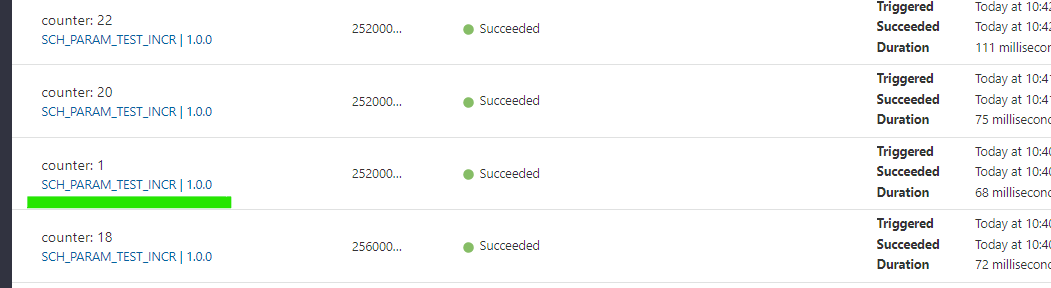
# Doubt1: Running as adhoc vs part of schedule

Created a scheduled integration which takes a number and increments it by 2 inside the OIC. This parameter (**counter**) is added as Primary tracking variable.

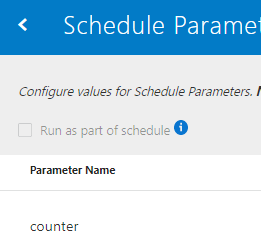
**Running an adhoc run**

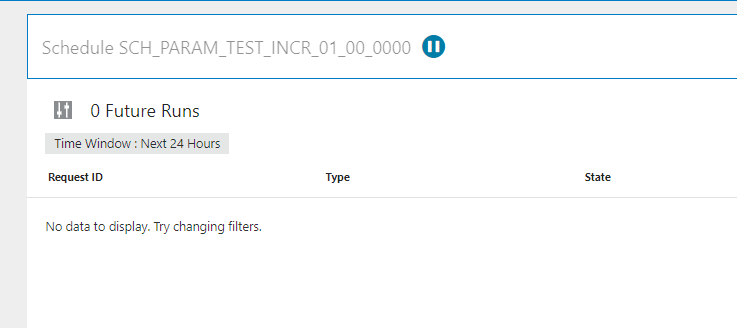
Below instance (without counter 1) was run as adhoc and not as a part of the Schedule as a result, it was not updated in subsequent runs



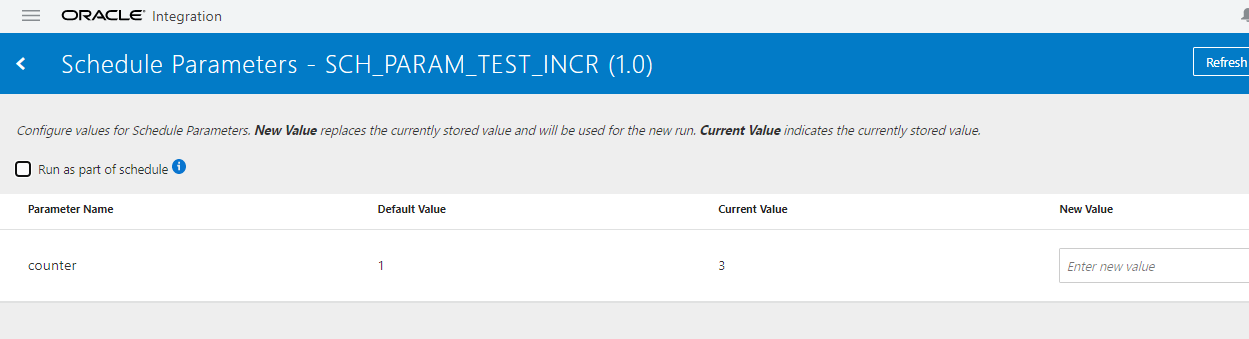
**Running as part of schedule**

You can only ran as a part of schedule if there is an active run of the schedule. In the below screenshot, we can see that the option is greyed out and that’s because the schedule is paued.

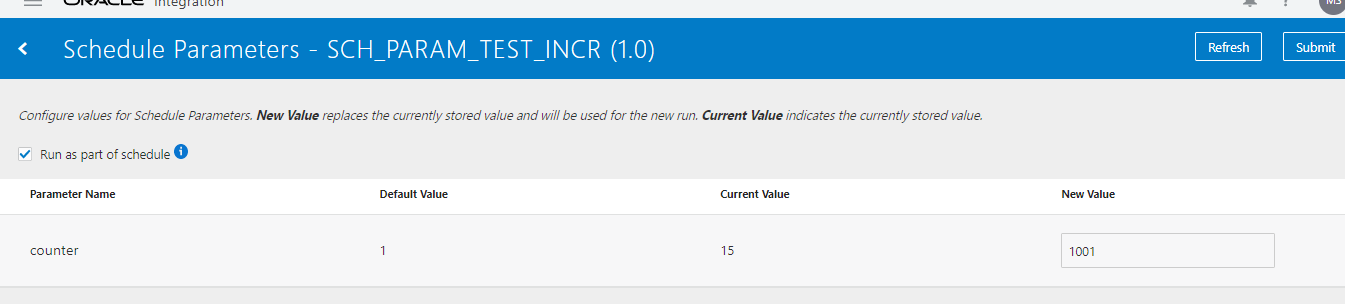


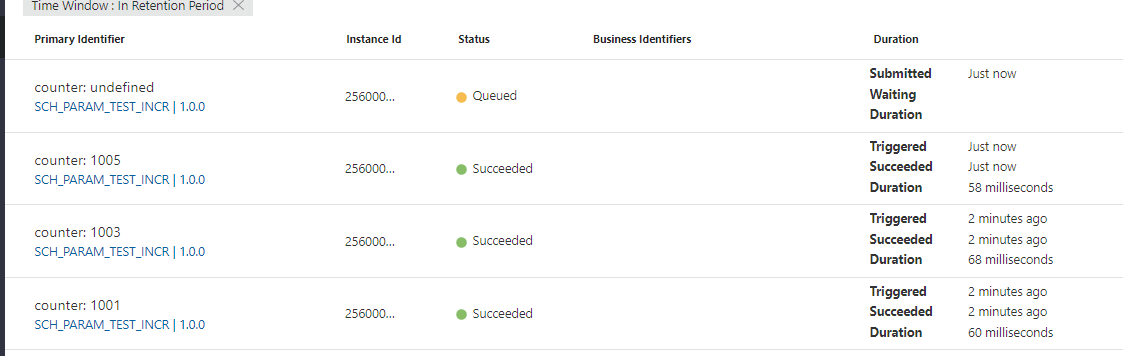


When we pass it as a part of schedule, the current value becomes the value as shown below ( the value 3 becomes 15 as soon as we check that box).



This value, can be over-ridden manually, so that subsequent runs will start the counter from 1001





# Doubt2: How to get data from SaaS from Database

## Use APEX\_WEB\_SERVICE from ATP

This is the easiest way to call as there are lots of certificates already installed

declare

l\_base64 CLOB;

l\_envelope CLOB;

l\_xml XMLTYPE;

l\_blob BLOB;

begin

l\_hostname:=https://<server.oracledemo.com>.com:443;

l\_param1 :=’param\_value’;

l\_envelope := '<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:pub="http://xmlns.oracle.com/oxp/service/PublicReportService">

<soap:Header/>

<soap:Body>

<pub:runReport>

<pub:reportRequest>

<pub:parameterNameValues>

<pub:item>

<pub:name>P\_PARAM\_NAME</pub:name>

<pub:values>

<pub:item>'||l\_param1||'</pub:item>

</pub:values>

</pub:item>

</pub:parameterNameValues>

<pub:reportAbsolutePath>'||reportPath||'</pub:reportAbsolutePath><pub:sizeOfDataChunkDownload>-1</pub:sizeOfDataChunkDownload>

</pub:reportRequest>

</pub:runReport>

</soap:Body>

</soap:Envelope>';

l\_xml := APEX\_WEB\_SERVICE.make\_request(

p\_url => l\_hostname||'/xmlpserver/services/ExternalReportWSSService',

p\_action => l\_hostname||'/xmlpserver/services/ExternalReportWSSService/runReport',

p\_version =>1.2,

p\_username => l\_username,

p\_password => l\_password,

p\_envelope => l\_envelope

);

End;

L\_xml should have the XML

Please note that the above script is not tested, but it should work with minimum changes.

## Other ways

<https://docs.oracle.com/en/cloud/paas/autonomous-database/adbsa/call-web-services.html#GUID-FFF2F734-090F-4914-88E2-8A9B2C8C18B5>

# Doubt3: Python code to convert base64 to binary\_file (pdf, zip etc)

This is helpful a lot of times in determining if the file has correct content or not.

Check the code base64ToFile.py in this repository