

10

Troubleshooting Oracle Database Advanced Queuing

These topics describe how to troubleshoot Oracle Database Advanced Queuing (AQ).

- [Debugging Oracle Database Advanced Queuing Propagation Problems](#)
- [Oracle Database Advanced Queuing Error Messages](#)

Debugging Oracle Database Advanced Queuing Propagation Problems

These tips should help with debugging propagation problems. This discussion assumes that you have created queue tables and queues in source and target databases and defined a database link for the destination database.

The notation assumes that you supply the actual name of the entity (without the brackets).



See Also:

["Optimizing Propagation"](#)

To begin debugging, do the following:

1. Check that the propagation schedule has been created and that a job queue process has been assigned.

Look for the entry in the `DBA_QUEUE_SCHEDULES` view and make sure that the status of the schedule is enabled. `SCHEDULE_DISABLED` must be set to 'N'. Check that it has a nonzero entry for `JOBNO` in table `AQ$_SCHEDULES`, and that there is an entry in table `JOB$` with that `JOBNO`.

To check if propagation is occurring, monitor the `DBA_QUEUE_SCHEDULES` view for the number of messages propagated (`TOTAL_NUMBER`).

If propagation is not occurring, check the view for any errors. Also check the `NEXT_RUN_DATE` and `NEXT_RUN_TIME` in `DBA_QUEUE_SCHEDULES` to see if propagation is scheduled for a later time, perhaps due to errors or the way it is set up.

2. Check if the database link to the destination database has been set up properly. Make sure that the queue owner can use the database link. You can do this with:

```
select count(*) from table_name@dblink_name;
```

3. Make sure that at least two job queue processes are running.
4. Check for messages in the source [queue](#) with:

```
select count (*) from AQ$<source_queue_table>
where q_name = 'source_queue_name';
```

5. Check for messages in the destination queue with:

```
select count (*) from AQ$<destination_queue_table>
where q_name = 'destination_queue_name';
```

6. Check to see who is using job queue processes.

Check which jobs are being run by querying `dba_jobs_running`. It is possible that other jobs are starving the propagation jobs.

7. Check to see that the queue table `sys.aq$_prop_table_instno` exists in `DBA_QUEUE_TABLES`. The queue `sys.aq$_prop_notify_queue_instno` must also exist in `DBA_QUEUES` and must be enabled for enqueue and dequeue.

In case of Oracle Real Application Clusters (Oracle RAC), this queue table and queue pair must exist for each Oracle RAC node in the system. They are used for communication between job queue processes and are automatically created.

8. Check that the [consumer](#) attempting to [dequeue](#) a [message](#) from the destination queue is a recipient of the propagated messages.

For 8.1-style queues, you can do the following:

```
select consumer_name, deq_txn_id, deq_time, deq_user_id,
       propagated_msgid from aq$<destination_queue_table>
where queue = 'queue_name';
```

For 8.0-style queues, you can obtain the same information from the history column of the [queue table](#):

```
select h.consumer, h.transaction_id, h.deq_time, h.deq_user,
       h.propagated_msgid from aq$<destination_queue_table> t, table(t.history) h
where t.q_name = 'queue_name';
```

Note:

Queues created in a queue table with `compatible` set to 8.0 (referred to in this guide as 8.0-style queues) are deprecated in Oracle Database Advanced Queuing 10g Release 2 (10.2). Oracle recommends that any new queues you create be 8.1-style or newer and that you migrate existing 8.0-style queues at your earliest convenience.

9. Turn on [propagation](#) tracing at the highest level using event 24040, level 10.

Debugging information is logged to job queue trace files as propagation takes place. You can check the trace file for errors and for statements indicating that messages have been sent.

Oracle Database Advanced Queuing Error Messages

The Oracle Database Advanced Queuing Error Messages are listed here.

ORA-1555

You might get this error when using the `NEXT_MESSAGE` navigation option for dequeue. `NEXT_MESSAGE` uses the snapshot created during the first dequeue call. After that, undo information may not be retained.

The workaround is to use the `FIRST_MESSAGE` option to dequeue the message. This reexecutes the cursor and gets a new snapshot. `FIRST_MESSAGE` does not perform as well as `NEXT_MESSAGE`, so Oracle recommends that you dequeue messages in batches: `FIRST_MESSAGE` for one, `NEXT_MESSAGE` for the next 1000 messages, then `FIRST_MESSAGE` again, and so on.

ORA-24033

This error is raised if a message is enqueued to a multiconsumer queue with no recipient and the queue has no subscribers (or rule-based subscribers that match this message). This is a warning that the message will be discarded because there are no recipients or subscribers to whom it can be delivered.

ORA-25237

When using the Oracle Database Advanced Queuing navigation option, you must reset the dequeue position by using the `FIRST_MESSAGE` option if you want to continue dequeuing between services (such as `xa_start` and `xa_end` boundaries). This is because XA cancels the cursor fetch state after an `xa_end`. If you do not reset, then you get an error message stating that the navigation is used out of sequence.

ORA-25307

Flow control has been enabled for the message sender. This means that the fastest subscriber of the sender's message is not able to keep pace with the rate at which messages are enqueued. The buffered messaging application must handle this error and attempt again to enqueue messages after waiting for some time.