BEA eLink Adapter for Vantive

User Guide

BEA eLink Adapter for Vantive Version 1.1
Document Edition 1.1
January 2000

Copyright

Restricted Rights Legend

Trademarks or Service Marks

BEA eLink Adapter for Vantive User Guide

Document Edition	Part Number	Date	Software Version
1.1		January 2000	BEA eLink Adapter for Vantive 1.1

Contents

3. Configuring the Environment for the eLink Adapter for Vantive

4. Running eLink Adapter for Vantive

A. Error and Informational Messages

Preface

This chapter consists of the following sections:

- Purpose of This Document
- How to Use This Document
- Related Documentation
- Contact Information

Purpose of This Document

This document describes the BEA eLink Adapter for Vantive product and gives instructions for transferring data between Vantive and TUXEDO applications.

Who Should Read This Document

This document is intended for system administrators who will install the eLink Adapter for Vantive, as well as programmers who will configure the eLink Adapter for Vantive and set up TUXEDO services to execute information transfers with Vantive Enterprise applications. This guide assumes knowledge of BEA TUXEDO and Vantive products.

How This Document Is Organized

The *BEA eLink Adapter for Vantive User Guide* is organized as follows:

- Chapter 1, "Understanding EAI and the Role of eLink Adapters," introduces the eLink Adapter component and explains how the eLink Adapter for Vantive fits into the BEA eLink platform environment.
- Chapter 2, "Installing eLink Adapter for Vantive," explains how to install the eLink Adapter for Vantive component.
- Chapter 3, "Configuring the Environment for the eLink Adapter for Vantive," provides information for configuring BEA eLink platform servers and Vantive Enterprise applications.
- Chapter 4, "Running eLink Adapter for Vantive," provides information about initiating information transfers between a BEA eLink platform environment and Vantive Enterprise applications.
- Appendix A, "Error and Informational Messages," describes error and informational messages as well as actions to resolve the errors.

e-docs Web Site

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

How to Print the Document

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

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

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

How to Use This Document

The *eLink Adapter for Vantive User Guide* is designed as a HTML and PDF document and can be viewed online or printed.

Documentation Conventions

The following documentation conventions are used throughout this document.

Item	Examples
Variable names	Variable names represent information you must supply or output information that can change; they are intended to be replaced by actual names. Variable names are displayed in italics and can include hyphens but not underscores. The following are examples of variable names in text: *error-file-name* The *when-return* value...

User input and screen output	For screen displays and other examples of input and output, user input appears as in the first of the following lines; system output appears as in the second through fourth lines:
	`dir c:\accounting\data`
	`Volume in drive C is WIN_NT_1`
	`Volume Serial Number is 1234-5678`
	`Directory of C:\BEADIR\DATA`

Syntax	Code samples can include the following elements: ♦ Variable names can include hyphens but not underscores (e.g., *error-file-name)* ♦ Optional items are enclosed in square brackets: [ ]. If you include an optional item, do not code the square brackets. ♦ A required element for which alternatives exist is enclosed in braces {}. The alternatives are separated by the pipe (vertical bar) character: |. You must include only one of the alternatives for that element. Do not code the braces or pipe character. ♦ An ellipsis (...) indicates that the preceding element can be repeated as necessary.
Omitted code	An ellipsis (...) is used in examples to indicate that code that is not pertinent to the discussion is omitted. The ellipsis can be horizontal or vertical.
Environment variables	Environment variables are formatted in an uppercase font. ENVFILE=${APPDIR}
Key names	Key names are presented in boldface type. Press **Enter** to continue.

Item	Examples
Literals	Literals are formatted in a monospace font. `class extendSample`
Window items	Window items are presented in boldface type. Window items can be window titles, button labels, text edit box names or other parts of the window. Type your password in the **Logon window**. Select **Export** to make the service available to the client.

Related Documentation

The following sections list the documentation provided with the eLink Adapter for Vantive software, and other publications related to TUXEDO technology.

BEA eLink Adapter for Vantive Documentation

The eLink Adapter for Vantive information set consists of the following documents:

BEA eLink Adapter for Vantive User Guide

BEA eLink Adapter for Vantive Release Notes

Note: The BEA eLink Adapter for Vantive Online Documentation CD also includes Adobe Acrobat PDF files of all of the online documents. You can use the Adobe Acrobat Reader to print all or a portion of each document.

BEA Publications

The following BEA publications are also available:

BEA TUXEDO Application Development Guide

TUXEDO System 6 Reference Manual

TUXEDO System 6 Programmer's Guide, Volumes 1 and 2

Vantive Publications

The following Vantive publications are also available:

Vantive API Reference Guide

Contact Information

The following sections provide information about how to obtain support for the documentation and software.

Documentation Support

If you have questions or comments on the documentation, you can contact the BEA Information Engineering Group by e-mail at **docsupport@beasys.com** (For information about how to contact Customer Support, refer to the following section.)

Customer Support

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

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

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

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

CHAPTER

1 Understanding EAI and the Role of eLink Adapters

This chapter contains the following topics:

- BEA eLink Solution Overview
- BEA eLink Adapter for Vantive Overview

BEA eLink Solution Overview

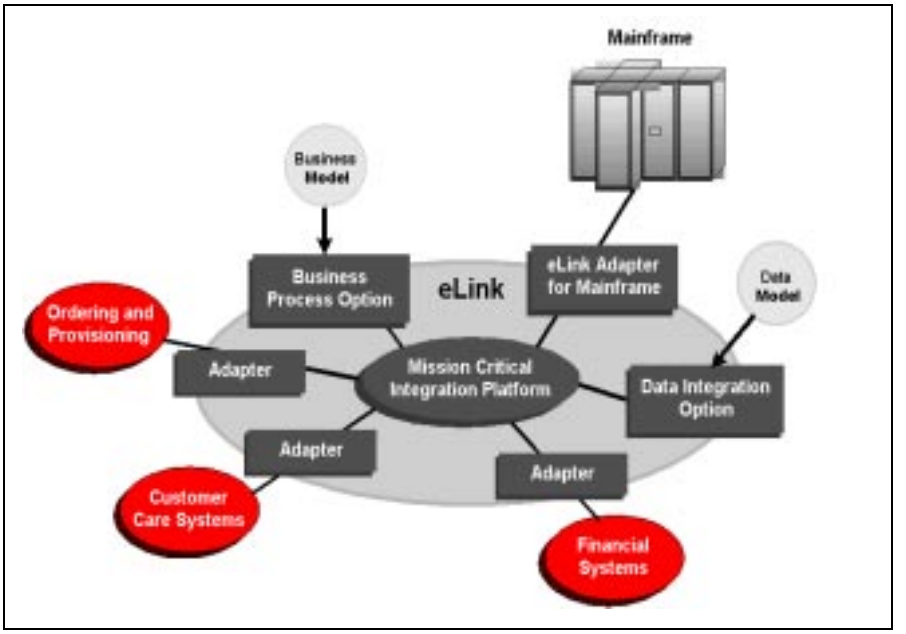
BEA Enterprise Application Integration (EAI) provides an open solution that allows applications throughout organizations to communicate seamlessly. Using EAI, you gain the long-term flexibility and investment protection you need to keep up with today's ever-changing business environment.

Typically, companies use packaged applications to automate internal operations, such as financial, manufacturing, human resources, etc. While they successfully address the needs of these specific areas, these proprietary platforms do not work together. To compete today, you need a much greater exchange of information. Systems need to communicate at both a database and a process level, within your own organization as well as with customer's and supplier's systems. BEA eLink Platform is the underlying

basis of BEA eLink, a family of off-the-shelf enterprise application integration (EAI) products that leverage BEA's transaction platform to integrate existing legacy applications with customer-focused and business-to-business e-commerce initiatives.

BEA eLink Platform provides a proven, rock-solid infrastructure for integrating applications within the enterprise and across the Web. BEA eLink Platform ensures high-performance, secure transactions and transparent access to mission-critical applications and information throughout the enterprise and across the Web. Figure 1-1 illustrates the eLink logical architecture and shows where the eLink Adapters fit into the process.

Figure 1-1 BEA eLink Solution Illustration



The BEA eLink Platform

The BEA eLink Platform (in addition to all options and adapters) is highly scalable. Multiple instances of BEA eLink Platforms can collaborate so that work is divided between eLink instances and domains. BEA eLink includes SNMP integration for

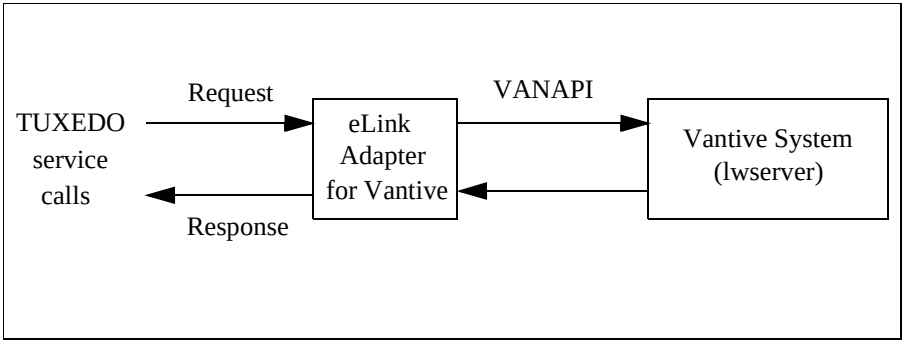
enterprise management. The BEA eLink Platform features compliance with the Open Group's X/Open standards including support of the XA standard for Two-phase commit processing, the X/Open ATMI API, and XPG standards for language internationalization. C, C++ and Java (via Jolt) are supported. The BEA eLink Platform connects to any RDBMS, OODBMS, file manager or queue manager. The following components operate with BEA eLink Platform:

- The Data Integration Option translates data models used by different applications into a common data format. It provides a cost-effective alternative to writing or generating programs to perform this function. It also handles complex translation with greater power and scalability than rules engines and formatters.
- The Business Process Option helps automate tasks in the business process and dynamically responds to business events and exceptions.
- The eLink Adapter provide the interface between the BEA eLink Platform and external applications, with out-of-the-box functionality (no programming required).

BEA eLink Adapter for Vantive Overview

The BEA eLink Adapter for Vantive software provides communication between Vantive Enterprise applications and BEA TUXEDO applications. The eLink Adapter for Vantive provides functionality to query and update information that is stored and maintained by Vantive.

Figure 1-2 BEA eLink Adapter for Vantive Illustration



The eLink Adapter for Vantive offers one TUXEDO service called EL_VANTIVE_IN. This is a generic service that processes all the requests to invoke Vantive functionality. The eLink Adapter for Vantive advertises the Vantive functionality that it invokes as TUXEDO services and aliases them to the EL_VANTIVE_IN service.

Vantive functionality that can be invoked and their advertised service names are configurable.

Understanding the EL_VANTIVE_IN Service

The eLink Adapter for Vantive consists of one generic service, EL_VANTIVE_IN. This service can be advertised with different names to invoke Vantive functionality. The eLink Adapter for Vantive service can be invoked by different TUXEDO applications.

The EL_VANTIVE_IN service is generic in nature and does not possess any knowledge about the Vantive business objects that it is going to operate upon. The information required to perform any business operation, such as the type of operation, business objects to act upon, or the input and output fields are read from the eLink Adapter for Vantive configuration file. This allows the service to be generic in nature.

When a request is made, the service receives the FML buffer. The service then determines the object or form to be used and opens it in Vantive using the VANAPI functions. The VANAPI allows the eLink Adapter for Vantive to call the Vantive

application and invoke business services. Any mandatory field error will be generated by Vantive functions and passed to calling applications using standard TUXEDO error management functions.

Upon completing the operation, the service returns the values for all the output fields listed in the configuration file using the FML32 buffer.

2 Installing eLink Adapter for Vantive

This chapter contains information for installing the eLink Adapter for Vantive software and consists of the following sections:

- Pre-Installation Considerations
- Installing BEA eLink Adapter for Vantive
- Installing on a UNIX Platform
- Installing on a Windows NT Platform
- Distribution Libraries and Executables

Pre-Installation Considerations

The eLink Adapter for Vantive software runs on HP-UX, Solaris, AIX, and Windows NT. Complete the following tasks prior to installing the eLink Adapter for Vantive:

- Read the *BEA eLink Adapter for Vantive Release Notes*
- Install and verify the operation of the BEA TUXEDO product
- Install the Vantive API product

The current BEA eLink Platform leverages the BEA TUXEDO infrastructure because it is based on a service-oriented architecture. Both BEA TUXEDO and BEA eLink communicate directly with each other and with other applications through the use of

services. Multiple services are grouped into “application servers” or “servers”. The terms, TUXEDO services/servers and eLink services/servers can be used interchangeably. Because this document is specifically addressing the eLink family, the term “eLink service” and “eLink server” is used throughout.

Configuring the Environment to Install the eLink Adapter for Vantive

Before installing the eLink Adapter for Vantive, you must configure the environment properly. Ensure that BEA TUXEDO is configured properly.

Installing BEA eLink Adapter for Vantive

The eLink Adapter for Vantive software will run on HP-UX, Solaris, AIX, and Windows NT. Refer to the following section for installation instructions.

Installing on a UNIX Platform

To install the eLink Adapter for Vantive software, you run the `install.sh` script. This script installs all the necessary software components.

Perform the following steps to install the eLink Adapter for Vantive software on a supported HP-UX, Solaris, or AIX platform.

1. Log on as root to install eLink for Vantive.

```
$ su -
Password:
```

2. Access the CD-ROM device.

```
# ls -1 /dev/cdrom
total 0
brw-rw-rw-   1 root   sys    27,   0 September 27  10:55 c1b0t010
```

3. Mount the CD-ROM.

   ```
   # mount -r -F cdfs /dev/cdrom/c1b0t010 /mnt
   ```

4. Change the directory to your CD-ROM device.

   ```
   # cd /mnt
   ```

5. List the CD-ROM contents.

   ```
   # ls
   install.sh  hp  ibm  sun5x
   ```

6. Execute the installation script.

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

7. The installation script runs and prompts you for responses. Listing 3-1 is an example of the installation script. The entries in bold represent user responses.

Listing 2-1 Install.sh Example

```
01) hp/hpux1020      02) hp/hpux11 03) ibm/aix43
04) sun5x/sol26      05) sun5x/sol7


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

** You have chosen to install from evan **

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

To terminate the installation at any time
press the interrupt key,
typically <del>, <break>, or <ctrl+c>.

The following packages are available:

  1     evan       BEA eLink Adapter for Vantive

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

BEA eLink Adapter for Vantive
(9000) Release 1.1
Copyright (c) 2000 BEA Systems, Inc.
All Rights Reserved.
```

```
Distributed under license by BEA Systems, Inc.
BEA eLink is a trademark of BEA Systems, Inc.


Directory where Vantive Adapter files are to be installed
(Enter your eLink Platform directory path) [?,q]: /bea/work/mytux

Using  /bea/work/mytux as the Vantive Adapter base directory

Determining if sufficient space is available ...
200 blocks are required
5080010 blocks are available to /bea/work/mytux

Unloading /cmhome/dist/ant-1/hp/hpux11/evan/EVANT65.Z ...
bin/ELINKVANO
bin/lic.sh
eLink/vantive/simpvantive/ReadCust
eLink/vantive/simpvantive/UpdateCust
eLink/vantive/simpvantive/elinkvantive.cfg
eLink/vantive/simpvantive/elinkvantive.env
eLink/vantive/simpvantive/elinkvantive.ubb
eLink/vantive/simpvantive/readme.txt
eLink/vantive/simpvantive/setenv.bat
eLink/vantive/simpvantive/setenv.sh
eLink/vantive/simpvantive/vam_clnt.c
lib/libadk.sl
udataobj/elinkvantive.fml
340 blocks
...finished

Changing file permissions...
...finished

If your license file is accessible, you may install it now.
Install license file? [y/n]:n

Please don't forget to use lic.sh located in your product bin
directory to install the license file from the enclosed floppy.

Refer to your product Release Notes for details on how to do this.

Installation of BEA eLink Adapter for Vantive was successful

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

8. Change the directory to your root directory.

   ```
   # cd /
   ```

9. Unmount the CD-ROM device.

Installing on a Windows NT Platform

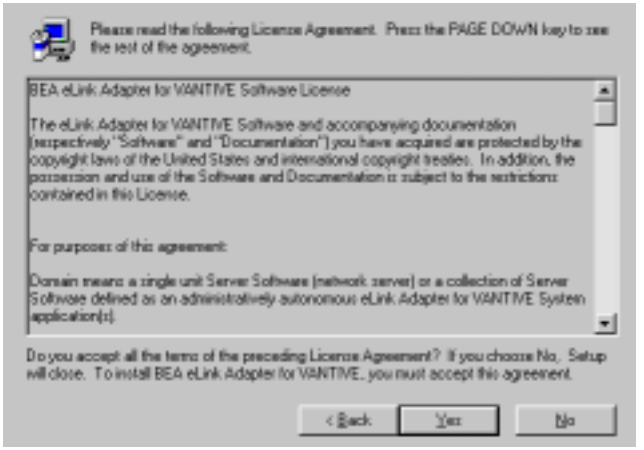
Perform the following steps to install the eLink Adapter for Vantive software on the Windows NT platform.

1. Insert the product CD_ROM and click the **Run** option from the **Start** menu. The **Run** window displays. Click **Browse** to select the CD_ROM drive. Change directories to the `winnt` directory and select the `Setup.exe` program. Click **OK** to run the executable and begin the installation. The following **Welcome** screen displays. Click **Next** to continue with the installation.

Figure 2-1 Welcome Screen

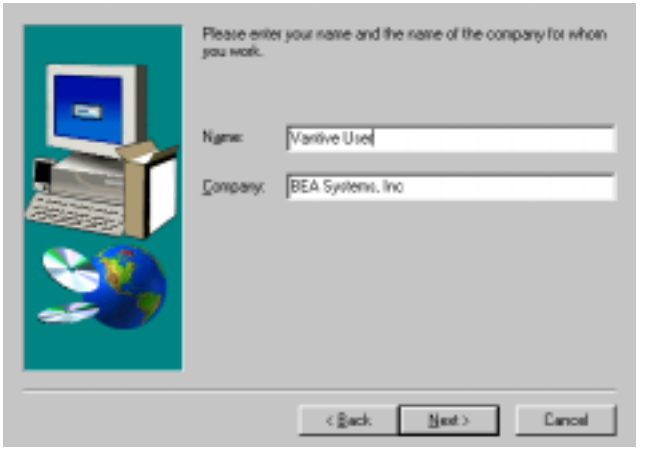
2. The BEA Software License Agreement displays. Click **Yes** to accept the terms of the agreement and continue with the product installation. Click **No** to exit the installation process.

Figure 2-2 BEA Software License Agreement Screen



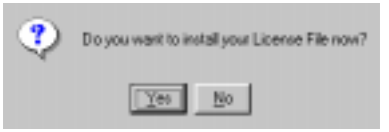
3. The **User Information** screen displays after the License Agreement. Enter the name of the eLink Platform System Administrator in the **Name** field. Enter the name of your company in the **Company** field. Click **Next** to continue with the installation.

Figure 2-3 User Information Screen



4. The **Install License File?** option screen displays next. You may select **Yes** to install your BEA Software License File, or you may select **No** to bypass this step and continue installing the eLink Adapter for Vantive software. If you select **Yes**, continue with Step 5. If you select **No**, continue with Step 7.

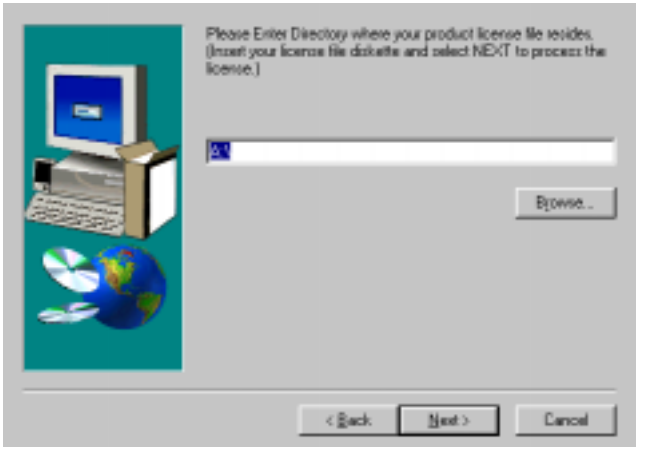
Figure 2-4 Install License File? Screen



5. The **Select License File Source Directory** screen displays. Enter the directory path where your license file resides in the field. You can browse and click directories by clicking the **Browse** button. Typically, the license file is installed in the `tuxedo/udataobj` directory.

   If you entered a valid directory path, click **Next** to continue with the installation. Go to Step 7. If you entered an invalid directory path, go to Step 6.

Figure 2-5 Select License File Source Directory Screen



6. If you do not enter a valid directory path for your license file, the installation software generates an error message dialog box. You can select **Yes** to enter a valid directory path, or you can select **No** to continue with the installation. If you select **No**, the installation software automatically searches for the TUXEDO software. If it finds TUXEDO installed, the installation software completes the process. If TUXEDO is not found, the installation software aborts the process.

   Note: If you select **No**, the installation continues but an error is generated in the ulog.*mm/dd/yy* file indicating that the product is unlicensed. Please refer to the "Using the License Key" section of the *BEA eLink Adapter for Vantive Release Notes* for instructions on using the license file.

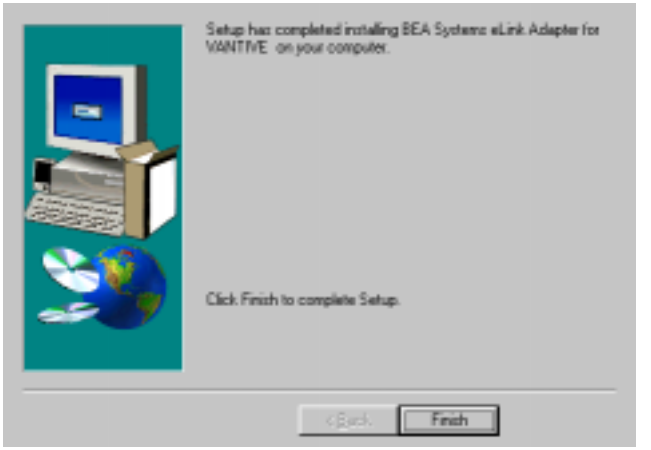
   Once you have entered a valid directory path, click **Next** to continue with the installation. Go to Step 7.

Figure 2-6 Invalid License File Directory Dialog Box

7. A progress bar displays showing the status of the installation.

8. The **Setup Complete** screen displays notifying you that the BEA eLink Adapter for Vantive product is installed on your system. Click **Finish** to complete the Setup process.

Figure 2-7 Setup Complete Screen

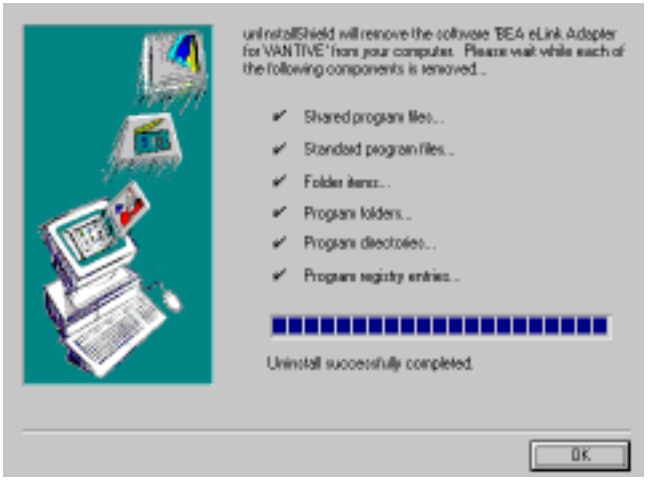


Uninstalling eLink Adapter for Vantive on Windows NT

Perform the following steps to uninstall the eLink Adapter for Vantive on a Windows NT system.

1. Click the Start button, and then point to **Settings**. Point to the folder that contains **Control Panel**, and then click **Control Panel**.
2. Double click on the **Add/Remove Programs** option from the **Control Panel** listings to access the **Add/Remove Programs** properties window.
3. In the **Add/Remove Program** properties window, select **eLink Adapter for Vantive** from the program list and click the **Add/Remove** button.
4. The uninstall process for elink Adapter for Vantive begins. The **Remove Programs From Your Computer** screen displays. Click **OK** to complete the uninstall process.

Figure 2-8 Remove Programs From Your Computer Screen



Distribution Libraries and Executables

The eLink Adapter for Vantive CD-ROM contains the following libraries and executable programs. After installing the eLink Adapter for Vantive software, verify that these libraries and programs are installed on your system.

Verify that the following files are installed by the eLink Adapter for Vantive software.

Table 2-1 Distribution Libraries and Executables

Directory	Files
/bin	ELINKVANO lic.sh

3 Configuring the Environment for the eLink Adapter for Vantive

This chapter consists of the following topics:

- Defining the ELINKVANO Server
- Setting Up the eLink Adapter for Vantive Configuration File
- Understanding Service Invocation Requirements

Defining the ELINKVANO Server

As with any TUXEDO server, you must update the SERVERS section of the UBBCONFIG file to establish a server group for the eLink Adapter for Vantive in the BEA TUXEDO configuration.

See the *BEA TUXEDO Administrator's Guide* for more information about the UBBCONFIG file. For more information about the UBBCONFIG file specific to servers, refer to the "Updating the SERVERS Section" of this document.

Note: Lines beginning with an asterisk (*) indicate the beginning of a specification section. Each such line contains the name of the section immediately following the *. The asterisk is required when specifying a section name.

Updating the SERVERS Section of the UBBCONFIG File

This section explains how to specify servers in the BEA TUXEDO configuration.

Define the ELINKVANO Server

You define the ELINKVANO server in the SERVERS section of the TUXEDO UBBCONFIG file as follows:

To define this server, add the ELINKVANO information in the SERVER section of the UBBCONFIG file. The following parameters are required for defining the ELINKVANO server.

Listing 3-1 Syntax for ELINKVANO Server Definition in the UBBCONFIG File

```
*SERVERS

ELINKVANO
       SRVGRP=.. SRVID=..
       CLOPT = " -- -i unique_id -C configuration filename"
```

For more information about the SRVGRP, SRVID, and CLOPT parameter syntax and definitions, refer to the *BEA TUXEDO Reference Manual.*

CLOPT= `" -- -i unique_id"`

specifies the unique id identifying the label section in the environment file containing the adapter environment variables. This is optional. If it is not specified, then the unique id will default to the process name.

`"-C configuration filename"`

specifies the configuration file specifying the detailed adapter configuration information.

Sample UBBCONFIG File

Listing 3-2 is a sample UBBCONFIG file. In this sample, the ELINKVANO server is defined in the SERVER section with the required CLOPT options specified.

Listing 3-2 Sample UBBCONFIG File

```
*RESOURCES

IPCKEY        123791
DOMAINID      simpapp
MASTER        simple

*MACHINES

DALNT6
       LMID          = simple
       TUXDIR        = "\tuxedo"
       TUXCONFIG     = "\myappdir\tuxconfig"
       APPDIR        = "\myappdir"
       FIELDTBL      = "sample.fml"
       FIELDTBL32    = "sample.fml"
       FLDTBLDIR     = "\myappdir"
       FLDTBLDIR32   = "\myappdir"
       ULOGPFX       = "\myappdir\ULOG"
                                # LD_LIBRARY_PATH=\vantive
                                # SHLIB_PATH=\vantive
                                PATH=\vantive


*GROUPS

eLINK
       LMID=simple   GRPNO=1

*SERVERS

DEFAULT:
       CLOPT="-C"

ELINKVANO
       SRVGRP=eLINK         SRVID=10
       REPLYQ=N
       CLOPT="-- -i unique_id -C elinkvantive.cfg"
```

```
*SERVICES

*ROUTING
```

Setting Up the eLink Adapter for Vantive Configuration File

The eLink Adapter for Vantive retrieves configuration information from its own configuration file (elvan_env.cfg) located in the APPDIR directory. This configuration parameters file defines detailed configuration information required by the eLink Adapter for Vantive. The eLink Adapter for Vantive reads-in the adapter label section of the specified environment file.

Note: The configuration file must be specified in the CLOPT line of the UBBCONFIG file or the eLink Adapter for Vantive will generate an error and exit the startup.

If you are unsure of a parameters value, consult your Vantive System Administrator for the proper configuration file values.

The eLink Adapter for Vantive configuration file consists of the following sections.

*SERVER

The configuration file SERVER section defines the parameters needed to establish connections with the Vantive application, and to set any global flags (such as debug flags) within the server. The beginning of the SERVER section is denoted by a line containing the following text string "*SERVER".

NAME=<*adapter unique id*>

Defines the label section containing the adapter environment variables. The adapter unique id will either be the value of the -i flag on the CLOPT line, or if not specified, then the default value of the process name.

EXIT_CONNECT_LOSS=<*Y*>

Optional. Specifies if the eLink Adapter for Vantive is to exit when it detects that it has lost the connection to the Vantive application server. If not specified or defined as a value other then Y, then the eLink Adapter for Vantive will not exit upon losing the connection and will try to restore the connection.

TRACE=<*Y*>

Optional. Defines if trace output will be produced. If not specified or defined as a value other than Y, then no trace output will be produced.

VANTIVE_HOST=<*Vantive application server host name*>

Defines the host name on which the Vantive application server (iwserver) resides.

VANTIVE_PORT=<*Vantive Application Server port number*>

Defines the port number to connect to for the Vantive application server (iwserver).

SERVICE_LIST=<*list of services to be advertised*>

Defines the services that the eLink Adapter for Vantive will advertise. The services will be specified as a list which is comma-delimited: *service_1, service_2, service_3,...*

LOGIN_NAME=<*Vantive application login name*>

Defines the login name to be used while connecting to the Vantive server. Every Vantive user is assigned a unique login name and it is configured using the VanTools utility. Each user is a part of a "group". Every group has access to specific Vantive applications such as Vantive Sales, Field Service, or Quality. Depending on the permissions assigned to every group, the user can access specific Vantive forms.

PASSWORD=<*Vantive Application login password*>

Defines the password to be used along with the login name while connecting to the Vantive application server.

FUNCTION_NAME=<*Vantive interface function name*>

Optional. Defines the Vantive interface function that is to be executed when the service specified in the label is invoked. If not specified, then the service name will be used as the interface function name. The set of interface function names will be specified by the eLink Adapter for Vantive and define the Vantive functionality that can be invoked.

Refer to the Vantive documentation for the specific format of the FUNCTION_NME parameter.

*SERVICE

For each service that is advertised by the adapter, a SERVICE section must be specified in the configuration file that defines what the service does within the Vantive application. Each SERVICE section defines the operation and data that is to be manipulated within Vantive. The beginning of the SERVICE section is denoted by a line containing the following text string "*SERVICE".

NAME=<*service name*>

For each service in the list defined by the SERVICE_LIST parameter, a label will exist where the label is the service name. For each service, the label section will contain parameters that define configuration information particular to that service. The parameters that are applicable to specific services are defined in the section describing that service.

PRIMARY_KEY=<*Primary Key field for the parent object*>

Specifies the name of the primary key for the parent Vantive object. When an operation is performed on a vantive child object, it needs the parent object name (OBJECT_NAME) and the primary key for the parent.

OBJECT_NAME=<*Vantive business object name*>

Defines the Vantive object name on which the service will operate. This object name along with the form name will be used by the VANAPI to access the Vantive business object related information from the Vantive application database. In case of a service operating on child form/object, this variable defines the parent object name.

Refer to the Vantive documentation for the specific format of the OBJECT_NAME parameter.

FORM_NAME=<*Vantive form name*>

Every TUXEDO service will manipulate a given Vantive business object. Every object is associated with a Vantive Form. Different groups can have different forms for the same business object to manipulate specific information. This variable will indicate which form the service will use to manipulate the given business object. Only the forms associated with child objects and/or tables will be aded to the configuration file. If this parameter is defined, then the form will be opened as a child form and the OBJECT_NAME will be used as parent object name.

Refer to the Vantive documentation for the specific format of the FORM_NAME parameter.

If the form name is not mentioned for the given service then it is the service dealing with main business object and main form associated with it.

OPERATION=<*requested operation*>

Specifies the operation to be performed on a Vantive object. The permitted values are

New

Specifies to create a new Vantive business object.

Read

Specifies to read the Vantive business object fields. This operation may return more than one instance of the object to satisfy the search criteria. Only the values of the fields specified in the OUTFLD_LIST variable in the configuration file will be returned to the caller. Every field will have as many occurrences as there are instances of the object returned by the VANAPI function.

Update

Specifies to update the existing Vantive business object fields. This operation will work for only one object at every invocation of the service. Therefore, the input field and mandatory field values should be able to identify a unique object so as to perform the operation successfully.

Delete

Specifies to delete the Vantive business object. This operation works only for one object at every invocation of the service. Therefore, the input field and mandatory field values should be able to identify a unique object so as to perform the operation successfully.

Note: Update and Delete operations require a record to be uniquely identified. When working with child objects in Vantive, the input fields and mandatory fields should uniquely identify the child record irrespective of selection criteria used for the parent object.

INPFLD_LIST=<*Input fields received by the service*>

Defines the field names that can be expected as input to the given service. This list should contain the mandatory fields for the execution of the service. The input fields will be specified as a list which is comma-delimited.

Refer to the Vantive documentation for the specific format of the INPFLD_LIST parameter.

MANFLD_LIST=<*Mandatory Input fields received by the service*>

Defines the field names that are mandatory as input to the given service. The optional fields are listed in the input field list and the fields shall not be repeated in both the input and mandatory field lists. When this field list is used for the Update operation, it will contain all the mandatory fields (Id fields) required to uniquely identify the record. All the linked fields such as swRegionId | sw_region.swName will be added to the list in its entirety.

Refer to the Vantive documentation for the specific format of the MANFLD_LIST parameter.

OUTFLD_LIST=<*Output fields returned by the service*>

Defines the field names that will be returned by the service on completion. The output fields will be specified as a list which is comma-delimited.

Refer to the Vantive documentation for the specific format of the OUTFLD_LIST parameter.

KEYFLD_LIST=<*Key fields for parent object*>

Defines the key fields that will be used to identify the parent object when working with child forms and objects/records. When more than one field is mentioned, an AND condition will be assumed for locating the object/record.

Refer to the Vantive documentation for the specific format of the KEYFLD_LIST parameter.

KEYFLD=<*Object name associated with every key field*>

Specifies the associated object name for every key field mentioned in the KEYFLD_LIST. Whenever Vantive contains more than one linked fields (those associated with the child objects) it is necessary to identify every key field with the corresponding Vantive object. This section repeats for every key field mentioned in the KEYFLD_LIST array and defines the Vantive object to which it is associated. The actual key is the primary key of the linked (child) table associated with the main form. In many cases the Key Field itself is a primary of the child table, in which case the actual key can be left blank.

Refer to the Vantive documentation for the specific format of the KEYFLD_LIST parameter.

Sample Configuration File

Listing 3-1 is a sample configuration file. The elvan_env.cfg file is the configuration file that the server reads.

Listing 3-3 Sample Environment Configuration File

```
*SERVER
NAME=SALES
VANTIVE_HOST=127.0.0.1
VANTIVE_PORT=1540
TRACE=Y
RESPONSE_BUFFER_SIZE=12000
LOGIN_NAME=vantest1
PASSWORD=vantest1
SERVICE_LIST=RdCustSales,NewCustSales,UpdCustSales,DelCustSales,
NewSiteSales, UpdSiteSales, RdSiteSales, DelSiteSales,
NewContSales, UpdContSales, RdContSales, DelContSales,
NewIndSales, UpdIndSales, RdIndSales, DelIndSales, NewRegSales,
UpdRegSales, RdRegSales, DelRegSales, NAddCustSales,
UAddCustSales, RAddCustSales, DAddCustSales, NAddSiteSales,
RAddSiteSales, DAddSiteSales, UAddSiteSales, NJorContSales,
UJorContSales, RJorContSales, DJorContSales

*SERVICE
NAME=RdCustSales
OBJECT_NAME=CUSTOMER
OPERATION=Read
MANFLD_LIST=swName
OUTFLD_LIST=swName, swParentId|sw_Customer.swname, swStatus,
swCreditRating, swMainPhoneCntry, swMainPhoneArea, swMainPhone,
swMainFaxCntry, swMainFaxArea, swMainFax, swType, swDuns, swUrl,
swOwnership, swRevenue, swRegionId|sw_Region.swName, swGeoCode,
swLocType, swEmpTotal, swIndustryId|SW_INDUSTRY.swName,
swDefaultReplyVia, swTpm, swCarrier, swNote

*SERVICE
NAME=NewCustSales
OBJECT_NAME=CUSTOMER
OPERATION=New
INPFLD_LIST=swParentId|SW_CUSTOMER.swName, swStatus,
swCreditRating, swMainPhoneCntry, swMainFaxCntry, swType, swDuns,
swUrl, swOwnership, swRevenue, swRegionId|SW_REGION.swName,
swGeoCode, swLocType, swEmpTotal, swIndustryId|SW_INDUSTRY.swName,
swDefaultReplyVia, swTpm, swCarrier, swNote
```

```
MANFLD_LIST=swName, swMainPhoneArea, swMainPhone, swMainFaxArea
,swMainFax
KEYFLD_LIST=swParentId, swRegionId, swIndustryId
swParentId=CUSTOMER, swCustomerId
swRegionId=REGION
swIndustryId=INDUSTRY
OUTFLD_LIST=swCustomerId

*SERVICE
NAME=UpdCustSales
OBJECT_NAME=CUSTOMER
FORM_NAME=
OPERATION=Update
INPFLD_LIST= swParentId|SW_CUSTOMER.swName, swStatus,
swMainPhoneCntry,  swType,  swUrl,  swRegionId|sw_Region.swName,
swIndustryId|SW_INDUSTRY.swName, swDefaultReplyVia, swTpm,
swCarrier, swNote, swMainPhoneArea, swMainPhone, swMainFaxArea
,swMainFax
MANFLD_LIST=swName
OUTFLD_LIST=swCustomerId
KEYFLD_LIST=swParentId, swRegionId, swIndustryId
swParentId=CUSTOMER, swCustomerId
swRegionId=REGION
swIndustryId=INDUSTRY

*SERVICE
NAME=DelCustSales
OBJECT_NAME=CUSTOMER
FORM_NAME=
OPERATION=Delete
INPFLD_LIST=
MANFLD_LIST=swName
OUTFLD_LIST=swName

*SERVICE
NAME=NewSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=
FORM_NAME=
OPERATION=New
INPFLD_LIST=swLanguage, swRegionId|sw_Region.swName,
swRepairDepotSite, swOfficePhoneCntry, swOfficePhoneArea,
swFaxCntry, swFaxArea, swFax, swOfficeCode
MANFLD_LIST=swSiteName, swCustomerId|sw_customer.swName
KEYFLD_LIST=swCustomerId, swRegionId
swCustomerId=CUSTOMER
swRegionId=REGION
OUTFLD_LIST=swSiteId
```

```
*SERVICE
NAME=UpdSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=
FORM_NAME=
OPERATION=Update
INPFLD_LIST=swLanguage, swRegionId|sw_Region.swName,
swRepairDepotSite, swOfficePhoneCntry, swOfficePhoneArea,
swFaxCntry, swFaxArea, swFax, swOfficeCode
MANFLD_LIST=swSiteName
KEYFLD_LIST=swRegionId
swRegionId=REGION
OUTFLD_LIST=swSiteId

*SERVICE
NAME=RdSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=
FORM_NAME=
OPERATION=Read
INPFLD_LIST=
OUTFLD_LIST=swLanguage, swRegionId|sw_Region.swName,
swRepairDepotSite, swOfficePhoneCntry, swOfficePhoneArea,
swFaxCntry, swFaxArea, swFax, swOfficeCode
MANFLD_LIST=swSiteName

*SERVICE
NAME=DelSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=
FORM_NAME=
OPERATION=Delete
INPFLD_LIST=
OUTFLD_LIST=swSiteName
MANFLD_LIST=swSiteName

*SERVICE
NAME=NewContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=
FORM_NAME=
OPERATION=New
INPFLD_LIST=swTitle, swCustomerId|sw_Customer.swName,
swSiteId|sw_Site.swSiteName, swRegionId|sw_Region.swName,
swDepartment, swJobTitle, swType, swSalutation, swEmailType,
swLogin, swUrl, swOfficePhoneCntry, swOfficePhoneArea,
swOfficePhone, swOfficePhoneExt, swFaxCntry, swFaxArea, swFax,
swAltPhoneCntry, swAltPhoneArea, swAltPhone, swAltPhoneExt,
swEmailAddress, swExtEmailAddress, swOs, swOsversion, swPlatform,
```

```
swNetwork, swUI, swEnvironment
MANFLD_LIST=swFirstName, swLastName
KEYFLD_LIST=swCustomerId, swRegionId, swSiteId
swCustomerId=CUSTOMER
swRegionId=REGION
swSiteId=SITE
OUTFLD_LIST=swFirstName, swLastName

*SERVICE
NAME=UpdContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=
FORM_NAME=
OPERATION=Update
INPFLD_LIST=swTitle, swCustomerId|sw_Customer.swName,
swSiteId|sw_Site.swSiteName, swRegionId|sw_Region.swName,
swDepartment, swJobTitle, swType, swSalutation, swEmailType,
swLogin, swUrl, swOfficePhoneCntry, swOfficePhoneArea,
swOfficePhone, swOfficePhoneExt, swFaxCntry, swFaxArea, swFax,
swAltPhoneCntry, swAltPhoneArea, swAltPhone, swAltPhoneExt,
swEmailAddress, swExtEmailAddress, swOs, swOsversion, swPlatform,
swNetwork, swUI, swEnvironment
MANFLD_LIST=swFirstName, swLastName
KEYFLD_LIST=swCustomerId, swRegionId, swSiteId
swCustomerId=CUSTOMER
swRegionId=REGION
swSiteId=SITE
OUTFLD_LIST=swFirstName, swLastName

*SERVICE
NAME=RdContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=
FORM_NAME=
OPERATION=Read
OUTFLD_LIST=swTitle, swCustomerId|sw_Customer.swName,
swSiteId|sw_Site.swSiteName, swRegionId|sw_Region.swName,
swDepartment, swJobTitle, swType, swSalutation, swEmailType,
swLogin, swUrl, swOfficePhoneCntry, swOfficePhoneArea,
swOfficePhone, swOfficePhoneExt, swFaxCntry, swFaxArea, swFax,
swAltPhoneCntry, swAltPhoneArea, swAltPhone, swAltPhoneExt,
swEmailAddress, swExtEmailAddress, swOs, swOsversion, swPlatform,
swNetwork, swUI, swEnvironment, swFirstName, swLastName
MANFLD_LIST=swFirstName, swLastName

*SERVICE
NAME=DelContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=
```

```
FORM_NAME=
OPERATION=Delete
OUTFLD_LIST=swFirstName, swLastName
MANFLD_LIST=swFirstName, swLastName

*SERVICE
NAME=NewIndSales
OBJECT_NAME=INDUSTRY
PRIMARY_KEY=
FORM_NAME=
OPERATION=New
OUTFLD_LIST=swIndustryId, swName
MANFLD_LIST=swName
INPFLD_LIST=swSIC, swParentId|sw_industry.swName, swOrder, swNote
KEYFLD_LIST=swParentId
swParentId=INDUSTRY, swIndustryId

*SERVICE
NAME=UpdIndSales
OBJECT_NAME=INDUSTRY
PRIMARY_KEY=
FORM_NAME=
OPERATION=Update
OUTFLD_LIST=swIndustryId, swName
MANFLD_LIST=swName
INPFLD_LIST=swSIC, swParentId|sw_industry.swName, swOrder, swNote
KEYFLD_LIST=swParentId
swParentId=INDUSTRY, swIndustryId

*SERVICE
NAME=RdIndSales
OBJECT_NAME=INDUSTRY
PRIMARY_KEY=
FORM_NAME=
OPERATION=Read
MANFLD_LIST=swName
OUTFLD_LIST=swSIC, swParentId|sw_industry.swName, swOrder, swNote,
swName

*SERVICE
NAME=DelIndSales
OBJECT_NAME=INDUSTRY
PRIMARY_KEY=
FORM_NAME=
OPERATION=Delete
MANFLD_LIST=swIndustryId
OUTFLD_LIST=swName, swIndustryId
```

```
*SERVICE
NAME=NewRegSales
OBJECT_NAME=REGION
PRIMARY_KEY=
FORM_NAME=
OPERATION=New
OUTFLD_LIST=swRegionId, swName
MANFLD_LIST=swName, swRegionTypeId|sw_Region_type.swType
INPFLD_LIST=swOrder, swParentRegionId|sw_Region.swName, swActive,
swNote
KEYFLD_LIST=swRegionTypeId, swParentRegionId
swParentRegionId=REGION, swRegionId
swRegionTypeId=REGION_TYPE

*SERVICE
NAME=UpdRegSales
OBJECT_NAME=REGION
PRIMARY_KEY=
FORM_NAME=
OPERATION=Update
OUTFLD_LIST=swRegionId, swName
MANFLD_LIST=swRegionId
INPFLD_LIST=swParentRegionId|SW_REGION.swName, swOrder,
swTopRegionId|SW_REGION.swName, swActive, swNote,
swRegionTypeId|SW_REGION_TYPE.swType
KEYFLD_LIST=swRegionTypeId, swParentRegionId, swTopRegionId
swRegionTypeId=REGION_TYPE
swParentRegionId=REGION, swRegionId
swTopRegionId=REGION, swRegionId

*SERVICE
NAME=RdRegSales
OBJECT_NAME=REGION
PRIMARY_KEY=
FORM_NAME=
OPERATION=Read
OUTFLD_LIST=swName, swRegionTypeId|SW_REGION_TYPE.swType,
swParentRegionId|SW_REGION.swName, swOrder,
swTopRegionId|SW_REGION.swName, swActive, swNote
MANFLD_LIST=swRegionId

*SERVICE
NAME=DelRegSales
OBJECT_NAME=REGION
PRIMARY_KEY=
FORM_NAME=
OPERATION=Delete
OUTFLD_LIST=swName, swRegionId
MANFLD_LIST=swRegionId
```

```
*SERVICE
NAME=NAddCustSales
OBJECT_NAME=CUSTOMER
PRIMARY_KEY=swCustomerId
FORM_NAME=swAddressSuPu
OPERATION=New
OUTFLD_LIST=swAddressId
INPFLD_LIST=swType, swRegionId|SW_REGION.swName, swAddress1,
swAddress2, swAddress3, swAddress4, swAddress5, swCity, swState,
swZip, swCountry, swPOBox, swMailStop, swNote
KEYFLD_LIST=swRegionId
swRegionId=REGION

*SERVICE
NAME=UAddCustSales
OBJECT_NAME=CUSTOMER
PRIMARY_KEY=swCustomerId
FORM_NAME=swAddressSuPu
OPERATION=Update
OUTFLD_LIST=swAddressId
INPFLD_LIST=swType, swRegionId|SW_REGION.swName, swAddress1,
swAddress2, swAddress3, swAddress4, swAddress5, swCity, swZip,
swCountry, swPOBox, swMailStop, swNote, swState
MANFLD_LIST=swAddressId
KEYFLD_LIST=swRegionId
swRegionId=REGION

*SERVICE
NAME=RAddCustSales
OBJECT_NAME=CUSTOMER
PRIMARY_KEY=swCustomerId
FORM_NAME=swAddressSuPu
OPERATION=Read
OUTFLD_LIST= swType, swRegionId|SW_REGION.swName, swAddress1,
swAddress2, swAddress3, swAddress4, swAddress5, swCity, swState,
swZip, swCountry, swPOBox, swMailStop, swNote, swAddressId
MANFLD_LIST=swAddressId

*SERVICE
NAME=DAddCustSales
OBJECT_NAME=CUSTOMER
PRIMARY_KEY=swCustomerId
FORM_NAME=swAddressSuPu
OPERATION=Delete
OUTFLD_LIST= swAddressId
MANFLD_LIST=swAddressId
```

```
*SERVICE
NAME=NAddSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=swSiteId
FORM_NAME=swAddressSuPu
OPERATION=New
OUTFLD_LIST=swAddressId
INPFLD_LIST= swType, swRegionId|SW_REGION.swName, swAddress1,
swAddress2, swAddress3, swAddress4, swAddress5, swCity, swState,
swZip, swCountry, swPOBox, swMailStop, swNote
KEYFLD_LIST=swRegionId
swRegionId=REGION

*SERVICE
NAME=RAddSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=swSiteId
FORM_NAME=swAddressSuPu
OPERATION=Read
OUTFLD_LIST= swType, swRegionId|SW_REGION.swName, swAddress1,
swAddress2, swAddress3, swAddress4, swAddress5, swCity, swState,
swZip, swCountry, swPOBox, swMailStop, swNote
MANFLD_LIST=swAddressId

*SERVICE
NAME=DaddSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=swSiteId
FORM_NAME=swAddressSuPu
OPERATION=Delete
OUTFLD_LIST= swAddressId
MANFLD_LIST=swAddressId

*SERVICE
NAME=UAddSiteSales
OBJECT_NAME=SITE
PRIMARY_KEY=swSiteId
FORM_NAME=swAddressSuPu
OPERATION=Update
OUTFLD_LIST=swAddressId
INPFLD_LIST=swType, swRegionId|SW_REGION.swName, swAddress1,
swAddress2, swAddress3, swAddress4, swAddress5, swCity, swState,
swZip, swCountry, swPOBox, swMailStop, swNote
MANFLD_LIST=swAddressId
KEYFLD_LIST=swRegionId
swRegionId=REGION

*SERVICE
NAME=NjorContSales
```

```
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=swPersonId
FORM_NAME=swWorkLogContactOLPu
OPERATION=New
OUTFLD_LIST=swSubject, swWorkLogId
MANFLD_LIST=swAction, swVisibility, swSubject
INPFLD_LIST= swNote, swTotalTime, swStatus

*SERVICE
NAME=UjorContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=swPersonId
FORM_NAME=swWorkLogContactOLPu
OPERATION=Update
OUTFLD_LIST=swSubject, swWorkLogId
MANFLD_LIST=swWorkLogId
INPFLD_LIST=swNote, swStatus, swAction, swVisibility, swSubject

*SERVICE
NAME=RJorContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=swPersonId
FORM_NAME=swWorkLogContactOLPu
OPERATION=Read
OUTFLD_LIST= swNote, swTotalTime, swStatus, swAction, swVisibility,
swSubject, swWorkLogId
MANFLD_LIST=swWorkLogId

*SERVICE
NAME=DJorContSales
OBJECT_NAME=OUTSIDECONTACT
PRIMARY_KEY=swPersonId
FORM_NAME=swWorkLogContactOLPu
OPERATION=Delete
OUTFLD_LIST=swWorkLogId, swSubject
MANFLD_LIST=swWorkLogId
```

Understanding Service Invocation Requirements

Each unique business-level function that can be invoked by the eLink Adapter for Vantive is advertised as a TUXEDO service. To invoke a service, a calling application prepares an FML32 request buffer specifying the input values that are to be passed to Vantive. The calling application then invokes the corresponding TUXEDO service, passing the FML32 request buffer.

The eLink Adapter for Vantive has a generic EL_VANTIVE_IN service that processes all the service requests. When invoked, the service code determines what service name is used to invoke it. The service code then calls a function that processes the requests. This function takes the service name and FML32 request buffer as input parameters, and returns the FML32 response buffer and error information as ouput parameters. From the service name, the function determines the Vantive functionality to invoke. The function that processes the FML32 request buffer, invokes the Vantive functionality, and then returns the response parameters in an FML32 buffer. If any errors ocur, then the function returns error information.

The service code processes the response from the function. If the Vantive functionality was invoked successfully, then it will return TPSUCCESS with the `tpurcode` set to 0. If the invocation failed then the service code returns the corect error indication.

The eLink Adapter for Vantive uses only FML32 Field names, not Field IDs, when processing the request and response buffers. The Field names must be defined in the TUXEDO FML Field Table file. This allows the actual Field IDs to be customer-defined.

4 Running eLink Adapter for Vantive

The eLink Adapter for Vantive connects to the Vantive system using the VANAPI to communicate and access data from the Vantive database using different forms. The elink Adapter for Vantive reads a server configuration file and attempts to connect to the specified Vantive server.

Running the eLink Adapter for Vantive consists of the following startup operations:

- Specifying the Configuration File
- Determining the Adapter Unique ID
- Configuring Environment Variables
- Advertising Services for Vantive Applications

Specifying the Configuration File

The eLink Adapter for Vantive configuration file must be specified on the CLOPT line of the UBBCONFIG file or the eLink Adapter for Vantive will generate an error and exit the startup.

Determining the Adapter Unique ID

The eLink Adapter for Vantive determines the eLink Adapter unique id from the eLink Adapter label section in the eLink Adapter for Vantive configuration file. This will either be the value of the -i flag on the CLOPT line of the UBBCONFIG file, or if not specified, the default value of the process name. For more information on determining the eLink Adapter unique id, refer to the ‘Setting Up the eLink Adapter for Vantive Configuration File” on page 3-5.

Configuring Environment Variables

The eLink Adapter for Vantive reads the configuration in the eLink Adapter label section of the specified eLink Adapter for Vantive configuration file. If any required configuration parameters are missing, then the eLink Adapter for Vantive generates an error and exits the startup procedure.

Vantive server host name, port number, login name and password are stored in the configuration file. Every service, on reading this file, can connect to the Vantive application server with a specific login name.

Advertising Services for Vantive Applications

The eLink Adapter for Vantive parses the list of services specified by the SERVICE_LIST configuration parameter and performs the following configuration for each service.

Each service has a label in the configuration file where the label name is the service name. This label section contains the specified configuration parameters for that service.

The eLink Adapter for Vantive then caches the service configuration in memory and this information is used when the service is invoked. Once all the service names specified in the configuration file have been processed, the eLink Adapter for Vantive advertises the service names.

A Error and Informational Messages

This document contains the following descriptions of error, informational, and warning messages that may be encountered while using the BEA eLink Adapter for Vantive component.

1001	**ERROR: No memory allocated by the system**	
	Description:	A general memory allocation failure has occured.
	Action:	Try closing some applications running on the server host, and boot the server again.
	See Also:	None.
1002	**ERROR: The Environment file is absent**	
	Description:	The eLink Adapter for Vantive cannot find the eLink Adapter for Vantive configuration file listed in the UBBCONFIG file. This configuration file contains environment variables used by the eLink Adapter for Vantive.
	Action:	Check the filename supplied with the -e option in the UBBCONFIG file and check if the file is present in the application directory.
	See Also:	None.

1004	**ERROR: The Environment variable SERVICE_LIST is absent in the configuration file.**	
	Description:	The eLink Adapter for Vantive configuration file does not contain a list of services to be advertised.
	Action:	Define the SERVICE_LIST variable indicating the list of services to be advertised by the eLink Adapter for Vantive.
	See Also:	None.
1005	**ERROR: The Environment variable TRACE is absent in the configuration file.**	
	Description:	The eLink Adapter for Vantive configuration file does not define the TRACE variable. This variable is used to trace the adpter output.
	Action:	Set TRACE to **y** or **n**.
	See Also:	None.
1006	**ERROR: The Servie label is absent in the configuration file.**	
	Description:	The eLink Adapter for Vantive configuration file does not contain a SERVICE section for the service referenced in the SERVICE_LIST.
	Action:	Add a SERVICE section for the service to be advertised.
	See Also:	None.
1010	**ERROR: Duplicate Service. This Service already exists.**	
	Description:	The eLink Adapter for Vantive configuration file contains duplicate service names.
	Action:	Remove the service name duplication from the configuration file.
	See Also:	None.
1011	**ERROR: Error occured during tpadvertise()**	
	Description:	The tpadvertise() cal for the given service failed.
	Action:	Check the TUXEDO configuration file for possible errors.

	See Also:	None.
1012	**ERROR: Error finding the desired Configuration Node in the tree.**	
	Description:	The eLink Adapter for Vantive cannot locate the node in the tree. The eLink Adapter for Vantive stores all the configuration parameters in a tree format.
	Action:	Check the eLink Adapter for Vantive configuration file to make sure that the configuration information for the service is present.
	See Also:	None.
1016	**ERROR: The OPERATION Environmnt variable is absent in the Configuration**	
	Description:	Define the OPERATION VARAIBLE to be defined by the service. The operation can be New, Delete, Update, and Read.
	Action:	Add a operation type in the service section.
	See Also:	None.
1017	**ERROR: The Environment variable FORM_NAME is absent in the Configuration file.**	
	Description:	Form name is not mentioned in the service section. Form name is used when the service is acting on Vantive child objects.
	Action:	Add a name in the SERVICE section when working with a Vantive child object.
	See Also:	None.
1019	**ERROR: The PRIMARY _KEY Environment variable is absent in the configuration file.**	
	Description:	The Primary Key for the parent option is not defined. This is a required field when working with Vantive child objects.
	Action:	Add a name in the SERVICE section of the configuration file when working with Vantive child objects.
	See Also:	None.

1020	**ERROR: Can't determine appropriate VANAPI function for service**	
	Description:	The OPERATION DATA in the configuration file is wrong. The proper coarse of action cannot be established.
	Action:	Update the configuration file.
	See Also:	None.
1021	**ERROR: Configuration not specified for the service name.**	
	Description:	Configuration data for the service is absent from the configuration file.
	Action:	Update the configuration file.
	See Also:	None.
1022	**ERROR: Required field not found in the request buffer.**	
	Description:	A mandatory field was not supplied in the FML request buffer.
	Action:	Add the mandatory field to the request buffer, and repeat the request.
	See Also:	None.
1023	**ERROR: The Object Name is absent in the Configuration file.**	
	Description:	An object name that is required for the service named in the message has not been supplied in the configuration file.
	Action:	Update the configuration file.
	See Also:	None.