Trading Technology Week

September 16, 2002 www.tradingtechnologyweek.com

Grid Computing Needs Parallel Data and Processing Layers

Michael V. DiStefano, CEO at consultancy Integrasoft, discusses how grid computing needs a parallel, easily accessible distributed data layer.

Grid computing evolved in the scientific and defense communities specifically to support parallel processing of large data sets, but it is quickly finding a new home among financial services firms.

Current financial uses of grid technology have centered around pricing and management of derivative and mortgage -backed instruments.

To get the most out of grid computing, users need to investigate how a flexible data grid can support what most people know as a processing grid.

For this article, we will define a processing grid as any distributed cluster of computing resources that provides an environment for sharing, managing, and distributing non-batch tasks based on configurable service level policies.

There are two dominant IT business cases driving the demand for processing grids in finance.

The first is the need to reduce operational costs associated with the maintenance of data centers. This is particularly a priority in firms where data center silos were built, and cost allocated, on a per business unit basis. The integration and dissolution of lines of business has created surplus capacity, and a drive to cannibalize that capacity for other business lines. Grids are ideally suited to this business case as they provide resource management, virtualization, scheduling, and reliability across silos of previously unrelated servers.

The second case is the need to provide production IT service levels on cheaper, easier to maintain hardware. Grids provide a necessary component of any effort to harness commodity hardware because they enable the creation of reliable, secure, and schedulable servers based on Intel platforms.

Most trading applications rely on a combination of static, derived, and real-time data to perform their business functions. Utilizing processing grids, these applications can be distributed to compute farms. However, these applications rely on data that is typically centrally stored. Because the data is not ubiquitously available to the compute farm, it is typically passed through a variety of ad hoc mechanisms including arguments, databases and flat files that limit grid implementations in two ways. Firstly, data available only centrally to a grid becomes a performance bottleneck. Secondly, grid nodes cannot efficiently cooperate and thus can address only problems that do not require coordination.

To address this critical issue, grid vendors need to develop a reliable, transactional, and real-time data grid plane, which I define as a set of functionality that facilitates the transparent access of shared data across a grid.

The plane provides both localized and distributed caching functions to grid compute servers to support intra-node and inter-node collaboration. Localized and distributed caching functions of the data grid plane include cache consistency, synchronization, garbage collection, on-demand loading, namespace access, object expiration and fault tolerance.

Grid tasks distributed over a data grid plane can utilize the caching functionality of the plane to coordinate and cooperate more effectively.

There are several vendors that can be used to create the data grid plane. Whatever one you choose, it should have an architecture that enables processing grids to share data and cooperate both on an intra- and inter-node basis.

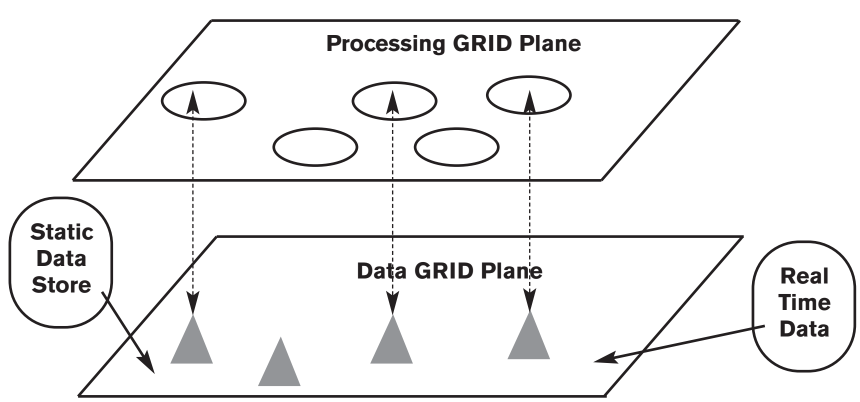
Grid computing must evolve if it is to address needs beyond the narrow niche of computational engines. With the implementation of the data grid plane, processing grids can be leveraged to address a much broader problem domain. Any application that can be built within the current paradigm of development, could in the near future be built to run against such a processing/data grid.

If this comes to pass, it will spearhead a paradigm shift in the application development and deployment process that will enable IT staffs to deliver applications as services and thereby lower the TCO, scale systems on demand, and better fulfill their SLAs to their business units.

Michael V. DiStefano can be reached at mdistefano@integrasoftware.com.

Grid Computing on Two Planes

To avoid data access bottlenecks, Integrasoft suggests users develop a distributed data grid in tandem with the processing grid.



Source: Integrasoft