

Confluent Oracle XStream CDC Source Connector - Status check

To prepare for our upcoming discussion about the Confluent Oracle XStream CDC Source Connector, which utilizes the XStream (GoldenGate) API, please answer the following questions to help us assess your requirements:

Oracle related questions.....	1
Confluent Related.....	5
General Requirements.....	7
CDC related.....	8
Scripts and Queries for Information Retrieval.....	9
How many core used by database.....	9
Long running transactions.....	9
Redolog Structure.....	10
measure the archive log generation.....	11
Table Growth.....	12
Log switches.....	12
All columns with LOB support data types:.....	14
All tables with having LOB data > 20MB and the biggest sizes of LOB cols:.....	14
SGA settings.....	17
Send Results back to Confluent.....	18

Oracle related questions

Does this database host a vendor application?

If not, please briefly describe its function.

If yes, do you possess a full Oracle license?

[Answer](#)

Please specify the number of CPU cores utilized by your Oracle Database Instance. If this information is unknown, a script can be run to determine it (details provided later in this document).

[Answer](#)

What Oracle version are you using?

[Answer](#)

What Oracle Edition is being used (Standard or Enterprise)?

[Answer](#)

Select the database type: NON-CDB, CDB, or CDB and PDB.

[Answer](#)

What Oracle features and options are enabled (e.g., Data Guard, Active Data Guard, RAC, Exadata, Partitioning, etc.)?

[Answer](#)

Is advanced security enabled, including EUS, Database Vault, and Label Security?

[Answer](#)

Is the database instance self-managed, or is it running on Oracle Cloud Infrastructure (OCI) as a cloud service or with other cloud providers like AWS RDS?

[Answer](#)

Do you periodically change the DDL (Data Definition Language) of objects you intend to capture, such as dropping or renaming columns?

[Answer](#)

How do you authenticate with your Oracle database (e.g., using a username and password, Kerberos, or other methods)?

[Answer](#)

How many database instances require CDC integration?

[Answer](#)

How many tables per database need to be integrated with the CDC Connector? Is the number of tables for capture:

- <10
- <100
- All tables
- Don't know

[Answer](#)

Have integrated tables experienced growth? (Refer to the script below for details.)

[Answer](#)

What is the change frequency of your tables (e.g., lookup tables, run tables)?

[Answer](#)

For the largest tables intended for capture, what are the approximate row and column counts? Additionally, what is the size of a single record (e.g., 10,000,000 records at 560 Bytes each)?

[Answer](#)

Do any of these tables contain special data formats such as BLOB, LOB, CLOB, JSON, XMLType, LONG, or LONG RAW? (Refer to script for details.)

[Answer](#)

Do any of these specially formatted tables exceed 20MB in content? (Refer to script)

[Answer](#)

Could you provide details on the Redolog setup? Specifically, how many groups and members are configured, and what is the size of each member? This information can be found in the script (see later) or an AWR Report or in OEM or SQL Developer.

[Answer](#)

What is the daily redo log data change throughput in GB? (Refer to the script below.)

[Answer](#)

What is the peak hourly redo log data change throughput in GB? (Refer to the script below.)

[Answer](#)

How many Redo Log switches occur per hour? (Refer to the script below for details.)

[Answer](#)

What is the typical size of your transactions? For example, does one transaction typically consist of a single record, or a batch of 100,000 records?

[Answer](#)

Is ARCHIVELOG currently enabled for this instance?

[Answer](#)

Are there long-running transactions in your database, and if so, what duration constitutes "long" (e.g., 5 minutes, 3 hours, 5 days)?

[Answer](#)

Do you have experience with XStream Out Concepts, including performance and monitoring? See [Oracle documentation](#) for more details.

[Answer](#)

Will your DBA be fully integrated into the CDC project?

[Answer](#)

The XStream configuration must be set up on your database. This involves:

- Enabling XStream and configuring ARCHIVELOG mode.
- Configuring supplemental logging (full or partial).
- Setting up two users as detailed in the product documentation of Confluent:
 - An XStream administrator for the capture process, who is not required outside the source database.
 - An XStream user with minimal privileges, requiring only access to consume XStream API output for the connector.
- Verifying the SGA configuration. Please provide the values of the relevant SGA parameters (see script).

We will require assistance from your DBA team for this setup. Are there any issues with this?

[Answer](#)

Confluent Related

Do you utilize Confluent Platform, Confluent Cloud, or a hybrid of both?

[Answer](#)

If you intend to run self-managed connectors to Confluent Cloud, please refer to this documentation:

<https://docs.confluent.io/cloud/current/cp-component/connect-cloud-config.html>

[Answer](#)

What version of Confluent Platform are you currently using?

[Answer](#)

Will the Oracle instance and Confluent Connect be in the same location (e.g., both on-premise or both in the same cloud service provider), or will they be in a hybrid configuration (e.g., on-premise Oracle and AWS/Confluent Cloud)?

[Answer](#)

Is your cluster a Confluent Cloud fully managed one?

[Answer](#)

Will you be running the Confluent Oracle XStream CDC Connector as a fully-managed connector in Confluent Cloud?

[Answer](#)

Are you planning to deploy the Oracle CDC Connector as self-managed within an existing Connect cluster? Please provide details on the number of workers, deployment location and type (e.g., Kubernetes), and sizing specifications.

[Answer](#)

Is resizing the self-managed Connect cluster for the CDC use case part of your plans?

[Answer](#)

Are you familiar with the Confluent Oracle XStream CDC Connector? You can find documentation for Confluent Platform [here](#) and for Confluent Cloud [here](#).

YES/NO

Are there any known concerns regarding the [Confluent Oracle CDC Requirements of CP](#) or [CCloud features](#)? If so, please specify which ones.

[Answer](#)

What type of client application consumes the CDC table topics: an application, a JDBC Sink Connector, a replicator/cluster-link, or another client?

[Answer](#)

Are you planning to utilize Databricks or another modern cloud data warehouse for CDC table topics?

[Answer](#)

What are the prerequisites for using this connector?

The following are required:

- Oracle Enterprise Edition 19c or newer
- Java 17 for the connector worker node
- Confluent Platform 7.6 or higher

Are there any issues with these requirements?

[Answer](#)

General Requirements

What are your specific throughput requirements for the Oracle XStream CDC Connector (e.g., in MB/s or records per second)?

[Answer](#)

Are there specific latency requirements?

[Answer](#)

Regarding networking, are there any specific requirements concerning the locations of the database and the Confluent Cluster?

[Answer](#)

Will the Confluent Oracle XStream CDC Connector increase the workload on the database?

[Answer](#)

Will presenting both "before" and "after" records, transformed by the CDC Connector, in table topics meet your requirements, or do you need a flat structure in the topics, similar to your database tables?

[Answer](#)

Are DDL changes, specifically common ones on your tables, to be included in the CDC process? If so, please list them.

[Answer](#)

CDC related

Is a CDC solution currently implemented? If so, please specify which one. Are you encountering technical issues? If so, could you describe them?

[Answer](#)

Is your Oracle database license a Full Use license?

[Answer](#)

Do you have a GoldenGate license? If so, would you be interested in using it with the Confluent XStream Oracle CDC Connector? (Note: This is currently not possible.)

[Yes/No](#)

To continue using the Confluent CDC solution, what features are essential from your current setup? Additionally, please describe the primary use cases of your existing CDC solution, such as data replication or streaming.

[Answer](#)

Are you currently using the legacy Confluent Logminer CDC Connector? If so, are there plans to decommission the current Oracle CDC connector? Should you require assistance with the migration from the old to the new connector, a comprehensive migration guide will be available shortly.

[Answer](#)

Scripts and Queries for Information Retrieval

The script located at [XStream Info Gathering Script](#) contains several queries, including those listed below. Please review and execute this script.

How many core used by database

Customers can obtain the CPU count for each database by executing a straightforward command. Note that every 2 CPUs are equivalent to 1 core in most cases.

```
None
-- Amount of CPUs
select value  from v$parameter where name like 'cpu_count';
--or
SELECT TO_CHAR(value) num_cpus FROM v$osstat WHERE stat_name = 'NUM_CPUS';
-- amount of cores
SELECT TO_CHAR(value) num_cores FROM v$osstat WHERE stat_name =
'NUM_CPU_CORES';
-- check the socket
SELECT TO_CHAR(value) num_sockets FROM v$osstat WHERE stat_name =
'NUM_CPU_SOCKETS';
-- or
select value  from v$parameter where name like 'cpu_count';
```

Long running transactions

It is crucial to monitor long-running transactions in the customer database. Uncommitted data from these transactions requires special handling.

```
None
-- Open Transactions = Active
select s.username,
       s.program,
       s.status as sessionstatus,
```

```
        t.status as transactionstatus,
        t.name, t.START_SCN,
        t.START_TIME,
        to_char(sysdate, 'MM/DD/YY HH24:MI:SS') as current_time,
        o.sql_text
    from v$session s, v$transaction t, v$open_cursor o
    where s.taddr = t.addr
        and t.status='ACTIVE'
        and o.sql_id = s.prev_sql_id;
```

```
None
-- or checking
select * from V$SESSION_LONGOPS;
```

Redolog Structure

```
None
-current size of redolog files
SELECT
    a.group#,
    substr(b.member,1,30) name,
    a.members,
    a.bytes/1024/1024 as MB,
    a.status
FROM
    v$log      a,
    v$logfile b
WHERE
    a.group# = b.group#;
```

measure the archive log generation

When calculating throughput, remember to update the **date** in the **WHERE** clause to reflect the current date.

None

```
-- We need to cover the peaks
-----
--Hourly Archive generation from 01-OCT-2024 to 06-NOV-2024
-----

set lines 300 pages 300
set num 6
col Day for a9
SELECT
  TRUNC(COMPLETION_TIME), THREAD#, TO_CHAR(COMPLETION_TIME, 'Day') Day,
  COUNT(1) "Count Files",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '00', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H0",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '01', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H1",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '02', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H2",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '03', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H3",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '04', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H4",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '05', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H5",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '06', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H6",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '07', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H7",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '08', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H8",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '09', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H9",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '10', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H10",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '11', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H11",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '12', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H12",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '13', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H13",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '14', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H14",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '15', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H15",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '16', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H16",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '17', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H17",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '18', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H18",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '19', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H19",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '20', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H20",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '21', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H21",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '22', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H22",
  ROUND(SUM(DECODE(TO_CHAR(COMPLETION_TIME, 'HH24'), '23', ((BLOCKS * BLOCK_SIZE) / 1024 / 1024), 0))) "H23",
  ROUND(SUM(BLOCKS * BLOCK_SIZE) / 1024 / 1024) "Total Size (MB)"
FROM V$ARCHIVED_LOG
WHERE COMPLETION_TIME BETWEEN TO_DATE('01/10/2024', 'DD/MM/YYYY') AND TO_DATE('06/11/2024', 'DD/MM/YYYY')
GROUP BY TRUNC(COMPLETION_TIME), THREAD#, TO_CHAR (COMPLETION_TIME, 'Day')
ORDER BY 1;
```

This script calculates the hourly generation of archive files in MB. By identifying the peak value, you can determine the necessary throughput capacity.

Table Growth

Select the tables for Change Data Capture (CDC). Note that the **DBA_HIST_SEG_STAT** view is part of a Management Pack and requires an Oracle license.

None

```
-- connect
sys/confluent123@orclcdb as sysdba
SQL> set lines 200
COLUMN owner FORMAT A15
COLUMN OBJECT_NAME FORMAT A25
COLUMN SUBOBJECT_NAME FORMAT A15
COLUMN OBJECT_TYPE FORMAT A15
COLUMN NAME FORMAT A15
COLUMN bytes HEADING 'Megabytes' FORMAT 9999999
SELECT o.OWNER , o.OBJECT_NAME , o.SUBOBJECT_NAME , o.OBJECT_TYPE ,
       t.NAME "Tablespace Name", s.growth/(1024*1024) "Growth in MB",
       (SELECT sum(bytes)/(1024*1024)
        FROM dba_segments
        WHERE segment_name=o.object_name) "Total Size(MB)"
FROM DBA_OBJECTS o,
     ( SELECT TS#,OBJ#,
           SUM(SPACE_USED_DELTA) growth
       FROM DBA_HIST_SEG_STAT
       GROUP BY TS#,OBJ#
       HAVING SUM(SPACE_USED_DELTA) > 0
       ORDER BY 2 DESC ) s,
     v$tablespace t
WHERE s.OBJ# = o.OBJECT_ID
AND o.OBJECT_TYPE = 'TABLE'
AND s.TS#=t.TS#
ORDER BY 6 DESC
/
# OUTPUT
# OWNER      OBJECT_NAME  OBJECT_TYPE  Tablespace Name  Growth in MB  Total Size(MB)
# -----
# ORDERMGMT  REGIONS     TABLE       USERS            63.3485899    88
```

This script helps in sizing the connector by showing the amount of data in tables, measured in MB.

Log switches

What is the current number of log switches per Hour?

None

-- You can view the redo_buffer_allocation_retries in V\$SYSSTAT to see if your redo log files are too small.

```
select * from V$SYSSTAT where name = 'redo_buffer_allocation_retries';
```

-- shows you the time between log switches

```
select  b.recid,
        to_char(b.first_time, 'dd-mon-yy hh:mi:ss') start_time,
        a.recid,
        to_char(a.first_time, 'dd-mon-yy hh:mi:ss') end_time,
        round(((a.first_time-b.first_time)*25)*60,2) minutes
from    v$log_history a, v$log_history b
where   a.recid = b.recid + 1
order  by a.first_time asc;
```

-- log switches

```
SELECT
    substr(to_char(first_time, 'DD-MON-YYYY HH24:MI:SS'),13, 2) as Hour ,
    substr(to_char(first_time, 'DD-MON-YYYY HH24:MI:SS'),1, 12) as Day ,
    count(1) as Nb_switch_per_hour
FROM (
    SELECT DISTINCT CAST(FIRST_TIME AS VARCHAR2(20)), FIRST_TIME,
    FIRST_CHANGE#, NEXT_TIME, NEXT_CHANGE#,BLOCKS, BLOCK_SIZE,BLOCKS*BLOCK_SIZE AS
    BLOCK_TOT, ARCHIVED, END_OF_REDO, 1 AS Nb_redo
    FROM v$archived_log
    WHERE CAST(FIRST_TIME AS VARCHAR2(20)) LIKE '%1%-APR-23%'
)
GROUP BY
    substr(to_char(first_time, 'DD-MON-YYYY HH24:MI:SS'),13, 2),
    substr(to_char(first_time, 'DD-MON-YYYY HH24:MI:SS'),1, 12)
--WHERE ROWNUM < 10
ORDER BY
    Hour ASC;
```

Most redo generation and archive statistics can be retrieved from the Automated Workload Repository (AWR), which is a repository of STATSPACK reports.

Keep in mind that a redo log switch forces a checkpoint, which can degrade database performance. You can check `redo_buffer_allocation_retries` in `V$SYSSTAT` to determine if your redo log files are undersized.

All columns with LOB support data types:

None

```
select OWNER||'.'||TABLE_NAME||'('||COLUMN_NAME||' '||DATA_TYPE||')' as
mycolumns from all_tab_columns where OWNER='ORDERMGMT' and data_type in
('CLOB','BLOB','NCLOB','LONG','LONG RAW','XMLTYPE','JSON');
```

All tables with having LOB data > 20MB and the biggest sizes of LOB cols:

Update your Schema User to: **ordermgmt (please use your own schema)**.

This script identifies records exceeding 20MB in size and displays the largest column size for each table.

None

```
-- give test user unlimited access to tablespace in my case USERS TS
connect sys/confluent123@XEPDB1 as sysdba
alter user ordermgmt quota unlimited on users;

-- Create test data in my sample DB (19c), to see that this script is working
connect ordermgmt/kafka@orclpdb1
-- create test table
create table testclob
(id number
 name varchar2(100),
 text clob)
```

```
/

-- generate 21MB data in CLOB
declare
    -- generate 21MB Data
    c clob;
begin
    for i in 1..1024
        loop
            c := c||rpad('a', 21504, 'a');
        end loop;
    insert into testclob values (1,'CLOB1', c);
    commit;
end;
/

-- generate 10 MB clob data
declare
    -- generate 10MB Data
    c clob;
begin
    for i in 1..1024
        loop
            c := c||rpad('a', 10240, 'a');
        end loop;
    insert into testclob values (2,'CLOB2', c);
    commit;
end;
/

-- 30 MB
declare
    -- generate 30MB Data
    c clob;
begin
    for i in 1..1024
```

```
        loop
            c := c||rpad('a', 30720, 'a');
        end loop;
        insert into testclob values (3,'CLOB3', c);
        commit;
    end;
/

-- #####
-- ## Check if there are tables having data > 20MB

DECLARE
    v_sql VARCHAR2(4000);
    v_table VARCHAR2(128);
    v_column VARCHAR2(128);
    v_count NUMBER;
BEGIN
    -- show all LOB tables in Schema ORDERMGMT
    FOR rec IN (
        SELECT table_name, column_name
        FROM all_tab_columns
        WHERE owner = 'ORDERMGMT' AND data_type = 'CLOB'
        ORDER BY table_name
    ) LOOP
        v_table := rec.table_name;
        v_column := rec.column_name;
        v_count := 0; -- Default count
        -- Construct dynamic SQL query for show records > 20MB
        BEGIN
            v_sql := 'SELECT COUNT(*) FROM ORDERMGMT.' || v_table ||
                ' WHERE DBMS_LOB.GETLENGTH(' || v_column || ') > 20971520';
            EXECUTE IMMEDIATE v_sql INTO v_count;
            -- Print only if large records exist
            IF v_count > 0 THEN
                DBMS_OUTPUT.PUT_LINE('Table: ' || v_table || ', Large Records (>
20MB): ' || v_count);
            END IF;
        END;
```



```
        END IF;
    EXCEPTION
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE('Skipping ' || v_table || ': ' || SQLERRM);
    END;
    -- Get the biggest size of CLOB column in these tables
    BEGIN
        v_sql := 'SELECT MAX(DBMS_LOB.GETLENGTH(' || v_column || ')) FROM
ORDERMGMT.' || v_table ;
        EXECUTE IMMEDIATE v_sql INTO v_count;
        -- Print max clob size
        IF v_count > 0 THEN
            DBMS_OUTPUT.PUT_LINE('Table: ' || v_table || ', Column: ' ||
v_column || ' has max LOB size of ' || v_count/1024/1024 || 'MB');
        END IF;
    EXCEPTION
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE('Skipping ' || v_table || ': ' || SQLERRM);
    END;
    END LOOP;
END;
/
```

SGA settings

Run the script to display the SGA settings for your SID/DB Instance.

None

```
PROMPT ==== displays SGA Setup Part 1====
show parameter sga_max_size
show parameter sga_min_size
show parameter sga_target
show parameter pga_aggregate_target
```

```
PROMPT ==== displays Automatic Memory Management (AMM) Setup Part 2  
(MEMORY_TARGET, MEMORY_MAX_TARGET > 0 if enabled)====  
show parameter MEMORY_TARGET  
show parameter MEMORY_MAX_TARGET
```

Send Results back to Confluent

Kindly forward your responses to your account team.
Thank you for your support and time.