

EOD Configuration Guide

Oracle Banking Liquidity Management

Release 14.5.3.0.0

Part Number F50376-01

November 2021

EOD Configuration Guide

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (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 Oracle Banking Liquidity Management Job 12

1 Preface

1.1 Background

Oracle Banking Liquidity 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 Oracle Banking Microservice Architecture 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. Oracle Banking Microservice Architecture Orchestrator needs to be deployed as per installation guide.
2. Oracle Banking Microservice Architecture Batch needs to be deployed as per installation guide.

1.4 Acronyms, Abbreviations and Definitions

Acronyms	Definition
EOD	End of Day

1.5 Related Documents

The related documents are as follows:

- Oracle Banking Common Core User Guide
- Oracle Banking Liquidity Management User Guide

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 Oracle Banking Liquidity Management that includes the list of batch tasks to be automatically executed in a sequence.



OBLMEod.json

Note: Refer to PDF attachments for downloading json file.

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

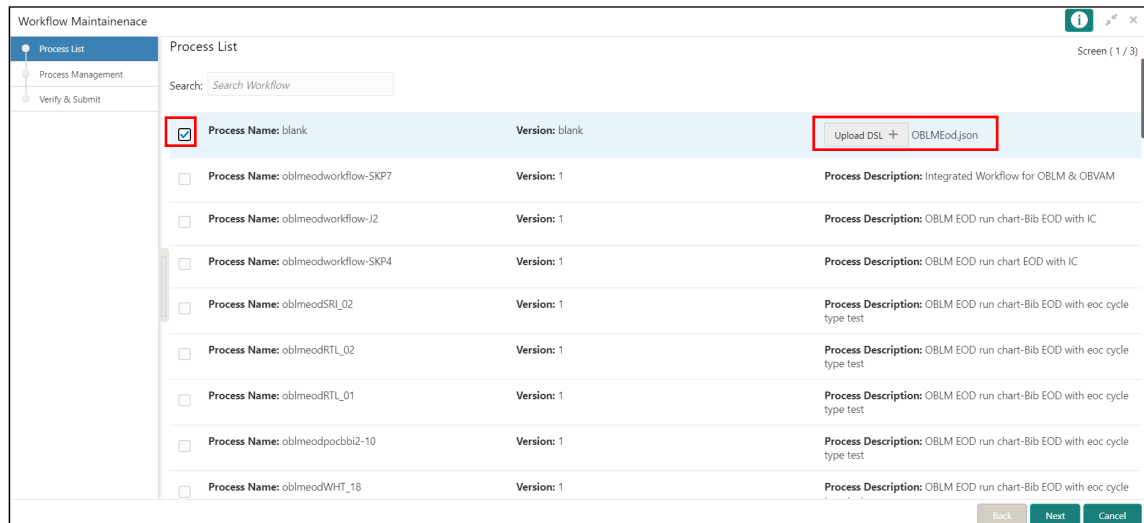
→ **Product List** screen is displayed.

Figure 1: Process List

Process Name	Version	Process Description
blank	blank	
oblmeodworkflow-SKP7	1	Integrated Workflow for OBLM & OBVAM
oblmeodworkflow-J2	1	OBLM EOD run chart-Bib EOD with IC
oblmeodworkflow-SKP4	1	OBLM EOD run chart EOD with IC
oblmeodSRL_02	1	OBLM EOD run chart-Bib EOD with eoc cycle type test
oblmeodRTL_02	1	OBLM EOD run chart-Bib EOD with eoc cycle type test
oblmeodRTL_01	1	OBLM EOD run chart-Bib EOD with eoc cycle type test
oblmeodpocbbi2-10	1	OBLM EOD run chart-Bib EOD with eoc cycle type test
oblmeodWHT_18	1	OBLM EOD run chart-Bib EOD with eoc cycle type test

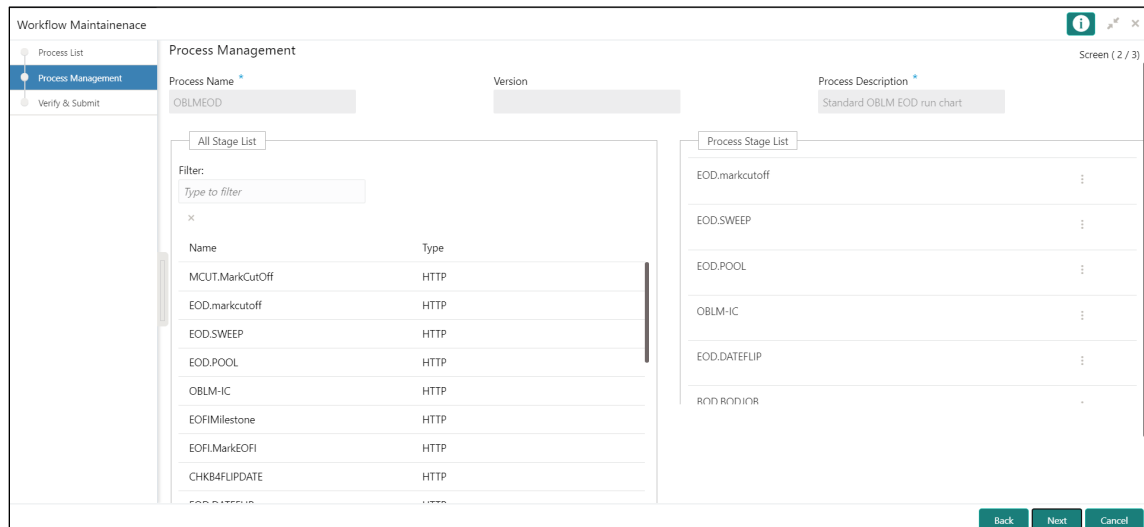
3. Select the **Process Name: blank** checkbox. Click on **Upload DSL+** button and choose file **OBLMEod.json** from local folder.

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.

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 branch code to configure the batch.

Note: The value specified in Workflow name field must be same as the workflow **name** attribute specified in 3rd line of batch script **OBLMEod.json** file

```

OBLMEod.json - Notepad
File Edit Format View Help
{
  "createTime": 1594656285069,
  "name": "OBLMEOD",
  "description": "Standard OBLM EOD run chart",
  "tasks": [{
    "type": "HTTP",
    "name": "EOD.markcutoff",
    "taskReferenceName": "EOD.markcutoff",
    "inputParameters": {
      "http_request": {
        "connectionTimeout": "0",
        "readTimeout": "0",
        "vipAddress": "OBLM-BATCH-SERVICES",
        "uri": "/oblm-batch-services/jobscheduler/markEodCutOff/${workflow.input.branchCode}",
        "method": "GET",
        "headers": {
          "appId": "LMB",
          "branchCode": "${workflow.input.branchCode}",
          "userId": "${workflow.input.userId}",
          "eocType": "EOD"
        }
      }
    },
    "startDelay": 0,
    "optional": false,
    "asyncComplete": false
  }],
  "type": "HTTP",

```

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

The screenshot displays the 'Invoke EOD' interface. At the top, there are input fields for 'Branch Code' (containing 'ME1'), 'Description' (containing 'ME1'), and 'Current Branch Date' (containing '2020-03-26'). Below these are 'Start', 'Retry', and 'Reset' buttons. A section titled 'View End of Cycle Processes' contains a 'Refresh' button and an 'Auto Refresh(15s)' toggle. The main area shows a process flow diagram with the following steps: EOD.markutoff1 (green bar), EOD.SWEEP1 (green bar), EOD.POOL1 (green bar), OBLM-IC1 (yellow bar), EOD.DATEFLUP1 (grey bar), and BOD.BODJOB1 (grey bar). Arrows indicate the sequence of operations.

2. Select the branch to run EOD.
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 must be prefixed with "MS-". Ex: MS-BranchCutOff

Milestone stage

Milestone stage will pause the batch execution till it is manually resumed.

Sample template for milestone stage

```
{
  "name": "MS-CHKAFTEREOTI",
  "taskReferenceName": "MS-CHKAFTEREOTI",
  "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"
          }
        ]
      }
    }
  },
  "type": "HTTP",
  "startDelay": 0,
  "optional": false,
  "asyncComplete": true
}
```

Steps to integrate Custom Jobs

1. If the custom job uses Oracle Banking Microservice Architecture 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 uses Oracle Banking Microservice Architecture 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":{
        "appld":"${workflow.input.appld}",
        "branchCode":"${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 uses Oracle Banking Microservice Architecture 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.

Method	POST	
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> { "workflowInstancelId": "<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 Oracle Banking Liquidity Management Job

S No	EOD stage	Job Name	Description	Input Parameters
1	MCUT	markcutoff	Job will check for pending tasks and any existing running process before starting EOD	BranchCode
	MS-EOD	MS-EOD	Milestone stage for running EOD	
2	EOD	SWEEP	Job will execute sweep process scheduled to run during EOD	BranchCode
3	EOD	POOL	Job will execute all pool structures.	BranchCode
4	EOD	IC	Job will execute Interest batch	BranchCode
5	EOD	DATEFLIP	Job will change system date to next working date	BranchCode
6	MS-BOD	MS-BOD	Milestone stage for running BOD	BranchCode
6	BOD	BOD	Job will execute Reallocation, Reverse sweep and bod sweeps in sequence	BranchCode