Excel Generator

With the **Excel Generator** you can create Excel documents out of complex structured data.

The Excel Generator uses the HSSF and XSSF component of Apache POI.

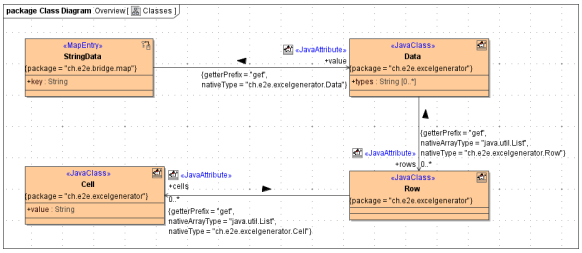
Example File (Builder project ExcelGenerator):

<your example path>\Libraries\ExcelGenerator.zip

On this Page:

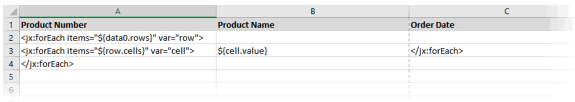
Data Model of the Excel Generator

This adapter uses a template file (Excel 97-2003 or Excel 2007 format) to create an Excel document. The data are specified by a map. The key values are used to reference their corresponding **Data** structure.



Procedure for Using the Excel Generator

1. The adapter uses jXLS for configuring and building the resulting Excel documents. Create an Excel template file containing your jXLS.



	A	B	C
1	Product Number	Product Name	Order Date
2	<jx:forEach items="${data0.rows}" var="row">		
3	<jx:forEach items="${row.cells}" var="cell">	${cell.value}	</jx:forEach>
4	</jx:forEach>		
5			
6			

   To get a deeper understanding of the syntax used in the XLS templates, refer to the jXLS documentation.
2. Create the needed objects in an action script in your UML model and provide the data.

Action Script

```
create local c0 using cellReference;
create local c1 using cellReference;
create local c2 using cellReference;
set c0.value = "00001";
set c1.value = "AF-1300";
set c2.value = currentDateTime().convertToString();

create local array cells using cellReference;
set cells[0] = c0;
set cells[1] = c1;
set cells[2] = c2;

create local r using rowReference;
set r.cells = cells;

create local array rows using rowReference;
set rows[0] = r;
set rows[1] = r;

create local array types using typeOf("");
//set types[0] = "Text";
//set types[1] = "Text";
set types[2] = "Date";

create local d using dataReference;
set d.types = types;
set d.rows = rows;

create data;
create local x using stringDataReference;
set x.key = "data0";
set x.value = d;
set data[0] = x;
```

3. Call the Excel generator.

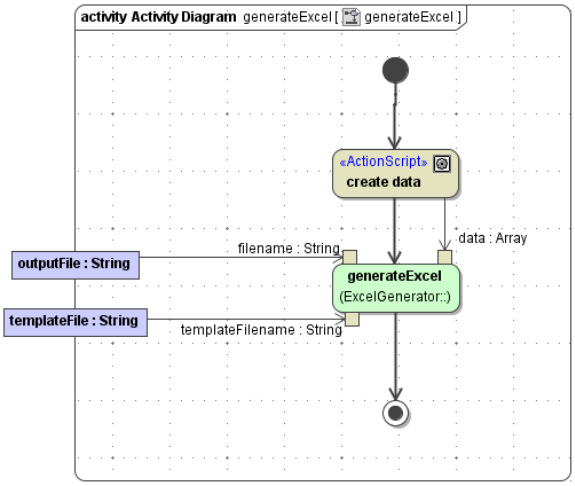
Building an Excel Document

Figure: Relationships between the Action Script and the jXLS

generateExcel(filename : String, data : StringData[], templateFilename : String)

Parameter	Direction	Description
filename	in	The filename of the Excel output file.
data	in	The map holding the data to be written to the Excel file.
templateFilename	in	The filename of the Excel template file.

The creation of an Excel document then is as simple as shown in the activity diagram below. Action **creat e data** contains the action script.



You can extract the **javadoc** folder from **excelgenerator.jar** to get additional information on the Java methods wrapped by this adapter.