

## **Design Tip #76 Creating the Advantages of a 64-bit Server**

By Joy Mundy

Data Warehouse / Business Intelligence systems love memory. This has been true for decades, since 64 MB was a lot of system memory. It remains true today, when 64 GB is a lot of system memory.

Memory is the most common bottleneck affecting the performance of a DW/BI system. Adding more memory is often the easiest way to improve system performance. DW/BI systems have a strong affinity for 64-bit hardware. The improved processing performance is nice, but 64-bit is particularly important to us because of the vastly larger addressable memory space. In-memory operations are orders of magnitude faster than operations that need to access a lot of disk.

All components of the DW/BI system benefit from additional memory. Memory helps the relational engine answer queries and build indexes much faster. The ETL system can be a significant memory hog. Good design for an ETL system performs transformations in a pipeline and writes data to disk as seldom as possible. OLAP technology uses memory during processing when the cube is computing aggregations and during query time. Even the reporting application can use significant memory. The query that underlies a report uses the memory of the underlying relational or OLAP engine. And if you're using a reporting server to manage and render reports, you may be surprised at how much memory it can consume.

High end hardware for both Windows and Unix systems are dominated by 64-bit, as you can see by reviewing the TPC-H benchmark results. But 64-bit is a no-brainer for DW/BI systems of any size. You can buy a commodity server with four dual core 64-bit processors, 16 GB of memory, and 700 GB of storage for about \$25,000. That kind of hardware should easily support a smallish DW/BI system of several billion fact rows and dozens of users.

A server like we've described is particularly compelling for the single vendor "stack" such as Microsoft SQL Server 2005. Microsoft charges a flat rate per processor, no matter how many components of the product you install, or even whether your system is 64-bit or has dual core processors. Many smaller organizations are deploying their DW/BI system on a single server that has the relational database engine, Integration Services for ETL, Analysis Services for OLAP and data mining, and Reporting Services for report management and rendering.

A single-server solution is particularly compelling if you can sequence processing to use system resources efficiently. For example, use Integration Services and the relational engine to load the relational data warehouse. When the heavy ETL work is done, use Analysis Services and the relational engine to process incremental cube updates. Next, pre-execute and cache the big, popular, and demanding reports. This step will use Reporting Services and Analysis Services or the relational engine, depending on where the reports are sourced from. During the workday, rely primarily on Analysis Services to support ad hoc analysis and on-demand reporting. Integration Services is not busy during the day at all, and by pre-caching reports you're reducing the daytime load on Reporting Services and the relational engine.

Although we've characterized the 64-bit hardware as a no-brainer even for small to medium systems, there are some things you need to worry about. The first question is the chip architecture. Those of us who've grown comfortable with the simplicity of the "Wintel" world are faced with new decisions: Itanium, x64, or AMD64? These are fundamentally different chips, and the operating system and

application code (such as SQL Server) must be separately compiled, tested, and supported. You're betting that the operating system and database software will continue to support your chosen chipset for the expected life of your hardware. Consider software requirements beyond the core server software, such as specialized ETL functionality or database management tools.

A secondary issue is the architecture of your development and test environments. Your DW/BI test system must use the same architecture as your production system. In an ideal world, the development database servers would also use the same architecture, but it's common to use cheaper systems for development. As an aside, it's amazingly common to use a different architecture for the test system—including not having a test system at all—but that's just asking for trouble.

For a large DW/BI system, 64-bit is the only way to go. But even for smaller systems, a modest investment in a 64-bit server and a decent amount of memory will pay for itself in system performance. We used to make fun of people who'd buy bigger hardware so they don't have to tune their DW/BI system. But in the case of 64-bit and large memory, it's a sensible thing to do.