STPivot Tutorial, Clickables members
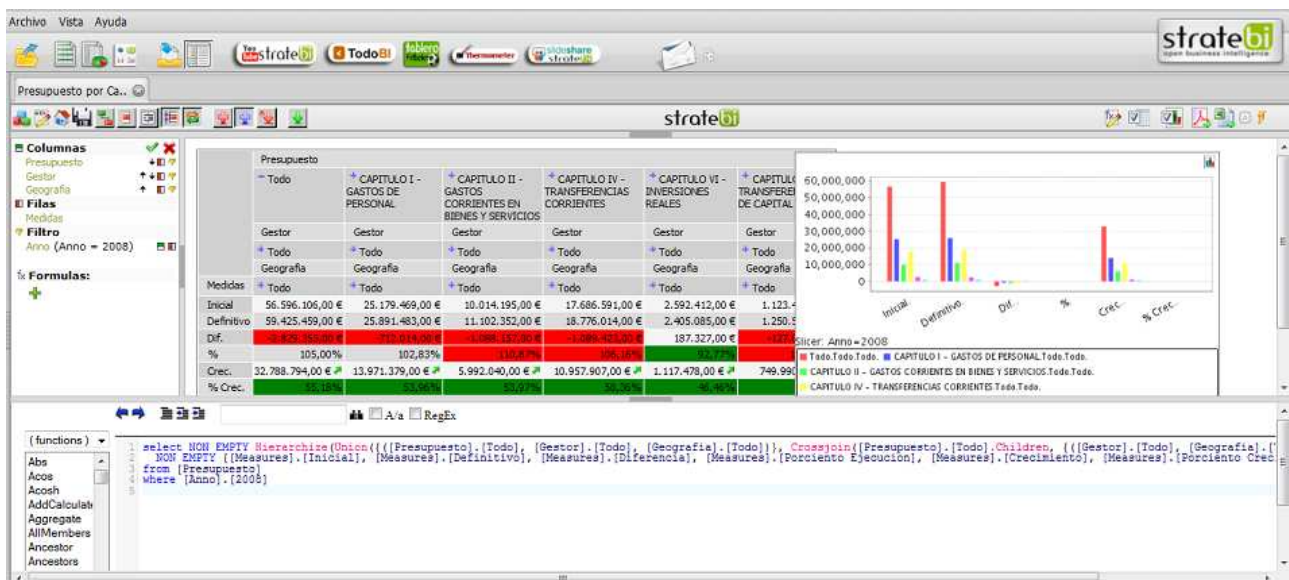
January 2012

About This Document

If you have questions that are not covered in this guide, or if you find errors in the instructions or language, please contact the Stratebi Technical Publications team at info@stratebi.com. The Publications team cannot help you resolve technical issues with products.

Support-related questions should be submitted through the STPivot Home Page at

http://code.google.com/p/stpivot

Limits of Liability and Disclaimer of Warranty

The author(s) of this document have used their best efforts in preparing the content and the programs contained in it. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher make no warranty of any kind, express or implied, with regard to these programs or the documentation contained in this book.

The author(s) and Stratebi shall not be liable in the event of incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of the programs, associated instructions, and/or claims.

Company Information

Stratebi (www.stratebi.com/en/) are the main experts in Open Source Business Intelligence solutions in Spain, with a lot of client references, and their own developments.

They hold the main blog and social web about Open Source Analytics: Todobi (www.todobi.com) and Redopenbi (www.redopenbi.com)

You can check several **Online Demos** based on Open Source BI here: http://sample.stratebi.es and http://www.tablerofutbolero.com

Table of Contents

1. Introduction

This document is intended to guide you when learning how to define links over the members of a dimension or level. We assume that you have already reviewed the "Introducing STPivot" and "STPivot Installation Guide" documents, successfully deploying STPivot in your Pentaho BI Server of choice; you are also familiar with action sequence definitions and feel yourself ready to try this advanced feature. Previous understanding on how <jp:clickable> tag functions is also desirable.

We'll be using the action sequence "Clickables Sample" as shown in the following picture.
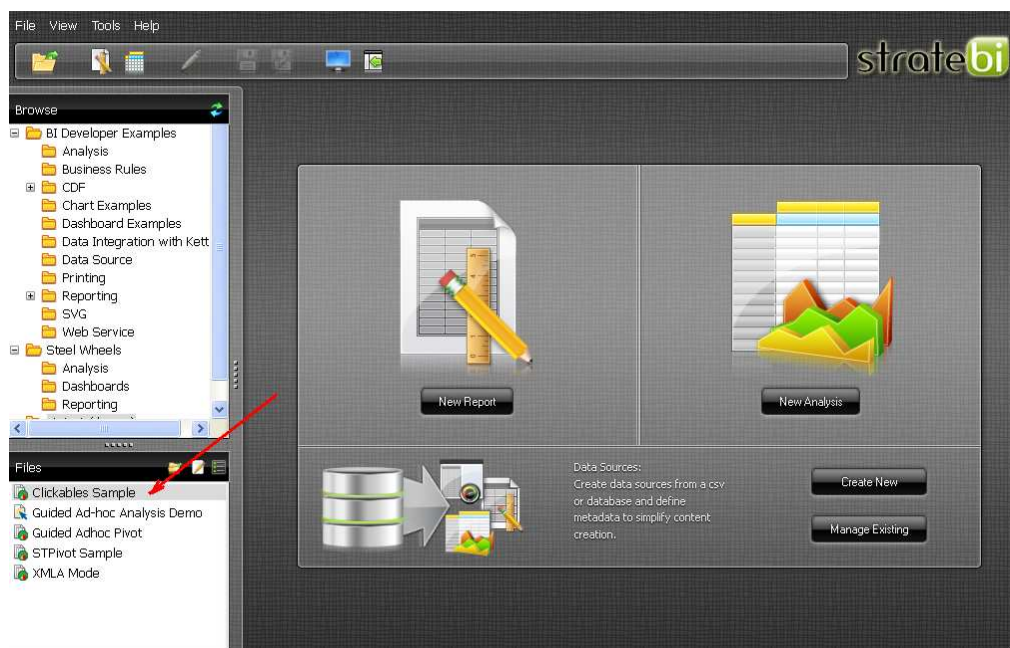
2. About Clickables in JPivot

JPivot defines a "clickable" tag that makes all members of a dimension or level clickable (i.e. generates an anchor), where the generated URL contains the unique name of the member.

This tag must be nested either inside a table tag or inside a query tag. Inside the table tag the clickable is available for all queries, inside the query tag its only available for that query. Borrowed from JPivot's documentation (link), following table lists all available attributes:

Attribute	Description
urlPattern	The pattern to construct the url. If sessionParam attribute is NOT present, then "{0}" will be replaced with the unique name of the member. If urlPattern starts with "/" its relative to the application context, otherwise its absolute. Do not use in conjunction with session parameters because the browser back button does not work as expected. Use "page" attribute instead.
page	Name of the target page, must start with "/". If not set the current page is re-displayed.
uniqueName	Unique name of a dimension, hierarchy or level, whose members will be clickable . It is required always.
menuLabel	If there are multiple clickables specified, they will be displayed in a popup menu, this text will be the label for the menu entry
sessionParam	Name of the parameter, that will be placed into the com.tonbeller.jpivot.param.SessionParamPool
propertyName	If present, the value of that member property will be taken for the com.tonbeller.jpivot.param.Parameter sqlValue property.
propertyPrefix	If present, multiple SessionParam will be created, one for each member property whose name starts with propertyPrefix. The name of the SessionParam will be the name of the reminder of the member properties name after the prefix. If present, the attributes sessionParam and propertyName are not allowed.
providerClass	If present, an instance of this class will extract the SessionParam instances from the member. The class must implement com.tonbeller.jpivot.table.navi.ClickableMember.ParameterProvider. The attributes sessionParam, propertyName, propertyPrefix are not allowed in this case.

For example:

```
<jp:table ...>
    ...
    <jp:clickable urlPattern="/otherpage.jsp?param={0}" uniqueName="[Customers]"/>
    <jp:clickable page="/yetotherpage.jsp" uniqueName="[Products].[Category]" sessionParam="Category"/>
    ...
</jp:table>
```

The behavior depends on the sessionParam attribute. If its present, then the parameter value will be written into the com.tonbeller.jpivot.param.SessionParamPool before the new page is shown. If its not present, the parameter will be coded into the hyperlink and must be treated in some way by the receiving page.

The problem with classic Pivot viewer is that you must hardcode a new viewer, based on classic, to indicate what dimensions and/or levels will be clickable. That's why STPivot uses an incipient mechanism to parameterize this feature in conjunction with the action sequence.

STPivot looks for a parameter containing the list of clickables tag that must be included into the viewer. Then they are placed into a "table" tag, so remains available for further queries. In the next topic you'll see the action sequence definition behind the sample solution.

3. Action Sequence Definition

You'll find two files in the solution folder (stpivot-demos) containing the necessary code for this sample. First of them (demo_clickables.analysisview.properties) is used only for internationalization reasons, and don't need an explanation. The entire action sequence definition is in file "demo_clickables.analysisview.xaction", pasted here for further analysis.

```xml
1.    <?xml version="1.0" encoding="UTF-8"?>
2.    <action-sequence>
3.
4.      <name>demo_clickables.analysisview.xaction</name>
5.      <title>%title</title>
6.      <version>1</version>
7.      <logging-level>ERROR</logging-level>
8.      <documentation>
9.        <author></author>
10.       <help/>
11.       <result-type>report</result-type>
12.       <description>%description</description>
13.       <icon></icon>
14.     </documentation>
15.
16.     <inputs>
17.       <mode type="string">
18.         <default-value/>
19.         <sources>
20.           <request>mode</request>
21.         </sources>
22.       </mode>
23.       <!--
24.       In order to define clickable members, stpivot looks for an input param named "clickables".
25.       It is formmed by the attributes of <jp:clickable> tags, delimited by semi-colons (";").
26.
27.       Current PivotViewComponent in Pentaho doesn't allow such param in its definition.
28.       That's why we have to define it as input parameter with the default value
29.       we want to send, and then as output param from xaction.
30.       -->
31.       <!-- BEGIN code to enable clickables in stpivot -->
32.       <clickables type="string">
33.         <default-value>urlPattern="http://www.google.com/search?q={0}" uniqueName="[Time]" menuLabel="Search with Google";
urlPattern="http://www.google.com/search?q={0}" uniqueName="[Markets]" menuLabel="Search with Google"; urlPattern="http://search.yahoo.com/search?p={0}" uniqueName="[Markets]"
menuLabel="Search with Yahoo"</default-value>
34.         <sources>
35.           <request>clickables</request>
36.         </sources>
37.       </clickables>
38.       <!-- END code to enable clickables in stpivot -->
39.     </inputs>
40.
41.     <outputs>
42.       <model type="string"/>
43.       <connection type="string"/>
44.       <mdx type="string"/>
45.       <options type="list"/><!-- Ignored in STPivot -->
46.       <title type="string"/>
47.       <url type="string">
48.         <destinations>
49.           <response>redirect</response>
50.         </destinations>
51.       </url>
52.       <charttype type="string"/>
53.       <chartlocation type="string"/>
54.       <chartheight type="string"/>
55.       <chartwidth type="string"/>
56.       <showgrid type="string"/>
57.       <!-- BEGIN code to enable clickables in stpivot -->
58.       <clickables type="string"/>
59.       <!-- END code to enable clickables in stpivot -->
60.     </outputs>
61.
62.     <resources/>
```

```xml
63.
64.                    <actions>
65.
66.                        <action-definition>
67.                            <component-name>PivotViewComponent</component-name>
68.                            <action-type>Pivot View</action-type>
69.                            <action-name>Pivot View</action-name>
70.                            <logging-level>ERROR</logging-level>
71.                            <action-inputs>
72.                                <mode type="string"/>
73.                            </action-inputs>
74.                            <action-outputs>
75.                                <model type="string"/>
76.                                <connection type="string"/>
77.                                <mdx type="string"/>
78.                                <options type="list"/><!-- Ignored by STPivot -->
79.                                <title type="string"/>
80.                                <url type="string"/>
81.                                <charttype type="string"/>
82.                                <chartlocation type="string"/>
83.                                <chartheight type="string"/>
84.                                <chartwidth type="string"/>
85.                                <showgrid type="string"/>
86.                            </action-outputs>
87.                            <component-definition>
88.                                <title>Drill Down to Pivot Table</title>
89.                                <viewer>STPivot</viewer>
90.                                <charttype>8</charttype>
91.                                <chartlocation>right</chartlocation>
92.                                <chartheight>500</chartheight>
93.                                <chartwidth>600</chartwidth>
94.                                <showgrid>true</showgrid>
95.                                <model><![CDATA[stpivot-demos/steelwheels.mondrian.xml]]></model>
96.                                <!--connection>jdbc/SampleData</connection-->
97.                                <!--query>default</query-->
98.                                <options><!-- Ignored by STPivot -->
99.                                    <personal/>
100.                                   <cube-nav/>
101.                                   <mdx-edit/>
102.                                   <sort-conf/>
103.                                   <spacer/>
104.                                   <level-style/>
105.                                   <hide-spans/>
106.                                   <properties/>
107.                                   <non-empty/>
108.                                   <swap-axes/>
109.                                   <spacer/>
110.                                   <drill-member/>
111.                                   <drill-position/>
112.                                   <drill-replace/>
113.                                   <drill-thru/>
114.                                   <spacer/>
115.                                   <chart/>
116.                                   <chart-conf/>
117.                                   <spacer/>
118.                                   <print-conf/>
119.                                   <print-pdf/>
120.                                   <spacer/>
121.                                   <excel/>
122.                                </options>
123.                                <jndi>SampleData</jndi>
124.                                <query><![CDATA[
125.                                   SELECT
126.                                       { [Markets].[All Markets].Children } ON COLUMNS,
127.                                       { [Time].[Years].Members } ON ROWS
128.                                   FROM [SteelWheelsSales]
129.                                   WHERE [Measures].[Sales]
130.                                   ]]></query>
131.                            </component-definition>
132.                        </action-definition>
133.
134.                    </actions>
135.
136.            </action-sequence>
```

You'll find this action sequence and the one used in First Steps tutorial very similar, differing only in the input and output sections.

If STPivot found a parameter named "clickables", it is assumed that contains a semicolon separated list of clickable tags (all the attributes except for container tag). No line breaks are allowed for the moment, other wise you'll receive unexpected results. Since the PivotViewComponent provided by Pentaho does not allows you to pass this parameter to STPivot, we are setting them as input parameters; until we develop a new STPivotViewComponent that fix this and other similar issues (by including this parameters in the component-definition section).

You can try now defining your own Clickable members, and give us your feedback so we can fix possible error and/or enhance current solution.