

Accessing kdb+ from PHP to create dynamic webpages

John Ludlow

June 20, 2011

1 Introduction

Open Database Connectivity (ODBC) can be used as an interface for exchanging data between databases. To access data from non-kdb+ databases from a kdb+ client, see:

<https://code.kx.com/trac/wiki/Cookbook/ODBC/qclient>

and

<https://code.kx.com/trac/browser/contrib/jludlow/docs/odbc.pdf?format=raw>

Here we will examine the alternative setup of kdb+ as the server, with a non-kdb+ client. Currently this is possible only for Windows. See:

<https://code.kx.com/trac/wiki/Cookbook/ODBC/qserver>

Details are presented for connecting from Excel and Visual Basic.

Currently, a popular set-up for many dynamic websites involves a web server such Apache, the MySQL database and a scripting language such as PHP. PHP code is included in a HTML source file, and allows server-side processing, along with requests to MySQL databases, thus enabling dynamic websites. Here we demonstrate how the kdb+ ODBC server can enable connections to be made from PHP to kdb+. This will be demonstrated via a simple proof of principle example.

The work presented in this report was carried out on a laptop running Windows Vista.

2 Connecting kdb+ and PHP

1. Download and run the appropriate kdb+ ODBC driver (32 or 64 bit).

In this case for 32 bits:

<https://code.kx.com/trac/browser/kx/kdb+/w32/odbc.exe>

For 64 bits, download:

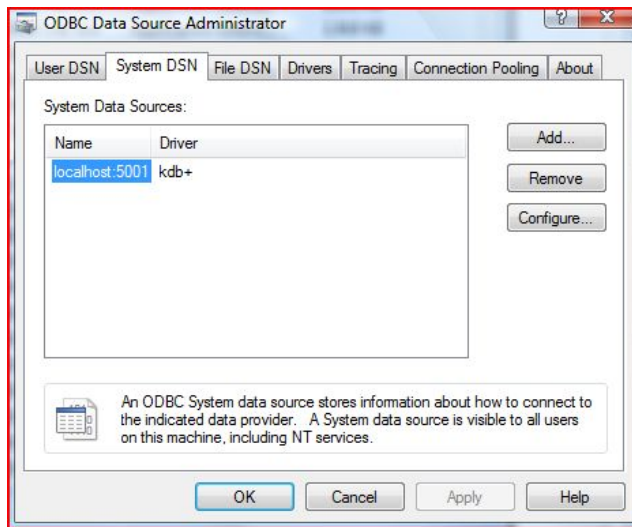
<https://code.kx.com/trac/browser/kx/kdb+/w64/odbc.zip>

and run d0.exe to install the ODBC driver.

2. Now a system DSN needs to be defined.

Run C:\Windows\System32\odbcas32.exe or C:\Windows\SysWOW64\odbcad32.exe depending on your operating system type.

You will need to specify a host and port, in this case localhost and port 5001. Remember to set the correct system permissions for your q folder.



3. Now we will install a web server and PHP. For Windows, an excellent free package is WampServer

<http://www.wampserver.com/en/download.php>

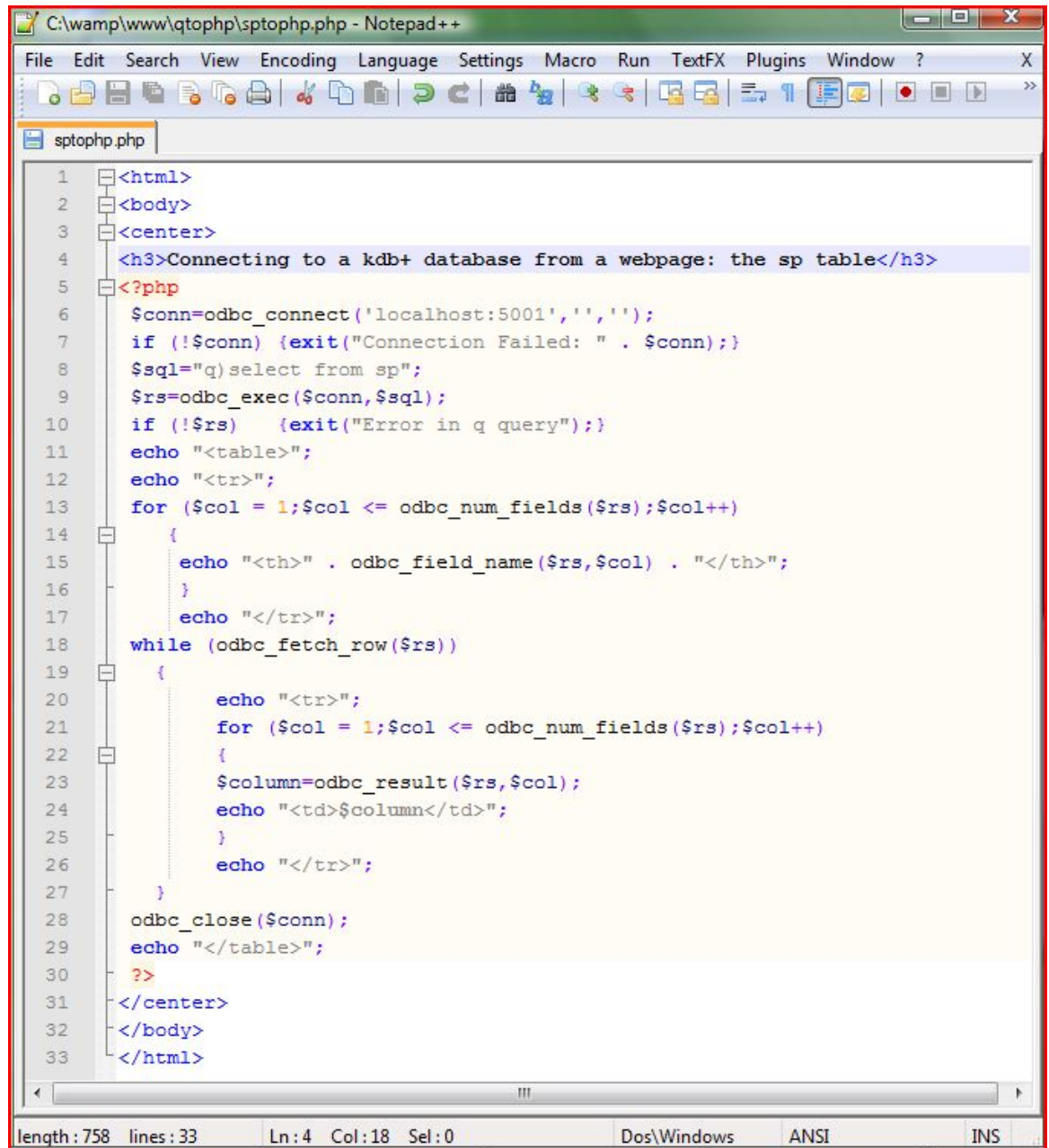
4. For the example data we will make use of the Suppliers and Parts database that can be loaded into kdb+ via the sp.q script.

<https://code.kx.com/trac/wiki/Startingkdbplus/tables#a4.3SuppliersandParts>

5. Now, we will write a simple PHP script to pull information from a kdb+ database via ODBC. For this we make use of the php ODBC

connector:

http://www.w3schools.com/PHP/php_db_odbc.asp



```
1 <html>
2 <body>
3 <center>
4 <h3>Connecting to a kdb+ database from a webpage: the sp table</h3>
5 <?php
6 $conn=odbc_connect('localhost:5001','','');
7 if (!$conn) {exit("Connection Failed: " . $conn);}
8 $sql="q)select from sp";
9 $rs=odbc_exec($conn,$sql);
10 if (!$rs) {exit("Error in q query");}
11 echo "<table>";
12 echo "<tr>";
13 for ($col = 1;$col <= odbc_num_fields($rs);$col++)
14 {
15     echo "<th>" . odbc_field_name($rs,$col) . "</th>";
16 }
17 echo "</tr>";
18 while (odbc_fetch_row($rs))
19 {
20     echo "<tr>";
21     for ($col = 1;$col <= odbc_num_fields($rs);$col++)
22     {
23         $column=odbc_result($rs,$col);
24         echo "<td>$column</td>";
25     }
26     echo "</tr>";
27 }
28 odbc_close($conn);
29 echo "</table>";
30 ?>
31 </center>
32 </body>
33 </html>
```

length: 758 lines: 33 Ln: 4 Col: 18 Sel: 0 Dos\Windows ANSI INS

6. The php script sptophp.php will be saved in a subdirectory under

C:\wamp\www

Note that the ODBC connection is established via the line (no user-name or password is needed here):

```
$conn=odbc_connect('localhost:5001','','')
```

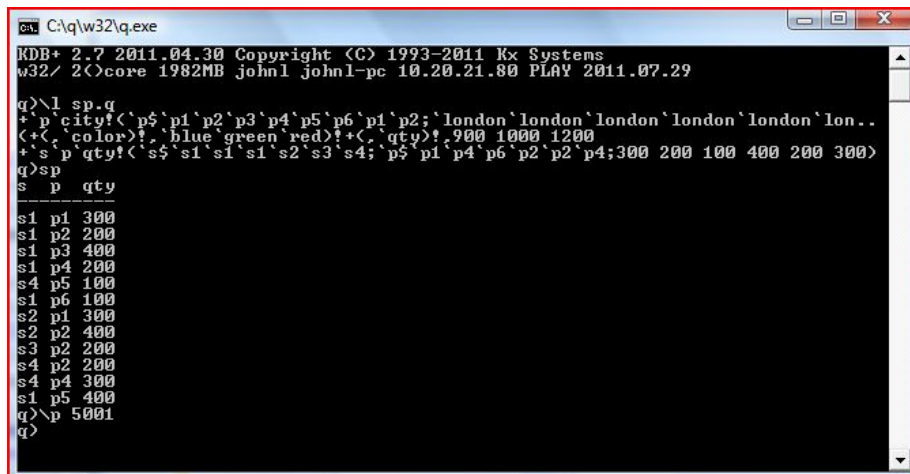
and the query as:

```
$sql="q)select from sp";  
$rs=odbc_exec($conn,$sql);
```

Note that the query can also be written in standard SQL as:

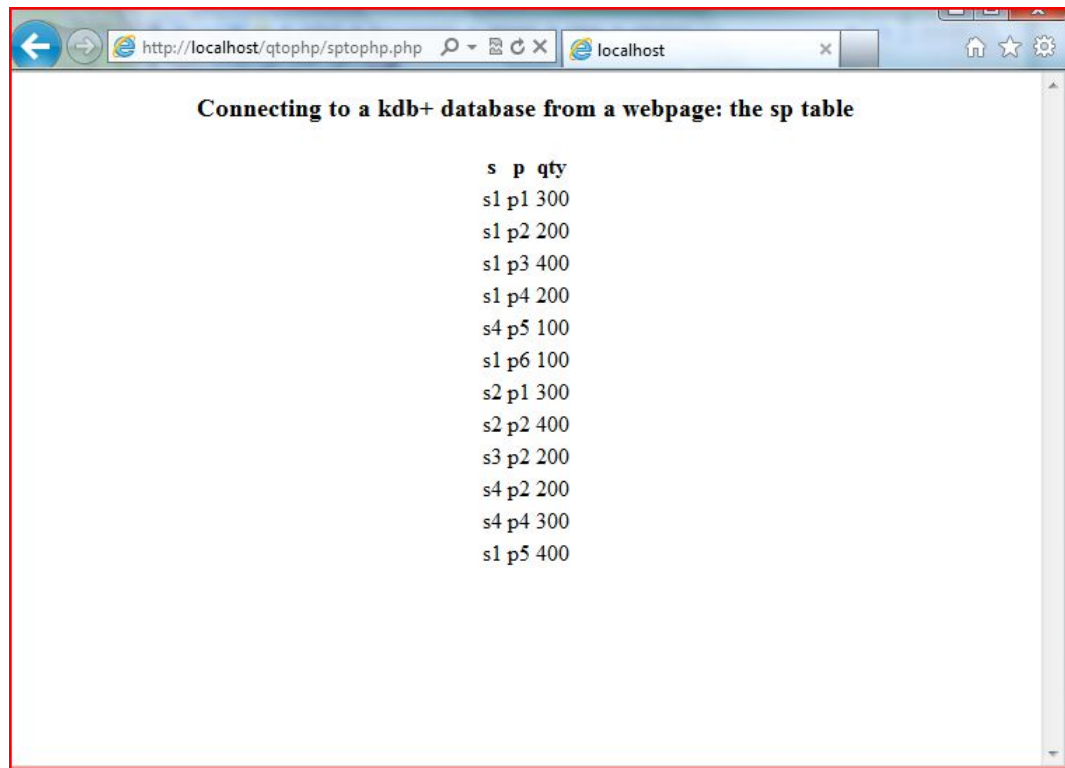
```
$sql="s)select * from sp";
```

7. Open a q session, load the sp.q script and listen on port 5001:



```
C:\q\w32\q.exe  
KDB+ 2.7 2011.04.30 Copyright (C) 1993-2011 Kx Systems  
w32/ 2(<core 1982MB john1 john1-pc 10.20.21.80 PLAY 2011.07.29  
  
q)\l sp.q  
*p city!(<p$`p1`p2`p3`p4`p5`p6`p1`p2;`london`london`london`london`london`lon..  
<(< color>!,blue`green`red)!*(< qty>!,900 1000 1200  
*s p qty!(<s$`s1`s1`s1`s2`s3`s4;`p$`p1`p4`p6`p2`p2`p4;300 200 100 400 200 300>  
q)sp  
s p qty  
-----  
s1 p1 300  
s1 p2 200  
s1 p3 400  
s1 p4 200  
s4 p5 100  
s1 p6 100  
s2 p1 300  
s2 p2 400  
s3 p2 200  
s4 p2 200  
s4 p4 300  
s1 p5 400  
q)\p 5001  
q>
```

8. Finally, open a browser and view the sp table on a webpage:



3 Comments

This short report offers a proof of principle on how to access kdb+ data from a webpage that is written in PHP using the kdb+ ODBC server. Following this approach, more complex implementations are possible, allowing users interactive access to a kdb+ server via a webpage, or the development of dynamic trading applications.