

ADF Code Corner

045. How-to base a router decision on the outcome of a method execution

ORACLE
CODE CORNER



twitter.com/adfcodecorner

Abstract:

This article describes a declarative approach to use the outcome of the ADF Business Components `getEstimatedRowCount` method in a router decision within a bounded task flow. The usecase is a bounded task flow that creates a new record if no record exist before navigating to an edit form. If a record exist then the same taskflow directly navigates to the edit page.

Author:

Frank Nimphius, Oracle Corporation
twitter.com/fnimphiu
20-FEB-2009

Oracle ADF Code Corner is a loose blog-style series of how-to documents that provide solutions to real world coding problems.

Disclaimer: All samples are provided as is with no guarantee for future upgrades or error correction. No support can be given through Oracle customer support.

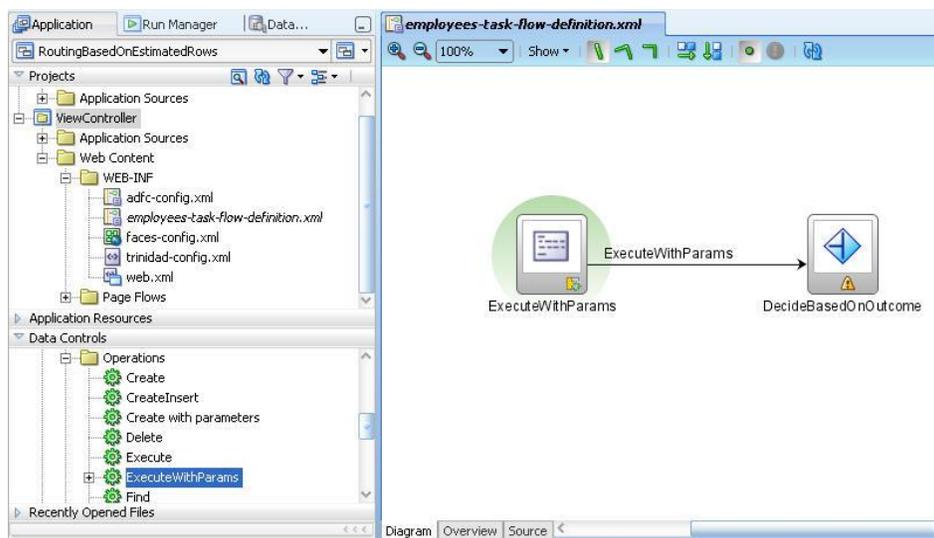
Please post questions or report problems related to the samples in this series on the OTN forum for Oracle JDeveloper: <http://forums.oracle.com/forums/forum.jspa?forumID=83>

Introduction

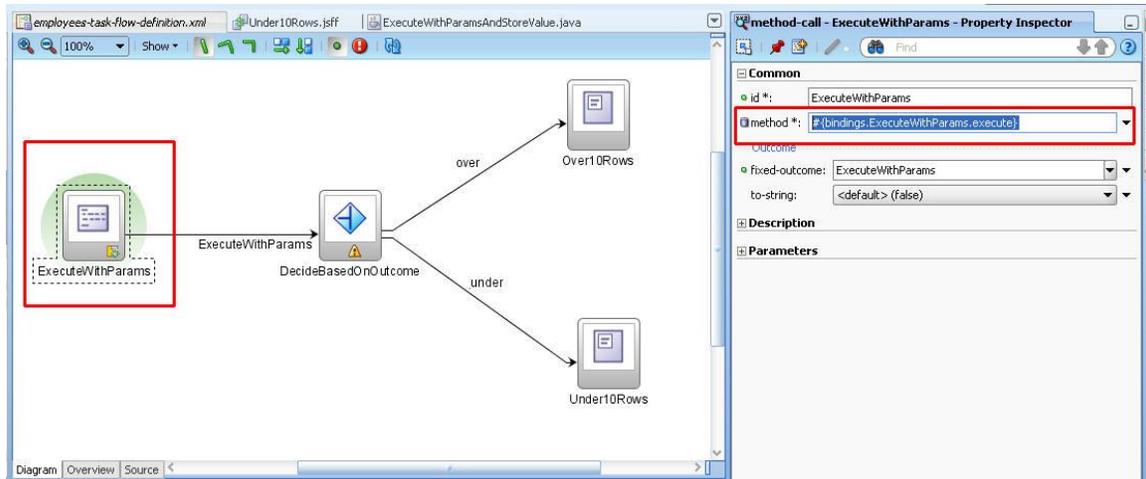
Operations exposed on a View Object in ADF Business Components can be the unbounded or bounded task flow by a drag and drop operation from the Data Control palette. The operation is added as a method activity with its own binding file, <operation-name>pageDef.xml, registered in the databindings.cpx file. While a declarative approach to execute a View Object during transit is preferable over a programmatic solution because of better readability, it doesn't allow you to base a router decision on its outcome. The reason for this is that the pageDefinition file that is created for the operation in the task flow only lives for the duration of the method activity, which means that a following router activity cannot access it. Another problem we need to solve is to create a page definition file for a method activity that points to a managed bean.

The Solution

In the example below, the ExecuteWithParams operation is dragged from the Data Control Palette and dropped as a method activity to the bounded taskflow. This creates a pagedef file for the "ExecuteWithParams". From here a navigation case points to a router activity, which will route the request based on the number of records reported by the getEstimatedRowCount method.



First step is to replace the binding reference of the method activity to a managed bean reference. Adding the operation binding to the task flow created the page definition file - the binding file. The managed bean is configured in the task flow configuration file with a scope of "None" because it has no state to hold.



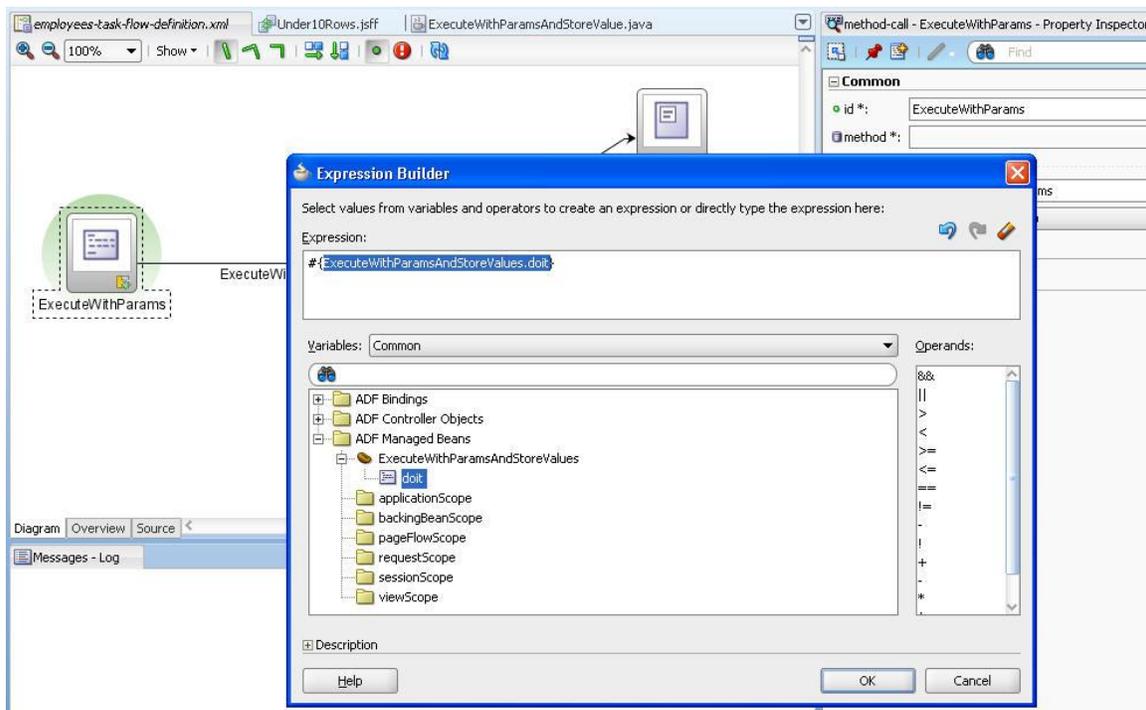
The managed bean code looks as follows:

```
import oracle.adf.model.BindingContext;
import oracle.adf.model.binding.DCBindingContainer;
import oracle.binding.OperationBinding;
import org.apache.myfaces.trinidad.context.RequestContext;

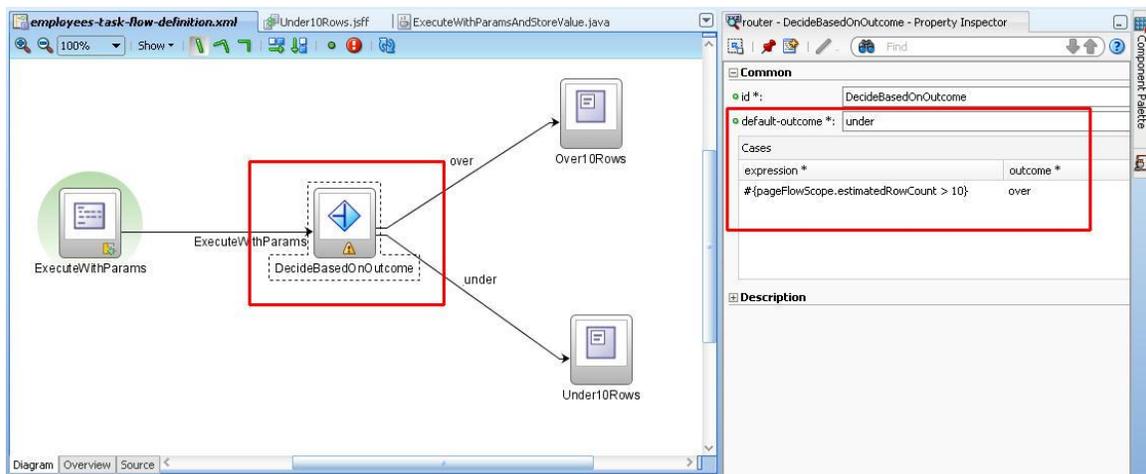
public class ExecuteWithParamsAndStoreValue {
    public ExecuteWithParamsAndStoreValue() {
    }

    public void doit() {
        BindingContext bctx = BindingContext.getCurrent();
        DCBindingContainer bindings =
            (DCBindingContainer) bctx.getCurrentBindingsEntry();
        OperationBinding oper =
            bindings.getOperationBinding("ExecuteWithParams");
        oper.execute();
        long count = bindings.findIteratorBinding("EmployeesView1Iterator")
            .getEstimatedRowCount();
        RequestContext.getCurrentInstance().getPageFlowScope()
            .put("estimatedRowCount", count);
    }
}
```

As you can see, the managed bean accesses the binding layer that now exists for the method activity to execute the operation binding. Once executed, it calls the `getEstimatedRowCount` method and stores the information in the `PageFlowScope` so the router activity can access it. You can try and add it to the `RequestScope` as well, as it should work too (though I didn't test).



The router activity now accesses the outcome of the ExecuteWithParams operation using ExpressionLanguage.



The usecase in which such a bounded task flow could be used is an input form that is used for edit and create. If the task flow input parameter produces a specific number of rows, you show a navigable form, if it is more you start with a table view for the use to scroll for the record he wants to edit..