

Attaching updated component related files code below. please look into this code and let me know how to update code as per my requirement.

Code of component with file name:

**tRapidSolrInsert1\_Test\_java.xml:--**

```
<?xml version="1.0" encoding="UTF-8"?>

<COMPONENT>
<HEADER AUTHOR="THE DIGITAL GROUP" COMPATIBILITY="ALL"
DATA_AUTO_PROPAGATE="false" PLATFORM="ALL" RELEASE_DATE="20150903A"
SERIAL="" STARTABLE="false" STATUS="ALPHA" VERSION="1.0">
<SIGNATURE />
</HEADER>
<FAMILIES>
<FAMILY>RapidSolrComponent</FAMILY>
</FAMILIES>
<DOCUMENTATION>
<URL />
</DOCUMENTATION>
<CONNECTORS>
<CONNECTOR CTYPE="FLOW" MIN_INPUT="1" MAX_INPUT="1" MAX_OUTPUT="1" />
<CONNECTOR CTYPE="ITERATE" MAX_OUTPUT="1" MAX_INPUT="1" />
<CONNECTOR CTYPE="SUBJOB_OK" MAX_INPUT="1" />
<CONNECTOR CTYPE="SUBJOB_ERROR" MAX_INPUT="1" />
<CONNECTOR CTYPE="COMPONENT_OK" />
<CONNECTOR CTYPE="COMPONENT_ERROR" />
<CONNECTOR CTYPE="RUN_IF" />
</CONNECTORS>
<PARAMETERS>
<PARAMETER NAME="SOLR_CORE_URL" FIELD="TEXT" NUM_ROW="1" REQUIRED="true">
<DEFAULT>"http://localhost:8983/solr/Core1"</DEFAULT>
</PARAMETER>
<PARAMETER NAME="SCHEMA" FIELD="SCHEMA_TYPE" NUM_ROW="2" REQUIRED="true">
</PARAMETER>
<PARAMETER NAME="QUEUE_SIZE" FIELD="TEXT" NUM_ROW="3"
REQUIRED="false">
<DEFAULT>100</DEFAULT>
</PARAMETER>

<PARAMETER NAME="THREAD_COUNT" FIELD="TEXT" NUM_ROW="4"
REQUIRED="false">
<DEFAULT>3</DEFAULT>
</PARAMETER>
<PARAMETER NAME="SOLR_ACTION" FIELD="CLOSED_LIST" NUM_ROW="5">
<ITEMS DEFAULT="INSERT">
<ITEM NAME="DELETE" VALUE="delete" />
```

```
<ITEM NAME="INSERT" VALUE="insert" />
<ITEM NAME="CLEAR_INSERT" VALUE="clear_insert" />
</ITEMS>
</PARAMETER>
```

```
<PARAMETER NAME="BATCH_SIZE" FIELD="TEXT" NUM_ROW="6"
REQUIRED="true">
<DEFAULT>0</DEFAULT>
</PARAMETER>
```

```
<PARAMETER NAME="JOBID" FIELD="TEXT" NUM_ROW="7"
REQUIRED="false">
<DEFAULT>0</DEFAULT>
</PARAMETER>
```

```
<PARAMETER NAME="SOURCE" FIELD="TEXT" NUM_ROW="8"
REQUIRED="false">
<DEFAULT>0</DEFAULT>
</PARAMETER>
```

```
<PARAMETER NAME="BATCH_COUNT" FIELD="TEXT" NUM_ROW="9"
REQUIRED="false">
<DEFAULT>0</DEFAULT>
</PARAMETER>
```

```
<PARAMETER NAME="BATCH_ID" FIELD="TEXT" NUM_ROW="9"
REQUIRED="false">
<DEFAULT>0</DEFAULT>
</PARAMETER>
```

```
</PARAMETERS>
```

```
<CODEGENERATION>
```

```
<IMPORTS>
```

```
<IMPORT MODULE="commons-codec-1.4.jar" NAME="commons-codec-1.4" REQUIRED="true"/>
<IMPORT MODULE="commons-httpclient-3.1.jar" NAME="commons-httpclient-3.1" REQUIRED="true"/>
<IMPORT MODULE="commons-io-2.4.jar" NAME="commons-io-2.4" REQUIRED="true"/>
<IMPORT MODULE="commons-logging-1.1.1.jar" NAME="commons-logging-1.1.1" REQUIRED="true"/>
<IMPORT MODULE="httpclient-4.4.1.jar" NAME="httpclient-4.4.1" REQUIRED="true"/>
<IMPORT MODULE="httpcore-4.4.1.jar" NAME="httpcore-4.4.1" REQUIRED="true"/>
<IMPORT MODULE="httpmime-4.4.1.jar" NAME="httpmime-4.4.1" REQUIRED="true"/>
<IMPORT MODULE="noggit-0.6.jar" NAME="noggit-0.6" REQUIRED="true"/>
<IMPORT MODULE="slf4j-api-1.5.5.jar" NAME="slf4j-api-1.5.5" REQUIRED="true"/>
<IMPORT MODULE="slf4j-jdk14-1.5.5.jar" NAME="slf4j-jdk14-1.5.5" REQUIRED="true"/>
<IMPORT MODULE="solr-solrj-5.2.1.jar" NAME="solr-solrj-5.2.1" REQUIRED="true"/>
<IMPORT MODULE="stax2-api-3.1.4.jar" NAME="stax2-api-3.1.4" REQUIRED="true"/>
```

```
</IMPORTS>
```

```
</CODEGENERATION>
```

```
<RETURNS>
```

```
<RETURN AVAILABILITY="AFTER" NAME="NB_LINE" TYPE="id_Integer" />
```

```
</RETURNS>
```

```
</COMPONENT>
```

**tRapidSolrInsert1\_Test\_messages.properties:--**

SOLR\_CORE\_URL.NAME=Solr Core URL  
QUEUE\_SIZE.NAME=Queue Size  
THREAD\_COUNT.NAME=Thread Count  
LONG\_NAME=Solr component to Index or Delete documents using SolrJ 5.2.1  
NB\_LINE.NAME=Number of lines  
SCHEMA.NAME=Schema  
JOBID.NAME=Jobid  
SOURCE.NAME=Source  
BATCH\_COUNT.NAME=Batch Count  
BATCH\_ID.NAME=Batch\_ID

SOLR\_ACTION.ITEM.DELETE=Delete  
SOLR\_ACTION.ITEM.INSERT=Insert  
SOLR\_ACTION.ITEM.CLEAR\_INSERT=Delete all before insert  
SOLR\_ACTION.NAME=Actions  
BATCH\_SIZE.NAME=Batch size (0 means no limit)

**tRapidSolrInsert1\_Test\_begin.javajet:--**

```
<%@ jet
imports="
org.talend.core.model.process.INode
org.talend.core.model.process.ElementParameterParser
org.talend.core.model.metadata.IMetadataTable
org.talend.core.model.metadata.IMetadataColumn
org.talend.core.model.process.IConnection
org.talend.core.model.process.IConnectionCategory
org.talend.designer.codegen.config.CodeGeneratorArgument
org.talend.core.model.metadata.types.JavaTypesManager
org.talend.core.model.metadata.types.JavaType
java.util.List
java.util.Map
"
%>
<%
CodeGeneratorArgument codeGenArgument = (CodeGeneratorArgument) argument;
INode node = (INode)codeGenArgument.getArgument();
String cid = node.getUniqueName();
String solrCoreUrl = ElementParameterParser.getValue(node, "__SOLR_CORE_URL__");
String queueSize = ElementParameterParser.getValue(node, "__QUEUE_SIZE__");
String threadCount = ElementParameterParser.getValue(node, "__THREAD_COUNT__");
String solrAction = ElementParameterParser.getValue(node, "__SOLR_ACTION__");
```

```

String batchSize = ElementParameterParser.getValue(node,"__BATCH_SIZE__");
String jobId = ElementParameterParser.getValue(node,"__JOBID__");
String source = ElementParameterParser.getValue(node,"__SOURCE__");
String batchcount = ElementParameterParser.getValue(node,"__BATCH_COUNT__");
String batchid = ElementParameterParser.getValue(node,"__BATCH_ID__");
int queueSize_i = Integer.parseInt(queueSize);
int threadCount_i = Integer.parseInt(threadCount);
int batchSize_i = Integer.parseInt(batchSize);

%>

org.apache.solr.client.solrj.impl.ConcurrentUpdateSolrClient solrServer_<%=cid%> = null;
solrServer_<%=cid%> = new org.apache.solr.client.solrj.impl.ConcurrentUpdateSolrClient(<%=solrCoreUrl%>, <%=queueSize_i%>, <%=threadCount_i%>);

int nb_line_<%=cid%> = 0;

java.util.List<org.apache.solr.common.SolrInputDocument> docs_to_insert_<%=cid%> = null;
java.util.List<String> docs_to_del_<%=cid%> = null;

<%
if(solrAction.equals("clear_insert") || solrAction.equals("insert")) {

if(batchSize_i==0) {
%>
docs_to_insert_<%=cid%> = new java.util.ArrayList<org.apache.solr.common.SolrInputDocument>();
<%
} else {
%>
docs_to_insert_<%=cid%> = new java.util.ArrayList<org.apache.solr.common.SolrInputDocument>(<%=batchSize_i%>);
<%
}

} else if(solrAction.equals("delete")) {
if(batchSize_i==0) {
%>
docs_to_del_<%=cid%> = new java.util.ArrayList<String>();
<%
} else {
%>
docs_to_del_<%=cid%> = new java.util.ArrayList<String>(<%=batchSize_i%>);
<%
}
}
%>

<%
if(solrAction.equals("clear_insert")) {
%>
solrServer_<%=cid%>.deleteByQuery( ":*:*" );

```

```
<%  
}  
%>
```

### tRapidSolrInsert1 Test main.javajet:--

```
<%@ jet  
imports="  
org.talend.core.model.process.INode  
org.talend.core.model.process.ElementParameterParser  
org.talend.core.model.metadata.IMetadataTable  
org.talend.core.model.metadata.IMetadataColumn  
org.talend.core.model.process.IConnection  
org.talend.core.model.process.IConnectionCategory  
org.talend.designer.codegen.config.CodeGeneratorArgument  
org.talend.core.model.metadata.types.JavaTypesManager  
org.talend.core.model.metadata.types.JavaType  
java.util.List  
java.util.Map  
"  
%>  
<%  
CodeGeneratorArgument codeGenArgument = (CodeGeneratorArgument) argument;  
INode node = (INode)codeGenArgument.getArgument();  
String cid = node.getUniqueName();  
  
String solrAction = ElementParameterParser.getValue(node,"__SOLR_ACTION__");  
String batchSize = ElementParameterParser.getValue(node,"__BATCH_SIZE__");  
String jobId = ElementParameterParser.getValue(node,"__JOBID__");  
String source = ElementParameterParser.getValue(node,"__SOURCE__");  
int batchSize_i = Integer.parseInt(batchSize);  
//Create object here i.e. first code write here  
  
if (node.getIncomingConnections()!=null && node.getIncomingConnections().size(>0) { //1  
  
IConnection iConnection = node.getIncomingConnections().get(0);  
String iConnectionName = iConnection.getName();  
IMetadataTable preMetadata = iConnection.getMetadataTable();  
  
List<IMetadataColumn> columnList = preMetadata.getListColumns();  
  
if(solrAction.equals("clear_insert") || solrAction.equals("insert")) { //2  
%>  
org.apache.solr.common.SolrInputDocument doc_<%=cid%> = new org.apache.solr.common.SolrInputDocument();  
<%
```

```

for (int i = 0; i < columnList.size(); i++) { //3
IMetadataColumn iMetadataColumn = columnList.get(i);
String talendDataType = JavaTypesManager.getTypeToGenerate(iMetadataColumn.getTalendType(),true);

if(talendDataType.equals("List")) {
%>
if(<%=iConnectionName%>.<%=iMetadataColumn.getLabel()%> != null) {
for(Object obj : <%=iConnectionName%>.<%=iMetadataColumn.getLabel()%>) {
doc_<%=cid%>.addField( "<%=iMetadataColumn.getLabel() %>", obj);
}
}
<%
} else {
%>
doc_<%=cid%>.addField( "<%=iMetadataColumn.getLabel() %>", <%=iConnectionName%>.<%=iMetadataColumn.getLabel() %>);
<%
}
} //3
%>
docs_to_insert_<%=cid%>.add(doc_<%=cid%>);
nb_line_<%=cid%>++;
<%
} //2
else if(solrAction.equals("delete")) { //4

String primaryKeyFieldName = null;

for (int i = 0; i < columnList.size(); i++) { //3
IMetadataColumn iMetadataColumn = columnList.get(i);
if(iMetadataColumn.isKey()) {
primaryKeyFieldName=iMetadataColumn.getLabel();
break;
}
} //3

if(primaryKeyFieldName!=null) {
%>
docs_to_del_<%=cid%>.add(<%=iConnectionName%>.<%=primaryKeyFieldName%>);
nb_line_<%=cid%>++;
<%
} else {
%>
System.err.println("Primary Key missing");
<%
}

} //4

if(batchSize_i>0 ) { //5
%>
if(docs_to_insert_<%=cid%>.size() >= <%=batchSize_i%>) { //6

<%

```

```

if(solrAction.equals("clear_insert") || solrAction.equals("insert")) {
%>

solrServer_<%=cid%>.add(docs_to_insert_<%=cid%>);
//System.out.println("Inserted documents : " + docs_to_insert_<%=cid%>.size());
//Add second code in this place
<%
}
%>

docs_to_insert_<%=cid%>.clear();

//6
} else if(docs_to_del_<%=cid%>.size() >= <%=batchSize_i%>) { //7
<%
if(solrAction.equals("delete")) {
%>
solrServer_<%=cid%>.deleteById(docs_to_del_<%=cid%>);
//System.out.println("Deleted documents : " + docs_to_del_<%=cid%>.size());

<%
}
%>

docs_to_del_<%=cid%>.clear();

}

<%
} //5

} //1
%>

```

### **tRapidSolrInsert1\_Test\_end.javajet:--**

```

<%@ jet
imports="
org.talend.core.model.process.INode
org.talend.core.model.process.ElementParameterParser
org.talend.core.model.metadata.IMetadataTable
org.talend.core.model.metadata.IMetadataColumn
org.talend.core.model.process.IConnection
org.talend.core.model.process.IConnectionCategory
org.talend.designer.codegen.config.CodeGeneratorArgument
org.talend.core.model.metadata.types.JavaTypesManager
org.talend.core.model.metadata.types.JavaType

```

```
java.util.List
java.util.Map
"
%>
<%
CodeGeneratorArgument codeGenArgument = (CodeGeneratorArgument) argument;
INode node = (INode)codeGenArgument.getArgument();
String cid = node.getUniqueName();

String solrAction = ElementParameterParser.getValue(node,"__SOLR_ACTION__");

if(solrAction.equals("clear_insert") || solrAction.equals("insert")) {
%>
if(docs_to_insert_<%=cid%>.size() > 0) {
solrServer_<%=cid%>.add(docs_to_insert_<%=cid%>);
//System.out.println("Inserted documents : " + docs_to_insert_<%=cid%>.size());
}
<%
} else if(solrAction.equals("delete")) {
%>
if(docs_to_del_<%=cid%>.size() > 0) {
solrServer_<%=cid%>.deleteById(docs_to_del_<%=cid%>);
//System.out.println("Deleted documents : " + docs_to_del_<%=cid%>.size());
}
<%
}
%>
solrServer_<%=cid%>.commit();
globalMap.put("<%=cid%>_NB_LINE",nb_line_<%=cid%>);
```

Thanks,

Bhushan Pande.