

JdbcXaSupportEvaluation

Databases XA support evaluation

Those are the information collected while testing BTM against the most common databases. Also included:

- An overall evaluation of the database's XA support
- Special features and/or limitations
- A configuration example

XA only

Databases not providing an implementation of `javax.sql.XADataSource` have not been included.

If you find mistakes, inconsistencies or simply want some other database to be tested (as long as there is at least a free trial version available) just let us know via the [mailing list](#) and we'll consider your request.

Contents

- [Apache Derby](#)
- [IBM DB2](#)
- [Informix](#)
- [Microsoft SQL Server](#)
- [Oracle](#)
- [PostgreSQL](#)
- [Sybase ASE](#)

Apache Derby

Supported version(s)

Tested against versions 10.3.1.4 and 10.2.2.0.

Settings

All default settings are fine. Local transactions seems to mix well with global transactions thus you can enable `allowLocalTransactions`.

Derby seems to support Transaction Interleaving but not very well as it is slower and generates deadlocks.

Heuristics

I could not find how to list in-doubt transactions nor how to manually terminate them.

Example ResourceLoader configuration

```
resource.ds.className=org.apache.derby.jdbc.  
EmbeddedXADataSource  
resource.ds.uniqueName=derby  
resource.ds.maxPoolSize=5  
resource.ds.driverProperties.user=users1  
resource.ds.driverProperties.password=users1  
resource.ds.driverProperties.databaseName=us  
ers1
```

Evaluation

XA support in the driver and the DB look good. Documentation is too light.

IBM DB2

Supported version(s)

Tested against 9.1 Express-C.

Settings

All default settings are fine.

Heuristics

A single command allows listing in-doubt transactions, forcing their termination and forgetting them:

LIST INDOUBT TRANSACTIONS WITH PROMPTING

See: <http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp?topic=/com.ibm.db2.udb.admin.doc/doc/t0004636.htm>

Example ResourceLoader configuration

```
resource.ds.className=com.ibm.db2.jcc.DB2XAD
ataSource
resource.ds.uniqueName=dbtwo
resource.ds.maxPoolSize=5
resource.ds.keepConnectionOpenUntilAfter2Pc=
true    # (1.1 only)
resource.ds.driverProperties.user=db2user
resource.ds.driverProperties.password=db2pas
sword
resource.ds.driverProperties.databaseName=us
ers1
resource.ds.driverProperties.driverType=4
resource.ds.driverProperties.serverName=loca
lhost
resource.ds.driverProperties.portNumber=5000
0
```

Evaluation

BTM 1.0 cannot be used with IBM DB2. This is because IBM's JDBC driver does not allow XAResource calls after the java.sql.Connection object acquired from javax.sql.XAConnection has been closed (see <https://issues.apache.org/jira/browse/DERBY-246>). This is fixed in BTM 1.1 thanks to the new keepConnectionOpenUntilAfter2Pc parameter. BTM 1.2 removed that flag so DB2 just works without setting anything special.

Informix

Supported version(s)

Tested against versions 10.00.TC3TL and 10.00.TC5I1 both with driver version 3.00.JC3.

Settings

All default settings are fine. Local transactions seems to mix well with global transactions thus you can enable allowLocalTransactions.

Informix fully supports Transaction Interleaving so you should set deferConnectionRelease to false if you

properly configured the datasource and the database: all tables must be configured with row-level locking and you have to update the statistics in the system catalogs that the optimizer uses to determine the lowest-cost query plan. This basically means executing these commands:

```
ALTER TABLE [table 1] LOCK MODE (ROW);
ALTER TABLE [table 2] LOCK MODE (ROW);
ALTER TABLE [table 3] LOCK MODE (ROW);
...
UPDATE STATISTICS HIGH;
```

Heuristics

This gets the list of prepared transactions: onstat -G.

I could not find to to heuristically terminate the TX however. Documentation speaks about onmode -Z <address> but I get this error when I try:

```
onmode: Cannot kill transaction <address>.
Only I-STAR subordinates that are PREPARE'd
or HEURISTICALLY ABORT'd
may be heuristically completed.
```

Example ResourceLoader configuration

```
resource.ds.className=com.informix.jdbcx.Ifx
XADataSource
resource.ds.uniqueName=informix
resource.ds.maxPoolSize=5
resource.ds.driverProperties.user=informix
resource.ds.driverProperties.password=inform
ix
resource.ds.driverProperties.serverName=ifxt
est
resource.ds.driverProperties.databaseName=us
ers1
resource.ds.driverProperties.portNumber=1526
resource.ds.driverProperties.ifxIFXHOST=loca
lhost
resource.ds.driverProperties.ifxIFX_LOCK_MOD
E_WAIT=5
resource.ds.driverProperties.ifxIFX_XASPEC=Y
```

Evaluation

You have to create your databases with `create database with log` or else the XA datasource will refuse to connect to it. If you do not configure driver property `ifxIFX_LOCK_MODE_WAIT` to be different than 0, you will get this exception if you run multiple transactions concurrently:

```
java.sql.SQLException: Could not insert new
row into the table.
...
Caused by: java.sql.SQLException: ISAM
error: key value locked
...
```

See <http://publib.boulder.ibm.com/infocenter/idshelp/v10/index.jsp?topic=/com.ibm.jdbc.doc/jdbc55.htm>.

Another cause for this problem is that by default XA transactions with the same GTRID don't share locks. Setting `ifxix_xaspec` to `Y` allows those locks to be shared which not only helps solving locking problems but will also improve overall speed.

See <http://www-128.ibm.com/developerworks/db2/library/techarticle/dm-0602sudhakar/index.html#tight>.

Microsoft SQL Server

Supported version(s)

Tested against version 2005 Express with Microsoft JDBC driver version 1.2

Settings

All default settings are fine. Local transactions seems to mix well with global transactions thus you can enable `allowLocalTransactions`.

Heuristics

XA transactions are controlled by Microsoft DTC. Run Control Panel > Administrative Tools > Component Services. Then choose Component Services > Computers > My Computer > Distributed Transaction Coordinator > Transaction List.

From there, you have a list of all in-doubt transactions. You can force termination or forget by right clicking on the entries.

Example ResourceLoader configuration

```
resource.ds.className=com.microsoft.sqlserver.jdbc.SQLServerXADataSource
resource.ds.uniqueName=sqlsrv
resource.ds.maxPoolSize=5
resource.ds.driverProperties.user=users1
resource.ds.driverProperties.password=users1
resource.ds.driverProperties.serverName=localhost
```

Evaluation

XA seems to work fine when everything is setup properly but this is no simple task.

You need to grab the Microsoft JDBC driver from <http://msdn.microsoft.com/data/jdbc/>. Version 1.1 does not work so

grab version 1.2 instead. You then have to copy the `sqljdbc_xa.dll` inside SQL Server's `bin` folder then execute `xa_install.sql` as user `sa` as documented.

You must also enable MS DTC's XA support and grant `SqlJDBCXAUser` role to your user as explained here: <http://msdn2.microsoft.com/en-us/library/aa342335.aspx>

Note to Windows XP users:

Windows XP lacks some internal components related to DTC as discussed here: <http://forums.microsoft.com/msdn/showpost.aspx?postid=262525>. Microsoft has fixed this issue and released a Hotfix: <http://support.microsoft.com/kb/922668>. Unfortunately you cannot freely download it, you must call Microsoft support to get it. Also, it refuses to install on any non-English version of Windows.

Windows 2000 and Windows Server 2003 are said to be unaffected by this problem.

Oracle

Supported version(s)

Tested against 9.2.0.1 and 10 XE both with Oracle driver 10.2.0.1.0.

Settings

All default settings are fine. Local transactions seems to mix well with global transactions thus you can enable `allowLocalTransactions`.

Heuristics

I could not heuristically terminate a prepared transaction with `COMMIT FORCE` / `ROLLBACK FORCE`. Could be a bug in my version of XE or me not understanding how to do it. I followed these instructions without success: http://www.akadia.com/services/ora_important_part_1.html#How%20to%20remove%20%ABin-doubt%BB%20transactions. There is no show-stopper except for this problem but I suspect it works fine in other versions.

To view current XA transactions, query system view `SYS.DBA_2PC_PENDING`.

Example ResourceLoader configuration

```
resource.ds.className=oracle.jdbc.xa.client.  
OracleXADataSource  
resource.ds.uniqueName=oracle  
resource.ds.maxPoolSize=5  
resource.ds.driverProperties.user=users1  
resource.ds.driverProperties.password=users1  
resource.ds.driverProperties.URL=jdbc:oracle  
:thin:@localhost:1521:XE
```

Evaluation

The driver and the DB support all the features of XA but sometimes in an exotic and non-standard way. Handling of `XAResource.XA_RDONLY` is quite bizarre. Error messages returned by the driver are poor at best and often non-existent. Documentation is low in quality and sparse.

You need these privileges to run XA transactions:

```
grant select on sys.dba_pending_transactions
to user_test;
grant select on sys.pending_trans$ to
user_test;
grant select on sys.dba_2pc_pending to
user_test;
grant execute on sys.dbms_system to
user_test;
```

I could not find documentation about this anywhere, I had to spy the driver to figure that out.

There are several limitations to Oracle's XA support. One of them is for instance the use of temporary tables which is forbidden, see: <http://forums.oracle.com/forums/thread.jspa?threadID=638791&tstart=0>. There is a suggestion of a workaround using two connections.

Some user reported to have tested direct use of the underlying `OracleConnection` instead of the BTM connection handler but that didn't work. Other workarounds have been tried too without much success either. See <http://www.nabble.com/Oracle-exception-to16479456.html> for more details.

If someone can suggest a working solution be welcome to contribute it.

Oracle Spatial cannot be used together with XA anymore starting with 10g due to this limitation, see: <http://escuelle.blogspot.com/2008/11/ora-14450.html>

PostgreSQL

Supported version(s)

Tested against version 8.3 with driver version 8.3-603.

Settings

All default settings are fine.

Heuristics

Not fully supported. You can list in-doubt transactions and force their termination but the database will immediately

forget about them.

To get the list of in-doubt transactions, query the `pg_prepared_xacts` system view.

You can then forcibly commit or rollback any pending one with

```
COMMIT TRANSACTION '<content of the GID column>' or ROLLBACK TRANSACTION '<content of the GID column>'
```

Example ResourceLoader configuration

```
resource.ds.className=org.postgresql.xa.PGXA  
DataSource  
resource.ds.uniqueName=pgsql  
resource.ds.maxPoolSize=5  
resource.ds.driverProperties.user=postgres  
resource.ds.driverProperties.password=postgres  
resource.ds.driverProperties.databaseName=postgres  
resource.ds.driverProperties.serverName=localhost
```

Evaluation

Started to work really well very recently. Driver version 8.3-603 is the very first one that is fully working with BTM, do not try using any earlier version.

There is a configuration setting called `max_prepared_transactions` in `postgresql.conf` which defaults to 5. This is the maximum amount of XA transactions that can exist at any single time in the database. The default setting is way too low to be useful. It is recommended to set it at least to twice the maximum amount of connections in your pool.

Sybase ASE

Supported version(s)

Tested against versions 12.5.3 and 15 both with driver jConnect version 5.5.

Settings

All default settings are fine. Local transactions seems to mix well with global transactions thus you can enable `allowLocalTransactions`.

Heuristics

the `sp_transactions` stored procedure shows in-doubt transactions.

Terminate an in-doubt transaction with `dbcc complete_xact ('<content of xactname column of sp_transactions>', '<commit/rollback>')`.

Forget a transaction with `dbcc forget_xact ('<content of xactname column of sp_transactions>')`.

User must have `sa_role` role to call `sp_transactions` or `dbcc`, see: http://infocenter.sybase.com/help/topic/com.sybase.help.ase_12.5.1.dtm/html/dtm/dtm1.htm.

Example ResourceLoader configuration

```
resource.ds.className=com.sybase.jdbc2.jdbc.SybXADataSource
resource.ds.uniqueName=ase
resource.ds.maxPoolSize=5
resource.ds.driverProperties.user=users1
resource.ds.driverProperties.password=users1
resource.ds.driverProperties.serverName=localhost
resource.ds.driverProperties.portNumber=5000
```

Evaluation

XA support is handled by Distributed Transaction Manager that is not part of the default install (must be manually selected during the installation) and must be manually enabled in the DB before it can be used: http://manuals.sybase.com/onlinebooks/group-as/asg1250e/xainterf/@Generic_BookTextView/1079.

After too many unfinished transactions (*not sure here*) the database segments get full because of transaction logs. Symptoms are that connections get stuck and eventually even opening a new connection freezes the client process. You can clean that using these commands as user `sa`:

```
dump tran <database name> with truncate_only
```

See: http://www.isug.com/Sybase_FAQ/ASE/section1.3.html#1.3.1.

There is a bug in the JDBC driver that messes up recovery: calling commit or rollback on a heuristically terminated transaction, an `XAException` is thrown with an invalid error code (0) instead of `XA_HEUR<something>`. Stack

trace looks as follow in that case:

```
javax.transaction.xa.XAException:  
Unrecognized return code from server: -259  
at  
com.sybase.jdbc3.jdbc.SybXAResource.a(Unknown  
Source)  
at  
com.sybase.jdbc3.jdbc.SybXAResource.a(Unknown  
Source)  
at  
com.sybase.jdbc3.jdbc.SybXAResource.a(Unknown  
Source)  
at  
com.sybase.jdbc3.jdbc.SybXAResource.rollback  
(Unknown Source)  
at  
bitronix.tm.drivers.SybaseTest.testRecover(S  
ybaseTest.java:101)  
...
```

See http://search.sybase.com/kbx/changerequests?bug_id=417417

There is another bug in the driver that makes it add a `SQLWarning` to a connection each time it is enlisted.

See http://search.sybase.com/kbx/changerequests?bug_id=531772

Driver and DB seems to be reasonably well implemented and documented otherwise.