ExtenDB Connectivity Guide

ExtenDB Parallel Server

Version 1.1

October 2006

Table of Contents

1 Connectivity

This document discusses how to connect to an ExtenDB database server.

There are three different methods for connecting to the ExtenDB database: using JDBC, ODBC and perl DBI. The command line utility, cmdline, can also be used, which itself essentially uses JDBC to connect.

1.1 JDBC

The JDBC Driver is found in the xdbjdbc.jar included in the distribution.

The class name to use is com.extendb.connect.XDBDriver.

The JDBC string used must include username and password information for connecting. There are two alternative formats that are allowed:

jdbc:xdb:<database>:<username>/<password>@<host>[:<port>]

jdbc:xdb:<database>:<host>[:<port>]?user=<username>&password=<password>

The following additional options are also available as parameters to the JDBC url:

Parameter	**Description**
waittimeout	The timeout value in seconds to use after which if there is no response the request errors out. The recommended and default value is 0, which means wait an unlimited amount of time.
unicode	Whether or not to use Unicode. Default is false. If the database uses Unicode, then this must be set to true when connecting in the JDBC url.
charset	Which character set to use. Default is windows-1252.
mode	Setting this to P indicates that the underlying connections associated with this connection should persist, instead of releasing them back to the pools. When executing many statements in rapid succession, this may allow for a small increase in performance.

1.2 ODBC

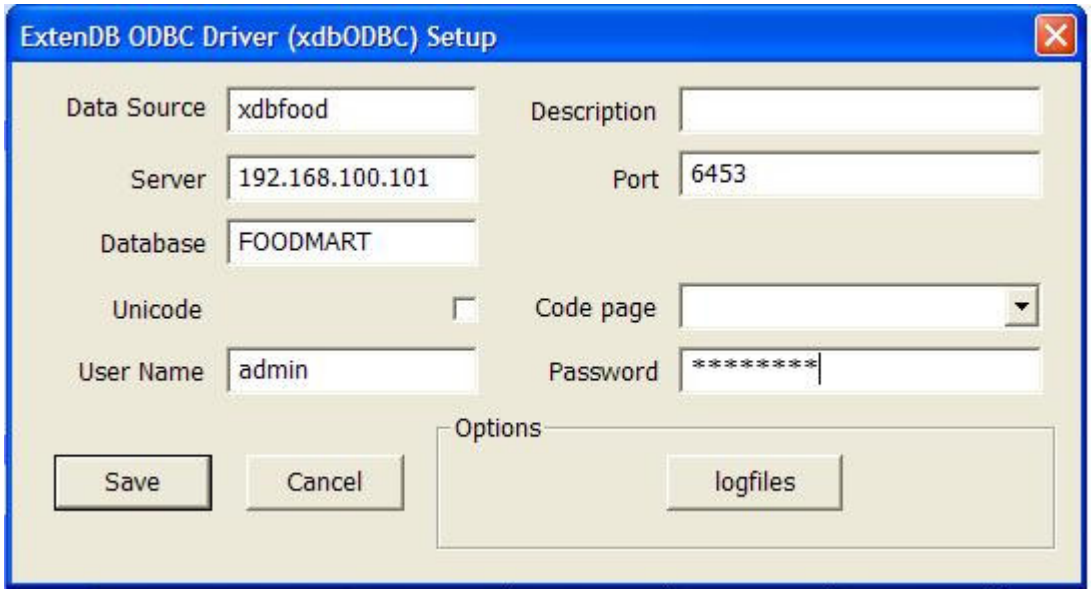
For Microsoft Windows clients, an ODBC driver is available.

To install, execute xdbsetup.exe that was included in the ExtenDB distribution.

After installation, ODBC data sources can be configured from the administration tool, Data Sources (ODBC), either under Administrative Tools or the Control Panel.

To create a System DSN, select the System DSN tab, and click Add.
Choose ExtenDB for the driver type.

Afterwards, you will see a window like this:



Fill in the corresponding information, naming the data source in the Data Source text box, and specifying the database and server you wish to connect to. The default port of 6453 is pre-filled, but can be changed if necessary. Finally, include a valid user name and password to connect to the database.

In addition, a button appears for log file options, which may be helpful in debugging any issues with the ODBC Driver.

1.3 Perl DBI

ExtenDB also includes a perl DBI library for communicating with the ExtenDB server.

For more information about DBI, see http://dbi.perl.org.

1.3.1 Dependencies

This module requires these other modules and libraries:

```
DBI
IO::Select
IO::Socket
POSIX
```

1.3.2 Installation

Before installing, make sure that Perl 5.005_04 is installed and working, and that DBI 1.14 or higher is installed and working.

For testing, you will need a working ExtenDB environment with a test database that a table called 'test' can be created in.

To build the module, execute the following commands:

```
perl Makefile.PL
make
```

To install the module:

```
make install
```

Setting the DSN

To test the module, first set the following environment variables:

The datasource name must be in the format

dbi:ExtenDB:database=<database>[:host=<host>][:port=<port>]

The default host is localhost and the default port is 6453.

A sample script for connecting to an ExtenDB database and performing a SELECT query appears below.

```
#!/usr/bin/perl -w
```

```
## check that we can connect and disconnect

use strict;
use Test::More;
use DBI;


plan tests => 6;

my $dbh = DBI->connect(
     'dbi:ExtenDB:database=mydb:host=localhost:port=6453',
     'admin', 'password',
     {RaiseError => 0, PrintError => 1, AutoCommit => 1});
ok(defined($dbh), 'Connect to database');

my $rows = 0;
my $sth = $dbh->prepare(q{SELECT id, name FROM test});
$sth->execute;

while (my $record = $sth->fetch()) {
    if (defined($record->[0]) && defined($record->[1])) {
        ++$rows;
    }
}
ok($rows == 3, 'Select rows');

ok($dbh->disconnect, 'Disconnect from database');
```