



White Paper

**Populating a DropDown in a Form
Using SQLPlus**

A Workflow Design Pattern

By Howard Treisman

Senior Technical Director

February 2007

Copyright © 2007 – Avoka Technologies ⁽¹⁾

www.avoka.com

Overview

One of the most common tasks that a process designer is required to do is to populate a DropDown field in a Form using data that is obtained from a database query.

This paper describes a simple way to achieve this, using a combination of Avoka's SQLPlus QPAC and a few lines of Javascript code in the form.

In a nutshell

The fundamental idea of this approach is to perform the query using SQLPlus, and save the results as a comma separated list into a hidden field in the form. Some simple Javascript code in the form then populates the Drop Down field.

Designing your form

- In your form, create a normal DropDown combo box. We will add code to it later.
- Add a TextField, and make it hidden (or invisible).
- Ensure that you have an XML schema defined for your form, and select the "embedded" option – this will make it easier to locate your hidden field within your Workflow.
- Bind your hidden field to an element in your schema.
- Please refer to Form Designer documentation for further information on how to perform these steps.
- Deploy your form to Form Manager, as usual.

Configuring the SQLPlus QPAC

- Insert an instance of the SQLPlus QPAC in a step prior to your User step.
- Design and test your SQL statement using the SQLPlus QPAC (or your Query Editor of choice). Please refer to SQLPlus documentation for more details.

- On the Output tab of the SQLPlus QPAC, select the data output type as "XPATH" the concatenation option as "CONCATENATE COLUMNS", and the delimiter as "," as shown in Fig 1.
- Click the Refresh button to populate the table of database columns that will be returned, or if you're not using a Direct JDBC connection, simply add the columns to match your SQL statement. (Note that in JDBC, columns indexes start at 1, NOT 0.)
- In the Destination name for the column that you're interested in, enter an XPath expression that evaluates to the element name of the hidden field within your form. (Note that the screenshot below does not show the correct Xpath expression for a form element.) This means that the result of the SQL query will be inserted as the value of your hidden form field.

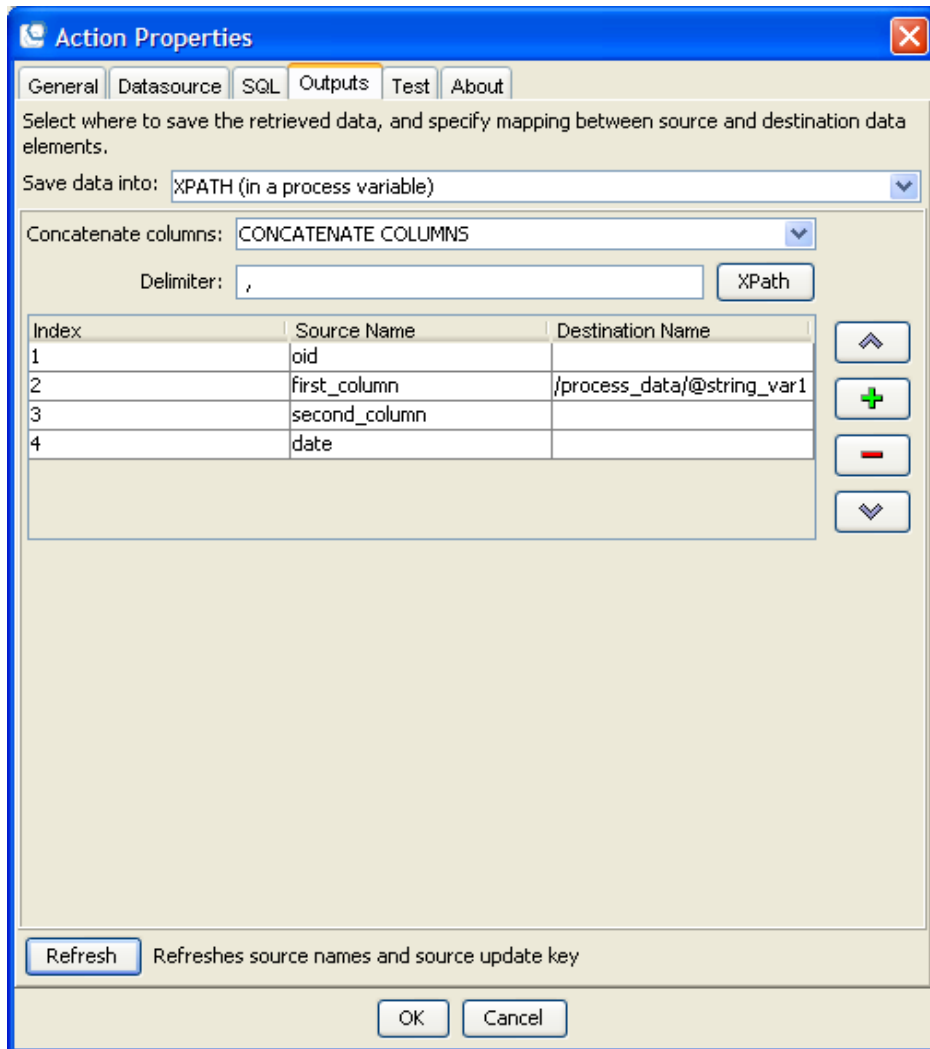


Fig 1: Output tab

Javascript in your Form

In your form, you will need to write a few lines of Javascript, similar to the following:

On the Init event of the Dropdown field:

```
var choiceList;
var choices;

// Clear drop-down list box
this.clearItems();

    choiceList = hiddenField.rawValue;

// Split the choiceList into an array of values
choices = choiceList.split(",");
var numchoices = choices.length;

for (var i = 0; i < numchoices; i=i+1) {
    this.addItem(choices[i]);
}
```

You're done!!!

Extra Information

- This technique does not work where you need to populate a DropDown for an Init Form, as the Workflow isn't invoked until after the Init Form has been submitted. In this case, please refer to a sample on the Adobe site for populating a Dropdown list using WebServices. (You can still use SQLPlus to do the actual SQL part of it.)
- Avoka also has a special QPAC that simplifies this task, if you simply want a list of options stored in a system table in the database. Please see "Lookup List" at: http://www.avoka.com/avoka/qpac_library.shtml
- If the data in your columns contains commas, then choose a different delimiter in both the QPAC and the Javascript. You can choose something that is unlikely ever to occur like "|#|".
- You can populate other types of fields within your form, such as Lists and Tables, using a similar technique. You may need several hidden field to populate multiple columns in a table.

(1) – this document may be freely copied and redistributed, as long as it is reproduced in its entirety, including the above copyright notice.

About the Author



Howard Treisman is the Technical Director of Avoka Technologies, an Adobe Enterprise Partner based in Sydney Australia.

Howard has been working with the Adobe LiveCycle Workflow product (and its predecessor) for four years, and the LiveCycle suite for the last two years. He has been instrumental in delivering several enterprise-class LiveCycle applications to customers in Australia and internationally.

In addition, Howard spear-headed Avoka's development of over 50 QPACs (re-usable workflow components), many of which have been developed in direct response to customer needs, as well as the Workflow QPAC development environment for Adobe.

Howard's hands-on experience has provided a thorough understanding of the intricacies of the LiveCycle technologies, and how to apply these technologies to solve real-world problems.