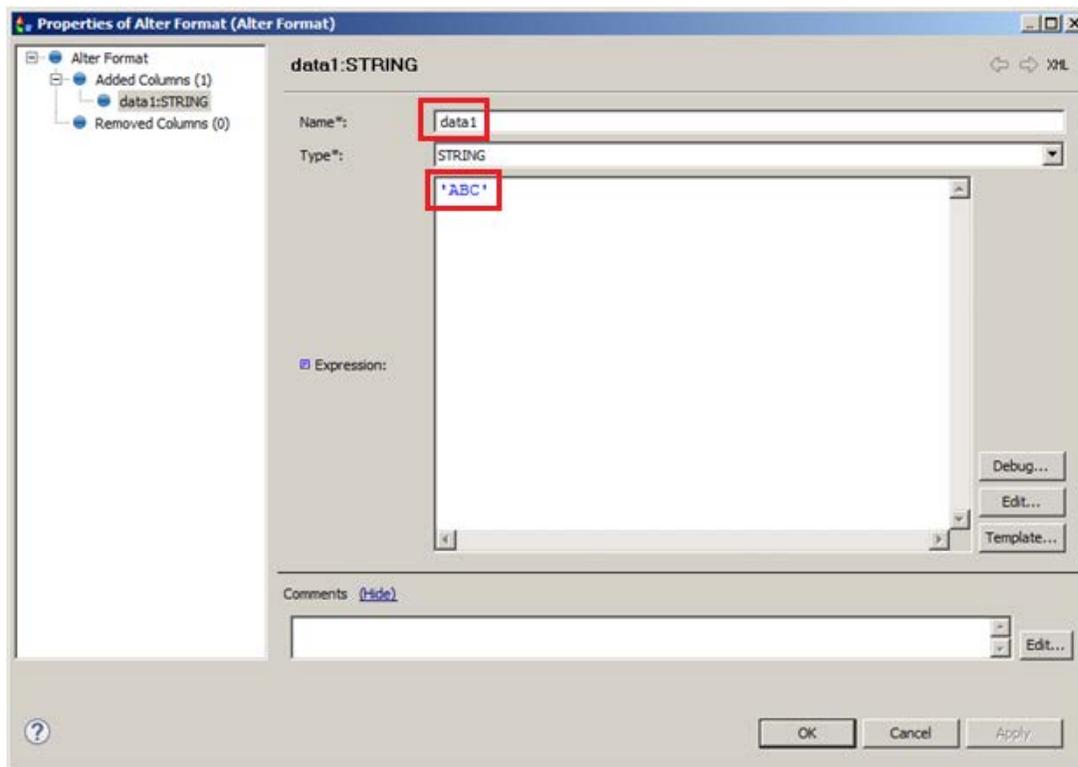


## Calling an SQL Transaction Dynamically Using iWay Data Quality Server

This topic describes how to call an SQL transaction dynamically using iWay Data Quality Server (DQS). It is a supplement to the *Parameterizing an iWay Data Quality Server Component* how-to.

**Note:** The *sql.zip* archive file that is provided with this how-to contains the preconfigured DQS component, sample input, and sample output.

A DQS solution can be configured to make an SQL statement call dynamically using the SQL Select and SQL Execute steps. In iWay DQS, these SQL steps can become dynamic by using an Alter Format step to assign a column, which can have a mapped expression, as shown in the following image.



In the Expression field, 'ABC' is the default value. However, the map name of the expression is `data1`. Once the configured iWay DQS component is called using the `runcif` shell script, or a `.bat` (batch) file, the `-params` switch can be specified to indicate the name of the mapping, and the assigned value is the required value.

### Sample Configuration

The objective of this sample configuration is to dynamically assign a value to a column in the `WHERE` clause of the SQL statement. A Text File Reader step is used first, which is required to pass input data to the

component. The Integration Step cannot be used to map a value dynamically, since it will not pass data to the components. Therefore, either a Text File Reader step or Random Record Generator step must be used.

The Alter Format step allows for an application data column to be specified where the expression can be mapped to a value that is dynamically passed in through the `runcif.sh` or `runcif.cmd` scripts. Subsequently, the SQL Select step will reference the value of the mapped expression using the following SQL statement:

```
select COLUMN1 FROM SALREPRO WHERE COLUMN1 = ${data1}
```

As mentioned earlier, an SQL Execute step can also be used in combination with the Alter Format step to map the value.

The above SQL statement contains the name of the map, which is `data1`. The `runcif.bat` script is called (see [Sample runcif.bat Script](#)), which dynamically passes a parameter with the name of the map, and the corresponding value of the input file that is read by the File Reader step.

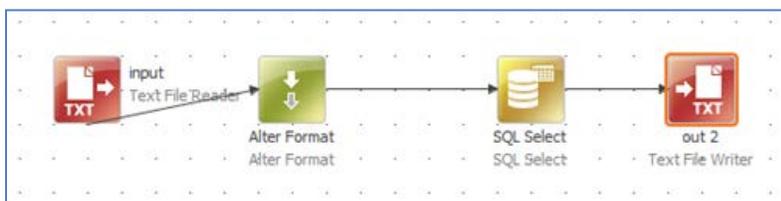
**Note:** This data is not used in the application as described earlier, but the step is required to initialize the mappings.

The script passes in a second parameter that specifies the name of the map for the expression (`data1`), and the corresponding value which is bound to the SQL statement. In the sample `runcif.bat` script (see [Sample runcif.bat Script](#)), the value `'SAL_HOBAICA'` is passed in, and bound to the SQL statement. As a result, the statement resolves to the following:

```
SELECT COLUMN1 FROM SALREPRO WHERE COLUMN1='SAL_HOBAICA'
```

The Text File Writer step emits the response of the SQL statement, which contains the data seen in the sample output (see [Output of the DQS Component](#)).

The following image shows a completed flow in the iWay DQS workbench containing the steps that are mentioned in this sample configuration.



## Sample runcif.bat Script

The following shows the structure of a sample `runcif.bat` script for reference purposes.

```
runcif.bat -params.fileName=c:\test.txt -params.data1='SAL_HOBAICA' -  
runtimeConfig ..\..\workspace\test\runtimeCfg.xml  
..\..\workspace\test\MAPPINGTESTORIG
```

INAL.comp

## Output of the DQS Component

The Text File Writer step emits the response of the SQL statement, which contains the data shown in the sample output below.

```
"data1"  
"SAL_HOBAICA"
```