

OBVAM EOD Configuration Guide

Oracle Banking Virtual Account Management

Release 14.5.0.0.0

Part Number F41346-01

May 2021

OBVAM EOD Configuration Guide

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Gurgaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax: +91 22 6718 3001

<https://www.oracle.com/industries/financial-services/index.html>

Copyright © 2018, 2021, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited. The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

1	Preface	1
1.1	Background	1
1.2	Introduction.....	1
1.3	Document Accessibility	1
1.4	Acronyms, Abbreviations and Definitions	1
1.5	Related Documents.....	1
2	EOD Configuration Steps	2
2.1	Steps to run EOD for a branch.....	5
3	Job definition Naming Convention	6
4	OBVAM Job.....	12

1 Preface

1.1 Background

Oracle Banking Virtual Account Management allows you to execute several functions every day on a routine basis as part of the End of Day (EOD) process. These functions can be run at various stages of the EOD process.

1.2 Introduction

The End of Day process is to tie up all the operations for a financial day and prepare the system for the next day. The EOD process should be defined for a branch and executed separately for each branch. When the process is running, you could choose to monitor it from Invoke EOD screen.

EOD uses OBMA Orchestrator and Batch service for orchestrating all the jobs required to complete End of Day processing. This document helps in the required set up to run EOD.

1.3 Document Accessibility

1. OBMA Orchestrator needs to be deployed as per installation guide.
2. OBMA Batch needs to be deployed as per installation guide.

1.4 Acronyms, Abbreviations and Definitions

Acronyms	Definition
OBVAM	Oracle Banking Virtual Account Management
OBMA	Oracle Banking Micro service Architecture
EOD	End of Day

1.5 Related Documents

The related documents are as follows:

- Oracle Banking Common Core User Guide
- Oracle Banking Virtual Account Management User Guides

2 EOD Configuration Steps

The following Functional Activities needs to be maintained in user's role to perform EOD operations

CMC_FA_BRANCH_EOD_PROCESS

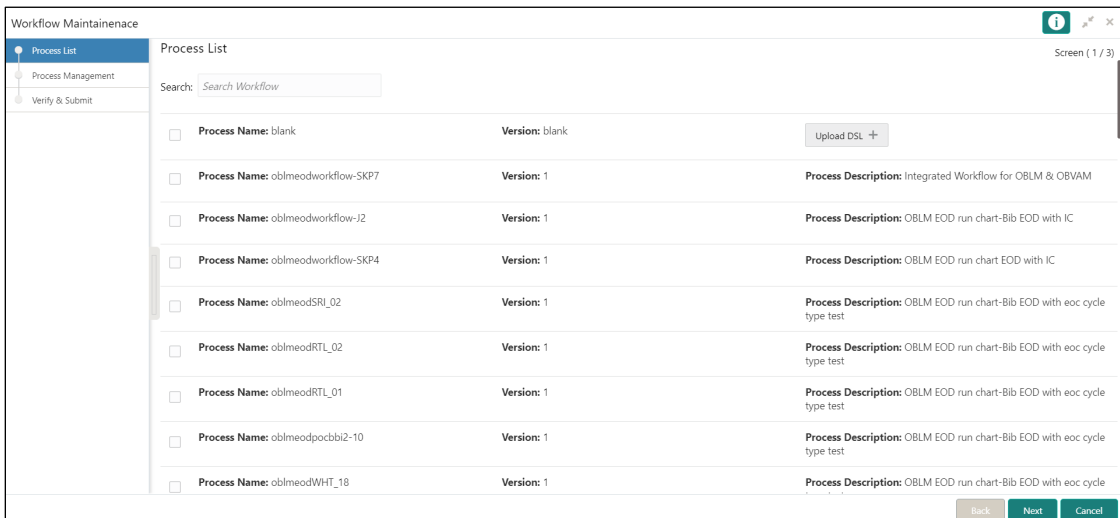
1. Save the below attachment to local folder. This is a standard batch process definition script for OBVAM that includes the list of batch tasks to be automatically executed in a sequence.



2. On **Home Screen**, under **Tasks** menu, click Business Process Maintenance to import, create or modify batch process definition.

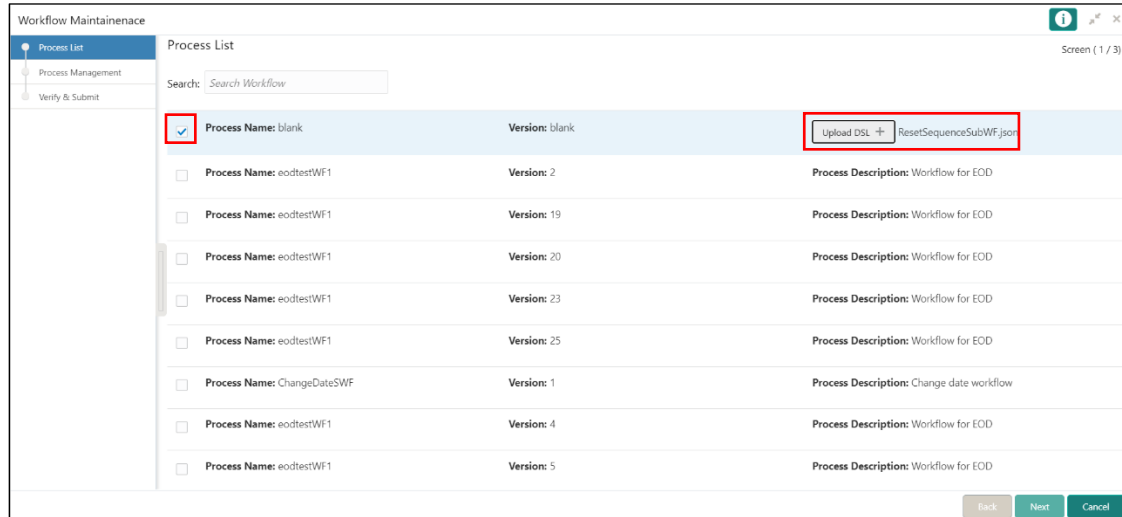
→ **Product List** screen is displayed.

Figure 1: Process List



3. Select the **Process Name: blank** checkbox. Click on **Upload DSL+** button to upload batch process definition, and choose file **ResetSequenceSubWF.json** and **EODWF.json** in order from the local folder.

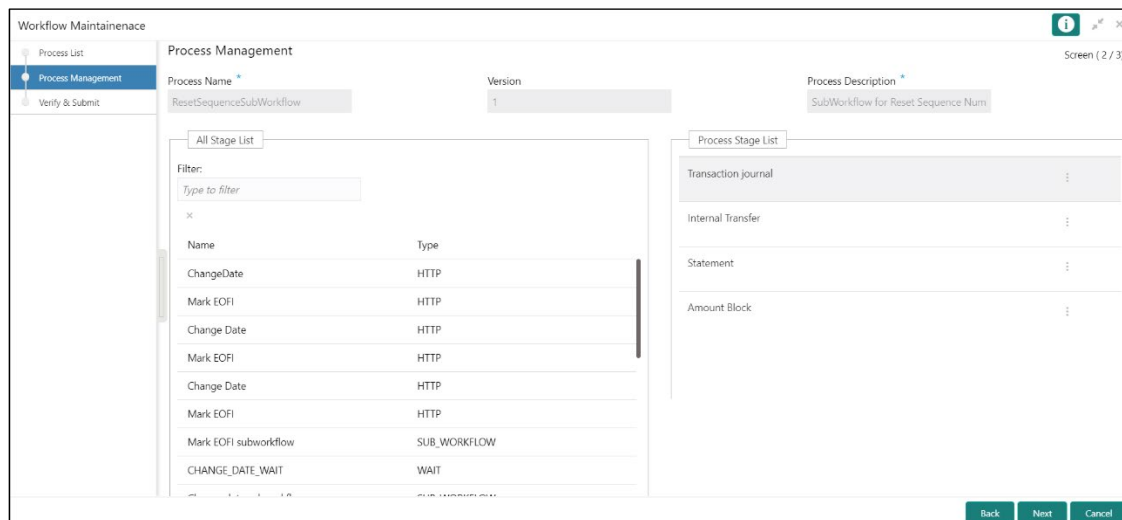
Figure 2: Process List – Upload DSL



4. Click **Next** button.

→ **Product Management** screen is displayed.

Figure 3: Process Management

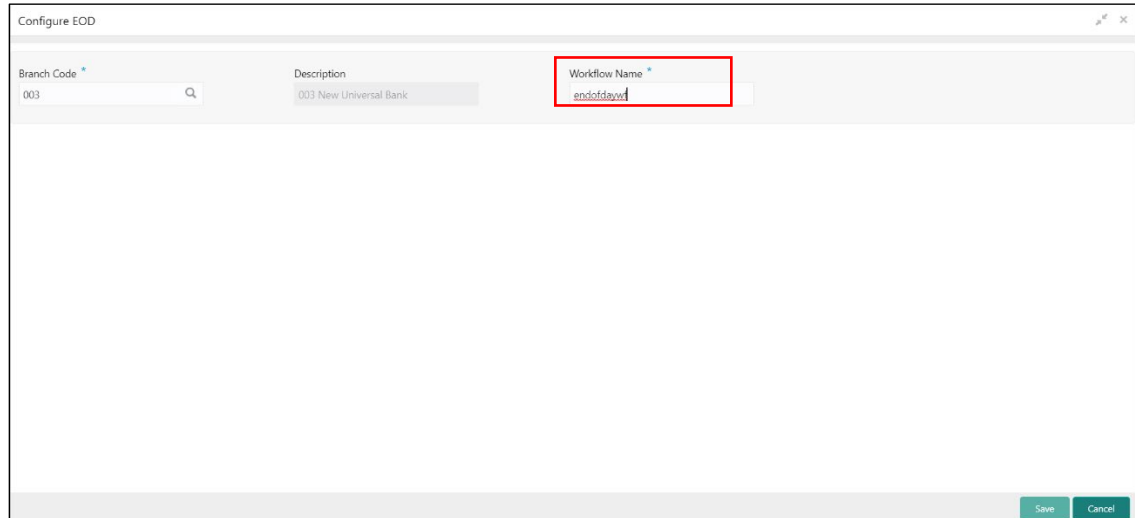


5. Click **Next** button and Click **Review** or **Create Process** in **Verify & Submit** screen to register the batch. Click **Process List** again to create new batch definition for **EODWF.json**.

6. On **Core Maintenance** menu, under **Branch EOD**, click **Configure EOD** to configure batch for a branch. Refer **Section 2.5** in **Oracle Banking Common Core User Guide**.

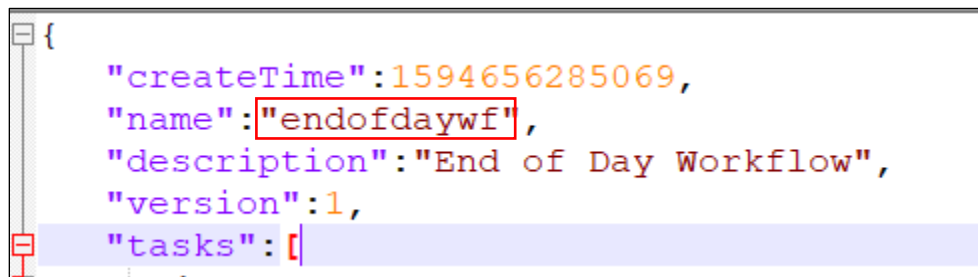
→ **Configure EOD** screen is displayed.

Figure 4: Configure EOD



7. Select the **Branch Code** to configure the batch.

Note: The value specified in **Workflow name** field in above screen must be exactly same as the **first name** attribute specified in batch process definition file **EODWF.json** file



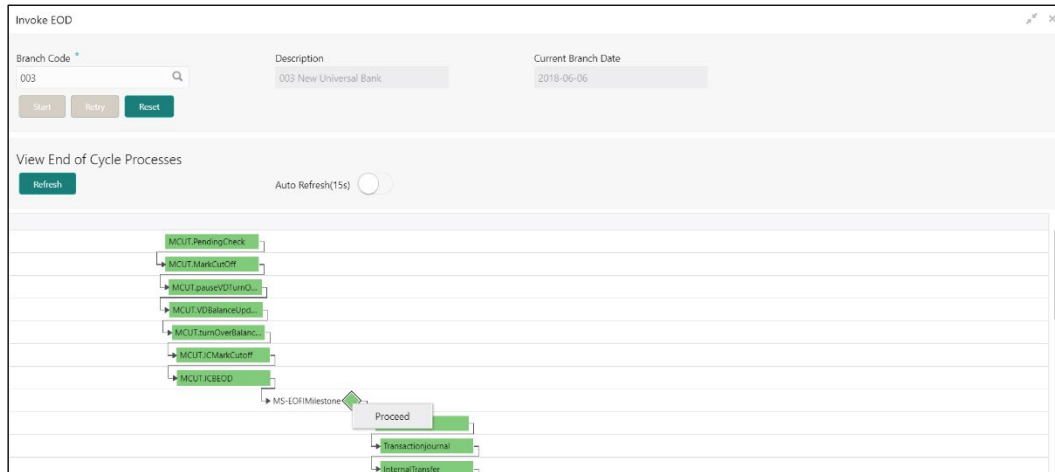
```
{
  "createTime":1594656285069,
  "name":"endofdaywf",
  "description":"End of Day Workflow",
  "version":1,
  "tasks":[]
}
```

2.1 Steps to run EOD for a branch

1. On **Core Maintenance** menu, under **Branch EOD**, click **Invoke EOD**.

→ **Invoke EOD** screen is displayed.

Figure 5: Invoke EOD



2. Select the branch to run EOD. Refer **Section 2.5** in **Oracle Banking Common Core User Guide**.
3. Click **Refresh** to view the current status of branch.

3 Job definition Naming Convention

Naming convention to be followed when a custom job is introduced as a task into EOD process

1. **Milestone task name** and **taskReferenceName** must be same and prefixed with “**MS-**“. Ex: MS-EOFIMilestone

Milestone

EOD run pause at each **Milestone** shall be resumed by clicking **Proceed** button manually.

Refer **Section 2.5** in **Oracle Banking Common Core User Guide**.

Sample template for milestone stage

```
{
  "type": "HTTP",
  "name": "MS-EOFIMilestone",
  "taskReferenceName": "MS-EOFIMilestone",
  "inputParameters": {
    "http_request": {
      "connectionTimeout": "0",
      "readTimeout": "0",
      "vipAddress": "CMC-BRANCH-SERVICES",
      "uri": "/cmc-branch-services/brancheod/milestone",
      "method": "POST",
      "headers": {
        "appId": "CMNCORE",
        "branchCode": "${workflow.input.branchCode}",
        "userId": "${workflow.input.userId}"
      },
      "body": {
        "data": [
          {
            "workflowId": "${workflow.workflowId}",
            "taskId": "${CPEWF_TASK_ID}",
            "waitTime": "5000"
          }
        ]
      }
    }
  },
  "asyncComplete": true,
  "startDelay": 0,
  "optional": false,
  "asyncComplete": true
},
```

Steps to integrate Custom Jobs

1. If the custom job uses OBMA Batch service, then use the below template to include the job as a task in EOD Flow definition.

```
{
  "type": "HTTP",
  "name": "<MilestoneCode.JobName>",
  "taskReferenceName": "<MilestoneCode.JobName>",
  "inputParameters": {
    "http_request": {
      "connectionTimeout": "0",
      "readTimeout": "0",
      "vipAddress": "PLATO-BATCH-SERVER",
      "uri": "/plato-batch-server/jobLauncher/launch/",
      "method": "POST",
      "headers": {
        "appld": "${workflow.input.appld}",
        "branchCode": "${workflow.input.branchCode}",
        "userId": "${workflow.input.userId}"
      },
    },
    "body": {
      "jobName": "<JobName>",
      "jobParameters": [
        {
          "key": "appld",
          "value": "<Application ID of microservice>"
        },
        {
          "key": "microServiceName",
```

```

    "value": "<Microservice name>"
  },
  {
    "key": "contextRoot",
    "value": "<Context root of microservice>"
  },
  {
    "key": "workflowId",
    "value": "${workflow.workflowId}"
  },
  {
    "key": "referenceTaskName",
    "value": "<MilestoneCode.JobName>"
  },
  {
    "key": "userId",
    "value": "${workflow.input.userId}"
  },
  {
    "key": "branchCode",
    "value": "${workflow.input.branchCode}"
  },
  {
    "key": "isCallback",
    "value": "Y"
  },
  {
    "key": "callbackType",

```

```

        "value":"PLATOORCH"
      }
    ]
  }
},
  "asyncComplete":true
},
  "startDelay":0,
  "optional":false,
  "asyncComplete":true
}

```

2. If the custom job doesn't use OBMA Batch service and the Batch API is implemented as a synchronous call, then use the below template to include the job as a task in EOD Flow definition

```

{
  "type":"HTTP",
  "name":"<MilestoneCode.JobName>",
  "taskReferenceName":"<MilestoneCode.JobName>",
  "inputParameters":{
    "http_request":{
      "connectionTimeout":"0",
      "readTimeout":"0",
      "vipAddress":"<Microservice name registered in eureka>",
      "uri":"<relative URL>",
      "method":"<HTTP Method>",
      "headers":{
        "appId":"${workflow.input.appId}",
        "branchCode":"${workflow.input.branchCode}",

```

```

        "userId":"${workflow.input.userId}"
    }
},
    "asyncComplete":false
},
    "startDelay":0,
    "optional":false,
    "asyncComplete":true
}

```

NOTE: HTTP Method - One of the GET, PUT, POST, DELETE, OPTIONS, HEAD

3. If the custom job doesn't use OBMA Batch service and if the Batch API is implemented as an asynchronous call, then call back needs to be implemented in the respective API. Please use the below template to include the job as a task in EOD Flow Definition.

```

{
    "type":"HTTP",
    "name":"<MilestoneCode.JobName>",
    "taskReferenceName":"<MilestoneCode.JobName>",
    "inputParameters":{
        "http_request":{
            "connectionTimeOut":"0",
            "readTimeOut":"0",
            "vipAddress":"<Microservice name registered in eureka>",
            "uri":"<relative URL>",
            "method":"<HTTP Method>",
            "headers":{
                "appld":"${workflow.input.appld}",
                "branchCode":"${workflow.input.branchCode}",
                "userId":"${workflow.input.userId}"
            }
        }
    }
}

```

```

    },
    "asyncComplete":true
  },
  "startDelay":0,
  "optional":false,
  "asyncComplete":true
}

```

The following API should be used as a call back to update the status of a task.

S No	Milestone	Job Name
Url	http://<hostname>:<port>/plato-orch-service/api/tasks	
Headers	userId : <Logged in user id> branchCode : <Logged in branch code> appld : platoorch Content-Type : application/json Accept : application/json	userId – User who updates the task branchCode – Branch where the update is performed
Body	<pre>{ "workflowInstanceId": "<EOD_Workflow_ID", "taskId": "<Task_ID>", "status": "<Status>" }</pre>	EOD_Workflow_ID – A Workflow ID gets generated when EOD is invoked Task_ID – Unique task ID gets generated for each task once it starts Status – COMPLETED / FAILED_WITH_TERMINAL_ERROR / FAILED / IN_PROGRESS

NOTE: asyncComplete – field in EOD workflow definition should be set to true if the Http task makes an asynchronous call and the task has to be updated explicitly by calling above update APIs. Only after successful update, next task will get executed.

4 OBVAM Job

S No	Milestone	Job Name	Description
1	MCUT	Pending Check	Task to check if any pending maintenance or transaction exist. This pending check task will fail if there is any unauthorized maintenance or transaction. If pending check task fails, you should check for unauthorized maintenance or transaction and take necessary action. This action could be authorizing/ deleting maintenance/ transaction.
2	MCUT	pauseVDTurn Over	Job to pause Intraday VdBalance and Turnover job.
3	MCUT	VDBalanceUp date	Job to calculate value dated balances for virtual accounts
4	MCUT	turnOverBalanceUpdate	Job to calculate turnover balance for a virtual account which is used for charge calculations
5	MCUT	ICMarkCutoff	Job to mark cutoff so that interest processing can start
6	MCUT	ICBEOD	Job to process interest calculations
7	EOFI	EodStatement	Job to generate EOD statement
8	EOFI	ForgetEntity	Job to forget virtual entity
9	EOFI	ForgetVirAccount	Job to forget virtual account
10	BOD	ICFlipDate	Job to change branch date

S No	Milestone	Job Name	Description
11	BOD	ResetSequenceWorkflow	Job to reset the sequence used to generate processing reference number for transactions, amount block/eca, internal transfer and statements
12	RCUT	ICReleaseCutoff	Job to release cutoff after interest processing is done
13	RCUT	UntankBalance	Job to untank accounting entries
14	RCUT	MarkAccountsInactive	Job to mark virtual accounts inactive
15	RCUT	AmountBlockExpiry	Job to mark amount block expired based on expiry date
16	RCUT	CreditLimitUtil	Job to re- valuate credit limit utilization based on updated exchange rates

S No	Milestone	Job Name	Description
11a	BOD	ResetSequenceSubWorkflow	Job to reset the sequence used to generate processing reference number for transactions, amount block/eca, internal transfer and statements