



BEA eLink Adapter for R/3 ALE User's Guide

BEA eLink Adapter for R/3 ALE, Version 1.5
Document Edition 1.5
January 2000

Copyright

Copyright © 1998, 1999, 2000 BEA Systems, Inc. All Rights Reserved.

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

BEA, ObjectBroker, TOP END, and TUXEDO are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Connect, BEA Manager, BEA MessageQ, Jolt and M3 are trademarks of BEA Systems, Inc.

eSolutions™ and eLink™ are trademarks.

All other company names may be trademarks of the respective companies with which they are associated.

eLink Adapter for R/3 ALE User's Guide

Document Edition	Part Number	Date	Software Version
1.5	870-001028-001	January 2000	eLink Adapter for R/3 ALE v1.5

About This Document

This document provides instructions for installing and configuring the BEA eLink Adapter for R/3 ALE product on a UNIX or Windows NT system.

The BEA eLink Adapter for R/3 ALE User's Guide is organized as follows:

- Chapter 1, "Introducing BEA eLink Adapter for R/3 ALE," introduces you to BEA eLink Adapter for R/3 ALE and provides overview information on integrating applications with R/3.
- Chapter 2, "Integrating with ALE," provides information about ALE integration, including processing IDOCs from R/3 and IDOCs into R/3.
- Chapter 3, "Preinstallation Tasks," describes system requirements and information that you need to review before installing BEA eLink Adapter for R/3 ALE.
- Chapter 4, "Installing BEA eLink Adapter for R/3 ALE," provides instructions for installing BEA eLink Adapter for R/3 ALE on UNIX and Windows NT systems.
- Chapter 5, "TUXEDO Initialization File," describes how to configure TUXEDO to enable integration with BEA eLink Adapter for R/3 ALE.
- Chapter 6, "Configuring ALE Integration," describes how to configure BEA eLink Adapter for R/3 ALE to enable access to Application Linking and Embedding (ALE) on your SAP R/3 system.
- Chapter 7, "Configuring R/3 Connections," describes how to configure the connections to your SAP R/3 system.
- Appendix A, "Sample cr3_ale.ubb File," describes the sample `cr3_ale.ubb` file that accompanies BEA eLink Adapter for R/3 ALE.

What You Need to Know

This document is intended for system managers who are responsible for installing products in the UNIX or Windows NT environments and for configuring layered products. Portions of this guide are intended for experienced R/3 users with knowledge of ALE configuration.

e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation or go directly to the “e-docs” Product Documentation page at <http://e-docs.beasys.com>.

How to Print the Document

You can print a copy of this document from a Web browser, one file at a time, by using the File—>Print option on your Web browser.

A PDF version of this document is available on the BEA eLink Adapter for R/3 ALE documentation Home page on the e-docs Web site (and also on the documentation CD). You can open the PDF in Adobe Acrobat Reader and print the entire document (or a portion of it) in book format. To access the PDFs, open the BEA eLink Adapter for R/3 ALE documentation Home page, click the PDF files button and select the document you want to print.

If you do not have the Adobe Acrobat Reader, you can get it for free from the Adobe Web site at <http://www.adobe.com/>.

Related Information

The following BEA WebLogic Enterprise documents contain information that is relevant to using the `idltjava` compiler and understanding how to implement Java CORBA applications in the WLE system.

For more information in general about Java IDL and Java CORBA applications, refer to the following sources.

- The OMG Web Site at <http://www.omg.org/>
- The Sun Microsystems, Inc. Java site at <http://java.sun.com/>

For more information about CORBA and distributed object computing, transaction processing, and Java, refer to the Bibliography at <http://edocs.beasys.com/>.

Contact Us!

Your feedback on the BEA WebLogic Enterprise documentation is important to us. Send us e-mail at docsupport@beasys.com if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the BEA eLink Adapter for R/3 ALE documentation.

In your e-mail message, please indicate that you are using the documentation for the BEA BEA eLink Adapter for R/3 ALE 5.0 release.

If you have any questions about this version of BEA BEA eLink Adapter for R/3 ALE, or if you have problems installing and running BEA BEA eLink Adapter for R/3 ALE, contact BEA Customer Support through BEA WebSupport at www.beasys.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address

-
- Your machine type and authorization codes
 - The name and version of the product you are using
 - A description of the problem and the content of pertinent error messages

Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Item
boldface text	Terms defined in the glossary.
Ctrl+Tab	You must press two or more keys simultaneously.
<i>italics</i>	Emphasis or book titles.
monospace text	Code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard. <i>Examples:</i> #include <iostream.h> void main () the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float
monospace boldface text	Significant words in code. <i>Example:</i> void commit ()
<i>monospace italic text</i>	Variables in code. <i>Example:</i> String <i>expr</i>

Convention	Item
UPPERCASE TEXT	Device names, environment variables, and logical operators. <i>Examples:</i> LPT1 SIGNON OR
{ }	A set of choices in a syntax line. The braces themselves should never be typed.
[]	Optional items in a syntax line. The brackets themselves should never be typed. <i>Example:</i> buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...
	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.
...	One of the following in a command line: <ul style="list-style-type: none"> ■ That an argument can be repeated several times in a command line. ■ That the statement omits additional optional arguments. ■ That you can enter additional parameters, values, or other information. The ellipsis itself should never be typed. <i>Example:</i> buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...
.	Omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.



Contents

About This Document

What You Need to Know	ii
e-docs Web Site	ii
How to Print the Document	ii
Related Information	iii
Contact Us!	iii
Documentation Conventions	iv

1. Introducing BEA eLink Adapter for R/3 ALE

About BEA eLink Adapter for R/3 ALE	1-1
SAP R/3 Environment Integration Issues	1-2
BEA Enterprise Application Integration Solution	1-2
Key Benefits	1-2
Business Applicability and Scalability	1-3
Licensing Requirements	1-3
Architectural Overview	1-3
Integrating Applications with R/3	1-5
TUXEDO ATMI	1-5
FML32 Field Definitions	1-6

2. Integrating with ALE

ALE Integration	2-1
Usage Scenarios for ALE Integration	2-2
Integrating R/3 and Non-R/3 Systems	2-2
Communicating Among R/3 Logical Systems	2-2
Information Flow for ALE Integration	2-4
Key ALE Concepts	2-5

Intermediate Documents (IDOCs)	2-5
Transaction IDs (TIDs)	2-7
Processing eLink to R/3 IDOCs	2-7
ALE eLink to R/3 Server	2-8
ALE eLink to R/3 Services (CR3_SUBMIT and CR3_IDOC_IN) ...	2-8
FML32 Field Definitions for eLink to R/3 IDOCs	2-8
Information Flow for eLink to R/3 IDOCs	2-9
Splitting eLink to R/3 IDOC Packets	2-11
Load Balancing High Volumes of eLink to R/3 IDOCs	2-12
Managing Transactional Integrity for eLink to R/3 IDOCs	2-13
About the TID Log File Used for eLink to R/3 IDOCs	2-13
Processing TIDs with eLink to R/3 IDOCs.....	2-14
Handling Problems with eLink to R/3 IDOCs	2-16
Processing R/3 to eLink IDOCs	2-17
ALE R/3 to eLink Client	2-18
Information Flow for R/3 to eLink IDOCs	2-18
Splitting R/3 to eLink IDOC Packets Into Individual IDOCs	2-21
Queuing R/3 to eLink IDOCs Into Multiple Queues	2-22
About the Destination Map File	2-23
Examples of Using a Destination Map File	2-24
Managing Data-Depending Routing (DDR)	2-28
Load Balancing High Volumes of R/3 to eLink IDOCs	2-28
Registering Multiple Program IDs	2-30
Managing Transactional Integrity for R/3 to eLink IDOCs	2-31
About the TID Log File Used for R/3 to eLink IDOCs	2-31
Processing TIDs with R/3 to eLink IDOCs.....	2-32
Handling Problems with R/3 to eLink IDOCs	2-34

3. Preinstallation Tasks

System Requirements	3-1
Determining the Installation Node	3-2
TUXEDO Requirements	3-3

4. Installing BEA eLink Adapter for R/3 ALE

Installing on a UNIX System	4-1
-----------------------------------	-----

Creating the conr3 User	4-2
Logging In as con3	4-2
Running install.sh	4-2
Installing on a Windows NT System	4-6

5. TUXEDO Initialization File

6. Configuring ALE Integration

Configuring the ALE eLink to R/3 Server	6-2
Configuring the UBB File for ALE eLink to R/3	6-2
Defining the Server Group	6-2
Defining the ALE eLink to R/3 Server and Environment File	6-3
Defining ALE eLink to R/3 Services	6-3
Setting Environment Variables for ALE eLink to R/3	6-4
Environment Variables for ALE eLink to R/3	6-4
Sample Environment File for ALE eLink to R/3 (cr3alein.env).....	6-6
Setting Up the eLink to R/3 Connection to R/3	6-6
Configuring Queues for ALE eLink to R/3	6-6
Specifying the TID File for ALE eLink to R/3	6-7
Splitting eLink to R/3 IDOC Packets Containing Multiple IDOCs ...	6-7
Configuring Logging for ALE eLink to R/3	6-7
Configuring Load Balancing for eLink to R/3 IDOCs	6-8
Configuring the ALE R/3 to eLink Client	6-9
Starting the ALE R/3 to eLink Client	6-9
Using the Startup Script for ALE R/3 to eLink	6-9
Command Line Options for ALE R/3 to eLink	6-10
Setting Environment Variables for ALE R/3 to eLink	6-11
Sample Startup Script for ALE R/3 to eLink (cr3aleout.bat)	6-12
Configuring Queues for ALE R/3 to eLink	6-13
Setting the Default Data-Dependent Routing Value	6-13
Configuring a Destination Map File	6-14
Splitting R/3 to eLink IDOC Packets	6-16
Specifying the TID File for ALE R/3 to eLink	6-17
Configuring Logging for ALE R/3 to eLink	6-17
Configuring Load Balancing for R/3 to eLink	6-18

Configuring Multiple Program IDs	6-18
Configuring Queues.....	6-20
TUXEDO Queues.....	6-20
Setting Up Queues for eLink to R/3 IDOCs.....	6-20
Defining the TMQUEUE and TMQFORWARD Servers in the UBB File	
6-21	
Setting Environment Variables	6-21
Setting Up Queues for R/3 to eLink IDOCs	6-22
Configuring the TID File Manager.....	6-22
Command Line Options	6-23
Running the TID File Manager	6-23
How the TID File Manager Works.....	6-24

7. Configuring R/3 Connections

Configuring eLink to R/3 Connections	7-1
Setting Environment Variables for the eLink to R/3 Connection	7-2
Configuring the sideinfo File	7-3
Contents of the sideinfo File	7-3
Sample Settings	7-4
Specifying the Location of the sideinfo File	7-5
Configuring R/3 to eLink Connections	7-5
Troubleshooting Connection Problems	7-6

A. Sample cr3_ale.ubb File

UNIX	A-2
Windows NT.....	A-4

Glossary

Index

1 Introducing BEA eLink Adapter for R/3 ALE

This topic introduces BEA eLink Adapter for R/3 ALE. It includes the following main sections:

- About BEA eLink Adapter for R/3 ALE
- SAP R/3 Environment Integration Issues
- BEA Enterprise Application Integration Solution
- Architectural Overview
- Integrating Applications with R/3

About BEA eLink Adapter for R/3 ALE

BEA eLink Adapter for R/3 ALE (application link enabling technology) is the infrastructure of choice for application integration of the SAP R/3 environment. BEA eLink Adapter for R/3 ALE works with mission-critical, high-performance middleware to enable easy application integration with and real-time access to SAP R/3 transactions, functions and data.

A production-proven solution, eLink Adapter for R/3 ALE provides an enterprise-based approach to integrating SAP R/3 with applications in all industry-leading computer environments.

SAP R/3 Environment Integration Issues

SAP R/3 is a complete information technology solution that provides an integrated suite of financial, distribution, human resources, and manufacturing applications that can be customized to meet customer needs.

Many firms implementing SAP R/3 realize that full integration between SAP R/3 and their own applications is crucial in order to maximize the effectiveness of their entire environment. However, because numerous, heterogeneous computing environments exist across the enterprise, integration has been costly to implement and difficult to maintain.

BEA Enterprise Application Integration Solution

BEA eLink Adapter for R/3 ALE assures ready, high-performance and transparent access to mission-critical applications and information across the network with a single, standard programming interface.

BEA eLink Adapter for R/3 ALE builds gateways that can be customized for specific customer needs. These gateways enable interoperability between BEA middleware applications and legacy mainframe environments. Utilization of BEA eLink Adapter for R/3 ALE provides distributed access to SAP R/3 data from industry-leading applications and eliminates the need for applications running in the SAP R/3 environment in order to access ‘bet-the-business’ data and functions.

Key Benefits

- Incorporates third-party packages into the SAP R/3 environment
- Empowers non-SAP R/3 applications to execute SAP R/3 applications

- Enables the reduction of application development time
- Allows significant "reutilization" of existing servers
- Provides faster and smoother deployment of SAP R/3
- Accommodates links between non-SAP R/3 GUIs and SAP R/3 or between cooperating servers

Business Applicability and Scalability

In addition to empowering the successful integration of TUXEDO-enabled applications with SAP R/3 in heterogeneous multi-platform environments, eLink Adapter for R/3 ALE makes it possible for companies to access SAP R/3 services, maintain secure business data, and simplify forward migration to new releases of SAP R/3. BEA eLink Adapter for R/3 ALE's integration efficiency has the robustness and high-performance required of a permanent element in the overall business solution.

Licensing Requirements

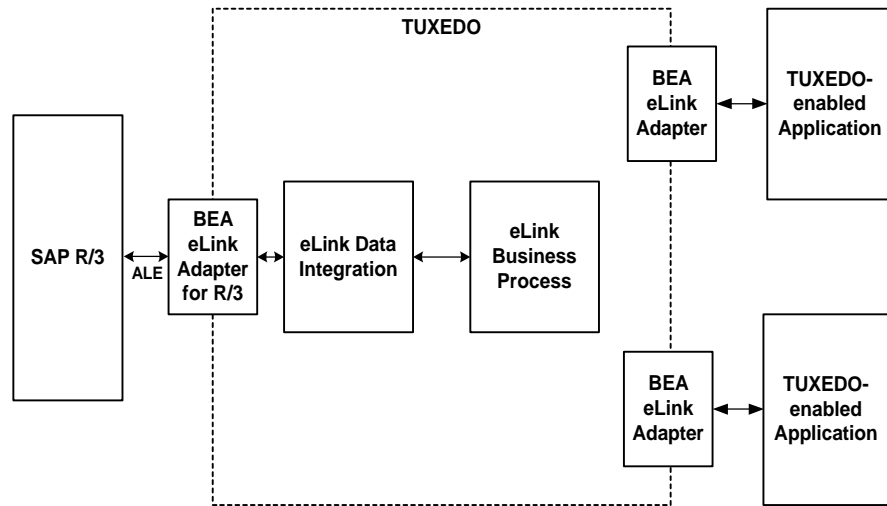
The eLink adapter for R/3 ALE requires the following two licenses for operation:

- eLink Platform 1.1
- eLink Adapter for R/3 ALE 1.5

Architectural Overview

Figure 1-1 shows the architecture of BEA eLink Adapter for R/3 ALE:

Figure 1-1 Architecture of BEA eLink Adapter for R/3 ALE



BEA eLink Adapter for R/3 ALE works with the following components:

- **eLink Data Integration** manages data transformation: data formats, data content, and rules.
- **eLink Business Process** manages process flow: state-based business processes consisting of multiple tasks.

BEA eLink Adapter for R/3 ALE communicates with R/3 via SAP's Application Link Enabling (ALE) technology. The adapter enables bi-directional communication between R/3 and eLink.

The BEA eLink Adapter for R/3 ALE provides asynchronous communication between eLink and SAP R/3, as shown in Figure 1-2. See your SAP documentation for more information about SAP ALE technologies.

Figure 1-2 Communication between BEA eLink Adapter for R/3 ALE and SAP R/3



Integrating Applications with R/3

This section introduces the key concepts involved in integrating TUXEDO-enabled applications with R/3 using BEA eLink Adapter for R/3 ALE. It includes the following topics:

- TUXEDO ATMI
- FML32 Field Definitions

TUXEDO ATMI

BEA eLink Adapter for R/3 ALE and TUXEDO-enabled applications communicate by exchanging TUXEDO FML32 buffers. Client applications that need access to the data and functionality of R/3 use the TUXEDO application to transaction monitor interface (ATMI) to send request messages to the BEA eLink Adapter for R/3 ALE and receive the response messages. The messages exchanged are FML32 buffers. Server applications that allow R/3 access to their data and functionality will use the TUXEDO ATMI to receive request messages from BEA eLink Adapter for R/3 ALE and send response messages.

1 Introducing BEA eLink Adapter for R/3 ALE

For more information on the features of TUXEDO, programming with the TUXEDO ATMI, and encoding and decoding FML buffers, see the *TUXEDO Programmer's Guide and the BEA TUXEDO Reference Manual* (Section 3C).

FML32 Field Definitions

BEA eLink Adapter for R/3 ALE comes with an FML field table (`cr3_ale.fml` file) that defines the fields used in FML32 buffers. This file resides in the following directory:

your eLink installation path/elink/adapters/sapr3_ale/config

The `cr3_ale.fml` file contents are as follows:

Listing 1-1 FML Field Table (`cr3_ale.fml` file)

```
# BEA eLink Adapter for R/3 ALE - FML fields

# name                number type      flags  comments
CR3 RFC TID           100  string      -      -
CR3 IDOC CONTROL      101  string      -      -
CR3 IDOC DATA        102  string      -      -
CR3 IDOC               103  string      -      -
CR3 TARGET ID         104  string      -      -
CR3 ERROR TEXT        105  string      -      -

# Data Integration (Mercator) FML fields

# eLink error FML fields

# name                number type      flags  comments
ELINK ADAPTER ERROR CODE  200  string      -      -
ELINK ADAPTER ERROR      201  string      -      -
ELINK APP ERR            202  string      -      -
```

Note: The field numbers are default values only. You can change these values if they are already in use by another application.

Table 1-1 describes these FML fields:

Table 1-1 FML Field Definitions in the cr3_ale.fml File

Field Name	Description
CR3_RFC_TID	Field that specifies the transaction ID. See “Transaction IDs (TIDs)” in Chapter 2, “Integrating with ALE.”
CR3_IDOC_CONTROL	Contains one or more control records for the IDOC packet.
CR3_IDOC_DATA	Contains one or more data records for the IDOC packet.
CR3_IDOC	Field that contains IDOC packet data. See Chapter 2, “Integrating with ALE.”
CR3_TARGET_ID	Data Dependent Routing value. See Chapter 2, “Integrating with ALE.”
CR3_ERROR_TEXT	Error text.
ELINK_ADAPTER_ERR_CODE	eLink error category.
ELINK_ADAPTER_ERR	BEA eLink Adapter for R/3 ALE error code and text.

Note: BEA eLink Adapter for R/3 ALE imposes no restriction on the length of string values sent and received using TUXEDO. The size is determined by the target application. For example, the length of an import parameter is specified by R/3. Note that the maximum size of a TUXEDO message is 2GB.

See your TUXEDO documentation for more information about FML32 buffers and field definition tables.

1 *Introducing BEA eLink Adapter for R/3 ALE*

2 Integrating with ALE

This topic describes how to integrate SAP R/3 with SAP Application Linking and Embedding (ALE) technology in the BEA eLink environment. It includes the following main sections:

- ALE Integration
- Processing eLink to R/3 IDOCs
- Processing R/3 to eLink IDOCs

For information about setting up ALE processing, see the following:

- Chapter 6, “Configuring ALE Integration”
- Chapter 7, “Configuring R/3 Connections”

ALE Integration

The following topics provide important conceptual information about integrating with ALE:

- Usage Scenarios for ALE Integration
- Information Flow for ALE Integration
- Key ALE Concepts

Usage Scenarios for ALE Integration

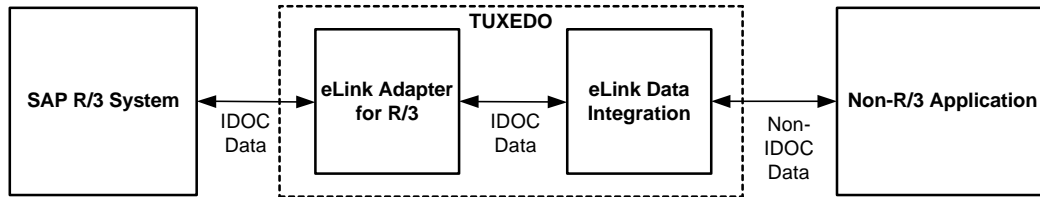
Common ALE integration implementations of eLink Adapter for R/3 ALE include:

- **Integrating R/3 and non-R/3 systems** by using TUXEDO and eLink Adapter for R/3 ALE to exchange intermediate documents (IDOCs) and non-IDOCs across application and platform boundaries. For a discussion of IDOCs, see “Key ALE Concepts” later in this section.
- **Communicating among R/3 systems** by using TUXEDO and eLink Adapter for R/3 ALE to transport IDOCs reliably and efficiently among R/3 logical systems.

Integrating R/3 and Non-R/3 Systems

Figure 2-1 shows how eLink Adapter for R/3 ALE, in conjunction with BEA eLink Data Integration, can be used to integrate R/3 with non-R/3 systems:

Figure 2-1 Integrating R/3 and Non-R/3 Systems

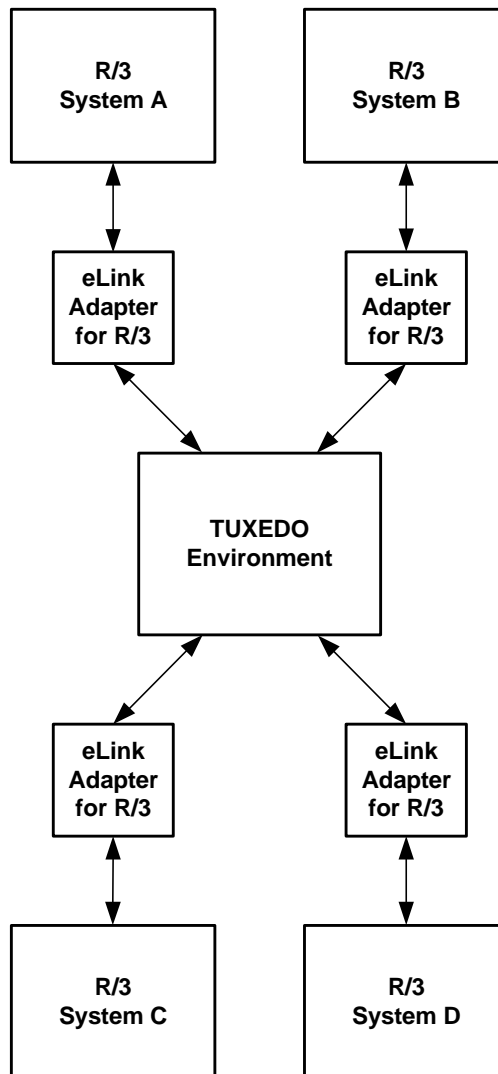


In this scenario, these BEA components provide communication and data transformation services that enable the exchange of IDOC (R/3) and non-IDOC (non-R/3) data between R/3 and non-R/3 systems.

Communicating Among R/3 Logical Systems

Figure 2-2 shows how TUXEDO and eLink Adapter for R/3 ALE can transport IDOCs among R/3 logical systems:

Figure 2-2 Communication Among R/3 Logical Systems

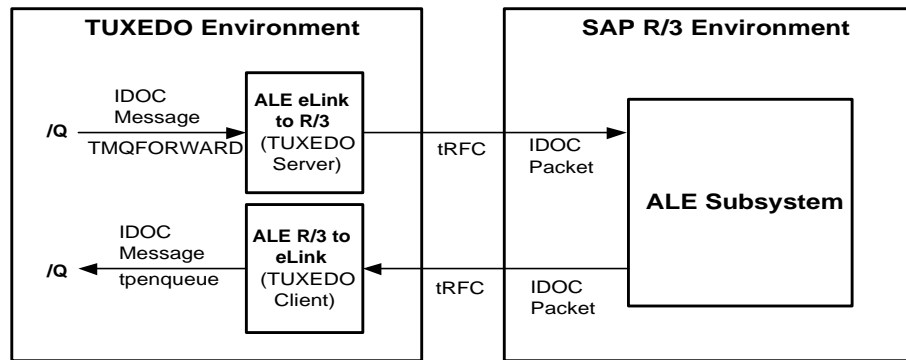


In this scenario, eLink Adapter for R/3 ALE and TUXEDO provide reliable and efficient communication services that enable the transport of IDOC packets between R/3 logical systems, thereby reducing the load on SAP communication services.

Information Flow for ALE Integration

Figure 2-3 shows the information flow for the two main ALE processes: ALE R/3 to eLink and ALE eLink to R/3:

Figure 2-3 Overview of Information Flow for ALE Processing



- **ALE eLink to R/3** is a TUXEDO server that submits IDOC packets to R/3. ALE eLink to R/3 receives each IDOC packet as an FML32 message buffer (forwarded from a TUXEDO queue), adds the IDOC data for R/3, and submits the IDOC packet to R/3 via transactional RFC (tRFC). See Information Flow for eLink to R/3 IDOCs later in this section for more information.
- **ALE R/3 to eLink** is a TUXEDO client that receives IDOC packets from R/3 via Transactional RFC (tRFC). ALE R/3 to eLink adds each IDOC packet data into an FML32 message buffer and queues it into one or more TUXEDO queues. See Information Flow for R/3 to eLink IDOCs later in this section for more information.

Key ALE Concepts

Two key concepts are used in ALE processing:

- Intermediate Documents (IDOCs)
- Transaction IDs (TIDs)

Intermediate Documents (IDOCs)

In SAP's R/3 environment, an *intermediate document (IDOC)* is a container for distributing R/3 application data among R/3 logical systems and for exchanging R/3 application data with non-R/3 systems.

In ALE processing, an IDOC consists of two types of records:

- The *control record* uniquely identifies the IDOC, specifying such information as the identity of the sender, the target (Logical System ID), message type, IDOC-type, and status. The length of a control record is 464 bytes.
- The *data records* uniquely identify a segment that contains application data. The length of a data record is 1055 bytes, which consists of a 55-byte header and a 1000-byte segment. The header identifies the segment type and hence the segment's structure.

Each IDOC is a sequential buffer that contains one control record and one or more data records, as shown in Figure 2-4:

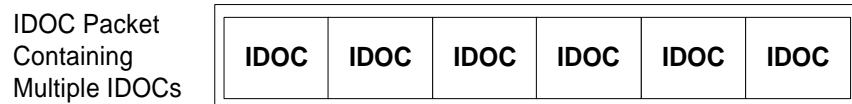
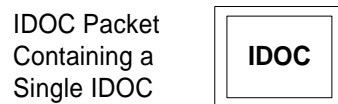
Figure 2-4 Structure of an ALE IDOC

Control Record	
Sender	
Receiver	
Message Type	
IDOC Type	
Status	
Data Records	
Header	Segment
Header	Segment
Header	Segment
Header	Segment ...

For eLink to R/3 IDOCs, ALE eLink to R/3 validates the size and structure of control records and data records. For each IDOC, ALE eLink to R/3 also verifies that the DOCNUM data in the control record matches the DOCNUM data in associated data records.

An *IDOC packet* is a message that contains one or more individual IDOCs, as shown in Figure 2-5:

Figure 2-5 Types of IDOC Packets



The R/3 system separately maintains status information about the creation, receipt, and processing of IDOCs. See your SAP R/3 documentation for more information about IDOCs.

In the TUXEDO environment, IDOC packets are transmitted in FML32 messages. These field definitions are specified in the `cr3_ale.fml` file, as described in “FML32 Field Definitions” in Chapter 1, “Introducing BEA eLink Adapter for R/3 ALE.”

Transaction IDs (TIDs)

R/3 assigns a unique Transaction ID (TID) to each IDOC packet it processes. R/3 uses TIDs to manage transactional integrity:

- **eLink to R/3.** For eLink to R/3 IDOC packets, R/3 uses TIDs to guarantee receipt once and only once.
- **R/3 to eLink.** For R/3 to eLink IDOC packets, R/3 uses TIDs to guarantee delivery once and only once.

In the TUXEDO environment, ALE R/3 to eLink and ALE eLink to R/3 monitor TIDs through the use of TID log files. See *Managing Transactional Integrity for eLink to R/3 IDOCs* and *Managing Transactional Integrity for R/3 to eLink IDOCs* for more information.

Processing eLink to R/3 IDOCs

The following topics describe how to process eLink to R/3 IDOCs using the ALE eLink to R/3 server. It includes the following topics:

- ALE eLink to R/3 Server
- Information Flow for eLink to R/3 IDOCs
- Splitting eLink to R/3 IDOC Packets
- Load Balancing High Volumes of eLink to R/3 IDOCs
- Managing Transactional Integrity for eLink to R/3 IDOCs
- Handling Problems with eLink to R/3 IDOCs

ALE eLink to R/3 must be properly configured before it can process IDOCs. See “Configuring ALE eLink to R/3 Server” in Chapter 6, “Configuring ALE Integration” for more information.

ALE eLink to R/3 Server

ALE eLink to R/3 is a TUXEDO server that submits IDOC packets to R/3. ALE eLink to R/3 receives each IDOC packet as an FML32 message buffer (forwarded from a TUXEDO queue) and it submits the IDOC packet to R/3 via tRFC. ALE eLink to R/3 uses a TID log file to track the TIDs associated with IDOC packets to guarantee delivery to R/3 once and only once. The name of the executable for ALE eLink to R/3 is `cr3alein`.

ALE eLink to R/3 Services (CR3_SUBMIT and CR3_IDOC_IN)

ALE eLink to R/3 provides two services that process eLink to R/3 IDOC packets:

Service Name	Description
CR3_SUBMIT	Receives an incoming FML32 buffer containing an IDOC packet from a TUXEDO queue; validates the IDOC packet data; obtains a TID from R/3 for the IDOC packet; binds the TID into the IDOC packet; and queues the IDOC message into the CR3_IDOC_IN queue.
CR3_IDOC_IN	Receives the IDOC packet from the CR3_IDOC_IN queue; encodes the IDOC data for R/3; and submits the IDOC packet to R/3 for processing.

FML32 Field Definitions for eLink to R/3 IDOCs

ALE eLink to R/3 uses the following FML32 field definitions in IDOC messages:

Table 2-1 FML32 Fields for ALE eLink to R/3 Messages

Field	Data Type	Description
CR3_IDOC	string	Contains an IDOC packet consisting of one or more IDOCs.
CR3_TARGET_ID	string	Contains a data-dependent routing value. Required even if it contains only a dummy value.

Table 2-1 FML32 Fields for ALE eLink to R/3 Messages

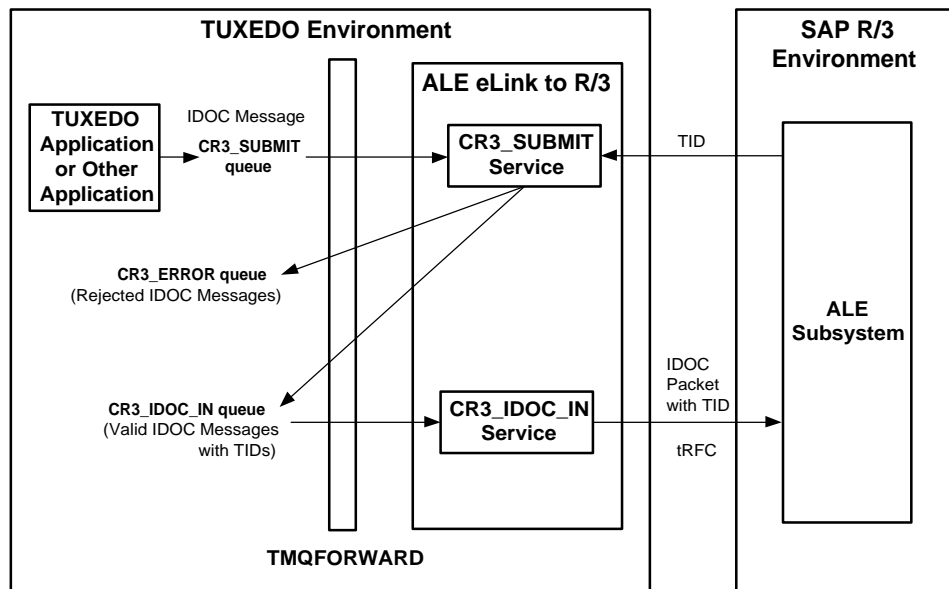
Field	Data Type	Description
CR3_RFC_TID	string	Contains the Transaction ID (TID) for the IDOC packet.
CR3_IDOC_CONTROL	string	Contains one or more control records for the IDOC packet.
CR3_IDOC_DATA	string	Contains one or more data records for the IDOC packet.

These fields are defined in the `cr3_ale.fml` file, as described in “FML32 Field Definitions” in Chapter 1, “Introducing BEA eLink Adapter for R/3 ALE.”

Information Flow for eLink to R/3 IDOCs

Figure 2-6 shows the information flow for eLink to R/3 IDOCs:

Figure 2-6 Information Flow for eLink to R/3 IDOCs



The information flow for ALE eLink to R/3 proceeds in the following sequence:

1. One or more instances of ALE eLink to R/3 (a TUXEDO server) start up.
2. An application (a TUXEDO application, eLink Data Integration, or some other tool) constructs an FML32 buffer containing IDOC data and queues it into the CR3_SUBMIT queue.
3. The TMQFORWARD TUXEDO service dequeues the IDOC message from the CR3_SUBMIT queue and submits it to the CR3_SUBMIT service of the ALE eLink to R/3 server.
4. The CR3_SUBMIT service receives the IDOC message and validates its contents:
 - It checks for the existence of the CR3_TARGET_ID field in the message buffer.
 - It checks the structure and size of the control record and associated data records.

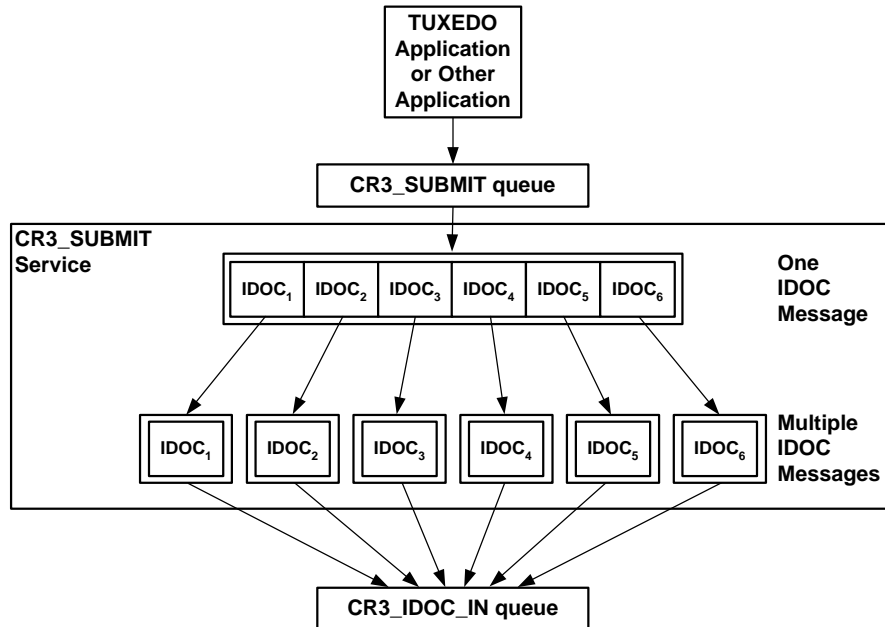
- Within the IDOC, it compares the DOCNUM fields in the control record and associated data records to verify that they match.
5. After validation, the CR3_SUBMIT service takes one of the following actions:
 - If the IDOC message fails validation, CR3_SUBMIT sends it to the error queue (CR3_ERROR queue).
 - If the IDOC message passes validation, CR3_SUBMIT obtains a TID from R/3, binds the TID to the IDOC packet in the FML32 buffer by encoding the TID in the message buffer (the CR3_RFC_TID field), and queues the IDOC message into the CR3_IDOC_IN queue.
 6. TMQFORWARD dequeues the IDOC message from the CR3_IDOC_IN queue and submits it to the CR3_IDOC_IN service of the ALE eLink to R/3.
 7. The CR3_IDOC_IN service submits the IDOC packet and TID to R/3.

ALE eLink to R/3 uses a TID log file to manage transactional integrity. See *Managing Transactional Integrity for R/3 to eLink IDOCs* later in this section for more information.

Splitting eLink to R/3 IDOC Packets

By default, ALE eLink to R/3 passes an IDOC message containing multiple IDOCs to R/3 in a single packet. You can configure ALE eLink to R/3 to split IDOC messages containing multiple IDOCs into individual IDOC messages, each with its own TID. For example, if an IDOC message contains six IDOCs, ALE eLink to R/3 can create six IDOC separate packets, each containing a single IDOC and associated with a unique TID. Figure 2-7 shows splitting IDOC packets and queuing them into the CR3_IDOC_IN queue:

Figure 2-7 Splitting eLink to R/3 IDOC Packets



Splitting IDOC packets provides additional flexibility for processing eLink to R/3 IDOCs. However, this configuration can also increase load on the R/3 system and reduce throughput performance. For example, an IDOC packet containing six IDOCs requires two RFC calls: one to request the TID and another to submit the IDOC packet to R/3. Six IDOC packets containing a single IDOC each, however, requires twelve separate RFC calls: six to request TIDs and six to submit each IDOC packet to R/3.

See “Splitting eLink to R/3 IDOC Packets Containing Multiple IDOCs” in Chapter 6, “Configuring ALE Integration,” for instructions.

Load Balancing High Volumes of eLink to R/3 IDOCs

Multiple instances of ALE eLink to R/3 can log onto R/3 and submit eLink to R/3 IDOCs for processing. For deployments that involve high volumes of eLink to R/3 IDOCs, you can enhance system performance by balancing the load across multiple instances of ALE eLink to R/3. See “Configuring Load Balancing for eLink to R/3 IDOCs” in Chapter 6, “Configuring ALE Integration,” for instructions.

Managing Transactional Integrity for eLink to R/3 IDOCs

ALE eLink to R/3 manages transactional integrity for eLink to R/3 IDOCs to guarantee that it delivers an IDOC packet to R/3 once and only once. R/3 uses the TID to guarantee that it processes the IDOC packet exactly once. If an attempt to submit an IDOC packet to R/3 fails, ALE eLink to R/3 retries using the same TID. ALE eLink to R/3 uses a TID log file to track the transaction IDs (TIDs) that R/3 assigns to each eLink to R/3 IDOC packet. See Transaction IDs (TIDs) earlier in this document for an introduction to TIDs.

About the TID Log File Used for eLink to R/3 IDOCs

The TID log file used with eLink to R/3 IDOCs contains information about TIDs that ALE eLink to R/3 has received and processed. Each row in the TID file represents the TID for a separate IDOC packet and contains three fixed-position columns of information:

Table 2-2 Columns in the TID Log File for eLink to R/3 IDOCs

Column	Description
Date-Time Stamp	Date and time at which the state of this TID was last updated in the TID log file.
TID	TID that R/3 assigned to the IDOC packet.
Status	One of the following strings: <ul style="list-style-type: none"> ■ CREATED indicates that ALE eLink to R/3 has successfully associated a TID with the IDOC packet and queued it into the CR3_IDOC_IN queue. ■ CONFIRMED indicates that ALE eLink to R/3 has successfully passed the IDOC packet onto R/3.

The following example shows a sample TID file for ALE eLink to R/3:

```
Tue Apr 27 14:27:40 1999 0A0201FD03F937262C600004 CONFIRMED
Tue Apr 27 14:29:39 1999 0A0201FD03F937262CD90007 CONFIRMED
Tue Apr 27 14:46:58 1999 0A0201FD03F9372630E8000A CONFIRMED
```

2 Integrating with ALE

Tue Apr 27 15:52:30 1999 0A0201FD041637263FC60013 CONFIRMED

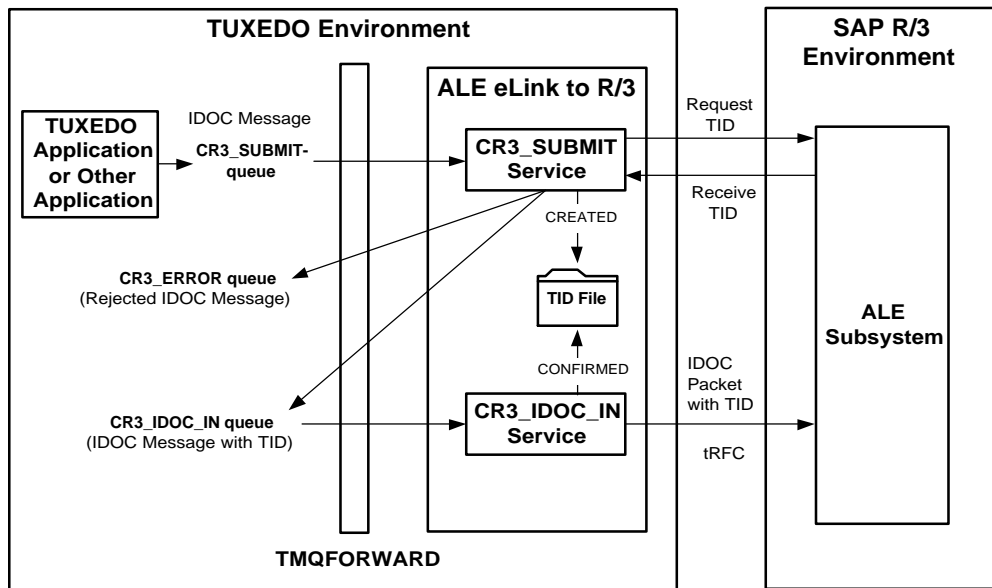
The `CR3_ALEIN_TID_FILE` environment variable specifies the location of the TID log file for ALE eLink to R/3. See “Setting Environment Variables” in Chapter 6, “Configuring ALE Integration,” for more information.

Use the `cr3tidmanager` program to manage the size and number of entries kept in TID files. See “Configuring the TID File Manager” in Chapter 6, “Configuring ALE Integration,” in this guide.

Processing TIDs with eLink to R/3 IDOCs

Figure 2-8 shows how ALE eLink to R/3 uses the TID log file to manage transactional integrity for eLink to R/3 IDOCs:

Figure 2-8 TID Processing for ALE eLink to R/3



The eLink to R/3 IDOC process involves two transaction boundaries:

- The first transaction boundary ensures that the `CR3_SUBMIT` service has bound the IDOC packet and TID together and has successfully queued the FML32 message buffer into the `CR3_IDOC_IN` queue. This transaction is the entry point for any external application that submits an IDOC packet to ALE eLink to R/3.

- The second transaction boundary ensures that the CR3_IDOC_IN service has successfully submitted the IDOC packet and associated TID to R/3. This transaction is an internal process that submits the IDOC packet and TID to R/3 until it succeeds.

First Transaction Boundary

For the first transaction boundary, the information flow proceeds in the following sequence:

1. TMQFORWARD starts a new TUXEDO transaction, unqueues an IDOC message from the CR3_SUBMIT queue, and submits the IDOC message to the CR3_SUBMIT service of the ALE eLink to R/3 server.
2. After validating the IDOC data, the CR3_SUBMIT service requests a TID from R/3.
3. R/3 generates a unique TID and returns it to the CR3_SUBMIT service.
4. The CR3_SUBMIT service opens the TID log file.
5. The CR3_SUBMIT service searches for the TID in the TID log file:
 - If the TID is not found, the CR3_SUBMIT service appends a new entry for the TID, writes the date-time stamp, TID, and state (CREATED) in the entry, and then proceeds to the next step.
 - If the TID is found, ALE R/3 to eLink requests a new TID from R/3 because it is already processing the current TID.
6. The CR3_SUBMIT service binds the TID to the IDOC message (by encoding the TID in the CR3_RFC_TID field in the buffer) and assigns the message to the CR3_IDOC_IN queue.
7. The CR3_SUBMIT service returns TPSUCCESS or TPFAIL, as appropriate, to TMQFORWARD.
8. TMQFORWARD closes the transaction, committing the transaction if TPSUCCESS was returned, or rolling back the transaction if TPFail was returned. If the transaction is rolled back, the IDOC message remains in the CR3_SUBMIT queue.

Second Transaction Boundary

For the second transaction boundary, the information flow proceeds in the following sequence:

1. TMQFORWARD starts a new transaction, dequeues an IDOC message from the CR3_IDOC_IN queue, and submits the IDOC message to the CR3_IDOC_IN service of the ALE eLink to R/3 server.
2. The CR3_IDOC_IN service encodes the IDOC packet for R/3 and submits the IDOC packet to R/3.
3. If the IDOC packet is successfully sent, the CR3_IDOC_IN service opens the TID log file, finds the TID, and updates the date-time stamp and state (CONFIRMED) in the log file.
4. CR3_IDOC_IN returns the result of the send request (TPSUCCESS or TPFail) to TMQFORWARD.
5. TMQFORWARD closes the transaction, committing the transaction if TPsUCCESS was returned, or rolling back the transaction if TPFail was returned. If the transaction is rolled back, the IDOC message remains in the CR3_IDOC_IN queue.

Handling Problems with eLink to R/3 IDOCs

ALE eLink to R/3 uses TUXEDO's transaction management capabilities to ensure transactional integrity for eLink to R/3 IDOCs. The following table lists problems that can occur with eLink to R/3 IDOCs:

Table 2-3 Handling Problems with eLink to R/3 IDOCs

Problem	Description
Invalid IDOC structure	If an IDOC packet fails validation, the CR3_SUBMIT service queues the FML32 message into the CR3_ERROR queue and returns TPsUCCESS to TMQFORWARD.
No CR3_TARGET_ID	If an IDOC message contains no CR3_TARGET_ID field, the CR3_SUBMIT service queues the FML32 message into the CR3_ERROR queue and returns TPsUCCESS to TMQFORWARD.

Table 2-3 Handling Problems with eLink to R/3 IDOCs

Problem	Description
TID Not received from R/3	If R/3 does not return a TID, CR3_SUBMIT CR3_SUBMIT returns TPFAIL to TMQFORWARD, and TMQFORWARD rolls back the transaction.
Send attempt to R/3 failed	If the CR3_IDOC_IN service does not successfully send the IDOC packet to R/3 (for example, the R/3 system is down), CR3_IDOC_IN returns TPFAIL to TMQFORWARD, and TMQFORWARD rolls back the transaction. The IDOC packet remains in the CR3_IDOC_IN queue until a subsequent send attempt succeeds.

Note: You must write an application to explicitly unqueue and handle messages in the CR3_ERROR queue.

Processing R/3 to eLink IDOCs

The following topics describe how to process R/3 to eLink IDOCs using the ALE R/3 to eLink client:

- ALE R/3 to eLink Client
- Information Flow for R/3 to eLink IDOCs
- Splitting R/3 to eLink IDOC Packets Into Individual IDOCs
- Queuing R/3 to eLink IDOCs Into Multiple Queues
- Managing Data-Depending Routing (DDR)
- Load Balancing High Volumes of R/3 to eLink IDOCs
- Registering Multiple Program IDs
- Managing Transactional Integrity for R/3 to eLink IDOCs
- Handling Problems with R/3 to eLink IDOCs

ALE R/3 to eLink must be configured properly before it can process R/3 to eLink IDOCs. For information about setting up ALE R/3 to eLink, see “TUXEDO Initialization File” in Chapter 6, “Configuring ALE Integration.”

ALE R/3 to eLink Client

ALE R/3 to eLink is a TUXEDO client that receives IDOC packets from R/3 via Transactional RFC (tRFC). ALE R/3 to eLink encodes each IDOC packet into an FML32 message buffer and queues it into a TUXEDO queue. ALE R/3 to eLink uses a TID log file to track the IDOC packets that it processes to ensure that it queues an IDOC packet from R/3 once and only once. The name of the executable for ALE R/3 to eLink is cr3aleout.

ALE R/3 to eLink uses the following FML32 field definitions in IDOC messages:

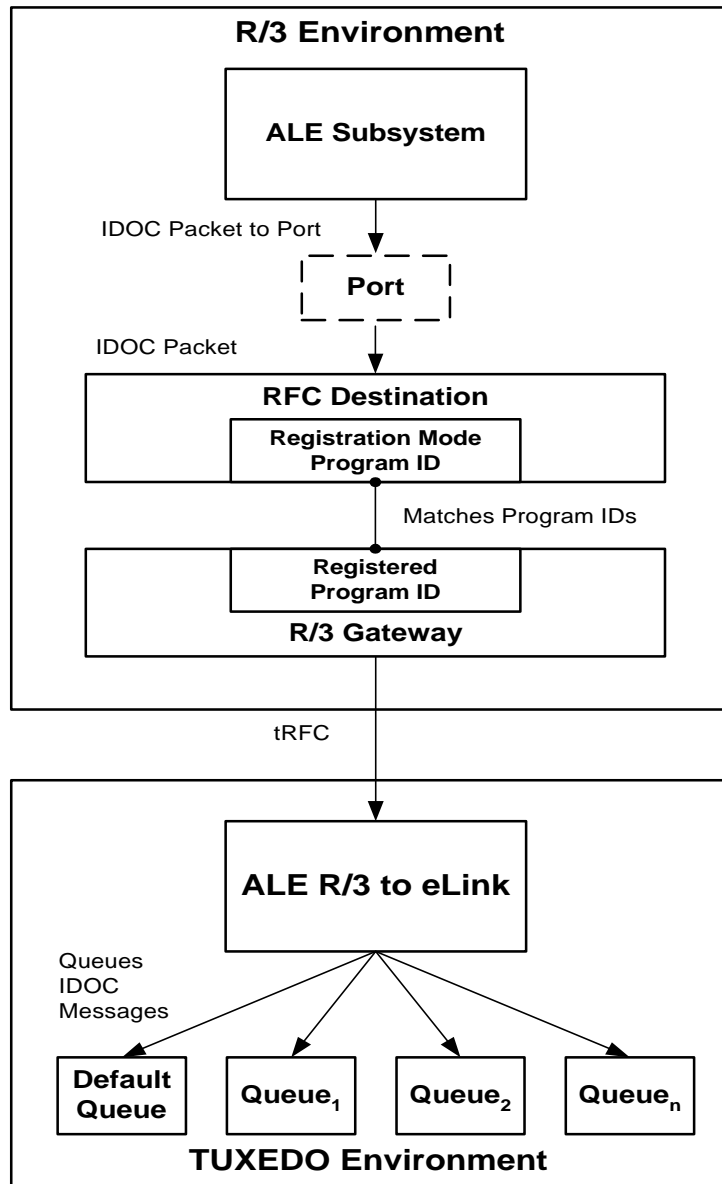
Table 2-4 FML32 Fields for ALE R/3 to eLink Messages

Field	Data Type	Description
CR3_IDOC	string	Contains a string of one or more IDOCs.
CR3_TARGET_ID	string	Data-dependent routing value.
MERCATOR_FV_IN	string	Either Y or N. See <i>the BEA eLink Data Integration Option v1.x</i> for more information.

Information Flow for R/3 to eLink IDOCs

Figure 2-9 illustrates the information flow for ALE R/3 to eLink:

Figure 2-9 Information Flow for R/3 to eLink IDOCs



The information flow proceeds in the following sequence:

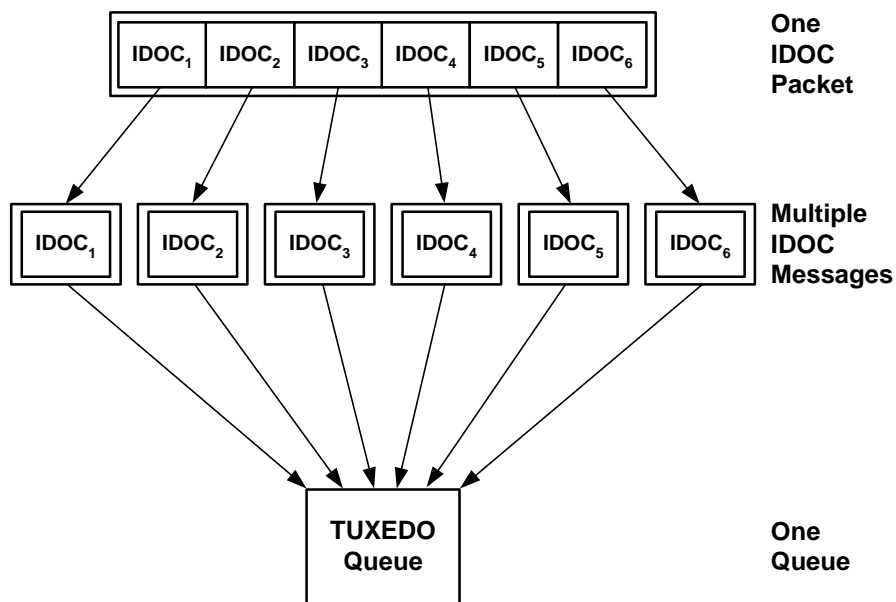
1. One or more instances of ALE R/3 to eLink (a TUXEDO client) start up and register a program ID with the R/3 Gateway. ALE R/3 to eLink runs in register mode and listens for IDOC packets associated with that program ID on the registered port. This program ID corresponds to a particular RFC destination.
2. R/3 submits an IDOC packet to a port (rather than to a file or another R/3 system) for a specific RFC destination.
3. R/3 sends the IDOC packet to an instance of ALE R/3 to eLink that is registered on the program ID of the RFC destination.
4. ALE R/3 to eLink receives the IDOC packet and processes the IDOC data according to the way that ALE R/3 to eLink is configured, with or without a destination map file:
 - If a destination map file is *not* configured, then ALE R/3 to eLink sends the IDOC packet to the default queue. If the CR3_ALE_DEFAULT_IDOC_SPLIT environment variable is set to “Y”, then ALE R/3 to eLink splits IDOC packets containing multiple IDOCs into separate IDOC messages, each containing a single IDOC, as described in “Splitting R/3 to eLink IDOC Packets Into Individual IDOCs” on page 2-21.
 - If a destination map file is configured, then ALE R/3 to eLink automatically splits IDOC packets containing multiple IDOCs into separate IDOC messages, each containing a single IDOC. ALE R/3 to eLink uses the settings in the destination map file to determine the target queue for each IDOC as well as other processing options. See “Queuing R/3 to eLink IDOCs Into Multiple Queues” on page 2-22 for more information.

ALE R/3 to eLink uses a TID log file to manage transactional integrity. See Managing Transactional Integrity for R/3 to eLink IDOCs later in this topic for more information.

Splitting R/3 to eLink IDOC Packets Into Individual IDOCs

You can configure ALE R/3 to eLink to split IDOC packets containing multiple IDOCs into separate IDOC messages, each containing a single IDOC. By default, ALE R/3 to eLink encodes the entire IDOC packet into a single occurrence of the CR3_IDOC field in the message buffer and queues the entire IDOC packet into a single message. If you set the CR3_ALE_DEFAULT_IDOC_SPLIT environment variable to “Y”, however, ALE R/3 to eLink will split the IDOC packet into individual IDOC messages and then queue all IDOC packets into a single queue as shown in Figure 2-10:

Figure 2-10 Splitting IDOC Packets and Queuing IDOC Messages to a Single Queue



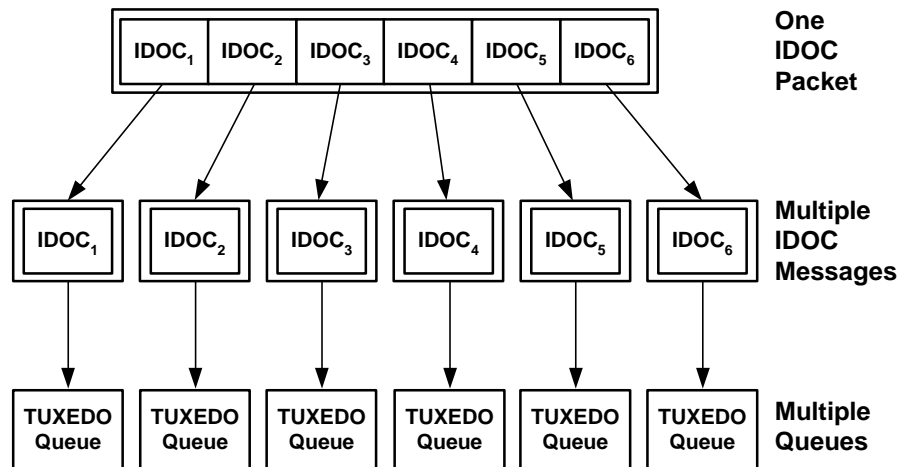
See “Setting Environment Variables for ALE R/3 to eLink” in Chapter 6, “Configuring ALE Integration,” in this guide for more information about setting the CR3_ALE_DEFAULT_IDOC_SPLIT environment variable.

Queuing R/3 to eLink IDOCs Into Multiple Queues

ALE R/3 to eLink can be configured to use a destination map file so that it can place IDOC messages into different target queues, manage data-dependent routing, and group similar IDOC messages into larger IDOC messages. ALE R/3 to eLink makes routing and grouping decisions about individual IDOCs according to two settings specified in an IDOCs control record: the logical system ID of the target R/3 system and the IDOC message type.

If a destination map file is used, ALE R/3 to eLink automatically splits IDOC packets containing multiple IDOCs into separate IDOC messages, each one containing a single IDOC. It then queues the IDOC messages using the settings in the destination map file:

Figure 2-11 Splitting IDOC Packets and Queuing to Multiple Queues



Before you can use a destination map file, you must set up the target queue spaces and queues (including the default queue space and queue), create the destination map file, and specify the name and location of the file in the CR3_ALEOUT_DEST_MAP_FILE environment variable. See “Configuring a Destination Map File” in Chapter 6, “Configuring ALE Integration,” for this procedure.

About the Destination Map File

The destination map file is an ASCII text file. Each row in the file denotes a separate entry and specifies the following information:

- Logical system ID of the target R/3 logical system.
- Message type specified in the IDOC.
- Compress flag indicating whether to compress IDOC packets (with matching logical system and IDOC message type values) into a single IDOC message in the queue.
- Target ID representing the data-dependent routing value.
- Destination queue space name.
- Destination queue name.

For each R/3 to eLink IDOC, ALE R/3 to eLink searches the destination map file for the target R/3 system and IDOC message type specified in an IDOC's control record. If it finds a matching entry, ALE R/3 to eLink assigns the IDOC to the destination queue space and queue specified in the entry. ALE R/3 to eLink also processes the IDOC according to the Compress flag and the Target ID settings in the entry.

Compressing R/3 to eLink IDOCs

You use the Compress column in the destination map file to combine IDOC messages with matching logical system IDs and IDOC message types into an IDOC message containing multiple IDOCs. For each matching entry, if the Compress column is “Y”, then ALE R/3 to eLink aggregates this IDOC with other matching IDOCs in a single, larger IDOC message that it then assigns to the appropriate target queue. If the Compress column is “N”, then ALE R/3 to eLink queues each IDOC message separately.

Setting the Data-Dependent Routing Value

You use the TargetId column in the destination map file to associate an IDOC with a data-dependent routing (DDR) value. For each matching entry, ALE R/3 to eLink adds the specified TargetId value in the CR3_TARGET_ID FML32 field of the IDOC message. If this DDR is unspecified, or if no matching entry is found in the destination map file, then ALE R/3 to eLink encodes the default DDR value, which is defined in the CR3_ALE_DEFAULT_TARGET_ID environment variable.

Note: To use DDR, the DDR value must match the field ID configured in the ROUTING section of the `cr3_ale.ubb` configuration file.

See “Setting the Data-Dependent Routing Value” in Chapter 6, “Configuring ALE Integration,” in this guide for more information. See your TUXEDO documentation for more information about data-dependent routing.

Examples of Using a Destination Map File

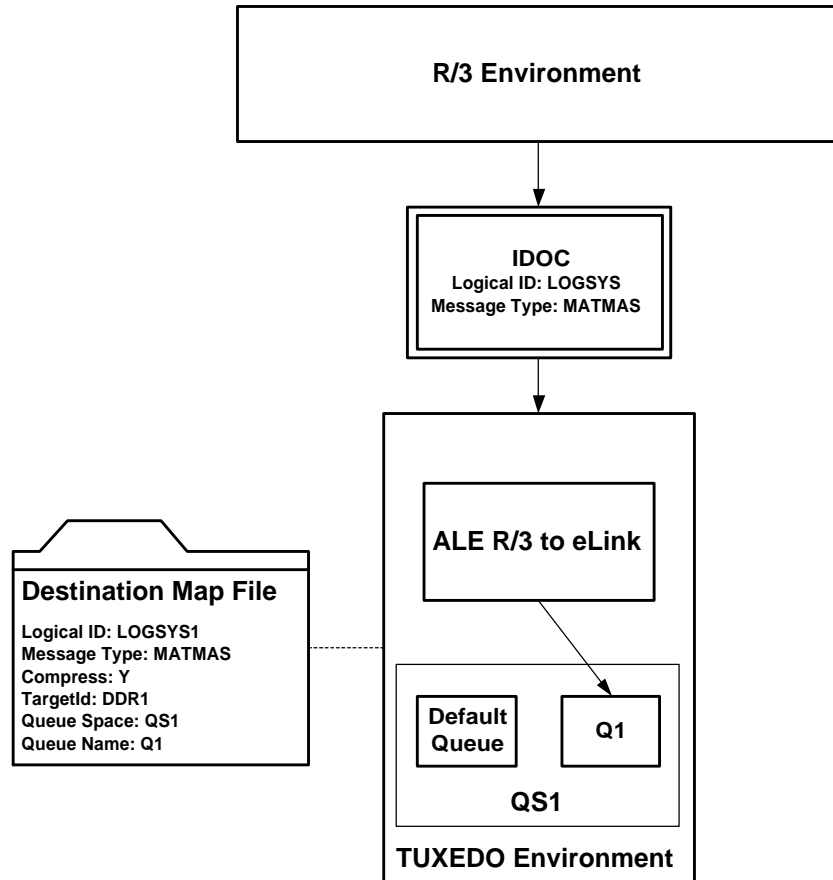
The first example shows how ALE R/3 to eLink processes an IDOC packet that contains a single IDOC. Suppose the destination map file contains the entry shown in Table 2-5:

Table 2-5 Settings in a Sample Destination Map File

LogicalId	MsgType	Compress	TargetId	QueueSpace	QueueName
LOGSYS1	MATMAS	Y	DDR1	QS1	Q1

The control record in an R/3 to eLink IDOC specifies a target logical ID of “LOGSYS” and a message type of “MATMAS”. Figure 2-12 shows how ALE R/3 to eLink would process this IDOC packet according to the settings in the destination map file:

Figure 2-12 Queuing a Single IDOC According to the Destination Map File



In this scenario, ALE R/3 to eLink finds the matching entry in the destination map file and sends the IDOC message to the Q1 queue in queue space QS1. ALE R/3 to eLink encodes the specified TargetId value (“DDR1”) in the CR3_TARGET_ID FML32 field. Compression does not apply in this case because the IDOC packet contained only one IDOC.

The second example shows how ALE R/3 to eLink processes an IDOC packet that contains multiple IDOCs. Table 2-6 shows the destination map file entries:

Table 2-6 Settings in a Sample Destination Map File

LogicalId	MsgType	Compress	TargetId	QueueSpace	QueueName
LOGSYS1	MATMAS	Y	DDR1	QS1	Q1
LOGSYS2	MATMAS	Y	DDR2	QS1	Q2
LOGSYS3	DEBMAS	N	DDR3	QS2	Q3
LOGSYS4	CREMAS	Y	DDR4	QS3	Q4

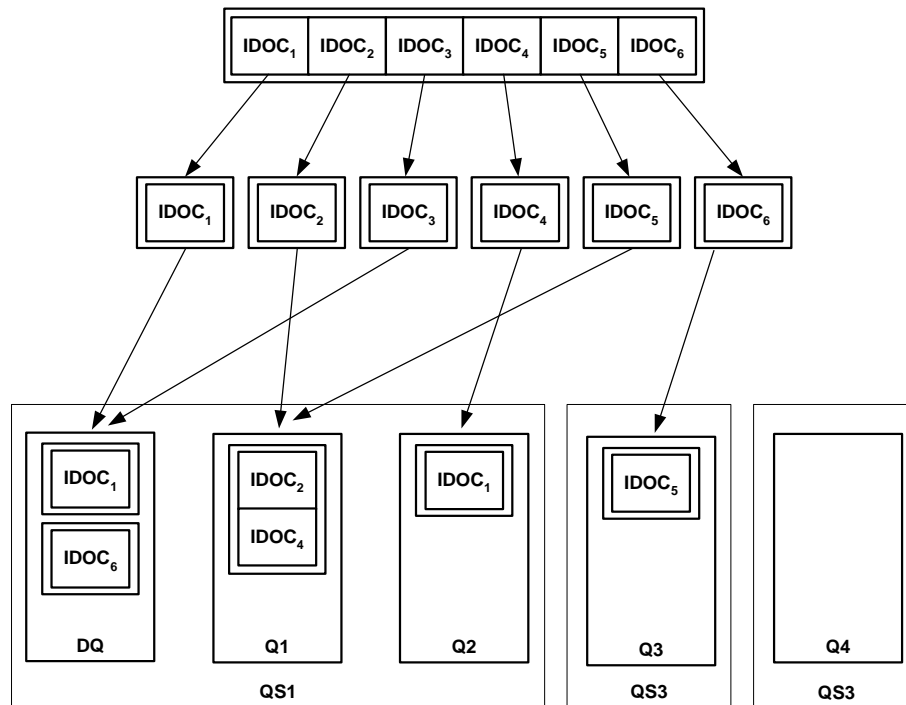
An R/3 to eLink IDOC packet contains six IDOCs with the following settings in the control record of each IDOC, as shown in Table 2-7:

Table 2-7 Sample IDOC Packet Containing Multiple IDOCs

IDOC	LogicalId	MsgType
IDOC ₁	LOGSYS1	CREMAS
IDOC ₂	LOGSYS1	MATMAS
IDOC ₃	LOGSYS3	MATMAS
IDOC ₄	LOGSYS2	MATMAS
IDOC ₅	LOGSYS1	MATMAS
IDOC ₆	LOGSYS3	DEBMAS

Figure 2-13 shows how ALE R/3 to eLink would process this IDOC packet according to the settings in the destination map file:

Figure 2-13 Queuing Multiple IDOCs According to the Destination Map File



ALE R/3 to eLink splits the IDOC packet into individual IDOCs and queues each IDOC in the following manner:

- ALE R/3 to eLink assigns IDOC₁ and IDOC₃ to the default queue because no matching entry was found in the destination map file.
- ALE R/3 to eLink assigns IDOC₂, IDOC₄, IDOC₅, and IDOC₆ to their respective queues.
- ALE R/3 to eLink compresses IDOC₂ and IDOC₅ into a single IDOC packet in Q1 because Compress="Y" in the matching entry in the destination map file.

Managing Data-Depending Routing (DDR)

You can configure the default data-dependent routing (DDR) value that ALE R/3 to eLink assigns to each IDOC message (in the CR3_TARGET_ID field) that it queues. The behavior of ALE R/3 to eLink depends on whether a DDR value is defined in the destination map file:

- If no destination map file is used, or if no matching entry is found in the destination map file for an IDOC, then ALE R/3 to eLink assigns a default DDR value to the IDOC messages (CR3_ALE_DEFAULT_TARGET_ID environment variable).
- If a matching entry is found in the destination map file for an IDOC, then ALE R/3 to eLink assigns the DDR value (TargetId) from the map file.

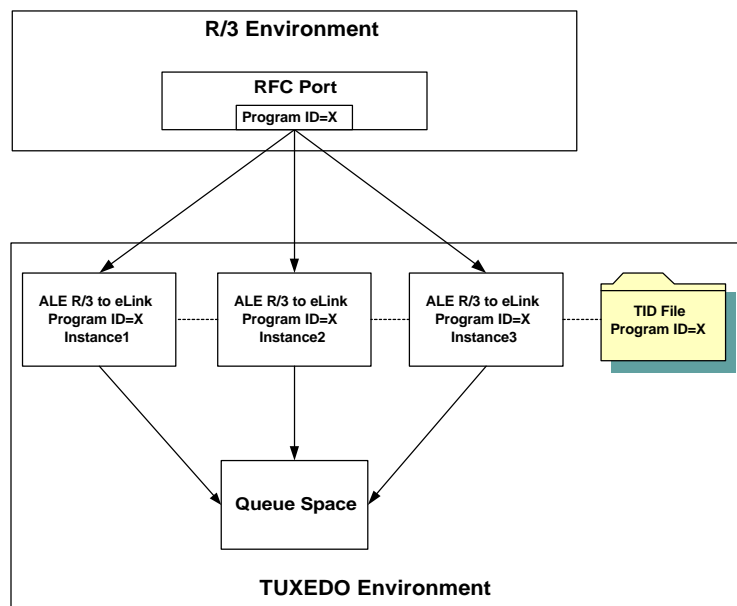
See “Setting the Default Data-Dependent Routing Value” and “Setting Data-Dependent Routing Values in the Map File” in Chapter 6, “Configuring ALE Integration,” in this guide for instructions. See your TUXEDO documentation for more information about data-dependent routing.

Load Balancing High Volumes of R/3 to eLink IDOCs

Multiple instances of ALE R/3 to eLink can register using the same program ID. For deployments that involve high volumes of IDOC packets, you can enhance system performance by balancing the load across multiple instances of ALE R/3 to eLink. Instances that register under the same program ID must also share the same TID file. Figure 2-14 shows multiple instances of ALE R/3 to eLink listening for IDOCs on the same program ID and sharing the same TID file:

Figure 2-14 Multiple Instances of ALE R/3 to eLink Sharing the Same Program ID

ALE Outbound -- Multiple Instances



The number of ALE R/3 to eLink instances should match the anticipated number of IDOC packets that R/3 sends concurrently to port. For example, if R/3 sends five IDOC packets concurrently to port during peak loads, you should load five instances of ALE R/3 to eLink.

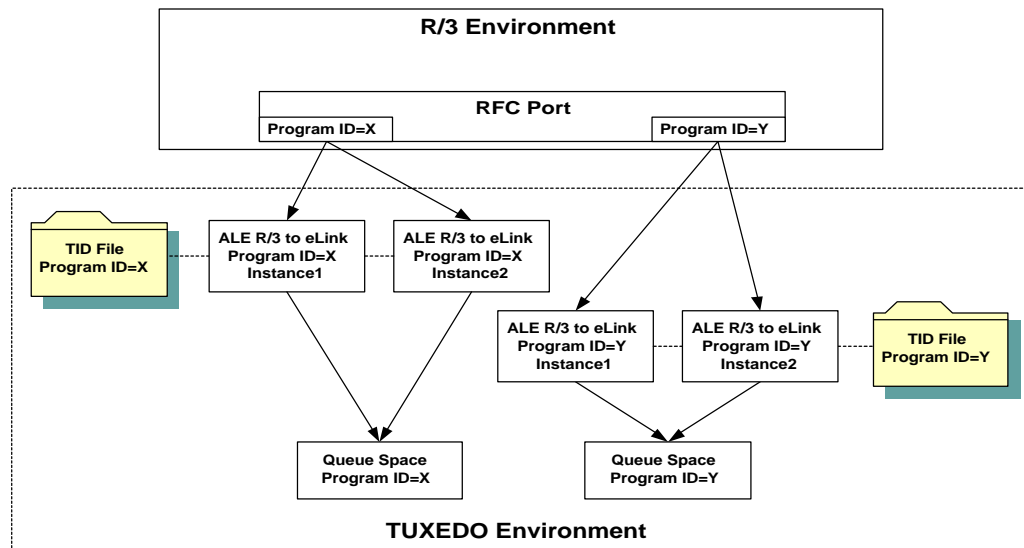
See “Configuring Load Balancing” in Chapter 6, Configuring ALE Integration,” in this guide.

Registering Multiple Program IDs

If R/3 is configured to send R/3 to eLink IDOCs to different program IDs, you can configure ALE R/3 to eLink to handle these IDOCs by running multiple instances of ALE R/3 to eLink using different program IDs. You must make sure that all instances sharing the same program ID also share the same TID file, and that all instances sharing the same TID file also share the same program ID. Instances that register under different program IDs must *not* share the same TID file.

Figure 2-15 shows two groups of instances of ALE R/3 to eLink, each of which is listening for IDOCs on a shared program ID and sharing the same TID file:

Figure 2-15 Multiple Instances of ALE R/3 to eLink Using Different Program IDs



See “Configuring Load Balancing for R/3 to eLink” in Chapter 6, Configuring ALE Integration,” in this guide.

Managing Transactional Integrity for R/3 to eLink IDOCs

ALE R/3 to eLink manages transactional integrity for R/3 to eLink IDOCs to ensure that an IDOC packet has been queued successfully. ALE R/3 to eLink uses a TID log file to track the transaction IDs (TIDs) associated with the IDOC packets it processes to ensure that it queues an IDOC packet from R/3 exactly once. See “Transaction IDs (TIDs)” later in this topic for an introduction to TIDs.

About the TID Log File Used for R/3 to eLink IDOCs

ALE R/3 to eLink uses a TID file to track the IDOC packets it processes to ensure that it queues an IDOC packet once and only once. The R/3 system assigns a TID to each R/3 to eLink IDOC packet.

The TID file for ALE R/3 to eLink is a log of all the TIDs that ALE R/3 to eLink has received and processed. Each row in the TID file represents the TID for a separate IDOC packet and contains three fixed-position columns of information:

Column	Description
Date-Time Stamp	Date and time at which the TID log file was last updated.
TID	TID that R/3 assigned to the IDOC packet.
State	The processing state. One of the following strings: <ul style="list-style-type: none"> ■ CREATED indicates that ALE R/3 to eLink has received the TID from R/3. ■ EXECUTED indicates that ALE R/3 to eLink has queued the IDOC message with the TID and has committed the transaction. ■ ROLLBACK indicates that ALE R/3 to eLink has rolled back the IDOC packet from the queue. ■ CONFIRMED indicates that ALE R/3 to eLink has confirmed that the IDOC message has been queued and the transaction has been committed.

The following example shows a sample TID file for ALE R/3 to eLink:

```
Tue Apr 27 14:27:36 1999 0A0201FD03F937262C5B0001 CONFIRMED
```

2 Integrating with ALE

Tue Apr 27 14:29:38 1999 0A0201FD03E937262CD70004 CONFIRMED

Tue Apr 27 14:46:56 1999 0A0201FD03F9372630E60009 CONFIRMED

Tue Apr 27 15:50:21 1999 0A0201FD03E837263F98003F CONFIRMED

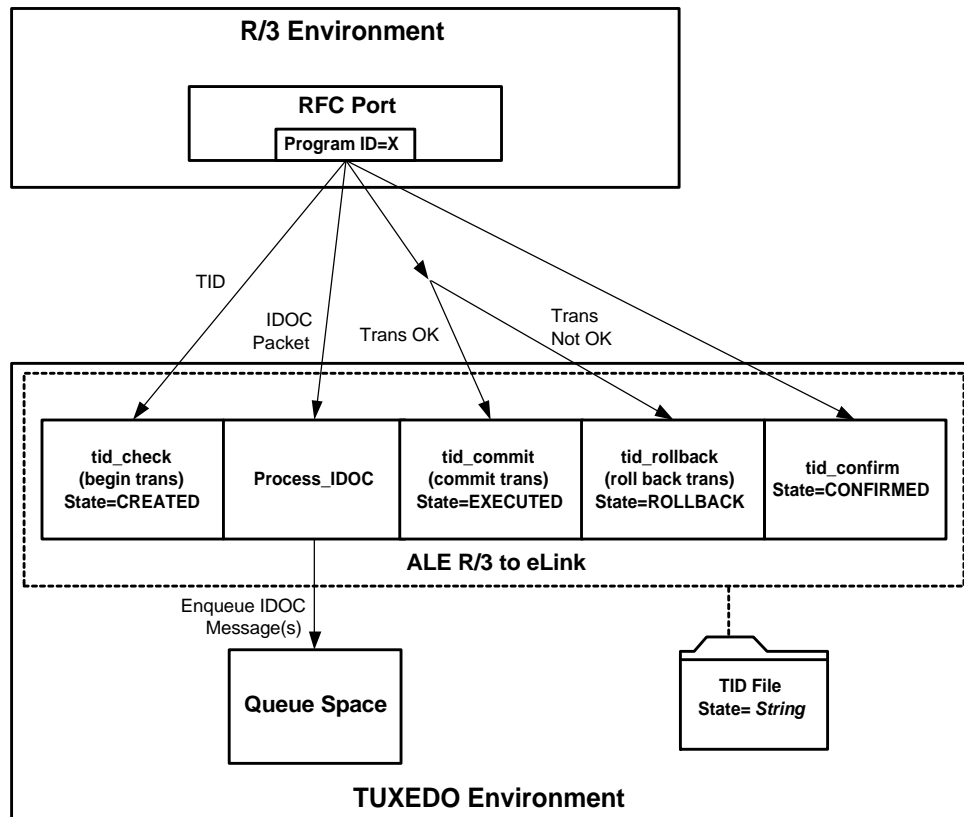
The CR3_ALEOUT_TID_FILE environment variable specifies the location of the TID log file for ALE R/3 to eLink. See “Setting Environment Variables for ALE R/3 to eLink” in Chapter 6, “Configuring ALE Integration,” in this guide for more information.

Use the cr3tidmanager program to manage the size and number of entries kept in TID files. See “Configuring the TID File Manager” in Chapter 6, “Configuring ALE Integration,” in this guide.

Processing TIDs with R/3 to eLink IDOCs

Figure 2-16 shows how ALE R/3 to eLink uses the TID log file to manage transactional integrity for R/3 to eLink IDOCs:

Figure 2-16 TID Processing for ALE R/3 to eLink



The information flow proceeds in the following sequence:

1. R/3 sends a TID to an instance of ALE R/3 to eLink that is registered on the matching program ID.
2. ALE R/3 to eLink receives the TID and checks the TID file to determine whether it has previously received this TID from R/3. If the TID is not found in the TID file, then ALE R/3 to eLink appends an entry to the TID file, specifying the date-time stamp, TID, and the State (CREATED). ALE R/3 to eLink returns a code to R/3 indicating whether the TID was found, and the TID state determines whether R/3 continues processing.
3. If R/3 continues processing, ALE R/3 to eLink starts a new transaction.

4. R/3 sends the IDOC packet associated with the TID to the same instance of ALE R/3 to eLink.
5. ALE R/3 to eLink receives the IDOC packet and processes the IDOC data according to the way that ALE R/3 to eLink is configured (such as splitting IDOC packets, making routing decisions based on a map file, and so on). ALE R/3 to eLink encodes the IDOC data in one or more FML32 message buffers and queues the message(s) into one or more queues.
6. After processing the IDOC data, ALE R/3 to eLink returns success or a SAP exception (if, for example, the target queue is full) to R/3.
7. Based on the status returned from ALE R/3 to eLink, R/3 instructs the same instance of ALE R/3 to eLink to commit or roll back the transaction:
8. ALE R/3 to eLink takes one of the following actions:
 - ALE R/3 to eLink commits the transaction and updates the date-time stamp and State (EXECUTED) in the TID file.
 - ALE R/3 to eLink calls rolls back the transaction and updates the date-time stamp and State (ROLLBACK) in the TID file.
9. If the transaction is successfully committed, ALE R/3 to eLink updates the date-time stamp and State (CONFIRMED) in the TID file.

Handling Problems with R/3 to eLink IDOCs

ALE R/3 to eLink uses TUXEDO's transaction management capabilities to ensure transactional integrity for R/3 to eLink IDOCs. Figure 2-8 lists problems that can occur with R/3 to eLink IDOCs:

Table 2-8 Handling Problems with R/3 to eLink IDOCs

Problem	Description
Unable to Lock the TID File	The TID file might be locked by another instance of ALE R/3 to eLink or the TID File Manager. ALE R/3 to eLink retries the lock attempt. After a configurable number of retry attempts, ALE R/3 to eLink returns a lock error to R/3. R/3 then attempts to retry the operation until it succeeds or stops trying.

Table 2-8 Handling Problems with R/3 to eLink IDOCs

Problem	Description
Unable to Update the TID File	The file might be corrupted. If ALE R/3 to eLink can lock the TID file but cannot update it, ALE R/3 to eLink retries the lock attempt. After a configurable number of retry attempts, ALE R/3 to eLink returns a lock error to R/3. R/3 then attempts to retry the operation until it succeeds or stops trying.
ALE R/3 to eLink cannot queue an IDOC message(s)	One or more target queues might be full. ALE R/3 to eLink returns a SAP exception to R/3, and R/3 instructs ALE R/3 to eLink to roll back the transaction. R/3 will subsequently resubmit the IDOC packet to ALE R/3 to eLink.

2 *Integrating with ALE*

3 Preinstallation Tasks

This topic describes important information that you need to review before installing BEA eLink Adapter for R/3 ALE. It includes the following main sections:

- System Requirements
- Determining the Installation Node
- TUXEDO Requirements

System Requirements

This document is validated for the following versions of the layered products:

Table 3-1 Supported Platforms

Product	Version(s)
SAP R/3	3.1H
TUXEDO	6.5
HP-UX	10.20, 11.00
Solaris	2.6, 2.7
AIX	4.3
Windows NT	4.0 (Service Pack 4)

Determining the Installation Node

Where you install eLink Adapter for R/3 ALE depends on the node configuration:

- **Single Node.** If the R/3 environment is a single node configuration, then you should install BEA eLink Adapter for R/3 ALE on that node.
- **Multiple Nodes.** If the R/3 environment is a multiple node configuration, then you should install BEA eLink Adapter for R/3 ALE on one or more of the application servers. BEA eLink Adapter for R/3 ALE must *not* be installed on the database server.

BEA eLink Adapter for R/3 ALE interacts with R/3 using the SAP RFC library. The RFC library uses IP sockets to communicate with the R/3 application servers. Installing BEA eLink Adapter for R/3 ALE on an R/3 application server node minimizes the risk of failure in this IP socket connection. If BEA eLink Adapter for R/3 ALE is installed on an R/3 application server node, it is recommended that no other TUXEDO applications run on that same node, in order to reduce the workload on the node.

However, BEA eLink Adapter for R/3 ALE can be installed on a node other than an R/3 application server. In this situation, the IP socket connection will transparently extend across the network to the R/3 application server.

TUXEDO Requirements

BEA eLink Adapter for R/3 ALE requires that you properly install and configure TUXEDO:

- **Installation Node.** TUXEDO must be installed on the node running BEA eLink Adapter for R/3 ALE. For installation instructions, see the *BEA TUXEDO Administrator's Guide* for your platform.
- **TUXEDO Configuration.** After you install BEA eLink Adapter for R/3 ALE, you must configure TUXEDO according to the instructions described in Chapter 5, "TUXEDO Initialization File." BEA eLink Adapter for R/3 ALE provides a semi-configured TUXEDO initialization file, which requires only minimal configuration for the TUXEDO environment.

3 *Preinstallation Tasks*

4 Installing BEA eLink Adapter for R/3 ALE

This topic describes how to install BEA eLink Adapter for R/3 ALE ALE on UNIX and Windows NT systems. It covers the following main sections:

- Installing on a UNIX System
- Installing on a Windows NT System

Installing on a UNIX System

Installing eLink Adapter for R/3 ALE on UNIX systems requires the three main steps:

- Creating the conr3 User
- Logging In as con3
- Running install.sh

Creating the conr3 User

A UNIX user with non-administrative privileges should be created on the R/3 application server to provide a context in which to run both TUXEDO and BEA eLink Adapter for R/3 ALE. This user should be used to run TUXEDO and BEA eLink Adapter for R/3 ALE *only* so that processes associated with TUXEDO and BEA eLink Adapter for R/3 ALE can be identified easily.

BEA suggests that you use `conr3` for the user name. This document refers to this UNIX user as `conr3`. If your organization uses a different user name, then substitute your user name whenever you encounter a reference to `conr3` in this document.

Note: BEA strongly recommends that you do *not* use the UNIX R/3 administration user (`nnnadm`, where `nnn` represents the R/3 system identifier) to run TUXEDO and BEA eLink Adapter for R/3 ALE.

Logging In as con3

After creating the `conr3` user name, log in as that user.

Running install.sh

Run the `install.sh` script to install the BEA eLink Adapter for R/3:

1. Insert the CD-ROM in the CD-ROM drive.

If your system is not already configured to access data from a CD-ROM directory, you will need to create a mount directory (for example, `mkdir /cdrom`) and mount the CD-ROM device (with the `mount` command).

Change directories to your CD-ROM directory using the following command:

```
cd /cdrom
```

2. Type `ls` to view the contents of the CD.

The CD should contain the following files and directories:

```
HPUX/   IBM/   install.sh  readme.txt  SUN5X/  WINNT/
```

3. Start the installation by typing the following at the command line prompt:

```
sh ./install.sh
```

Press **Enter**.

This invokes the installation script.

The UNIX system installation script provides a set of step-by-step instructions to help you quickly install the BEA eLink Adapter for R/3. This script lets you specify your platform, operating system, and the directory where you want to install. The installation script prompts you through the entire installation process. You can cancel the installation at any time by pressing **CTRL-C** simultaneously.

4. Type the number that corresponds to the name of the operating system you are using (for example, if using HPUX1100, type 1).

```
01) HP/HPUX1020    02) HP/HPUX1100    03) IBM/AIX414
04) IBM/AIX421    05) IBM/AIX43      06) SUN5X/SOL251
07) SUN5X/SOL26
```

```
Install which platform's files? [01-5, q to quit, 1 for list]: 2
```

Press **Enter**.

5. You are prompted to confirm your choice. If correct, type **y** for “yes,” or **n** for “no” or **q** to “quit.” Press **Enter**.

```
** You have chosen to install from HP/HPUX1100 **
```

```
BEA eLink Adapter for R/3 ALE Release 1.5
```

```
This directory contains the BEA eLink Adapter for R/3 ALE for
HP-UX 11.0 on 9000/800 series.
```

```
Is this correct? [y,n,q]: y
```

6. The script indicates which packages are available for the chosen operating system. Indicate which package to install and press **Enter**.

```
The following packages are available:
```

```
1 eLinkR3ALE BEA eLink Adapter for R/3 ALE
```

```
Select the package(s) you wish to install (or 'all' to install
all packages) (default: all) [?,??,q]: 1
```

The following copyright information is displayed about the product you are about to install:

4 Installing BEA eLink Adapter for R/3 ALE

```
BEA eLink Adapter for R/3 ALE Release BEA eLink Adapter for R/3
ALE Release 1.5
```

```
Copyright (c) 1999 BEA Systems, Inc.
```

```
All Rights Reserved.
```

```
Distributed under license by BEA Systems, Inc.
```

```
BEA and eLink are trademarks of BEA Systems, Inc.
```

7. You are prompted for the directory location where you want to install the product. Type the install directory and press **Enter**.

Note: It is recommended that you install BEA eLink Adapter for R3 under a directory named `eLink`. The host machine where you install and the prefix path to the `eLink` directory is up to you. For this example, we install on `/home/miller/elink/`.

```
Directory where eLink files are to be installed [?,q]:
/home/miller/elink
```

8. Watch the screen messages to verify that the installation is successful. The messages will be similar to the following.

```
Creating /home/miller/elink
Using /home/miller/elink as the eLink base directory
Creating /home/miller/elink/adapters
Creating /home/miller/elink/adapters/sapr3_ale
Determining if sufficient space is available ...
5320 blocks are required
720596 blocks are available to
/home/miller/elink/adapters/sapr3_ale
Using /home/miller/elink/adapters/sapr3_ale
as the eLink Adapter for R/3 ALE install directory
Unloading /cdrom/HP/HPUX1100/ELINKR3/ELINKR3.Z ...
bin/cr3alein
bin/cr3aleout
bin/cr3tidmanager
config/cr3_queues.sh
config/cr3_tlog.sh
```



```

config/cr3aleout.sh
config/setenv.sh
config/cr3alein.env
config/cr3_ale.ubb
config/cr3_ale.fml
config/sideinfo
5320 blocks
... finished
Installation of BEA eLink Adapter for R/3 ALE was successful
The following packages are available:
  1  eLinkR3ALE  BEA eLink Adapter for R/3 ALE
9.  When the installation completes successfully, exit the install script. (Or proceed
    with another installation, as appropriate.)

    Select the package(s) you wish to install (or 'all' to install
    all packages) (default: all) [?,??,q]: q

    Please don't forget to fill out and send in your registration
    card #

```

Table 4-1 describes the files and directories that are installed for BEA eLink Adapter for R3 on a UNIX system.

Table 4-1 Directory Structure of BEA eLink Adapter for R/3 ALE on a UNIX System

File or Directory Name	Description
bin/cr3alein	ALE eLink to R/3 executable
bin/cr3aleout	ALE R/3 to eLink executable
bin/cr3tidmanager	TID manager executable
config/cr3_queues.sh	Semi-configured script for creating the TUXEDO queue space and queues
config/cr3_tlog.sh	Semi-configured script for creating the TUXEDO TLOG file

4 Installing BEA eLink Adapter for R/3 ALE

Table 4-1 Directory Structure of BEA eLink Adapter for R/3 ALE on a UNIX System

File or Directory Name	Description
config/cr3aleout.sh	Semi-configured startup script for ALE R/3 to eLink
config/setenv.sh	Semi-configured setup file for generic eLink for R/3 environment variables
config/cr3alein.env	Environment file for ALE eLink to R/3
config/cr3_ale.ubb	TUXEDO UBB configuration file
config/cr3_ale.fml	CR3 FML field table
configsideinfo	Example R/3 sideinfo file

Installing on a Windows NT System

The BEA eLink Adapter for R/3 product install program for the Windows NT platform is located in the `winnt\setup.exe` file on the CD-ROM.

To install the product, insert the CD into the CD-ROM drive and double click on the `winnt\setup.exe` file to launch the installation wizard.

When the installation program launches, follow the prompts shown in the windows as described.

1. The Welcome window describes the product you are about to install. Click **Next** to continue (or click **Cancel** to cancel the installation process).

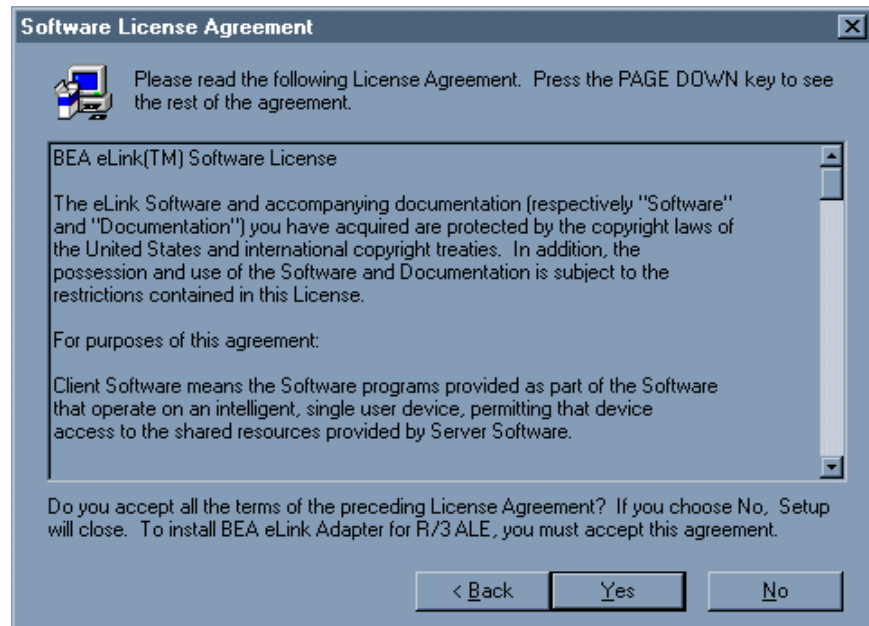
Figure 4-1 Welcome to BEA eLink Adapter for R3 Setup Program



2. In order to proceed with the installation, you must accept the license agreement. Click **Yes** to accept (or click **No** to cancel the installation process).

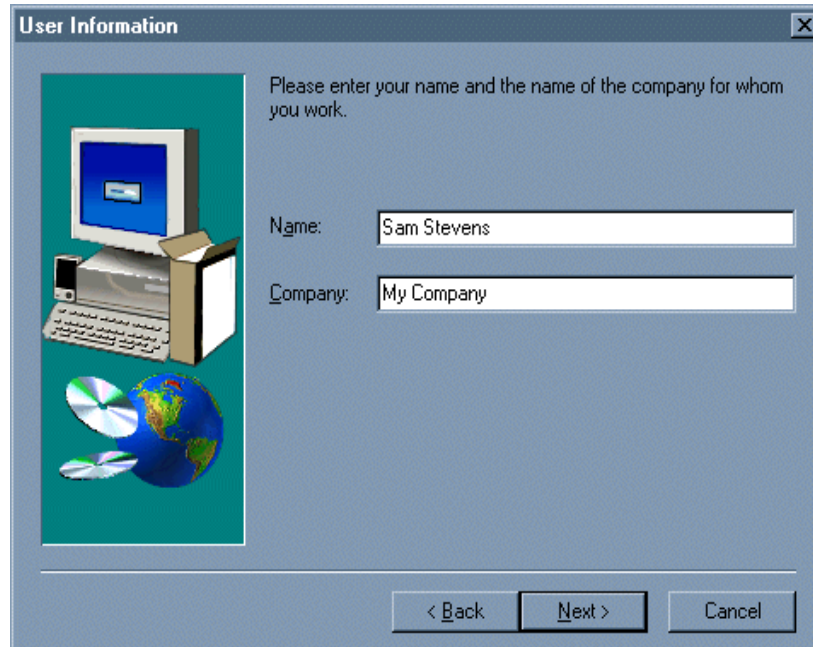
4 Installing BEA eLink Adapter for R/3 ALE

Figure 4-2 License Agreement



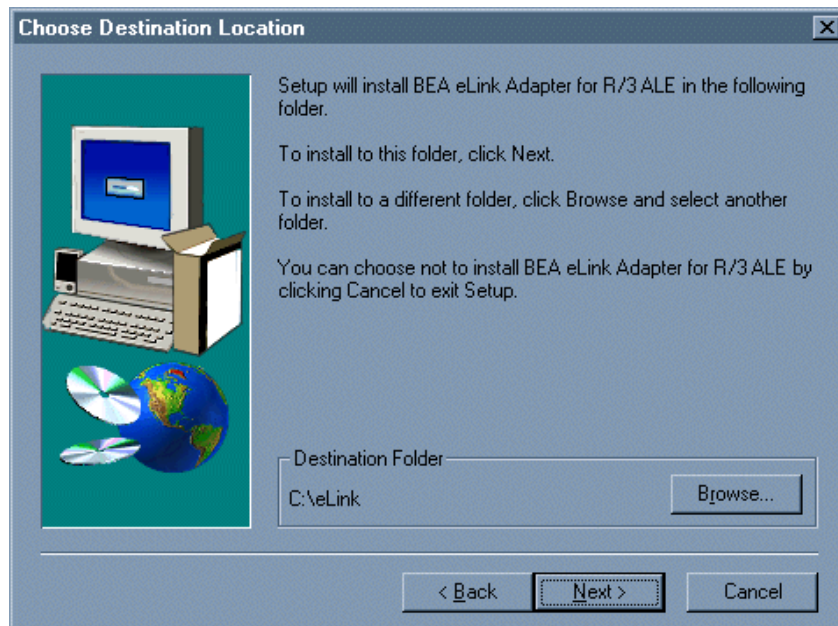
3. After you accept the license agreement, the User Information window is displayed. Provide the requested information, and click **Next** to continue (or **Back** to return to a previous window).

Figure 4-3 Provide User Information



4. The path shown for the “Destination Folder” on the Choose Destination Location window is the location where the BEA eLink product(s) will be installed. The default Destination Folder is *YourDrive:\eLink* as shown in Figure 4-4. To accept the default, click **Next** to continue. Otherwise, use the file browser to choose a different destination folder as explained below.

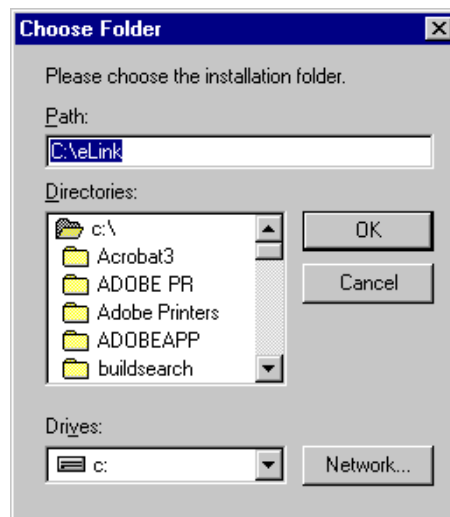
Figure 4-4 Choose Directory Location for Product Installation



If you want to install the BEA eLink product in a folder other than the default, click the **Browse** button to get the file browser. Use the file browser to locate the directory in which you want to install the BEA eLink product and click **OK** to specify this new path name as the Destination Folder.

Note: It is recommended that you install BEA eLink Adapter for R/3 under a directory named eLink. The host machine where you install and the prefix path to the eLink directory is up to you. For this example, we install on C:\eLink\.

Figure 4-5 File Browser for Choosing an Install Location

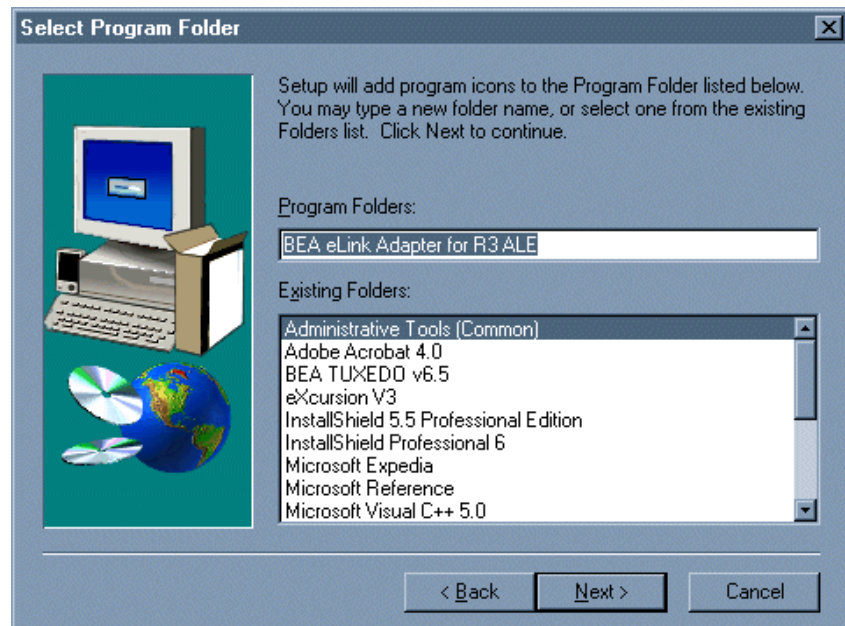


If you do change the Destination Folder to something other than the default, you should now see the new path name reflected in the Choose Folder window.

5. Select the program folder where you want to store the startup icon for the BEA eLink Adapter for R3 product. The default folder is already chosen for you. If you want to choose a different folder, do so. Click **Next** to continue (or **Back** to return to a previous window).

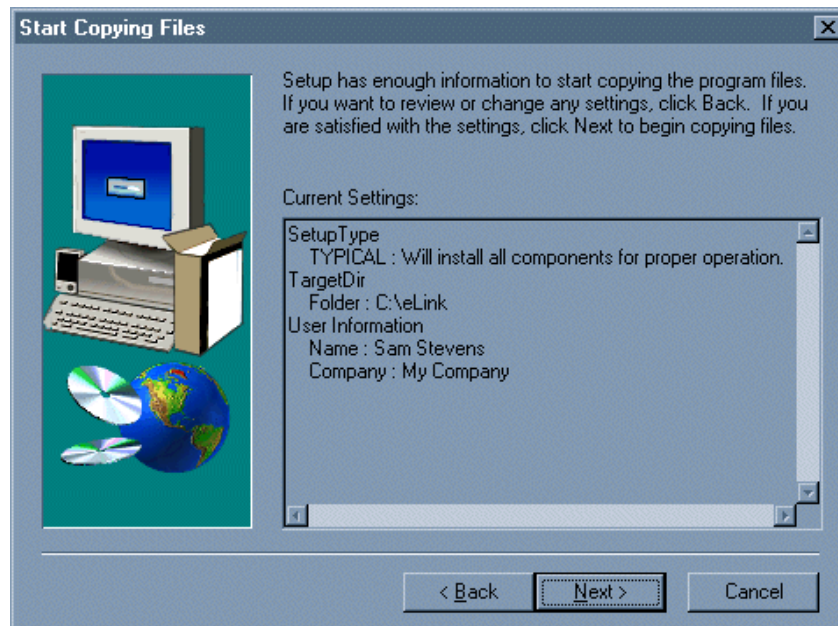
4 Installing BEA eLink Adapter for R/3 ALE

Figure 4-6 Select a Program Folder



6. If you are satisfied with the current settings, click **Next** to proceed with the installation. (Otherwise, click **Back** to return to a previous window.)

Figure 4-7 Current Settings



7. When all files have been successfully copied to the destination folders, the Setup Complete window is displayed. Click **Finish** to exit the installation.

Figure 4-8 Setup Complete



Table 4-2 describes the files and folders that are installed for BEA eLink Adapter for R3 on a Windows NT system.

Table 4-2 Directory Structure of BEA eLink Adapter for R/3 ALE on Windows NT

File or Folder Name	Description
eLink\bin\	Directory containing executables
eLink\bin\cr3alein.exe	ALE eLink to R/3 executable
eLink\bin\cr3aleout.exe	ALE R/3 to eLink executable
eLink\bin\cr3rfcin.env	Environment file for RFC eLink to R/3
eLink\bin\cr3tidmanager.exe	TID manager executable
eLink\bin\librfc.dll	R/3 RFC Dynamic Link Library
eLink\config	Directory containing configuration files

Table 4-2 Directory Structure of BEA eLink Adapter for R/3 ALE on Windows NT

File or Folder Name	Description
eLink\config\cr3_queues.nt	Semi-configured makefile for creating TUXEDO queue space and queues
eLink\config\cr3_tlog.nt	Semi-configured makefile for creating the TLOG file
eLink\config\setenv.bat	Semi-configured setup file for generic eLink for R3 environment variables
eLink\config\cr3aleout.bat	Semi-configured startup script for ALE R/3 to eLink
eLink\config\cr3alein.env	Environment file for ALE eLink to R/3
eLink\config\cr3rfcin.exe	RFC eLink to R/3 executable
eLink\config\cr3_ale.ubb	TUXEDO UBB configuration file
eLink\config\cr3_ale.fml	CR3 FML field table
eLink\config\sideinfo	Example R/3 sideinfo file

4 *Installing BEA eLink Adapter for R/3 ALE*

5 TUXEDO Initialization File

BEA eLink Adapter for R/3 ALE requires some configuration of the TUXEDO initialization file. For configuration instructions, see the *BEA TUXEDO Administrator's Guide* for your platform.

BEA eLink Adapter for R/3 ALE provides a semi-configured TUXEDO initialization file, `cr3_ale.ubb`, in the `config` sub-directory of the BEA eLink Adapter for R/3 ALE installation directory. This `cr3_ale.ubb` file is configured for the BEA eLink Adapter for R/3 ALE servers and it requires further configuration for integration into the global TUXEDO environment. For example, in the `GROUPS` section of your UBB file, you must define the server group to which the ALE eLink to R/3 belongs, such as `CR3`. See Appendix A, “Sample `cr3_ale.ubb` File,” for a listing of this file.

Note: ALE eLink to R/3 and ALE R/3 to eLink use TUXEDO transactions. For these components, you need to create a transaction log (TLOG) using TUXEDO administrative tools. See the *BEA TUXEDO Administrator's Guide* for your platform.

5 *TUXEDO Initialization File*

6 Configuring ALE Integration

This topic describes how to configure your BEA eLink Adapter for R/3 ALE installation for integration with SAP Application Linking and Embedding (ALE) technology. It includes the following main sections:

- Configuring the ALE eLink to R/3 Server
- Configuring the ALE R/3 to eLink Client
- Configuring Queues
- Configuring the TID File Manager

Note: You need to complete the tasks in this section only if you require application integration with ALE.

Using BEA eLink Adapter for R/3 ALE, TUXEDO applications can exchange SAP Intermediate Documents (IDOCs) with R/3 systems. See Chapter 2, “Integrating with ALE,” in this guide for more information about ALE integration.

Configuring the ALE eLink to R/3 Server

The following topics describe how to configure the ALE eLink to R/3 server:

- Configuring the UBB File for ALE eLink to R/3
- Setting Environment Variables for ALE eLink to R/3
- Configuring Load Balancing for eLink to R/3 IDOCs

In addition, see the following topics for further configuration instructions:

- Setting Up Queues for eLink to R/3 IDOCs
- “Configuring eLink to R/3 Connections” in Chapter 7, “Configuring R/3 Connections”

ALE eLink to R/3 is a TUXEDO server that submits IDOC packets to R/3 via tRFC. ALE eLink to R/3 receives IDOC packets forwarded from a TUXEDO queue, encodes the IDOC data for R/3, and submits the message to R/3 via tRFC. See “Processing eLink to R/3 IDOCs” in Chapter 2, “Integrating with ALE,” in this guide for more information about ALE eLink to R/3.

Configuring the UBB File for ALE eLink to R/3

ALE eLink to R/3 is a standard TUXEDO server that is defined in the UBB file. You must configure this file to specify the server group, server, environment file, and services associated with the ALE eLink to R/3 server. See Appendix A, “Sample cr3_ale.ubb File,” for a listing of the sample UBB file that comes with BEA eLink Adapter for R/3 ALE.

Defining the Server Group

First define the server group (such as CR3) to which the ALE eLink to R/3 belongs, as described in Chapter 5, “TUXEDO Initialization File.”

Defining the ALE eLink to R/3 Server and Environment File

In the **SERVICES** section of your UBB file, define the ALE eLink to R/3 server, specifying its server group and server ID, as shown in the following example:

```
cr3alein
  SRVGRP=CR3 SRVID=3
```

Define the environment file for the ALE eLink to R/3 Server, as shown in the following example:

```
ENVFILE="cr3alein.env"
```

Defining ALE eLink to R/3 Services

In the **SERVICES** section of your UBB file, define the **CR3_SUBMIT** and **CR3_IDOC_IN** services, as shown in the following example:

```
CR3_SUBMIT
CR3_IDOC_IN
```

Setting Environment Variables for ALE eLink to R/3

The ALE eLink to R/3 server requires environment variables that specify connection information and the behavior of the server to R/3. These environment variables are defined in an environment file that is specified in the UBB file as the ENVFILE parameter for the server.

Environment Variables for ALE eLink to R/3

Configure the required and optional environment variables for ALE eLink to R/3 as shown in Table 6-1:

Table 6-1 Environment Variables for the ALE eLink to R/3 Server

Category / Variable	Set To	Description
Connecting To R/3		
CR3_DESTINATION	System Name	Destination in the sideinfo that defines the R/3 application server to connect to. Default is undefined. <i>Required.</i>
CR3_CLIENT	Client Number	R/3 login client number. Default is undefined. <i>Required.</i>
CR3_USER	User	R/3 login user. Default is undefined. The user must be of type CPIC. <i>Required.</i>
CR3_PASSWORD	Password	R/3 login user password. Default is undefined. <i>Required.</i>
CR3_LANGUAGE	E	R/3 login language. Default is E for English. <i>Required.</i>
SIDE_INFO	Path and File	Full path to the sideinfo file. <i>Required.</i>
Managing the TID File		
CR3_ALEIN_TID_FILE	Path and File	Full path and filename for the ALE eLink to R/3 TID file, <i>without</i> the .tid file extension. <i>Required.</i>

Table 6-1 Environment Variables for the ALE eLink to R/3 Server

Category / Variable	Set To	Description
CR3_ALEIN_TID_PER_IDOC	Y or N	Specify “Y” to split IDOC packets containing multiple IDOCs and pass them as individual IDOCs to R/3. Specify “N” (the default) to pass multiple IDOCs to R/3 in one batch. <i>Optional.</i>
Managing Queues		
CR3_ALEIN_IDOCIN_QUEUE	Queue Name	Name of the queue for messages containing processed IDOCs and their corresponding TIDs. By default, should be set to CR3_IDOC_IN. TMQFORWARD dequeues messages from this queue and forwards them to the CR3_IDOC_IN service for submission to R/3. <i>Required.</i>
CR3_ALEIN_IDOCIN_QUEUE_SPACE	Queue Space	Queue space that contains the CR3_IDOC_IN queue. <i>Required.</i>
CR3_ALEIN_ERROR_QUEUE	Queue Name	Name of the queue for messages containing IDOCs that the CR3_SUBMIT has received and rejected due to IDOC format or consistency errors. By default, should be set to CR3_ERROR. <i>Required.</i>
CR3_ALEIN_ERROR_QUEUE_SPACE	Queue Space	Queue space that contains the CR3_ERROR_QUEUE. <i>Required.</i>
Logging		
CR3_ALEIN_PRINT_OPTION	F	Default output is to the TUXEDO USERLOG. You can write logging and tracing output to the file specified in CR3_ALEIN_PRINT_FILE by setting this environment variable to “F”. <i>Optional.</i>
CR3_ALEIN_PRINT_FILE	Path and File	Specifies the full path and file for logging and tracing output if CR3_ALEIN_PRINT_OPTION is set to “F”. <i>Optional.</i>
CR3_ALEIN_TRACE	Y or N	Specify “Y” to include detailed trace information in the log output. <i>Optional.</i>

6 Configuring ALE Integration

Sample Environment File for ALE eLink to R/3 (cr3alein.env)

BEA eLink Adapter for R/3 ALE provides a semi-configured TUXEDO environment file (`cr3alein.env`) for use with ALE eLink to R/3. You should create a separate environment file for each ALE eLink to R/3 server specified in the `cr3_ale.ubb` configuration file.

Note the settings in the sample `cr3alein.env` file:

Listing 6-1 Sample Environment File (cr3alein.env)

```
CR3_DESTINATION=<R/3 system name>
CR3_CLIENT=<R/3 client id>
CR3_USER=<R/3 user>
CR3_PASSWORD=<R/3 password>
CR3_LANGUAGE=E
CR3_ALEIN_TID_FILE=<your eLink app directory>\cr3alein
CR3_ALEIN_TID_PER_IDOC=N
CR3_ALEIN_IDOCIN_QUEUE=CR3_IDOC_IN
CR3_ALEIN_IDOCIN_QUEUE_SPACE=QSPACE
CR3_ALEIN_ERROR_QUEUE=CR3_ERRQUE
CR3_ALEIN_ERROR_QUEUE_SPACE=QSPACE
CR3_ALEIN_PRINT_OPTION=F
CR3_ALEIN_PRINT_FILE=<your eLink app directory>\cr3alein.log
CR3_ALEIN_TRACE=Y
```

Note: You can use any directory for the eLink application directory. However, it is recommended that you use the same directory as the TUXEDO APPDIR.

Setting Up the eLink to R/3 Connection to R/3

To set up the eLink to R/3 connection to R/3, you must set environment variables and configure the `sideinfo` file. See “Configuring eLink to R/3 Connections” in Chapter 7, “Configuring R/3 Connections” for instructions.

Configuring Queues for ALE eLink to R/3

ALE eLink to R/3 requires that you set up two TUXEDO queues, `CR3_ALE_ERROR_QUEUE`, and `CR3_ALE_IDOC_IN_QUEUE`. See “Setting Up Queues for eLink to R/3 IDOCs” later in this topic for instructions.

Specifying the TID File for ALE eLink to R/3

ALE eLink to R/3 uses a TID file to log TIDs that it has already processed to guarantee that it delivers an IDOC packet to R/3 once and only once. You must specify the path and file name of the TID file (*without* an extension) in the CR3_ALEIN_TID_FILE environment variable, as shown in the following example:

```
CR3_ALEIN_TID_FILE=<your eLink app directory>\cr3alein
```

Upon start-up, if ALE eLink to R/3 does not find a TID file in the specified location, it creates one automatically. See “Configuring the TID File Manager” on page 6-22 for information about configuring and administering the TID file.

Splitting eLink to R/3 IDOC Packets Containing Multiple IDOCs

By default, ALE eLink to R/3 passes an IDOC message containing multiple IDOCs to R/3 in a single packet. You can configure ALE eLink to R/3 to split IDOC messages containing multiple IDOCs into individual IDOC messages, each with its own TID. For example, if an IDOC message contains six IDOCs, ALE eLink to R/3 can create six IDOC separate packets, each containing a single IDOC and associated with a unique TID.

To split eLink to R/3 IDOC packets, set the CR3_ALEIN_TID_PER_IDOC environment variable to “Y”, as shown in the following example:

```
CR3_ALEIN_TID_PER_IDOC=Y
```

See “Splitting eLink to R/3 IDOC Packets” in Chapter 2, “Integrating with ALE,” in this guide for more information.

Configuring Logging for ALE eLink to R/3

By default, ALE eLink to R/3 writes logging output to the TUXEDO USERLOG. You can set the CR3_ALEIN_PRINT_OPTION environment variable to “F” to send output to a log file. If sending output to a file, you must specify its name and location in the CR3_ALEIN_PRINT_FILE environment variable.

You can also include trace information in the log by setting the CR3_ALEIN_TRACE environment variable to “Y”. You exclude trace information by omitting this environment variable or setting it to “N”.

The following example sets logging options for ALE eLink to R/3:

```
CR3_ALEIN_PRINT_OPTION=F
CR3_ALEIN_PRINT_FILE=<your eLink app directory>\cr3alein.log
CR3_ALEIN_TRACE=Y
```

Configuring Load Balancing for eLink to R/3 IDOCs

Multiple instances of ALE eLink to R/3 can log onto R/3 and submit eLink to R/3 IDOCs for processing. For deployments that involve high volumes of eLink to R/3 IDOCs, you can enhance system performance by balancing the load across multiple instances of ALE eLink to R/3.

Use TUXEDO's load balancing capabilities to manage the number of ALE eLink to R/3 instances running. See your BEA TUXEDO documentation regarding TUXEDO administrative tools for more information.

Note: Each instance of ALE eLink to R/3 requires a separate R/3 login, so be sure to tune the number of concurrent instances to optimize system performance without requiring excessive R/3 resources. Instances can log in using the same user name and password.

Configuring the ALE R/3 to eLink Client

The following topics describe how to configure the ALE R/3 to eLink client:

- Starting the ALE R/3 to eLink Client
- Setting Environment Variables for ALE R/3 to eLink
- Configuring Load Balancing for R/3 to eLink
- Configuring Multiple Program IDs

See “Setting Up Queues for R/3 to eLink IDOCs” later in this topic for configuration instructions.

ALE R/3 to eLink is a TUXEDO client that connects to R/3 by using the register mode; receives R/3 to eLink IDOC packets from R/3 via Transactional RFC (tRFC); encodes each IDOC packet to an FML32 message buffer; and queues the IDOC message into a TUXEDO queue for subsequent processing. See “Processing R/3 to eLink IDOCs” in Chapter 2, “Integrating with ALE,” in this guide for more information about ALE R/3 to eLink.

Starting the ALE R/3 to eLink Client

You start ALE R/3 to eLink with a startup script that specifies environment variables and command line options.

Using the Startup Script for ALE R/3 to eLink

ALE R/3 to eLink is started with a script that defines a number of environment variables used to control the behavior of the server. BEA eLink Adapter for R/3 ALE provides a sample ALE R/3 to eLink startup program that is partially configured:

- **UNIX:** `cr3aleout.sh` (Bourne shell format)
- **Windows NT:** `cr3aleout.bat`

Command Line Options for ALE R/3 to eLink

To connect to R/3, you specify the following command line options to the `cr3aleout` command:

```
cr3aleout -a program-id -g gateway_host_name -x  
gateway_service_name
```

where `program-id`, `gateway_host_name`, and `gateway_service_name` are the appropriate values for your R/3 environment, as shown in the following example:

```
cr3aleout -a bea.cr3.beadoc1.ale -g hpsap -x sapgw00
```

These parameters are required to register with the SAP Gateway. The `program-id` must match a program ID that has already been configured for the specified gateway service on the specified R/3 host so that the ALE Subsystem can send IDOC packets to the correct ALE R/3 to eLink instance. See “Configuring R/3 to eLink Connections” in Chapter 7, “Configuring R/3 Connections,” for more information about these start-up options.

Note: The `cr3aleout` command resides in the `\adapters\sapr3_ale\bin` subdirectory of your BEA eLink Adapter for R/3 ALE installation directory.

Setting Environment Variables for ALE R/3 to eLink

You can configure the following required and optional environment variables for ALE R/3 to eLink:

Table 6-2 Environment Variables for the ALE R/3 to eLink Client

Variable	Set To	Description
Configuring IDOC Routing		
CR3_ALE_DEFAULT_TARGET_ID	Data-dependent routing (DDR) value	Default data-dependent routing value that is encoded in the CR3_TARGET_ID field in the FML32 message (if it is not specified in the destination map file). This value matches the field ID configured in the ROUTING section of the <code>cr3_ale.ubb</code> configuration file. To use DDR, the server group to receive the IDOC must also be configured. This environment variable <i>must</i> be defined even if it is a dummy value that is not defined in <code>cr3_ale.ubb</code> . <i>Required.</i>
CR3_ALEOUT_DEST_MAP_FILE	File Name	Destination map file specifies the mapping between R/3 to eLink IDOCs and the destination queue. <i>Optional.</i>
Splitting IDOC Packets		
CR3_ALE_DEFAULT_IDOC_SPLIT	Y or N	If no destination map file is specified, then specifies whether IDOC packets containing multiple IDOCs are passed either together in a single message or split into separate IDOC packets, with one IDOC per message. <i>Optional.</i>
Configuring Queues		
CR3_ALE_TARGET_QUEUE	Queue	Name of the default target TUXEDO queue. Used if no destination map file is specified or if no matching entry is found for the IDOC in the destination map file. <i>Required.</i>
CR3_ALE_TARGET_QUEUE_SPACE	Queue Space	Name of the TUXEDO queue space containing the default target queue. <i>Required.</i>

6 Configuring ALE Integration

Table 6-2 Environment Variables for the ALE R/3 to eLink Client

Variable	Set To	Description
Configuring the TID Log File		
CR3_ALEOUT_TID_FILE	Path and File Name	Name and location of the TID file. The file name must <i>not</i> include an extension. <i>Required.</i>
Logging Options		
CR3_ALEOUT_PRINT_OPTION	S or F	Default output is to the TUXEDO USERLOG. <i>Optional.</i> You can write logging and tracing output to file or stdout using these settings: <ul style="list-style-type: none">■ “S” sends output to stdout.■ “F” sends output to the file specified in CR3_ALEOUT_PRINT_FILE.
CR3_ALEOUT_PRINT_FILE	Path and File	Specifies the full path and file for logging and tracing output if CR3_ALEOUT_PRINT_OPTION is set to “F”. <i>Optional.</i>
CR3_ALEOUT_TRACE	Y or N	Specify “Y” to include detailed trace information in the log output. <i>Optional.</i>

These environment variables are defined in the server startup program.

Sample Startup Script for ALE R/3 to eLink (cr3aleout.bat)

The following sample Windows NT startup script specifies TUXEDO and ALE R/3 to eLink settings:

Listing 6-2 Sample Startup Script (cr3aleout.bat)

```
set CR3_BIN=<your eLink install directory>\adapters\sapr3_ale\bin
set TUXDIR=<your Tuxedo directory>
set APPDIR=<your eLink app directory>
set PATH=%TUXDIR%\bin;%APPDIR%;%PATH%
set TUXCONFIG=%APPDIR%\tuxconfig
```

```
set TLOGDEVICE=<your eLink app directory>\TLOG
set QMCONFIG=<your eLink app directory>\QUE
set FIELDTBLS32=cr3_ale.fml
set FLDTBLDIR32=<your eLink app directory>;<your Tuxedo
directory>\udataobj
set CR3_ALE_TARGET_QUEUE=CR3_SUBMIT
set CR3_ALE_TARGET_QUEUE_SPACE=QSPACE
rem set CR3_MERCATOR_FV_IN=Y
set CR3_ALE_DEFAULT_TARGET_ID=CR3_ERROR
set CR3_ALE_DEFAULT_IDOC_SPLIT=N
rem set CR3_ALEOUT_DEST_MAP_FILE=<your eLink app
directory>\cr3aleout.map
set CR3_ALEOUT_TID_FILE=<your eLink app directory>\cr3aleout
set CR3_ALEOUT_PRINT_OPTION=S
set CR3_ALEOUT_TRACE=Y
set CR3_ALEOUT_DUMP_IDOC_BUFFER=N
set CR3_ALEOUT_DUMP_IDOC_FML_BUFFER=N
%CR3_BIN%\cr3aleout.exe -a <your.program.id> -g <R/3 system> -x
sapgw00
```

Note: This example applies to Windows NT. For Unix startup scripts, use the `setenv` command for the C shell or the `export` command for the Bourne or Korn shells.

Configuring Queues for ALE R/3 to eLink

ALE R/3 to eLink requires that you set up at least one default TUXEDO queue and additional queues as needed. See “Setting Up Queues for R/3 to eLink IDOCs” later in this topic for instructions.

Setting the Default Data-Dependent Routing Value

You can configure the default data-dependent routing (DDR) value that ALE R/3 to eLink assigns to each IDOC message (in the `CR3_TARGET_ID` field) that it queues. See your TUXEDO documentation for more information about data-dependent routing.

Note: To use DDR, the DDR value must match the field ID configured in the ROUTING section of the `cr3_ale.ubb` configuration file.

The following example sets the `CR3_ALE_DEFAULT_TARGET_ID` environment variable to a DDR value:

6 Configuring ALE Integration

```
set CR3_ALE_DEFAULT_TARGET_ID=DDR_VALUE
```

See “Setting Data-Dependent Routing Values in the Map File” later in this topic and “Managing Data-Dependent Routing” in Chapter 2, “Integrating with ALE,” in this guide for more information.

Configuring a Destination Map File

You can configure ALE R/3 to eLink to use a destination map file so that ALE R/3 to eLink can queue IDOC messages into different target queues, manage data-dependent routing, and group similar IDOC messages into larger IDOC messages. ALE R/3 to eLink makes routing and grouping decisions about individual IDOCs according to two settings specified in an IDOC’s control record: the logical system ID of the target R/3 system and the IDOC message type. If a destination map file is used, ALE R/3 to eLink automatically splits IDOC packets containing multiple IDOCs into separate IDOC messages, with each message containing a single IDOC.

Structure of the Destination Map File

The destination map file is an ASCII text file. Each line denotes a separate entry and consists of the columns shown in Table 6-3:

Table 6-3 Columns in the Destination Map File

#	Column	Length	Description
1	RCVPRN	10	Logical System ID of the target R/3 logical system. If the logical system ID is fewer than ten characters long, you must pad spaces at the end until its length is ten characters.
2	MESTYP	6	Message type specified in the IDOC. If the message type is less than six characters long, you must pad spaces at the end until its length is six characters.
3	Compress	1	Compress (Y or N). Starting position is column 18. If Compress=Y, then ALE R/3 to eLink concatenates IDOC packets with matching logical system and IDOC message type values into a single IDOC message in the queue. If Compress=N, then ALE R/3 to eLink keeps the IDOCs in individual IDOC messages.

Table 6-3 Columns in the Destination Map File

#	Column	Length	Description
4	TargetId	variable	Data-dependent routing (DDR) value. Starting position is column 20. The value is encoded in the FML field CR3_TARGET_ID for use with DDR.
5	QueueSpace	variable	Name of the destination queue space for the IDOC message.
6	QueueName	variable	Name of the destination queue for the IDOC message.

Sample Destination Map File

The following is a sample destination map file:

Listing 6-3 Sample Destination Map File

```
LOGSYS1    MATMAS  N  DDR_VALUE_1  QSPACE1  QUEUE1
LOGSYS2    MATMAS  Y  DDR_VALUE_2  QSPACE1  QUEUE2
LOGSYS3    DEBMAS  N  DDR_VALUE_3  QSPACE2  QUEUE3
LOGSYS4    CREMAS  Y  DDR_VALUE_4  QSPACE3  QUEUE4
```

Setting the CR3_ALEOUT_DEST_MAP_FILE Environment Variable

To use a destination map file, you must specify its name and location in the CR3_ALEOUT_DEST_MAP_FILE environment variable, as shown in the following example:

```
set CR3_ALEOUT_DEST_MAP_FILE=your eLink app
directory\cr3aleout.map
```

Specifying Default Target Queue Spaces and Queue Names in the Map File

You use the QueueSpace and QueueName columns in the destination map file to associate an IDOC with a destination TUXEDO queue space and queue. If an entry in the destination map file matches the logical system ID of the target R/3 system and the IDOC message type specified in an IDOC's control record, then ALE R/3 to eLink queues the IDOC into the specified queue space and queue name. If no matching entry

6 *Configuring ALE Integration*

is found in the destination map file, then ALE R/3 to eLink queues the IDOC message into the default queue space and queue, which are defined in the CR3_ALE_TARGET_QUEUE_SPACE and CR3_ALE_TARGET_QUEUE environment variables.

Note: You must create the default queue and any individual queues before processing R/3 to eLink IDOCs. See “Setting Up Queues for R/3 to eLink IDOCs” later in this topic for information.

Setting Data-Dependent Routing Values in the Map File

You use the TargetId column in the destination map file to associate an IDOC with a data-dependent routing value. If an entry in the destination map file matches the logical system ID of the target R/3 system and the IDOC message type specified in an IDOC’s control record, then ALE R/3 to eLink encodes the specified TargetId value in the CR3_TARGET_ID FML32 field of the IDOC message. If no matching entry is found in the destination map file, then ALE R/3 to eLink encodes the default DDR value, which is defined in the CR3_ALE_DEFAULT_TARGET_ID environment variable. See “Setting the Default Data-Dependent Routing Value” on page 6-13 for more information.

Note: To use DDR, the DDR value must match the field ID configured in the ROUTING section of the cr3_ale.ubb configuration file.

Compressing R/3 to eLink IDOCs

You use the Compress column in the destination map file to combine IDOC messages with matching logical system IDs and IDOC message types into an IDOC message containing multiple IDOCs. If an entry in the destination map file matches the logical system ID of the target R/3 system and the IDOC message type specified in an IDOC’s control record, and if the Compress column is “Y”, then ALE R/3 to eLink aggregates the entry in the destination map file with other matching IDOCs in a single, larger IDOC message that it then queues into the appropriate target queue. If the Compress column is “N”, then ALE R/3 to eLink queues each IDOC message separately.

Splitting R/3 to eLink IDOC Packets

If no destination map file is specified, you can configure ALE R/3 to eLink to split IDOC packets containing multiple IDOCs into separate IDOC messages, each containing a single IDOC. By default, ALE R/3 to eLink queues the entire IDOC

packet. If you set the `CR3_ALE_DEFAULT_IDOC_SPLIT` environment variable to “Y”, ALE R/3 to eLink will split the IDOC packet into individual IDOC packets. See “Splitting R/3 to eLink IDOC Packets Into Individual IDOCs” in Chapter 2, “Integrating with ALE,” in this guide for more information.

Specifying the TID File for ALE R/3 to eLink

ALE R/3 to eLink uses a TID file to log TIDs that it has already processed to guarantee that it queues an IDOC packet from R/3 once and only once. You must specify the path and file name of the TID file (*without* an extension) in the `CR3_ALEOUT_TID_FILE` environment variable, as shown in the following example (Windows NT):

```
CR3_ALEOUT_TID_FILE=<your eLink app directory>\cr3aleout
```

Upon start-up, if ALE R/3 to eLink does not find a TID file in the specified location, it creates one automatically. See “Configuring the TID File Manager” later in this topic for information about configuring and administering the TID file.

Configuring Logging for ALE R/3 to eLink

By default, ALE R/3 to eLink writes logging and tracing output to the TUXEDO USERLOG. You can set the `CR3_ALEOUT_PRINT_OPTION` environment variable to “S” to send output to stdout, or you can set it to “F” to send output to a log file. If you are sending output to a file, you must specify its name and location in the `CR3_ALEOUT_PRINT_FILE` environment variable.

You can also include trace information in the log by setting the `CR3_ALEOUT_TRACE` environment variable to “Y”. You exclude trace information by omitting this environment variable or setting it to “N”.

The following example from a startup script file (Windows NT) sets logging options for ALE R/3 to eLink:

```
set CR3_ALEOUT_PRINT_OPTION=S
set CR3_ALEOUT_TRACE=Y
```

Configuring Load Balancing for R/3 to eLink

Multiple instances of ALE R/3 to eLink can register using the same program ID. For deployments that involve high volumes of R/3 to eLink IDOC packets, you can enhance system performance by balancing the load across multiple instances of ALE R/3 to eLink. Instances that register under the same program ID must also share the same TID file.

To run multiple instances of ALE R/3 to eLink, specify the shared TID file (set the `CR3_ALEOUT_TID_FILE` environment variable), and then run the `cr3aleout` command multiple times. The following example launches three instances of ALE R/3 to eLink using the program ID `bea.cr3.beadoc1.ale`:

Listing 6-4 Sample Commands Launching Multiple ALE R/3 to eLink Instances

```
set CR3_BIN=<your eLink install directory>\adapters\sapr3_ale\bin
CR3_ALEOUT_TID_FILE=<your eLink app directory>\cr3aleout
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.ale -g hpsap -x sapgw00
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.ale -g hpsap -x sapgw00
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.ale -g hpsap -x sapgw00
```

Note: Instances of ALE R/3 to eLink that share the same TID file should all be launched on the same machine.

See “Load Balancing High Volumes of R/3 to eLink IDOCs” in Chapter 2, “Integrating with ALE,” of this guide.

Configuring Multiple Program IDs

If R/3 is configured to send R/3 to eLink IDOCs to different program IDs, you can configure ALE R/3 to eLink to handle these IDOCs by running multiple instances of ALE R/3 to eLink using different program IDs. You must make sure that all instances sharing the same program ID also share the same TID file, and that all instances sharing the same TID file also share the same program ID. Instances that register under different program IDs must *not* share the same TID file.

The following example launches three instances of ALE R/3 to eLink using the program ID `bea.cr3.beadoc1.aleX` and three other instances of ALE R/3 to eLink using the program ID `bea.cr3.beadoc1.aleY`:

Listing 6-5 Sample Commands Launching Multiple ALE R/3 to eLink Instances

```
set CR3_BIN=<your eLink install directory>\adapters\sapr3_ale\bin
CR3_ALEOUT_TID_FILE=<your eLink app directory>\cr3aleoutX
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.aleX -g hpsap -x sapgw00
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.aleX -g hpsap -x sapgw00
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.aleX -g hpsap -x sapgw00
CR3_ALEOUT_TID_FILE=<your eLink app directory>\cr3aleoutY
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.aleY -g hpsap -x sapgw00
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.aleY -g hpsap -x sapgw00
%CR3_BIN%\cr3aleout.exe -a bea.cr3.beadoc1.aleY -g hpsap -x sapgw00
```

Note: Instances of ALE R/3 to eLink that share the same TID file should all be launched on the same machine.

See “Registering Multiple Program IDs” in Chapter 2, “Integrating with ALE,” in this guide for more information.

Configuring Queues

The following topics describe how to create and configure TUXEDO queues for ALE processing:

- TUXEDO Queues
- Setting Up Queues for eLink to R/3 IDOCs
- Setting Up Queues for R/3 to eLink IDOCs

TUXEDO Queues

You must set up queues for both ALE eLink to R/3 and ALE R/3 to eLink. You need to create and configure the Server Group (TMS_QM), Message Queue Server (TMQUEUE) and Message Forwarding Server (TMQFORWARD). See the */Q Guide* in your BEA TUXEDO documentation for more information about TUXEDO queues.

Setting Up Queues for eLink to R/3 IDOCs

ALE eLink to R/3 uses the following queues:

Queue Name	Description
CR3_SUBMIT	IDOC submit queue. Handles IDOC packets that a TUXEDO application or other application has submitted to the CR3_SUBMIT service for processing.
CR3_ERROR	IDOC error queue. Handles invalid IDOC packets that have failed validation by the CR3_SUBMIT service.
CR3_IDOC_IN	IDOC eLink to R/3 queue. Handles IDOC packets that the CR3_SUBMIT service has validated, assigned a TID, and queued. TMQFORWARD forwards each IDOC packet to the CR3_IDOC_IN service for submission to R/3.

Use the TUXEDO Queue manager administration utility (qmadmin) to create these queue spaces and queues. See the */Q Guide* in your BEA TUXEDO documentation for instructions.

Defining the TMQUEUE and TMQFORWARD Servers in the UBB File

In the SERVERS section of your UBB file, define the TMQUEUE and TMQFORWARD servers:

Listing 6-6 Sample SERVERS Section of UBB File

```
TMQUEUE
  SRVGRP=QUE SRVID=1
  CLOPT="-s QSPACE:TMQUEUE -- "
TMQFORWARD
  SRVGRP=QUE SRVID=2
  CLOPT=" -- -i 10 -q CR3_IDOC_IN,CR3_SUBMIT"
```

Setting Environment Variables

After creating the error and eLink to R/3 queue spaces and queues, specify their names in environment variables:

Listing 6-7 Setting Environment Variables for Queues

```
SET CR3_ALEIN_IDOCIN_QUEUE=CR3_IDOC_IN
SET CR3_ALEIN_IDOCIN_QUEUE_SPACE=QSPACE
SET CR3_ALEIN_ERROR_QUEUE=CR3_ERROR
SET CR3_ALEIN_ERROR_QUEUE_SPACE=QSPACE
```

See Table 6-1, “Environment Variables for the ALE eLink to R/3 Server,” earlier in this topic for a description of these environment variables.

Setting Up Queues for R/3 to eLink IDOCs

ALE R/3 to eLink can queue R/3 to eLink IDOC messages into one or more queues. At a minimum, you must create at least one default queue. If you are using a destination map file to queue IDOC messages to multiple queues, you must create all the target queues, including the default queue.

Use the TUXEDO Queue manager administration utility (qadmin) to create these queue spaces and queues. See the */Q Guide* in your BEA TUXEDO documentation for instructions.

After creating the R/3 to eLink queues, specify the default queue space and queue name in environment variables, as shown the following example:

```
set CR3_ALE_TARGET_QUEUE=CR3_SUBMIT
set CR3_ALE_TARGET_QUEUE_SPACE=QSPACE
```

See Table 6-2, “Environment Variables for the ALE R/3 to eLink Client,” earlier in this topic for a description of these environment variables.

Configuring the TID File Manager

The following topics describe the TID File Manager, a command line utility that manages the size of the TID file in the run-time environment:

- Command Line Options
- Running the TID File Manager
- How the TID File Manager Works

Command Line Options

The TID File Manager uses the command line options shown in Table 6-4:

Table 6-4 Command Line Options for the TID File Manager

Option	Description
-h or -?	Help text that describes each option.
-f	Name of the TID file (path is optional). <i>Required.</i>
-n	Number of lines to keep in the TID file. Default is 10 lines. If the date (-d) option is also specified, this setting overrides the date setting. The TID File Manager retains at least this number of lines in the original TID file.
-t	Frequency (in seconds) with which the TID File Manager periodically checks the TID file. Default is 60 seconds. If this option is omitted, then the TID File Manager runs once and exits. If this option is specified, the TID File Manager keeps running. <i>Optional.</i>
-d	Last save date. The TID File Manager removes any CONFIRMED entries that are older than the specified date. Valid date formats include: <ul style="list-style-type: none"> ■ Jan/1/2000 ■ 2000,1,jan ■ "1 JAN 2000"

Running the TID File Manager

To run the TID File Manager, enter the following command:

```
cr3tidmanager -f FileName [OptionalParameters]
```

where *FileName* is required and *OptionalParameters* include any of the other options in Table 6-4, “Command Line Options for the TID File Manager,” on page 6-23.

Note: You must run a *separate* TID File Manager for *each* TID file used by instances of ALE eLink to R/3 and ALE R/3 to eLink.

How the TID File Manager Works

Upon startup, the TID File Manager does the following:

- Locks the specified TID file.
- Creates a backup file using the original file name plus the date-time and a .bck extension.
- Removes from the TID log file only CONFIRMED entries that meet the configured criteria.
- Appends each removed entry to the backup file.
- Saves both files.
- Releases the file lock.

Because the TID File Manager locks the file while editing, be careful to specify a large enough frequency to contain the file size without slowing run-time performance.

7 Configuring R/3 Connections

This topic describes how to configure your BEA eLink Adapter for R/3 ALE installation for eLink to R/3 and R/3 to eLink communication with R/3. It includes the following main sections:

- Configuring eLink to R/3 Connections
- Configuring R/3 to eLink Connections
- Troubleshooting Connection Problems

Configuring eLink to R/3 Connections

The ALE eLink to R/3 server uses the sideinfo method of connecting to R/3. The following topics describe how to configure eLink to R/3 connections:

- Setting Environment Variables for the eLink to R/3 Connection
- Configuring the sideinfo File

Setting Environment Variables for the eLink to R/3 Connection

Configure the environment variables for eLink to R/3 connections as shown in Table 7-1:

Table 7-1 Environment Variables for Connecting to R/3

Variable Name	Set To	Description
CR3_DESTINATION	System Name	Destination in the <code>sideinfo</code> file that defines the R/3 application server to connect to. Default is undefined.
CR3_CLIENT	Client Number	R/3 login client number. Default is undefined.
CR3_USER	User	R/3 login user. Default is undefined. The user must be of type CPIC.
CR3_PASSWORD	Password	R/3 login user password. Default is undefined.
CR3_LANGUAGE	E	R/3 login language. Default is E for English.
SIDE_INFO	Path and File	Full path to the <code>sideinfo</code> file. See “Specifying the Location of the <code>sideinfo</code> File” in a later section.

The following are sample settings in an environment file:

Listing 7-1 Sample Environment Variable Settings for eLink to R/3 Connections

```
CR3_DESTINATION=SAPNODE
CR3_CLIENT=SAPCLIENT
CR3_USER=LOGINNAME
CR3_PASSWORD=LOGINPASSWORD
CR3_LANGUAGE=E
SIDE_INFO=<your eLink app directory>\sideinfo
```

The settings you specify depend on the configuration of your R/3 installation. See your R/3 system administrator for more information.

Configuring the sideinfo File

The ALE eLink to R/3 server uses the `sideinfo` method of connecting to R/3. The `sideinfo` file is an ASCII text file that specifies the connection point for external programs to communicate with the R/3 application server. See your SAP R/3 documentation for more information about the `sideinfo` file.

The `sideinfo` file is associated with the parameters provided to the ALE eLink to R/3 server when it is started. A sample `sideinfo` file comes with BEA eLink Adapter for R/3 ALE, but you must configure this sample file for your particular R/3 environment and BEA eLink Adapter for R/3 ALE installation. Errors in this file will prevent the ALE eLink to R/3 server from connecting to R/3.

Contents of the sideinfo File

The following table describes the attributes in the `sideinfo` file:

Table 7-2 Attributes in the sideinfo File

Attribute	Description
DEST	Connection identifier. You can set this to any name, usually a few characters in all uppercase. It is recommended that you use the node name of the R/3 application server. The ALE eLink to R/3 servers uses this identifier to determine the R/3 application server to connect to. The ALE eLink to R/3 server gets the identifier name from the <code>CR3_DESTINATION</code> environment variable that is defined in the server start-up script. The identifier links its setting to the connection.
LU	TCP/IP name of the node on which the R/3 application server is running. The ALE eLink to R/3 server will open a TCP/IP socket to the R/3 application node. The LU can be specified as the full TCP/IP name or as an alias. This name or alias must match exactly what is specified in the hosts file, or it must be translated by the directory name service. Test this by using the <code>ping</code> command and the exact string used in the LU line item (<code>ping node</code>).

7 Configuring R/3 Connections

Table 7-2 Attributes in the sideinfo File

Attribute	Description
TP	Name of the R/3 user presentation connection socket (the network socket to which the SAPGUI connects). This name is configured in R/3 and is specific to the instance of R/3. The last two characters in the name specify the instance number. This name must appear in the services file on the node on which the ALE eLink to R/3 is running, and the value of the socket number must match the same number in the services file on the R/3 application server.
GWHOST	Node name for the R/3 Gateway. As with the LU attribute, it must be translated correctly into a TCP/IP address (test it by using the <code>ping</code> command to ping the node). The gateway host is usually the same node as the R/3 application server, although this is not required. The R/3 Gateway process might reside on a different node. If the gateway must be moved, then the gateway identifier must be changed in the R/3 profile. Be sure to check with the R/3 system manager to ensure that this value matches the gateway location.
GWSERV	Name of the TCP/IP socket that is the R/3 Gateway connection. All processes connecting to R/3 in the background use this socket number as an entry point. As with the TP attribute, this name must be defined in the services file on the node on which the ALE eLink to R/3 server is running, and the definition must match the definition of the R/3 application server.

In an R/3 environment with multiple R/3 application servers, this list of attributes is simply repeated for different node names, different instances of R/3, and possibly different gateways.

Sample Settings

The following listing shows sample settings in a `sideinfo` file:

Listing 7-2 Sample Settings for sideinfo File

```
DEST=SAPNODE
LU=nodename
TP=sapdp00
PROTOCOL=I
GWHOST=nodename
GWSERV=sapgw00
```

Specifying the Location of the sideinfo File

The `sideinfo` file is usually located in the directory where the ALE eLink to R/3 server is running (for example, the TUXEDO `appdir`). If you cannot use this location in your environment (for example, a number of other processes also connect to R/3 and use the same `sideinfo` information), then you can avoid duplicating the file by assigning the `SIDE_INFO` environment variable to a path containing the `sideinfo` table, as shown in the following examples:

UNIX C shell (csh):

```
setenv SIDE_INFO <your eLink app directory>/sideinfo
```

Windows NT:

```
set SIDE_INFO=<your eLink app directory>\sideinfo
```

The `sideinfo` file must have protections set so that all processes can read it.

For the ALE eLink to R/3 server, this environment variable is defined in the server start-up script. See “Setting Environment Variables for ALE eLink to R/3” in Chapter 6, “Configuring ALE Integration,” for more information.

Configuring R/3 to eLink Connections

The ALE R/3 to eLink server uses the register mode method of connecting to R/3. When the ALE R/3 to eLink server is started, it connects to R/3 and registers with a specific Program ID. This Program ID is defined in the RFC destination, as defined in transaction SM59. When an RFC is executed with the specified destination, the request is passed to the server registered on the Program ID. If no server is registered, then an error occurs. The server *must* be started and registered on the Program ID before the RFC is executed.

7 Configuring R/3 Connections

To register with R/3, the following information must be passed to ALE R/3 to eLink as command line parameters (which are specified in the server start-up script):

Setting	Description
<i>Program-Id</i>	Program ID (text string). It must match the <code>program-id</code> defined in transaction SM59.
<i>Gateway Host Name</i>	Name of the host running the SAP gateway server. It must match the <code>GWHOST</code> parameter specified in the <code>sideinfo</code> file.
<i>Gateway Service Name</i>	Service name of the SAP gateway server. It must match the <code>GWSERV</code> parameter specified in the <code>sideinfo</code> file.

Troubleshooting Connection Problems

R/3 provides extensive help in isolating and testing connection problems involving the R/3 Gateway. The R/3 Gateway is sometimes called the CPIC Gateway because the CPIC protocol is the lowest-level protocol used on the TCP/IP socket when connecting to R/3. RFCs are essentially layered on CPIC.

To troubleshoot connection problems:

- Verify that the TCP/IP network is functioning by using network testing.
- Examine your R/3 installation. Common problems arise from typographical errors in the `sideinfo` file, the `hosts` file, and the `services` file. Examine these files, bearing in mind that UNIX is case-sensitive (including TCP/IP translation and socket name translation).
- If problems persist, thoroughly examine the CPIC implementation on the R/3 application server. The R/3 documentation provides extensive material about the CPIC setup under the heading "BC - SAP Communication: Configuration." If necessary, the CPIC implementation can be modified to accommodate the BEA eLink Adapter for R/3 ALE installation.

A Sample cr3_ale.ubb File

This topic describes the two preconfigured sample cr3_ale.ubb files (for UNIX and Windows NT) that come with BEA eLink Adapter for R/3 ALE.

UNIX

```
#####  
*RESOURCES  
#####  
  
IPCKEY40000  
DOMAINIDCR3  
MASTERSITE1  
MAXACCESSERS30  
MAXSERVERS20  
MAXSERVICES30  
SCANUNIT10  
SANITYSCAN12  
BLOCKTIME30  
MAXGTT200  
MODELSHM  
LDBALY  
  
#####  
*MACHINES  
#####  
  
<system name>  
LMID=SITE1  
TYPE="HP-UX"  
TUXDIR="<your Tuxedo install directory>"  
TUXCONFIG="<your Tuxedo app directory>/tuxconfig"
```

```
TLOGDEVICE = "<your Tuxedo app directory>/TLOG"
TLOGSIZE=10
APPDIR="<your Tuxedo app directory>"
ULOGPFX="<your Tuxedo app directory>/ULOG"
MAXWSCLIENTS=2
UID=278

#####
*GROUPS
#####

QUE
LMID=SITE1  GRPNO=1
TMSNAME=TMS_QM  TMSCOUNT=2
OPENINFO="TUXEDO/QM:<your Tuxedo app directory>/QUE:QSPACE"

CR3
LMID=SITE1  GRPNO=2
TMSNAME=TMS  TMSCOUNT=2

#####
*SERVERS
#####

DEFAULT:
CLOPT= "-A"

TMQUEUE
SRVGRP=QUE  SRVID=1
```

A Sample cr3_ale.ubb File

```
GRACE=0 RESTART=Y CONV=N MAXGEN=10
CLOPT="-s QSPACE:TMQUEUE -- "

TMQFORWARD
SRVGRP=QUE SRVID=2
GRACE=0 RESTART=Y CONV=N MAXGEN=10
CLOPT=" -- -i 10 -q CR3_IDOC_IN,CR3_SUBMIT"

cr3alein
    SRVGRP=CR3 SRVID=3
ENVFILE="cr3alein.env"

#####
*SERVICES
#####

CR3_SUBMIT
CR3_IDOC_IN
```

Windows NT

```
#####
*RESOURCES
#####

IPCKEY      40000
```



```
DOMAINIDCR3
MASTER      SITE1
MAXACCESSERS30
MAXSERVERS20
MAXSERVICES30
SCANUNIT10
SANITYSCAN12
BLOCKTIME30
MAXGTT      200
MODEL       SHM
LDBAL       Y
```

```
#####
*MACHINES
#####
```

```
<system name>
  LMID=SITE1
  TYPE="HP-UX"
  TUXDIR="<your Tuxedo install directory>"
  TUXCONFIG="<your Tuxedo app directory>/tuxconfig"
  TLOGDEVICE ="<your Tuxedo app directory>/TLOG"
  TLOGSIZE=10
  APPDIR="<your Tuxedo app directory>"
  ULOGPFX="<your Tuxedo app directory>/ULOG"
  MAXWSCLIENTS=2
  UID=278
```

A Sample cr3_ale.ubb File

```
#####  
*GROUPS  
#####  
  
QUE  
    LMID=SITE1  GRPNO=1  
    TMSNAME=TMS_QM  TMSCOUNT=2  
    OPENINFO="TUXEDO/QM:<your Tuxedo app directory>/QUE:QSPACE"  
  
CR3  
    LMID=SITE1  GRPNO=2  
    TMSNAME=TMS  TMSCOUNT=2  
  
#####  
*SERVERS  
#####  
  
DEFAULT:  
    CLOPT="-A"  
  
TMQUEUE  
    SRVGRP=QUE  SRVID=1  
    GRACE=0  RESTART=Y  CONV=N  MAXGEN=10  
    CLOPT="-s QSPACE:TMQUEUE -- "  
  
TMQFORWARD
```

```
SRVGRP=QUE SRVID=2
GRACE=0 RESTART=Y CONV=N MAXGEN=10
CLOPT=" -- -i 10 -q CR3_IDOC_IN,CR3_SUBMIT"
```

cr3alein

```
SRVGRP=CR3 SRVID=3
ENVFILE="cr3alein.env"
```

```
#####
*SERVICES
#####
```

```
CR3_SUBMIT
CR3_IDOC_IN
```

A *Sample cr3_ale.ubb File*

Glossary

ABAP/4

The SAP internal programming language of R/3.

ALE

SAP's Application Link Enabling technology that provides distributed processing for R/3 systems and third-party systems through the broadcast and guaranteed delivery of IDOCs to their appropriate destination(s).

ALE eLink to R/3

The BEA eLink Adapter for R/3 ALE component server that processes and submits eLink to R/3 IDOCs to R/3 for ALE processing.

ALE R/3 to eLink

The BEA eLink Adapter for R/3 ALE component client that receives and processes R/3 to eLink IDOCs from R/3.

BAPI

SAP's Business Application Programming Interface, which provides remotely-callable RFCs (methods) that are associated with R/3 Business Objects.

BDC

SAP's Batch Data Control, which processes screen-oriented data in transactions.

BO

SAP Business Object framework that provides an object-oriented model of the enterprise. For example, a Sales Order is represented in R/3 as a Business Object.

BOR

SAP's Business Object Repository that contains the definitions of R/3 Business Objects and their associated BAPIs.

eLink Adapter for R/3 ALE

BEA eLink Adapter for R/3 ALE (application link enabling technology) is the infrastructure of choice for application integration of the SAP R/3 environment. It works with mission-critical, high-performance middleware to enable easy application integration with and real-time access to SAP R/3 transactions, functions and data.

FML

BEA's Field Manipulation Language, a type of message buffer in ATMI. In BEA eLink Adapter for R/3 ALE documentation, FML *always* refers to FML32.

IDOC

SAP's Intermediate Document (a flat file record of data) that is distributed via ALE.

R/3

SAP's client-server product that provides access to the SAP system via a three-tier architecture consisting of database, application, and presentation components.

RFC

SAP's Remote Function Call interface that allows non-R/3 systems to invoke remotely callable ABAP/4 functions. Synchronous RFC, transactional RFC (tRFC), and asynchronous RFC (aRFC) styles are supported.

SAPGUI

SAP's graphical user interface utility.

TID

Transaction ID for tracking IDOC transactions.

Index

A

about

- ALE eLink to R/3 2-8
- ALE integration 2-1
- ALE R/3 to eLink 2-18
- BEA eLink Adapter for R/3 1-1
- destination map files 2-23
- FML field definitions 1-6
- intermediate documents (IDOCs) 2-5
- transaction IDs (TIDs) 2-7

ALE

- about ALE integration 2-1
- information flow 2-4
- key concepts 2-5

ALE eLink to R/3

- about ALE eLink to R/3 2-8
- configuring 6-2
- CR3_IDOC_IN service 2-8
- CR3_SUBMIT service 2-8
- eLink to R/3 connections, configuring
 - 7-1
- environment variables 6-4
- environment variables for connections
 - 7-2
- FML32 field definitions 2-8
- handling problems 2-16
- information flow 2-9
- load balancing 2-12, 6-8
- logging 6-7
- queues
 - configuring 6-20

- splitting IDOC packets 6-7

- TID file 2-13, 6-7

- transactional integrity 2-13

- UBB file 6-2

ALE R/3 to eLink

- about ALE R/3 to eLink 2-18
- command line options 6-10
- compressing R/3 to eLink IDOCs 2-23
- compressing R/3 to eLink IDOCs 6-16
- configuring 6-9
- data-dependent routing
 - environment variable 2-28, 6-13
 - in destination map file 2-23, 6-16
- destination map file 6-14
- destination map files 2-23
- enqueueing into multiple queues 2-22
- environment variables 6-11
- handling problems 2-34
- information flow 2-18
- load balancing 2-28, 6-18
- logging 6-17
- program IDs, multiple 6-18
- queues
 - configuring 6-22
- R/3 to eLink connections, configuring
 - 7-5
- registering multiple program IDs 2-30
- splitting IDOC packets 2-21, 6-16
- starting 6-9
- startup program 6-9
- TID file 2-31, 6-17

transactional integrity 2-31

B

batch files

cr3aleout.bat 6-12

BEA eLink Adapter for R/3

about 1-1

benefits 1-2

C

command line options

ALE R/3 to eLink 6-10

compressing R/3 to eLink IDOCs 2-23, 6-16

configuring

ALE eLink to R/3 6-2

ALE R/3 to eLink 6-9

R/3 connections

eLink to R/3 7-1

R/3 to eLink 7-5

connections

eLink to R/3 connections, configuring
7-1

R/3 to eLink connections, configuring
7-5

troubleshooting 7-6

cr3_ale.ubb file 5-1

CR3_ALE_DEFAULT_IDOC_SPLIT

environment variable 6-11

CR3_ALE_DEFAULT_TARGET_ID

environment variable 6-11

CR3_ALE_TARGET_QUEUE environment

variable 6-11

CR3_ALE_TARGET_QUEUE_SPACE

environment variable 6-11

CR3_ALEIN_ERROR_QUEUE

environment variable 6-5

CR3_ALEIN_ERROR_QUEUE_SPACE

environment variable 6-5

CR3_ALEIN_IDOCIN_QUEUE

environment variable 6-5

CR3_ALEIN_IDOCIN_QUEUE_SPACE

environment variable 6-5

CR3_ALEIN_PRINT_FILE environment
variable 6-5

CR3_ALEIN_PRINT_OPTION

environment variable 6-5

CR3_ALEIN_TID_FILE environment

variable 6-4

CR3_ALEIN_TID_PER_IDOC

environment variable 6-5

CR3_ALEIN_TRACE environment variable
6-5

CR3_ALEOUT_DEST_MAP_FILE

environment variable 6-11

CR3_ALEOUT_PRINT_FILE environment
variable 6-12

CR3_ALEOUT_PRINT_OPTION

environment variable 6-12

CR3_ALEOUT_TID_FILE environment

variable 6-12

CR3_ALEOUT_TRACE environment

variable 6-12

CR3_CLIENT environment variable 7-2

CR3_DESTINATION environment variable
7-2

CR3_ERROR_TEXT 1-7

CR3_IDOC 1-7

CR3_IDOC_CONTROL 1-7

CR3_IDOC_DATA 1-7

CR3_IDOC_IN service 2-8

CR3_LANGUAGE environment variable
7-2

CR3_PASSWORD environment variable 7-2

CR3_RFC_TID 1-7

CR3_SUBMIT service 2-8

CR3_TARGET_ID 1-7

CR3_USER environment variable 7-2

cr3alein.env 6-6

cr3aleout.bat file 6-12

customer support contact information iii

D

data-dependent routing
 environment variable 2-28, 6-13
 in destination map file 2-23, 6-16
destination map file 6-14
destination map files
 about destination map files 2-23
 examples 2-24
documentation, where to find it ii

E

eLink to R/3 connections, configuring 7-1
ELINK_ADAPTER_ERR 1-7
ELINK_ADAPTER_ERR_CODE 1-7
environment files
 cr3alein.env 6-6
environment variables
 CR3_ALE_DEFAULT_IDOC_SPLIT
 6-11
 CR3_ALE_DEFAULT_TARGET_ID
 6-11
 CR3_ALE_TARGET_QUEUE 6-11
 CR3_ALE_TARGET_QUEUE_SPAC
 E 6-11
 CR3_ALEIN_ERROR_QUEUE 6-5
 CR3_ALEIN_ERROR_QUEUE_SPAC
 E 6-5
 CR3_ALEIN_IDOCIN_QUEUE 6-5
 CR3_ALEIN_IDOCIN_QUEUE_SPA
 CE 6-5
 CR3_ALEIN_PRINT_FILE 6-5
 CR3_ALEIN_PRINT_OPTION 6-5
 CR3_ALEIN_TID_FILE 6-4
 CR3_ALEIN_TID_PER_IDOC 6-5
 CR3_ALEIN_TRACE 6-5
 CR3_ALEOUT_DEST_MAP_FILE
 6-11
 CR3_ALEOUT_PRINT_FILE 6-12
 CR3_ALEOUT_PRINT_OPTION 6-12
 CR3_ALEOUT_TID_FILE 6-12

CR3_ALEOUT_TRACE 6-12
CR3_CLIENT 7-2
CR3_DESTINATION 7-2
CR3_LANGUAGE 7-2
CR3_PASSWORD 7-2
CR3_USER 7-2
for ALE eLink to R/3 6-4
for ALE R/3 to eLink 6-11
SIDE_INFO 7-2

F

files
 cr3_ale.ubb file 5-1
 cr3alein.env 6-6
 cr3aleout.bat 6-12
 sideinfo file 7-3
FML field definitions
 about FML field definitions 1-6
 ALE eLink to R/3 2-8
 CR3_ERROR_TEXT 1-7
 CR3_IDOC 1-7
 CR3_IDOC_CONTROL 1-7
 CR3_IDOC_DATA 1-7
 CR3_RFC_TID 1-7
 CR3_TARGET_ID 1-7
 ELINK_ADAPTER_ERR 1-7
 ELINK_ADAPTER_ERR_CODE 1-7

G

gateway host name 7-6
gateway service name 7-6

I

information flow
 ALE 2-4
 ALE eLink to R/3 2-9
 ALE R/3 to eLink 2-18
installing

on UNIX 4-1
on Windows NT 4-6
intermediate documents (IDOCs), about 2-5

K

key concepts
ALE 2-5

L

load balancing
ALE eLink to R/3 2-12, 6-8
ALE R/3 to eLink 2-28, 6-18
logging
ALE eLink to R/3 6-7
ALE R/3 to eLink 6-17
login user, UNIX 4-2

N

nodes
R/3 installation nodes 3-2
TUXEDO installation node 3-3

P

printing product documentation ii
problems
ALE eLink to R/3 2-16
ALE R/3 to eLink 2-34
program IDs
registering multiple 2-30
program IDs, multiple 6-18
program-Id 7-6

Q

queues
configuring for ALE 6-20
specifying default queues for ALE R/3 to
eLink 6-15

R

R/3
nodes 3-2
R/3 to eLink connections, configuring 7-5
related information iii
requirements
system requirements 3-1
TUXEDO requirements 3-3
RFC eLink to R/3
eLink to R/3 connections, configuring
7-1

S

services
CR3_IDOC_IN service 2-8
CR3_SUBMIT service 2-8
SIDE_INFO environment variable 7-2
sideinfo file 7-3
splitting IDOC packets
ALE eLink to R/3 6-7
ALE R/3 to eLink 2-21, 6-16
starting
ALE R/3 to eLink 6-9
support
technical iii
system requirements 3-1

T

TID file
ALE eLink to R/3 2-13, 6-7
ALE R/3 to eLink 2-31, 6-17
transaction IDs (TIDs), about 2-7
transaction log (TLOG) 5-1
transactional integrity
ALE eLink to R/3 2-13
ALE R/3 to eLink 2-31
troubleshooting
connection problems 7-6
TUXEDO

initialization file (cr3_ale.ubb) 5-1
installation node 3-3
transaction log (TLOG) 5-1

U

UBB file

ALE eLink to R/3 6-2
UNIX listing A-2
Windows NT listing A-4

UNIX

installation instructions 4-1
login user 4-2
system requirements 3-1
UBB file A-2

usage scenarios

ALE integration 2-2

W

Windows NT

installation instructions 4-6
system requirements 3-1
UBB file A-4