



▶▶ SSA® Integration Pack 1.3.1 for SSA ERP LN Vertex O Series

Installation and Configuration Guide

© Copyright 2006 by SSA Global Technologies, Inc. and its Subsidiaries and Affiliates

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any way or by any means, including, without limitation, photocopying or recording, without the prior written consent of SSA Global Technologies™, Inc.

Important Notices

The material contained in this publication (including any supplementary information) constitutes and contains confidential and proprietary information of SSA Global Technologies, Inc. or its Subsidiaries and Affiliates.

By gaining access to the attached, you acknowledge and agree that the material (including any modification, translation or adaptation of the material) and all copyright, trade secrets and all other right, title and interest therein, are the sole property of SSA Global Technologies, Inc. and/or its Subsidiaries and Affiliates and that you shall not gain right, title or interest in the material (including any modification, translation or adaptation of the material) by virtue of your review thereof other than the non-exclusive right to use the material solely in connection with and the furtherance of your license and use of software made available to your company from SSA Global Technologies, Inc. or its Subsidiaries and Affiliates (as applicable) pursuant to a separate agreement ("Purpose").

In addition, by accessing the enclosed material, you acknowledge and agree that you are required to maintain such material in strict confidence and that your use of such material is limited to the Purpose described above.

Although SSA Global Technologies, Inc. and/or its Subsidiaries and Affiliates has taken due care to ensure that the material included in this publication is accurate and complete, neither SSA Global Technologies, Inc. or its Subsidiaries or Affiliates can warrant that the information contained in this publication is complete, does not contain typographical or other errors, or will meet your specific requirements. As such, neither SSA Global Technologies, Inc. or its Subsidiaries or Affiliates assumes and hereby disclaims all liability, consequential or otherwise, for any loss or damage to any person or entity which is caused by or relates to errors or omissions in this publication (including any supplementary information), whether such errors or omissions result from negligence, accident or any other cause.

Trademark Acknowledgements

SSA, SSA Global and SSA Global Technologies are trademarks or registered trademarks of SSA Global Technologies, Inc. in the United States and/or other countries.

Microsoft WINDOWS®, is a registered trademark of Microsoft Corporation.

IBM®, DB2®, AIX®, AS/400®, OS/400®, Informix and Informix® Dynamic Server™ are all trademarks of IBM Corporation in USA and/or other countries.

UNIX® is a registered trademark of the Open Group.

HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA® is a registered trademark of Sun Microsystems, Inc.

All other company, product, trade or service names referenced may be registered trademarks or trademarks of their respective owners.

Publication Information

Document code: U9035B USU9035B US

Release: SSA® Integration Pack 1.3.1 for SSA ERP LN Vertex O Series

Publication date: July 06

Table of Contents

Chapter 1	Introduction.....	1-1
Chapter 2	Installation.....	2-1
	Prerequisites.....	2-1
	To install the SSA Integration Pack	2-2
	To start the SSA Integration Pack.....	2-5
	Windows	2-5
	UNIX	2-5
	To specify a BusComponent.....	2-6
	To stop the SSA Integration Pack.....	2-6
	Test the SSA Integration Pack Installation.....	2-6
	To run the SSA Integration Pack as Service.....	2-6
	Windows	2-6
	Troubleshooting	2-9
Chapter 3	Configuration	3-1
	General	3-1
	Defaults.....	3-2
	The BusComponent	3-2
	The SSA Integration Pack.....	3-3
	Logging	3-4
	Other configuration files	3-5

Chapter 4 Auxiliary Tools4-1

 Parameters4-1

 vertexTest4-2

 getGeocode4-2

 calculateTax.....4-2

 quoteTax.....4-3

 purchaseOrderTPP4-3

 TaxOnlyAdjustmentSales.....4-3

Chapter 5 Troubleshooting5-1

 OW_C4JAVA_HOME not defined.....5-1

 Unknown bus component5-2

About this Guide

This document is an Installation and Configuration Guide that describes how to install and configure the SSA Integration Pack 1.3.1 for SSA ERP LN and Vertex O Series. This document is intended for system administrators.

This document contains the following chapters and appendices:

Chapter 1, “Introduction,” describes the general concepts of the Integration Pack on a functional level. This chapter describes the various components of the Integration Pack, and describes how these components work together.

Chapter 2, “Installation,” describes the procedure to install the Integration Pack.

Chapter 3, “Configuration,” describes how to configure the Integration Pack so that you can begin to adopt applications.

Chapter 4, “Auxiliary Tools,” describes tools to test the Integration Pack.

Chapter 5, “Troubleshooting,” describes problems and solutions.

Definitions, acronyms, and abbreviations

Term	Definition
API	Application Programming Interface
BAANHOME	Environment variable for SLM installation directory, which is %BAANHOME% for Windows and \$BAANHOME for UNIX and AS/400.
BCLM	Baan Common License Manager: The former name of the SSA License Manager. The SSA License Manager is still abbreviated as BCLM on implementation level to keep the license manager backward compatible.
BusComponent	Socket stored in an LDAP
Company Names	SSA Global Technologies, Inc.

Term	Definition
DLL	Dynamic Link Library: A collection of precompiled and frequently used routines not explicitly linked to every program that uses them.
HTTP	Hyper Text Transfer Protocol
IP	Internet Protocol
J2SE	Server base Java development environment from sun
LDAP	Directory Service of SSA-OA
Port	Communication Endpoint for a computer
SLM	SSA License Manager: The product previously known as the Baan Common License Manager (BCLM).
SLM server	The SLM engine that handles license requests and management requests.
SLSA	Software License and Support Agreement: The contract between the customer and SSA Global, which states, among other things, the applications, the type of licensing, and the amount of licenses. You can usually find that information in Schedule A of the SLSA.
Socket	Communication endpoint in an application, which is addressed with an IP address and a port number
UI	User Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
Users File	The XML data file with all the users of a named user license for a particular product, including the user name, the user type, and the number of licenses a user is permitted to use.
XML	Extensible Markup Language

Send us your comments

We continually review and improve our documentation. Any remarks/requests for information concerning this document or topic are appreciated. Please e-mail your comments to documentation@ssaglobal.com.

In your e-mail, refer to the document number and title. More specific information will enable us to process feedback efficiently.

Chapter 1

Introduction

1

SSA Integration Pack 1.3.1 connects to SSA ERP LN 6.1 SP3 by means of SSA Open Architecture 6.0 with the Vertex O Series 2.2 interface.

The following figure illustrates the software components involved in the integration to Vertex O Series:

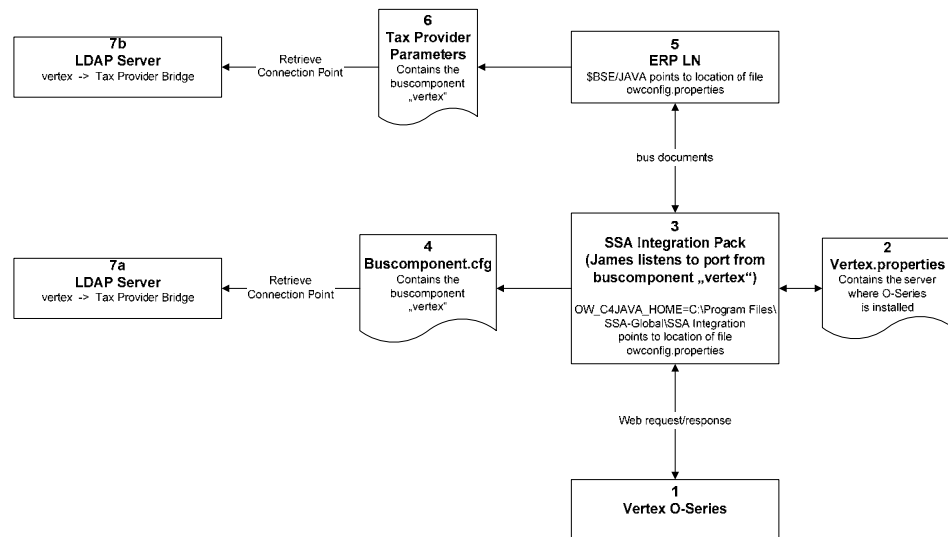


Figure 1-1: Component Architecture

The following list describes the various components labeled 1 through 7b in the figure:

- 1 Server on which the Vertex O Series Web service runs.
- 2 File that contains the socket the Vertex O Series Web service runs.
- 3 The Integration Pack converts the bus documents into a Vertex O Series Web service request, and vice versa.
- 4 File that contains the BusComponent of the LDAP specified in OW_C4JAVA_HOME, which is used to receive incoming BusDocuments sent by ERP LN.
- 5 In \$BSE/JAVA, owconfig.properties contain the location of the LDAP used by SSA ERP LN Open Architecture.
- 6 The Tax Provider Parameter table contains the BusComponent of the LDAP to be used to connect to the SSA Integration Pack.
- 7 Consists of the following LDAPs:
 - a LDAP of the SSA Integration Pack
 - b LDAP used by SSA ERP LN

Note: These LDAP servers can be identical.

For technical details on how to configure your SSA Integration Pack, refer to the “Configuration” chapter.

This chapter describes how to install the SSA Integration Pack.

For detailed information on how to configure a working SSA Integration Pack, refer to the “Configuration” chapter.

Prerequisites

Before you install the SSA Integration Pack, check that your system meets the following prerequisites:

- Java Runtime Environment 1.4
- Software License Manager (SLM) installed
- SSA Open Architecture 6.0 Server for Java with a license for the Java Server’s server component (product ID 7062)
- Vertex O Series 2.2 with Web-service component installed and running
- SSA ERP _{LN} 6.1 FP3
- For Windows: Microsoft Resource Kit to run the SSA Integration Pack as service on the installation server. Download the Resource Kit applicable to the Windows Server version on which you want to install the SSA Integration Pack from <http://www.microsoft.com/downloads>.

To install the SSA Integration Pack

You receive the SSA Integration Pack in the form of a .zip archive. If you work with a Windows system, use the .zip archive. If you work with a UNIX system, use the .gz archive.

Windows:

Use WinZip to extract the archive to the directory (*<installDir>*) on which you want the SSA Integration Pack to run.

UNIX:

- 1 Extract the .gz archive, for example, using the following command:

```
gzip -d SSA Integration Pack 1.3.1.tar.gz
```

- 2 Unpack the resulting .tar archive, for example, using the following command:

```
tar xf SSA Integration Pack 1.3.1.tar
```

- 3 Move the contents of the unpacked archive to the directory in which you want to run the SSA Integration Pack.

The *<installDir>* directory must contain the following directories and files:

- bin/
- doc/
- etc/
- lib/
- README
- version.txt

You must make the software aware of the other software on which it depends. You can do this in one of two ways:

- Set the environment variables to point to the relevant installation directories:

Environment Variables

Variable	Description
BAANHOME	<p>The SSA License Manager installation directory. In particular, the following file, among others must be present:</p> <p><code><BAANHOME>/java/com_baan_bclm.jar</code></p> <p>If SSA Open Architecture is installed, this variable is already set.</p>
OW_C4JAVA_HOME	<p>The Connector for Java installation directory. In particular, the following files, among others, must be present:</p> <p><code><OW_C4JAVA_HOME>/java/ow.jar</code> <code><OW_C4JAVA_HOME>/java/ow3p.jar</code></p> <p>On Windows, for example, the directory is: C:\Program Files\SSA-Global\SSA Integration</p> <p>You must set this variable.</p>
JVM_OPTIONS	<p>Set this file to reserve, for example, additional memory for the VM:</p> <p><code>JVM_OPTIONS = -Xmx256m</code></p>

- Create or edit a configuration file or script to supply the software with information about where the other components are installed. You can do this in various ways, depending on the platform you want to install.

Be sure to modify the files buscomponent.cfg and vertex.properties, as described in the “Configuration” chapter.

SSA Global recommends the following setups for your system:

Set the environment variables on Microsoft Windows

- 1 Edit the setEnv.bat file, which is available in the following directory:

```
<installDir>/bin/
```

- 2 Uncomment the appropriate lines and fill the variables, as described previously in the Environment Variables table.

Set the environment variables on UNIX and AS/400

Two methods are available to configure the SSA Integration Pack's software dependencies on UNIX platforms:

- System-wide
Involves setting up an .owTpBridge file in the /etc directory.
- User-specific
Involves setting up an .owTpBridge file in the \${HOME} directory.

The user-specific variant takes precedence over a system-wide file. The syntax of the two files is exactly the same. In fact, both files are sourced in from a shell script. A template is available in:

```
<installDir>/etc/owTpBridge.template.
```

You can use this template to uncomment the appropriate lines and fill in the variables, as described previously in this chapter in the Environment Variables table.

The following is an example for the owTpBridge file:

```
#####
# Installation configuration properties
#####
## Uncomment this line to enable debug output
# DEBUG=on
## Uncomment and set this variable to specify the SLM installation directory
# BAANHOME=/usr/baan/shared
## Uncomment and set this variable to specify the SSA Open Arch. Connector
## for Java installation directory
# OW_C4JAVA_HOME=/baan/b61A_cus/bse
## Uncomment and change this variable if your java installation has not added
## the java command to the PATH environment variable or if you wish to use an
## alternative java implementation
# JAVA_CMD=/opt/java1.4/jre/bin/java
## Uncomment and set this variable if you wish to pass extra options to the
## Java VM
# JVM_OPTIONS=
```

To start the SSA Integration Pack

The process to start the SSA Integration Pack is similar on all supported platforms. You must first configure the SSA Integration Pack; otherwise, the SSA Integration Pack will most likely fail to start. The configuration options are described previously in this chapter. At a minimum, you must add a BusComponent to the LDAP of your SSA Open Architecture installation and you must configure the address.lookup.tax_areas and address.calculate.tax properties in the following file: `<installDir>/vertex.properties`.

Windows

- 1 Open a command console and run the following batch file:

```
<installDir>\bin\startBridge
```

- 2 Pass the following as an argument if you do not use the default BusComponent **vertex**:

```
-Dbc=<mybuscomponentName>
```

Example

```
<installDir>\bin\startBridge -Dbc=myBusComponent
```

UNIX

- 1 Run the following script:

```
$ <installDir>/bin/startBridge
```

- 2 Start the SSA Integration Pack using the following command if you do not use the standard BusComponent **vertex**,:

```
$ <installDir>/bin/startBridge -Dbc=myBusComponent
```

To specify a BusComponent

On both Windows and UNIX platforms, you can optionally specify the BusComponent to listen for incoming requests from SSA ERP. For example, on a UNIX machine, you specify the BusComponents as follows:

```
$ <installDir>/bin/startBridge -Dbc=myBusComponent
```

If you do not specify a BusComponent, the BusComponent set in the configuration files is used. For more information, refer to the file `configure.txt`.

To stop the SSA Integration Pack

To stop the SSA Integration Pack, you must end the SSA Integration Pack process, for example, press CTRL+C, or send a hang-up (HUP) signal to the process, such as the following:

```
$ kill <processID>
```

Test the SSA Integration Pack Installation

As part of this delivery, some tools are packaged that enable you to test whether vertex is up and running and whether the SSA Integration Pack is up and running. The `tools.txt` document describes each of these test tools in detail.

To run the SSA Integration Pack as Service

Windows

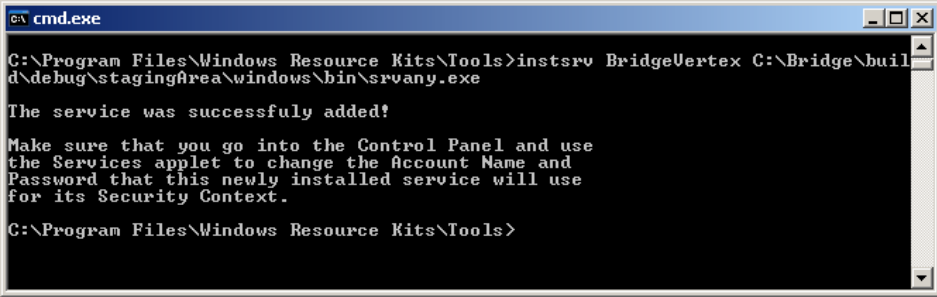
- 1 Open a command prompt (running `cmd.exe`) and change to the Microsoft Resource Kit directory that contains `instsrv.exe`; this directory is called `<resourceKitDir>`.

```
> cd C:\Program Files\Windows Resource Kits\Tools
```

- 2 Run `instsrv <myService> <installDir>\bin\srany.exe` to create a service to start the SSA Integration Pack.

```
> cd C:\Program Files\Windows Resource Kits\Tools
> instsrv BridgeVertex
C:\Bridge\build\debug\stagingArea\windows\bin\srany.exe
```

If the process is successful, a message appears similar to the following:



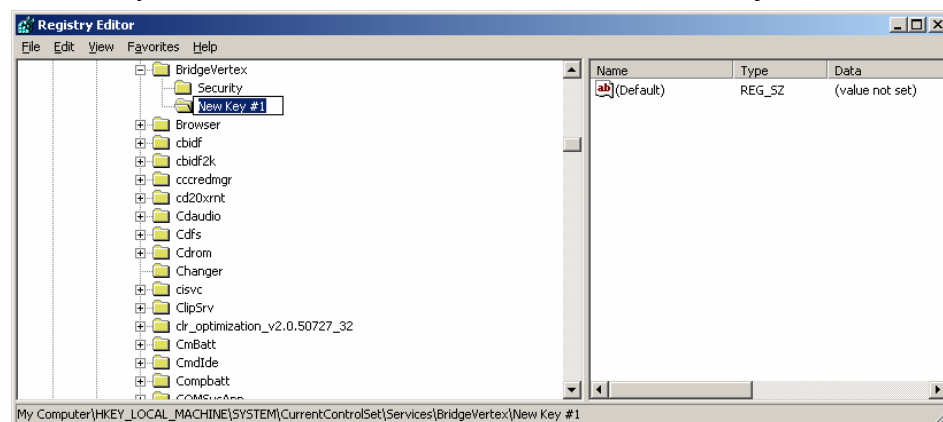
```
C:\Program Files\Windows Resource Kits\Tools>instsrv BridgeVertex C:\Bridge\build\debug\stagingArea\windows\bin\srany.exe

The service was successfully added!

Make sure that you go into the Control Panel and use
the Services applet to change the Account Name and
Password that this newly installed service will use
for its Security Context.

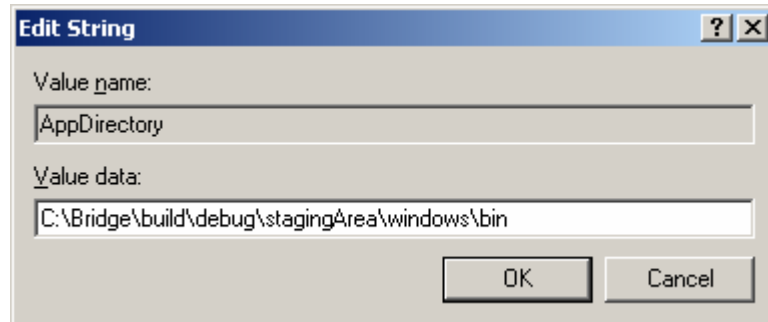
C:\Program Files\Windows Resource Kits\Tools>
```

- 3 Run `regedit` and go to the following directory:
My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
- 4 Select `<myService>` and, on the **Edit** menu, click **New → Key**.

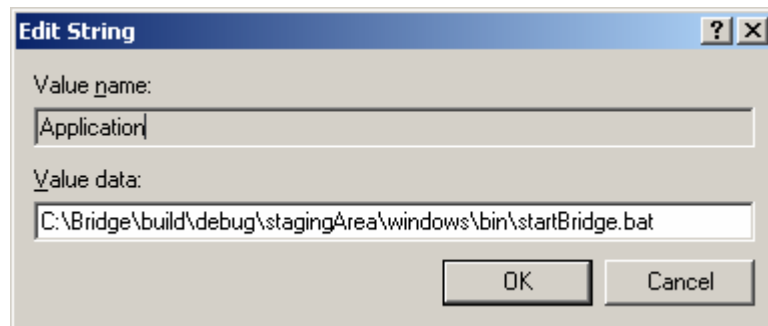


- 5 Change the name to **Parameters**, select **Parameters**, and add two new string keys:
 - AppDirectory
 - Application

- 6 Change the **Value Data** setting of AppDirectory to `<installDir>\bin`.

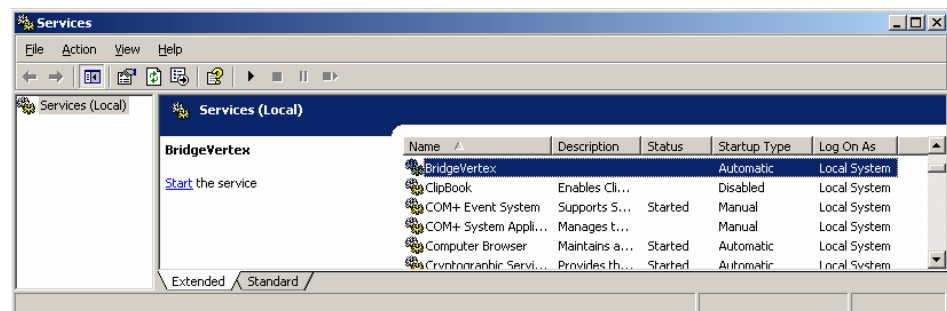


- 7 Change the **Value Data** setting of **Application** to `<installDir>\bin\startBridge.bat`.



Do not change any other changes to the settings

- 8 Close the the Registry Editor.
- 9 On the Control Panel, click **Administrative Tools** → **Services**:



- 10 Start the service and run the test scripts as described in the “Auxiliary Tools” chapter.

Troubleshooting

The standard configuration produces a log file in your home directory that records the main input/output events in the SSA Integration Pack and prints errors and warnings on the console. For more details, refer to the logging configuration section in the `configure.txt` document. Refer also to the SSA Open Architecture log files, both on the ERP machine and on the SSA Integration Pack machine, or to the vertex log files.

This chapter describes the process to configure the SSA Integration Pack. This chapter assumes that the SSA Integration Pack is already installed. For more specific details on how to install the SSA Integration Pack, refer to the “Installation” chapter.

General

All SSA Integration Pack-specific configuration files are available in the following directory:

- `<installDir>/etc`

where `<installDir>` is the SSA Integration Pack's installation directory.

As shipped, the files contain reasonable defaults. However, you will most likely need to edit the configuration to achieve a working setup. In particular, refer to “The BusComponent” and “The SSA Integration Pack,” later in this chapter.

Defaults

As shipped, the configuration files are suitable for a system described as follows:

- An installation of SSA Open Architecture 6.0 that uses an LDAP that includes a BusComponent called **vertex**, configured for your SSA Integration Pack installation to listen on. For example, a BusComponent with a URI of the form `socket ://<host>:9090`, where `<host>` is the name of the machine on which the SSA Integration Pack is installed.
- The **vertex** Web service running on the same machine as the SSA Integration Pack.
- The **vertex** Web service listening on port 8080 and installed in the default Web context, in other words, the location the **vertex** documentation suggests the Web service be installed.
- A user **ssa** defined in the **vertex** environment with password **ssa123** that has sufficient rights to perform tax calculations and update the tax register.

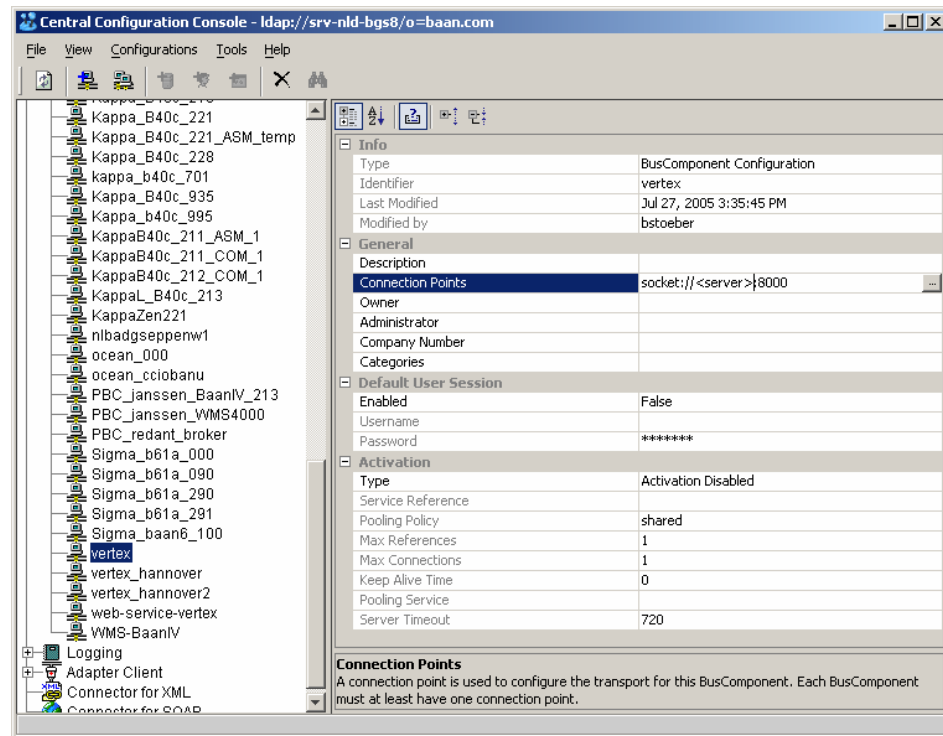
If your configuration fails to meet any of these conditions, you must alter the configuration files to fit.

The BusComponent

A relevant configuration file is: `<installDir>/etc/buscomponent.cfg`

The SSA Integration Pack needs a BusComponent in the LDAP that the SSA Open Architecture installation uses to listen on. SSA ERP can then send requests for the SSA Integration Pack to this BusComponent. By default, this BusComponent is named **vertex**. For information on how to create and edit BusComponents, refer to the SSA Open Architecture documentation. If you want the SSA Integration Pack to listen on a BusComponent with a different name, edit the `buscomponent.cfg` configuration file, which simply contains the name of the BusComponent to use. Be careful not to add any unnecessary additional characters in this file.

You can use the Central Configuration Console of the SSA Open Architecture Adapter Tools, as shown in the following figure, to retrieve the name of the BusComponent for the Vertex software:



The SSA Integration Pack

A relevant configuration file is: `<installDir>/etc/vertex.properties`.

To establish a connection to the Vertex Web service, several parameters are required, which you can configure in the `vertex.properties` configuration file. The file contains a list of key/value pairs, each pair on a separate line. The following lines are ignored:

- Lines that start with a pound sign (#)
- Lines that start with an exclamation mark (!)
- Empty lines

Example configuration lines do not begin with a pound sign (#). For details of the file format, such as how to include non-ASCII characters, refer to the documentation for the standard Java class `java.util.Properties`:

<http://java.sun.com>.

```
# Vertex tax provider configuration file
# Address where the Vertex LookupTaxAreas web service (document based) can
be reached
address.lookup.tax_areas = http://localhost:8080/vertex-
ws/services/LookupTaxAreasDoc

# Address where the Vertex LookupTaxAreas web service (document based) can
be reached
address.calculate.tax = http://localhost:8080/vertex-
ws/services/CalculateTaxDoc

# User name to use to log on to Vertex
user.name = ssa

# Password to use to log on to Vertex
password = ssal23
```

The recognized keys are the following:

- `address.lookup.tax_areas`
The HTTP URL that you use to look up tax areas
- `address.calculate.tax`
The HTTP URL that you use to calculate tax and update the tax register
- `user.name`
The user name that you use to connect to Vertex
- `password`
The password that you use to connect to Vertex

Logging

The following is a relevant configuration file:

`<installDir>/etc/logging.properties`

The SSA Integration Pack uses the standard Java logging facility. Various configuration options are available for this framework. For details, refer to the documentation that comes with the J2SE download at <http://java.sun.com>.

The default configuration sends errors and warnings to the console. A log file is also configured that contains additional information, such as details of all requests and responses.

The log file is configured to rotate every 1,000,000 bytes. By default the current log file (if only one SSA Integration Pack is running) is called **taxbridge0.log0** and can be found in the home directory of the user who is running the SSA Integration Pack. An example of how to alter the logging configuration is to maximize the amount of detail shown in the log file. To do this, change the following line:

```
.level= INFO
```

into

```
.level= FINEST
```

in the **logging.properties** configuration file.

If you cannot diagnose a problem using the log file described previously, you might need to examine the Open Architecture logging. For information on how to configure and use Open Architecture logging, refer to the Open Architecture documentation.

Other configuration files

The relevant configuration files are as follows:

- *<installDir>/etc/james.cfg*
This file lists all Java Classes required for the communication between ERP_{LN} and Vertex O Series.
- *<installDir>/etc/bridge.properties*
This file lists the component to be used for incoming calls.

You must not change these two files. These files are only of interest to the developers of the SSA Integration Pack. Changes to the contents of these files will cause the SSA Integration Pack to break down.

In addition to the startBridge script, the *<installDir>/bin* directory contains a number of other scripts, for example, to start tools that help you diagnose problems and verify your installation. This chapter describes each of tools in detail.

Parameters

Input parameters are passed to the tools on the command line in the form of key-value pairs. These command-line arguments look like the following:

- -D<key>=<value>

Example

Dcity=WASHINGTON

Many parameters have default values that make the tools easy to use for verification purposes. When you run a tool, the selected values for the various parameter keys are echoed to the console.

Most available parameters are not described here, but rather only the parameters minimally required to run the tools. For details on which parameters are available for a specific tool, run the tool with no arguments.

To run the test batches in a Windows environment, you must use a second command console.

vertexTest

The vertexTest script contacts the vertex Web service directly to look for a tax area ID. You can use this tool to check if the Web service is running. The tool checks the **City**, **State**, and **ZIP Code** parameters. You must set the **Backend** parameter to the address of the LookupTaxAreasDoc vertex Web service, for example:

```
-Dbackend=http://vertex.my.domain.com:8080/vertex-  
ws/services/LookupTaxAreasDoc
```

By default, the **Backend** parameter is set to:

```
http://localhost:8080/vertex-ws/services/LookupTaxAreasDoc
```

getGeocode

This script does the same thing as the vertexTest script, but goes through the SSA Integration Pack. You can use this tool to check if both the SSA Integration Pack and the vertex Web service are running.

Set the **bc** parameter to the BusComponent to contact the SSA Integration Pack. By default, the **bc** parameter is set to **vertex**.

calculateTax

This script sends a request to calculate tax for a sales invoice to the vertex Web service through the SSA Integration Pack. The script involves a large number of parameters.

You must set the **bc** parameter to the BusComponent to contact the SSA Integration Pack. By default, the **bc** parameter is set to **vertex**.

quoteTax

This script sends a request to quote tax for a sales invoice to the vertex Web service through the SSA Integration Pack. This process takes a large amount of parameters.

You must set the **bc** parameter to the BusComponent to contact the SSA Integration Pack. By default, the **bc** parameter is set to **vertex**.

purchaseOrderTPP

This script sends a request to quote tax for a purchase order to the vertex Web service through the SSA Integration Pack. This process takes a large amount of parameters.

You must set the **bc** parameter to the BusComponent to contact the SSA Integration Pack. By default, the **bc** parameter is set to **vertex**.

TaxOnlyAdjustmentSales

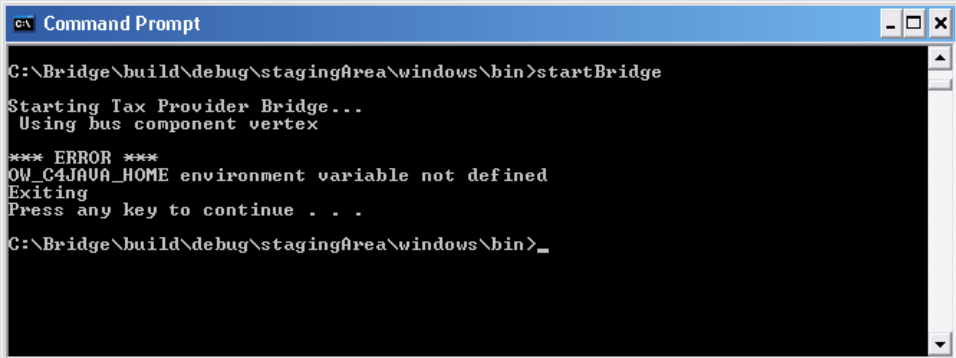
This script performs a tax-only adjustment for a sales invoice to the vertex Web service through the SSA Integration Pack. This process requires a large number of parameters.

You must set the **bc** parameter to the BusComponent to contact the SSA Integration Pack. By default, the **bc** parameter is set to **vertex**.

This chapter contains examples, which problems could occur and how to solve them.

OW_C4JAVA_HOME not defined

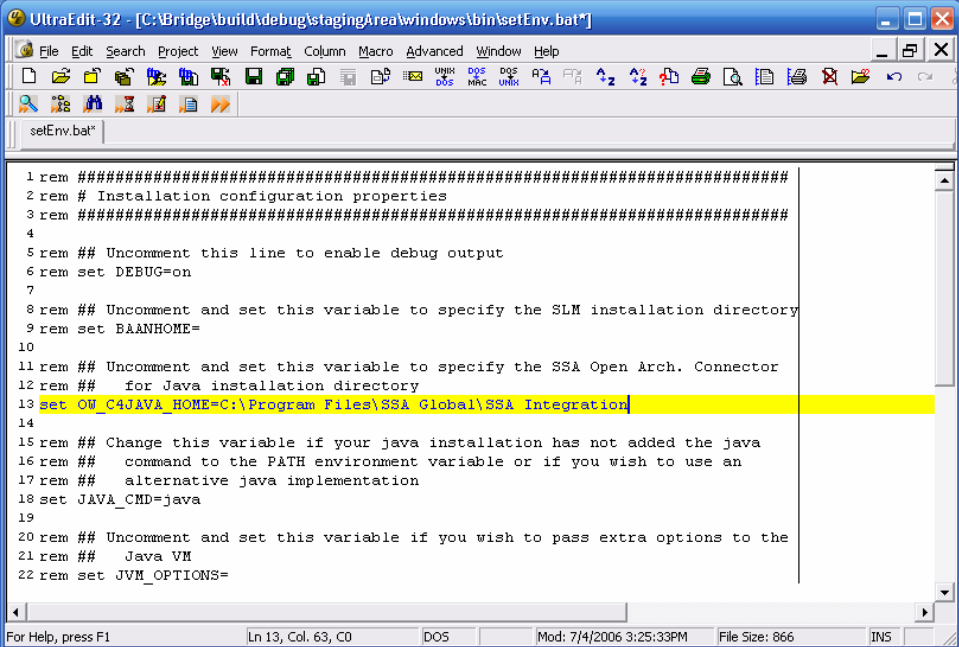
When starting the bridge the following error message may appear:

A screenshot of a Windows Command Prompt window. The title bar reads "Command Prompt". The command prompt shows the following text:

```
C:\Bridge\build\debug\stagingArea\windows\bin>startBridge
Starting Tax Provider Bridge...
Using bus component vertex

*** ERROR ***
OW_C4JAVA_HOME environment variable not defined
Exiting
Press any key to continue . . .
C:\Bridge\build\debug\stagingArea\windows\bin>_
```

Edit the file `setenv.bat` (located in subdirectory `/bin` of the SSA Integration Pack installation directory) and set the variable `OW_C4JAVA_HOME` to specify the directory where your SSA Open Architecture Connector for Java has been installed.



```
1 rem #####
2 rem # Installation configuration properties
3 rem #####
4
5 rem ## Uncomment this line to enable debug output
6 rem set DEBUG=on
7
8 rem ## Uncomment and set this variable to specify the SLM installation directory
9 rem set BAANHOME=
10
11 rem ## Uncomment and set this variable to specify the SSA Open Arch. Connector
12 rem ## for Java installation directory
13 set OW_C4JAVA_HOME=C:\Program Files\SSA Global\SSA Integration
14
15 rem ## Change this variable if your java installation has not added the java
16 rem ## command to the PATH environment variable or if you wish to use an
17 rem ## alternative java implementation
18 set JAVA_CMD=java
19
20 rem ## Uncomment and set this variable if you wish to pass extra options to the
21 rem ## Java VM
22 rem set JVM_OPTIONS=
```

Unknown bus component

When starting the bridge the following error message may appear:

“Can not start bus component ‘vertex’ by James”

```

C:\Bridge\build\debug\stagingArea\windows\bin>startBridge

Starting Tax Provider Bridge...
Using bus component vertex
CLASSPATH=C:\Bridge\build\debug\stagingArea\windows\bin\..\etc;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\ow-tpbridge.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxb-api.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxb-impl.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxb-libs.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jax-gname.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\namespace.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\relaxngDatatype.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\saa-j-api.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxp-api.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\dom.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\sax.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\xalan.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\xercesImpl.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\activation.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxrpc-api.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxrpc-impl.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\jaxrpc-spi.jar;C:\Bridge\build\debug\stagingArea\windows\bin\..\lib\mail.jar;C:\Program Files\Baan\shared\java\com_baan_bclm.jar;C:\Program Files\SSA Global\SSA Integration\java\ow.jar;C:\Program Files\SSA Global\SSA Integration\java\owxml.jar;C:\Program Files\SSA Global\SSA Integration\java\erprt.jar;C:\Program Files\SSA Global\SSA Integration\java\Path="C:\Program Files\Baan\shared\bin;C:\Program Files\Baan\shared\shlib;C:\Program Files\Windows Resource Kits\Tools;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\Program Files\ATI Technologies\ATI Control Panel;C:\Program Files\UltraEdit;C:\Program Files\scopus32\sybase\ddl;C:\Program Files\ManageSoft\Common;C:\Program Files\Baan\shared\bin;C:\Program Files\Baan\shared\shlib;C:\Program Files\Rational\ClearCase\bin;C:\Program Files\Rational\common;C:\Program Files\Microsoft SQL Server\80\Tools\BINN;C:\Program Files\Java\jre1.4.2_08\bin\client;C:\j2sdk1.4.2_08\bin;C:\Program Files\SSA Global\SSA Integration\bin;C:\Program Files\Java\j2re1.4.2_08\bin\client
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.UBE;.JS;.JSE;.WSF;.WSH
Environment variable JUM_OPTIONS not defined

Executing:
"java" "-Dcom.ssa.ta.taxprovider.install.dir=C:\Bridge\build\debug\stagingArea\windows\bin\.." "-Djava.util.logging.config.file=C:\Bridge\build\debug\stagingArea\windows\bin\..\etc\logging.properties" com.eibus.java.james.James vertex

James - Server for Java
(c) Copyright 2006 SSA Global Technologies, Inc. and its Subsidiaries and Affiliates

Exception: java.lang.Exception: Can not start bus component 'vertex' by James
C:\Bridge\build\debug\stagingArea\windows\bin>

```

Edit the configuration file buscomponent.cfg in the /etc subdirectory where the SSA Integration Pack has been installed and insert the name of the buscomponent, the SSA Integration Pack shall listen on, for example:
vertex_bstoeber

