We are updating an Ignition 7.9 application to Ignition 8.1. The application relies extensively on the SDK Block and Connector framework. There are a few situations where I haven't been able to figure out the proper classes to use in 8.1.

In the sections below, I'll state the objective, and show a snippet of what I did in 7.9 code. All I ask is for a similar snippet of 8.1 code to accomplish the same thing.

1 - Find the data type of a tag given the tag path. Designer context. Note that the method getTag() on the TagManager no longer exists.

ClientTagManager tmgr = context.getTagManager();

**try** {

TagPath tp = TagPathParser.*parse*(path);

Tag tag = tmgr.getTag(tp);

type = tag.getDataType();

...

}

**catch**(IOException ioe) {

...

}

2 - Find other tag properties, like is it an expression, the expression itself, the SQL if it is a query tag.

ClientTagManager tmgr = context.getTagManager();

DataType typ = **null**;

**try** {

TagPath tp = TagPathParser.*parse*(tagPath);

Tag tag = tmgr.getTag(tp);

Integer tagProp = (Integer)tag.getAttribute(

TagProp.***ExpressionType***).getValue();

typ = tag.getDataType();

}

**catch** (IOException e) {

. . .

}

3 - Create a project resource from scratch.

**final** **long** newId = context.newResourceId();

SerializableDiagram diagram = **new** SerializableDiagram();// My class

diagram.setName(pbv.getName());

diagram.setResourceId(newId);

diagram.setId(UUID.*randomUUID*());

**try**{

json = mapper.writeValueAsString(diagram);

}

**catch**(JsonProcessingException jpe) {

log.warnf("%s: Unable to serialize diagram (%s)",

***CLSS***,jpe.getMessage());

}

**byte**[] bytes = json.getBytes();

ProjectResource resource = **new** ProjectResource(newId,

BLTProperties.***MODULE\_ID***, BLTProperties.***DIAGRAM\_RESOURCE\_TYPE***,

pbv.getName(), ApplicationScope.***GATEWAY***, bytes);

resource.setParentUuid(getActiveDiagram().getId());

}

4 - Save a project resource back into the project.

sharedLock.lock();

Project diff = *context*.getProject().getEmptyCopy();

ProcessDiagramView view = **null**;

**long** resourceId = res.getResourceId();

BlockDesignableContainer tab = (BlockDesignableContainer)workspace.findDesignableContainer(resourceId);

**if**( tab!=**null** ) {

view = (ProcessDiagramView)tab.getModel();

SerializableDiagram sd = view.createSerializableRepresentation();

ObjectMapper mapper = **new** ObjectMapper();

**try**{

**byte**[] bytes = mapper.writeValueAsBytes(sd);

res.setData(bytes);

}

**catch**(JsonProcessingException jpe) {

***. . .***

}

view.setDirty(**false**);

**try** {

**if**( *context*.requestLockQuietly(res.getResourceId()) ) {

*context*.updateResource(res.getResourceId(),res.getData());

diff.putResource(res, **true**);

DTGatewayInterface.*getInstance*().

saveProject(IgnitionDesigner.*getFrame*(), diff,

**true**, "Committing ...");

**for**(ProjectResource pr:diff.getResources()) {

**if**( pr.isLocked()) pr.setLocked(**false**);

}

Project project = *context*.getProject();

project.applyDiff(diff,**false**);

project.clearAllFlags();

*context*.updateLock(res.getResourceId());

*context*.releaseLock(res.getResourceId());

}

**catch**(IllegalArgumentException iae) {

***. . .***

}

**catch**(GatewayException ge) {

***. . .***

}

sharedLock.unlock();

}

}