XMLIndiceCocoon2.1

Integration of XML:DB into cocoon has slightly changed since 2.0 and is now quite more simple than before. The below description is even the complicated one, i.e. doing it by hand or having the Xindice database independent on Cocoon, i.e. standalone mode.

If you use the Cocoon build system and include the xmldb block, the changes to cocoon.xconf and the deployment of cocoon-xmldb-block.jar and xindice.jar are done automatically and you can test the samples add http://127.0.0.1/samples/xmldb/. *(Which don't exist at the moment.)*

Xindice standalone vs. embedded mode

Xindice 1.1 provides two modes for running. On the one hand there is the embedded mode, which should work per default when the xindice.jar is deployed with Cocoon (as it is when including xmldb block since 2003-11-28). On the other hand there is the standalone mode, the database runs independently on Cocoon.

cocoon.xconf

Edit cocoon.xconf to add the new schemes/pseudo protocols that is `xmldb:xindice:` for the standalone mode and `xmldb:xindice-embed:` for the embedded mode. Therefore locate the `<source-factories>` element that should look as follows:

```
<source-factories>
    <component-instance class="org.apache.excalibur.source.impl.ResourceSourceFactory" name="resource"/>
    ...
</source-factories>
```

And add the `<component-instance>` element for making the `xmldb:` pseudo protocol available:

```
<source-factories>
    <component-instance class="org.apache.excalibur.source.impl.ResourceSourceFactory" name="resource"/>
    <!-- xmldb pseudo protocol -->
    <component-instance class="org.apache.cocoon.components.source.impl.XMLDBSourceFactory" name="xmldb">
        <!-- Xindice driver -->
        <driver class="org.apache.xindice.client.xmldb.DatabaseImpl" type="xindice"/>
        <!-- Xindice 1.1 Embedded driver -->
        <driver class="org.apache.xindice.client.xmldb.embed.DatabaseImpl" type="xindice-embed"/>
        <!-- Add other XML:DB compliant database's drivers here -->
    </component-instance>
    ...
</source-factories>
```

Cocoon XML:DB block

Don't forget to add the cocoon-xmldb-block.jar to your classpath that contains the `org.apache.cocoon.components.source.impl.XMLDBSourceFactory` class.

NOTE: The xmldb block is available in current cvs-snapshots not in 2.1.3

Using Xindice in Cocoon

Assuming that, in XIndice, you created a collection named "myCollection" into a database named db your URI to access to this collection is:

```
xmldb:xindice-embed:///db/myCollection/#xpathquery
```

For instance, if you want to query the first one document URI is:

```
xmldb:xindice-embed:///db/myCollection/#/*[[1]]
```

So, into sitemap.xmap that defines your pipelines, simply add:

```
<map:match pattern="test.html">
    <map:generate src="xmldb:xindice-embed:///db/myCollection/#/*[[1]]"/>
    <map:serialize type="xhtml" mime-type="text/xml"/>
</map:match>
```

Specifying a Database in Embedded mode

To specify a database, you need to set a system property. You can do this by adding:
`-Dxindice.configuration=<path to your system.xml>` to the command line when starting your servlet container.

Then, create a system.xml file, based upon the one you'll find in `config/system.xml`. In there you'll find the path to your database `db` specified. (Note - I couldn't find one in the XIndice 1.1b1 release, I had to get it from CVS Upayavira

Xindice in standalone mode

Now that it works in embedded mode, let's try the standalone mode.

You need to copy and update a few libs which are coming with the xindice ditribution into your cocoon lib directory

- replace the old xindice-1.1b1.jar with the one from your xindice ditribution (if you made a build in xindice, it can be found in the %xindice_home%/dist directory)
- copy from %xindice_home%/java/lib the xmlrpc-1.1.jar, xmldb-xupdate.jar and xmldb-common.jar files into your cocoon lib directory
- check if xmldb-api-20021118.jar, xml-apis, xerces and xalan version are diffrent (at the moment of writing xindice1.1b3 and cocoon2.1.4-dev (cvs from january 2004) have the same versions)
- don't forget to restart your cocoon webapp

Instead of using `xindice-embed` you now have to change to `xindice`. Furthermore you have to specify where the XIndice database can be found /accessed. If it is on the same machine it's `localhost:4080` - at least if you didn't change the default XIndice port. You can also replace `localhost` with the address of a remote machine.

Example:

```
xmldb:xindice://remote.computer.com:4080/db/myCollection/#/*[[1]]
```

NOTE: While for Xindice 1.0 default port is 4080, Xindice 1.1 server is deployed on top of servlet engine, and you should use servlet engine's port number, such as 8080 or 8888. Xindice 1.1b3 comes with an optional Jetty and can run independently on port 8888

NOTE: If you want to use Xupdate from within xsp with the standalone Xindice you need to write a patch, because of a Xalan/Xerces bug see mail archive. A possible patch is described here. (It's quiet hard to recognize this bug, but if you read your Xindice log carefully. you'll notice the missing namespace declaration)