGVzdCBtZXNzYWdl </Data>
</DataDescriptor>
    <QualityOfService>
        <Priority>1</Priority>
        <Integrity>true</Integrity>
        <Confidentiality>true</Confidentiality>
        <Compression>true</Compression>
        <CompressionId>LZW</CompressionId>
    </QualityOfService>
</v101:ReceiveRequest>

```

4.4.1.1 Parameters

- *Partner*: Code of the country that sent the message
- *MessageType*: CCN Message type
- *applicationType*: Application name detected by the CCN Adapter
- *Data*: Actual message payload, encoded in base64.

4.4.2 ReportProblem

The ReportProblem operation is invoked by the LXR.CCN Connector to transmit to the target application errors occurred in the asynchronous process.

The ReportProblem operation must be implemented by the partner application.

Here below, we provide a sample request sent by the LXR.CCN Connector:

```

<v10:ReportProblemRequest>
  <message>08BE0930495346</message>
  <severity>ERROR</severity>
  <detail>
    <error>
      <errorcode>XCA_BE0008</errorcode>
      <errorstring>Error while contacting CCN gateway</errorstring>
      <errorpointer>CcnProcessor.send</errorpointer>
    </error>
  </detail>
</v10:ReportProblemRequest>

```

** Limited. Please do not disclose without prior consent of Vivansa. **

```
</error>  
</detail> </v10:ReportProblemRequest>
```

4.4.2.1 Parameters

- *message*: contains the reference of the message where a problem was detected
- *severity*: can be ERROR or WARNING
- *error*: repeated for every error to report
- *errorcode*: error code mentioned in the technical specification ([\[R1\]](#))
- *errorstring*: human-readable error message
- *errorpointer*: location of the error in the application.

** Limited. Please do not disclose without prior consent of Vivansa. **

References

- [R1] LXR.CCN - Installation Guide (Ref: IG18839)
- [R2] LXR.CCN - Administration Guide (Ref: AG18840)
- [R3]
- [R4]
- [R5]
- [R6]
- [R6]

Acronyms

CCN	Common Communication Network
CCN/CSI	Common Communication Network/ Common System Interface
ID	Identification
ISO	International Standard Organisation
QoS	Quality of Service
WLS	WebLogic Server
WSDL	Web Service Description Language

Entire contents (C) 2009 by Vivansa (www.vivansa.com). All rights reserved. The reader of this publication shall not disclose, transfer or use and shall not allow any third parties to disclose, transfer or use any contents of this publication. The reader shall, at its sole expense, take reasonable measures to ensure such confidential contents is not disclosed, transferred or used, including but not limited to binding any employees, contractors or assigns to a similar confidentiality and non-disclosure agreement. The information contained in this publication has been obtained from sources believed to be reliable. Vivansa disclaims all warranties as to the accuracy, completeness or adequacy of such information. Vivansa shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The opinions expressed herein are subject to change without notice.