

Using Oracle Streams Advanced Queueing: Best Practices

*An Oracle White Paper
June 2004*

Using Streams Advanced Queueing: Best Practices

Introduction	3
Storage.....	3
Tablespace allocation.....	3
Retention.....	3
Disk Striping.....	3
init.ora Parameters	4
Enqueue/Dequeue Tuning.....	4
Queue Monitor processes.....	4
Using REMOVE_NODATA.....	4
NEXT Navigation Mode.....	4
Correlation Identifiers for Dequeueing.....	4
Multiple Subscribers for Multiple Dequeue Processes.....	4
Propagation	5
Propagation Frequency	5
Disabling Propagation.....	5
Conclusion.....	5

Using Streams Advanced Queueing: Best Practices

INTRODUCTION

Oracle Streams Advanced Queueing (AQ) is the high-performance integrated message queueing feature of the Oracle Database. Negating the need to purchase, test, and integrate, AQ provides the best value and performance to customers for their queueing needs. However, as with all enterprise-level software, some amount of tuning is required to get the very best performance out of AQ. This document assists our customers in maximizing the AQ performance and efficiency in their hardware and software environments.

STORAGE

Information storage is by far the most important aspect of running your operations. How well the storage performs is directly related to how well the Oracle Database and Advanced Queueing will perform. The following tips on storage tuning will take you a long way toward yielding the best performance from your hardware.

Tablespace allocation

Advanced Queueing is more than just queue tables under the covers. There are many different data structures that facilitate the speedy operation of the queue tables. A responsible DBA should thus make sure there is enough tablespace allocated for all of the underlying structures.

Retention

Retention is an important feature of AQ, but not required by every type of application. When it is not necessary, make sure not to use it. That way, disk space can be conserved, and operation will be slightly faster.

Disk Striping

Advanced Queueing is a very IO intensive process. Almost anything that increases IO throughput will increase Advanced Queueing performance. One of the most important IO bottlenecks revolves around the disk access. Thus, a great way to improve AQ performance is to stripe the data across drives. The appropriate level of data striping will help AQ performance considerably.

init.ora Parameters

The 'initrans' parameter should be set to the average number of concurrent transactions

'pctfree' should be sufficient to allow the ITL table to expand in times of heavy load.

ENQUEUE/DEQUEUE TUNING

startup the queue monitor to ensure block cleanout of the dequeue IOT

Queue Monitor processes

Queue Monitor (QMN) processes are responsible for a variety of queue maintenance tasks, especially those tasks related to timing and deletion. It is thus important to make sure at least one QMN process is always running. Preferably, run a few QMN processes to manage multiple queues simultaneously, since each QMN can only access a single queue at a time.

Using REMOVE_NODATA

When a subscriber is interested in purging messages from a queue while in the BROWSE dequeue mode, using the REMOVE_NODATA instead of REMOVE will save a substantial amount of time, especially for messages with large payloads. REMOVE allows the application to read a message and delete it. However, by using the BROWSE dequeue mode, the application already has knowledge of the message and thus has no need to retrieve the message again during the REMOVE operation. The REMOVE_NODATA is thus the preferred dequeue mode during BROWSE.

NEXT Navigation Mode

When dequeuing messages, it is best to use the default navigation mode, known as NEXT_MESSAGE. This queue navigation method allows the consumer process to iterate through the list of messages in a queue, amortizing the cost of accessing the queue over many messages by caching the results. However, for priority ordered queues, FIRST_MESSAGE is the recommended navigation mode, since the queue is no longer a FIFO queue, making cached results potentially stale.

Correlation Identifiers for Dequeueing

Correlation identifiers are application specific identifiers that are uninterpreted by Advanced Queueing. Many applications use these identifiers to categorize messages while enqueueing for later dequeueing in batch. When used as a criterion for dequeueing, correlation identifiers should be indexed for fastest operation.

Multiple Subscribers for Multiple Dequeue Processes

When an application has multiple processes that access the same queue, it is better that each process be a separate subscriber. There will be less contention among

processes for data, since one subscriber will not have a lock over multiple messages. Concurrency can increase without much overhead.

PROPAGATION

Each propagation schedule is assigned a job queue process. We recommend that you have twice the number of job queue processes as propagation schedules, up to a maximum of 999 job queue processes.

$\text{job queue processes} = \min (2 * \text{propagation_schedules}, 999)$

Propagation Frequency

Tuning the propagation frequency should only be done using the latency and next_time parameters in the propagation schedule. The duration only controls how long a single propagation operation will last.

Disabling Propagation

Propagation is useful for distributing messages to different databases, but still consumes some computational resources. Thus, if it is not required, turn it off.

CONCLUSION

Following a few of the simple tuning tips in this document will help you on your way to having a high performance solution. For more tuning expertise, please contact your local Oracle Consultant, who will be more familiar with your particular hardware and software setup and requirements.



Using Oracle Streams Advanced Queuing: Best Practices

June 2004

Author: Toliver Jue

Contributing Authors:

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

www.oracle.com

Copyright © 2004, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.