

**LIBEXJAVA(8)**

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>SYNOPSIS</b>	<b>1</b>
<b>2</b>	<b>DESCRIPTION</b>	<b>2</b>
<b>3</b>	<b>BUGS</b>	<b>3</b>
<b>4</b>	<b>SEE ALSO</b>	<b>4</b>
<b>5</b>	<b>COPYING</b>	<b>5</b>

## Chapter 1

# SYNOPSIS

libexjava.so libexjava.dylib

## Chapter 2

# DESCRIPTION

This library provides core functionality of the Enduro/X Java binding. It implements the actual logic for Java JNI native calls to be translated to Enduro/X API. Library provides functionality for XATMI clients, where the calls are only directed from Java to XATMI and also provides functionality for XATMI servers, where the initial "AtmiCtx.tprun()" is called and once service is called, the call is redirected back to Java side where call is mapped to specific interface/object where is being invoked, so for servers it gets Java → C → Java chain of calls.

Library provides and access to XA distributed transactions functionality. Firstly calls are being to translated to **tpopen(3)**, **tpbegin(3)**, etc. But secondly, the library provides an XA Switch for JDBC libraries. Thus developers can use JDBC drivers to use two phase commit with Enduro/X. For such setup **NDRX\_XA\_RMLIB** is set to **libexjava.so** (or corresponding counterpart) and **NDRX\_XA\_DRIVERLIB** is set to **libndrxxajdbc.so**. When Java process performs the XA init, the **libexjava.so** is not loaded twice. In case if **tmsrv(8)** is configured with JDBC driver, then **libexjava.so** is loaded, jvm is started and **ndrxj\_xa\_init()** is called.

Library provides all buffer format accesses like UBF, JSON, STRING, etc and XATMI APIs, logging APIs.

When java loads the JNI library it loads **libexjava.so** lib which in turn globally loads the **libexjava**.

## Chapter 3

# BUGS

Report bugs to [support@mavimax.com](mailto:support@mavimax.com)

## Chapter 4

## SEE ALSO

**libexjavalld(8)** **tmsrv(8)**

## Chapter 5

# COPYING

© Mavimax, Ltd