**BIP Job Document Information**

|  |  |
| --- | --- |
| **Title** | Oracle Reusable Components - BIP Job using schedule Service Web service |
| **Version** | 1.0 |
| **Date** | 28.01.2021 |
| **Authors** | Vinith Syam Busi |
| **User Story** |  |

**Document Review & Sign-off**

***Reviewers***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SNO** | **Name** | **Comments Received** | **Approved** | **Date** |
|  |  |  |  |  |
|  |  |  |  |  |

***Revision History***

|  |  |  |  |
| --- | --- | --- | --- |
| **Version No.** | **Date** | **Author** | **Revision Description** |
| 1.0 | 28.01.2021 | Vinith Syam Busi | First version |
| 1.1 |  |  |  |
| 1.2 |  |  |  |
| 1.3 |  |  |  |

**Other Related Documents**

| **Related Document** | **Comment** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Contents

[**1** **Overview** 2](#_Toc78447480)

[**1.1** **Configuration** 3](#_Toc78447481)

[**1.2** **Process Usage** 4](#_Toc78447482)

[**1.3** **BIP\_Schedule Service\_Caller** 5](#_Toc78447483)

[**1.4** **Web services used in Process** 6](#_Toc78447484)

### **Overview**

This document gives the overview of integration with Oracle via BIP using Schedule Service web Service.

**Details on BIP:**

BIP based data extracts are for:

* Data that are not available via SOAP/REST APIs
* Bulk data extracts

Using these reporting tools, design reports to generate data in CSV formats. Upon executing the process, the reports will be placed in UCM (Universal content Management) server. Users/Applications can download it from the UCM server.

From Integration perspective the report related extracts can be generated using below services.

* **ScheduleService**: Using this API we can schedule a Report in Oracle Cloud. We need to make following sequence of calls to fetch the data from Schedule service web service.
  1. scheduleReport
  2. getScheduledReportStatus
  3. getAllJobInstanceIDs
  4. getScheduledReportOutputInfo
  5. getDocumentData
* WSDL URL to be used:

https://{HostName}/xmlpserver/services/v2/ScheduleService?WSDL

**Boomi Implementation:**

Reusable component has been built to submit the BIP Schedule service for dynamic variables. Dell Boomi sends a Request to Oracle SaaS with input request parameters and Oracle BIP reports extracts the required data based on input parameter as a result. The response from Oracle is Base64 encoded.

All these steps have been arranged in this process such a way we can reuse this process.

In below sections will illustrate in detail how to use and setup the process along with example snapshots.

## **Configuration**

To establish the connection from Dell Boomi to Oracle SaaS we required WSDL URL, endpoint, username and password etc., these can be configured from atom level by extending these connection paraments at process level.

|  |  |
| --- | --- |
| Enable the extensions in Caller process for Connections and extended properties |  |
| **Connections:** Set the oracle connection parameters according to your connection in caller process Extensions. |  |

## **Process Usage**

BIP Schedule service jobs can be submitted with or without parameters. Please set up below parameters as Dynamic Document Properties in caller process

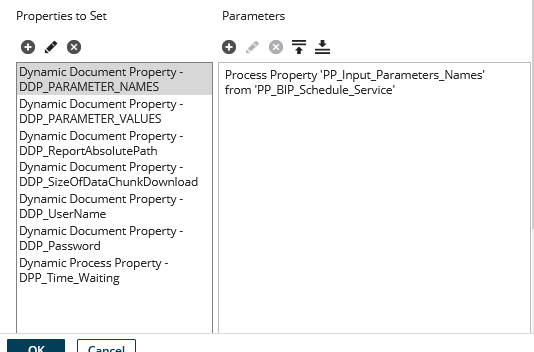
Parameter Name, Parameter Value details should be in sequence with ‘#’ as a delimiter.

For input with value range, pass the input value as comma delimited string.

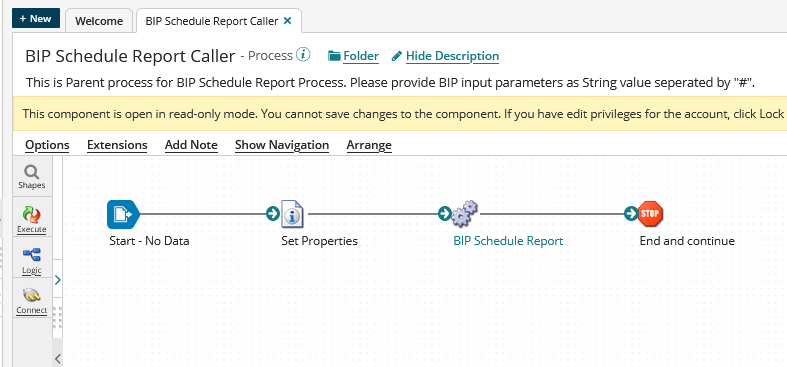
|  |  |  |  |
| --- | --- | --- | --- |
| Name of the Input property | Description | Mandatory/Optional | Property example |
| DDP\_PARAMETER\_NAMES | This property to set the input param names. Set the string value separated by “#” | Yes | ParamName1#ParamName2#ParamName3 |
| DDP\_PARAMETER\_VALUES | This property to set the input param value. Set the string value separated by “#” | Yes | Paramvalue1#Paramvalue2#Paramvalue3 |
| DDP\_ReportAbsolutePath | This property to set the path of the BIP Run Report job (from Oracle SaaS) | Yes | /Custom/BIPforBoomi/DemoRPT.xdo |
| DDP\_SizeOfDataChunkDownload | This property to set the size of data chunk download (from Oracle SaaS) | Yes | -1 |
| DDP\_UserName | This is the username of Oracle login | Yes |  |
| DDP\_Password | This is the password of Oracle login | Yes |  |

## **BIP\_Schedule Service\_Caller**

This Demo Process sets the required document properties using extended process properties. The document properties are consumed by reusable component to make BIP call.



The Demo Process calls the subprocess ‘BIP Schedule Report’ and on successful invocation receives the decoded report as response.



## **Web services used in Process**

1. **scheduleReport**

Use the scheduleReport() method to schedule the reports. You can submit and run the reports immediately or create a job to schedule the reports to run.

1. **getScheduledReportStatus**

|  |  |  |  |
| --- | --- | --- | --- |
| Input Message: getScheduledReportStatusRequest | | | |
| Parameter | Type | Nillable | Description |
| scheduledJobID | string | no | The job ID of the scheduled report. |
| userID | string | no | The BI Publisher user name. |
| password | string | no | The password for the entered user name. |

Use getScheduledReportStatus to get status information about a scheduled report

1. **getAllJobInstanceIDs**

Fetch the instance IDs of the successful jobs.

1. **getScheduledReportOutputInfo**

Use the getScheduledReportOutputInfo() method to return information about a specific scheduled report output.

1. **getScheduledReportDeliveryInfo**

Use the getScheduledReportdeliveryInfo() method to retrieve information about the delivery of a scheduled job output. For each scheduled Job, it could have multiple outputIDs. For each outputID, there could be multiple delivery info.

1. **getDocumentData**

Use the getDocumentData() method to return the document generated by scheduled job. You can use the JobOutputID returned from the scheduleReport() method to retrieve the generated report document.