

# **webMethods EntireX**

## **EntireX Administration under z/VSE**

Version 10.8

October 2022

This document applies to webMethods EntireX Version 10.8 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 1997-2022 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

**Document ID: EXX-ADMIN-108-20220601VSE**

## Table of Contents

1 About this Documentation .....	1
Document Conventions .....	2
Online Information and Support .....	2
Data Protection .....	3
2 EntireX Administration under z/VSE .....	5
Available Stubs .....	6
Transport Methods for Broker Stubs .....	6
Using the Batch Stub Interface Module BKIMB .....	8
Using the CICS Stub Interface Module BKIMC .....	9
Tracing for Broker Stubs .....	10



# 1 About this Documentation

---

- Document Conventions ..... 2
- Online Information and Support ..... 2
- Data Protection ..... 3

## Document Conventions

---

Convention	Description
<b>Bold</b>	Identifies elements on a screen.
Monospace font	Identifies service names and locations in the format <code>folder.subfolder.service</code> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies:  Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies:  Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the   symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

## Online Information and Support

---

### Product Documentation

You can find the product documentation on our documentation website at <https://documentation.softwareag.com>.

In addition, you can also access the cloud product documentation via <https://www.software-ag.cloud>. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

### Product Training

You can find helpful product training material on our Learning Portal at <https://knowledge.softwareag.com>.

## Tech Community

You can collaborate with Software AG experts on our Tech Community website at <https://tech-community.softwareag.com>. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at <https://github.com/softwareag> and <https://hub.docker.com/publishers/softwareag> and discover additional Software AG resources.

## Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at <https://empower.softwareag.com>. Many services on this portal require that you have an account. If you do not yet have one, you can request it at <https://empower.softwareag.com/register>. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

## Data Protection

---

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

---



## 2 EntireX Administration under z/VSE

---

▪ Available Stubs .....	6
▪ Transport Methods for Broker Stubs .....	6
▪ Using the Batch Stub Interface Module BKIMB .....	8
▪ Using the CICS Stub Interface Module BKIMC .....	9
▪ Tracing for Broker Stubs .....	10



**Note:** Support for EntireX Broker and RPC servers under z/VSE ends on 31 October 2020 for all EntireX versions (including version 9.6, the last released version on z/VSE). See *Features Dropped in Version 10.7*.

## Available Stubs

This table lists all Broker stubs available under the z/VSE operating system that are to be used with the programming languages Natural, COBOL, Assembler and C. The stub you choose depends on your environment (CICS or Batch).

Environment	Transport			Trace	Stub Module
	NET	TCP	SSL		
All environments that use Batch	Yes	Yes	<sup>(1)</sup>	Yes <sup>(2)</sup>	<a href="#">BKIMB</a>
All environments that use CICS	Yes	Yes	<sup>(1)</sup>	Yes <sup>(2)</sup>	<a href="#">BKIMC</a>



### Notes:

1. Use BSI's Automatic Transport Layer Security (ATLS). Refer to the *BSI SSL Installation, Programming and User's Guide* for more information. See also *Using SSL/TLS with EntireX Components*.
2. The request needs to use TCP transport method. Tracing is not available with NET transport.

## Transport Methods for Broker Stubs

- [Transport Method Values](#)
- [Setting the Timeout for the Transport Method](#)
- [Limiting the TCP/IP Connection Lifetime](#)

### Transport Method Values

Transport Value	Tips
TCP	<ul style="list-style-type: none"> <li>■ Provides remote machine and cross-platform communication.</li> </ul>
NET	<ul style="list-style-type: none"> <li>■ Provides the best performing transport if the application and Broker kernel reside on the same machine.</li> <li>■ Provides for remote communications if Entire Net-Work is also installed on the application and Broker kernel machines.</li> <li>■ Requires the installation of Adabas components. We recommend installing the Adabas modules delivered with EntireX installation kit. See your Adabas documentation for more information on installing the Adabas SVC.</li> </ul>

Transport Value	Tips
	<ul style="list-style-type: none"> <li>■ Tracing is not supported.</li> </ul>

For Secure Sockets Layer/Transport Layer Security (SSL/TLS) as transport method, see table *Using SSL/TLS with EntireX Components*.

To use the stubs' internal security functionality, API version 8 or higher needs to be used by the application. (e.g., EntireX RPC Server, NAT42). The delivered phases are linked for use with internal security.

## Setting the Timeout for the Transport Method

### Introduction

If the transport layer is interrupted, communication between the broker and the stub - that is, client or server application - is no longer possible. A client or server might possibly wait infinitely for a broker reply or message in such a situation. To prevent this and return control to your calling application in such a situation, set a timeout value for the transport method.

The timeout settings for transport layers are independent of the timeout settings of the broker.

Setting the timeout for the transport layer is possible for the transport method TCP, and is supported by all broker stubs under z/VSE.

### Transport Timeout Values

The timeout value for the transport method is set by the environment variable `ETB_TIMEOUT` on the stub side. This transport timeout is used together with the broker timeout - which is set by the application in the `WAIT` field of the broker ACI control block - to calculate the actual value for the transport layer's timeout. The following table describes the possible values for the transport timeout:

Transport Timeout Value	Description
0	Infinite wait for the application.
<i>n</i>	The transport method additionally waits this time in seconds. A negative value is treated as <code>TIMEOUT=0</code> (infinite wait for the application).
nothing set	Transport method waits additional 20 seconds.

The actual timeout for transport layer equals broker timeout (`WAIT` field) + timeout value for transport method.

## Limiting the TCP/IP Connection Lifetime

With transport method TCP/IP, the broker stub establishes one or more TCP/IP connections to the brokers specified with `BROKER-ID`. These connections can be controlled by the transport-specific `CONNECTION-NONACT` attribute on the broker side, but also by the transport-specific environment variable `ETB_NONACT` on the stub side. If `ETB_NONACT` is not 0, it defines the non-activity time (in seconds) of active TCP/IP connections to any broker. See `ETB_NONACT` under *Environment Variables in EntireX*. Whenever the broker stub is called, it checks for the elapsed non-activity time and closes connections with a non-activity time greater than the value defined with `ETB_NONACT`. Stubs capable of running in SRB mode do not support `ETB_NONACT` handling.

Transport Non-activity Value	Description
0	Infinite lifetime until application is stopped.
<i>n</i> (seconds)	Transport connections with non-activity time greater than <i>n</i> will be closed.
Nothing set	Infinite lifetime until application is stopped.

## Using the Batch Stub Interface Module BKIMB

You can use BKIMB for all batch environments. This stub interface module is delivered as a phase, which can be loaded by your application dynamically, and as an object for linking. At runtime, the EXX960 library and the WAL842 library need to be included into the LIBDEF search chain. If you need to statically link your application with the interface object, include the following objects:

```
PHASE <app1_phase_name>,*
INCLUDE <app_obj>
INCLUDE BKIMB
INCLUDE ETBVPRE
INCLUDE ETBVEVA
INCLUDE ETBENC
INCLUDE ETBTB
ENTRY <app_entry>
```

### ➤ To set up a secure environment

- 1 Statically link your application with the following interface objects:

```

PHASE <appl_phase_name>,*
INCLUDE <app_obj>
INCLUDE BKIMB
INCLUDE ETBUPRE
INCLUDE ETBUEVA
INCLUDE ETBVPRE
INCLUDE ETBVEVA
INCLUDE ETBENC
INCLUDE ETBTB
ENTRY <app_entry>

```

Or:

If BKIMB is to be loaded dynamically, you can relink the phase for use with security. Refer to the delivered job control example BKIMB.J.

- 2 Rename phase SECUEXI0 in library EXX960 to SECUEXIT.

## Using the CICS Stub Interface Module BKIMC

You can use BKIMC for all CICS environments. This stub interface module is delivered as a phase, which can be loaded by your application dynamically, and as an object for linking. To enable CICS to find the various programs, include the EXX960 sublibrary in the DFHRPL chain and add following definition to your CICS environment:

```

DEFINE PROGRAM(BKIMC) GROUP(EXX) LANGUAGE(ASSEMBLER) (only required if not linked ←
to your application)
DEFINE PROGRAM(BROKERC) GROUP(EXX) LANGUAGE(C)

```

If you need to statically link your application with the interface object, include the following objects:

```

PHASE <appl_phase_name>,*
INCLUDE <app_obj>
INCLUDE BKIMC
INCLUDE ETBVPRE
INCLUDE ETBVEVA
INCLUDE ETBENC
INCLUDE ETBTB
ENTRY <app_entry>

```

### ➤ To set up a secure environment

- 1 Statically link your application with the following interface objects:

```
PHASE <appl_phase_name>,*  
INCLUDE <app_obj>  
INCLUDE BKIMC  
INCLUDE ETBUPRE  
INCLUDE ETBUEVA  
INCLUDE ETBVPRE  
INCLUDE ETBVEVA  
INCLUDE ETBENC  
INCLUDE ETBTB  
ENTRY <app_entry>
```

Or:

If BKIMC is to be loaded dynamically, you can relink the phase for use with security. Refer to the delivered job control example BKIMC.J.

- 2 Rename phase SECUExI0 in library EXX960 to SECUExIT.

## Tracing for Broker Stubs

---

If transport method TCP is used, a stub trace may be turned on for diagnostic purposes. Set up the following environment variable in your application job control or CICS startup.

```
//SETPARM STUBLOG=2
```