Increasing Performance and Availability in Microsoft SQL Server
Using Micron SSDs + SQL Server Availability Groups

Randy Glissmann, Enterprise SSD Marketing Manager

Overview
Microsoft’s SQL Server is widely used by enterprises to ensure that mission-critical production applications do not fail. Because high performance is essential for these applications, many IT professionals prefer to use solid state drives (SSDs), which can provide much higher transaction rates than hard disk drive (HDD)-based storage. This document explains how using Micron’s PCIe or SATA SSDs in conjunction with SQL Server’s AlwaysOn Availability Groups (AAG) enables the high performance and availability required by mission-critical applications while decreasing total cost of ownership (TCO).

Using Server Clusters for High Availability
When one server fails within a cluster of servers, a secondary server supports a failover to ensure minimal downtime. To enable access to storage in the event of a failover, NAS- or SAN-connected storage has traditionally been used. Because this storage is shared between the clustered servers, a secondary server has immediate access to the database after a failover and can continue to process SQL jobs.

To address the movement toward using software-defined storage, Microsoft introduced AAG in their SQL Server 2008, which enables the use of direct-attached storage. This dramatically lowers the cost of storage, even as performance is significantly boosted through the use of PCIe- and SATA-connected flash storage. Micron’s P420m PCIe and M500DC SATA SSDs are flash solutions that can help significantly reduce transaction latencies compared to using HDDs while lowering TCO compared to using network-connected storage.

Increasing Availability With AAG Mirroring
AAG enables every transaction to a primary database to be mirrored to a secondary database on a separate physical server. This mirroring is performed using either asynchronous-commit or synchronous-commit implementations; however, when physical servers...
are separated by significant distance, communication delays occur. If an application using SQL Server waits for the secondary database to be updated, these delays increase latency. The asynchronous-commit mode enables secondary databases to lag behind the primary database, which reduces the impact to latency, but this risks data loss from a server failure.

On the other hand, when using the synchronous-commit mode, the primary database must wait for the secondary database to acknowledge when it is finished saving the log to storage. Synchronous-commit mode ensures that no data loss occurs because the secondary database is always synchronized with the primary database. This is a good solution when the primary and secondary servers are collocated because communication delays are minimal. Any potential increase to transaction latency is caused by delays in writing to storage, but because data can be quickly written onto flash storage, an increase in latency is minimal.

Conclusion

Solutions for providing high performance and continuous availability are critical for Microsoft’s SQL Server users. Flash storage dramatically reduces high-latency transactions compared to using HDDs. Direct-attached flash storage, like Micron’s PCIe and SATA SSDs, can also lower TCO compared to network-connected flash storage; however, a server or storage failure can still affect application availability. By using SQL Server’s AAG to mirror the database along with SSDs, applications using Microsoft’s SQL Server can support the high availability that enterprises need while lowering TCO.