THE CLIPPER GROUP
Navigator™
Navigating Information Technology Horizons℠
Published Since 1993 Report #TCG2007085 August 28, 2007

Sun's Honeycomb — A Sweet, Next-Generation Information Repository

Analyst: Anne MacFarland

Management Summary

The nature of information solutions (focused on information use, rather than just its generation, preservation or presentation) is different from solutions targeted at process optimization or throughput. In promoting information's use, you have to come to terms with the reality that different people use the same information for different purposes – in different ways. A researcher will use information (say, in chemical analysis) in one way, as part of her project. Her supervisor will see that information in a different fashion, as a costed asset that supports her project. Another colleague may see that information as something that can be relevant to what she or he is doing. Someone in a different field may use the chemical analysis to make, not a cure for cancer, but a new solvent. The power of this reuse is that it leverages the costs of acquiring the information – and it also drives quality of information as an inherent part of that value. If information is sloppily gathered or – in the worst case – fabricated, this will become evident sooner.

The cost of this reuse is that lots of metadata, and different kinds of metadata, will be associated, over time, with the information assets. Managing and leveraging this metadata will drive a need for iterative data services. Such services are possible because technology, unlike the libraries and other information repositories of old, supports re-indexing. Java application environments are a good fit for these use-driven services, which might include re-indexing and extraction of new entities, complex queries, pre-staged joins, and data validation. Such an approach is made possible by an architecture designed to optimize metadata management and queries as well as to store data.

File systems and other hierarchical structures, important organizational tools in traditional storage environments, presume knowledge of the path through a tree. This is not helpful in a large repository of diverse data sources to be held for a long time. An object approach to data, coupled with metadata and powerful search, fulfills requests for information in a natural, intuitive way that is less dependent on external structures. Sun's *Honeycomb* takes a similarly decentralized, symmetric cluster approach to data resilience and data migration, based on a double parity RAID-like philosophy.

The Sun *StorageTek 5800* system – known as Honeycomb –takes a flat, object-oriented approach. With embedded, extensible metadata query and *Storage Beans* data services, Honeycomb provides a flexible infrastructure for organizations focused on better use of huge repositories of unstructured data. Read on for details.

IN THIS ISSUE

Honeycomb Architecture

To the *Honeycomb* design team, storing metadata externally in a database or repository seemed to make less sense than storing it in the array. Localizing the rich metadata makes the object more self-contained and less reliant on external control structures (directories, access control lists) that may change over the years. With "local control" and locally-generated high availability (see *Physical Architecture*, below), the comprehensive control systems that manage a vast repository or archive can shed a layer of complexity.

With both data objects and metadata stored on the array, the processing power on each node and the instructions in the Storage Beans applications let routine administration be accomplished locally. (For example, how many degrees of resilience? What retention period?) In addition, local metadata can be manipulated and used, both synchronously, in response to a query, and asynchronously, in the background, to groom the data for extended use and different kinds of reuse.

Local metadata, in conjunction with Storage Beans,[1] could allow images to be stored once, and retrieved in different resolutions, as contrasted with storing multiple copies of the same image at different resolutions[2]. For this and other similar actions, the metadata is used as a filter to find the relevant objects against which Storage Beans can act. This kind of functionality could allow older data to be translated into newer or standard formats (a tedious task) at idle times.

Offloading this kind of functionality onto array-based processors would not interfere with the use of storage software and products like Hierarchical Storage Management (HSM), or with the functionality of repository applications. But, over time, it will let complex content and storage management policies be applied in a more rational, tiered way.

Physical Architecture

Honeycomb, also known as the *Sun StorageTek 5800 System*, is available as a full (32 TB) or half (16 TB) cell. One Cell is sixteen nodes of drives[3], a unit of service processors (in the form of a *SunFire 2100)*, and two level-two switches. This hardware is not an array meant for small-scale file sharing, nor is it for high-throughput transaction processing.

- Each node is made up of a 64-bit AMD Opteron processor, four 500-GB SATA Drives, three GBs of memory.
- The service processor tray is used to load and upgrade the software on the system. The Service Processors are *Opteron* 64-bit X2100, 3.3 GHz with 3 GB RAM and dual GbE connections.
- On top of that, two layer 2 switches do active-passive load sharing and contain the algorithms to place the data.

This is not a SAN in a box. There is no volume management. Honeycomb uses Reed-Solomon algorithms to implement RAID 6 on a multi-node scale. Each object is broken into seven segments – five data and two parity – placing each on a separate node. Segments are check-summed as they are written.

Because each object is spread about, if you lose a node, the other 15 nodes in the cell can quickly rebuild the information contained in the failed node, since the rebuild is done locally. For technology migration (installing new drives), you can take a tray out, replace new drives and let the Honeycomb self-heal. When you add another cell, you can elect to rebalance the data fragments cross the cells, rather than just expanding gradually into the new capacity. Sun calls this *Data Sloshing*[4]. These two features give the automated self-perpetuation that is needed in an array designed for long-term storage.

Logical Architecture

The classic IT practices of hierarchical storage management (HSM), or Network Attached Storage (NAS) can become counter-productive when data use is opportunistic. File systems are great for organizing an individual's or department's current information, but, over time, these arrangements can become a complexity that must be learned to be properly used.

With search, query, and indexing, less organization of information has to be done up front. When information is being held for reuse by people with multiple interests, tagging is preferable to static hierarchical arrangements. Even directories/folders can become an obstruction.

[1] Storage Beans are a concept that has excited early customers and other stakeholders. They are not apart of the original Honeycomb release but will be available in mid 2008. Preliminary thinking is important, for their use can profoundly change administrative strategies in many domains (database, storage, applications, system recovery).

[2] Storing multiple images not only uses more storage, but it creates naming ambiguities. Resolving the data to a particular resolution on retrieval requires more functionality but is organizationally *much* simpler.

[3] The 5800 contains 500 GB SATA drives. They will move to 1 TB drives when they become available.

[4] Sun calls retiring of data *Data Evacuation*.

For long-term use, the flatter the organization of information, the better. With objects, permissions and security can be precipitated to the object level. With storage beans for data services, enforcing and modifying these permissions, and checking the logs of use, can be performed as needed. The system becomes a control point, to which tasks are delegated by an administrator.

When an object is ingested, it gets a unique ID, as is true in all object stores. In Honeycomb, that *Object ID* (OID) is then *linked to the object's metadata as well*. This lets the focus of manipulation and retrieval move from the data object itself to the metadata that provides the link to retrieve the actual information. Over time, more links can evolve how the object is used without changing the object itself. Honeycomb, at this iteration, provides a software form of WORM, but Sun will not consider it a target for rigid compliance information until IT administrators are precluded from subverting the system. This capability will come in future releases.

Retrieval of data can be done in three modes.

- It can be via the Object ID.
- It can be by a SQL Search-like aggregation or integration of Boolean sets of attributes. Honeycomb supports flexible schemas that can address different attributes. It supports schema to pre-group logical groups of data, or collections, for faster retrieval when the need arises – a virtual staging.
- Or, the attributes can be built into a virtual file system view using WebDAV.[5] Such views can be defined for different roles.

With the WebDAV mode of retrieval, a user can point and navigate the virtual system. With the role-defined views, an administrator can segregate the attributes a user sees, so, as an example, an MRI machine repairperson can have images to test his repairs, but no patient information is exposed. This supports deployment of an n-tier permissions model. Honeycomb achieves its security by allowing for the configuration of authorized clients that can have access to the system via an IP. This can be coupled with front-end security applications that manage access to the system.

Sun offers a downloadable software development kit (SDK) and emulator to allow people to try out the system without committing to a purchase. It is available at sun.com.

Use of Honeycomb

Honeycomb is designed to hold fixed information. Digital content is by its nature, fixed. Of course, even transactional information becomes fixed when the transaction is complete (if it did not, its virtue as documentation and information for analysis would be nil). But Honeycomb is not meant for transactional computing or active database data, such as ERP stuff.

Honeycomb is targeted at large-scale information sharing that would benefit from the support of a flat namespace (no hierarchies) and metadata enhancement. Medical imaging and digital content repositories are early sweet spots. Honeycomb has been qualified as part of a Carestream Health (formerly known as Kodak Health Group) solution. Sun is in discussions with other partners about how DICOM medical information can take advantage of Honeycomb's capabilities. DICOM, like any standard, is inherently amenable to management by metadata.

Honeycomb is being used by university libraries who find themselves with a surprising breadth of digital content. Customers like Johns Hopkins University are not merely trying to emulate existing library and archive methodologies, but to focus more on flexible use of data. Sayeed Choudhury, Dean for Library Digital Programs and Hodson Director of the Digital Curation and Research Center, compared the difference in approach to the difference between early movies where a camera was focused on a theater's proscenium and recorded what happened on stage and a modern movie that features multi-camera action, special effects, and animation.

Choudhury also noted the significant challenge of keeping digital data long term. The data will be kept longer than any hardware, any software, or any repository system. Pervasive use of open standards is mandatory. And, hardware that will work well without imposing its own requirements is needed. As Choudhury put it, "We focused on scaffolding, not structure." Choudhury felt that Sun understood these requirements and was willing to go beyond the traditional vendor-customer role to produce a solution that would work on this new frontier.

Other early customers are expecting to be ingesting a terabyte of information a day, as part of large-scale digitization projects. Honeycomb can be (and is being) integrated with open repositories

[5] WebDAV is a web-based distributed authoring and versioning protocol based on HTTP developed by the W3C. It includes the basic functionalities of file properties, locking, name spaces, and ordered collections. Most current operating systems support WebDAV, and it is integrated with many commercial Enterprise Content Management products.

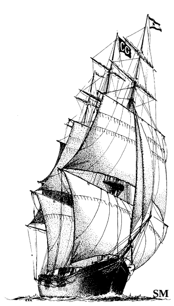
like *Fedora* and *DSpace*. Storage Beans are then expected to be developed by communities for industry-specific purposes. Sun and its customers are working to help optimize Fedora and will contribute the solution back to open source.

Honeycomb can also be the back end storage to SAM-FS Hierarchical Storage Management systems. Honeycomb can capture and store SAM-FS metadata. In high-performance computing systems, Honeycomb could sit behind Sun's *Thumper*[6], which is used as a front-side cache.

Conclusion

The old way of doing things is not the only way of doing things. By adding local services in the box, digital information can be better prepared for a variety of uses. If your storage can manage the metadata of the data it contains, retrieval of that data is much more straightforward. If it can rebuild and rebalance storage, it becomes more of a self-managing asset. For many application vendors, the idea of data services in the box to enrich their application functionality (a linked-service paradigm) gives them a way to add functionality without adding more complexity to the application itself. The combination of automated self-management with the ability to leverage application functionality within the box is very attractive.

If you are haunted by the inevitability of very-large, very-actively-used fixed information repositories becoming a critical asset to your organization, you may want to think about what kinds of infrastructure will best support them. Any idea that catches the attention of a wide range of beta customers and several application vendors should be on your list of new ideas worth considering. Check it out.



[6] The Sun StorageTek X4500 4U *Thumper* combines a four processor core x64 processor and 48 500-GB SATA drives in a balanced system design cell meant to be deployed as networked storage. For more information, see **The Clipper Group Navigator** *Sun Gains Edge with Sun Fire Extensions,* dated August 8, 2006, and available at http://www.clipper.com/research/TCG2006068.pdf.

About The Clipper Group, Inc.

The Clipper Group, Inc., is an independent consulting firm specializing in acquisition decisions and strategic advice regarding complex, enterprise-class information technologies. Our team of industry professionals averages more than 25 years of real-world experience. A team of staff consultants augments our capabilities, with significant experience across a broad spectrum of applications and environments.

- ***The Clipper Group can be reached at 781-235-0085 and found on the web at www.clipper.com.***

About the Author

***Anne MacFarland* is Director of Data Strategies and Information Solutions for The Clipper Group.** Ms. MacFarland specializes in strategic business solutions offered by enterprise systems, software, and storage vendors, in trends in enterprise systems and networks, and in explaining these trends and the underlying technologies in simple business terms. She joined The Clipper Group after a long career in library systems, business archives, consulting, research, and freelance writing. Ms. MacFarland earned a Bachelor of Arts degree from Cornell University, where she was a College Scholar, and a Masters of Library Science from Southern Connecticut State University.

- ***Reach Anne MacFarland via e-mail at Anne.MacFarland@clipper.com or at 781-235-0085 Ext. 128. (Please dial "128" when you hear the automated attendant.)***

Regarding Trademarks and Service Marks

Disclosure

Officers and/or employees of The Clipper Group may own as individuals, directly or indirectly, shares in one or more companies discussed in this bulletin. Company policy prohibits any officer or employee from holding more than one percent of the outstanding shares of any company covered by The Clipper Group. The Clipper Group, Inc., has no such equity holdings.

Regarding the Information in this Issue

The Clipper Group believes the information included in this report to be accurate. Data has been received from a variety of sources, which we believe to be reliable, including manufacturers, distributors, or users of the products discussed herein. The Clipper Group, Inc., cannot be held responsible for any consequential damages resulting from the application of information or opinions contained in this report.