

# CVP Reporting Server Overview and Troubleshooting



Document ID: 116379

Contributed by Taylan Kucuk and Ramiro Amaya, Cisco TAC Engineers.

Sep 30, 2013

## Contents

### Introduction

#### Database Tables

Summary and Aggregate Tables

#### Differences Between CVP 8.x and 9.0 Reporting Database

#### Reporting Server Overload

#### Troubleshooting

Failure and Restoration – Reporting Server Partial

Reporting Server Failure

Database Connection Failure

Unfinished Calls

Recovery Mode

Trace Level and Log Files

Aggregator Debugs and Logs

Database Space Issues

## Introduction

This document gives an overview of the Cisco Unified Customer Voice Portal (CVP) reporting server and offers troubleshooting steps.

- The reporting service receives reporting data from the Cisco Unified IP Interactive Voice Response (IVR) service, the Session Initiation Protocol (SIP) service (if used), and the Cisco Unified CVP Voice XML (VXML) server. The service then transforms and writes the data to the Informix reporting database, which serves as historical reporting in a call center environment.
- The reporting service then parses those messages, converts them into batches of applicable Structured Query Language (SQL) statements, and executes them into an SQL database via the Java Database Connectivity (JDBC) API.

## Database Tables

CVP tables are categorized as:

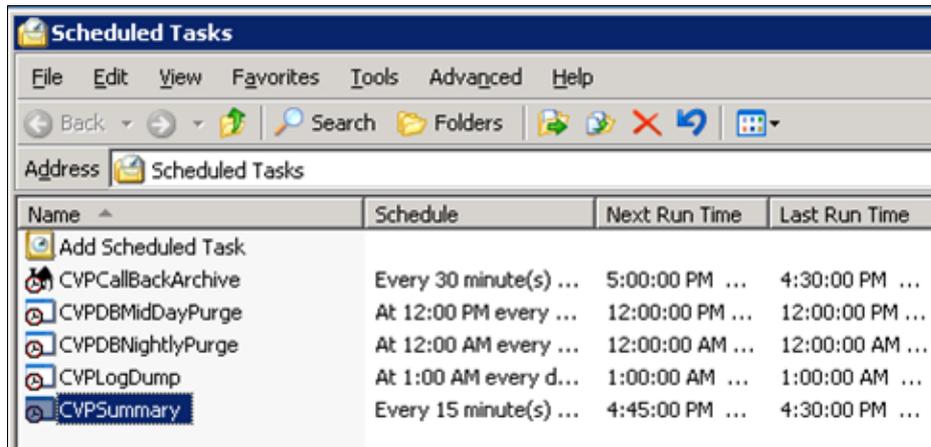
- Call tables
- VXML tables
- Summary / Aggregate tables
- Lookup and Reference tables
- Courtesy CallBack tables

Calls start at the Call table and are linked to the VXMLSession table by the CallGUID column.

*Note:* Unified CVP DateTimes are recorded as UTC time; the exception is dbdatetime, which is recorded as a local time. Localtimezoneoffset is a column in the Call table that contains the number of minutes offset from UTC in order to derive the Local Time. This can be used as an interval.

## Summary and Aggregate Tables

The unified CVP reporting server includes a summary process that aggregates data from the Call and VXMLElement tables into new summary tables.



Name	Schedule	Next Run Time	Last Run Time
Add Scheduled Task			
CVPCallBackArchive	Every 30 minute(s) ...	5:00:00 PM ...	4:30:00 PM ...
CVPDBMidDayPurge	At 12:00 PM every ...	12:00:00 PM ...	12:00:00 PM ...
CVPDBNightlyPurge	At 12:00 AM every ...	12:00:00 AM ...	12:00:00 AM ...
CVPLogDump	At 1:00 AM every d...	1:00:00 AM ...	1:00:00 AM ...
CVPSummary	Every 15 minute(s) ...	4:45:00 PM ...	4:30:00 PM ...

The reporting summary tables are:

- ApplicationSummary\_15
- ApplicationSummary\_Daily
- ApplicationSummary\_Weekly
- ApplicationSummary\_Monthly
- Call\_15
- Call\_Daily
- Call\_weekly

The tables are created based upon this schedule:

- Daily tables are populated one day behind the 15 minute tables.
- Weekly tables are populated from daily tables, once those have been fully populated for the week in question.
- Monthly tables are populated from weekly tables, once those have been fully populated for the month in question.

See Cisco Bug ID CSCue65248, "CVP Reporting summary tables are not populated." In the CVP reporting server, the summary tables are not populated. The issue is caused by the script for monthly summary, which was introduced in CVP 9.0.

## Differences Between CVP 8.x and 9.0 Reporting Database

The unified CVP 9.0(1) reporting database is supported only on the Windows 2008 R2 server. Because the unified CVP 8.x reporting database is supported by Windows 2003, there is no direct update to the unified CVP 9.0(1) reporting database.

For migration instructions, see the installation guide. Note that:

- The reporting server component cannot be installed if the machine is already part of the domain.
- The reporting server component cannot be installed on a server that is part of a domain. Installation of the component requires that local security policies take higher precedence than the same security policies in the group policy console, which is not possible if the server is part of a domain. However, after installation, the server can be added back to the domain because this does not require any local policies to be run on the server.

Differences in post installation tasks include:

- 4.x, 7.x reporting database – Log in as an Informix user, and run these batch files:
  - ◆ ReportingRunAsInformix.bat.
  - ◆ ReportingRunAsCVP\_DbAdmin.bat.
- 8.x reporting database – Log in as an Informix user, and run this batch file:
  - ◆ CVP\_Database\_Config.bat.
- 9.x reporting database – No batch files are required as part of the post installation process.

A key difference in users is that, with 9.x, there is no more Informix user. Instead, the cvp\_dbadmin user is the owner of the database.

## Reporting Server Overload

Cisco MCS-7845 reporting servers can handle 420 messages per second.

Use this equation in order to determine the number of reporting messages generated per second for each VoiceXML application:

$$A\# = \%CPS * CPS * MSG$$

where:

- %CPS = the percentage of calls that use this VoiceXML application
- CPS = the number of calls per second
- MSG = the number of reporting messages this application generates

Use this equation in order to add the messages generated by each application:

$$A(\text{total}) = A1 + A2 + \dots + An$$

where A(total) is the total number of reporting messages generated per second by your VoiceXML applications.

The number of reporting messages per element or activity is in Table 17 in of the Cisco Unified Customer Voice Portal (CVP) Solution Reference Network Design (SRND) Release 9.0(1).

For simplicity, you can use this query in order to calculate the average number of messages written to the vxmlsession table for one second:

```
select count(*)/86400 from vxmlsession where dbdatetime between  
'2012-12-12 00:00:00' and '2012-12-13 00:00:00'
```

Run this query against these 14 tables:

1. Call
2. CallEvent
3. CallICMInfo
4. OutgoingECCVariable (CCB related)
5. VXMLCustomContent
6. VXMLElement
7. VXMLElementDetail
8. VXMLElementFlag
9. VXMLError
10. VXMLHotEvent
11. VXMLHotLink
12. VXMLSession
13. VXMLSessionVariable
14. VXMLVoiceInteractDetail

Add the results in order to obtain the average number of messages per second received by the reporting server.

## Troubleshooting

If the reporting server is overloaded, the reporting logs contain these alerts:

```
CVP_8_0_RPT-1-REPORTING_DB_ALERT_RAISE ALERT!!!! The total JDBC messages queue size has exceeded the critical limit 300000 .... All the JDBC messages will be dropped. [id:4014]
```

```
CVP_8_0_RPT-1-REPORTING_DB_ALERT_RAISE ALERT!!!! The total JDBC messages queue size has exceeded the max limit 250000 .... Some of the JDBC messages may be dropped. [id:4014]
```

## Failure and Restoration – Reporting Server Partial

There are several scenarios where the reporting server goes to Partial Service. However, Partial Service does not necessarily mean that there is a problem.

### Reporting Server Failure

If the reporting server fails, messages destined for the reporting server are buffered by the Call Server, in memory, up to 200,000 messages. After that limit is reached, all new message detail information is dropped.

**Note:** For UCS platforms, you must modify the Number of Receive Buffers setting on the reporting server in order to maximize full reporting and call load on the virtual platform. If this setting is not modified after you install the OS, messages back up on the CallServer, and Message Queues fill up. You then see a sharp drop in the calls per second (CPS) rate.

Take these steps in order to set the Number of Receive Buffers on the Reporting Server TCP settings to 4096 (max):

1. On the reporting server, click **Control Panel** > **Network Connection**.
2. Right-click **Network Connection**.
3. Click the **Advanced** tab.
4. Under the Property tab, select **Number of Receive Buffers**.
5. In the Value text box, enter 4096.
6. Restart the reporting server.

Refer to Virtualization for Unified CVP: UCS Network Configuration for additional information.

## Database Connection Failure

If the database connection fails, the reporting server sends out a Simple Network Management Protocol (SNMP) alert and starts to store messages to a persistent file (%CVP\_HOME%\tmp\CVPReporting.tmp) up to a user-specified limit. During this time, the reporting server stays In Service. When 75% of the limit is reached, a warning is written to the log file. When 100% of the limit is reached, an SNMP alert is sent out, and the reporting server goes into Partial Service. Any new messages might be dropped.

When the database connection comes back up, the reporting server goes into recovery mode and changes its state to Partial Service (if it is not in that state already). It then starts to read messages from the %CVP\_HOME%\tmp\CVPReporting.tmp file and to commit them to the database. Depending upon the size of the file, it may take hours to commit all of the data to the database. New messages that come in during recovery are buffered in memory.

There is, however, a limit to the number of messages that the reporting server can buffer, regardless of the mode or state of the server:

- When the number of buffered messages reaches 100,000, an SNMP alert is sent out as a warning to the user.
- At 200,000 messages, another SNMP alert is sent out, detail information for all new messages is dropped, and only basic data like call, call event, and session information are kept. At 200,000 messages, the reporting server also changes its state to Partial Service (if it is not already in that state).
- When the total number of buffered messages reaches 300,000, another SNMP alert is sent out, and all new messages are dropped from that point forward.
- When the number of messages in memory drops back below 50,000, an SNMP alert is sent out that states the queue size is back to normal, and the state of the reporting server goes back to In Service.

If a persistent file exists on startup, the reporting server stays in Partial Service and goes into recovery mode.

## Unfinished Calls

The reporting server can also go to Partial Service when it is recovering unfinished calls.

This message is seen in the reporting server logs:

```
%CVP_8_0_RPT-1-REPORTING_STATE_CHANGE:  REPORTING Subsystem state changed to  
RPT SS RPT1 changes its state to Partial Service cause Unfinished calls  
recovery started [id:4001]
```

The logs also include information about recovery of these calls. Remember that the recovery process can take a long time!

```
%CVP_8_0_RPT-6-REPORTING_INFO:  Recover Uncompleted call: 73  
  CallGUID:90DAAAC91000013C01075FC253EF37A4 Event Id: 11 CauseId: 0 [id:4000]  
...  
%CVP_8_0_RPT-6-REPORTING_INFO:  Recover Uncompleted call:  
  129 CallGUID:673A58361000013C087A209E53EF37A5 Event Id: 0 CauseId: 0 [id:4000]
```

Once unfinished calls are completed, these messages are seen, and the reporting server goes back to In Service state:

```
%CVP_8_0_RPT-6-REPORTING_INFO:  Recover CallRegistry finished [id:4000]  
%CVP_8_0_RPT-6-REPORTING_INFO:  initKeepAliver() -- processed unfinished calls  
  [id:4000]
```

```
%CVP_8_0_RPT-1-REPORTING_STATE_CHANGE:  REPORTING Subsystem state changed to RPT
SS RPT1 changes its state to In Service cause Normal Operation [id:4001]
```

## Recovery Mode

You can remove the %CVP\_HOME%\tmp\CVPReporting.tmp file in order to avoid the recovery process and bring the reporting server back in service. This procedure describes how to bypass the recovery process:

1. Stop the CVP Call Server service.
2. Navigate to %CVP\_HOME%\tmp.
3. Rename the CVPReporting.tmp file.
4. Start the CVP Call Server service.

See Cisco Bug ID CSCtu43570, "CVPReporting.tmp grows beyond size limit and is not timely recovered." New call reporting data was lost because the file could not be completely read in. The hard drive was filling up, which eventually caused an 'out of disk space' condition.

This issue was fixed in the Unified CVP 8.5(1)SR18 and 8.5(1)SR6 reporting database.

## Trace Level and Log Files

Edit the <install\_dir>\Cisco\CVP\conf\reporting.properties file in order to set the trace level in the reporting server logs. This is an example:

```
RPT.traceMask = 0x810000
RPT.logLevel = DEBUG
```

## Aggregator Debugs and Logs

The summaries use two tables in the ciscoadmin database: agg\_schedule and agg\_statements.

The <CVP\_HOME>\logs\reporting.txt file shows whether aggregation has run.

This procedure describes how to enable additional tracing for the aggregator.bat job:

1. Wait for a minimum of 15 minutes.
2. Edit the <CVP\_HOME>\bin\aggregator.bat file. Change the value of the .bat file from this default:

```
echo call sp_sched_agg(); | dbaccess ciscoadmin

to:

echo call sp_sched_agg('D'); | dbaccess ciscoadmin
```

Debug logs are written in the CVP\_HOME>\logs\Agg\_Debug.out file.

## Database Space Issues

**Note:** This space issue has been seen on different implementations. Because the solution to this issue involves deep technical knowledge of Informix, contact the Cisco Technical Assistance Center (TAC) if you encounter this problem.

This procedure describes the troubleshooting process:

1. Upgrade to the Unified CVP 8.0(1) reporting database.
2. Run the CVP\_Database\_Config.bat file, and see the issue:

```
C:\Cisco\CUP\bin>CVP_Database_Config.bat
23:41:54 Wed Dec 19 2012 dbaccess sysmaster c:/temp/cvpck.sql 2>NUL

Modifying Locale for Informix shortcut
C:\Cisco\CUP\bin\chg.exe C:\db\informix\cvp_db_wp17cvprpt1a.cmd /F:"CP1252" /R:"
UTF8" > NUL
C:\Cisco\CUP\bin\chg.exe C:\db\informix\cvp_db_wp17cvprpt1a.cmd /F:"8859-1" /R:"
UTF8" > NUL
C:\Cisco\CUP\bin\chg.exe C:\db\informix\cvp_db_wp17cvprpt1a.cmd /F:"codepage sel
ect=1252" /R:"codepage select=850" > NUL
dbaccess sysmaster c:/temp/cvpck.sql 2>NUL
dbaccess cvp_data C:\Cisco\CUP\informix_frag\upg_est.sql
dbaccess cvp_data c:/temp/cvpupg.sql 2>NUL
No such file or directory
```

3. Review the c:\temp\cvpupg.sql file. The content is:

```
call upg_est(); UNLOAD to "c:/temp/upgvars.out" SELECT estimate1,estimate2,
retention,log_space_needed,minlog,maxlog FROM cvp_data:upg_estimate;
```

4. Connect to the database. When you attempt to check the *upg\_estimate* table, it does not exist.

```
c:\DBAccess
SQL: _ New Run Modify Use-editor Output Choose Save Info Drop Exit
Modify the current SQL statements using the SQL editor.
----- cvp_data@cvp_db_wp17cvprpt+ ----- Press CTRL-W for Help -----
select * from cvp_data:upg_estimate

206: The specified table (upg_estimate) is not in the database.
111: ISAM error: no record found.
```

5. Check the c:\Temp\CiscoUnifiedCVPDB.log file, and see these errors:

```
23:41:54 Wed Dec 19 2012 : dbaccess cvp_data
C:\Cisco\CUP\informix_frag\upg_est.sql
Database selected.
312: Cannot update system catalog (sysprocbody).
131: ISAM error: no free disk space
Error in line 26
Near character position 11
```

```
23:41:54 Wed Dec 19 2012 : dbaccess cvp_data c:/temp/cvpupg.sql 2>NUL
Database selected.
206: The specified table (upg_estimate) is not in the database.
```

6. Check upg\_est.sql. The content of the upg\_estimate table is:

```
SELECT COUNT(*)
  INTO tmp_int
  FROM systables
  WHERE tabname='upg_estimate';

IF tmp_int=0 THEN
  CREATE TABLE upg_estimate (
    estimate1 INTERVAL HOUR TO MINUTE,
    estimate2 INTERVAL HOUR TO MINUTE,
    retention SMALLINT,
    log_space_needed INTEGER,
    minlog INTEGER,
    maxlog INTEGER
  );
```

But, this table does not get created.

7. Connect to DBAccess, and run this query:

```
SELECT COUNT(*) FROM systables WHERE tabname='upg_estimate';
```

The query returns 0, so the table should have been created.

8. Try to create the table manually in DBAccess:

```
CREATE TABLE upg_estimate (
  estimate1 INTERVAL HOUR TO MINUTE,
  estimate2 INTERVAL HOUR TO MINUTE,
  retention SMALLINT,
  log_space_needed INTEGER,
  minlog INTEGER,
  maxlog INTEGER
);
```

You receive the error message:

```
261: Cannot Create file for table (informix.upg_estimate).
131: ISAM error: no free disk space
```

9. Run the *onstat -d* command, and verify that the last chunk does not have enough space:

```

E:\IFMXDATA\cvp_db_wpl7cvprpt1a>
E:\IFMXDATA\cvp_db_wpl7cvprpt1a>onstat -d

IBM Informix Dynamic Server Version 10.00.TC10 -- On-Line -- Up 06:54:31 -- 1657824 Kbytes

Dbspaces
address number flags fchunk nchunks pgsize flags owner name
8010E7E8 1 0x40001 1 1 4096 N B informix rootdbc
8085B498 2 0x40001 2 1 4096 N B informix cvp_pria
8085B5F8 3 0x40001 3 1 4096 N SB informix sbspace
8085B758 4 0x40001 4 1 4096 N TB informix cvp_temp_dbSPACE
8085B8B8 5 0x40001 5 1 4096 N B informix cvp_llog_dbSPACE
8085BA18 6 0x40001 6 1 4096 N B informix cvp_plog_dbSPACE
8085BB78 7 0x40001 7 1 4096 N B informix cvp_data_dbSPACE
? active, 2047 maximum

Chunks
address chunk/dbs offset size free bpages flags pathname
8010E948 1 1 0 12800 10659 PO-B E:\IFMXDATA\cvp_db_wpl7cvprpt1a\rootdbc_dat.000
8029C808 2 2 0 12800 12747 PO-B E:\IFMXDATA\cvp_db_wpl7cvprpt1a\cvp_pria_dat.000
8029CC88 3 3 0 12800 11879 PO-SB E:\IFMXDATA\cvp_db_wpl7cvprpt1a\sbspace_dat.000
Metadata 868 554 868
8029CE08 4 4 0 1048576 1048523 PO-B E:\ifmxdata\cvp_db_wpl7cvprpt1a\cvp_temp_dat.000
8085BB18 5 5 0 1150976 50123 PO-B C:\IFMXDATA\CVP_LLOGS\cvp_llog_dat.000
8085B198 6 6 0 287744 4947 PO-B E:\ifmxdata\cvp_db_wpl7cvprpt1a\cvp_plog_dat.000
8085E318 7 7 0 18350080 205 PO-B E:\ifmxdata\cvp_db_wpl7cvprpt1a\cvp_data_dat.000
? active, 32766 maximum

NOTE: The values in the "size" and "free" columns for DbSPACE chunks are
displayed in terms of "pgsize" of the DbSPACE to which they belong.

Expanded chunk capacity node: always

```

10. Log in to the CVP Informix box as 'Informix' user.

11. Create a new empty file called new\_space under E:\ifmxdata\cvp\_db\_wpl7cvprpt1a\ . Open a command prompt, and enter these commands. Do not copy and paste:

```

onspaces -a cvp_data_dbSPACE -
E:\ifmxdata\cvp_db_wpl7cvprpt1a\cvp_data_dbSPACE\new_space -o 0 -s 10240

```

**Note:** -s denotes the size of the new chunk and is given in KBs. For 100 MB, it is 100 \* 1024 = 102400.

This command adds 100 MB of dbSPACE to the CVP Informix server.

12. Rerun the CVP\_Database\_Config.bat file. It will now work properly.

This example shows how to connect to the database with DBAccess:

```
DBAccess
SELECT DATABASE SERVER >>
Select a server with the Arrow Keys, or enter a name, then press Return.
----- Press CTRL-W for Help -----
cvp_db_cvp
```

```
DBAccess
USER NAME >> cvp_dbuser_
Enter the login name you want to use for this connection.
----- Press CTRL-W for Help -----
cvp_db_cvp
```

```
DBAccess
SELECT DATABASE >>
Select a database with the Arrow Keys, or enter a name, then press Return.
----- @cvp_db_cvp ----- Press CTRL-W for Help -----
callback@cvp_db_cvp
cisoadmin@cvp_db_cvp
cvp_data@cvp_db_cvp
sysmaster@cvp_db_cvp
sysuser@cvp_db_cvp
sysutils@cvp_db_cvp
```

```
DBAccess
DBACCESS: Query-language Connection Database Table Session Exit
Use SQL query language.
----- cvp_data@cvp_db_cvp ---- Press CTRL-W for Help -----
```

```
DBAccess
SQL: New Run Modify Use-editor Output Choose Save Info Drop Exit
Enter new SQL statements using SQL editor.
----- cvp_data@cvp_db_cvp ---- Press CTRL-W for Help -----
```

```
DBAccess
NEW: ESC = Done editing CTRL-A = Typeover/Insert CTRL-R = Redraw
CTRL-X = Delete character CTRL-D = Delete rest of line
----- cvp_data@cvp_db_cvp ---- Press CTRL-W for Help -----
select max(dbdatetime) from UXMLElement_
```

```
DBAccess
SQL: New Run Modify Use-editor Output Choose Save Info Drop Exit
Run the current SQL statements.
----- cvp_data@cvp_db_cvp ---- Press CTRL-W for Help -----
<max>
2012-12-04 12:53:30.000
```