ExternalDTD

I want to read in an XML file in my component on startup and parse it into Java objects. For this I wrote a simple XML file:

```
<root>
 <element attr="some attr">
   <sub-element>value</sub-element>
 </element>
 <!-- some more elements -->
</root>
```

This didn't work. I needed a DTD, so I created one inline, but it turned out it had to be external. So I added

```
<!DOCTYPE root SYSTEM "root.dtd">
```

to the XML file. No luck. I had to create a catalog that refers to the proper location of the DTD.
Result:

- I added

```
    <parameter name="local-catalog" value="file:///d:/path/to/file/mycatalog.cat"/>
```

to the /cocoon/entity-resolver in cocoon.xconf. Note the file:/// for the Windows path, otherwise it won't work.

Note: set parameter verbosity to 10 to see if the catalog file is actually read in.

- I created mycatalog.cat in the proper directory

```
-- Catalog file (mycatalog.cat) for myProject --

-- Driver file for the root DTD --
PUBLIC "-//MYPROJECT//DTD root V1.0//EN"
        "file:///d:/path/to/file/mycatalog.dtd"

-- end of catalog file for myProject --
```

- I finally added the following line to my root.xml file:

```
<!DOCTYPE root "-//MYPROJECT//DTD root V1.0//EN" "">
```

- Now it works.

Update:

When I changed my Java code to use the org.apache.excalibur.xml.DomParser following the common Avalon rules (servicemanager.lookup(DomParser.ROLE) rather than initializing directly) I was able to change the above absolute paths to relative paths:

```
    <parameter name="local-catalog" value="relative_path/to/file/mycatalog.cat"/>

relative to Cocoon root dir
```

and

```
-- Catalog file (mycatalog.cat) for myProject --

-- Driver file for the root DTD --
PUBLIC "-//MYPROJECT//DTD root V1.0//EN"
       "mycatalog.dtd"

-- end of catalog file for myProject --

relative to the directory of the .cat file
```