How to import a Stored Procedure and Prepare it for use in Framework Manager

Product(s): Composite Software 3.1.1

Area of Interest: Infrastructure

Copyright

This document is maintained by the Best Practices, Product and Technology team. You can send comments, suggestions, and additions to cscogpp@ca.ibm.com .

Contents

1 Introduction

1.1 Purpose

This document outlines proven practices for importing the Composite sample data within Composite Studio.

1.2 Applicability

The techniques and product behaviours outlined in this document apply to Composite Software 3.1.1.

1.3 Exclusions and Exceptions

This import method and its required settings may change in future releases.

2 Importing the Stored Procedure

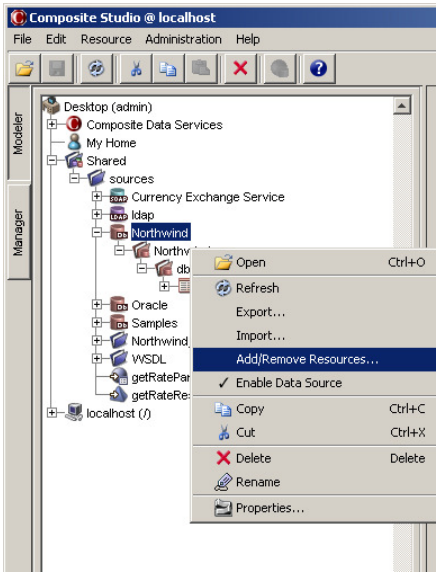
A stored procedure which returns a cursor will need to have the output parameters remapped within Composite studio. This example uses a variation of the MSSQL Northwind sample stored procedure CustOrdersOrders.

```
CREATE PROCEDURE ACustOrdersOrders @MinId int, @MaxId int
AS
SELECT OrderID, CustomerID, EmployeeID
FROM Orders where OrderId  > @MinId and OrderId < @MaxId
GO
```

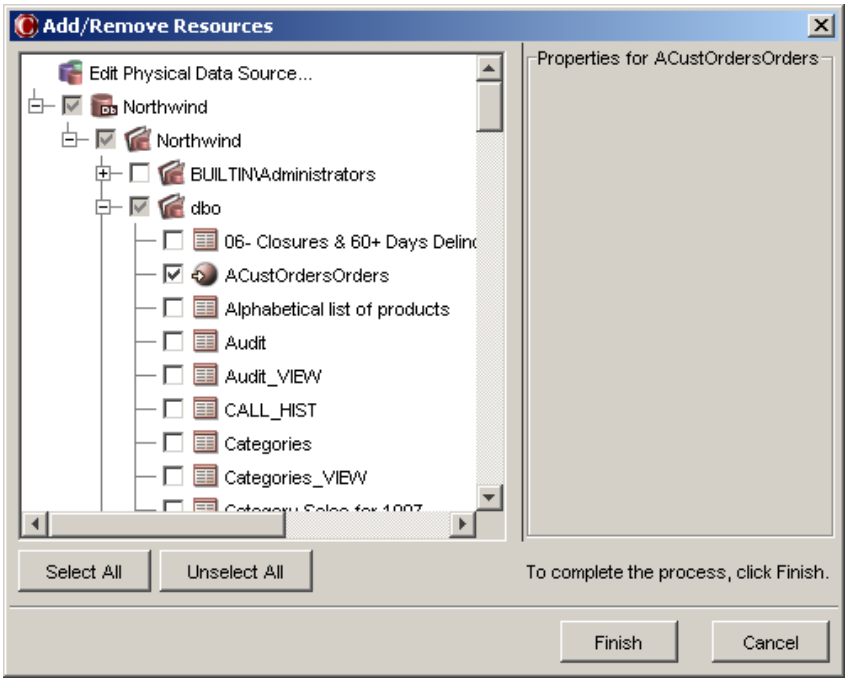
The variation takes in two int parameters and returns a cursor with the three columns. OrderID int, CustomerID nchar(5) and EmployeeID int.

This document makes the assumption that MSSQL JDBC connectivity has already been established.

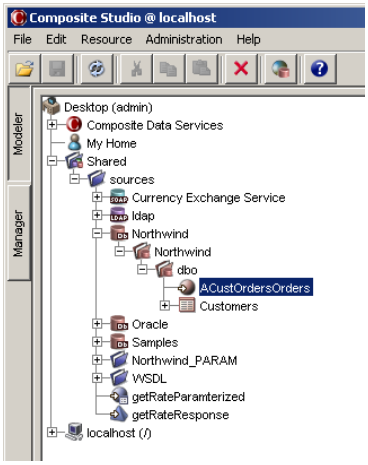
Right click on the Northwind data source under Shared\ sources and select Add Remove/ Resources …



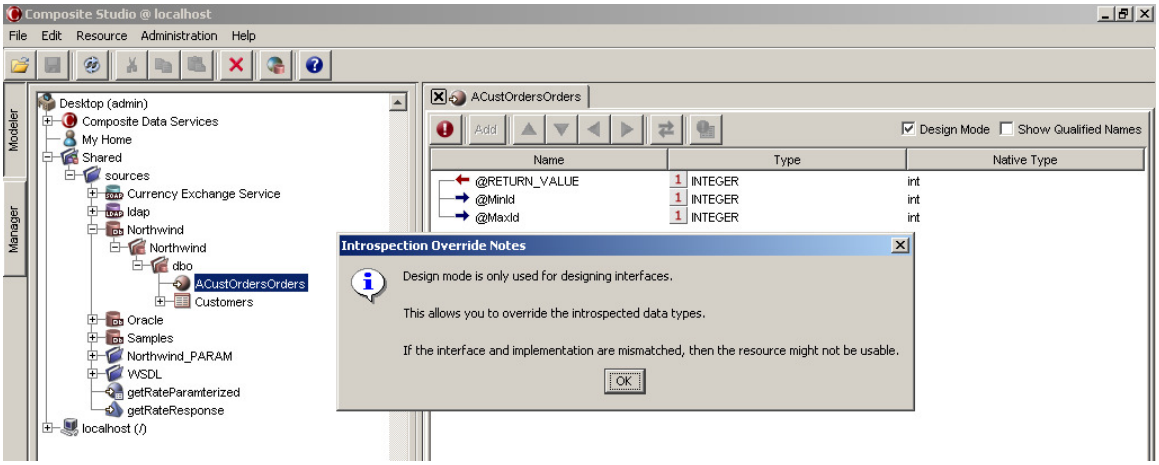
Select the ACustOrdersOrders stored procedure and click Finish.



The stored procedure should now be visible under Shared/sources within the Composite Studio.

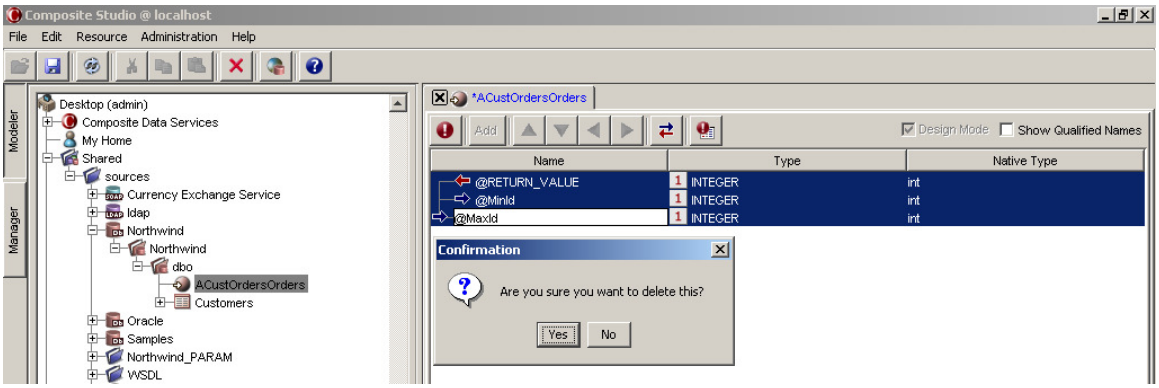


Right click on the ACustOrdersOrders stored procedure and select Open. Then in the top right of the screen check the Design Mode checkbox.



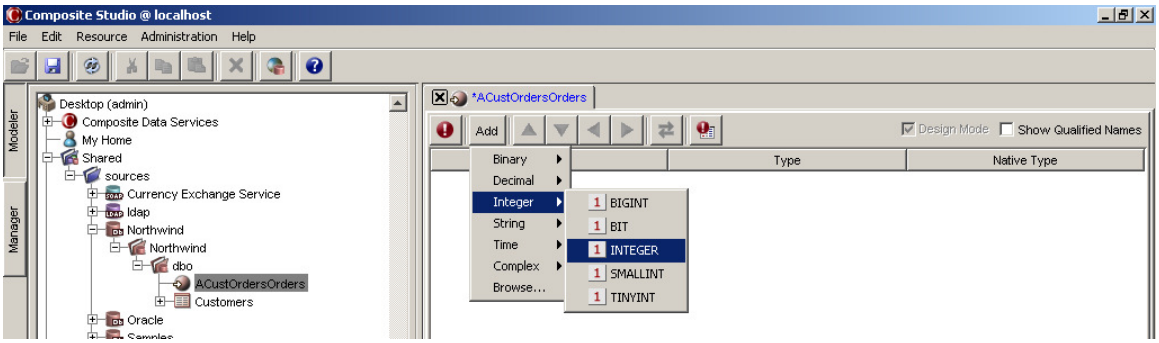
Click OK to the Introspection Override Notes message box.

Select all the input and output parameters via a shift click and delete.

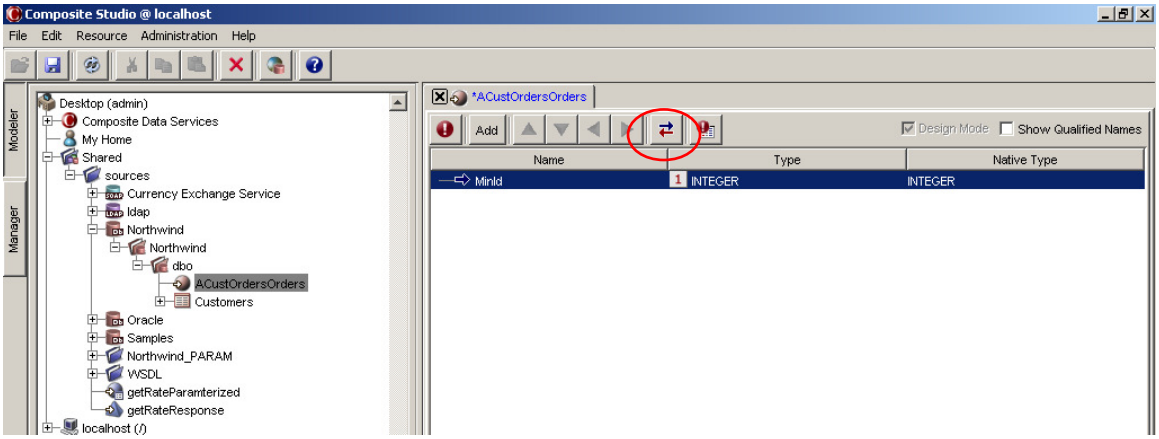


We know that this stored procedure takes in two integer input parameters. We also know that the return cursor will return two integer fields and one nchar(5) fields. All 5 objects will need to be created manually.

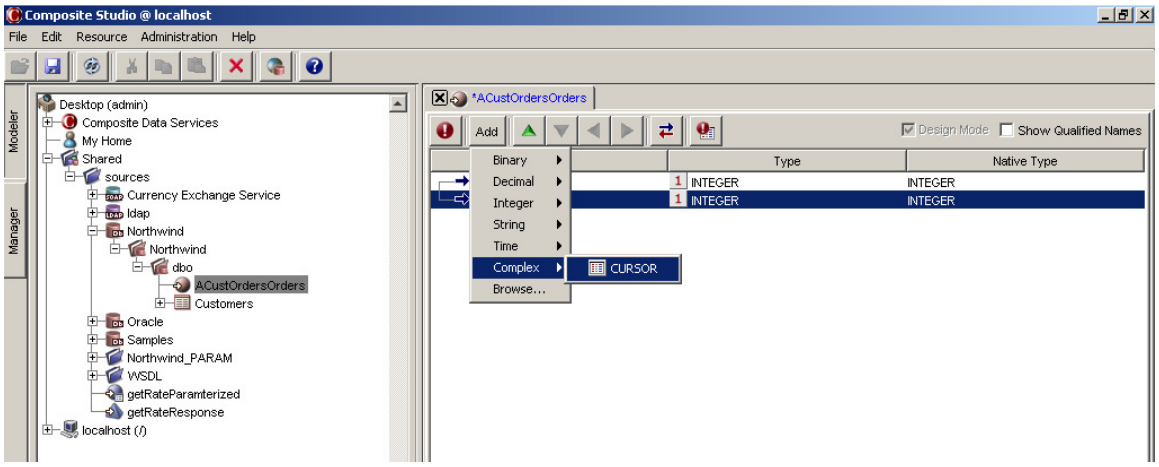
Click on the Add button and select Integer\Integer.



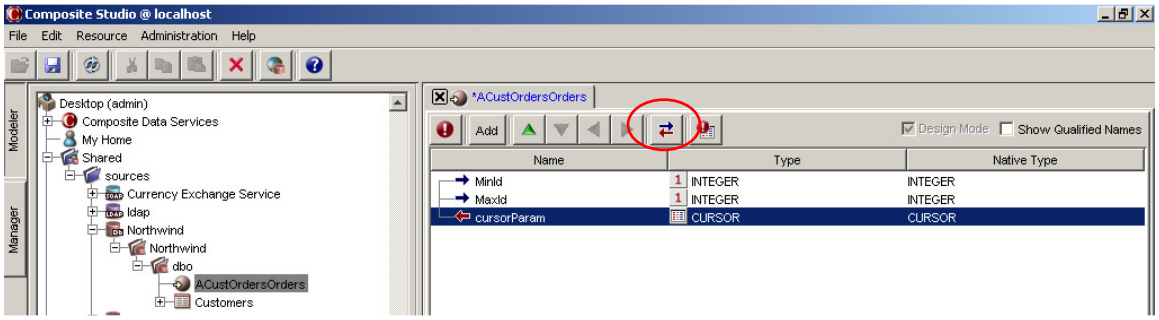
Change the name of the parameter to MinId. Then make it an in parameter by clicking on the cycle I/O direction button on the toolbar.



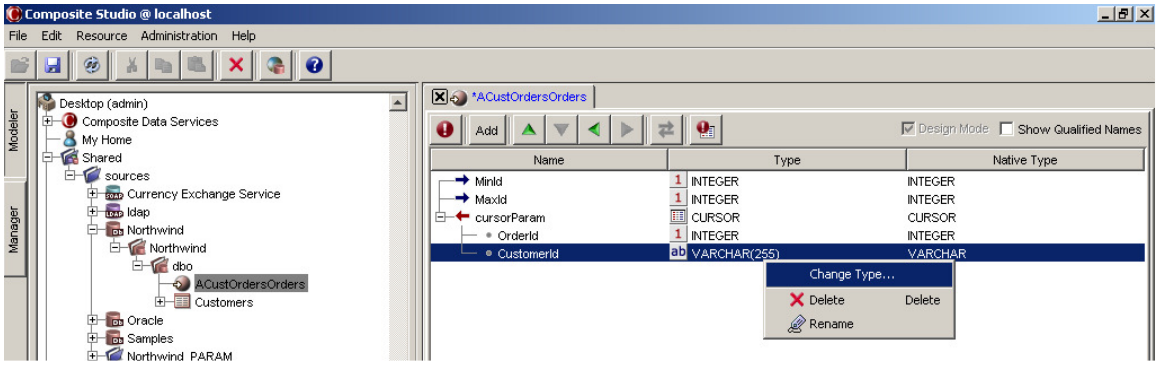
Repeat the above step for the second input parameter MaxId. To map the output cursor, click on Add and select Complex Cursor.



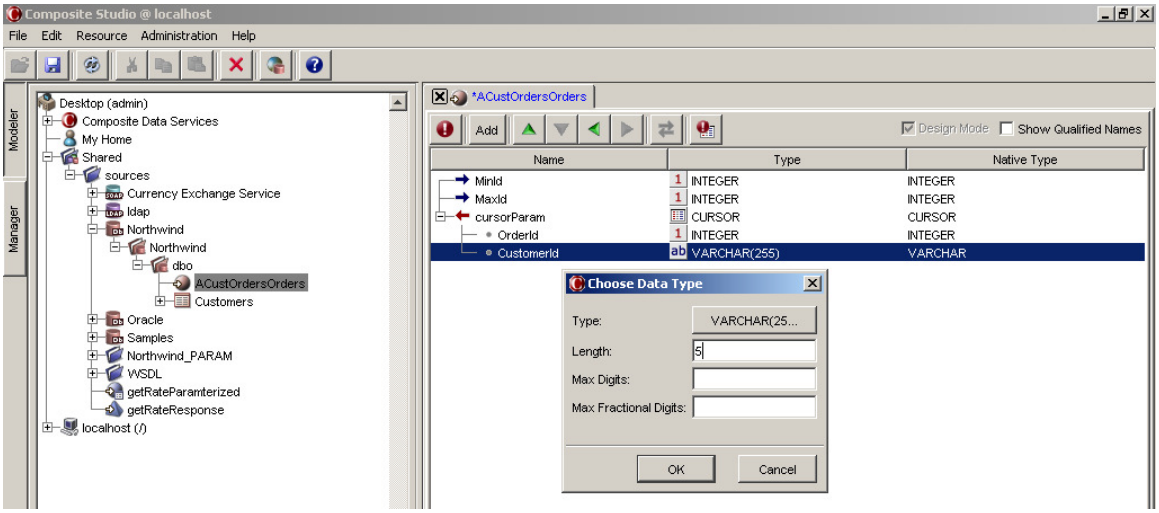
Change the cursor to an out parameter using the cycle I/O direction button on the toolbar.



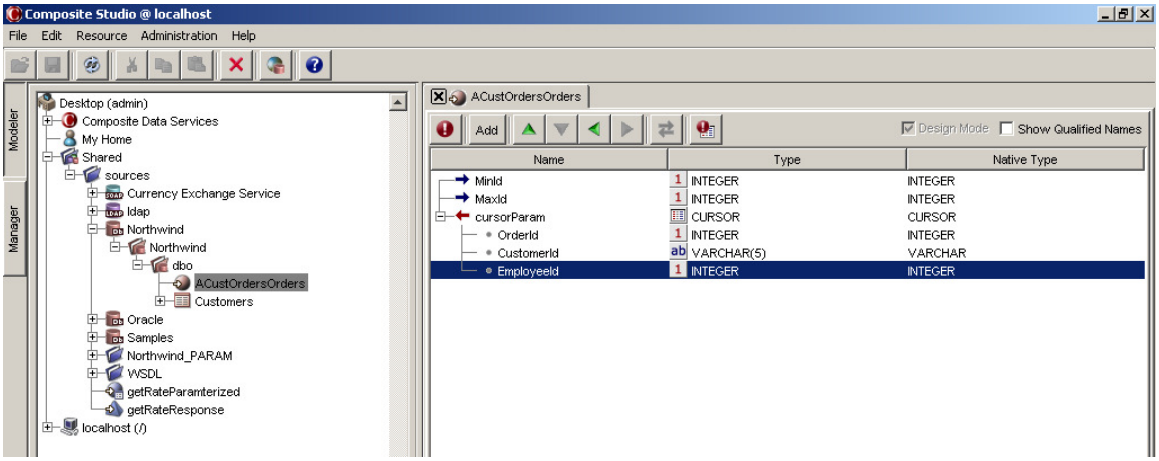
Now add the column names and the data types expected to be returned by the stored procedure. OrderID int, CustomerID nchar(5) and EmployeeId int. Notice that the size of the varchar can be specified by right clicking on the parameter and selecting Change Type…



Specify the length of the varchar and select Ok.

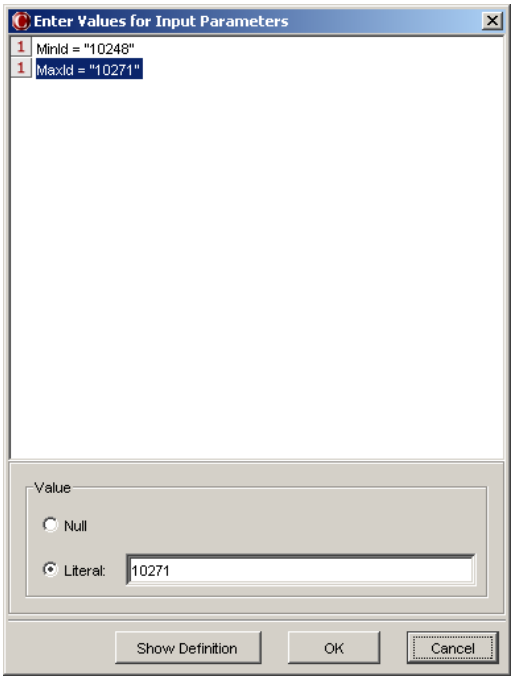


Finish the mapping procedure by adding EmployeeId. Once completed hit the save button on the tool bar.

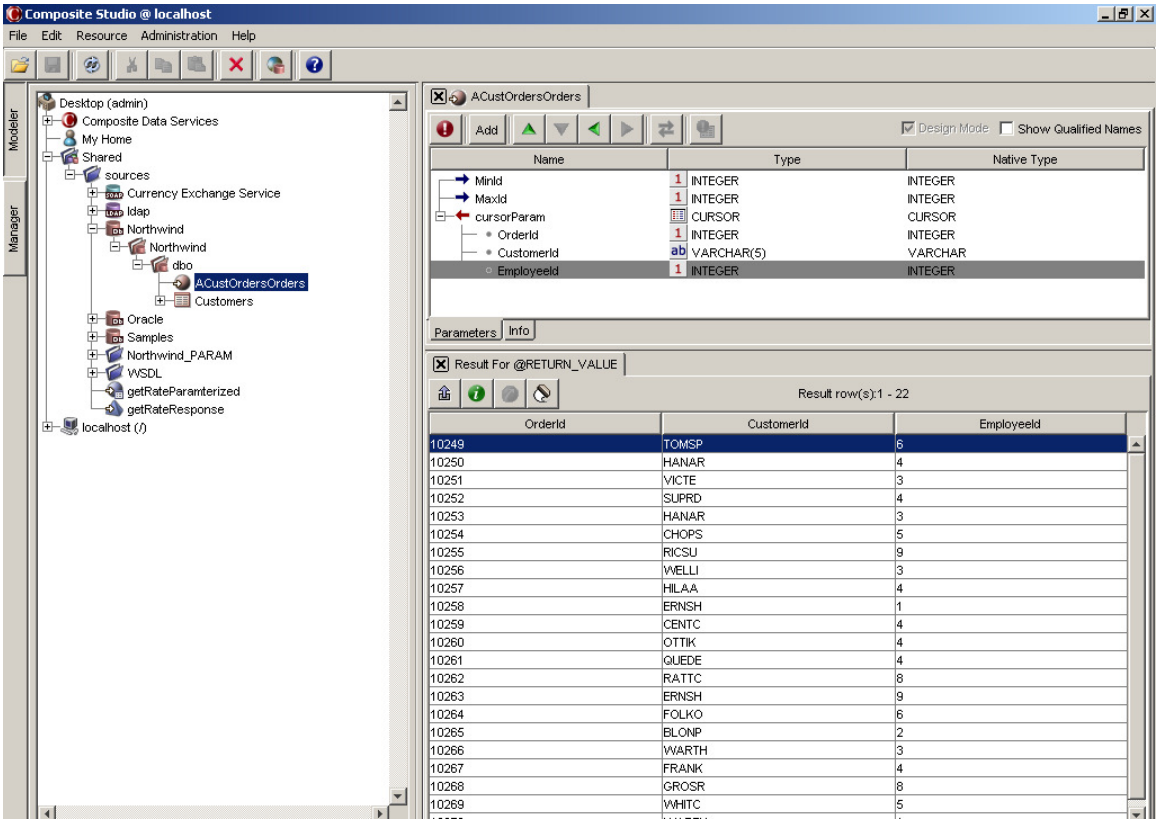


COGNOS®
AN IBM® COMPANY

The stored procedure is now ready to be tested. Click on the exclamation mark on the toolbar and enter the prompt values. For this example the Literals of 10248 and 10271 were used for MinId and MaxId.



Click OK to execute the Stored Procedure.



If the stored procedure returns a result in the bottom right window, it is ready to be published to the Composite Data Services.

3 Publishing to the Composite Data Services

Publish the object by right mouse clicking on the ACustOrdersOrders object and selecting Publish.

Select or Create the appropriate Composite Data Service Database, Catalog and Schema and click OK.

The Stored Procedure can now be utilized within Framework Manager via the Composite ODBC driver.