DB Port Tutorial

Installation and Startup

1. Compile all classes in the dbport folders with your favourite Java compiler. Make sure that the myDB package is in the classpath.

   Example: `javac -classpath <path of the myDB folder> dbport/*.java`

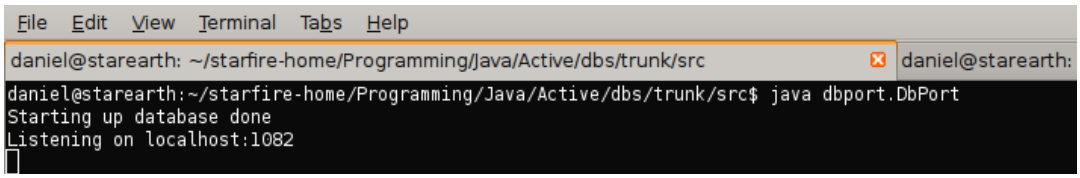
2. Create a folder “disk” in the folder that you want to run dbport in (without this folder, your database might be unable to persist its data):

   ```
   mkdir disk
   ```

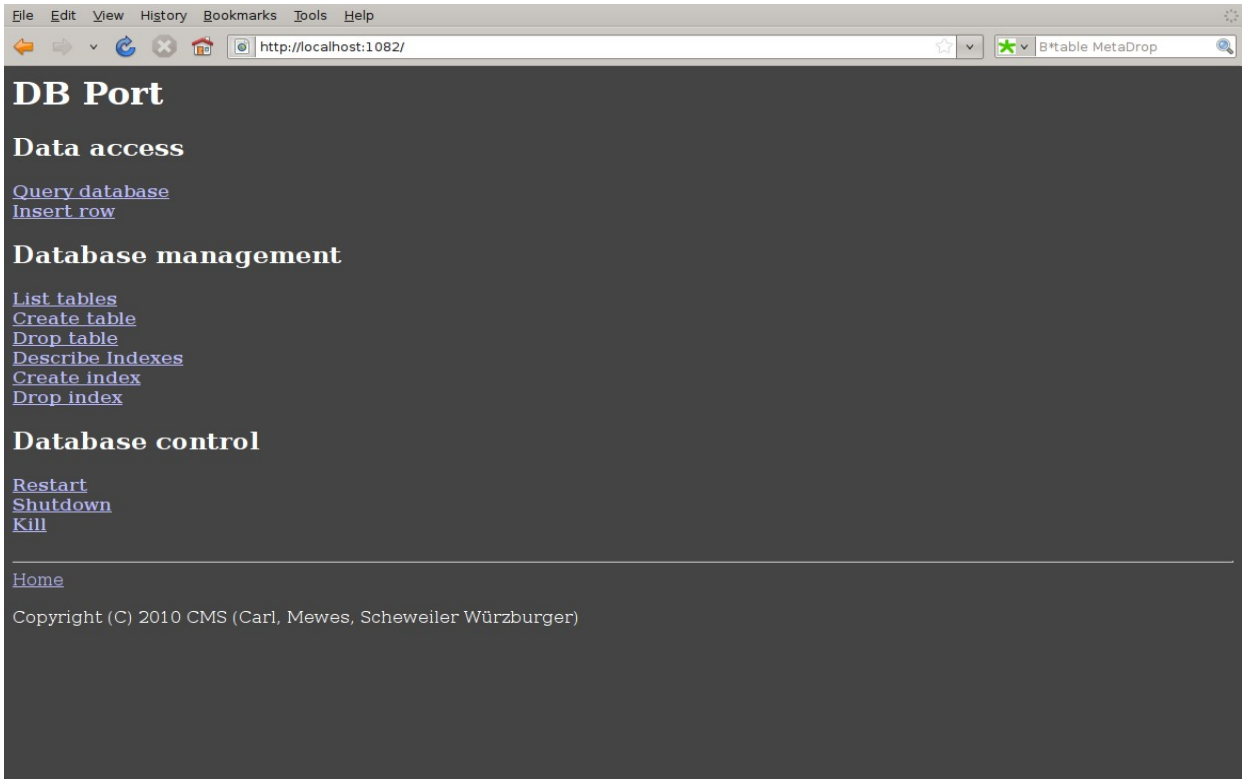
3. Run dbport.DbPort with your favorite Java virtual machine. Again, the myDB package must be in the path.

   Example: `java -classpath ./:<path of the myDB folder> dbport.DbPort`

   You should see a message similar to this one:



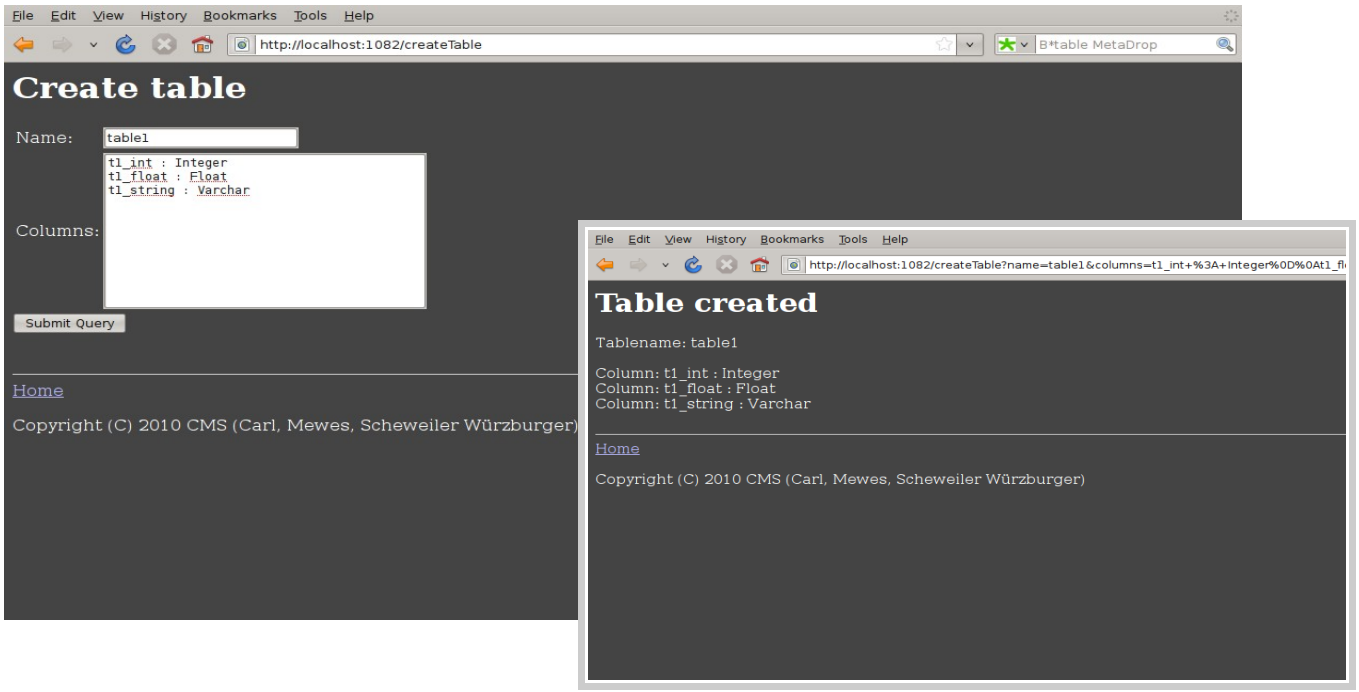
4. Per default, DB Port binds to localhost on port 1082 (check DbPort.java if you want to change it). Therefore, DB Port is accessible from you local host only. Open up your favorite web browser and navigate to http://localhost:1082



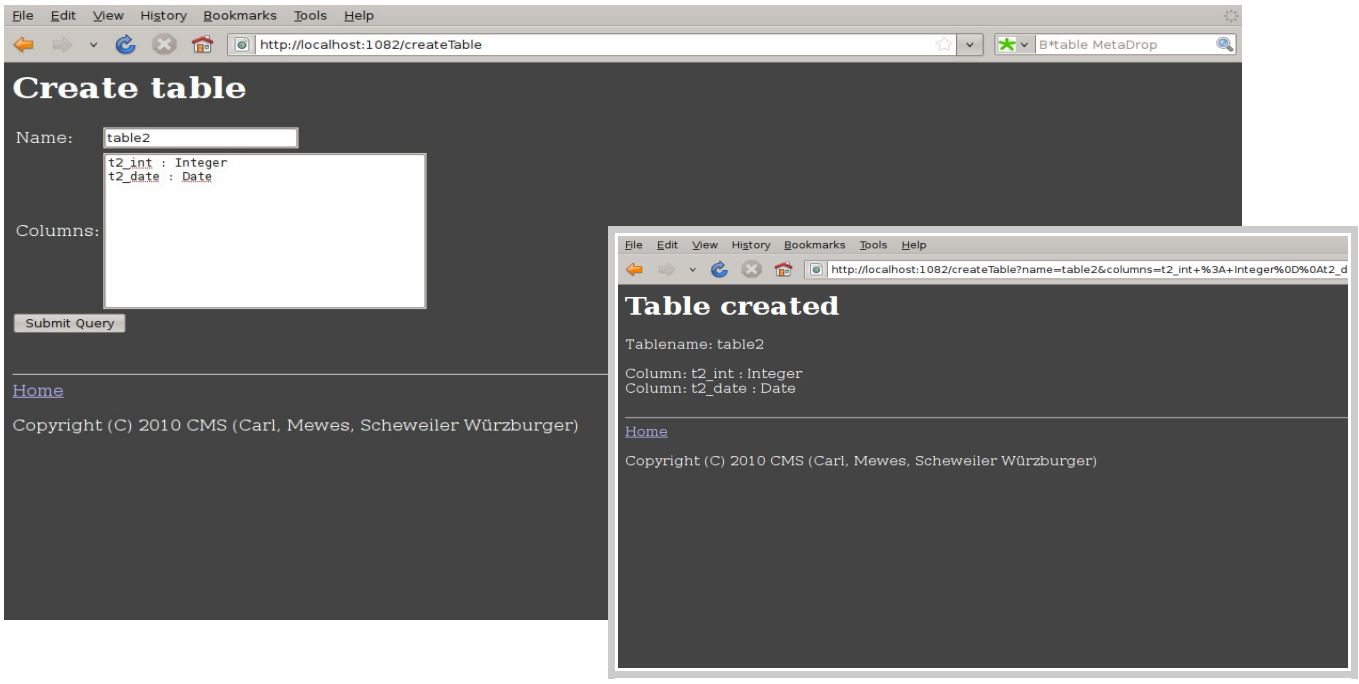
Database Management

We now want to create two tables and an index.

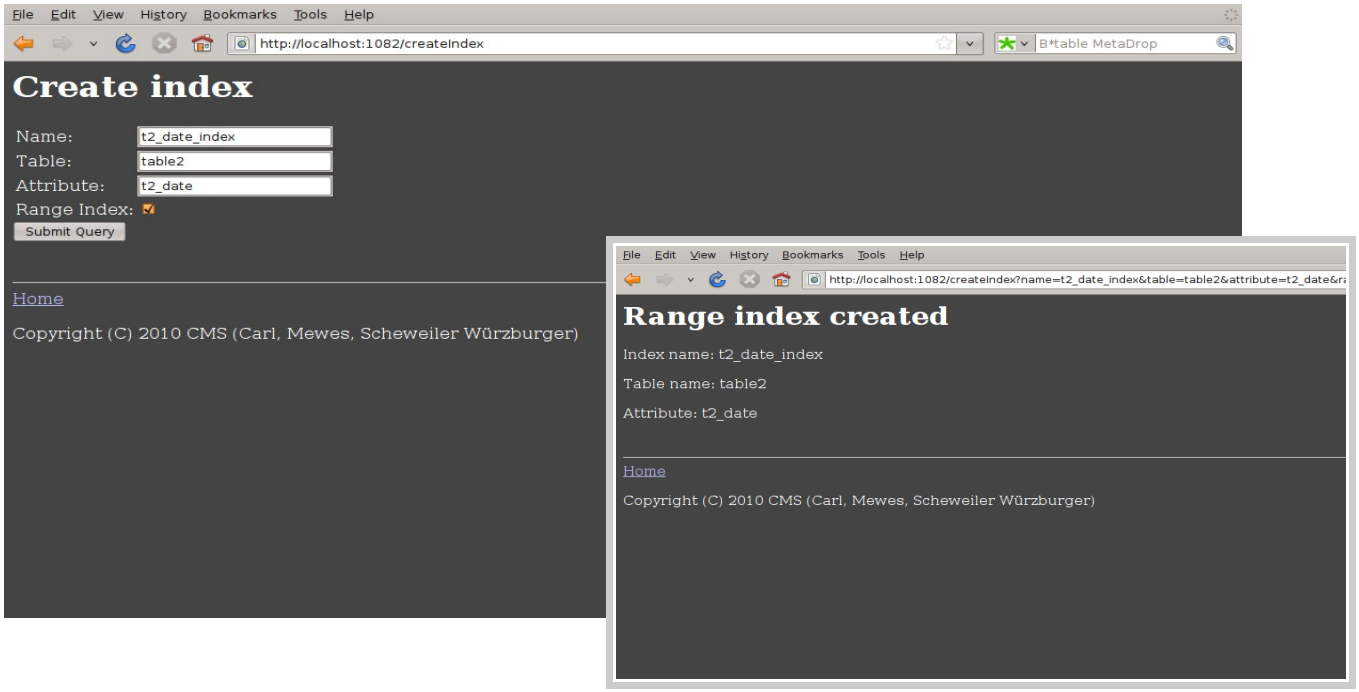
1. One the DB Port home page, click “Create table”.
2. Enter the table name “table1” as well as the schema. We want to have three columns in table1: “t1_int” of type Integer, “t1_float” of type Float and “t1_string” of type Varchar. To enter the schema, write one column per line in the format “<column name> : <type>”. Please note that the type is case sensitive and must be one of “Integer”, “Long”, “Float”, “Double”, “Date”, “Varchar” and “Char(<length>)”. Enter the data and click the submit button.



3. Create a second table with the name “table2” and two columns: “t2_int” of type Integer and “t2_date” of type Date.



4. Now we want to add an index for table2's date column. Therefore, return to DB Port's home page and click “Create index”. Enter the index name (e.g. “t2_date_index”), the table (“table2”) and the attribute (“t2_date”). Also check the “Range index” option, to enable the creation of an index with range query support. Enter all data and submit.

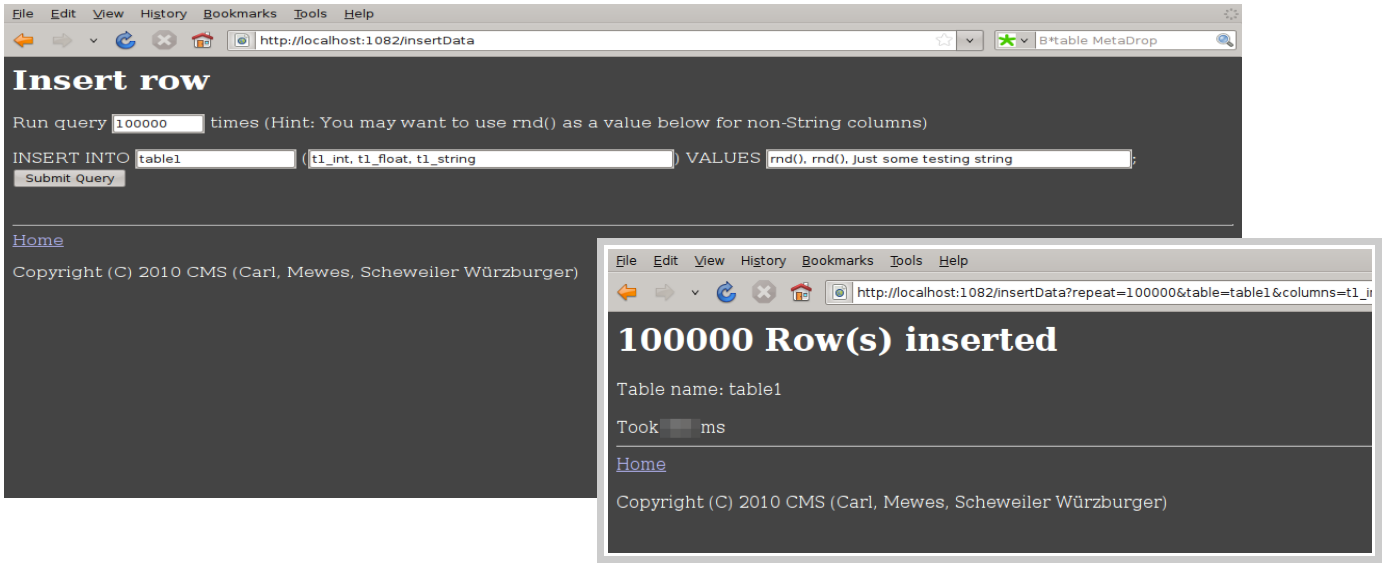


Inserting Data

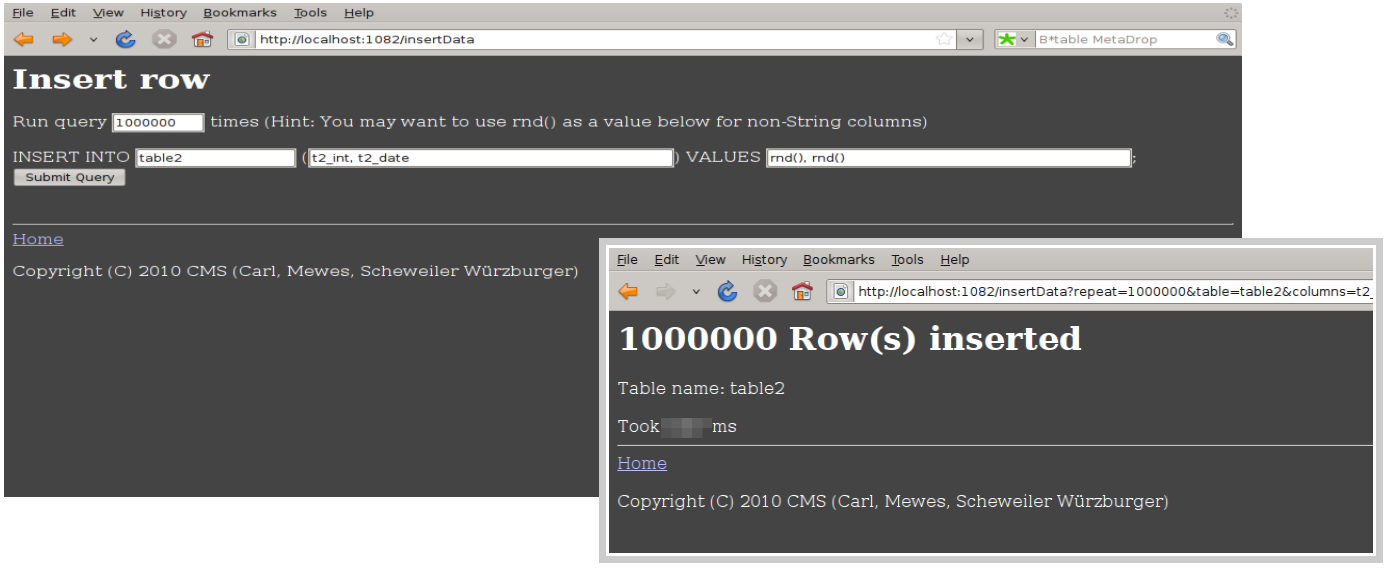
In the last steps, we have created two tables in our database. Now we want to fill them with some data.

1. Click the “Insert row” link on DB Port's home page.
2. We want to fill table1 with 100,000 rows. DB Port comes with two features, which greatly simplify the creation of test data. First, it has a repetition field that you can use to let DB port perform a certain action (inserting rows or querying the database) multiple times. The second feature is the insertion of random values (not supported for Varchar and Char type columns currently). We now use both features to create some test data.

   For this, enter “100000” into the repetition field. Make the insertion query read like “INSERT INTO table1 (t1_int, t1_float, t1_string) VALUES rnd(), rnd(), just some testing string;”. As you can see, we assign random values to t1_int and t1_float and the constant value “just some testing string” to t1_string (additional information: string values must not contain commata, enter null as a value to assign null). Click the submit button to insert the rows.



3. Repeat the procedure for table2 in order to insert a million random rows:

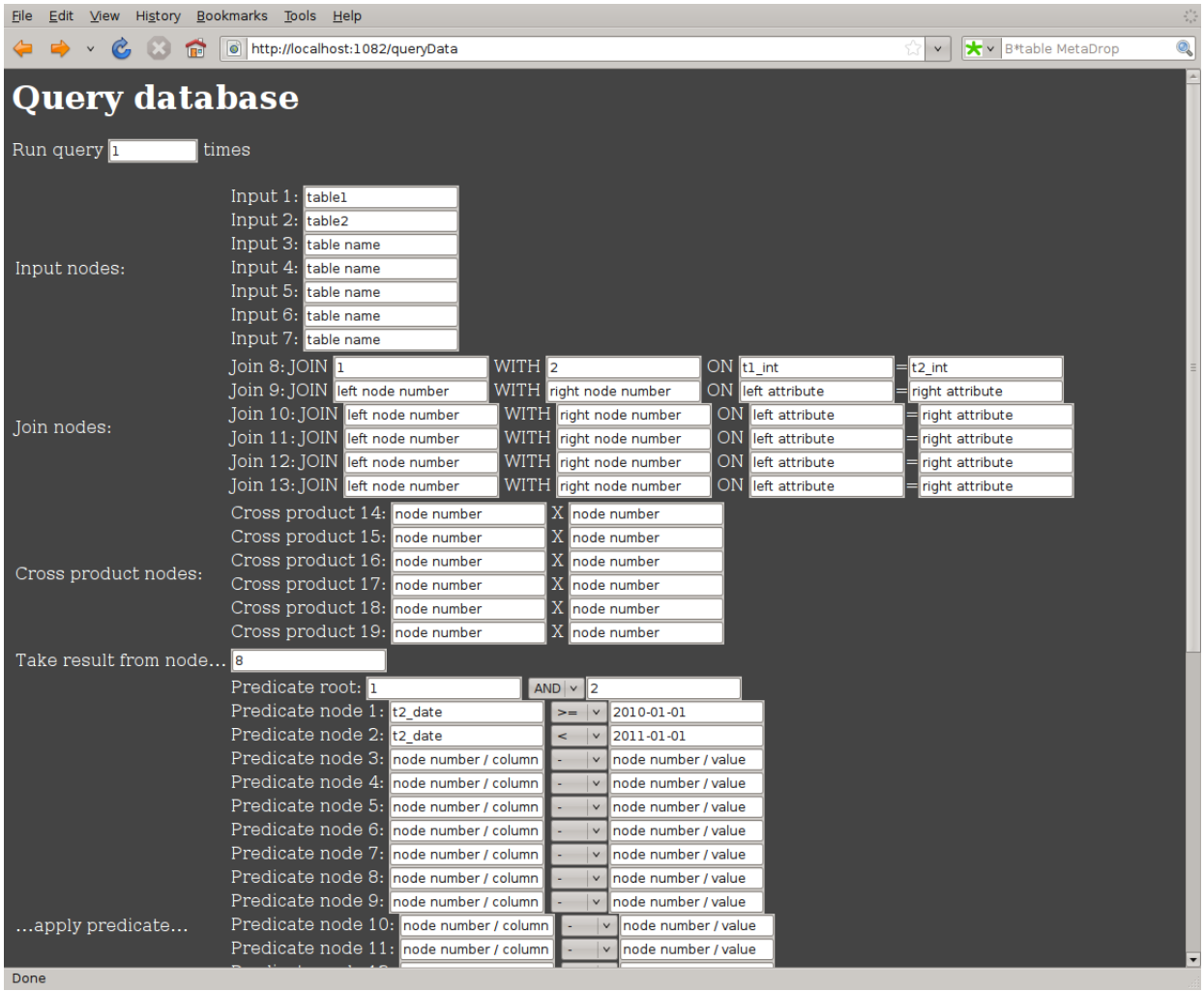


Querying the Database

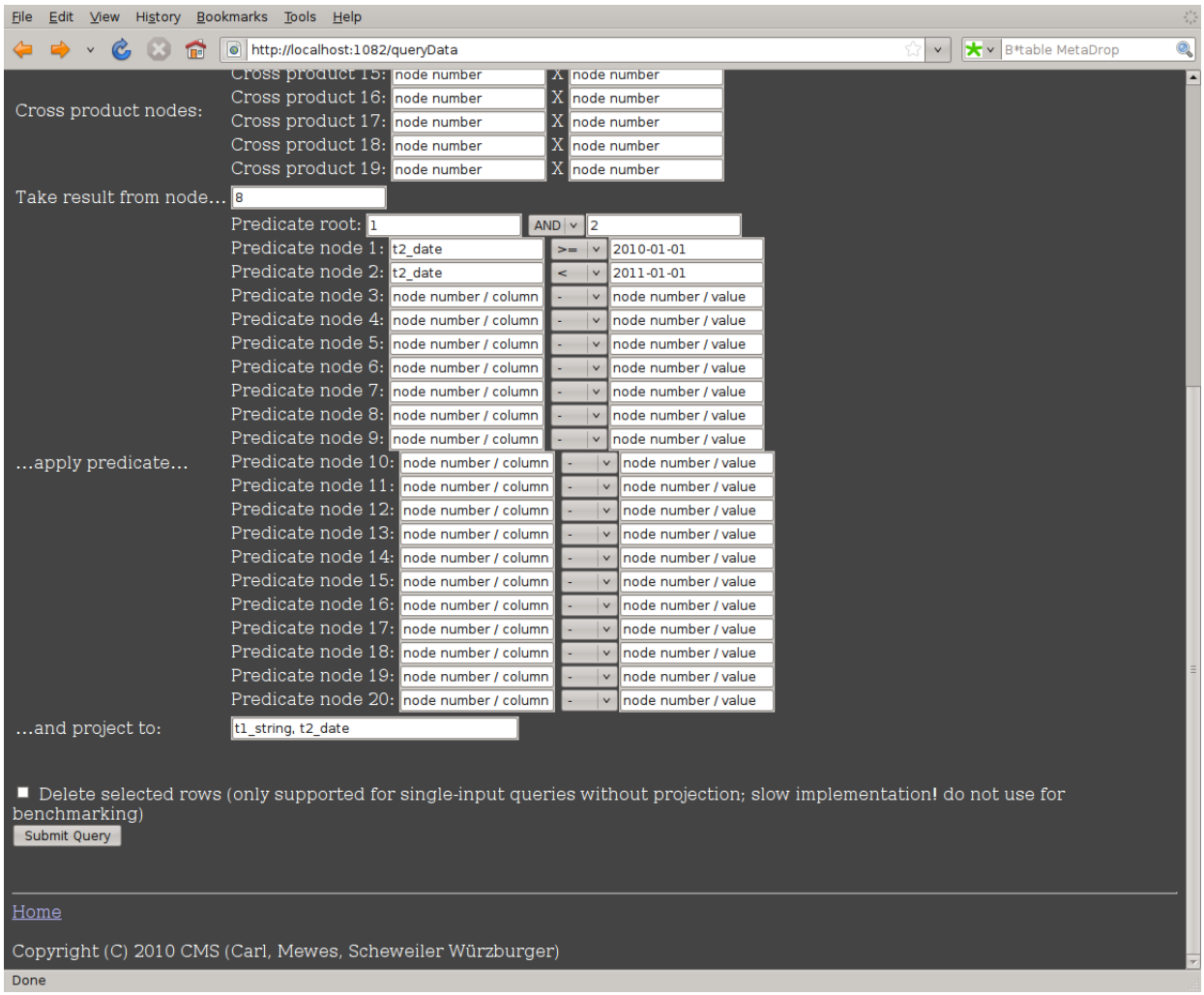
We have just inserted some rows into our tables. We now want to formulate a query to the database.

1. Navigate to “Query database” from DB Port's home page.
2. Formulating queries in DB Port works by
    1. defining a set of input, join and cross product nodes
    2. selecting a result node (like the root of a relational algebra expression)
    3. specifying predicate nodes which are applied as a selection on top of the result node
    4. specifying which columns the result should be projected on
3. We want to formulate the following query (in SQL): “SELECT t1_string, t2_date FROM table1 JOIN table2 ON t1_int = t2_int WHERE t2_date >= '2010-01-01' AND t2_date < '2011-01-01'; ” (i.e. select all dates from table2 which are in 2010 and also get the corresponding – with regard to t1_int=t2_int – t1_string values from table1).

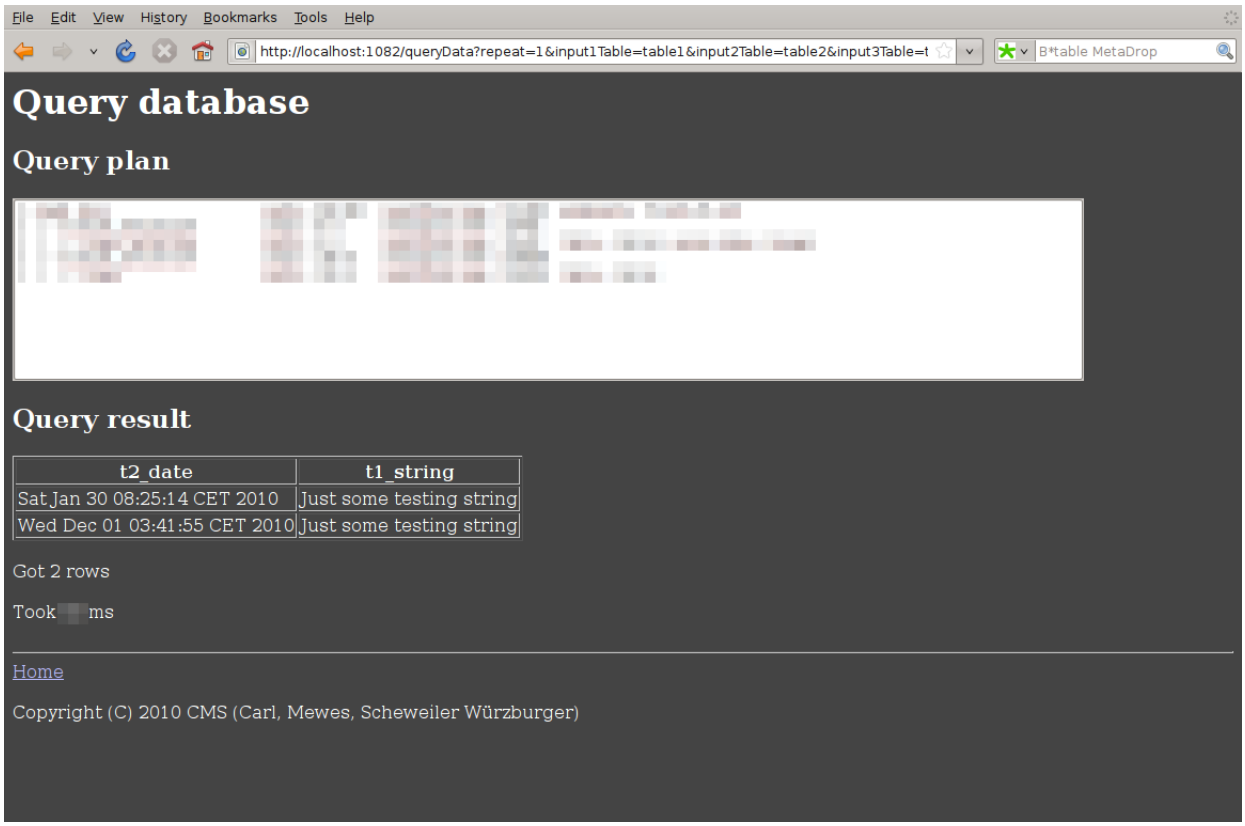
    Therefore, we first specify two input nodes. Each node has a node number. At node 1, enter “table1” as the table name. At node 2, enter “table2”.
4. To join both inputs, we use a join node. At node 8, enter “1” as the left node number (=table1) and “2” as the right node number (=table2). Also specify the join condition as “t1_int” = “t2_int”. This essentially specifies a tree with the join node as the root and the two input nodes as its children / as the leafs.



5. The root of our tree is join node 8. Enter “8” as the result node.
6. To specify the selection (“WHERE” clause in SQL), first set predicate node 1 to “t2_date >= 2010-01-01” and predicate node 2 to “t2_date < 2011-01-01”. To “AND” both predicates together and specify the result of the logical and as the predicate's tree root, set the predicate root to “1 AND 2”.
7. Enter “t1_string, t2_string” into the projection field. (Hint: to not apply a projection at all, simply empty the projection field).



8. Submit the query! DB Port displays the query plan as provided by the database's explainQuery() method and also shows the result of the query.



(sorry, we do not give away our ingenious query plans! ;-))

Finally...

... you should shutdown the database by using the “Shutdown” link on DB Port's home page. This ensures that the database gets properly persistated to disk. To not persistate any changes you have made, you may use the “Kill” function instead.