# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>8</td>
</tr>
<tr>
<td>Intended Reader</td>
<td>8</td>
</tr>
<tr>
<td>Organization</td>
<td>8</td>
</tr>
<tr>
<td>Nomenclature</td>
<td>9</td>
</tr>
<tr>
<td>Online Use</td>
<td>9</td>
</tr>
<tr>
<td>Writing Conventions</td>
<td>9</td>
</tr>
<tr>
<td><strong>Chapter 1</strong></td>
<td></td>
</tr>
<tr>
<td>Introduction</td>
<td>10</td>
</tr>
<tr>
<td>SAP Interface Options</td>
<td>10</td>
</tr>
<tr>
<td>The ALE Interface</td>
<td>11</td>
</tr>
<tr>
<td>SAP Intermediate Document Format</td>
<td>11</td>
</tr>
<tr>
<td>Nomenclature</td>
<td>11</td>
</tr>
<tr>
<td>Structure</td>
<td>11</td>
</tr>
<tr>
<td>The SAP ALE e*Way</td>
<td>13</td>
</tr>
<tr>
<td>Overview</td>
<td>13</td>
</tr>
<tr>
<td>e*Gate to SAP</td>
<td>13</td>
</tr>
<tr>
<td>SAP to e*Gate</td>
<td>14</td>
</tr>
<tr>
<td>Features</td>
<td>15</td>
</tr>
<tr>
<td>Components</td>
<td>15</td>
</tr>
<tr>
<td>Availability</td>
<td>16</td>
</tr>
<tr>
<td><strong>Chapter 2</strong></td>
<td></td>
</tr>
<tr>
<td>Installation</td>
<td>17</td>
</tr>
<tr>
<td>System Requirements</td>
<td>17</td>
</tr>
<tr>
<td>Environment Configuration</td>
<td>17</td>
</tr>
<tr>
<td>External System Requirements</td>
<td>18</td>
</tr>
<tr>
<td>External Configuration Requirements</td>
<td>18</td>
</tr>
<tr>
<td>Installing the e*Way</td>
<td>19</td>
</tr>
<tr>
<td>Windows Systems</td>
<td>19</td>
</tr>
</tbody>
</table>
Chapter 3

System Implementation 24

Overview 24
Pre-Implementation Tasks 24
Implementation Topics 24
Implementation Sequence 25
Viewing e*Gate Components 25

Creating a Schema 26

Creating Event Types 27

Creating Event Type Definitions 28
Downloading the IDoc Description File 29
Using the Monk ETD Editor’s Build Tool 32
Using the Java IDoc Wizard 37
Assigning ETDs to Event Types 39

Defining Collaborations 41

Creating Intelligent Queues 42

Known Limitations 42
Sending a NAK to SAP 42

Chapter 4

SAP R/3 Configuration 43

Overview 43

SAP Hierarchies 43

Distribution Model Configuration 45
Naming the Logical System 45
Specifying the Distribution Model 47

Communications Configuration 50
Defining the RFC Destination 50
Defining the Communications Port 53
Creating a Partner Profile 55
Configuring a Partner Profile 56

Security Issues 60
Contents

ALE/RFC Setup 88
SAP System Type 88
SAP IDoc Version 88
Program ID (SAP to e*Gate) 88
SAP Load Balancing Usage (e*Gate to SAP) 89
SAP Application Server (e*Gate to SAP) 89
SAP Router String (e*Gate to SAP) 90
SAP System Number (e*Gate to SAP) 90
SAP Gateway Host (SAP to e*Gate) 90
SAP Gateway Host (e*Gate to SAP) 91
SAP Router String (SAP to e*Gate) 91
SAP Gateway Service 91
RFC Trace On 92
Validate SAP Sender Identity (SAP to e*Gate) 92
SAP System Number (SAP to e*Gate) 93
SAP System ID (SAP to e*Gate) 93
SAP Client Number (SAP to e*Gate) 93
SAP User Name (SAP to e*Gate) 93
SAP User Language (SAP to e*Gate) 94
SAP System Client Number (e*Gate to SAP) 94
Logon User Name (e*Gate to SAP) 94
Logon Password (e*Gate to SAP) 94
Logon Language (e*Gate to SAP) 95
SAP Message Server (e*Gate to SAP) 95
SAP Destination System (e*Gate to SAP) 95
SAP Application Server Group (e*Gate to SAP) 95
SAP System Number (SAP to e*Gate) 96
SAP System ID (SAP to e*Gate) 96
SAP Client Number (SAP to e*Gate) 96
SAP User Name (SAP to e*Gate) 96
SAP User Language (SAP to e*Gate) 96
SAP System Client Number (e*Gate to SAP) 96
Logon User Name (e*Gate to SAP) 96
Logon Password (e*Gate to SAP) 96
Logon Language (e*Gate to SAP) 97
SAP Message Server (e*Gate to SAP) 97
SAP Destination System (e*Gate to SAP) 97
SAP Application Server Group (e*Gate to SAP) 97
Communication Setup 96
Time to Wait Before Becoming Server (SAP to e*Gate) 96
e*Gate System Polling Interval (SAP to e*Gate) 96
Time to Wait Before Attempting Connection (e*Gate to SAP) 96
Connection Duration Strategy (e*Gate to SAP) 96
Idle Interval Before Disconnecting (e*Gate to SAP) 97
Connection Error Resend Strategy (e*Gate to SAP) 97
Maximum Resend Attempts (e*Gate to SAP) 98
Transaction Processing 99
Enforce Transactional RFC 99
Transactional ID Verification Database (SAP to e*Gate) 99
Transactional ID Verification Database (e*Gate to SAP) 100
Verification Database Operating Mode 100
Database Purge Schedule 100
Database Entry Lifetime 101
Expunge Database Entry On Confirm 101
Transaction Identification and Translation (SAP to e*Gate) 102
Transaction Identification and Translation (e*Gate to SAP) 102
Translation Status Monk Variable 103
Output File Monk Variable 104
Output Record Delimiter Monk Variable 104
Persistent Memory File Variable 105
Persistent Memory File Name (SAP to e*Gate) 105
SAP IDoc Syntax Error Code (SAP to e*Gate) 107
SAP IDoc Translation Error Code (SAP to e*Gate) 107
SAP e*Way General Error Code (SAP to e*Gate) 107
Advanced Monk Setup 108
Load Custom Monk File Name 108
Startup Monk Function Name (SAP to e*Gate) 108
Positive Acknowledgment Monk Function (SAP to e*Gate) 108
Positive Acknowledgment Function (e*Gate to SAP) 109
Negative Acknowledgment Function (SAP to e*Gate) 109
Negative Acknowledgment Function (e*Gate to SAP) 109
Appendix A

Utilities 110
stcjdump.exe 110

Appendix B

IDoc Java Methods 111
next 111
available 111
send 112
reset 112
getIdocMsgFormat 112
setIdocMsgFormatALE 113
setIdocMsgFormatEDI 113
idocMsgFormatIs 113

Index 115
Preface

This Preface contains information regarding the User’s Guide itself.

P.1 Intended Reader

The reader of this guide is presumed to be a developer or system administrator with responsibility for maintaining the SeeBeyond™ e*Gate™ Integrator system, and have a working knowledge of:

- Operation and administration of the appropriate operating systems (see Availability on page 16)
- Windows-style GUI operations
- SAP ALE and IDoc format
- Integrating SAP R/3 with external systems

P.2 Organization

This User’s Guide is organized into two parts. The first part, consisting of Chapters 1-5, introduces the e*Way and describes the procedures for installing the e*Way and implementing a working system incorporating the e*Way. This part should be of particular interest to a System Administrator or other user charged with the task of getting the system up and running.

The second part, consisting of Chapters 6-7 and the Appendix, describes the architecture and internal functionality of the e*Way. This part should be of particular interest to a Developer involved in customizing the e*Way for a specific purpose. Information contained in this part that is necessary for the initial setup of the e*Way is cross-referenced in the first part of the guide, at the appropriate points in the procedures.
Nomenclature

Note that for purposes of brevity, the e*Way Intelligent Adapter for SAP (ALE) is frequently referred to as the SAP ALE e*Way, or simply the e*Way.

Online Use

This User’s Guide is provided in Adobe Acrobat’s Portable Document Format (PDF). As such, it can be printed out on any printer or viewed online. When viewing online, you can take advantage of the extensive hyperlinking imbedded in the document to navigate quickly throughout the Guide.

Hyperlinking is available in:

- The Table of Contents
- The Index
- Within the chapter text, indicated by blue print

Existence of a hyperlink hotspot is indicated when the hand cursor points to the text. Note that the hotspots in the Index are the page numbers, not the topics themselves. Returning to the spot you hyperlinked from is accomplished by right-clicking the mouse and selecting Go To Previous View on the resulting menu.

Writing Conventions

The writing conventions listed in this section are observed throughout this document.

Monospaced (Courier) Font

Computer code and text to be typed at the command line are set in Courier as shown below:

```
Configuration for BOB_Promotion
java -jar ValidationBuilder.jar
```

Variables within a command line are set in the same font within brackets <> as shown below:

```
stcregutil -rh <host-name> -un <user-name> -up <password> -sf
```

Bold Sans-serif Font

- User Input: Click Apply to save, or OK to save and close.
- File Names and Paths: In the Open field, type D:\setup\setup.exe.
- Parameter, Function, and Command Names: The default parameter localhost is normally only used for testing; the Monk function iq-put places an Event into an IQ.
Introduction

The SeeBeyond e*Way Intelligent Adapter for SAP (ALE) has been designed specifically to connect e*Gate to SAP enterprise-management software within a network of diverse hardware and software systems. Using one or more SAP e*Ways, e*Gate can act as a bus, linking SAP applications and other software systems, or differently-configured SAP systems. This e*Way allows bidirectional data exchange between e*Gate and an SAP system via SAP’s Application Link Enabling (ALE) interface method, which employs SAP’s Intermediate Document (IDoc) Format.

1.1 SAP Interface Options

SAP offers several interface options, including Application Link Enabling (ALE), Business Application Programming Interface (BAPI), and Electronic Data Interchange (EDI). The Batch Data Communication (BDC) interface actually is a user-emulation method that can be either batch or event-driven.

Figure 1  SAP Interface Options

All of these interfacing methods are supported by SeeBeyond, for both SAP-inbound and SAP-outbound data transfer.
1.1.1 The ALE Interface

Real-time communications with an SAP system is accomplished through the use of SAP’s ALE layer running on top of SAP’s Remote Function Call (RFC)—SAP’s version of Remote Procedure Call. To guarantee uniqueness of transactions, SAP offers a version of RFC called Transactional RFC (tRFC). SAP provides the API libraries for:

- Enabling connection to the SAP system, given the appropriate Host and Identification parameters
- Marshalling of arguments to and from the SAP system
- Executing (client-mode) or defining (server-mode) RFC Services on the SAP system

ALE supports the transfer of information between applications by means of messaging, rather than file transfer. Transactions are exchanged using the SAP Intermediate Document (IDoc) format, which is basically a fixed message. The IDoc file is interpreted by correlating with an IDoc Description file, which specifies all relevant fields.

1.2 SAP Intermediate Document Format

Intermediate Documents are used as containers for information, and are used to upload data to and download data from other systems. IDocs allow independence between the format and content of the message.

1.2.1 Nomenclature

Several hundred IDocs are supplied with each R/3 system, serving as templates for a wide variety of applications. The IDoc hierarchy is represented by the following terminology:

- **Message Types** are related to specific applications such as Orders.
- **IDoc Types** are different versions of standard Message Types, such as Orders for specific items or services.

1.2.2 Structure

A typical SAP IDoc consists of Control, Data, and Status records, as shown in Figure 2 on page 12. Status records, however, are not used by the ALE interface.
### Figure 2  SAP IDoc Structure

**Control Record**

<table>
<thead>
<tr>
<th>Sender</th>
<th>Recvr</th>
<th>Messg. Type</th>
<th>IDoc-Type</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Data Record/Segments**

| HEADER1      | XXXXXXXXXXXXXXX |
| ITEMS        | XXXXXXXXXXXXXXXXXXXX |
| SUBITEMA     | XXXXX |
| SUBITEMB     | XXXXX |
| TEXT         | XXXXXXXXXXXX |
| ITEMS        | XXXXXXXXXXXXXXXXXXXX |
| SUBITEMC     | XXXXX |
| SUBITEMD     | XXXXX |
| ACCUM        | XXXXXXXXXXXXXXXX |

**Status Record**

- 'To be processed' 14:33:48
- 'Processing successful' 14:33:57
1.3 The SAP ALE e*Way

The SAP e*Way controls the communication protocol layer between the SAP host and e*Gate, and can be configured to process data in either direction. As with most other SeeBeyond e*Ways, it contains its own Monk engine to process Collaborations without drawing on e*Gate resources.

The e*Gate system, in turn, connects to another application through an e*Way designed specifically for that system. This external system may be either another (differently-configured) SAP system or a non-SAP system.

1.3.1 Overview

e*Gate to SAP

![Diagram of ALE SAP-Inbound Data Flow]
An application external to the SAP system generates a transaction designated for an SAP application. The e*Gate system receives the transaction through an e*Way, performs any necessary processing or routing, and sends the information to the SAP ALE e*Way. This e*Way converts the data to SAP IDoc format and sends it to the SAP system’s ALE Interface. Here, it is converted to the correct internal data format and stored in the application database.

**SAP to e*Gate**

**Figure 4  ALE SAP-Outbound Data Flow**

During routine operation, some application on the SAP system generates a transaction designated for an external system. The ALE Interface converts the data from the internal data format to the IDoc format, and sends it to the SAP ALE e*Way. The e*Gate system receives the transaction from the e*Way, performs any necessary processing or routing, and sends the information to another e*Way connected to the recipient system. Here, it is converted to the correct format for the target application.
1.3.2 **Features**

The SAP ALE e*Way provides the following:

- Inbound/outbound access to SAP R/3 System via ALE (Application Linking Enabling).
- Monk-enabled connection to the SAP system, given the appropriate host and identification parameters.
- Emulates/calls ABAP/4 INBOUND_IDOC_PROCESS function on SAP R/3 System using Transactional RFC (tRFC).
- Executing (client-mode) or defining (server-mode) RFC Services on the SAP system.

1.3.3 **Components**

The SAP ALE e*Way incorporates the following components:

- An executable file, stcewsapale.exe
- An accompanying dynamic load library, librfc32.dll (lbrfccm.sl for UNIX), containing RFC extensions
- Default configuration files, stcewsapalein.def and stcewsapaleout.def
- SeeBeyond utilities stcjdump.exe and stcpxcmd.exe
- Monk and Java-based Wizards used to build Event Type Definitions

For a list of installed files, see **Chapter 2**.
1.3.4 Availability

The e*Way Intelligent Adapter for SAP ALE currently supports the following combinations of operating systems and releases of SAP R/3.

### English

**Table 1** English-language Version

<table>
<thead>
<tr>
<th>Operating System</th>
<th>4.0B</th>
<th>4.5B</th>
<th>4.6B</th>
<th>4.6C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 2000 SP1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Windows 2000 SP2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Windows NT 4.0 SP6a</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Solaris 2.6</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Solaris 7</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Solaris 8</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>AIX 4.3.3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>AIX 5.1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>HP-UX 11.0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>HP-UX 11i</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Compaq Tru64 4.0F</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Compaq Tru64 5.0A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

### Japanese

**Table 2** Japanese-language Version

<table>
<thead>
<tr>
<th>Operating System</th>
<th>4.0B</th>
<th>4.5B</th>
<th>4.6B</th>
<th>4.6C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows NT 4.0 SP6a (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Windows 2000 SP1 (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Windows 2000 SP2 (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Solaris 2.6 (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Solaris 7 (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Solaris 8 (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>HP-UX 11.0 (Japanese)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
</tbody>
</table>

### Korean

**Table 3** Korean-language Version

<table>
<thead>
<tr>
<th>Operating System</th>
<th>4.0B</th>
<th>4.5B</th>
<th>4.6B</th>
<th>4.6C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solaris 8 (Korean)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>AIX 4.3.3 (Korean)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>HP-UX 11.0 (Korean)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
</tbody>
</table>
Installation

This chapter describes the requirements and procedures for installing the e*Way software. Procedures for implementing a working system, incorporating instances of the e*Way, are described in Chapter 3.

Note: Please read the readme.txt file located in the addons\ewsapale directory on the installation CD-ROM for important information regarding this installation.

2.1 System Requirements

To use the e*Way Intelligent Adapter for SAP ALE, you need the following:

1. An e*Gate Participating Host, version 4.5.1 or later.
3. Sufficient free disk space to accommodate e*Way files:
   - Approximately 2.6 MB on Windows systems
   - Approximately 4.6 MB on Solaris systems
   - Approximately 5.4 MB on HP-UX systems
   - Approximately 6.3 MB on AIX systems
   - Approximately 5.9 MB on Tru64 UNIX systems

Note: Additional disk space is required to process and queue the data that this e*Way processes; the amount necessary varies, based on the type and size of the data being processed.

2.1.1 Environment Configuration

No changes are required to the Participating Host’s operating environment to support this e*Way.
2.2 External System Requirements

The e*Way Intelligent Adapter for SAP ALE supports the following applications (see also Availability on page 16).

English

- SAP R/3, version 4.0B, 4.5B, 4.6B, and 4.6C

Japanese

- SAP R/3, version 4.6C

Korean

- SAP R/3, version 4.6C

2.2.1 External Configuration Requirements

Your SAP R/3 system must be configured to communicate with the SAP ALE e*Way (see SAP R/3 Configuration on page 43).
2.3 Installing the e*Way

2.3.1 Windows Systems

Installation Procedure

Note: The installation utility detects and suggests the appropriate installation directory. Use this directory unless advised otherwise by SeeBeyond.

To install the e*Way on a Microsoft Windows system

1. Log in as an Administrator on the workstation on which you want to install the e*Way (you must have Administrator privileges to install this e*Way).

2. Exit all Windows programs and disable any anti-virus applications before running the setup program.

3. Insert the e*Way installation CD-ROM into the CD-ROM drive.

4. Launch the setup program.

   A If the CD-ROM drive’s Autorun feature is enabled, the setup program should launch automatically. Follow the on-screen instructions until the Choose Product dialog box appears (see Figure 5). Check Add-ons, then click Next.

   Figure 5 Choose Product Dialog Box

   B If the setup program does not launch automatically, use the Windows Explorer or the Control Panel’s Add/Remove Applications feature to launch the following file on the CD-ROM drive (bypassing the Choose Product dialog):

   setup\addons\setup.exe
5 Follow the on-screen instructions until the Select Components dialog box appears (see Figure 6). Highlight—but do not check—eWays and then click Change.

**Figure 6** Select Components Dialog Box

![Select Components Dialog Box](image)

6 When the Select Sub-components dialog box appears (see Figure 7), check the SAP ALE e*Way.

**Figure 7** Select e*Way Dialog

![Select Sub-components Dialog Box](image)

7 Click Continue, and the Select Components dialog box reappears.

8 Click Next and continue with the installation.
Subdirectories and Files

By default, the InstallShield installer creates the following subdirectories and installs the following files within the \eGate\client tree on the Participating Host, and the \eGate\Server\registry\repository\default tree on the Registry Host.

<table>
<thead>
<tr>
<th>Subdirectories</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>\bin\</td>
<td>librfc32.dll stcewsapale.ctl stcewsapale.exe stcjdump.exe stcpxcmd.exe stcsapconvert.exe</td>
</tr>
<tr>
<td>\configs\stcewsapale\</td>
<td>sapAleIn3.6To4.1Rule.txt sapAleOut3.6To4.1Rule.txt stcewsapalein.def stcewsapaleout.def</td>
</tr>
<tr>
<td>\etd\</td>
<td>idocwizard.ctl</td>
</tr>
<tr>
<td>\stcgui\ctls\</td>
<td>guisapale.ctl</td>
</tr>
</tbody>
</table>

By default, the InstallShield installer also installs the following file within the \eGate\Server\registry\repository\default tree on the Registry Host.

<table>
<thead>
<tr>
<th>Subdirectories</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>\</td>
<td>sapale.ctl</td>
</tr>
</tbody>
</table>
2.3.2 UNIX Systems

Installation Procedure

Note: You are not required to have root privileges to install this e*Way. Log on under the user name that you wish to own the e*Way files. Be sure that this user has sufficient privilege to create files in the e*Gate directory tree.

To install the e*Way on a UNIX system

1. Log onto the workstation containing the CD-ROM drive and, if necessary, mount the drive.
2. Insert the CD-ROM into the drive.
3. At the shell prompt, type
   
   ```
   cd /cdrom
   ```
4. Start the installation script by typing:
   
   ```
   setup.sh
   ```
5. A menu appears, with several options. Select the install e*Way option, and follow any additional on-screen instructions.

Note: The installation utility detects and suggests the appropriate installation directory. Use this directory unless advised otherwise by SeeBeyond. Note also that no spaces should appear in the installation path name.

Subdirectories and Files

The preceding installation procedure creates the following subdirectories and installs the following files within the /eGate/client tree on the Participating Host, and the /eGate/Server/registry/repository/default tree on the Registry Host.

Table 6 Participating Host & Registry Host

<table>
<thead>
<tr>
<th>Subdirectories</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>/bin/</td>
<td>lbrfccm.sl (HP-UX only)</td>
</tr>
<tr>
<td></td>
<td>stcewsapale</td>
</tr>
<tr>
<td></td>
<td>stcjdump</td>
</tr>
<tr>
<td></td>
<td>stcpxcmd</td>
</tr>
<tr>
<td>/configs/stcewsapale/</td>
<td>sapAleIn3.6To4.1Rule.txt</td>
</tr>
<tr>
<td></td>
<td>sapAleOut3.6To4.1Rule.txt</td>
</tr>
<tr>
<td></td>
<td>stcewsapalein.def</td>
</tr>
<tr>
<td></td>
<td>stcewsapaleout.def</td>
</tr>
</tbody>
</table>
The preceding installation procedure also installs the following file only within the `/eGate/Server/registry/repository/default` tree on the Registry Host.

**Table 7** Registry Host Only

<table>
<thead>
<tr>
<th>Subdirectories</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>/</td>
<td>sapale.ctl</td>
</tr>
</tbody>
</table>
Chapter 3

System Implementation

In this chapter we take a more detailed look at the information presented in the Introduction, and describe the steps required for setting up a working system incorporating the SeeBeyond e*Way Intelligent Adapter for SAP (ALE). Please refer to the e*Gate Integrator User’s Guide for additional information.

3.1 Overview

This e*Way provides a specialized transport component for incorporation in an operational schema. The schema also contains Collaborations, linking different data or Event types, and Intelligent Queues. Typically, other e*Way types also are used as components of the schema.

3.1.1 Pre-Implementation Tasks

- Installation of SeeBeyond software
  The first task, of course, is to install the SeeBeyond software as described in Chapter 2.
- Modification of SAP R/3 environment
  After you have installed the SeeBeyond software, your existing SAP R/3 system must be configured to interact properly with the SAP ALE e*Way. See Chapter 4.

3.1.2 Implementation Topics

The following topics are discussed in this chapter:

- Creating a Schema on page 26
- Creating Event Types on page 27
- Creating Event Type Definitions on page 28
- Defining Collaborations on page 41
- Creating Intelligent Queues on page 42
- Known Limitations on page 42
3.1.3 Implementation Sequence

1. The first step is to create a new Schema—the subsequent steps apply only to this Schema (see Creating a Schema on page 26).

2. The second step is to define the Event Types you are transporting and processing within the Schema (see Creating Event Types on page 27).

3. Third, you need to associate the Event Types created in the previous step with Event Type Definitions (ETDs) derived from the applicable Business Rules (see Creating Event Type Definitions on page 28).

4. The fourth step is to create and configure the required e*Ways (see Chapter 5).

5. Next is to define and configure the Collaborations linking the Event Types from step 2 (see Defining Collaborations on page 41).

6. Now you need to create Intelligent Queues to hold published Events (see Creating Intelligent Queues on page 42).

7. Finally, you must test your Schema. Once you have verified that it is working correctly, you may deploy it to your production environment.

3.1.4 Viewing e*Gate Components

Use the Navigator and Editor panes of the e*Gate Enterprise Manager to view the various e*Gate components. Note that you may only view components of a single schema at one time, and that all operations apply only to the current schema. All procedures in this chapter should be performed while displaying the Components Navigator pane. See the e*Gate Integrator User’s Guide for a detailed description of the features and use of the Enterprise Manager.
Chapter 3
System Implementation

Section 3.2
Creating a Schema

3.2 Creating a Schema

A schema is the structure that defines e*Gate system parameters and the relationships between components within the e*Gate system. Schemas can span multiple hosts.

Because all setup and configuration operations take place within an e*Gate schema, a new schema must be created, or an existing one must be started before using the system. Schemas store all their configuration parameters in the e*Gate Registry.

To select or create a schema

1. Invoke the Open Schema dialog box (Figure 8) and Open an existing schema, or click New to create a new schema.

   Figure 8  Open Schema Dialog

   ![Open Schema Dialog](image)

2. Clicking New invokes the New Schema dialog box (Figure 9).

   Figure 9  New Schema Dialog

   ![New Schema Dialog](image)

3. Enter a new schema name and click Open.

4. The e*Gate Enterprise Manager then opens under your new schema name.

5. From the Options menu, click on Default Editor and select Monk.

6. Select the Components tab, found at the bottom of the Navigator pane of the e*Gate Enterprise Manager window.

7. You are now ready to begin creating the necessary components for this new schema.
3.3 Creating Event Types

Within e*Gate, messages and/or packages of data are defined as Events. Each Event must be categorized into a specific Event Type within the schema.

To define the Event Types

1. In the e*Gate Enterprise Manager’s Navigator pane, select the Event Types folder.

2. On the Palette, click the New Event Type button.

3. In the New Event Type Component box, enter the name for the input Event Type and click Apply. Use this method to create all required Event Types, for example:
   - InboundEvent
   - ValidEvent
   - InvalidEvent

4. After you have created the final Event Type, click OK.
3.4 Creating Event Type Definitions

Before e*Gate can process any data to or from an SAP system, you must create an Event Type Definition to package and route that data within the e*Gate system. The ETD Editor automatically creates an Event Type Definition file based upon structural metadata, using a Wizard-driven build tool.

With e*Gate release 4.5, two options are available: a Monk IDoc Converter, and a Java IDoc Wizard. You choose the method by selecting the default ETD editor (either Monk or Java) from the Enterprise Manager’s Options menu.

See the e*Gate Integrator User’s Guide for additional information about Event Type Definitions and the e*Gate ETD Editor.
3.4.1 Downloading the IDoc Description File

You must download the IDoc Description file to the host where the SAP ALE e*Way is being run from within the SAP application. To download the IDoc Description File from SAP to the e*Way:

Transaction: WE63

**Figure 11**  SAP R/3 System Window

1 Type WE63 into the command field of the SAP R/3 System window and select , which displays the Documentation IDoc Record Types and IDoc Types (Parser) window (alternatively, you can execute RSE IDOC3 in SE38).
2 Type the IDoc name for which you wish to generate a Description File into the command field, or locate by invoking the drop-down menu for existing IDoc types.

3 Make sure the Control record, Data record, and Status record boxes are all checked.

4 Select to begin parsing the IDoc message.
5 After the Description File appears, select List > Download from the Menu bar, which displays the Save list in file... dialog box.

Figure 14  Save List in File... Dialog Box

6 Select the unconverted option, since no format conversion is required.

7 Select ✅, which displays the Transfer List to a Local File window.
3.4.2 Using the Monk ETD Editor’s Build Tool

The Event Type Definition Editor’s Build tool automatically creates an Event Type Definition file based upon sample data. Use this procedure to create an Event Type Definition based upon the data your installation requires.

**Note:** Be sure to set the Default Editor to Monk, from the Options menu in the e*Gate Enterprise Manager.

To create an Event Type Definition using the Build tool

1. Launch the ETD Editor by clicking in the e*Gate Enterprise Manager tool bar.
2. On the ETD Editor’s Toolbar, click Build, which displays the initial Build an Event Type Definition dialog box.

**Figure 16** Initial Build an Event Type Definition Dialog Box

3. In the File name box, type the name of the ETD file you wish to build.

**Note:** The Editor automatically supplies the .ssc extension.

8. Enter the name and path of the local file to receive the IDoc Description File.
9. Selecting Transfer downloads the file and returns you to the SAP R/3 System window.
4 Click Next.

**Figure 17** Second Build an Event Type Definition Dialog Box

![Build an Event Type Definition](image)

5 In the **Input file** box, type the name of the input file (Optional).

6 Select the **Build from Library Converter** option to display the list of installed converters.

7 From the list, select the **SAP IDOC, DXOB Converter**.

8 In the **Additional Command Line Arguments** box, type any additional arguments, if desired.

9 Click Finish, and the **IDOC, DXOB Converter Wizard** appears.
Chapter 3  
System Implementation  

Section 3.4  
Creating Event Type Definitions  

Figure 18  SAP IDoc, DXOB Converter Wizard (1)

10  Select the IDoc option.

11  Select or enter the IDoc Version and the SAP R/3 IDoc Release.

Note:  An IDoc Release refers to the corresponding SAP R/3 release, whereas an IDoc Version specifies whether it’s the older, shorter format or the newer, longer format (see the following table).

<table>
<thead>
<tr>
<th>SAP R/3 Release</th>
<th>IDoc Version</th>
<th>Control Record Length (EDI_DC)</th>
<th>Data Record Length (EDI_DD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.0C to 3.0H</td>
<td>2</td>
<td>464</td>
<td>1055</td>
</tr>
<tr>
<td>4.0A and above</td>
<td>3</td>
<td>524</td>
<td>1063</td>
</tr>
</tbody>
</table>

12  Select Terse or Verbose commenting.

Note:  Do not select the Verbose option while attempting to use the SAP IDoc/DXOB Converter in a meaningful way, since it leads to copious debugging (this is a result of the STC_TRACEMASK variable setting).

13  Select the Character Set, if applicable.
Note: If you use the English version of Microsoft Windows, ASCII is the only Character Set option. If you use the Japanese version of Microsoft Windows, you also have the option of using SJIS.

14 Selecting Next displays the following screen.

**Figure 19** SAP IDoc, DXOB Converter Wizard (2)

Enter the path of the IDoc Description File you have downloaded to the e*Gate host into the text box, or locate using Browse. (Remember that the file must be located on the local file system.)

16 The Validation Collaboration ID Rule Name is generated automatically. If you need to modify it for some reason, you may do so. Then select Next, which displays the following information:
Since you are configuring for ALE-mode operation, select **No**.

Click **Finish** to invoke the conversion process.

The **Event Type Definition Editor** window is then displayed, showing the Event Type Definition for the selected file. Refer to the SeeBeyond guide *Creating an End-to-End Scenario with e*Gate Integrator* for instructions on defining Collaboration Rules.
3.4.3 Using the Java IDoc Wizard

The Java ETD Editor contains a Java-based IDoc Wizard, which takes an IDoc and converts it to a .xsc file.

To create an Event Type Definition using the IDoc Wizard

1. Start the e*Gate Enterprise Manager, and open the schema for which you want to create an IDoc ETD.
2. From the Options menu, select Default Editor > Java.
3. Launch the ETD Editor.
4. Select New on the Java ETD Editor’s toolbar. The New Event Type Definitions window appears, displaying all installed ETD Wizards.
5. Invoke the IDoc Wizard by selecting its icon (Figure 22).

**Figure 22**  IDoc Wizard Icon

![IDoc Wizard Icon](image)

The initial Wizard dialog window now appears (see Figure 23).

**Figure 23**  IDoc Wizard - Introduction

![IDOC Wizard - Introduction](image)

Welcome to the IDOC Wizard

The IDOC Wizard helps you create an XSC file from an IDOC file.

Follow these steps:
1. Specify the IDOC file from which to create an XSC file.
2. Enter a package name where all of the Java source files are generated.
3. Select the SAP R/3 IDoc Segment Release being used.
4. Select the format of the Data Records in the IDoc Message.

6. Select Next.
Figure 24  IDoc Wizard - Step 1

7 Enter a Package Name for the container in which the Wizard places the generated Java classes.

8 Enter an IDoc File Name for the IDoc description file to be used to create this IDoc.

9 Select the SAP segment release version to be used.

10 Click Next to view the next Wizard dialog box.

Figure 25  IDoc Wizard - Step 2

11 Click Finish to view the ETD.
3.4.4 Assigning ETDs to Event Types

After you have created the e*Gate system’s ETD files, you can assign them to Event Types you have already created.

To assign ETDs to Event Types

1. In the Enterprise Manager window, select the Event Types folder in the Navigator/Components pane.
2. In the Editor pane, select one of the Event Types you created.
3. Right-click on the Event Type and select Properties (or click \(e\)) in the toolbar. The Event Type Properties dialog box appears. See Figure 27.

The message format for the IDoc is stored in the resulting Java file, shown in this example as IDOC_DEBMAS03_4X_46C, in the field idocMsgFormat. This field determines how data is marshaled or unmarshaled. If you want to change the type of data that is being processed in the collaboration, you can use one of the set methods; however, you must make sure these set methods are called before calling the marshal and unmarshal methods.

See Appendix B for descriptions of the Java methods associated with the SAP IDoc.
Figure 27  Event Type Properties Dialog Box

4 Under Event Type Definition, click Find, and the Event Type Definition Selection dialog box appears (it is similar to the Windows Open dialog box).

5 Open the monk_scripts\common folder, then select the desired file name (*.ssc).

6 Click Select. The file populates the Event Type Definition field.

7 To save any work in the properties dialog box, click Apply to enter it into the system.

8 When finished assigning ETDs to Event Types, click OK to close the properties dialog box and apply all the properties.

Each Event Type is associated with the specified Event Type Definition.
3.5 Defining Collaborations

After you have created the required Event Type Definitions, you must define a Collaboration to transform the incoming Event into the desired outgoing Event.

Collaborations are e*Way components that receive and process Event Types, then forward the output to other e*Gate components. Collaborations consist of the Subscriber, which “listens” for Events of a known type or from a given source, and the Publisher, which distributes the transformed Event to a specified recipient. The same Collaboration cannot be assigned to more than one e*Gate component.

Figure 28 Collaborations

The Collaboration is driven by a Collaboration Rule script, which defines the relationship between the incoming and outgoing ETDs. You can use an existing Collaboration Rule script, or use the Monk programming language to write a new Collaboration Rule script. Once you have written and successfully tested a script, you can then add it to the system’s run-time operation.

Collaborations are defined using the e*Gate Monk Collaboration Rules Editor. See the e*Gate Integrator User’s Guide for instructions on using this Editor. The file extension for Monk Collaboration Rules is .tsc; for Java Collaboration Rules, .xts.
3.6 Creating Intelligent Queues

IQs are components that provide nonvolatile storage for Events within the e*Gate system as they pass from one component to another. IQs are intelligent in that they are more than just a “holding tank” for Events. They actively record information about the current state of Events.

Each schema must have an Q Manager before you can add any IQs to it. You must create at least one IQ per schema for published Events within the e*Gate system. Note that e*Ways that publish Events externally do not need IQs.

For more information on how to add and configure IQs and IQ Managers, see the e*Gate Integrator System Administration and Operations Guide. See the e*Gate Integrator Intelligent Queue Services Reference Guide and the SeeBeyond JMS Intelligent Queue User’s Guide for complete information on working with IQs.

3.7 Known Limitations

3.7.1 Sending a NAK to SAP

Problem

When the e*Way detects that an Inbound IDoc (from SAP) cannot be posted successfully to an e*Gate IQ, it has no choice but to throw an exception with the RfcRaise API, consisting of the string “05” (error during translation), back to the sending SAP R/3 system. However, SAP only accepts a positive acknowledgment, accomplished by calling the RfcSendData API; otherwise, it just marks the IDoc status as “03” (data passed to port OK). Thus, there appears to be no automatic way for the SAP ALE e*Way to set the IDoc status to a desired value.

Possible Solution

Try using the tRFC capabilities now in stc_monksap.dll to post the desired failed status of the IDoc by calling the IDOC_INBOUNDASYNCHRONOUS function (on the SAP R/3 server) to send it an ALEAUD IDoc (ALE Audit message type). These calls can be invoked via the Negative Acknowledgment Monk Function declared under the Advanced Monk Setup section of the ALE Inbound e*Way default configuration, stcewsapalein.def.
This chapter describes procedures for preparing your SAP R/3 system to interact with e*Gate.

4.1 Overview

Before you have an operational SAP-e*Gate system, your existing SAP R/3 system must be configured to interact properly with the SAP ALE e*Way. This configuration is non-intrusive, and does not interfere with other SAP R/3 operations.

The topics described in this chapter include the following:

- SAP Hierarchies on page 43
- Naming the Logical System on page 45
- Specifying the Distribution Model on page 47
- Defining the RFC Destination on page 50
- Defining the Communications Port on page 53
- Creating a Partner Profile on page 55
- Configuring a Partner Profile on page 56
- Security Issues on page 60

Note: The screen captures shown in this chapter correspond to SAP Frontend Server release 4.5B, and R/3 Kernel release 4.0B. They are shown simply to illustrate the general nature of the procedures, and contain only example values. They are not meant to replace the SAP documentation supplied with your system. Please refer to your SAP documentation to determine the exact procedures and corresponding appearance of the GUI for your installation.

4.2 SAP Hierarchies

To prepare your SAP R/3 system to recognize the SAP ALE e*Way, you must first define a Logical System in SAP to represent the e*Gate system as an ALE client (either sender or receiver). Next, you create a new Distribution Model view, which defines
how the Logical System exchanges messages. You must link it to a Client, a Logical System (the one you just created), and an IDoc type. The Distribution Model hierarchy is depicted in Figure 29, as it appears in the SAP GUI (IMG).

**Figure 29  Distribution Model Hierarchy**

![Distribution Model Hierarchy Diagram]

Following this high-level setup, you need to define Communications parameters in SAP to specify the correct routing of IDocs (either inbound to or outbound from SAP). The hierarchy of this Communication system is shown in Figure 30. The individual steps involved in the configuration are:

**Figure 30  Communications Hierarchy**

![Communications Hierarchy Diagram]

The RFC Destination defines the entity to which Remote Function Calls (RFCs) can be made; it is the same as the Logical System in the Distribution Model. The Communications Port defines a channel for communication of IDocs. The Partner Profile acts as an identifier for the e*Gate system, and provides a communications gateway by incorporating elements of the ALE interface.
4.3 Distribution Model Configuration

4.3.1 Naming the Logical System

Transaction: SALE

**Figure 31** SAP R/3 System Window

1. In the SAP R/3 System home window, type SALE into the command field and **Enter** to display the Distribution (ALE) Structure window.

**Figure 32** Distribution (ALE) Structure Display Window

2. Expand the tree to display Distribution (ALE) > Basic Settings > Set up logical system > Maintain logical systems.

3. Select **Maintain logical systems**, which displays the Logical Systems Overview window.

4. Select **New entries**, which displays the New Entries window.
5 Enter the logical name for your ALE e*Way (use all caps) and a brief descriptive name. Save by selecting , and the Change Request Entry window is displayed.

Figure 34  Change Request Entry Window (1)

6 Select , which displays the Create Request window.

7 Enter a short description (e.g., eWay Test) and Save by selecting . The Change Request entry window is again displayed.

8 Select to enter the new data into the system. You are now returned to the Logical Systems Overview window, and the new Logical System appears in the list.

9 Save , and select repeatedly until you return to the SAP R/3 System window.
4.3.2 Specifying the Distribution Model

Cautionary Notes

Two notes of caution are appropriate at this point:

1. Avoid placing e*Gate in the SAP Primary Model View. Use the SAP Primary Model View only as a template for your custom model view.

2. You should use the Z prefix when defining a name. This prefix is reserved for external use, and is not used in any standard SAP names.

Following these rules should prevent any interference with standard SAP functionality or conflicts with standard SAP terminology.

Transaction: SALE

Figure 35  SAP R/3 System Window

1. In the SAP R/3 System home window, type SALE into the command field and Enter to display the Distribution (ALE) Structure window.

Figure 36  Distribution Structure Window
1 Select Maintain distribution model, which displays the Maintain Distribution Model window.

**Figure 37** Maintain Distribution Model Window

2 Select the Menu path Edit > Model View > Create, which displays the Create Model View dialog window.

**Figure 38** Create Model View Dialog Window

3 Enter the logical name you want for the new Distribution Model View, along with a brief descriptive name or message (for your own use).

4 Select Enter, which returns you to the previous window. Your new Model View now appears in the tree, as shown in Figure 39.

**Figure 39** Maintain Distribution Model Tree

5 Highlight the new entry and select Add Message Type. This displays the Add Message Type dialog window.
6 Type the desired values for the four parameters into the text boxes, or select them from the drop-down menus. For example, CREMAS is the message type used for Creditor Master Data.

7 Select Enter, which returns you to the previous window. The values you selected now appear in the Distribution Model tree, as shown in Figure 41.

8 Save your entry, click Back and then Cancel to return to the Distribution Structure screen.
4.4 Communications Configuration

4.4.1 Defining the RFC Destination

Transaction: SM59

An RFC connection must be defined from the sender to the recipient. The first step is to define the RFC Destination.

**Figure 42** SAP R/3 System Window

1. In the SAP R/3 System home window, type SM59 into the command field and Enter to display the RFC Destination Maintenance window.

**Figure 43** RFC Destination Maintenance Window

2. Select the TCP/IP connections option and then Create, which displays the RFC Destination entry window.
Figure 44  RFC Destination Entry Window

3  Type in the name of the RFC Destination (use the Logical System name), an accompanying Description, and enter <T> for the Connection Type (TCP/IP).

4  Save, which returns a confirmation message and displays the RFC Destination window corresponding to your entry.
5 Select **Registration** for the Activation type and type in a Program ID of the form `<hostname>.<program name>`, and a Description.

**Note:** This Program ID must be exactly the same as that specified in the e*Way Configuration Parameter **Program ID (SAP to e*Gate)** on page 88 (the value is case-sensitive).
6 Select **Test connection**, which tests the connection for logon speed and message transfer speed. When the e*Way is running, the results are displayed in a table; otherwise, return code 3 is displayed.

**Figure 46  Connection Test Results**

<table>
<thead>
<tr>
<th>Connection type</th>
<th>TCP/IP connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logon: 0 KB:</td>
<td>255 nsec</td>
</tr>
<tr>
<td>10 KB:</td>
<td>491 nsec</td>
</tr>
<tr>
<td>20 KB:</td>
<td>504 nsec</td>
</tr>
<tr>
<td>30 KB:</td>
<td>505 nsec</td>
</tr>
</tbody>
</table>

7 Save and select repeatedly to return to the **SAP R/3 System** window.

### 4.4.2 Defining the Communications Port

**Transaction: WE21**

The Communications Port defines the type of connection with the Partner (see **Creating a Partner Profile** on page 55). In this step you specify the outbound file name, directory path, and any associated function modules.

**Figure 47  SAP R/3 System Window**

1 In the **SAP R/3 System** home window, type WE21 into the command field and enter to display the **WF-EDI Port Definition** window.
2 Expand the tree under **Transactional RFC** to display the currently-defined Ports.

3 Select the desired **Port** from the list, or select **Change** to display the **Port Definition for Asynchronous RFC Overview** window.

4 Type in a **Version** (specifies IDoc record type), **Logical destination**, and **Description**, matching the entries made previously.

5 Select **Enter**, which displays the **Change Request Query** dialog window. [Note that you must have CTS (Correction and Transport System) turned on for this screen to be displayed.]

6 Select **Create Request**, which displays the **Create Request** dialog window.

7 Enter a **Short description** and **Save**.
8 Select repeatedly to return to the *SAP R/3 System* window.

### 4.4.3 Creating a Partner Profile

**Transaction: WE20**

Here you create the Partner for the Logical System you created earlier. Note that the LS Partner Type is used for all ALE distribution scenarios.

**Figure 50** SAP R/3 System Window

---

1. In the SAP R/3 System home window, type WE20 into the command field and Enter to display the *Partner Profile: Initial Screen* window.

**Figure 51** Partner Profile: Initial Screen Window

---

2. Type the name of the logical system created previously into the Partner number field, select LS for the Partner type, and select . This creates the Partner, and displays the *Create Partner Profile <Partner Number>* window.
3 Select ALE for the Partner class and A (Active) for the Partner status, then Save. You now have created the Partner, and need to continue to the next section to configure the Partner Profile.

4.4.4 Configuring a Partner Profile

Transaction: WE20

In this section, you configure the Inbound or Outbound Parameters in the Partner Profile.
4 In the **Partner Profile: Initial Screen** window, select the desired Partner Number, for example **NEWALEEWY**.

5 Selecting **Inbound parameters**, for example, displays the **EDI Partner Profile: Inbound Parameters Overview** window for **NEWALEEWY**.

**Figure 54**  EDI Partner Profile: Inbound Parameters Overview Window (1)

6 Select **New entries**, which displays the **New Entries: Details of Created Entries** window for **NEWALEEWY**.
Select CREMAS as a Message type and CRE1 as a Process code from the drop-down menus, then Save. The entries now appear in the list in the EDI Partner Profile: Inbound Parameters Overview window.
Figure 56  EDI Partner Profile: Inbound Parameters Overview Window (2)

Follow the same procedure for Outbound parameters, if appropriate.

After making your entries, save and exit to the main SAP R/3 System window.
4.5 Security Issues

SAP uses authorization objects to allow access to various levels of operation. A minimum set of authorization objects required for the ALE e*Way to operate is described below. Please use this only as a reference for setting up your own profiles.

These settings are located under Cross-Application Authorization Objects. Please refer to your SAP R/3 documentation for additional information.

Function Group Access

Under Auth. check for RFC access, select:

- ARFC
- EDIN
- ERFC
- RFC1
- SCCR
- SYST
- ZDG1

Permission for Processing IDoc Type

Under ALE/EDI > Distributing master data and ALE/EDI, select:

- Receiving IDocs via RFC
Chapter 5

Setup Procedures

This chapter describes the procedure for customizing the SeeBeyond e*Way Intelligent Adapter for SAP (ALE) to operate with your system.

5.1 Overview

After creating a schema, you must instantiate and configure the SAP ALE e*Way to operate within the schema. A wide range of setup options allow the e*Way to conform to your system’s operational characteristics and your facility’s operating procedures.

The topics discussed in this chapter include the following:

Setting Up the e*Way

Creating the e*Way on page 62
Modifying e*Way Properties on page 63
Configuring the e*Way on page 64
Changing the User Name on page 68
Setting Startup Options or Schedules on page 68
Activating or Modifying Logging Options on page 70
Activating or Modifying Monitoring Thresholds on page 71

Troubleshooting the e*Way

Configuration Problems on page 72
System-related Problems on page 73
5.2 Setting Up the e*Way

Note: The e*Gate Enterprise Manager GUI runs only on the Windows operating system.

5.2.1 Creating the e*Way

The first step in implementing an e*Way is to define the e*Way component using the e*Gate Enterprise Manager.

To create an e*Way

1. Open the schema in which the e*Way is to operate.
2. Select the e*Gate Enterprise Manager Navigator's Components tab.
3. Open the host on which you want to create the e*Way.
4. Select the Control Broker you want to manage the new e*Way.

Figure 57 e*Gate Enterprise Manager Window (Components View)

5. On the Palette, click Create a New e*Way.
6. Enter the name of the new e*Way, then click OK.
7. All further actions are performed in the e*Gate Enterprise Manager Navigator's Components tab.
5.2.2 Modifying e*Way Properties

To modify any e*Way properties

1. Right-click on the desired e*Way and select Properties to edit the e*Way’s properties. The properties dialog opens to the General tab (shown in Figure 58).

*Note:* The executable file is `stcewsapale.exe`.

**Figure 58** e*Way Properties (General Tab)

![e*Way - Any_eWay Properties](image)

2. Make the desired modifications, then click OK.

5.2.3 Configuring the e*Way

The e*Way’s default configuration parameters are stored in an ASCII text file with a .def extension. The e*Way Editor provides a simple graphical interface for viewing and changing those parameters to create a working configuration (.cfg) file.

To change e*Way configuration parameters

1. In the e*Gate Enterprise Manager’s Component editor, select the e*Way you want to configure and display its properties.

   Note: The default configuration files are stcewsapalein.def and stcewsapaleout.def.

   **Figure 59**  e*Way Properties - General Tab

2. Under Configuration File, click New to create a new file or Find to select an existing configuration file. If you select an existing file, an Edit button appears. Click this button to edit the currently selected file.

3. You are now in the e*Way Configuration Editor.
Using the e*Way Editor

**Figure 60** The e*Way Configuration Editor

The e*Way Editor controls fall into one of six categories:

- The **Menu bar** allows access to basic operations (e.g., saving the configuration file, viewing a summary of all parameter settings, and launching the Help system).
- The **Section selector** at the top of the Editor window enables you to select the category of the parameters you wish to edit.
- **Section controls** enable you to restore the default settings, restore the last saved settings, display tips, or enter comments for the currently selected section.
- The **Parameter selector** allows you to jump to a specific parameter within the section, rather than scrolling.
- **Parameter controls** enable you to restore the default settings, restore the last saved settings, display tips, or enter comments for the currently selected parameter.
- **Parameter configuration controls** enable you to set the e*Way’s various operating parameters.
Section and Parameter Controls

The section and parameter controls are shown in Table 8 below.

Table 8  Parameter and Section Controls

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Restore Default" /></td>
<td>Restore Default</td>
<td>Restores default values</td>
</tr>
<tr>
<td><img src="image" alt="Restore Value" /></td>
<td>Restore Value</td>
<td>Restores saved values</td>
</tr>
<tr>
<td><img src="image" alt="Tips" /></td>
<td>Tips</td>
<td>Displays tips</td>
</tr>
<tr>
<td><img src="image" alt="User Notes" /></td>
<td>User Notes</td>
<td>Enters user notes</td>
</tr>
</tbody>
</table>

Note: The section controls affect all parameters in the selected section, whereas the parameter controls affect only the selected parameter.

Parameter Configuration Controls

Parameter configuration controls fall into one of two categories:

- Option buttons
- Selection lists, which have controls as described in Table 9

Table 9  Selection List Controls

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Add to List" /></td>
<td>Add to List</td>
<td>Adds the value in the text box to the list of available values.</td>
</tr>
<tr>
<td><img src="image" alt="Delete Items" /></td>
<td>Delete Items</td>
<td>Displays a “delete items” dialog box, used to delete items from the list.</td>
</tr>
</tbody>
</table>
Command-line Configuration

In the Additional Command Line Arguments box, type any additional command line arguments that the e*Way may require, taking care to insert them at the end of the existing command-line string. Be careful not to change any of the default arguments unless you have a specific need to do so.

Getting Help

To launch the e*Way Editor’s Help system
   From the Help menu, select Help topics.

To display tips regarding the general operation of the e*Way
   From the File menu, select Tips.

To display tips regarding the selected Configuration Section
   In the Section Control group, click.

To display tips regarding the selected Configuration Parameter
   In the Parameter Control group, click.

Note: “Tips” are displayed and managed separately from the Help system that launches from the Toolbar’s Help menu. You cannot search for Tips within the Help system, or view Help system topics by requesting Tips.

For detailed descriptions and procedures for using the e*Way Configuration Editor, see the e*Gate Integrator User’s Guide.
5.2.4 Changing the User Name

Like all e*Gate executable components, e*Ways run under an e*Gate user name. By default, all e*Ways run under the Administrator user name. You can change this if your site’s security procedures so require.

To change the user name

1. Display the e*Way’s properties dialog.
2. On the General tab, use the Run as user list to select the e*Gate user under whose name this component is to run.

See the e*Gate Integrator System Administration and Operations Guide for more information on the e*Gate security system.

5.2.5 Setting Startup Options or Schedules

SeeBeyond e*Ways can be started or stopped by any of the following methods:

- The Control Broker can start the e*Way automatically whenever the Control Broker starts.
- The Control Broker can start the e*Way automatically whenever it detects that the e*Way terminated execution abnormally.
- The Control Broker can start or stop the e*Way on a schedule that you specify.
- Users can start or stop the e*Way manually using an interactive monitor.

You determine how the Control Broker starts or shuts down an e*Way using options on the e*Way properties Start Up tab (see Figure 61). See the e*Gate Integrator System Administration and Operations Guide for more information about how interactive monitors can start or shut down components.
To set the e*Way’s startup properties

1. Display the e*Way’s properties dialog.
2. Select the Start Up tab.
3. To have the e*Way start automatically when the Control Broker starts, select the Start automatically check box.
4. To have the e*Way start manually, clear the Start automatically check box.
5. To have the e*Way restart automatically after an abnormal termination:
   A. Select Restart after abnormal termination.
   B. Set the desired number of retries and retry interval.
6. To prevent the e*Way from restarting automatically after an abnormal termination, clear the Restart after abnormal termination check box.
7. Click OK.
5.2.6 Activating or Modifying Logging Options

Logging options enable you to troubleshoot problems with the e*Way and other e*Gate components.

To set the e*Way debug level and flag

1. Display the e*Way’s Properties dialog.
2. Select the Advanced tab.
3. Click Log. The dialog window appears (see Figure 62).

**Figure 62**  e*Way Properties (Advanced Tab - Log Option)

4. Select DEBUG for the Logging level.
5. Select either e*Way (EWY) or e*WayVerbose (EWYV) for the Debugging flag. Note that the latter has a significant negative impact on system performance.
6. Click OK.

The other options apply to other e*Gate components and are activated in the same manner. See the e*Gate Integrator Alert and Log File Reference for additional information concerning log files, logging options, logging levels, and debug flags.
### 5.2.7 Activating or Modifying Monitoring Thresholds

Monitoring thresholds enable you to monitor the throughput of the e*Way. When the monitoring thresholds are exceeded, the e*Way sends a Monitoring Event to the Control Broker, which routes it to the e*Gate Monitor and any other configured destinations.

1. Display the e*Way’s properties dialog.
2. Select the Advanced tab.
3. Click Thresholds.
4. Select the desired threshold options and click OK.

See the *e*Gate Integrator Alert and Log File Reference for more information concerning threshold monitoring, routing specific notifications to specific recipients, or for general information about *e*Gate’s monitoring and notification system.
5.3 **Troubleshooting the e*Way**

In the initial stages of developing your e*Gate Integrator system administration system, most problems with e*Ways can be traced to configuration.

5.3.1 **Configuration Problems**

**In the Enterprise Manager**
- Does the e*Way have the correct Collaborations assigned?
- Do those Collaborations use the correct Collaboration Services?
- Is the logic correct within any Collaboration Rules script employed by this e*Way’s Collaborations?
- Do those Collaborations subscribe to and publish Events appropriately?
- Are all the components that “feed” this e*Way properly configured, and are they sending the appropriate Events correctly?
- Are all the components that this e*Way “feeds” properly configured, and are they subscribing to the appropriate Events correctly?

**In the e*Way Editor**
- Check that all configuration options are set appropriately.
- Check that all settings you changed are set correctly.
- Check all required changes to ensure they have not been overlooked.
- Check the defaults to ensure they are acceptable for your installation.

**On the e*Way’s Participating Host**
- Check that the Participating Host is operating properly, and that it has sufficient disk space to hold the IQ data that this e*Way’s Collaborations publish.
- Check that your *path* environment variable includes the location of the SAP ALE dynamically-loaded libraries. The name of this variable on the different operating systems is:
  - PATH (Windows)
  - LD_LIBRARY_PATH (Solaris/Compaq)
  - LIBPATH (AIX)
  - SHLIB_PATH (HP-UX)

**In the SAP Application**
- Check that the application is configured correctly, is operating properly, and is sending or receiving the correct data appropriately.
5.3.2 **System-related Problems**

- Check that the connection between the external application and the e*Way is functioning appropriately.
- Once the e*Way is up and running properly, operational problems can be due to:
  - External influences (network or other connectivity problems).
  - Problems in the operating environment (low disk space or system errors)
  - Problems or changes in the data the e*Way is processing.
  - Corrections required to Collaboration Rules scripts that become evident in the course of normal operations.

One of the most important tools in the troubleshooter’s arsenal is the e*Way log file. See the *Gate Integrator Alert and Log File Reference Guide* for an extensive explanation of log files, debugging options, and using the *Gate monitoring system to monitor operations and performance.*
Chapter 6

Operational Overview

This chapter contains an overview of the architecture of the SeeBeyond e*Way Intelligent Adapter for SAP (ALE) and the basic interactive processes taking place between the e*Way and SAP R/3.

6.1 e*Way Architecture

Conceptually, the SAP ALE e*Way can be viewed as a multi-layered structure (see Figure 63), consisting of:

- Monk SAP ALE Transport layer, which manages ALE communication logic
- Monk SAP RFC Transport layer, which manages RFC communications with the SAP R/3 system
- Event Processing layer, which manages the processing of data and subscribing or publishing to other e*Gate components

Figure 63 SAP ALE e*Way Architecture
The upper layers of the e*Way use Monk functions to perform Business Process modeling and ETD mapping, package data as e*Gate Events, send those Events to Collaborations, and manage interaction with the external system. These layers are built upon an e*Way Kernel layer that manages the basic operations of the e*Way, data processing, and communication with other e*Gate components.

The communication layers of the e*Way are single-threaded. Functions run serially, and only one function can be executed at a time. Processing layers are multi-threaded, with one executable thread for each Collaboration. Each thread maintains its own Monk environment; therefore, information such as variables, functions, path information, and so on cannot be shared between threads.

### 6.2 RFC Communications

Messages can be sent to the SAP R/3 host via Transactional RFC (tRFC) or regular RFC. With tRFC, the receiving SAP system relies on an unique Transactional ID (TID) sent with the message to ascertain whether or not a transaction has ever been processed by it before. Rather than comparing every message sent out by this e*Way against a database to determine if it has been processed previously, however, the SAP ALE e*Way assumes that all messages handled are new and assigns a new TID to each message (the counter is persistently stored by the e*Way).
Collaborations execute the business logic that enable the e*Way to do its intended work. In turn, each Collaboration executes a Collaboration Rule, containing the actual instructions to execute the business logic. Each Collaboration that publishes its processed Events internally (within e*Gate Integrator) requires one or more IQs to receive the Events, as shown in Figure 64. Any Collaboration that publishes its processed Events only to an external system does not require any IQs.

**Figure 64** Collaborations and IQs

Configuration options that control the Monk environment and define the Monk functions used to perform various e*Way operations are discussed in Chapter 7. You can create and modify these functions using the SeeBeyond Collaboration Rules Editor or a text editor (such as Microsoft Word or Notepad, or UNIX vi).

For more information on defining Collaborations, defining IQs, assigning Collaborations to e*Ways, or configuring Collaborations to publish Events, see the e*Gate Integrator User’s Guide.
The SAP ALE e*Way includes a front-end SAP Converter Wizard, integrated with the e*Gate Event Type Definition (ETD) Editor Build Library Converter. The Converter parses the SAP IDoc Description Files and builds the corresponding Event Type Definitions and IDs automatically.

The SAP Converter Wizard GUI is implemented in Java, and calls an ANSI-C program that performs the actual conversion between the IDoc and e*Gate Event Type Definitions and ID files.

When invoked by the ETD Editor > Build Library > Converter menu selection, the Java application is supplied with the specified Event Type Definition file path name and the appropriate environment variables.

**Figure 65** IDoc/DXOB Converter
6.5 Data Flow and Event Processing

6.5.1 e*Gate to SAP

Some application external to SAP generates data and sends the data to e*Gate via an e*Way. e*Gate then performs any necessary processing or routing, and sends the data to the SAP ALE e*Way. When the data is received by the SAP ALE e*Way, a Monk function is (optionally) called and the resultant IDoc sent via RFC/tRFC to a SAP R/3 function INBOUND_IDOC_PROCESS, with the IDoc as a parameter. That triggers an SAP workflow to import the transmitted IDoc into the SAP application database.

Messages can be sent to the SAP R/3 host via Transactional RFC (tRFC) or regular RFC. With tRFC, the receiving SAP system relies on an unique Transactional ID (TID) sent with the message to ascertain whether or not a transaction has ever been processed by it before. The SAP ALE e*Way assumes that all messages handled are new and assigns a new TID to each message (the counter is persistently stored by the e*Way).
Dynamic routing of messages to different SAP R/3 hosts is not supported by the e*Way, since the required routing information is not inherently part of the IDoc message format. In Client mode, a single instance of the SAP ALE e*Way can establish an SAP R/3 connection with only one host (and as one user) at a time. Additional instances are required to connect to a different SAP R/3 host or as a different user.

**Figure 67**  ALE SAP-Inbound Event Processing Flow
1 The e*Way reads in the required configuration parameters and:
   - Establishes a network connection with the SAP system, according to the specified Connection Duration Strategy.
   - Sends the appropriate Interface Up messages to e*Gate.
   - If desired, uses a Collaboration Rule to convert an Event containing multiple IDocs to several single IDocs.

2 The e*Way then uses another Collaboration Rule to set a Monk variable, MONK_X_Status, to reflect the status of the Collaboration. The allowed values are as follows:

<table>
<thead>
<tr>
<th>Value</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>The e*Way sends the Event to the SAP system.</td>
</tr>
<tr>
<td>MULTI</td>
<td>The e*Way reads and sends individual records to the SAP system from the file designated by the configurable Monk variable MONK_OutFile (and whose record delimiter is defined by the configurable Monk variable MONK_RecDelim).</td>
</tr>
<tr>
<td>HOLD</td>
<td>The e<em>Way acknowledges e</em>Gate, but does not forward the Event to the SAP system.</td>
</tr>
<tr>
<td>ERROR</td>
<td>The e<em>Way logs the Collaboration failure explanation and original outbound Event, but sends an ACK to e</em>Gate (since NAKing would be futile). Indication of the failure type follows the colon.</td>
</tr>
</tbody>
</table>

3 If tRFC is enabled, the e*Way associates the next TID (from a persistent resetable counter) with the transformed outbound Event and sends it via tRFC to the SAP host. Otherwise, it sends the Event via regular RFC.

4 If no exceptions are raised by the receiving SAP host, the next TID is incremented, if appropriate, and the e*Way sends an ACK to e*Gate.

5 If exceptions are raised, and the error is unrecoverable (i.e., re-transmission not feasible), the offending outbound Event is logged (to a configurable file) and the e*Way sends an ACK to the e*Gate server.

6 The e*Way repeats the procedure beginning with step 3.
6.5.2 SAP to e*Gate

During routine operation, some application on the SAP R/3 system generates a transaction designated for a target application. The data is converted to IDoc format by the ALE Interface and sent via RFC to the SAP ALE e*Way.

The SAP ALE e*Way receives this transaction in IDoc format and provides the required INBOUND_IDOC_PROCESS service. The transaction is translated into the proper Event Type Definition and passed to e*Gate. A separate e*Way then re-translates and communicates the transaction to the intended destination system.
Figure 69  ALE SAP-Outbound Event Processing Flow

![Diagram of ALE SAP-Outbound Event Processing Flow]

- **SAP ALE Interface**
  - IDoc
  - NAK
  - RFC

- **Retrieve IDoc**
  - Yes
  - Already Processed?
    - No
      - Identification and Transformation
        - MONK_X_Status
          - OK / MULTI
            - e*Gate Integrator

- **INBOUND_IDOC_PROCESS**
  - ACK / NAK
  - NAK
  - HOLD / ERROR:
    - Yes
      - Already Processed?
        - No
          - Identification and Transformation
            - MONK_X_Status
              - OK / MULTI
                - e*Gate Integrator
1 The e*Way reads in the required configuration parameters and:
   ▪ Establishes a network connection with the SAP system.
   ▪ Instantiates the INBOUND_IDOC_PROCESS function through RFC.
   ▪ Sends the appropriate Interface Up messages to e*Gate.
2 It then waits for transactions from an SAP system, occasionally breaking out at
   prescribed intervals to process e*Gate events.
3 If desired, the e*Way verifies the identity of the transaction sender.
4 If tRFC is enabled, the e*Way verifies that the Transactional ID (TID) of the received
   transaction has not previously been committed (processed successfully) by this
   e*Way. If it has, the e*Way sends a NAK to the sending SAP system.
5 A GNU-based DBMS is used to track transactions that have been committed
   successfully or rolled back, each with a timestamp. To expedite database searches,
   the database is purged periodically to delete all entries that have exceeded their
   specified lifetimes.
6 The e*Way attempts to identify the incoming transaction against a preset list of ID
   rules, each of which is associated with a Collaboration Rule.
7 If identified successfully, the process moves on to the next step. If not, the e*Way
   composes the appropriate response and raises an exception with the sender.

Note: To expedite processing, the Identification and Collaboration steps can be combined
by making the Collaboration Rule and Collaboration Rule pathnames identical.

8 The e*Way calls the corresponding Collaboration Rule to map the transaction.
9 If the Collaboration fails, the e*Way sets the configurable Monk Collaboration
   Status variable, MONK_X_Status, to ERROR: (along with indication of the failure
   type, after the colon).
10 If the Monk variable MONK_X_Status returned is:
   ▪ OK, the e*Way sends the Event to e*Gate for further Collaboration and routing.
   ▪ MULTI, the e*Way reads and sends individual records to e*Gate from the file
     designated by the configurable Monk variable MONK_OutFile and whose record
     delimiter is defined by the configurable Monk variable MONK_RecDelim.
   ▪ HOLD, the e*Way acknowledges the sending SAP system for the transaction, but
     does not forward any Event to the e*Gate server.
   ▪ ERROR:, the e*Way sends the translation failure code as an exception to the sender.
     Indication of the failure type follows the colon.
11 As appropriate, the e*Way waits for acknowledgment from e*Gate before
   acknowledging the sending SAP system.
12 If the Event sent to e*Gate is not identifiable and/or transformable, then a NAK is
   be returned, thereby causing an exception to be raised with the sender. (The SAP
   system needs a positive acknowledgment before committing the transaction.)
13 The e*Way then repeats the procedure beginning with step 4.
Chapter 7

Configuration Parameters

This chapter describes the configuration parameters for the SeeBeyond e*Way Intelligent Adapter for SAP (ALE).

7.1 Overview

The e*Way’s configuration parameters are set using the e*Way Editor; see Configuring the e*Way on page 64 for procedural information. The default configurations are provided in stcewsapalein.def and stcewsapaleout.def. The SAP ALE e*Way’s configuration parameters are organized into the following sections:

- General Settings on page 85
- ALE/RFC Setup on page 88
- Communication Setup on page 96
- Transaction Processing on page 99
- Advanced Monk Setup on page 108
7.2 General Settings

The General Settings set the direction of data or Event flow, relative to this e*Way.

**Data Direction (SAP to e*Gate)**

**Description**

When configured as an Inbound to e*Gate System, this e*Way receives Events from the SAP system and forwards them to e*Gate.

**Required Values**

A string; the default is **Inbound to e*Gate System**.

**Data Direction (e*Gate to SAP)**

**Description**

When configured as an Outbound from e*Gate System, this e*Way receives Events from e*Gate and forwards them to the SAP system.

**Required Value**

A string; the default is **Outbound from e*Gate System**.

**Interface Name (SAP to e*Gate)**

**Description**

Communicates a more verbose name to the status commands that originate from stccmd.exe or the e*Gate Monitor.

**Required Values**

A file name; the default is **SAP_Inbound_e*Way**.

**Additional Information**

Any spaces within the file name are converted automatically to underscores (_).

**Interface Name (e*Gate to SAP)**

**Description**

Communicates a more verbose name to the status commands that originate from stccmd.exe or the e*Gate Monitor.

**Required Values**

A file name; the default is **SAP_Outbound_e*Way**.
Additional Information

Any spaces within the file name are converted automatically to underscores (_).

---

**Incoming Raw Message Journal (SAP to e*Gate)**

**Description**

Specifies a path to the Journal file; journaling of all incoming Events/messages is enabled only when this parameter contains a value.

**Required Values**

A file name with an optional absolute path; if the file does not exist, it is created. If you provide a file name without an absolute path, then the value of `SystemData` from `.egate.store` is prefixed to it.

**Additional Information**

The Incoming Raw Message Journal contains all raw, untranslated messages sent in by the SAP system. This Journal can be converted to a readable flat file by using the `stcjdump.exe` utility (see Appendix A).

---

**Outgoing Translated Message Journal (e*Gate to SAP)**

**Description**

Specifies a path to the Journal file; journaling is enabled only when this parameter contains a value.

**Required Values**

A file name with an optional absolute path. If you provide a file name without an absolute path, then the value of `SystemData` from `.egate.store` is prefixed to it.

**Additional Information**

Messages are appended to the Journal file. If the file does not exist, it is created. This Journal can be converted to a readable flat file by using the `stcjdump.exe` utility (see Appendix A).

---

**Message Journal Archiving Schedule**

**Description**

This parameter contains one or more message journal archiving schedules. It can contain either a set of date/time based schedules or a single repeating timer (e.g., every n seconds). The schedule instructs the e*Way to archive the message journal at the specified times and open a new journal afterwards.

**Required Values**

A schedule set, containing one or more schedules.
Additional Information

The Message Journal is archived in a Month-Day named subdirectory (for example, \Feb-18) with a .hhmmss suffix attached to the original file name (for example, .181030 for 6:10:30PM).

Rejected Message Journal (SAP to e*Gate)

Description

Specifies a path to the Journal file; journaling of rejected Events/messages is enabled only when this parameter contains a value.

Required Values

A file name with an optional absolute path; if the file does not exist, it is created. If you provide a file name without an absolute path, then the value of SystemData from .egate.store is prefixed to it.

Additional Information

The Rejected Message Journal contains all raw messages sent in by the SAP system that resulted in a failed Collaboration or were rejected by e*Gate. This Journal can be converted to a readable flat file by using the stcjdump.exe utility (see Appendix A).
7.3 ALE/RFC Setup

The ALE/RFC Setup controls are used to activate the RFC Trace function. The following options are set within the ALE/RFC Setup section:

**SAP System Type**

Description

Specifies the SAP System Type.

*Note:* Only SAP R/3 (3 Tier) Systems are currently supported by this e*Way.

Required Values

A string; the default is R/3.

**SAP IDoc Version**

Description

Specifies the SAP IDoc Version.

- Release 3.x version:
  
  Control Record length of 464, Data Record length of 1055

- Release 4.x version:
  
  Control Record length of 524, Data Record length of 1063

Required Values

Either 3.x, 4.x, or All. The default is 4.x.

Additional Information

Specifies which version of Inbound (SAP-to-e*Gate) IDocs are being processed by this e*Way. The corresponding ABAP/4 function, INBOUND_IDOC_PROCESS (for 3.x) or IDOC_INBOUNDASYNCHRONOUS (for 4.x), is installed automatically on the R/3 application server.

A Monk variable, MONK_SAP_IDOC_VERSION, is set to 3.x or 4.x each time an IDoc is received to indicate the version. This variable is intended to be read-only and can be used in the Monk Collaboration to determine which versions of IDocs to translate.

**Program ID (SAP to e*Gate)**

Description

Specifies the SAP R/3 system Program ID of this e*Way.
Required Values

A string, which must be in the form: host_name.program_name.

Additional Information

This parameter must be exactly the same as that shown under the SAP GUI transaction sm59 for RFC Destination (the value is case-sensitive).

---

**SAP Load Balancing Usage (e*Gate to SAP)**

**Description**

Specifies whether or not to use SAP Load Balancing.

**Required Values**

Yes or No.

If Yes, the system routes requests to the Application Server that has the best response time at that moment (assuming multiple Application Servers).

**See also**

- SAP Application Server (e*Gate to SAP) on page 89
- SAP Gateway Host (e*Gate to SAP) on page 91
- SAP Message Server (e*Gate to SAP) on page 95
- SAP Application Server Group (e*Gate to SAP) on page 95.

**Note:** As of R/3 release 3.0C onwards, workload balancing is available within SAP to automatically route requests to the best-choice Application Server.

---

**SAP Application Server (e*Gate to SAP)**

**Description**

Specifies the hostname or IP address of the SAP application server to which this e*Way sends IDocs.

**Required Values**

A valid hostname or IP address.

**Additional Information**

This parameter should be configured only when using SAP Load Balancing.

**Note:** Do not append any Router String information to this value.
**SAP Router String (e*Gate to SAP)**

**Description**

Specifies the host names or IP addresses of all the SAP routers that are located between this e*Way and the SAP gateway host, allowing access to an SAP system that’s behind a firewall.

**Note:** This parameter is optional, and the field(s) may be left blank.

**Required Values**

A string containing host names or IP addresses. For example, if there are two routers, `saprouter1` and `saprouter2` (in that order) from the e*Way to the SAP Gateway Host, the SAP Router String is determined as:

```
saprouter1 204.79.199.5
saprouter2 207.105.30.146
SAP Router String /H/204.79.199.5/H/207.105.30.146/H/
```

**Note:** You must include the /H/ tokens to separate the router addresses as well as to indicate the beginning and end of the string.

---

**SAP System Number (e*Gate to SAP)**

**Description**

Specifies the System Number of the SAP R/3 Application Server.

**Required Value**

A numeric string, including any leading zeroes.

---

**SAP Gateway Host (SAP to e*Gate)**

**Description**

Specifies the SAP System Host.

**Required Values**

The hostname or IP address of the SAP gateway host (usually, it is the same as the SAP application server). Do not append any SAP Router String information.

**Additional Information**

The SAP gateway host is the communication front-end for the SAP application server. It supports the registration of external RFC programs which the application server calls (e.g., when sending outbound IDocs to this e*Way).
SAP Gateway Host (e*Gate to SAP)

Description
Specifies the SAP System Host.

Required Values
The hostname or IP address of the SAP gateway host (any necessary SAP router string information must precede the hostname).

Additional Information
The SAP gateway host is the communication front-end for the SAP application server and is normally configured as such. For those situations when it is not, enter the indicated value and also configure the SAP Gateway Service parameter.

Note: This parameter should be configured only when not using SAP load balancing.

SAP Router String (SAP to e*Gate)

Description
Specifies the host names or IP addresses of all the SAP routers that are in between this e*Way and the SAP Gateway Host, allowing access to an SAP system that’s behind a firewall.

Note: This parameter is optional, and the field(s) may be left blank.

Required Values
A string containing host names or IP addresses. For example, if there are two routers, saprouter1 and saprouter2 (in that order) from the e*Way to the SAP Gateway Host, the SAP Router String is determined as:

```
saprouter1 204.79.199.5
saprouter2 207.105.30.146
SAP Router String /H/204.79.199.5/H/207.105.30.146/H/
```

Note: You must include the /H/ tokens to separate the router addresses as well as to indicate the beginning and end of the string.

SAP Gateway Service

Description
Specifies the service number or name on the SAP Gateway sending transactions, which listens for this e*Way’s ALE registration. This is equivalent to a TCP/IP port number and can be referenced on the SAP R/3 system in the following directory.
Chapter 7
Configuration Parameters

Section 7.3
ALE/RFC Setup

**RFC Trace On**

**Description**
Activates RFC tracing. The trace file contains RFC API calls, and data sent to and received from the SAP R/3 host.

**Required Values**
Yes or No.

**Additional Information**
The location of the trace file is as follows.

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Directory/Folder</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIX</td>
<td>/etc/services</td>
</tr>
<tr>
<td>NT 4.0</td>
<td>C:\Winnt\system32\drivers\etc\Services</td>
</tr>
</tbody>
</table>

**Validate SAP Sender Identity (SAP to e*Gate)**

**Description**
Specifies whether or not validation of sender identity is to be performed.

**Required Values**
Yes or No.

**Additional Information**
If no validation is desired, any raw message received from an SAP R/3 system is processed. Otherwise, if the sender’s identity does not match exactly with the respective...
configuration parameters, an exception is raised with the sender and the raw message is logged but not processed.

---

**SAP System Number (SAP to e*Gate)**

**Description**

Specifies the incoming RFC connection (used for ID verification).

**Required Values**

A numeric string, including any leading zeroes.

**Additional Information**

If none is specified, then any SAP R/3 system number is allowed.

---

**SAP System ID (SAP to e*Gate)**

**Description**

Specifies the incoming RFC connection (used for ID verification).

**Required Values**

A valid system ID.

**Additional Information**

Typically, this is the Oracle back-end Database SID string and can be verified from the SAP GUI under transaction S000 (main menu) under menu bar item System > Status... in the Database Data section, text field Name. If none is specified, then any SAP R/3 System ID is allowed.

---

**SAP Client Number (SAP to e*Gate)**

**Description**

Specifies the incoming RFC connection (used for ID verification).

**Required Values**

A alphanumeric string, including any leading zeroes.

**Additional Information**

If none is specified, then any SAP R/3 Client Number is allowed.

---

**SAP User Name (SAP to e*Gate)**

**Description**

Specifies the incoming RFC connection (used for ID verification).
Required Values

A alphanumeric string.

Additional Information

If none is specified, then any SAP R/3 User Name is allowed.

SAP User Language (SAP to e*Gate)

Description

Specifies language of user.

Required Values

- EN - English
- DE - German (Deutsche)

SAP System Client Number (e*Gate to SAP)

Description

Specifies a Client Number, used to access the SAP R/3 Destination.

Required Value

A alphanumeric string, including any leading zeroes.

Logon User Name (e*Gate to SAP)

Description

Specifies the user’s logon name for the SAP R/3 System.

Required Value

A valid user logon name.

Logon Password (e*Gate to SAP)

Description

Specifies the user’s logon password for the SAP R/3 System.

Required Value

An encrypted string containing a user logon password.

Additional Information

You must enter the Logon User Name before entering the Password.
Logon Language (e*Gate to SAP)

**Description**
Specifies the user’s language for the SAP R/3 System GUI.

**Required Values**
One of the following:
- EN for English
- DE for German (Deutsche)
The default value is EN.

SAP Message Server (e*Gate to SAP)

**Description**
Specifies the hostname or IP address of the SAP message server which is distributing the IDocs sent by this e*Way to a group of SAP application servers.

**Required Value**
A valid SAP message server name. Do not prefix any router string to this value.

**Note:** This parameter should be configured only when using SAP load balancing (L/B).

SAP Destination System (e*Gate to SAP)

**Description**
Specifies the destination SAP R/3 System.

**Required Value**
A string.

**Additional Information**
Typically, this is the Oracle back-end database SID string, and can be verified from the SAPGUI. Select transaction S000 (main menu) under menu bar item System > Status... in the Database Data section, text field Name.

**Note:** This parameter should be configured only when using SAP load balancing (L/B).

SAP Application Server Group (e*Gate to SAP)

**Description**
Specifies the SAP application servers that is sharing the workload.

**Required Value**
A string.
7.4 Communication Setup

This group of parameters helps to fine tune the time-outs to which the e*Way is sensitive, or to which it acts on because of communications problems with the SAP R/3 system.

Time to Wait Before Becoming Server (SAP to e*Gate)

Description

Specifies the interval of time to wait between successive attempts to become an SAP ALE RFC Server (i.e., provide an available ABAP/4 function to the receiving SAP R/3 system).

Required Values

A time interval in Seconds.

e*Gate System Polling Interval (SAP to e*Gate)

Description

Specifies the interval of time the e*Way waits between successive attempts to check for e*Gate system activity.

Required Values

A time interval in Seconds. Maximum value = 5 seconds.

Time to Wait Before Attempting Connection (e*Gate to SAP)

Description

Specifies the time to wait between attempts to establish an RFC connection with the receiving SAP R/3 system.

Required Value

A time interval in Seconds.

Connection Duration Strategy (e*Gate to SAP)

Description

Specifies the interval the e*Way waits between successive attempts to establish an RFC connection with the receiving SAP R/3 System. If the attempt fails, the e*Way waits this interval before retrying the connections.

Note: This parameter should be configured only when using SAP load balancing (L/B).
Required Values

Maintain Connection  
Connection to the SAP host is maintained at all times, even when there are no outbound Events.

Connect as Needed  
Connection to the SAP host is terminated after a certain idle time when no more Outbound Events are received by this e*Way to be sent to the external system. Connection is then re-established with the next Outbound Event.

See also: Idle Interval Before Disconnecting (e*Gate to SAP) on page 97.

Idle Interval Before Disconnecting (e*Gate to SAP)

Description
Specifies the time to wait before considering a connection to the SAP R/3 system to be idle, and unilaterally disconnecting from the SAP system. This parameter is used only when the Connection Duration Strategy is set to Connect As Needed.

Required Values
A time interval in Seconds.

See also
Connection Duration Strategy (e*Gate to SAP) on page 96

Connection Error Resend Strategy (e*Gate to SAP)

Description
The strategy to use when re-sending a transaction to the SAP R/3 system after connection errors have occurred. These errors can be network interruptions or deliberate closures by the SAP host, indicating problems with the data being sent.

Required Values

Persistently Resend  
Keep re-sending the transaction, regardless of the number of attempts.

Skip After Max Attempts  
Keep re-sending up to a maximum number of attempts (see parameter below), after which the transaction is logged as REJECTED; the next outbound transaction is then processed.

See also: Maximum Resend Attempts (e*Gate to SAP) on page 98.
Maximum Resend Attempts (e*Gate to SAP)

Description

Specifies the maximum number of attempts to retry sending a transaction to the SAP R/3 host, due to connection closure errors. This parameter is used only when the Connection Error Resend Strategy is set to Skip After Max Attempts.

Required Values

A number within the range of 1 to 100, inclusive. The default is 5.
7.5 Transaction Processing

These parameters control the way that transactions are identified, transformed and sent to e*Gate for further processing and routing.

Enforce Transactional RFC

Description

Ensures that transactions having unique Transactional IDs (TID) are processed only once by this e*Way.

Required Values

Yes or No.

Additional Information

It is highly recommended that Transactional RFC (tRFC) be enforced. However, tRFC is not always deployable (such as for SAP R/3 release 3.0), so the option of No is provided.

Transactional ID Verification Database (SAP to e*Gate)

Description

Provides the path name to the database file in which the disposition of all transactions from SAP to this e*Way is to be recorded.

Required Values

A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter.

Additional Information

The database records whether transactions are:

- committed
- rejected
- reserved (pending)
- unprocessed (changed from reserved, but never committed or rejected)

Every time the e*Way is started up or suspended (which causes any RFC connection with SAP to be aborted), the Transactional ID Verification Database is scanned for entries that are in either the reserved (RESERV) or unprocessed (UNPROC) state.

See also

- Verification Database Operating Mode on page 100
- Database Purge Schedule on page 100
Transactional ID Verification Database (e*Gate to SAP)

Description

Provides the path name to the database file in which the disposition of all transactions from this e*Way to SAP is to be recorded.

Required Values

A file name with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter.

Additional Information

The database records whether transactions are:

- committed
- rejected
- reserved (pending)

See also

Verification Database Operating Mode on page 100
Database Purge Schedule on page 100

Verification Database Operating Mode

Description

Specifies the mode of operation for the e*Way

Required Values

Exclusive or Shared.

Additional Information

If load balancing is required and several instances of this e*Way are activated, the Transactional ID Verification Database may need to be shared with the other e*Ways in order to ensure transactional integrity. However, placing the database in Shared mode may impact performance due to extra disk I/O. It is highly recommended that Expunge Database Entry On Confirm be set to Yes when Shared mode is selected.

See also

Transactional ID Verification Database (SAP to e*Gate) on page 99
Transactional ID Verification Database (e*Gate to SAP) on page 100

Database Purge Schedule

Description

Specifies one or more schedules to purge the Transactional ID Verification Database.
Required Values

A schedule containing either a set of dates and times, or a single repeating timer (e.g. every n seconds). The schedule instructs the e*Way to purge the TID Verification Database at specified times of old entries that exceed a certain lifetime.

Note: Purging is mandatory—otherwise the file system becomes overloaded and performance degrades.

See also

Database Entry Lifetime on page 101

Database Entry Lifetime

Description

Specifies the lifetime of an entry in the TID Verification Database. At purge times, as determined by the Database Purge Schedule variable, all database entries having a time stamp older than that allowed by this lifetime are deleted (with the exception of Reserved entries).

Required Values

A number between 86400 to 2678400, inclusive. The default is 432000. The units are selectable as Minutes, Hours or Days. The default is Days.

See also

Transactional ID Verification Database (SAP to e*Gate) on page 99
Transactional ID Verification Database (e*Gate to SAP) on page 100
Verification Database Operating Mode on page 100

Expunge Database Entry On Confirm

Description

In order to minimize the operational size of the Transactional ID Verification Database, entries containing TIDs that were Confirmed at the conclusion of a transaction can be expunged or removed. This reduces the search time for TIDs that are still being processed.

Required Values

Yes or No; the default is No.

- Select Yes if higher performance is desired of this e*Way
- Select No to preserve all tracking of transactions in the Database

See also

Transactional ID Verification Database (SAP to e*Gate) on page 99
Transactional ID Verification Database (e*Gate to SAP) on page 100
Transaction Identification and Translation (SAP to e*Gate)

Description

Each incoming Transaction needs to be identified as a valid IDoc Structure Syntax and, if successful, translated to an appropriate message format.

Required Values

Pairs of Collaboration ID Rule and Collaboration Rule filenames, each name separated by a caret (^). The default is EG_ID_ALL^<none>.

Additional Information

Rules can be given either as pathnames (must contain a .monk or .tsc extension) or Monk functions (these must be predefined or loaded from a custom Monk File).

Note: When entering pathnames, use relative pathnames (these are keys to objects in the ETC e*Gate Registry). It is assumed that the Collaboration ID Rule and Collaboration Rule function names correspond to the names of their respective files.

Both the Collaboration ID Rule and Collaboration Rule may require extensive mapping of data into IDoc structures. In order to improve performance, both Collaborations can be combined into a single Collaboration File and the same file name entered for both Collaboration ID Rule and Collaboration Rule.

Additionally, if the Event ID Collaboration file name is exactly EG_ID_ALL (this is not a real file name), then all IDocs are identified successfully. Also, if the Collaboration Rule file name is exactly none, then no translation is performed on the IDoc. Thus, to allow all IDocs to be sent to e*Gate un-translated, type in: EG_ID_ALL^none.

If a Collaboration Rule file name is used, the Monk function must set a global Translation Status Monk Variable (whose name is configurable) to one of the recognized statuses before returning, using a Monk statement such as:

\[ \text{(define MONK_X_Status"OK")} \]

See also

Load Custom Monk File Name on page 108.

Transaction Identification and Translation (e*Gate to SAP)

Description

Identifies an outgoing e*Gate system Event and, if successful, translates it to the appropriate IDoc format.

Required Values

Pairs of Event ID Collaboration and Collaboration Rule filenames, each name separated by a caret (^). If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter.
Additional Information

Both the Event ID Collaboration and Collaboration Rule may require extensive mapping of data into IDoc structures. In order to improve performance, both Collaborations can be combined into a single Collaboration File and the same Filename entered for both Event ID Collaboration and Collaboration Rule.

If the Event ID Collaboration file name is exactly `EG_ID_ALL` (this is not a real file name.), then all e*Gate System Events are successfully Identified. Also, if the Collaboration Rule file name is exactly `none`, then no processing is performed on the Event. Thus, to allow all e*Gate System Events to be sent to the external system unprocessed, type in:

```
EG_ID_ALL\^none
```

Lastly, if a Collaboration Rule file name is used, the Monk function must set a global Translation Status Monk Variable (whose name is configurable) to one of the recognized statuses before returning, using a Monk statement such as:

```
(define MONK_X_Status "OK")
```

See also

Load Custom Monk File Name on page 108.

Translation Status Monk Variable

Description

Specifies the name of the global Monk variable that determines the status of a Collaboration Rule.

Required Values

A string; the default is `MONK_X_Status`.

After translation is completed, the Translation function must set this Monk Variable with one of the following status (the first part of the value must start with it).

<table>
<thead>
<tr>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>The Collaboration is successful and the messages returned as a single string from the Monk function.</td>
</tr>
<tr>
<td>MULTI</td>
<td>The Collaboration is successful and there are multiple messages to be sent to the e<em>Gate System. Instead of returning the messages as a single string, they are placed into a file (see Output File Monk Variable) where the e</em>Way can read and send them on to the e*Gate System.</td>
</tr>
<tr>
<td>Hold</td>
<td>The Collaboration is successful but no messages resulting from the Collaboration are to be sent to the e*Gate System at this time.</td>
</tr>
<tr>
<td>Error:</td>
<td>Indicates that an error has occurred; the appropriate error status code follows the colon, for example:</td>
</tr>
<tr>
<td>Error: 04</td>
<td>Error within control information of EDI subsystem.</td>
</tr>
</tbody>
</table>


For example, if the Collaboration Rule errored due to a problem in the Control Record of the IDoc, then use the following Monk statement to set the Collaboration Status global variable:

\[
\text{(define MONK_X_Status"Error:04")}
\]

See also

- **Output File Monk Variable** on page 104
- **Output Record Delimiter Monk Variable** on page 104.

---

## Output File Monk Variable

**Description**

Specifies the name of the global Monk variable that determines the location of the Translation Output File, used only when the Translation Status Monk Variable is **MULTI**.

**Required Values**

A string; the default is **MONK_OutFile**.

**Additional Information**

Prior to calling the Collaboration Rule function, the SAP ALE e*Way presets this Monk variable to a unique file name under `\egate.store\<SystemData>\tmp`, which the Collaboration Rule function can use as-is, or could overload to another unique file name if so required.

## Output Record Delimiter Monk Variable

**Description**

Specifies the name of the global Monk Variable that sets the record delimiter for the Collaboration Output file. The e*Way presets this Monk variable to a newline character prior to calling the Collaboration Rule function. When setting this Monk variable in the Collaboration function, use either Monk predefined character constants such as `#\newline` or the `integer->char` conversion function to build the string.

**Note:**  Do not use escaped character sequences such as `\\n`.

**Required Values**

A string; the default is **MONK_RecDelim**.

### Status | Meaning
--- | ---
Error: 05 | Error during Collaboration.
Error: 07 | Error during syntax check.
Error: 11 | Error during dispatch (general error during execution of Monk function).
Persistent Memory File Monk Variable

Description
Specifies the global Monk variable that locates the Persistent Memory File, useful for storing information that needs to transcend cycling (shutting down and restarting) of this e*Way and hence the associated Monk engine.

Required Values
A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter.

The default is MONK_PMemFile.

Additional Information
This Persistent Memory File is especially useful when the Collaboration Status is HOLD, and may contain the actual held Event itself or a pointer to the file containing the accumulated Event.

If the e*Way were to restart, Event accumulation could continue until some criterion is achieved. The e*Way sets this Monk variable to the value specified in the Persistent Memory File prior to calling the Collaboration Rule.

Persistent Memory File Name (SAP to e*Gate)

Description
Specifies the file name where the Collaboration Rule data is to be stored. This parameter is necessary only when you want to save transitional Collaboration Rule data, such as when the Collaboration Status is HOLD. This file is free-format and can be manipulated via Monk File I/O functions.

Required Value
A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter.

This file is free-format and can be manipulated via Monk I/O functions:

<table>
<thead>
<tr>
<th>Function/Parameter(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>file-exists? file_name</td>
<td>Returns a boolean #t or #f depending on whether or not the file exists.</td>
</tr>
<tr>
<td>open-output-file file_name</td>
<td>Returns a file port for a new output file. If the file existed already, it is truncated and reopened.</td>
</tr>
<tr>
<td>open-append-file file_name</td>
<td>Returns a file port to a file for appending. If the file doesn't already exist, a new one is created and opened.</td>
</tr>
</tbody>
</table>
For example, to create a file that contains the following:

```
LastIdocNumber=17
StorageFile=/home/eGate/data/gather.dat
```

The Monk code used could be:

```
(define LastIdocNumber 17)
(define StorageFile "/home/eGate/data/gather.dat")
(define fp (open-output-file MONK _PMemFile))
(display (format "LastIdocNumber=%d\n" (number->string LastIdocNumber)) fp)
(display (format "StorageFile=%s\n" StorageFile) fp)
(close-port fp)
```

To read the parameters back from the file, the following Monk code could be used:

```
(define fp (open-random-access-file MONK_PMemFile))
(define line (read-line fp 80))
(define LastIdocNumber
  (string->number (substring line 15 (string-length line))))
(define line (read-line fp 80))
(define StorageFile (substring line 12 (string-length line)))
(close-port fp)
```
SAP IDoc Syntax Error Code (SAP to e*Gate)

Description
Specifies the SAP-recognized code for syntax error in the IDoc data (in other words., the data received does not conform with the corresponding IDoc Description File and, therefore, the IDoc Event Structure).

Required Values
A string, which must follow the format: Status:Description
where Status is the code recognized by SAP, and its Description is separated by a colon (:) with no spaces.

SAP IDoc Translation Error Code (SAP to e*Gate)

Description
Specifies the SAP-recognized code for translation error occurring with the IDoc data received.

Required Values
A string, which must follow the format: Status:Description
where Status is the code recognized by SAP, and its Description is separated by a colon (:) with no spaces.

SAP e*Way General Error Code (SAP to e*Gate)

Description
Specifies the SAP-recognized code for a general error with this e*Way (i.e., that which is dispatched by the SAP R/3 Sender to do a certain task).

Required Values
A string, which must follow the format: Status:Description
where Status is the code recognized by SAP, and its Description is separated by a colon (:) with no spaces.
7.6 Advanced Monk Setup

These parameters are provided to allow additional flexibility in setting up your system. They are ignored if left blank (as supplied), which is the recommended procedure if your system does not require their use.

Load Custom Monk File Name

Description

Loads the specified file when the e*Way starts up or the configuration is reloaded, and contains used-defined Monk functions to customize the Monk engine. The e*Way exits if it fails to load this file.

Required Value

A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter. There is no default value.

Note: This parameter is optional, and the field(s) may be left blank.

Startup Monk Function Name (SAP to e*Gate)

Description

Invoked by the SAP e*Way at startup time or when the configuration is reloaded. This function is used to initialize the Monk environment before processing external Events. It can also be used to initialize global Monk variables that may be used by other Monk functions. The e*Way exits if it fails to invoke this function or if this function returns a “FAILURE” string.

Required Values

A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter. There is no default value.

Additional Information

The function must have no arguments, and returns a string type.

Note: This parameter is optional, and the field(s) may be left blank.

Positive Acknowledgment Monk Function (SAP to e*Gate)

Description

This Monk function is called when the SAP e*Way successfully processes and queues IDoc Events from SAP to e*Gate. The function is given the queued IDoc Event string as the only input argument.
Chapter 7
Configuration Parameters

Section 7.6
Advanced Monk Setup

Required Values
A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter. There is no default value.

Note: This parameter is optional, and the field(s) may be left blank.

Positive Acknowledgment Function (e*Gate to SAP)

Description
This Monk function is called when the SAP e*Way successfully processes and queues data to SAP.

Required Values
None.

Note: This parameter is optional, and the field(s) may be left blank.

Negative Acknowledgment Function (SAP to e*Gate)

Description
This Monk function is called when the SAP e*Way fails to process and queue data from SAP.

Required Value
A file name, with an optional absolute path. If you provide a file name that is not an absolute path, then the value of SystemData from .egate.store, is prefixed to the value of the parameter. There is no default value.

Additional Information
The function is given the failed IDoc Event string as the only input argument.

Note: This parameter is optional, and the field(s) may be left blank.

Negative Acknowledgment Function (e*Gate to SAP)

Description
This Monk function is called when the SAP e*Way fails to process and queue data to SAP.

Required Value
None.

Note: This parameter is optional, and the field(s) may be left blank.
Utilities

A.1 stcjdump.exe

Journal files can be converted to a readable flat file by using the **stcjdump.exe** utility, for example:

```bash
stcjdump.exe/home/eGate/journal/SapAleIn.journal/home/eGate/tmp/xxx
```

**Usage:**

Dumps an e*Gate System Journal file into a flat file or vice versa relative to the order of the **-j** and **-f** options

```bash
stcjdump.exe -j {journal_pathname} -f {flatfile_pathname} [{delim_char_1} [...]]
```

where:

- **-j** precedes a Journal path name
- **-f** precedes a flat file path name
- **?-pathname** can be a relative path name, fully qualified path name, and for infile/outfile, ‘’’ is contextually either standard in or standard out
- **delim_char_1** can be specified as ‘\r’ or 0xd or 13 or ^M and separates records in the flat file
- **--ver** show version
- **-h** this screen

**Note:** The addition of the **-j** and **-f** prefix before the respective filenames. Depending on whether the Journal Filename or Flat Filename is specified first, the action taken is to dump a Journal into a flat file or vice versa.
IDoc Java Methods

Object methods associated with the SAP IDoc are as follows.

next
Description
Gets the next source event.
Signature
next()
Parameters
None.
Return Type
boolean
Throws
None

available
Description
This method checks to see whether or not the ETD has input data.
Signature
available()
Parameters
None.
Returns
A boolean true if data exists, false if no data exists.
Throws
None
**send**

**Description**

Puts the Event into the IQ as the pertinent Event Type.

**Signature**

```java
send()
```

**Parameters**

None

**Return Type**

`void`

**Throws**

None

---

**reset**

**Description**

Resets the IDoc data content.

**Signature**

```java
reset()
```

**Parameters**

None.

**Returns**

A boolean `true` if data content was reset successfully; `false` if not.

**Throws**

None.

---

**getIdocMsgFormat**

**Description**

Returns the IDoc message format used by this IDoc.

**Signature**

```java
getIdocMsgFormat()
```

**Parameters**

None.

**Returns**

A string indicating the IDoc message format (either "ALE" or "EDI").
Appendix B
IDoc Java Methods

Throws
None.

setIdocMsgFormatALE
Description
Changes the IDoc message format to be ALE.
Signature
setIdocMsgFormatALE()
Parameters
None.
Return Type
void
Throws
None.

setIdocMsgFormatEDI
Description
Changes the IDoc message format to be EDI.
Signature
setIdocMsgFormatEDI()
Parameters
None.
Return Type
void
Throws
None.

idocMsgFormatIs
Description
Checks the IDoc message format.
Signature
idocMsgFormatIs(MsgFormat)
Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsgFormat</td>
<td>java.lang.String</td>
<td>A string indicating the IDoc message format (either &quot;ALE&quot; or &quot;EDI&quot;)</td>
</tr>
</tbody>
</table>

Return Type

boolean

Throws

None
Index

A
ACC_EMPLOYEE_EXP 58
Assigning ETDs to Event Types 39
authorization objects 60
Autorun 19
available method 111

B
Build tool 32

C
CD-ROM 22
Change Request Inquiry 46
Changing the User Name 68
Collaboration 41, 72, 76
  Rules 72, 73, 76
  Rules Editor 76
  Service 72
Communications Port 53
components, e*Way 15
configuration
  Advanced Monk Setup 108–109
  ALE/RFC Setup 88–96
  Communication Setup 96–98
  General Settings 85–87
  Transaction Processing 99–107
configuration parameters
  Connection Duration Strategy 96
  Connection Error Resend Strategy 97
  Data Direction 85
  Database Entry Lifetime 101
  Database Purge Schedule 100
  e*Gate System Polling Interval 96
  Enforce Transactional RFC 99
  Expunge Database Entry On Confirm 101
  Idle Interval Before Disconnecting 97
  Incoming Raw Message Journal 86
  Interface Name 85
  Load Custom Monk File Name 108
  Logon Language 95
  Logon Password 94
  Logon User Name 94
Maximum Resend Attempts 98
Message Journal Archiving Schedule 86
Negative Acknowledgment Function 109
Outgoing Translated Message Journal 86
Output File Monk Variable 104
Output Record Delimiter Monk Variable 104
Persistent Memory File Monk Variable 105
Persistent Memory File Name 105
Positive Acknowledgment Function 109
Positive Acknowledgment Monk Function 108
Program ID 88
Rejected Message Journal 87
RFC Trace On 92
SAP Application Server 89
SAP Application Server Group 95
SAP Client Number 93
SAP Destination System 95
SAP e*Way General Error Code 107
SAP Gateway Host 90, 91
SAP Gateway Service 91
SAP IDoc Syntax Error Code 107
SAP IDoc Translation Error Code 107
SAP IDoc Version 88
SAP Load Balancing Usage 89
SAP Message Server 95
SAP Router String 90, 91
SAP System Client Number 94
SAP System ID 93
SAP System Number 90, 93
SAP System Type 88
SAP User Language 94
SAP User Name 93
Startup Monk Function Name 108
Time to Wait Before Attempting Connection 96
Time to Wait Before Becoming Server 96
Transaction Identification and Translation 102
Transactional ID Verification Database 99, 100
Translation Status Monk Variable 103
Validate SAP Sender Identity 92
Verification Database Operating Mode 100
configuration procedures
  e*Way 64
  SAP 43
Connection Duration Strategy 80
Connection Duration Strategy parameter 96
Connection Error Resend Strategy parameter 97
conventions, writing 9
Create Request 46

D
Data Direction parameter 85
Database Entry Lifetime parameter 101
Database Purge Schedule parameter 100
Distribution Model 43

E

e*Gate System Polling Interval parameter 96
e*Way
  components 15
  configuration 64
  creating 62
  Installation 19
  Properties 63
  Schedules 68
  setup procedures 61
  Startup Options 68
  troubleshooting 72
Editor
  Collaboration Rules 76
  ETD 32
Enforce Transactional RFC parameter 99
ERROR 83
ETD Editor 32
Event Type 39
Event Type Definition (ETD) 28, 39
Event Type Definition Editor 36
Events 75
Expunge Database Entry On Confirm parameter 101

G

getDocMsgFormat method 112
GNU 83

H

HOLD 83

I

Idle Interval Before Disconnecting parameter 97
IDoc Description File 29, 31
IDoc Types 11
IDoc Wizard 37
idocMsgFormatIs method 113
INBOUND_IDOC_PROCESS 83
Incoming Raw Message Journal parameter 86
Installation procedure
  UNIX systems 22
  Windows systems 19
InstallShield 19
Intelligent Queue (IQ) 42, 72
Interface Name parameter 85

J

Java IDoc methods
  available 111
  getDocMsgFormat 112
  idocMsgFormatIs 113
  next 111
  reset 112
  send 112
  setIdocMsgFormatALE 113
  setIdocMsgFormatEDI 113
Java Idoc Wizard 37

L

Load Custom Monk File Name parameter 108
logging options 70
Logical System 55
Logon Language parameter 95
Logon Password parameter 94
Logon User Name parameter 94

M

Maximum Resend Attempts parameter 98
Message Journal Archiving Schedule parameter 86
Message Types 11
method, Java - see Java IDoc methods
monitoring thresholds 71
Monk 13
Monk variable 80, 83
MONK_RecDelim 83
MONK_X_Status 83
MULTI 83

N

NAK 83
Negative Acknowledgment Function parameter 109
next method 111

O

OK 83
Operating Systems 16
Outgoing Translated Message Journal parameter 86
Output File Monk Variable parameter 104
Output IDoc records 30
Output IDoc types 30
Output Record Delimiter Monk Variable parameter 104
Output segment fields 30
Index

P
parameters - see configuration parameters
Participating Host 72
Partner 55
Partner Type 55
Persistent Memory File Monk Variable parameter 105
Persistent Memory File Name parameter 105
Positive Acknowledgment Function parameter 109
Positive Acknowledgment Monk Function parameter 108
procedures
  configuration (e*Way) 64
  configuration (SAP) 43
  installation 19
  setup (e*Way) 61
Program ID parameter 88
Properties, e*Way 63

Q
Queue - see Intelligent Queue (IQ)

R
Rejected Message Journal parameter 87
Remote Function Call (RFC) 75
reset method 112
RFC Trace On parameter 92
RFC, Transactional (tRFC) 75, 80, 83

S
SALE 45, 47
SAP Application Server Group parameter 95
SAP Application Server parameter 89
SAP Client Number parameter 93
SAP Destination System parameter 95
SAP e*Way General Error Code parameter 107
SAP Gateway Host parameter 90, 91
SAP Gateway Service parameter 91
SAP IDoc Syntax Error Code parameter 107
SAP IDoc Translation Error Code parameter 107
SAP IDoc Version parameter 88
SAP Load Balancing Usage parameter 89
SAP Message Server parameter 95
SAP R/3 configuration 43
SAP Router String parameter 90, 91
SAP System Client Number parameter 94
SAP System ID parameter 93
SAP System Number parameter 90, 93
SAP System Type parameter 88
SAP User Language parameter 94
SAP User Name parameter 93
Schedules 68
send method 112
setIDocMsgFormatALE method 113
setIDocMsgFormatEDI method 113
Setting Startup Options or Schedules 68
Startup Monk Function Name parameter 108
Startup Options 68

T
Time to Wait Before Attempting Connection parameter 96
Time to Wait Before Becoming Server parameter 96
Transaction Identification and Translation parameter 102
Transactional ID (TID) 75, 80, 83
Transactional ID Verification Database parameter 99, 100
Transactional RFC (tRFC) 75, 80, 83
Transfer List to a Local File 31
Translation Status Monk Variable parameter 103
troubleshooting the e*Way 72

U
UNIX installation procedure 22
User name 68

V
Validate SAP Sender Identity parameter 92
Verification Database Operating Mode parameter 100

W
WEDI 55, 56
Windows installation procedure 19
Wizard
  IDoc (Java) 37
writing conventions 9