

Steph

Howto use a PostgreSQL database containing images (store and retrieve).

In this example, images are stored in a database, more precisely in the table images which looks like:

• Colonne*		Type
imgname		text
img		bytea

The type bytea stands for the binary one : 'bytes array'.

Why choosing the byte array type ?

First of all, I must say that I the method to retrieve images from the DB using byte array does not need to write any java code (see 2)). Also, using byte array allows you to save your images when you do a dump of your base (it's more complex with blobs).

1) How to store an image

As I don't know cocoon so well I decided to wrote a little program in JAVA which will be able to do this for me. I choose JAVA because I wanted to know how to use JDBC and then, I expect it will be easier to move it into an XSP once I have understood how it work.

You will find the program in the attached file.

This file contains also a simple method to retrieve an image from the database which helped me to know whether the first method was right or not.

Why not doing it simply with "insert into images values" ?

An image is composed of a lot of bytes, so you won't do "INSERT INTO images VALUES ("my_image", million_of_chars);" (Once the program downloaded and an image inserted in the DB, just try to select your image and you will see how tall is the field containing the bytea)

2) How to retrieve an image using COCOON ?

I used **DataBase Readers** to do this. As I told you earlier, there is no need to use JAVA. You will find the complete documentation about DataBase Readers [here](#).

Even if I found my way through it took me a little time to understand the syntax so this is an example:
I will consider that you connection to the postgresql is pooled and named : 'my_pooled_connection'. If you want to do :

```
SELECT img FROM images WHERE imgname = 'name_given_in_the_url';
```

You should add to your sitemap (at the right place) :

```
<!-- components -->
<map:components>

<map:readers default="resource">

<map:reader name="databasereader" src="org.apache.cocoon.reading.DatabaseReader">

<use-connection>my_pooled_connection</use-connection>

</map:reader>
</map:readers>

<!-- pipeline -->
</map:components>

<map:match pattern="images/*.jpg">

<map:read type="databasereader" src="{1}" mime-type="image/jpg">

<!-- option sitemap parameters -->

<map:parameter name="table" value="images"/>

<map:parameter name="image" value="img"/>
```

```
<map:parameter name="key" value="imgname"/>
```

```
</map:read>
```

```
</map:match>
```

So when you do : http://my_server:8888/path_to_my_app/images/my_image.jpeg , you will see the image stored in the DB under the name of "my_image".

You will be please to correct this page if you find any mistake.

Stephane

Attachment: [InsertImages.java](#)