How-To Library

Informatica

Performance Tuning Guidelines for PowerExchange for Amazon Redshift for PowerCenter 10.1

Abstract

When you use PowerExchange for Amazon Redshift to read data from or write data to Amazon Redshift, multiple factors such as hardware parameters, Amazon Redshift ODBC Connection parameters, and mapping parameters impact the adapter performance. You can optimize the performance by tuning these parameters appropriately. This article describes general reference guidelines to help you tune the performance of PowerExchange for Amazon Redshift and to improve performance of a reader mapping that uses an Amazon Redshift ODBC Connection.

Supported Versions

- PowerExchange for Amazon Redshift 10.1

Table of Contents

Overview

This document describes the key hardware, Cloud Applications tuning parameters, and PowerCenter mapping parameters that you can tune to optimize the performance of PowerExchange for Cloud Applications.

Performance tuning is an iterative process in which you analyze the performance, use guidelines to estimate and define parameters that impact the performance, and monitor and adjust the results as required.

Note: The performance testing results listed in this article are based on observations in an internal Informatica environment using data from real-world scenarios. The performance of PowerExchange for Cloud Applications might vary based on individual environments and other parameters even when you use the same data.

You can optimize the performance of Cloud Applications mappings by tuning the following areas:

- Hardware
- Heap size
- PowerCenter mapping

Performance Tuning Areas

You can optimize the performance of Cloud Applications mappings by tuning the following areas:

- Hardware
- Heap size

- PowerCenter mapping

Tune the Hardware

You can tune the following hardware parameters to optimize the performance of the machine where the PowerCenter Integration Service runs:

- CPU frequency
- NIC card ring buffer size

CPU Frequency

Dynamic frequency scaling adjusts the frequency of the processor either for power savings or to reduce heat. Ensure that the CPU operates at least at the base frequency.

When CPUs are underclocked, where they run below the base frequency, the performance degrades by 30% to 40%. Informatica recommends that you work with your IT system administrator to ensure that all the nodes on the cluster are configured to run at their supported base frequency.

To tune the CPU frequency for Intel multicore processors, perform the following steps:

1. Run the lscpu command to determine the current CPU frequency, base CPU frequency, and the maximum CPU frequency that the processor supports.
2. Request your system administrator to perform the following tasks:
   a. Increase the CPU frequency to the supported base frequency.
   b. Change the power management setting to **OS Control** at the BIOS level.
3. Run CPU-intensive tests to monitor the CPU frequency in real time and adjust the frequency for improved performance. On Red Hat operating systems, you can install a monitoring tool such as cpupower.
4. Work with your IT department to ensure that the CPU frequency and power management settings are persisted even for future system reboots.

NIC Card Ring Buffer Size

NIC configuration is a key factor in network performance tuning. When you deal with large volumes of data, it is crucial that you tune the Receive (RX) and Transmit (TX) ring buffer size. The ring buffers contain descriptors or pointers to the socket kernel buffers that hold the packet data.

You can run the ethtool command to determine the current configuration. For example, run the following command:

```
# ethtool -g eth0
```

The following sections show a sample output:

```
Ring parameters for eth0:
Pre-set maximums:
RX: 2040
RX Mini: 0
RX Jumbo: 8160
TX: 255

Current hardware settings:
RX: 255
RX Mini: 0
RX Jumbo: 0
TX: 255
```

The **Pre-set maximums** section shows the maximum values that you can set for each parameter. The **Current hardware settings** section shows the current configuration details.

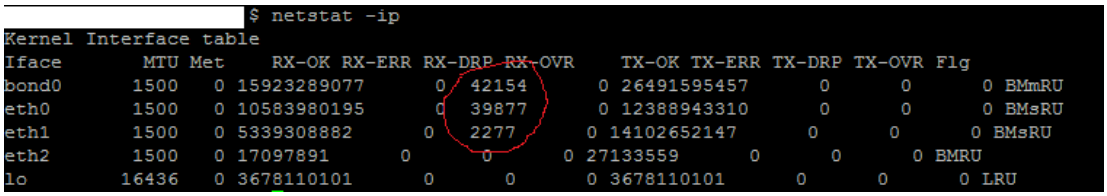
A low buffer size leads to low latency. However, low latency comes at the cost of throughput. For greater throughputs, you must configure large buffer ring sizes for RX and TX.

Informatica recommends that you use the ethtool command to determine the current hardware settings and the maximum supported values. Then, set the values based on the maximum values that are supported for each operating system. For example, if the maximum supported value for RX is 2040, you can use the ethtool command as follows to set the RX value to 2040:

```
# ethtool -G eth0 RX 2040
```

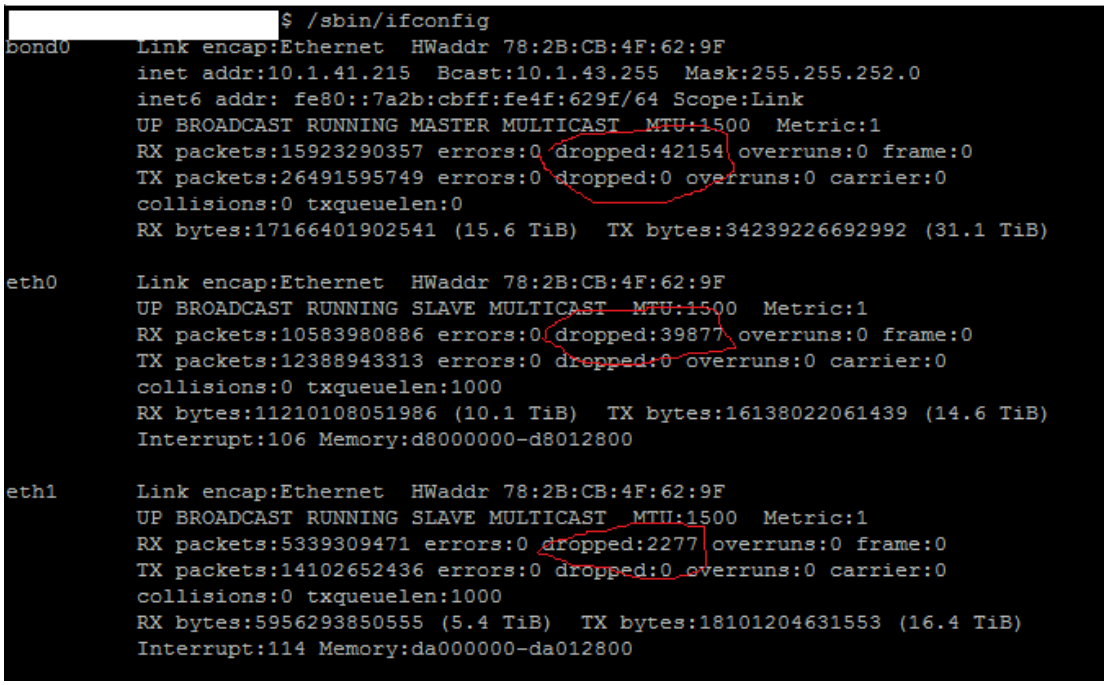
If you set a low ring buffer size for data transfer, packets might get dropped. To find out if packets were dropped, you can use the netstat and ifconfig commands.

The following image shows a sample output of the netstat command:



The RX-DRP column indicates the number of packets that were dropped. Set the RX value such that no packets get dropped and the RX-DRP column shows the values as 0. You might need to test several values to optimize the performance.

The following image shows a sample output of the ifconfig command:



The status messages indicate the number of packets that were dropped.

Tune the Buffer Size and the Block Size

You can add performance tuning parameters to the `caas.config` file to enhance the session performance.

The caas.config file is available on the PowerCenter Integration Service host machine, in the following directory:

`<Informatica installation directory>\server\bin\javalib\449304`

Add the following parameters to the `instanceName` option to tune the session performance:

`bufferSize`

`blockSize`

The following snippet shows the parameters added to the `instanceName` option:

```
"instanceName":"Informatica Connectivity as a Service","bufferSize":1024,"blockSize":102
```

Save the file after adding the parameters. The value for bufferSize and blockSize is in MB. The recommended bufferSize is 1024MB.

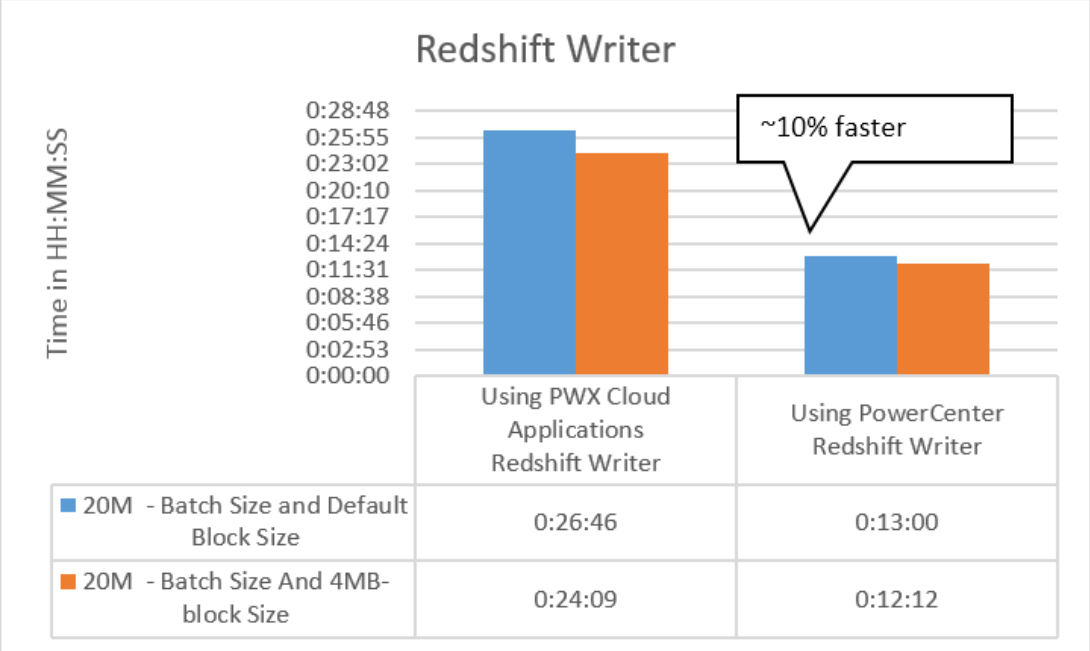
By default, the blockSize is equal to 1/10th of the bufferSize. These settings are applicable to all the sessions configured for PowerExchange for Cloud Applications.

Informatica Cloud Secure Agent uses the bufferSize and blockSize parameters defined in the `caas.config` file to improve the performance of a mapping running on Informatica Cloud. The bufferSize and blockSize parameters do not impact the PowerCenter session level configuration or performance.

You can also set the buffer memory in the PowerCenter session. For more information, see the *PowerCenter Advanced Workflow Guide* and *PowerCenter Performance Tuning Guide*.

Case study

The following image show the Writer performance comparison for Amazon Redshift when you select the buffer size as **45** and block size as **4**:



	Using PWX Cloud Applications Redshift Writer	Using PowerCenter Redshift Writer
20M - Batch Size and Default Block Size	0:26:46	0:13:00
20M - Batch Size And 4MB-block Size	0:24:09	0:12:12

Tune the Mapping

You can tune the following mapping parameters in the PowerCenter Client to optimize the performance:

- Batch size
- Buffer size and block size
- Ports precision

- Data movement mode
- Data type mapping

For more information, see the *PowerCenter Performance Tuning Guide*.

Ports Precision

Precision is the maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale.

You can tune the precision in PowerCenter repository mappings.

When mappings contain ports with a larger precision than required, the mapping performance degrades. Informatica recommends that you set the precision judiciously for all source ports, transformation ports, and target ports. For instance, if a string port can handle data of a maximum of 200 characters, set the precision to 200. Do not set the precision to a high value such as 1000.

Data Movement Mode

When you run mappings on the PowerCenter Integration Service, the logical Data Transformation Manager (LDTM) component of the PowerCenter Integration Service determines whether to use the ASCII or Unicode data movement mode for mappings that read from a flat file or relational source. The LDTM determines the data movement mode based on the character set that the mapping processes. When a mapping processes all ASCII data, the LDTM selects the ASCII mode. In ASCII mode, the Integration Service uses one byte to store each character, which can optimize mapping performance. In Unicode mode, the PowerCenter Integration Service uses two bytes for each character.

For more information about the data movement mode, see the "PowerCenter Integration Service Architecture" chapter in the *Informatica Application Service Guide*.

Data Type Mapping

When the PowerCenter Integration Service reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When the PowerCenter Integration Service writes data to a target, it converts the transformation data types to the comparable native data types.

When you map source ports to transformation ports and then to target ports, avoid unnecessary data type conversions. For instance, do not map a port of the string data type to a port of the date data type. Ensure that you map ports to the same data type in all components of the mapping. Also, remove all unconnected ports from the mapping.

Authors

Krishna Prabhakar Devarakonda

Chanchal Das