IBM Power10 I/O performance optimization

Best practices guide to maximize speed of I/O on IBM Power servers

IBM

Table of contents

Overview

This paper aims to provide guidance, best practice recommendations, and references to other data sources for optimizing the network and storage input output (I/O) performance of IBM Power® server applications. The primary goal is to assist users in achieving maximum throughput and minimum response time.

It is important to note that some of the critical activities that impact I/O performance are static and performed outside of programming, such as configuring hardware and tuning static parameters. Nonetheless, users can directly control certain dynamic factors that also influence I/O performance.

This paper does not cover all best practices and should be used in conjunction with other relevant documents, including PowerVM, AIX, Linux, and IBM i resources. These collective resources can aid in defining and building systems that can deliver the necessary I/O speed, configuring those systems for optimal hardware utilization, and providing recommendations for application software that effectively leverages I/O performance capabilities.

Hardware configuration

The first step towards meeting application throughput and response time requirements for I/O workloads is to ensure that the I/O subsystems are correctly sized and configured. In most cases, running these subsystems without making any adjustment to static tuning parameters that are set and rarely altered is sufficient.

When selecting I/O subsystem hardware and configuring it with a compute server, the factors such as performance, reliability, availability, serviceability, capacity, and cost must be considered. Configuration is highlighted here to make users aware of essential aspects that must be considered. For instance, hardware parallelism can be utilized to enhance throughput. An example of hardware parallelism is using features like Linux bonding or AIX Etherchannel for increasing Ethernet network throughput, which combines multiple physical paths into a single logical path. Similarly, storage interfaces like Fibre Channel can provide comparable increases in throughput with multi-path I/O (MPIO) features.

Static parameter tuning

To achieve more efficient use of the I/O subsystems, it may be necessary to carefully tune the parameters. This document does not provide details on tuning; however, the following list of documents can serve as good starting points for this essential task:

- Refer to Hints and Tips for Migrating Workload to IBM Power9 and Power10 Processor-Based Systems chapter 7 for all the basics that should be considered.
- Refer to Power10_Performance_Quick_Start_Guides.pdf on ibm.com for more detailed information on tuning options to consider. It does not specify exact parameter values. This is mainly because the performance gains or losses resulting from adjusting various parameters are highly dependent on the specific workload and configuration, and therefore there is no single solution that applies to all customers and workloads.
- Refer to Power10 Performance Best Practices on ibm.com for checklist that provides specific values for certain adapter tunable parameters that are likely to improve the performance of most workloads.
- Refer to Linux on Power I/O Performance Tuning Guide - IBM Power Community on ibm.com for performance optimization in Linux on Power systems.

Dynamic usage

After the hardware has been designed to address performance, capacity, reliability, availability, serviceability, and cost considerations, and static parameter tuning has been deemed required and implemented, it is recommended to consider more flexible and software-managed approaches to optimizing the I/O subsystems. To reduce the overhead associated with input output operations, the following methods can be utilized:

- **Caching**: Utilize caches to eliminate some I/O operations. Cache hits in an application/middleware, OS kernel, storage I/O adapter, or end device caches can reduce the response time of I/O requests by eliminating back-end I/O operations.
- **Coalescing**: Utilize methods to coalesce multiple smaller I/O operations into a single larger I/O operation to reduce command rate while maintaining the same data rate. This can be achieved using caches, but there is more benefit to be gained the higher up the I/O stack this is implemented. However, it is important to watch out for potential side effects such as increased response times for latency-sensitive applications.
- **Asynchronous I/O**: Reduce application dependence on individual I/O completion response time by utilizing asynchronous I/O Application Program Interfaces (APIs).

- **Minimizing Device/File/Database Opens and Closes**: Especially with IBM i, but applicable with any Operating System, minimizing the number of device, file, or database opens and closes can minimize I/O overhead.

To increase the throughput capability of the I/O stack with parallelism, taking advantage of queues in the stack can be helpful. This is both a software issue and a system admin/static parameter tuning issue, and synergy is needed between software developers and system administrators in this area. To take advantage of this, both the software and the configurations need to account for it.

Some of the possible ways to increase the throughput with parallelism and queues include:

- The I/O adapter (IOA) device drivers often have an adjustable number of queues, and those individual queues can be lengthened or shortened. Adjusting the number and length of queues can help to optimize the balance between parallelism and overhead.
- Some protocol device drivers, such as Small Computer System Interface (SCSI) or Transmission Control Protocol (TCP) or Non-volatile Memory Express (NVMe), also have similar queue number and queue length controls. Adjusting these parameters can help to optimize the performance of the system for specific workloads and use cases.

Networking specific tuning

The following documents provide recommendations that are specific to the AIX and Linux operating systems and focus on network performance tuning:

- Refer to AIX: Network performance on ibm.com for static tuning as well as software changes that can be made to improve network performance for AIX.
- Refer to TCP request and response workload tuning on ibm.com for examples of workload tuning that is highly dependent on the application and can be used to improve TCP request and response workloads.
- Refer to Linux on Power I/O Performance Tuning Guide on ibm.com for improving I/O performance on Linux for Power systems, including static tuning suggestions as well as workload specific tuning options.

A feature to reduce lock overhead is available in AIX 7.3 TL1 and later versions. This feature can help to improve network performance and reduce overhead associated with locking.

Storage specific tuning

The following documents provide information on how to reduce the application dependance upon individual I/O completion response time via asynchronous I/O APIs:

- For IBM i refer to Asynchronous I/O on ibm.com.
- For AIX refer to Asynchronous I/O Subsystem on ibm.com.

- For Linux refer to [Boost application performance using asynchronous I/O](#) on ibm.com.

Choosing the logical device types or APIs that determine the interaction between the OS, middleware, and applications with the backend devices is crucial for determining the perceived speed of the backend devices, apart from the type of backend storage physical device used. The logical device types or APIs are influenced by applications, ISVs, and other factors.

The following documents provide useful information on this topic:

- For AIX filesystems refer to [File system performance](#) on ibm.com.
- For AIX logical volume manager (LVM) refer to [Logical volume and disk I/O performance](#) on ibm.com.
- For Linux filesystems refer to [Anatomy of the Linux file system](#) on ibm.com.
- For Linux LVM refer to [Introduction to Logical Volume Manager | Enable Sysadmin](#) on redhat.com.
- For IBM i ASP setup refer to [Auxiliary storage pools](#) on ibm.com.
- For IBM i NPIV usage refer to [IBM i Virtual Fibre Channel Performance Best Practices](#) on ibm.com.

Virtual I/O server

IBM's Virtual I/O Server (VIOS) is a tool that enables a system to fully utilize and share the substantial capabilities of Power systems I/O subsystems across multiple logical partitions (LPARs), providing cost and functional advantages over dedicating I/O adapters to a single LPAR.

Properly sized VIOS CPUs, DRAM and I/O hardware is crucial to ensure that a VIOS can successfully serve I/O resources to the client LPARs, meeting all running application I/O requirements on all LPARs.

Following documents provide useful information on this topic:

- Refer to "Chapter 4.0 VIRTUAL I/O" of [Linux on Power I/O Performance Tuning Guide - IBM Power Community](#) on ibm.com for sizing and configuration recommendations when using Linux as a client.
- Refer to Nigel Griffith's [Sizing the Virtual I/O Server (VIOS) - My Rules of Thumb](#) on ibm.com for practical advice on sizing and configuration suggestions.
- Refer to [IBM PowerVM Best Practices](#) on redbooks.ibm.com for VIOS best practices.
- Refer to [NPIV Multiple-Queue support](#) and [Modernization of PowerVM NPIV stack: from Single to Multiple Queue Support](#) for more information on Multiple-Queue feature of NPIV.

Summary

Out-of-the-box performance is typically sufficient for native I/O configurations when the backend network and storage hardware is properly sized, configured, and tuned. For virtualized I/O configurations the sizing of CPUs and DRAM and which type of virtualization is used are especially important.

Some one-time, or static, tuning usually performed by system administrators can be done to enable more I/O subsystem performance if needed.

To unleash the full capability of the I/O subsystems, the software that initiates the I/O workloads needs to be written so that those features and capabilities can be utilized.

Get more information

To learn more about Power I/O Performance optimization refer to IBM Advanced Technical Services and/or refer the following from ibm.com:

- Modernize applications and infrastructure with a frictionless hybrid cloud experience
- IBM i on Power -Performance FAQ
- Systems management
- Advance Toolchain for Linux on IBM Power Systems

The IBM Developer website offers a three-part series called "Optimizing AIX 7 network performance" which provides a good reference for network performance optimization. The three parts include:

- Network overview – Monitoring the hardware
- NFS monitoring and tuning
- Monitoring your network packets and tuning the network

About the authors

The IBM Power Systems Performance I/O Squad would like to thank the many people who made invaluable contributions to this document. Contributions included authoring, insights, ideas, reviews, critiques, and reference documents.

This document was created by the AIX, IBM i, and Linux development teams, IBM Performance teams, and IBM Infrastructure Expert Labs. If you'd like to explore consulting services related to the concepts included in this document, contact Expert Labs at technologyservices@ibm.com.