Installing the EntireX Broker under BS2000/OSD

This section explains how to install and start the EntireX Broker on BS2000/OSD. It covers the following topics:

- Introduction
- Overview of Broker Installation Steps
- Verifying the Installation of the Broker

Introduction

When installing EntireX Broker, all modifications are done to the J and S elements - job control (J-elements) and parameter files (S-elements) - located in EXX811.JOBS. All job control and parameter elements contain a preconfiguration which enables you to install EntireX Broker with much less effort. Using this preconfiguration requires that all libraries be located under the same BS2000 user ID.

Note:
Installation prerequisites are described centrally. See *BS2000/OSD Prerequisites*. Make sure these are met before you start installation. It is important to upgrade your libraries first.

Overview of Broker Installation Steps

This section describes the following installation steps

- Install the License Certificate
- Customize the EntireX Broker Attribute File
- Customize the ADALNK Parameters File
- Customize the EntireX Broker Startup JCL and Start EntireX Broker
- Start EntireX Broker
- Stop EntireX Broker
- Creating a Broker Persistent Store (optional)
- Set up the EntireX Broker Security Server for BS2000/OSD (optional)
- Load the SYSETB INPL and ERRN Files (optional)

Install the License Certificate

There are two types of license file:

- One is delivered on installation medium (EXX811.LICS)
- The other, in ASCII format, you may have received by e-mail (EXX811.XML) or on a CD. To make this file available for EntireX, transfer it in binary format to BS2000/OSD, using FTP. Make sure that the target file on BS2000/OSD is allocated with `FILE-STRUCTURE = SAM` and `BUF-LEN = STD(2)`.

  When uploading the license file to BS2000/OSD, you can use the following FTP commands to create the required file structure:

  ```
  LITERAL FILE EXX811.LICS, RECSIZE=0, RECFORM=V, BLKSIZE=(STD,2), OPEN=UPDATE, FCBTYPE=SAM
  BIN
  PUT EXX811.XML EXX811.LICS
  ```

Customize the EntireX Broker Attribute File

ETB-ATTR is a sample broker attribute file. Customize the attribute settings to suit your needs.

To run a minimal configuration of EntireX Broker that is suitable to execute the verification programs BCOC and BCOS, set up the following parameters:

Parameter	Description
BROKER-ID= ETB<nnnnn>	Identifies the Broker to which the attribute file applies. The Broker ID must be unique per machine
NODE= <node-id>	A DBID under which EntireX Broker is visible in the system
IDTNAME= ADA<xxxxx>	Specifies the ID table name under which EntireX Broker will be accessible
PORT= <port>	A free port number EntireX Broker listens on for TCP/IP communication. Depending on the system settings, free port numbers under BS2000/OSD start with port numbers greater than 4096

For a full description of all parameters, see *Broker Attributes*.

Customize the ADALNK Parameters File

The verification programs `BCOS` and `BCOC`, the command and information services utilities `ETBCMD` and `ETBINFO`, the publish and subscribe examples `PUB` and `SUB` as well as the Adabas persistent store require the ADALNK parameter `IDTNAME` to be set. See also *Verifying the Installation of the Broker* and step *Creating a Broker Persistent Store (optional)* below.

Parameter	Description
ADALNK IDTNAME= ADA<xxxxx>	Specifies the ID table name under which EntireX Broker is accessible.

Customize the EntireX Broker Startup JCL and Start EntireX Broker

The job control delivered with EntireX Broker uses BS2000/OSD S-procedures. We strongly recommend you do not modify `START-BROKER`. This procedure is recursively called to establish the broker environment. Modifications should only be done to the `PARAMETER-DECLARATION` section as described below. If all EntireX Broker components are installed under the same BS2000/OSD user ID, and all previous installation steps have been done using the default settings, no modifications to `START-BROKER` are necessary.

Startup Parameter	Description	Default
LICENSE-FILE	License certificate file.	EXX811.LICS
EXX-LIB	EntireX Broker library.	EXX811.LIB
EXX-JOBS	EntireX Broker jobs library.	EXX811.JOBS
WAL-MOD	WAL module library.	WAL826.MOD
MLC-MOD	Licensing module library.	MLC*vrs*.MOD
BROKER-ATTRIBUTES	EntireX Broker attribute file.	ETB-ATTR
BROKER-VARIABLES	EntireX Broker attributes variables file.	ETB-VARS
ADABAS-PARAMETERS	Adabas ADALNK parameter file.	ETB-ADAPARM
FILE-PREFIX	File name prefix used for all files written to disk.	ETB
CONFIG-REPORT-FILE	EntireX Broker configuration report file name. *file-prefix.tsn*.CONFIG.REPORT	CONFIG.REPORT
PSTORE-REPORT-FILE	EntireX Broker Adabas persistent store report file name.*file-prefix.tsn*.PSTORE.REPORT	PSTORE.REPORT
STORAGE-REPORT-FILE	EntireX Broker storage report file name. *file-prefix.tsn*.STORAGE.REPORT	STORAGE.REPORT
LICENSE-REPORT-FILE	EntireX Broker license report file name. *file-prefix.tsn*.LICENSE.REPORT	LICENSE.REPORT
ACCOUNTING-FILE	EntireX Broker accounting file name. *file-prefix.tsn*.ACCOUNTING	ACCOUNTING
CLOGR1-FILE	EntireX Broker TRACE-LEVEL=1 command log file 1. *file-prefix.tsn*.CLOGR1	CLOGR1
CLOGR2-FILE	EntireX Broker TRACE-LEVEL=1 command log file 2. *file-prefix.tsn*.CLOGR2	CLOGR2
TASK-TYPE	Is filled during runtime with the EntireX Broker task type. Do *not* modify it !	MAIN
MAIN-TASK-TSN	Is filled during runtime with the main task TSN. The value is used by subsequently entered Broker tasks. Do *not* modify it!	*ETB

If EntireX Broker requires `SERVICE-UPDATES` to be set to YES, the attributes file ETB-ATTR cannot be held in LMS. It needs to be located on disk instead. This requires a change the `START-BROKER` procedure. See `SERVICE-UPDATES`.

Copy the attribute file to disk.

```
/ START-LMS
//  EXTRACT-ELEMENT ELEMENT=*LIB(LIB=EXXvrs.JOBS,ELEMENT=ETB-ATTR,TYPE=S)
//END

Change the ADD-FILE-LINK  in the section below in START-BROKER in EXXvrs.JOBS to read the file ETB-ATTR from disk.

…
/  "-----------------------------------------------------------------"
/  "  SET UP FILE LINKS                                               "
/  "-----------------------------------------------------------------"
/   ADD-FILE-LINK LINK-NAME = ETBLIC,   FILE-NAME = &(LICENSE-FILE)
/&* ADD-FILE-LINK LINK-NAME = ETBFILE,  FILE-NAME = #BROKER-ATTRIBUTES
/   ADD-FILE-LINK LINK-NAME = ETBFILE,  FILE-NAME = &(BROKER-ATTRIBUTES)
/   ADD-FILE-LINK LINK-NAME = ETBVARS,  FILE-NAME = #BROKER-VARIABLES
/   ADD-FILE-LINK LINK-NAME = DDLNKPAR, FILE-NAME = #ADABAS-PARAMETERS
...
```

Start EntireX Broker

To start the broker

- Enter the following SDF command:

  ```
  /ENTER-PROCEDURE *LIB(LIB=EXX811.JOBS,ELE=START-BROKER), -
  /JOB-NAME=ETB,LOGGING=*NO,RESOURCES=*PAR(CPU-LIMIT=*NO)
  ```

  We recommend using a three-character job name. The job name is taken as prefix for all subsequently started tasks. Because the job name is limited to eight characters, a longer job name will overwrite the suffix added by EntireX Broker. For example: EntireX Broker running with three worker tasks and NET-TCP communication, `JOB-NAME=ETB`, `CPU-LIMIT=*NO`:

  ```
  NAME       TSN TYPE       PRI      CPU-USED CPU-MAX ACCOUNT#
  ETB       5397 2 BATCH    9 255      2.2379  NTL    1
  ETBCOM    5398 2 BATCH    9 255      1.3577  NTL    1
  ETBWRK00  5399 2 BATCH    9 255      0.8970  NTL    1
  ETBWRK01  5400 2 BATCH    9 255      0.7571  NTL    1
  ETBWRK02  5401 2 BATCH    9 255      0.7445  NTL    1
  ETBTCP00  5402 2 BATCH    9 255      0.6124  NTL    1
  ETBTCPPX  5403 2 BATCH    9 255      0.5417  NTL    1
  ETBNET00  5404 2 BATCH    9 255      0.6555  NTL    1
  ETBTOM    5407 2 BATCH    9 255      6.4044  NTL    1
  ```

  The properties assigned to the main task (ETB), e.g. `JOB-CLASS`, `CPU-LIMIT`, will be inherited by all subsequently started tasks. For `CPU-LIMIT`, if specified, only `*NO` (no time limit) and `*STD` are inherited.

You can now configure and run the verification jobs `BCOS` and `BCOC`. See *Verifying the Installation of the Broker*.

Stop EntireX Broker

To stop the broker from a privileged user ID

- Enter the following command:

  ```
  /INFORM-PROGRAM MSG='ETBSTOP',JOB-IDENTIFICATION=*TSN(TSN=tsn)
  ```

  where *tsn* is the task number associated with the broker main task (in the example above the TSN of job name ETB)

All other tasks that were created as a result of starting the broker will be stopped automatically.

To stop the broker from an operator console

- Enter the following command:

```
/INTR tsn,ETBSTOP
```

where *tsn* is the task number associated with the broker main task (in the example above the TSN of job name ETB)

All other tasks that were created as a result of starting the broker will be stopped automatically.

To stop the broker from a non-privileged user ID

- Use the S-procedure `STOP-BROKER` in EXX811.JOBS

Startup Parameter	Description	Default

BROKER-ID	Depending on the communication method, the BrokerId can be specified in two different formats:
	● **TCP Transport Method**
	`ip:port:TCP`
	where `ip` is the address or DNS host name,
	`port` is the port number that EntireX Broker is listening on, and
	`TCP` is the protocol name
	● **NET Transport Method**
	`ETBnnn:SVCmmm:NET`
	where `nnn` is the ID under which EntireX Broker is connected to the Adabas ID table,
	`mmm` is the SVC number under which the Adabas ID table can be accessed, and
	`NET` is the protocol name
		none

ADABAS-PARAMETERS	Adabas parameters used for NET communication method.	ETB-ADAPARM
USERID	If EntireX Broker is running with EntireX Security, a user ID needs to be supplied.	none
PASSWORD	If EntireX Broker is running with EntireX Security, a password needs to be supplied.	none
EXX-LIB	EntireX Broker module library.	EXX811.LIB
EXX-JOBS	EntireX Broker jobs library.	EXX811.JOBS
WAL-MOD	WAL module library.	WAL826.MOD

Set the broker ID in the `PARAMETER-DECLARATION` section and enter following command:

```
/CALL-PROCEDURE (EXX811.JOBS, STOP-BROKER)
```

Creating a Broker Persistent Store (optional)

This step may be skipped if no persistent store is required. See *Managing the Broker Persistent Store* for more information. The persistent store resides in an Adabas database, which means that Adabas must be installed. See *BS2000/OSD Prerequisites*.

Customize job `CREATE-PSTORE-ADABAS`. Add the appropriate values in the `PARAMETER-DECLARATION` section and run the job.

```
/CALL-PROCEDURE (LIB=EXX811.JOBS,ELE=CREATE-PSTORE-ADABAS)
```

This Adabas `ADALOD` job loads the Adabas persistent store file FDT from EXX811.SYSF into the database where the Adabas persistent store is to reside.

Set the relevant attributes in the broker attribute file ETB-ATTR. See *Adabas-specific Attributes* and *Managing the Broker Persistent Store*.

Note:
The Adabas persistent store requires the ADALNK parameter `IDTNAME` in `ETB-ADAPARM` to be set properly. This must be the same ID table as used by the broker. See *Customize the EntireX Broker Attribute File* and *Customize the ADALNK Parameters File*.

Set up the EntireX Broker Security Server for BS2000/OSD (optional)

To activate authentication

- Set the following two parameters in the broker attribute file to switch on security:
  - **In the DEFAULTS=BROKER section**

    ```
    SECURITY=YES
    ```

  - **In the DEFAULTS=SECURITY section**

    ```
    ACCESS-SECURITY-SERVER=YES
    ```

  The Broker Security Server requires administrator rights and must be run under a privileged user ID.

  Note:
  If the parameter `ACCESS-SECURITY-SERVER` is set to "NO", the broker itself must be run under a privileged user ID to allow authentication. In this case, the Broker Security Server is not needed.

To start the Broker Security Server

1. Set up the correct broker library within `START-SECURITY-SERVER`, because the server task does not usually run under same user ID where the module library resides.
2. Issue the following command from a privileged user ID (TSOS) to run the server:

```
/ENTER-PROCEDURE *LIB(LIB=$kkk.EXXnnn.JOBS,     -
/                 ELE=START-SECURITY-SERVER),   -
/                 JOB-NAME=SECUSERV,LOG=*NO
```

where `$kkk` is the user ID under which the broker library resides.

To stop the Broker Security Server from a privileged user ID

- Enter:

```
/INFORM-PROGRAM MSG='EOJ',JOB-IDENTIFICATION=*TSN(TSN=tsn)
```

where <*tsn*> is the BS2000/OSD task number associated with the server.

To stop the Broker Security Server from an operator console

- Enter:

```
/INTR tsn,EOJ
```

where *tsn* is the BS2000/OSD task number associated with the server.

To stop the Broker Security Server from a non-privileged user ID

- Enter the following SDF command:

```
/CALL-PROCEDURE (EXX811.JOBS, STOP-SECURITY-SERVER)
```

Note:
This works from all user IDs in the system.

Load the SYSETB INPL and ERRN Files (optional)

(SMA Job I061, Step 7600 and 7602)

Perform this step only if you want to install the sample Natural programs.

Use the Natural system command INPL and ERRLODUS (see Natural User's Guide) to load the EntireX Broker system objects (EXB811.INPL and EXB811.ERRN).

This loads the following library:

Library	File	Contents
SYSETB	FNAT	Sample Natural programs.

Set the Natural profile parameter ESIZE=40.

Invoke Natural, logon to library SYSETB and edit the member PARM on library SYSETB to set parameters as required at your site, especially the `BROKER-ID`.

Parameter members can also be made user-dependent. Copy the PARM member and save it in a member with a user ID as name (Natural variable *USER). If a user logs on with this user ID, these parameters take effect instead of the PARM member.

Verifying the Installation of the Broker

Once you have installed the EntireX Broker you can verify that the installation was successful by modifying and using the following client/server pair with the Broker:

- BCOS
- BCOC

These procedures are located in library EXX811.JOBS. They, in turn, execute the BCOC (client) and BCOS (server) test programs, depending on a parameter value. These programs support communication with the Broker using either TCP/IP communication method, or the conventional XCOM (Adabas router) communication method. The communication protocol to be used depends on the format of the Broker ID used.

Both BCOS and BCOC are also delivered as example sources. BCOS and BCOC act as a pair, where BCOS provides a service that BCOC calls.

Customize the verification jobs before you run them. BCOS and BCOC require a `BROKER-ID` to be set up in the job control's PARAMETER-DECLARATION section. Since they may access an Adabas ID table (communication method XCOM), an IDTNAME must be provided. This IDTNAME is read from the ETB-ADAPARM element (see *Customize the ADALNK Parameters File*).

Start BCOS using following command.

```
/CALL-PROCEDURE (LIB=EXX811.JOBS,ELE=BCOS)
```

Wait until BCOS has registered the service.

Then start BCOC in a different session. Since BCOS sets a WAIT value of 60 seconds, it will terminate with a "WAIT timeout" if BCOC is not started within this time frame. See `WAIT` under *Broker ACI Fields*.

```
/CALL-PROCEDURE (LIB=EXX811.JOBS,ELE=BCOC)
```

Both should now run through 10 iterations of exchanging messages.

If the initialization messages appear but the verification program fails, check for either a TCP port conflict or a problem with the ID table name or version.

The client and server programs that are executed by these procedures need the following to be defined in the Broker attribute file (if the delivered attribute file is used, no changes are needed).

```
CLASS = BCLASS, SERVER = BSERVER, SERVICE = BSERVICE
```

See *Broker Attributes* for more information.

For further information on the usage of BCOC / BCOS please also see the comments in the delivered job control. For example, by setting the command string to: /COMMAND = ’-h’ detailed information on the usage will be printed.