



*transforming data into understanding<sup>®</sup> Series*

## **DATA WAREHOUSE APPLIANCES**

### **UNDERSTANDING APPLIANCE ARCHITECTURE**

**By James Newland**

Data Warehouse (DW) Appliances all advertise that they are 'fast', but what does that mean to you? How do their claims relate to each other?

The biggest roadblock to making the right DW appliance choice is in understanding how the different architectures relate to your business. In Part 1 of the series Data Warehouse Appliances, we will give you a brief introduction to some of the major differences between the DW appliance companies. This list is not exhaustive but will provide a good introduction to the most important issues, including:

- Column vs. Row Storage
- Proprietary and Commodity Hardware
- In-Memory Processing
- Relationship with Existing Architecture
- Shared Nothing Architecture

Should you need more information on any of these items or any other data warehouse appliance considerations, please give us a call – our consultants are happy to help.

### **Column vs. Row Storage**

Several data warehouse appliance companies offer databases that rely on columnar storage. While the idea is not new, it is catching on – and with good reason. For primarily ‘read’ applications, it makes great sense.

Columnar storage has several advantages over the traditional row-based storage used in most DW appliances. First, columnar storage only reads the exact columns on which you query while row-based has to read every column in the table. This leads to faster queries. Second, this storage method compresses data more aggressively. This compression can help you maximize your storage space.

So why not go with a columnar storage option? Columnar storage excels at ‘reads’ and does poorly with ‘writes’. That is, row-based storage has a distinct advantage over column-based storage when it comes to updating or inserting rows into tables. Columnar storage may not be appropriate for a data warehouse that requires near real-time updates.

Another potential disadvantage of the columnar design is that the natural physical data model form is the star schema. If you plan on creating a relational data model, a column-based storage DW appliance will lose some of its efficiency. A row-based architecture lends itself to a relational physical data model. Star schema logical views on top of a relational model can make row-based architecture more flexible from the reporting aspect.

So what is the ‘right’ solution for your business? Are you going to mostly read from your data warehouse appliance? Columnar storage could be right for you. Otherwise, row-based storage may offer more advantages.

### **Hardware – Proprietary vs. Commodity vs. Software-only**

Data warehouse appliances have three main options when it comes to appliance hardware – proprietary, commodity, and software-only (you purchase and configure your own hardware).

Proprietary hardware is built specifically for that DW appliance, generally by the same company that provides the appliance software. In this type of solution, the hardware and software are presented to the customer as inseparable. If you need to expand your system you will have to purchase additional hardware from the manufacturer. This hardware tends to be more expensive but is tuned to achieve maximum performance.

Some companies offer commodity standards-based hardware that is configured in a proprietary way. You may get some cost savings through this configuration but like pure proprietary hardware vendors, you will have no choice for hardware.

Commodity hardware is generally thought of as commercially available hardware that you can buy without the DW appliance to run other applications. It is based on commercially available, standards-based components available from many manufacturers. The appliance vendor will sometimes offer their solution optimized with a particular commodity-based hardware configuration, leaving you few choices. This option can be less expensive than proprietary hardware but no more flexible.

Finally, software-only vendors allow you to purchase your own hardware. Vendors with software-only offerings will specify an operating system (e.g. Linux) and provide recommendations on processors, storage capacity, and RAM. This option gives you more flexibility and may save money. Depending on the strength of your IT department however, it may be easier and cheaper to purchase a solution that comes with hardware.

### **In-Memory Processing**

In-memory processing systems eschew disk memory in favor of processing solely against random access memory (RAM). Searching data in RAM is much faster than finding data on disk-based data storage. Period.

Traditionally, data warehouses have relied upon racks of large hard drives, spreading data out between each of the drives. In this configuration, one of the major speed limiters is how fast a drive can read the data. In-memory processing eliminates the disk read, saving time and increasing performance.

Why would you NOT use in-memory processing? Data stored in-memory is not permanently stored – you still need hard-disk or other storage to perform backups. When an in-memory system goes down, that memory needs to be re-loaded when the system is brought back up. This is a consideration if you need a highly-available system. Additionally, memory capacity is still many times more expensive than hard-drive storage.

### **Relationship with Existing Architecture**

How will your new DW appliance co-exist with your existing architecture? Are you replacing some or all of your current solution? Are you consolidating data marts or augmenting your current capabilities?

There are probably more options than you think. If you want a complete replacement of your existing solution, some DW appliance vendors will buy back your old hardware as an incentive. The risk in a total replacement though is a potentially long, expensive implementation.

You can also choose to augment your existing solution by offloading some of the more intensive processing to a new solution. DW appliance vendors have come up with creative options for setting up new solutions in parallel with your old systems. While this can save money on expensive upgrades to existing systems, it also introduces data duplication and additional operational management—something well-designed data warehouses avoid where possible.

### **Shared Architecture**

'Shared Nothing' systems utilize self-contained processing units - each processor is given its own memory, I/O, and disk space. This type of architecture tends to be highly scalable (easy to increase capacity) as well as linearly scalable. With linear scalability you can add as much processing and storage as you need with performance increasing at a linear rate. That is, twice the number of processing units will (in theory) always double the performance.

With Shared Everything architecture, all processors have access to the same memory, inputs/outputs, and disk-space. Each shared resource puts a different strain on the ability to process in parallel and creates a potential bottleneck when different processors need the same resource. This architecture is difficult to scale and does not scale linearly. Doubling the number of processors will not result in twice the performance.

Hybrid architecture seeks to limit the poor scalability and performance problems of Shared Everything without resorting to every processor having its own resources. The effectiveness of the solution will vary widely on the architecture used as well as your data requirements.

Shared Nothing architecture allows better performance because the work is split up between many individual processing units working in parallel. In addition, this architecture is easily scalable without degrading performance. For true high-end data warehousing, a Shared Nothing or hybrid solution makes the most sense in the long-term.

### **Conclusion**

All of these architectural issues should influence your decision on which DW appliance will best meet your business requirements. The other major factor is of course, cost. In part II of this series, we will look at cost vs. performance issues including:

- Speed
- Storage Efficiency
- System Availability
- And More!

Understanding the architecture of each of the major DW appliances as well as where you can best target your IT dollars is critical to finding the best solution at the best price.

### **About Datric, Inc.**

Datric, founded in 2000, is headquartered in Charlotte, NC, USA. Datric is a premier provider of data integration and collaboration solutions, specializing in SAP and data warehousing. Datric consultants have experience both architecting and implementing data integration systems for many Fortune 100 companies. We combine a simple, time-honored approach to business with leading-edge technology to help you achieve your goals. For more information about our products, services and clients, please visit us at <http://datric.com>