

An Oracle White Paper
December 2013

Advanced Network Compression

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Executive Overview	2
Introduction	2
Parameters to enable Advanced Network Compression	3
In sqlnet.ora	3
In tnsnames.ora	4
Network Performance Analysis.....	5
Recommendations for Advanced Network Compression	7
Conclusion	7

Executive Overview

Oracle Database 12c introduces a new feature, Advanced Network Compression, as part of Advanced Compression Option. This paper provides an overview of Advanced Network Compression, its benefits, configuration details and performance analysis.

Introduction

Data compression techniques provide a way to convert data into alternate data formats which use less size. These techniques can be used to compress data to be transmitted over network at the sending side and then convert back to original data at the receiver to reduce the network traffic.

Advanced Network Compression reduces the size of Oracle Net Session Data Unit (SDU) transmitted over a data connection, thereby reducing the time required to transmit the SDU. This remains transparent to client applications and server processes.

Some of the benefits of using Advanced Network Compression are:

- **Increased Effective Network Throughput:** Compression allows transmission of large datasets in less time. SQL query response becomes faster due to the reduced transmission time, especially in constrained bandwidth environments.
- **Reduced bandwidth utilization:** Compression saves bandwidth by reducing the amount of data to be transmitted, allowing other applications to use it. This also helps in reducing the cost of providing network bandwidth.

Parameters to enable Advanced Network Compression

This section provides configuration details to enable compression for a client-server database connection. Compression options involve negotiation between client and server at connection establishment time before the option gets enabled on both sides.

In `sqlnet.ora`

Compression parameters can be configured in `sqlnet.ora` file, which resides both on the client and database server. Compression parameters include:

1) `SQLNET.COMPRESSION`:

Setting this parameter to ON or OFF enables or disables compression option for that environment. Default value for this parameter is OFF.

This option is negotiated between client and server at the time of connection establishment, and if both client and server have set it to ON, compression will be used that connection.

For example,

```
SQLNET.COMPRESSION = on
```

2) `SQLNET.COMPRESSION_LEVELS`:

Two levels of compression LOW and HIGH can be specified based on the requirement. LOW level uses less CPU but provides lower compression ratio, whereas HIGH level uses more CPU and provides higher compression ratio.

In general, we believe that LOW level provides a good trade-off between CPU usage and compression ratio. Hence, default setting for this parameter is LOW.

Compression levels are used at the time of connection establishment to verify which levels are used at both ends, and to select a common level.

For example,

```
SQLNET.COMPRESSION_LEVELS = (low, high)
```

3) SQLNET.COMPRESSION_THRESHOLD

This parameter can be used to specify the minimum data size, in bytes, for which compression should be performed. Compression is not performed on an SDU if the size of that SDU is less than this value. Default value for this parameter is 1024 bytes.

For example,

```
SQLNET.COMPRESSION_THRESHOLD = 2048
```

In tnsnames.ora

Clients can also enable compression and specify compression levels through connect descriptor for an individual connection. The following parameters can be set at the DESCRIPTION level of a connect-string in tnsnames.ora.

- 1) COMPRESSION
- 2) COMPRESSION_LEVELS

COMPRESSION_LEVELS is an optional parameter, and if not specified, LOW level is used. Default value for the COMPRESSION parameter is OFF.

For example,

```
net_service_name=
  (DESCRIPTION=
    (COMPRESSION=on)
    (COMPRESSION_LEVELS=(LEVEL=low) (LEVEL=high))
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-server) (PORT=1521))
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales2-server) (PORT=1521)))
    (CONNECT_DATA=
      (SERVER_NAME=sales.us.example.com)))
```

If parameters are set at connect descriptor level, they take precedence for that client connection over the values specified in sqlnet.ora.

Network Performance Analysis

This section evaluates the impact of compression on network performance for a typical SQL query under different bandwidth cases. In order to compare network performance, total time to transfer the same dataset (SQL query response time at client) was used as a performance metric.

Experiment using Text data

A table with four columns and 26,516 rows with each row containing around 10KB text data was stored in the database. SQL query “select * from table” was used by the client to fetch full table data. The array size was set to 5000. Network round trip time (RTT) between client and server in this setup was 37ms. The total time to receive the data from server was measured at client and used as performance metric.

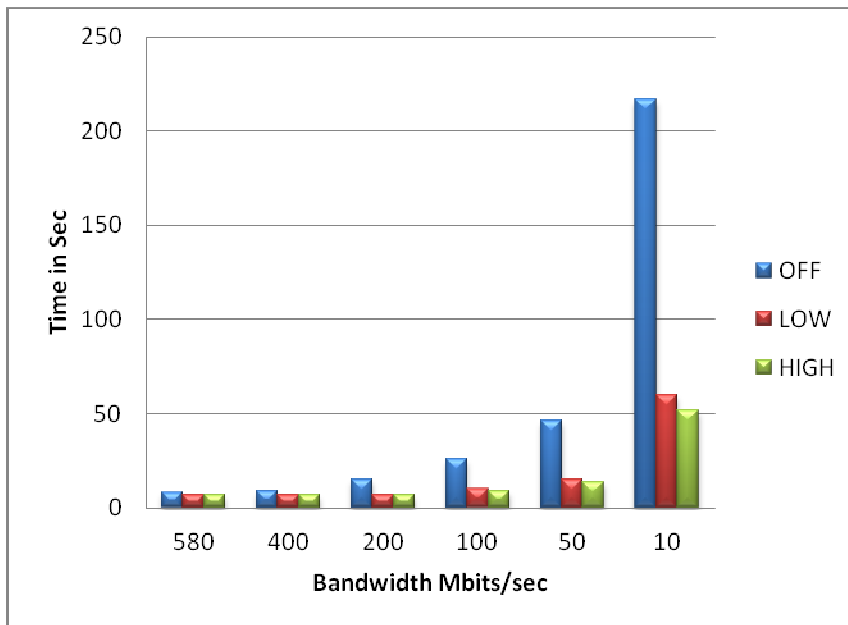


Figure 1: Query response time vs. Network bandwidth (text data)

Experiment using Mixed data

A table with five columns and 1 million rows with each row containing 50bytes of mixed character and number data was stored in database server. SQL query “select * from table” was used by client to fetch full table data. The array size was set to 5000. Network round trip time (RTT) between client and server in this setup was 37ms. The total time to receive the data from server was measured at client and was used as performance metric.

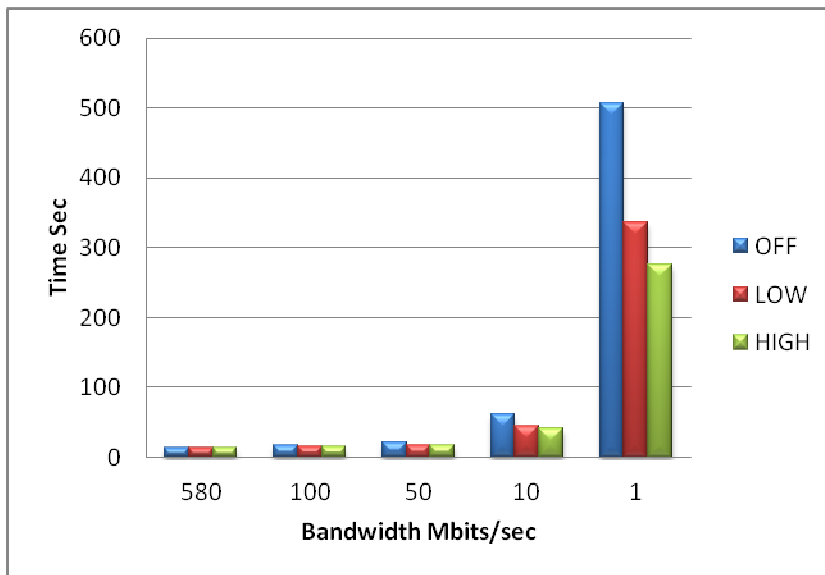


Figure 2: Query response time vs. Network bandwidth (mixed data)

These figures indicate the performance difference in SQL query response time with compression ON (both low and high) and OFF. SQL query response with compression is faster as compared to without compression. As the bandwidth becomes lower, the effect of compression becomes more visible.

Recommendations for Advanced Network Compression

Enabling compression may not always increase performance, and can be affected by several factors. This section provides guidelines on when Advanced Network Compression can increase performance.

Effective network throughput can be increased using network compression only when rate of generation of data to be transmitted is higher than the rate of sending data on the network. This is owing to the fact that, when bandwidth is higher, network does not block any data and the throughput achieved is equivalent to the rate of generation of data.

Advanced Network Compression is more performant when used on large result sets. It should be avoided when data to be sent is in small as it just adds extra CPU overhead and the effect on network throughput would be negligible.

CPU utilization increases with compression, so a faster CPU is recommended whenever compression is used. The faster the CPU, the faster the compression and higher the performance gain.

Performance gain also depends on the compression ratio of data being sent. Compression ratio depends on the nature of data. Text data is generally more compressible than binary or already-compressed data like image or movie files.

Compression should be configured with caution considering all above of the factors.

Conclusion

Advanced Network compression not only makes SQL query response faster but also saves bandwidth. This feature is transparent to client applications and server processes. When used on low-bandwidth connections between hosts with fast CPUs, it can significantly improve performance.



Advanced Network Compression
December 2013
Authors: Abhishek Dadhich, Kant Patel,
Feroz Khan, Srinivas Pamu

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
oracle.com



| Oracle is committed to developing practices and products that help protect the environment

Copyright © 2012, Oracle and/or its affiliates. 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.