Continuous Integration

Jenkins

Qpid uses the Apache Infrastructure CI service, which is a Jenkins installation. More details are available from the Apache wiki. Builds can run on any operating system instance available, or be locked to a particular type. It may be necessary to restrict some build types to Unix instances only, but using Jenkins will give access to Solaris and Windows Server build environments which will be useful for interoperability testing. If a build tool such as Maven is used, even tighter integration with the build process is available, including better test access and the ability to use the Apache Maven repository for publishing artefacts.

Operation

All builds are prefixed with Qpid- to allow grouping together, the Qpid view shows all jobs matching this pattern.

The following builds are configured:

<table>
<thead>
<tr>
<th>Build</th>
<th>Trigger</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qpid-Java-Java-MMS-TestMatrix</td>
<td>check-in</td>
<td>Tests the Java client with Java broker using the Memory message store. Matrix project permutating AMQP 0-10/0-9-1 protocols.</td>
</tr>
<tr>
<td>Qpid-Java-Java-BDB-TestMatrix</td>
<td>after MMS matrix pass</td>
<td>Tests the Java client with Java broker using BDB message store. Matrix project permutating AMQP 0-10/0-9-1 protocols.</td>
</tr>
<tr>
<td>Qpid-Java-Java-DBY-TestMatrix</td>
<td>every 24h</td>
<td>Tests the Java client with Java broker using Derby message store. Matrix project permutating AMQP 0-10/0-9-1 protocols.</td>
</tr>
<tr>
<td>Qpid-Java-Cpp-Test</td>
<td>check-in</td>
<td>Tests the Java client with the C++ Broker.</td>
</tr>
<tr>
<td>Qpid-Java-Java-Test-IBMJDK1.6</td>
<td>every 24h</td>
<td>Tests the Java client with Java broker under IBM JDK 1.6. AMQP 0-10 only.</td>
</tr>
<tr>
<td>Qpid-Java-Java-Test-JDK1.7</td>
<td>every 24h</td>
<td>Tests the Java client with Java broker under Sun JDK 1.7. AMQP 0-10 only.</td>
</tr>
<tr>
<td>Qpid-Python-Java-Test</td>
<td>every 24h</td>
<td>Tests the Python client with the Java Broker against AMQP 0-8, 0-9, and 0-10 protocols.</td>
</tr>
<tr>
<td>Qpid-Python-Cpp-Test</td>
<td>every 24h</td>
<td>Tests the Python client with the C++ Broker.</td>
</tr>
<tr>
<td>Qpid-Java-Test-0.16</td>
<td>every 24h</td>
<td>0.16 Release branch test. Tests the Java client with Java broker. AMQP 0-10 only.</td>
</tr>
<tr>
<td>Qpid-Java-Artefact-Release</td>
<td>every 24h</td>
<td>Generates binary release archives and Maven artefacts for the Java Client/Broker.</td>
</tr>
<tr>
<td>Qpid-proton-c</td>
<td>check-in</td>
<td>Performs proton-c build (cmake/make) and executes the Python system test suite</td>
</tr>
<tr>
<td>Qpid-proton-j</td>
<td>check-in</td>
<td>Performs proton-j build (maven) and executes the Python/Java system test suite</td>
</tr>
<tr>
<td>Qpid-proton-jni</td>
<td>check-in</td>
<td>Performs proton-c build (cmake/make) and executes the Python/Java system test suite via the Java (JNI Bindings)</td>
</tr>
<tr>
<td>Qpid-proton-j-deploy</td>
<td>every 24h</td>
<td>Deploys proton-j snapshots artefacts to <a href="https://repository.apache.org/content/repositories/snapshots/">https://repository.apache.org/content/repositories/snapshots/</a></td>
</tr>
</tbody>
</table>

A timeout of 60 minutes is set for all jobs.

Job Configuration Notes

Qpid Java Build/Release Configuration

The Java builds use the latest 1.6 release of the JDK, and the latest available Ant version, which are both dependent on the versions available on the Jenkins instance running the job.

Only JUnit test results are reported on currently.

The release job archives the following files:

- `release.zip`
- `release.tar.gz`
- `release/maven/*`

Qpid Cpp Build Configuration
The following commands are used to perform the Cpp build.

```bash
cd trunk/qpid/cpp
./bootstrap; ./configure CXXFLAGS="-Wno-unused-result"; make clean;make -j4
```

(As of February 2013, the packages dependencies for Ubuntu 12.04 were: automake, g+, libboost, libboost1.46-dev, libboost-filesystem-dev, libboost-program-options-dev, libicu48, libstdc++-4.4.4-dev, libtool, pkg-config, python2.7-dev , python-dev , ruby, sasl2-bin, subversion, uuid-dev, cmake, swig).

**Qpid Python Test Configuration**

The following commands are used to run the Python tests:

```bash
cd trunk/qpid/python
ulimit -n 1024 # Workaround QPID-3237 - reduce open filehandles
ant -file qpid-python-test-ant.xml -Dqpid.executable=../java/build/bin/qpid-server -Dpython.test.ignore=-I.. /java/test-profiles/python_tests/Java010PythonExcludes "-Dpython.test.modules=-mqpid_tests.broker_0_10 -m qmf.console" -Dqpid.port=25672 -Dqpid.executable.args="-p 25672 -m 25000" test
```

**Proton-C**

The build of proton-c required the addition of Ubuntu packages cmake, swig.

**Future**

Jenkins can be configured to provide many more features with its built-in plugins. These include:

- FIXME/TODO task reporting
- Javadoc publishing
- Coverage reports

The use of these plugins will be investigated once the basic build and test cycle is satisfactorily configured. Future functionality proposed includes:

- Switch CPP Build to use cmake
- Integrate Windows build for Java and CPP Brokers
- Publishing SNAPSHOT versions of Maven artefacts to the Apache repository after a successful nightly build

**Notifications**

All jobs are configured to send email to notifications@qpid.apache.org when a build fails, becomes unstable or returns to stable.

Currently Andrew, Rajith, Robbie and Keith have access to the build configurations.

**Sonar**

qpid-java-build (the Qpid Broker for Java and the AMQP 0-8..0-10 JMS Client) are set up in Sonar: