

Oracle® Banking Trade Finance

Common Core - Performance Diagnostic Plugin (OLTP) User Guide



Release 14.8.1.0.0
G46731-01
October 2025

ORACLE®

Copyright © 2017, 2025, Oracle and/or its affiliates.

Primary Authors: (primary author), (primary author)

Contributing Authors: (contributing author), (contributing author)

Contributors: (contributor), (contributor)

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.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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 fail-safe, 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.

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

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Purpose	i
Acronyms and Abbreviations	i
Audience	ii
Basic Actions	ii
Critical Patches	iii
Conventions	iii
Diversity and Inclusion	iii
Documentation Accessibility	iv
Prerequisite	iv
Related Resources	iv
Screenshot Disclaimer	iv
Symbols and Icons	iv

1 PDP Functionality

1.1 PDP Maintenance	1
1.1.1 Process Performance Diagnostics Plugin Summary	1
1.2 Simulation	2
1.2.1 Performance Issue at Screen Level	3
1.2.1.1 Problem Description	3
1.2.1.2 Problem Simulation	3
1.2.1.3 Results Captured	4
1.3 Results	4
1.3.1 Process Timings	5
1.3.2 Application Logs	6
1.3.3 TEST_USER_STDCIF_EXECUTEQUERY	6

2 Error Code and Messages

3 Frequently Asked Questions

Preface

This topic contains the following sub-topics:

- [Purpose](#)
- [Acronyms and Abbreviations](#)
- [Audience](#)
- [Basic Actions](#)
- [Critical Patches](#)
- [Conventions](#)
- [Diversity and Inclusion](#)
- [Documentation Accessibility](#)
- [Prerequisite](#)
- [Related Resources](#)
- [Screenshot Disclaimer](#)
- [Symbols and Icons](#)

Purpose

This user manual is designed to help you quickly get acquainted with the many functions routinely executed everyday in Oracle Banking Trade Finance.

To access information specific to a particular field, place the cursor on the relevant field and press **F1** on the keyboard.

Acronyms and Abbreviations

The list of the acronyms and abbreviations used in this guide are as follows:

Table 1 Abbreviation

Abbreviation	Description
POSTEOPD	Post End of Previous Day
MARKEOPD	Mark End of Previous Day
MARKTI	Mark Transaction Input
POSTEOBOD	Post End of Beginning of Day
MARKBOD	Mark Beginning of Day
MARKEOD	Mark End of Day
POSTEOED	Post End of End of Day
MARKEOFI	Mark End of Financial Input

Table 1 (Cont.) Abbreviation

Abbreviation	Description
POSTEOF	Post End of Financial Input
MARKEOTI	Mark End of Transaction Input
POSTEOTI	Post End of Transaction Input
TI	Transaction Input
EOC	End of Cycle
BOD	Beginning of Day
EOD	End of Day
EOPD	End of Previous Day
FI	Financial Input
EOTI	End of Transaction Input

Audience

This manual is intended for the following User/User Roles:

Table 2 Audience

Role	Function
Back office clerk	Input functions for contracts
Back office managers/officers	Authorization functions
Product Managers	Product definition and authorization
End of Day operators	Processing during End of Day/Beginning of Day
Financial Controller/Product Managers	Generation of reports

Basic Actions

Table 3 List of Basic Actions

Action	Description
Approve	Click Approve to approve the initiated report. This button is displayed, once the user click Authorize .
Audit	Click Audit to view the maker details, checker details of the particular record, and record status. This button is displayed only for the records that are already created.
Authorize	Click Authorize to authorize the record created. A maker of the screen is not allowed to authorize the report. Only a checker can authorize a record. This button is displayed only for the already created records.
Close	Click Close to close a record. This action is available only when a record is created.
Confirm	Click Confirm to confirm the performed action.
Cancel	Click Cancel to cancel the performed action.

Table 3 (Cont.) List of Basic Actions

Action	Description
Compare	Click Compare to view the comparison through the field values of old record and the current record. This button is displayed in the widget, once the user click Authorize .
Collapse All	Click Collapse All to hide the details in the sections. This button is displayed, once the user click Compare .
Expand All	Click Expand All to expand and view all the details in the sections. This button is displayed, once the user click Compare .
New	Click New to add a new record. The system displays a new record to specify the required data. Note: The fields which are marked in asterisk red are mandatory fields.
OK	Click OK to confirm the details in the screen.
Save	Click Save to save the details entered or selected in the screen.
View	Click View to view the report details in a particular modification stage. This button is displayed in the widget, once the user click Authorize .
View Difference only	Click View Difference only to view a comparison through the field element values of old record and the current record, which has undergone changes. This button is displayed, once the user click Compare .

Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at [Critical Patches, Security Alerts and Bulletins](#). All critical patches should be applied in a timely manner to make sure effective security, as strongly recommended by [Oracle Software Security Assurance](#).

Conventions

The following text conventions are used in this document:

Table 4 Conventions and Meaning

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also

mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Prerequisite

Specify the **User ID** and **Password**, and login to **Home** screen.

Related Resources

For more information on any related features, refer to the following documents in the FLEXCUBE Universal Banking Documentation Library:

- Procedures User Guide
- Products User Guide
- End of Day User Guide

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Symbols and Icons

The list of symbols and icons available on the screens are as follows:

Table 5 Symbols and Icons - Common

Symbol/Icon	Function
	Minimize
	Maximize
	Close
	Perform Search

Table 5 (Cont.) Symbols and Icons - Common

Symbol/Icon	Function
	Open a list
	Navigate to the first record
	Navigate to the last record
	Navigate to the previous record
	Navigate to the next record
	Grid view
	List view
	Refresh
	Click this icon to add a new row.
	Click this icon to delete a row, which is already added.
	Calendar
	Alerts
	Unlock Option
	View Option
	New
	Enter query
	Execute query
	Copy
	Delete

Table 5 (Cont.) Symbols and Icons - Common

Symbol/Icon	Function
	Save
	Search
	Advanced search
	Clear all
	Reset
	Export
	Print
	View Details
	Sorting

Table 6 Symbols and Icons - Widget

Symbol/Icon	Function
	Open status
	Unauthorized status
	Rejected status
	Closed status
	Authorized status
	Modification Number
	Hold
	Reverse

Table 6 (Cont.) Symbols and Icons - Widget

Symbol/Icon	Function
	Authorize
	Rollover

1

PDP Functionality

This topic describes the Performance Diagnostic Plugin tool and its functionality.

The Performance Diagnostic Plugin (PDP) is a tool to capture the performance lag reports like TKPROF and Hierarchical profiling (HPROF) along with capturing processing time for OLTP transactions in Oracle FLEXCUBE Universal Banking.

Existing methods of capturing these reports at a very high level causing difficulties in analyzing the precise point of issue and involves DBA's. The Performance Diagnostic Plugin helps to get the report precisely at the issue level that is action based reports for a user.

This topic has the following sub-topics:

- [PDP Maintenance](#)
This topic provides the information on PDP maintenance to generate performance report.
- [Simulation](#)
This topic describes the performance issue, issue results, process timings for easy analysis, and application logs.
- [Results](#)
This topic provides the information results captured by the system.

1.1 PDP Maintenance

This topic provides the information on PDP maintenance to generate performance report.

The Performance Diagnostic Plug-in tool depends on the initial maintenance of the performance issue faced by the user to generate the performance bottleneck reports for analysis.

This topics contains the following sub-topics:

- [Process Performance Diagnostics Plugin Summary](#)
This topic explains systematic instructions to process the Performance Diagnostics Plugin (OLTP) summary.

1.1.1 Process Performance Diagnostics Plugin Summary

This topic explains systematic instructions to process the Performance Diagnostics Plugin (OLTP) summary.

1. On **Homescreen**, type **STSAPDP** in the text box, and click **Next**.
The **Performance Diagnostic Plugin (OLTP) Summary** screen displays.

Figure 1-1 Performance Diagnostics Plugin (OLTP) Summary

Performance Diagnostic Plugin (OLTP) Summary

Search

Advanced Search

Reset

Clear All

Records per page15

Search (Case Sensitive)

Authorization Status

Function Id

Record Status

Action

User ID

Trace Value

Search Results

Lock Columns0

☐

Authorization Status

☐

Record Status

☐

User ID

☐

Function Id

☐

Action

☐

Trace Value

No data to display.

Page1Of1<<1>>

Exit

2. On the **Performance Diagnostic Plugin (OLTP) Summary** screen, specify the fields.

Note

The fields, which are marked with an asterisk, are mandatory.

For more information on fields, refer to the field description table.

Table 1-1 Performance Diagnostic Plugin (OLTP) Summary - Field Description

Field	Description
Authorization Status	Select the authorization status from the drop-down list: - Authorized - Unauthorized - Rejected
Record Status	Specify the record status from the drop-down list: - Open - Closed
User ID	Click Search and select the User ID from the list of values displayed.
Action	Click Search and select the Action .
Function ID	Click Search and select the Function ID .
Trace Value	Specify the trace value from the drop-down list: - Process Timing at Database Level - Process Timing at Application Level - Application Logs - ALL

3. Click **Search** after specifying the search parameters.
- The system identifies all records satisfying the specified criteria and displays the same.
4. Click **Exit** to end the transaction.

1.2 Simulation

This topic describes the performance issue, issue results, process timings for easy analysis, and application logs.

After the maintenance in STDAPPDP, a particular user facing the performance issue needs to simulate the case within the system to capture the required performance lag reports or process timings. The below sections, depict a sample performance issue faced by a user at both screen level as well as external transaction/gateway level.

This topic has the following sub-topics:

- [Performance Issue at Screen Level](#)
This topic describes the performance issues faced at the screen level.

1.2.1 Performance Issue at Screen Level

This topic describes the performance issues faced at the screen level.

This topic contains the following sub-topics:

- [Problem Description](#)
This topic describes the problem description.
- [Problem Simulation](#)
This topic describes the problem simulation in performance issues faced at the screen level.
- [Results Captured](#)
This topic describes the results captured in performance issues faced at the screen level.

1.2.1.1 Problem Description

This topic describes the problem description.

The user **TEST_USER** is facing a problem during executing query operation in the **Customer Maintenance** screen. The query operation takes more time to display the record and the support team has asked the user to share the process timings at both the Database and Application level, TKPROF, and Application Log to analyze the same.

1.2.1.2 Problem Simulation

This topic describes the problem simulation in performance issues faced at the screen level.

1. To capture the performance lag reports along with process timing, the user needs to maintain these details in the system under the **Performance Diagnostic Plugin (OLTP) Maintenance** screen as specified under Maintain Performance Diagnostic Plugin.

Table 1-2 Performance Diagnostic Plugin (OLTP) Maintenance - Example

Field	Description
User ID	TEST_USER
Function ID	STDCIF
Action	EXECUTEQUERY

As per the requirement, the user needs to capture process timings, TKPROF and Application Log. The selected trace value would be **ALL** that is it is a combination of all the trace value options.

Figure 1-2 Performance Diagnostic Plugin (OLTP) Maintenance

2. Once the above maintenance is completed, the user must log in to the **Customer Maintenance** screen with user ID **TEST_USER**, and query for a customer record. Click **Enter Query** and input the customer number. later, click **Execute Query**.
3. Due to the performance issue faced by the user, fetching of the customer details might takes few minutes. Once the details are displayed, the required performance lag reports are captured into their respective folders/tables as configured.

1.2.1.3 Results Captured

This topic describes the results captured in performance issues faced at the screen level.

- **Process Timings** - Captured process timings for the **Execute Query** operation are inserted into the table **GWTB_DBTIME_LOG** for both the Database and Application level as per the selection of the **Trace Value** in the maintenance screen. The below image shows the timing details captured for the entire operation of the **Execute Query**.

Figure 1-3 Process Timings

USERID	FUNCTION_ID	ACTION	CHANNEL	LAYER	START_TIMESTAMP	END_TIMESTAMP	TRN_INFO
TEST_USER	STDCIF	EXECUTEQUERY	UI	APP_TIMING	17-JUN-21 11.30.56.641000000 AM	17-JUN-21 11.30.57.553000000 AM	EXECUTEQUERY_SUCCESS
TEST_USER	STDCIF	EXECUTEQUERY	UI	Database Timings	17-JUN-21 11.30.56.768634000 AM	17-JUN-21 11.30.57.487362000 AM	EXECUTEQUERY_SUCCESS

- **Application Log** - Captured application logs are placed under application log directories. Refer to the **TEST_USER_STDCIF_EXECUTEQUERY.log** file for more information.

1.3 Results

This topic provides the information results captured by the system.

Results captured by the system are subjected to the maintenance of Trace. All files/timings captured are placed under configured directories/tables.

This topic contains the following sub-topics:

- [Process Timings](#)
This topic describes the process timings for easy analysis.
- [Application Logs](#)
This topic describes application logs generated by the system.
- [TEST_USER_STDCIF_EXECUTEQUERY](#)
This topic provides details of the file TEST_USER_STDCIF_EXECUTEQUERY.

1.3.1 Process Timings

This topic describes the process timings for easy analysis.

The Performance Diagnostic Plugin facilitates the capture of process timings for easy analysis and the timings captured at both Database and Application/Infra level based on the selection of the Trace Value. All the process timings are captured under the table **GWTB_DBTIME_LOG**.

Figure 1-4 GWTB_DBTIME_LOG Table

KEYID	USERID	FUNCTION_ID	ACTION	CHANNEL	LAYER	START_TIMESTAMP	END_TIMESTAMP	TRN_INFO

For more information on fields, refer to field description table.

Table 1-3 GWTB_DBTIME_LOG Table Description

Field	Description
User ID	This column displays the User ID of the simulating user facing performance issues.
Function ID	This column displays the Function ID used during the simulation of the issue.
Action	This column displays the operation for which the performance issue is being faced.
Channel	This column displays the channel via which the operation is carried out. UI - Operation via Screen GW - External System or Gateway
Layer	This column displays the layer at which the transaction timings were captured. Application Timings = Application layer Database Timings = Database Layer
Start Timestamp	This column displays the start timing of the operation.
End Timestamp	This column displays the end timing of the operation.
Transaction Info	This column specifies if the process was successful or failure along with the operation being pre-fixed.

Capturing of process timings happens on a selection of the below trace:

- **Process Timing at Database Level**
- **Process Timing at Application Level**
- **All**

1.3.2 Application Logs

This topic describes application logs generated by the system.

Application logs are the Java-level debug files generated by the system to capture the code flow along with errors for analysis in the Java layer. These logs are similar to the existing application-level logs generated but consist only of operation-level details.

The Performance Diagnostic Plugin captures the application logs for the specific user, function ID and operation as specified during the maintenance. These logs are not dependent on enabling/disabling at the **CSTB_DEBUG_USERS** table but are generated automatically on the selection of specific trace values during the maintenance of that operation.

These logs are generated under a specific directory configured during the initial setup which is **APPLICATION_WORK_AREA** specified under the **fcubs.properties** file. Generated logs follow a standard naming convention <USERID>_<FUNCTIONID>_<ACTION>.log

The generation of Application logs happens on the selection of the below trace values:

- **Application Logs**
- **ALL**

For the sample of the Application Log, refer to the **TEST_USER_STDCIF_EXECUTEQUERY.log** file.

1.3.3 TEST_USER_STDCIF_EXECUTEQUERY

This topic provides details of the file TEST_USER_STDCIF_EXECUTEQUERY.

```
[20210621 12:52:39 718]FCClientHandler.doPost-->User Debug true - pdpDbg_chk
true
[20210621 12:52:39 718]FCClientHandler.doPost-->Processing FCJ screen
[20210621 12:52:39 718]FCClientHandler.doPost-->contentLength=776
[20210621 12:52:39 718]FCClientHandler.doPost:param_operation-->Validating
the CSRF token
[20210621 12:52:39 718]FCClientHandler.doPost-->msgType=null
[20210621 12:52:39 719]FCClientHandler.doPost-->Validating request xml with
validateJavascript
[20210621 12:52:39 720]FCClientHandler.doPost-->userId=TEST_USER
[20210621 12:52:39 720]FCClientHandler.doPost-->FunctionId=STDCIF
[20210621 12:52:39 720]FCClientHandler.doPost-->Operation=EXECUTEQUERY
[20210621 12:52:39 720]FCClientHandler.doPost:param_operation--
>PARAM_OPERATIONnull
[20210621 12:52:39 720]FCClientHandler.doPost-->DbUpload=FALSE
[20210621 12:52:39 720]FCClientHandler.doPost-->dmsRefNo=null
[20210621 12:52:39 721]FCClientHandler.Trace Value fetched from
SMTB_USER_FUNCTION_TRACE
[20210621 12:52:39 721]FCClientHandler.doPost-->PDP check - Time capture
started
[20210621 12:52:39 721]FCClientHandler.doPost-->PDP check - Debug capture
started
[20210621 12:52:39 721]BranchLogger.renameDebugFile-->Log File rename
successfull - TEST_USER_2021-06-21_12-52-39
[20210621 12:52:39 721]FCClientHandler.doPost-->refreshing the request XML
header with session variables.
```



```

[20210621 12:52:39 725]FCClientHandler.doPost-->calling callHostServerNew
[20210621 12:52:39 725]FCClientHandler.doPost-->calling callHostServerNew for
New or Modify or ExecuteQuery
[20210621 12:52:39 725]ToHostDispatcher.callHostServer-->calling
callHostServer
[20210621 12:52:39 725]CentralizedCall.CentralizedCall-->calling host upload
servlet in centralized mode...
[20210621 12:52:39 725]CentralizedCall.CentralizedCall.callHostServer--
>oprationCode=EXECUTEQUERY
[20210621 12:52:39 726]HostUpload.processMsg-->***** APPSERVER LOG START *****
[20210621 12:52:39 726]HostUpload.processMsg-->getting FCUserGlobals from
FCApplicationGlobals.getFCUserGlobals
[20210621 12:52:39 726]HostUpload.processLaunchScreen-->Calling isLoginValid
[20210621 12:52:39 726]FCJSMSBean.isLoginValid-->userId = TEST_USER
branchCode = 000
[20210621 12:52:39 726]SMSOracleDAO.getUserLogStatus-->userId = TEST_USER
[20210621 12:52:39 726]SMSOracleDAO.getUserLogStatus-->signOnSerial = 38996
[20210621 12:52:39 727]SMSOracleDAO.getUserLogStatus-->loginCount = 1
[20210621 12:52:39 727]HostUpload.processLaunchScreen-->txnBranch=000
[20210621 12:52:39 727]HostUpload.processMsg-->Creating FCControllerSMS
[20210621 12:52:39 727]HostUpload.processMsg-->Creating
FCControllerSMS.handleRequest
[20210621 12:52:39 727]FCControllerMaint.handleRequest-->just Entered
[20210621 12:52:39 727]FCControllerMain.handleRequest-->checking
smsActionAccess in smsLocal
[20210621 12:52:39 727]FCControllerMain.handleRequest-->bActionAccess=true
[20210621 12:52:39 727]FCControllerMain.handleRequest-->calling
fcController.handleRequest -- NEWROUTE
[20210621 12:52:39 727]FCController.handleRequest-->calling parseRequest
[20210621 12:52:39 727]FCHandlerMaintenance.execute-->creating fcjeb and
miscjbb
[20210621 12:52:39 727]FCHandlerMaintenance.execute-->calling StaxParseUtility
[20210621 12:52:39 728]FCHandlerMaintenance.processMsgUBS-->just Entered
[20210621 12:52:39 728]FCHandlerMaintenance.processMsgUBS-->calling
preProcessMessage
[20210621 12:52:39 728]FCHandlerMaintenance.preProcessMsg-->Inside
preProcessMsg
[20210621 12:52:39 728]XmlAttachmentsParser.getUserName--> just Entered
[20210621 12:52:39 728]FCHandlerMaintenance.processMsgUBS-->calling
processBasedonRoutingType
[20210621 12:52:39 728]FCHandlerMaintenance.processBasedonRoutingType--
>Inside processBasedonRoutingType
[20210621 12:52:39 729]FCHandlerMaintenance.processBasedonRoutingType--
>Tanking is not Enabled,calling processJavaSrcRouter
[20210621 12:52:39 729]FCHandlerMaintenance.processJavaSrcRouter-->Calling
gwpksServiceRouter.processReqMsg
[20210621 12:52:39 729]GwpksServiceRouter.ProcessReqMsg()-->Just Entered..
[20210621 12:52:39 729]FCUtilityBean.getReleaseSmtbParameters()--
>Query=select /*+ result_cache */ nvl(release_type,'CUSTOM') from
smtb_parameters
[20210621 12:52:39 730]CspksReqGlobal().MsgInit()-->ReleaseType=CUSTOM
[20210621 12:52:39 730]GwpksServiceRouter.ProcessReqMsg()-->Calling
PrInitHeader()..
[20210621 12:52:39 730]GwpksServiceRouterMain.PrInitHeader()-->Initializing
Headers..
[20210621 12:52:39 730]GwpksServiceRouterMain.extractAddlTag()-->JustEntered..

```

```

[20210621 12:52:39 730]GwpksServiceRouterMain.extractAddlTag()-->Executes
When Source="FLEXBRANCH"
[20210621 12:52:39 730]GwpksServiceRouterMain.extractAddlTag()-->return
success from fn_extract_addl_tag()
[20210621 12:52:39 730]GwpksServiceRouterMain.extractAddlTag()-->Returning
from Extract_Addl_Tag..
[20210621 12:52:39 730]GwpksServiceRouter.ProcessReqMsg()-->Calling
populateMsgID()..
[20210621 12:52:39 730]FCUtilityBean.getPopulateMsgId()-->Query : SELECT
fcj_incoming.NEXTVAL from dual
[20210621 12:52:39 732]GwpksServiceRouterMain.populateMsgID()-->Message
ID=6121172000076994
[20210621 12:52:39 732]GwpksServiceRouter.ProcessReqMsg()-->Calling
dedupeLogMsg()..
[20210621 12:52:39 732]GwpksServiceRouter.ProcessReqMsg()-->Calling
enrichHeader()..
[20210621 12:52:39 732]GwpksServiceRouterMain.enrichHeader()-->Just Entered..
[20210621 12:52:39 732]GwpksServiceRouterMain.enrichHeader()-->Processing for
Functionid and Action..
[20210621 12:52:39 732]GwpksServiceRouterMain.enrichHeader()-->Calling
getRoutingTypeSmtbsMenu
[20210621 12:52:39 732]FCUtilityBean.getRoutingTypeSmtbsMenu()--
>Query=SELECT /*+ result_cache */ nvl(routing_type, '0'),nvl(logging_reqd,
'N'),type_string, nvl(Tank_modifications,'N') from smtbs_menu WHERE
function_id= ?
[20210621 12:52:39 733]GwpksServiceRouterMain.enrichHeader()-->lRoutingType=X
[20210621 12:52:39 733]GwpksServiceRouterMain.enrichHeader()-->lRoutingType=X
[20210621 12:52:39 733]GwpksServiceRouterMain.enrichHeader()-->ENTERED
[20210621 12:52:39 733]GwpksServiceRouter.ProcessReqMsg()-->setting the
enriched header into reqGlobal
[20210621 12:52:39 733]GwpksServiceRouter.ProcessReqMsg()--
>STDCIF,EXECUTEQUERY
[20210621 12:52:39 733]GwpksServiceRouter.ProcessReqMsg()-->Calling
processRequests()..
[20210621 12:52:39 733]GwpksServiceRouter.processRequests()-->calling
ProcessRequest()
[20210621 12:52:39 733]GwpksServiceRouter.ProcessRequest()-->The Operation/
Action isnull/EXECUTEQUERY
[20210621 12:52:39 733]GwpksServiceRouter.ProcessRequest()-->Calling
cspksReqHandler.processMsg()..
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->Preparing ~ seperated
Header nodes and Values..
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->25956930 :: l_msg_id ::
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->FCUBS_MSG_ID ::
6121172000076994
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->25956930 :: inside
if ::
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->headerNames--
>SOURCE~USERID~BRANCH~SERVICE~OPERATION~MULTITRIPID~FUNCTIONID~ACTION~MSGSTAT~
MODULEID~MSGID~DEBUG_MODE~PKVALS~SOURCE_OPERATION~MSG_XCHANGE_PATTERN~MFA_AUTH
ENTICATED~MAKERREMARKS~CHECKERREMARKS~SNAPSHOTID~CORRELID~CORRELATION_PATTERN~
EXT_TRIP_ID~
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->headerValues--
>FLEXCUBE~TEST_USER~000~~~6121172000076994~STDCIF~EXECUTEQUERY~~ST~61211720000
76994~N~~~IOFS~N~~~~~
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->calling

```

```

cspksXmlParser.fcdataParser()
[20210621 12:52:39 733]CspksXmlParser.fcDataParser()-->JustEntered..
[20210621 12:52:39 733]CspksReqHandler.processMsg()-->calling
cspksReqWrapper.processmsg()
[20210621 12:52:39 733]CspksReqWrapper.processmsg()-->Just Entered..
[20210621 12:52:39 734]CspksReqWrapper.execFun()-->Just Entered..
[20210621 12:52:39 734]CspksReqWrapper.execFun()-->tagNames=null
[20210621 12:52:39 734]CspksReqWrapper.execFun()--
>tagValues=N~000022~I~-----000~-----
-----N-----N-----N-----
~~~~~
[20210621 12:52:39 734]CspksReqWrapper.execFun()-->parentsList=BLK_CUSTOMER
[20210621 12:52:39 734]CspksReqWrapper.execFun()-->Calling processDBMain
[20210621 12:52:39 734]CspksReqWrapper.execFun()-->request passing to
dbmain..<?xml version="1.0" encoding="UTF-8"?
><FCUBS_REQ_ENV><FCUBS_HEADER><SOURCE>FLEXCUBE</SOURCE><UBSCOMP>FCUBS</
UBSCOMP><USERID>TEST_USER</USERID><BRANCH>000</BRANCH><SERVICE></OPERATION/
><MULTITRIPID></FUNCTIONID>STDCIF</FUNCTIONID><ACTION>EXECUTEQUERY</
ACTION><MSGSTAT></MODULEID>ST</MODULEID><MSGID></DEBUG_MODE>N</
DEBUG_MODE><ENTITY>ENTITY_ID1</ENTITY><MAKERREMARKS></MAKEROVDREMARKS/
><CHECKERREMARKS></SNAPSHOTID></PKVALS>000022~000</
PKVALS><PKFIELDS>BLK_CUSTOMER_CUSTNO~BLK_CUSTOMER_LBRN</
PKFIELDS><SQNO>38997</SQNO></FCUBS_HEADER><FCUBS_BODY><REC
TYPE="BLK_CUSTOMER" RECID="1"><FV><!
[CDATA[N~000022~I~-----000~-----
-----N-----N-----N-----
~]]></FV></REC><MISC><REMARKS></MISC></FCUBS_BODY></FCUBS_REQ_ENV>
[20210621 12:52:39 734]FCMainProcEJBBean.execFun()-->Just Entered..
[20210621 12:52:39 734]FCMainProcEJBBean.execFun()-->processDBMain-->calling
GWPKS_RESET_PKG.PR_RESET_PKG()***
[20210621 12:52:39 735]FCMainProcEJBBean.execFun()-->Setting Auto Commit to
False..
[20210621 12:52:39 735]FCMainProcEJBBean.execFun()-->OracleConnection before
wrap:null
[20210621 12:52:39 735]FCMainProcEJBBean.execFun()-->OracleConnection in
if:weblogic.jdbc.wrapper.PoolConnection_oracle_jdbc_driver_T4CConnection@18743
ab
[20210621 12:52:39 736]FCMainProcEJBBean.execFun()--
>l_storedProcedureName=cspks_req_global_wrapper.Pr_Process_RequestJ***
[20210621 12:52:39 737]FCMainProcEJBBean.execFun()-->gatewayType::
[20210621 12:52:39 737]FCMainProcEJBBean.execFun()-->DB_TIMEOUT::0
[20210621 12:52:40 374]FCMainProcEJBBean.execFun()-->Executed Sucessfully
[20210621 12:52:40 375]FCMainProcEJBBean.execFun()-->EXT Call Required:null
[20210621 12:52:40 375]FCMainProcEJBBean.execFun()-->Calling Commit
[20210621 12:52:40 377]FCMainProcEJBBean.execFun()-->Commit is done
[20210621 12:52:40 377]FCMainProcEJBBean.execFun()-->Sucesfully returned from
Main Package
[20210621 12:52:40 377]FCMainProcEJBBean.execFun()-->Setting SetAutoCommit
true
[20210621 12:52:40 377]FCMainProcEJBBean.execFun()-->Setting of SetAutoCommit
done
[20210621 12:52:40 377]CspksReqHandler.processMsg()-->ADDLINFO
NAMES=KEY_ID>TANKED_DATA, ADDLINFO VALUES=~STTM_CUSTOMER~000022~>FALSE
[20210621 12:52:40 377]CspksCreator.create()-->In Create
[20210621 12:52:40 377]CspksCreator.create()-->Calling in CreateFcXml()..
[20210621 12:52:40 377]CspksCreator.createFcXml()-->Just Entered..

```

```

[20210621 12:52:40 377]CspksCreator.buildHeader()-->PSTATUS :S
[20210621 12:52:40 377]CspksCreator.buildHeader()-->Just Entered..
[20210621 12:52:40 377]CspksCreator.buildHeader()-->25956930 :: l_msgId :
[20210621 12:52:40 377]CspksCreator.buildHeader()-->FCUBS_MSG_ID :
6121172000076994
[20210621 12:52:40 377]CspksCreator.buildHeader()-->CORRELATION_PATTERN:
[20210621 12:52:40 377]CspksCreator.buildHeader()-->CORRELID 1:
[20210621 12:52:40 377]CspksCreator.buildHeader()-->CORRELID 2:*
[20210621 12:52:40 377]CspksCreator.buildHeader()-->pstatus ->S, Fcdata req :Y
[20210621 12:52:40 377]CspksCreator.buildErrorNodes()-->Just Entered..
[20210621 12:52:40 377]CspksReqHandler.processMsg()-->calling
cspksReqUtils.populateAuditLog()
[20210621 12:52:40 377]CspksReqHandler.processMsg()-->calling
cspksReqUtils.updateOvdRemarks()
[20210621 12:52:40 377]CspksReqUtils.updateOvdRemarks()-->Just
Entered6121172000076994/1/EXECUTEQUERY
[20210621 12:52:40 378]CspksReqUtils.updateOvdRemarks()-->Maker/Checkler
Remarks :/
[20210621 12:52:40 378]CspksReqUtils.updateOvdRemarks()-->calling
cspksReqWrapperDao.updateRemarksCstbRequestOvdMaster
[20210621 12:52:40 378]FCUtilityBean.updateRemarksCstbRequestOvdMaster()--
>Query : Update Cstb_Request_Ovd_Master set Maker_Remarks = Nvl(?,
Maker_Remarks),Checker_Remarks = Nvl(?, Checker_Remarks) where Msg_Id = ? AND
Req_Seq_No = ?
[20210621 12:52:40 378]CspksReqUtils.updateOvdRemarks()-->Maker and Checker
remarks Updated
[20210621 12:52:40 378]CspksReqUtils.updateOvdRemarks()-->Returning Success
in updateOvdRemarks>>>>>
[20210621 12:52:40 378]GwpksServiceRouter.ProcessReqMsg()-->Calling
logMessage()..
[20210621 12:52:40 379]FCUtilityBean.insertCstbsMsgLog()-->Query=insert into
cstbs_msg_log(msg_id,source_code,msg_type,request,
response,fc_response,status) values(?, ?, ?, ?, ?,?,?)
[20210621 12:52:40 383]GwpksServiceRouterMain.logMessage()-->Inserted CSPKS
Req and CSPKS Res
[20210621 12:52:40 383]GwpksServiceRouter.ProcessReqMsg()-->Calling
dedupeLogMsg()..
[20210621 12:52:40 383]FCHandlerMaintenance.processMsgUBS-->calling
postProcessMessage
[20210621 12:52:40 383]FCHandlerMaintenance.inside postProcessMsg
[20210621 12:52:40 383]FCHandlerMaintenance.processMsgUBS-->processing for
function=STDCIF
[20210621 12:52:40 383]FCHandlerMaintenance.processMsgUBS-->getting response
SUCCESS
[20210621 12:52:40 383]FCHandlerMaintenance.processMsgUBS-->calling
fcCtx.destroyContext
[20210621 12:52:40 383]FCJMISCEJBBean.getSMSDAO-->Creating connection
[20210621 12:52:40 383]FCJMISCEJBBean.getSMSDAO--> connection created
[20210621 12:52:40 383]SMSOracleDAO.getActionOnSequence-->select
SMSQS_SMS_ACTION_LOG.nextval from dual
[20210621 12:52:40 384]HostUpload.appendDebugs-->Calling insertActionLog
[20210621 12:52:40 384]FCJMISCEJBBean.getSMSDAO-->Creating connection
[20210621 12:52:40 384]FCJMISCEJBBean.getSMSDAO--> connection created
[20210621 12:52:40 384]FCJMISCEJBBean.insertActionLog-->sequence=169330
[20210621 12:52:40 384]FCJMISCEJBBean.insertActionLog-->msgStatus= SUCCESS
[20210621 12:52:40 384]FCJMISCEJBBean.insertActionLog-->Calling

```

```
insertSmtbActionLog
[20210621 12:52:40 384]SMSOracleDAO.insertSmtbActionLog-->Query=insert into
SMTB_SMS_ACTION_LOG (SEQUENCE_NO, ACTION_SEQUENCE_NO,ACTION, EXITFLAG,
DESCRIPTION,PKVALS, CURR_BRANCH, HOME_BRANCH, TXN_BRANCH,
REQ_TIME,OBIEE_STATUS,USER_ID,LOGIN_SEQUENCE_NO,MODULE_CODE,REQ_XML,RESP_XML)
values(?,?, ?, ?, ?,?,?,?,?,,?,,?,,?)
[20210621 12:52:40 388]FCClientHandler.doPost-->Checking respXML
[20210621 12:52:40 388]FCClientHandler.doPost-->calling updateToRES_FCUBS
[20210621 12:52:40 392]FCClientHandler.doPost-->PDP check - Time Captured
Successfully
[20210621 12:52:40 392]FCClientHandler.doPost-->PDP check - Logs Captured
Successfully
```

2

Error Code and Messages

This topic describes a list of error codes and their messages.

Table 2-1 Error Codes

Function ID	Error Code	Message
STDAPPDP	ST-PDP-001	Invalid Trace Value

3

Frequently Asked Questions

This topic provides a list of frequently asked questions.

Table 3-1 FAQs

Questions	Answers
What is the SMTB_USER_FUNCTION_TRACE table? and How are values inserted into the SMTB_USER_FUNCTION_TRACE table?	The SMTB_USER_FUNCTION_TRACE table is a data store for capturing the function IDs related to maintenance for the Performance Diagnostic Plugin functionality which stores Function ID , Action , User ID , and Trace Value . Based on these parameters, Time Lag Report is generated. The data can be captured using the newly designed function ID STDAPPDP .
What is the use of the CSTB_PARAM flag? and When will the CSTB_PARAM be set to Y (Enabled)?	The CSTB_PARAM table have a Performance Diagnostic Plugin (PDP) specific flag called TIME_LOG , which is used to enable or disable the functionality of PDP (that is capturing of any performance lag reports or process timings or even specific application log) in the system. By default, the value is set as N . Currently, the Performance Diagnostic Plugin works for OLTP processes only.
What is the mechanism used to obtain the Application Log?	Existing Infra level functionality is used to capture the application log by retaining any existing logs related to the user and capturing a new log based on a specific naming convention (<USERID>_<FUNCTIONID>_<ACTION>.log) for a particular operation specified during maintenance. Application logs are enabled/disabled automatically at the infra level and generated application log will be written on the server as a file.
Is trace value ALL a combination of other trace values? and If yes, can this combination be achieved by inserting multiple records with the required trace values?	Yes, trace value ALL is a combination of other trace values specified like the capture of process timings, and application log to be obtained in one go. The code is designed to pick only one trace value for a combination of User ID , Function ID , and Action , due to which multiple record maintenance is not allowed. These trace values help to keep the table clean and away from confusion for the Bank IT team during maintenance.
Is this PDP functionality specific to any product?	No, this enhancement is in the Infra Layer and it is supported in all ODT framework-based products processors.
Provide a brief insight on results obtained for each trace value.	For more information on results of the trace values, refer to the below Trace Value Results table.

Table 3-2 Trace Value Results

Trace Value	Process Timing	Application Log	Trace File	HPROF File
Process Timing at Database Level	Y	N	N	N
Process Timing at Application Level	Y	N	N	N
Application Logs	N	Y		N
ALL	Y	Y	Y	N