MTConnect C++ Agent

The C++ Agent provides the a complete implementation of the HTTP server required by the MTConnect standard. The agent provides the protocol and data collection framework that will work as a standalone server. Once built, you only need to specify the XML description of the devices and the location of the adapter.

See the MTConnect standard documents for more information on the protocol and organization of the information. The Adapter framework is also provided with full source code. (Available shortly).

Usage

```
./agent [-di] [{-a <adapters>}...] [-c <file>]
               [-f <file>] [-l <file>] [-p <port>]
    -a  <adapters>       : Location of adapter
                         > 'device:address:port'
    -c  <file>           : Adapter configuration file
    -d                   : Daemonize
                         > Default: false
    -f  <file>           : Configuration file
                         > Default: probe.xml
    -i                   : Interactive shell
    -l  <file>           : Log file
                         > Default: agent.log
    -p  <port>           : HTTP Server Port
                         > Default: 5000
```

```
-a <adapter>   : Specifies the data source for this agent. The
     adapter is a separate process that provide a data feed in
     pipe (|) delimited format. One adapter must be supplied
     with either -a or -c. The adapter must be in the format:
     <device>:<address>:<port>. The device is the device name in
     the XML file. The address is the ip address or machine
     name, and the port is the port to bind to. The default is
     7878.

-c <file>      : This is similar to the -a argument, but each
     adapter is list one per line.
```

```
-d             : Specifies if this process should run as a
     daemon. This only works on *NIX like operating systems.

-f <file>      : The XML file that specifies all the devices,
     components, and data items.

-i        : Run an interactive shell that allows a user to enter
     single values for a give device.

-p <port> : The port the server binds to. Default is 5000. To
     run on the default HTTP port, specify 80.
```

Directories

agent/	This contains the main application entry point and the Makefile
lib/	Third party library source. Contains source for cppunit, dlib, and libxml++.
samples/	Sample XML configuration files.
src/	The main source for the app.
test/	Various unit tests.
win32/	Libraries required only for the win32 build and the win32 solution.

Building

Windows

Open the solution in the win32/cppagent directory and build the project.

***NIX**

Use make from the agent directory.