**PETER-XMLTYPE-ERROR:**

**ORA-21780**: Maximum number of object durations exceeded.

ORA-06512: at "SYS.XMLTYPE", line 343

ORA-06512: at line 1

**Versions 10.2, 11.1, 11.2, 12.1**

**Error:** ORA-21780 Maximum number of object durations exceeded.

---------------------------------------------------------------------------

**Cause:** This typically happens if there is infinite recursion in the PL/SQL

function that is being executed.

**Action:** User should alter the recursion condition in order to prevent infinite

recursion.

**Running a procedure having a loop of more than 65540 cycles which executes the query**

SELECT XMLELEMENT("ROOT\_COLLECTION",XMLAGG(T.node)) INTO v\_xml   
FROM ( SELECT COLUMN\_VALUE AS node  
FROM TABLE (CAST(v\_xml\_table AS XML\_TABLE))) T;

raises the error  
ORA-21780: Maximum number of object durations exceeded  
when the loop reaches about 65540 cycles.  
  
**The issue reproduces on 8bit character set such as WE8ISO8859P15 only, it does not reproduce on AL32UTF8.**

**How to recreate the SYS.ALERT\_QUE (Doc ID 430146.1)**

**In this Document**

|  |  |
| --- | --- |
|  | [Goal](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=176700433604728&parent=DOCUMENT&sourceId=1513651.1&id=430146.1&_afrWindowMode=0&_adf.ctrl-state=14lbw2hy2i_178#GOAL) |

|  |  |
| --- | --- |
|  | [Solution](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=176700433604728&parent=DOCUMENT&sourceId=1513651.1&id=430146.1&_afrWindowMode=0&_adf.ctrl-state=14lbw2hy2i_178#FIX) |

|  |  |
| --- | --- |
|  | [References](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=176700433604728&parent=DOCUMENT&sourceId=1513651.1&id=430146.1&_afrWindowMode=0&_adf.ctrl-state=14lbw2hy2i_178#REF) |

**APPLIES TO:**

Oracle Database - Enterprise Edition - Version 12.1.0.2 to 19.1.0.0.0 [Release 12.1 to 18]  
Oracle Database Cloud Schema Service - Version N/A and later  
Oracle Database Backup Service - Version N/A and later  
Oracle Database Cloud Service - Version N/A and later  
Oracle Database Exadata Cloud Machine - Version N/A and later  
Information in this document applies to any platform.

**GOAL**

"Oracle Database - Enterprise Edition - Version 10.2.0.1 to Release 19.1

The purpose of this article is to outline the steps for recreating the SYS.ALERT\_QUE  
  
The ALERT\_QUE is used by the Grid Control and DB Control Management Agents to monitor server-generated alerts

NOTE:  
By running the script below, the queue tables will be recreated and the messages  
in the queue will be lost.  
  
For 11g you can use catmwin.sql which has the steps to recreate the ALERT\_QT.  
Alternatively, for 11g or 12c you can use the catproc.sql to recreate.

**SOLUTION**

SQL> connect / as sysdba  
  
SQL>alter system enable restricted session;  
  
To drop the ALERT\_QT queue\_table execute:

exec dbms\_aqadm.drop\_queue\_table('ALERT\_QT',true)

To re-create tables, sequence, type and queue for server alert in version 10.2:  
  
SQL>@$ORACLE\_HOME/rdbms/admin/dbmsslrt.sql  
SQL>@$ORACLE\_HOME/rdbms/admin/catalrt.sql

To re-create tables, sequence, type and queue for server alert in version 11 and above:  
  
SQL>@$ORACLE\_HOME/rdbms/admin/dbmsslrt.sql  
SQL>@$ORACLE\_HOME/rdbms/admin/catalrt.sql  
SQL>@$ORACLE\_HOME/rdbms/admin/catmwin.sql  
SQL>@$ORACLE\_HOME/rdbms/admin/exechae.sql

To recompile the invalid objects  
  
SQL>@$ORACLE\_HOME/rdbms/admin/utlrp.sql  
  
SQL> alter system disable restricted session;

Note: This option may leave DBSNMP.MGMT\_BSLN\_INTERNAL invalid. To validate the same run catsnmp.sql

OR

SQL> connect / as sysdba  
  
SQL>alter system enable restricted session;  
  
To drop server alert schema.  
  
SQL>@$ORACLE\_HOME/rdbms/admin/catnoalr.sql  
  
Rerun catproc.sql  
  
SQL>@$ORACLE\_HOME/rdbms/admin/catproc.sql  
  
SQL> alter system disable restricted session;

**REFERENCES**

[NOTE:603289.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=430146.1&id=603289.1) - OBJECTS MGMT\_BSLN\_INTERNAL and MGMT\_BSLN INVALID FOR DBSNMP

with cols as (

select owner, table\_name,

listagg('c.'||column\_name, ',') within group (order by column\_id) collist

from all\_tab\_columns

where (owner, table\_name) in (

select owner, table\_name

from all\_tables

where owner in ('PAULZIP', 'HR') and

tablespace\_name is not null -- avoid external tables

) and

data\_type in ('VARCHAR2', 'CHAR', 'CLOB') -- you may wish to add table filtering, as views and GTTs won't work

group by owner, table\_name

)

,data as (

select owner, table\_name,

dbms\_xmlgen.getxmltype('select ' || collist || ', c.ROWID as AROWID from dual left join '||owner||'.'||table\_name ||' c on 1 = 1') thexml -- Must not be empty, so left join with dual

from cols

where collist is not null

)

select /\*+ no\_merge \*/ d.owner, -- no merge hint prevents SOME 11g merging bugs, but not all, see next line

case when d.thexml is not null then table\_name end table\_name, -- work around merge bug in 11.2.x.x, you have to refer to all columns in driving table or null results occur

r.arowid, x.\*

from data d

join xmltable( -- For each row

'/ROWSET/ROW'

passing d.thexml

columns

arowid rowid path 'AROWID',

arow xmltype path '.'

) r on 1 = 1

join xmltable( -- For each column in a row

'/ROW/\*'

passing r.arow

columns

col\_name varchar2(30) path 'name(.)',

val varchar2(50) path '.'

) x on 1 = 1

where upper(x.val) like 'CO%'

/

OWNER TABLE\_NAME AROWID COL\_NAME VAL

-----------------------------------------------------------------------------------

'HR' 'EMPLOYEES' 'AAAJb1AAEAAAADPAAT' 'LAST\_NAME' 'Colmenares'

'HR' 'EMPLOYEES' 'AAAJb1AAEAAAADPAA1' 'EMAIL' 'COLSEN'

'HR' 'DEPARTMENTS' 'AAAJbwAAEAAAACvAAM' 'DEPARTMENT\_NAME' 'Corporate Tax'

'HR' 'DEPARTMENTS' 'AAAJbwAAEAAAACvAAN' 'DEPARTMENT\_NAME' 'Control And Credit'

'HR' 'DEPARTMENTS' 'AAAJbwAAEAAAACvAAR' 'DEPARTMENT\_NAME' 'Construction'

'HR' 'DEPARTMENTS' 'AAAJbwAAEAAAACvAAS' 'DEPARTMENT\_NAME' 'Contracting'