How to boost application performance using Solid State Disk devices

A practical guide to implement SSDs quickly and easily with IBM i

Gottfried Schimunek April 12, 2011

This article provides an overview of Solid State Disks (SSDs), and how they can be implemented and used effectively on IBM i. It also discusses a simple method to determine and move potential frequently referenced objects to SSDs to take full advantage at the lowest cost of implementation.

Overview of the Solid State Disk drive technology

The SSD technology has been around for a couple of decades already in the form of memory sticks, thumb drives, and so on. Only recently have they developed up to an enterprise level with characteristics required by modern large scale commercial applications on IBM Power Systems with IBM i.

Performance characteristics of SSDs versus regular disk drives

Due to the memory storage technology used in SSDs, the performance is an order of magnitude higher than regular hard disk drives. A single random read operation is about a hundred times faster than the same operation using the latest available disk drives on the market today. Since there are no moving parts in SSDs, and any part of the memory can be read instantaneously, the number of I/Os which can be performed is also extremely higher on SSDs. Regular disk drives can perform about 120 – 150 I/O operations per second, whereas SSDs can easily go up to 30000 I/O ops/sec.

In addition, regular disk drives slow down when a larger area on disk is being traversed (for example, disks with less data stored have good response and seek times) the more data is stored and the disk becomes more full, the slower the average service/response time. SSDs on the other hand can be used all the way up to a hundred percent capacity, and the performance is exactly the same as if it were used only for a fraction. Figure 1 below shows this differences between regular disk drives and SSDs in terms of performance.

Figure 1. SSD performance versus HDD

What I/O operations benefit the most with SSDs

Not all I/O operations universally benefit from SSDs. Sequential reads or writes from and to database files, for example, are highly optimized already today due to big IOA caches and a sophisticated storage management function (double buffering) in IBM i. Write operations being done on SSDs in masses, like journal deposits, are also not able to perform in the same way as on regular disk drives due to the larger cache available, and the fact that write operations on SSDs are taking about twice as long as read operations. SSDs shine really bright when random I/Os are being executed. Fortunately, most of all I/O operations are random read/update operations. On average about 70 percent of all I/O ops are random reads/updates and 30 percent sequential ops.

However, certain applications tend to use more sequential than random operations and therefore are not able to benefit from SSDs as much as others. For example, a batch application producing a report of all processed data will likely be using sequential operations since all the data in the database needs to be processed anyway. Another application which processes only a small subset of the database likely uses random reads/updates via a key and benefits from the use of SSDs.

How to determine what objects to move to SSDs

The easiest approach would be simply to replace all regular disk drives with SSDs and move over all objects and data. Although, this approach would work and application performance would benefit, the cost of replacing regular disk drives with SSDs would be prohibitive for many installations, at least those with more then a few hundred GB of disk storage.

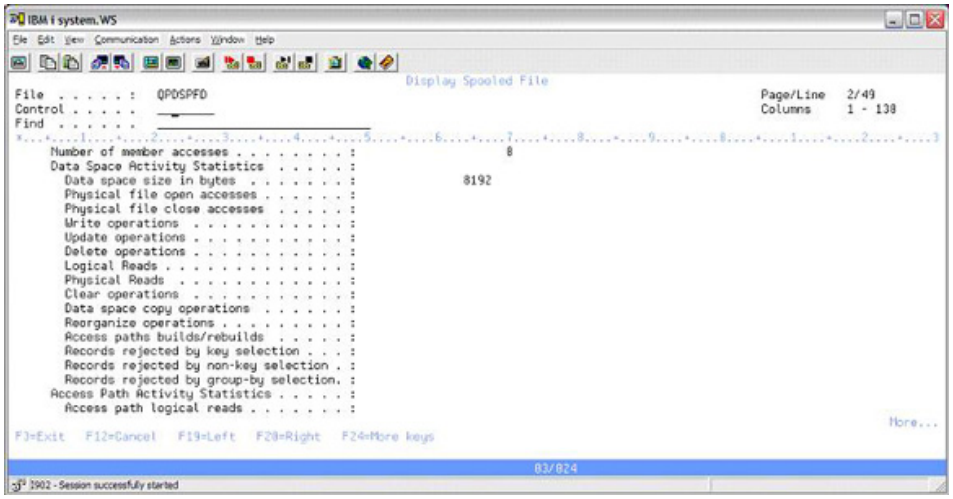
Since the cost per GB of storage on SSDs is about three to four times higher than regular disk drives, someone would have to compare the benefit of SSDs with the cost incurred. Many times though, SSDs outperform regular disk drives and have a better price/performance ratio per GB under the assumption that not all the data has to be moved over to SSDs.

Since certain objects are being accessed via certain access methods and I/O operations, at least for the majority of all operations—the old rule is an 80-20 split. If 80 percent of all operations are random read ops, this object would be a good candidate to be placed on SSDs. Fortunately, IBM i keeps statistics on all I/O operations performed for each individual database object in the system. All you have to do is to use the IBM i Display File Description (DSPFD) command to determine which objects have the majority of random read ops:

```
DSPFD QGPL/QORDDTL
```

The output might look somewhat like this:

Figure 2. DSPFD output



There are other more sophisticated methods to determine those I/O operations by using a performance trace and using performance analysis tools, but basically the DSPFD command provides a very good basis for the determination of SSD candidates.

How to move data to SSDs

When SSDs are added to the system, they are not immediately used by the system by default; they stay empty. To get data on SSDs, you have to specify the UNIT(*SSD) parameter in the CRTPF and CRTLF commands. To move existing physical or logical to SSDs, the UNIT(*SSD) parameter can be used on the CHGPF and CHGLF command, as well. All existing data is moved underneath the cover from regular disk drives to SSDs automatically.

SQL described tables and indexes can also be placed and moved by the corresponding ALTER TABLE or ALTER INDEX commands.

To move the data back to regular disk drives, the UNIT(*ANY) parameter can be used again.

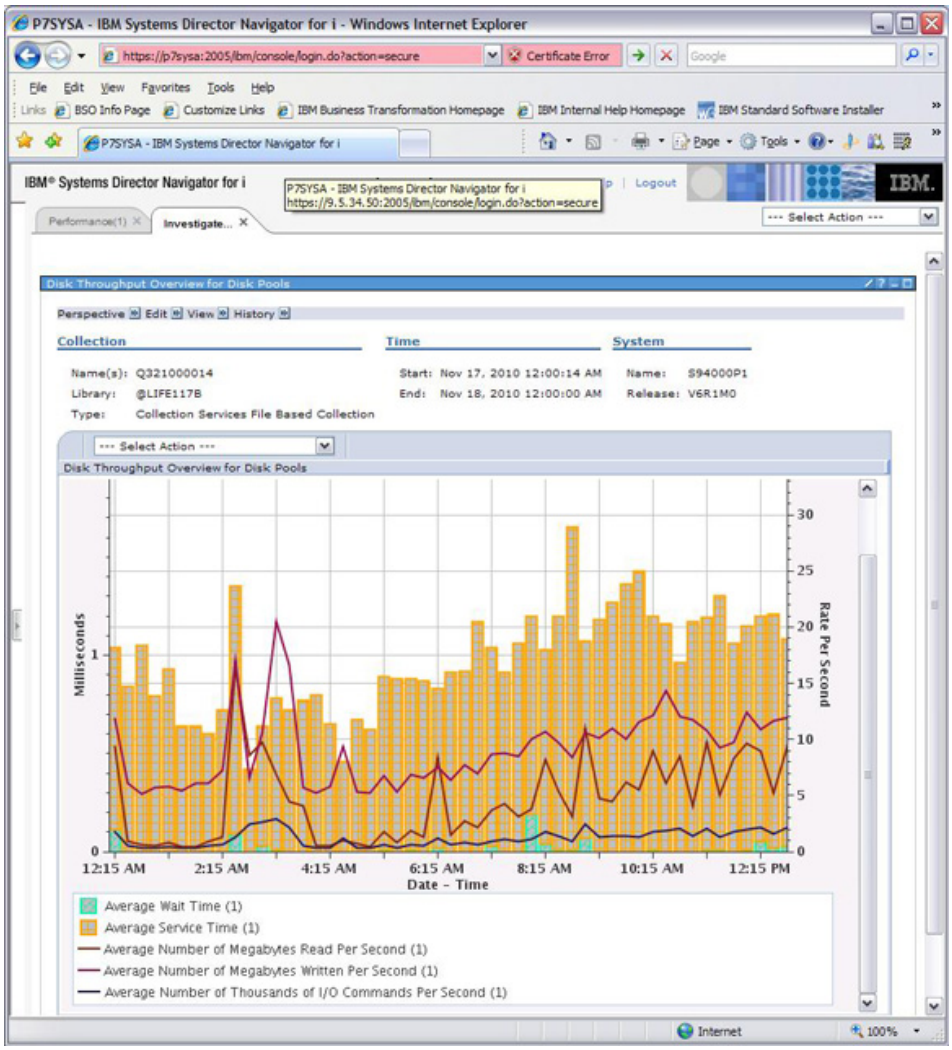
How to measure the performance benefits of SSDs

The benefits of SSDs are measured in many ways, such as the time it takes to run a certain batch job, the response time for interactive applications, or the number of transactions processed in a certain period of time. However, it all boils down to how fast an I/O is performed (eliminating the wait for I/O operations) and how many I/Os we can do in a certain period of time.

Regular performance data collection and the use of the performance data investigator function in IBM i can be used to determine those characteristics before and after the implementation of SSDs. Figure 3 below, the Performance Investigator example, shows the average disk service time and the average disk wait time for a given period of time. When SSDs are implemented, those average

values will be lower and show the performance benefits. Other graphs can be selected to show the total number of I/Os performed per second which will also indicate the improvement from SSDs.

Figure 3. Performance Investigator screen shot example



Summary

Solid State Drives (SSDs) can significantly improve performance of disk I/O operations for many types of applications. Although SSDs are typically more expensive then regular disk drives, moving only a subset of objects to SSDs make them a good viable and price/performance option to improve disk I/O performance at a competitive price.

IBM i has all the characteristics and information to be able to identify and move certain most frequently used objects to SSDs.