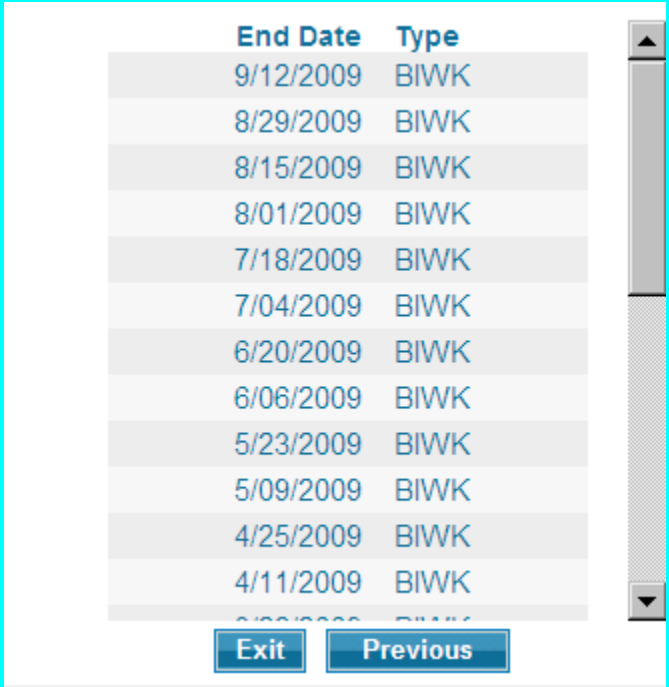


DEVELOPING A CUSTOM TABLE WIDGET

Whenever data is scattered on multiple pages on the mainframe host screen the best way is to use screen combination in order to fetch multiple screen data and display all in a single page. We did the same in one of the requirements from one of our clients; used the screen combination and rendered data as a table component, there was also one special requirement to show scrollbar next to the data and show each row of data as a hyperlink. Below screen shot presents our achieved result.



The screenshot shows a table with two columns: 'End Date' and 'Type'. The data is displayed in a list format with a vertical scrollbar on the right side. Below the table are two buttons: 'Exit' and 'Previous'.

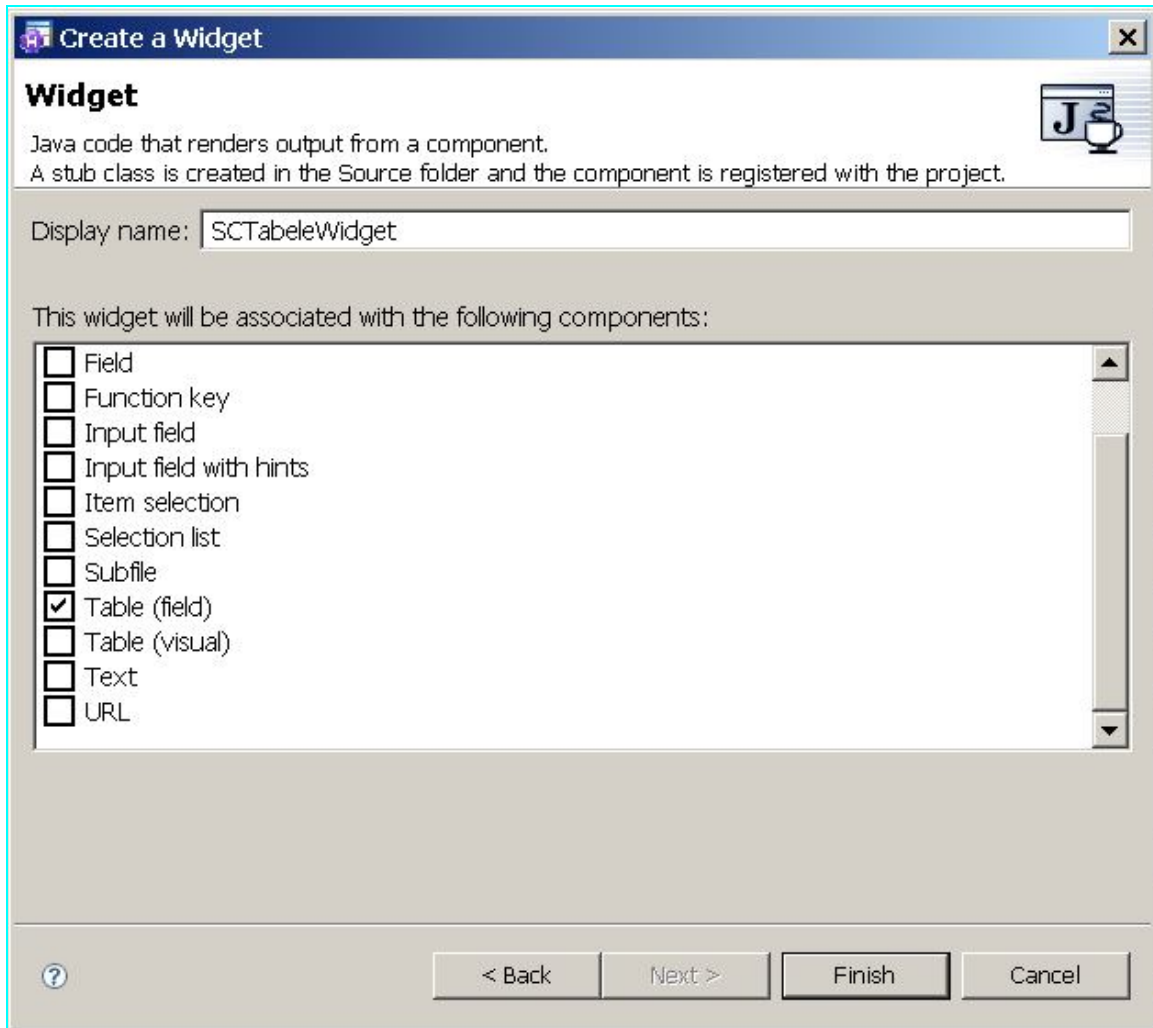
End Date	Type
9/12/2009	BIWK
8/29/2009	BIWK
8/15/2009	BIWK
8/01/2009	BIWK
7/18/2009	BIWK
7/04/2009	BIWK
6/20/2009	BIWK
6/06/2009	BIWK
5/23/2009	BIWK
5/09/2009	BIWK
4/25/2009	BIWK
4/11/2009	BIWK
3/28/2009	BIWK

[Exit](#) [Previous](#)

DEVELOPING A CUSTOM WIDGET

For meeting the above requirement we developed a custom Table widget as there wasn't any HATS widget available to meet the above requirement.

When creating the widget we associated our widget with the Table (field) widget as shown below.



Before creating the table for rendering the **headers** and **data** use the following line of code that will add scrollbar to the table.

```
buffer.append("<div style='align: center; width: 300px; height : 300px; color: black; scrollbar-  
base-color: silver; scrollbar-arrow-color: black; text-decoration: none; padding: 0px; overflow :  
auto;'>");
```

After adding the scrollbar add the following code segment that will add the static table heading with the following two Headers.

1. **End Date**
2. **Type**

```
buffer.append("<table align='center' cellspacing='0' cellpadding='0' border='0' >");  
  
    buffer.append("<tr >");  
    buffer.append("<td align='center' class=\"columnHeading\">");  
    buffer.append("End Date   Type");  
    buffer.append("</td >");  
    buffer.append("</tr >");  
buffer.append("</table >");
```

After adding the Heading table the below code segment adds the table for rendering the data. The code is explained step by step as below:

1. The following line of code is used to create the table.

```
buffer.append("<table align='center' width='240px' cellspacing='0' cellpadding='0'  
border='0' id='myTable'>");
```

2. We fetch the elements into `TableComponentElement` as demonstrated by the following code segment. We get the elements of our `cel` object in an Object `tce`. Next, we parse `tce` into a `TableCellComponentElement` object named `tcce`. This object will contain each row of data.

```
TableComponentElement tce=(TableComponentElement)elements[i];  
TableCellComponentElement tcce [][] =tce.getCells();
```




© Copyright IBM Corporation 2010
IBM Global Services
Route 100
Somers, NY 10589
U.S.A.
Produced in the United States of America
08-10
All Rights Reserved

IBM, the IBM logo, ibm.com, Lotus®, Rational®, Tivoli®, DB2® and WebSphere® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml Other company, product and service names may be trademarks or service marks of others. The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software. This document illustrates how one organization uses IBM products. Many factors have contributed to the results and benefits described; IBM does not guarantee comparable results elsewhere.