

LIBNDRXXAJDBC(8)

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	SYNOPSIS	1
2	DESCRIPTION	2
3	BUGS	4
4	SEE ALSO	5
5	COPYING	6

Chapter 1

SYNOPSIS

libndrxxajdbc.so libndrxxajdbc.dylib

Chapter 2

DESCRIPTION

This library shall be used as driver for loading Enduro/X JDBC XA Switch for Enduro/X. This library is set in **NDRX_XA_DRIVERLIB** environment variable. And resource manager library is set as **libexjava(8)** in **NDRX_XA_RMLIB**.

For running java processes, when they are performing XA init, the existing JVM and **libexjava** instance is used.

When driver is being loaded from **tmsrv(8)** which is pure C process, then JVM instance is created, **libexjava** is loaded and XA Switch is acquired.

The **NDRX_XA_OPEN_STR** must be configured in following format:

```
{ "class": "<JDBC Driver Class Name>",
  "set": {
    "<Set Method Of Class Object 1>": "<Value to bet set 1>"
    , "<Set Method Of Class Object 2>": "<Value to bet set 2>"
    , "<Set Method Of Class Object N>": "<Value to bet set N>"
    , "<Set Method of Properties 1>": {
      "<Property 1 Setting 1>": "<Value to bet set 1/1>"
      , "<Property 1 Setting N>": "<Value to bet set 1/N>"
    }
  }
}
```

For example:

```
; Postgresql DB XA Connection, JDBC driver
[@global/DB1_JDBC]
NDRX_XA_RES_ID=1
NDRX_XA_OPEN_STR="{ "class": "org.postgresql.xa.PGXADatasource",
  "set": {
    "setUser": "${EX_PG_USER}"
    , "setPassword": "${EX_PG_PASS}"
    , "setURL": "jdbc:postgresql://${EX_PG_HOST}:${EX_PG_PORT}/${EX_PG_DB}"
  }
}"
NDRX_XA_CLOSE_STR=${NDRX_XA_OPEN_STR}
NDRX_XA_DRIVERLIB=${NDRX_APPHOME}/../../xdrv/libndrxxajdbc.so
NDRX_XA_RMLIB=${NDRX_APPHOME}/../../libsrc/c/libexjava.so
NDRX_XA_LAZY_INIT=1
NDRX_XA_FLAGS=NOJOIN;NOSTARTXID
```

Thus in Open string class configuration, and set fields must be configured. The JSON is used when **XAResource** is configured.

For more details see [module_developer_guide\(guides\)\(Enduro/X Java Internal Developer Guide, Distributed transaction processing architecture chapter\)](#)

NOTE: the corresponding class of the JDBC driver must be accessible for Java class loader. Either it is compiled with **exjld(8)** within the Java binary, or provided by CLASSPATH variable for **tmsrv(8)** and/or Java process. For classical java process "-cp" command line argument applies too.

Chapter 3

BUGS

Report bugs to support@mavimax.com

Chapter 4

SEE ALSO

`exjld(8)` `libexjava(8)` `module_developer_guide(guides)`

Chapter 5

COPYING

© Mavimax, Ltd
