Grid Computing with Oracle

An Oracle Technical White Paper
March 2005

Grid Computing with Oracle

INTRODUCTION

Grid computing is a new IT architecture that produces more resilient and lower cost enterprise information systems. With grid computing, groups of independent, modular hardware and software components can be connected and rejoined on demand to meet the changing needs of businesses.

Grid computing has increased momentum as the enterprise IT architecture of choice. Forrester Research reports that 37 percent of enterprises are piloting, rolling out, or have implemented some form of grid computing.[1] IDC identifies grid computing as the Fifth Generation of computing, after Client-Server and Multi-tier.[2] Leading businesses, such as Dell and the Chicago Stock Exchange, have begun deploying enterprise grids.[3]

Read more about how companies like Ohio Savings Bank are using grid computing at oracle.com/customers

The grid style of computing aims to solve some common problems with enterprise IT: the problem of application silos that lead to underutilized, dedicated hardware resources; the problem of monolithic, unwieldy systems that are expensive to maintain and difficult to change; and the problem of fragmented and disintegrated information that cannot be fully exploited by the enterprise as a whole.

Benefits of Grid Computing

Compared to other models of computing, IT systems designed and implemented in the grid style deliver a higher quality of service, at a lower cost, with greater flexibility. Higher quality of service results from having no single point of failure, a powerful security infrastructure, and centralized, policy-driven management. Lower costs derive from increasing the utilization of resources and dramatically reducing management and maintenance costs. Rather than dedicating a stack of software and hardware to a specific task, all resources are pooled and allocated on demand, which eliminates underutilized capacity and redundant capabilities. Grid computing also enables the use of smaller individual hardware components, which reduces the cost of each individual component and providing more flexibility to devote resources in accordance with changing needs.

GRID COMPUTING DEFINED

The grid style of computing treats collections of similar IT resources holistically as a single pool, while exploiting the distinct nature of individual resources within the pool. To address simultaneously the problems of monolithic systems and fragmented resources, grid computing achieves a balance between the benefits of holistic resource management and flexible independent resource control. IT resources managed in a grid include:*

- **Infrastructure** - the hardware and software that create a data storage and program execution environment;
- **Applications** - the program logic and flow that define specific business processes;
- **Information –** the meanings inherent in all different types of data used to conduct business.

Core Tenets of Grid Computing

Two core tenets uniquely distinguish grid computing from other styles of computing, such as mainframe, client-server, or multi-tier: virtualization and provisioning.

- With ***virtualization***, individual resources (e.g. computers, disks, application components and information sources) are pooled together by type then made available to consumers (e.g. people or software programs) through an abstraction. Virtualization means breaking hard-coded connections between providers and consumers of resources, and preparing a resource to serve a particular need without the consumer caring how that is accomplished.
- With ***provisioning***, when consumers request resources through a virtualization layer, behind the scenes a specific resource is identified to fulfill the request and then it is allocated to the consumer. Provisioning as part of grid computing means that the system determines how to meet the specific need of the consumer, while optimizing operation of the system as a whole.

The specific ways in which information, application, or infrastructure resources are virtualized and provisioned are specific to the type of resource, but the concepts apply universally. Similarly, the specific benefits derived from grid computing are particular to each type of resource, but all share the characteristics of better quality, lower costs, and increased flexibility.

Infrastructure Grid

Virtualize Infrastructure Resources: Pool hardware and systems software into a single virtual resource.

Provision Infrastructure Resources: Allocate capacity on demand based on policies to meet individual needs and optimize the system as a whole.

Infrastructure grid resources include hardware resources such as storage, processors, memory, and networks, as well as software designed to manage this hardware, such as databases, storage management, system management, application servers, and operating systems.

* Infrastructure, applications, and information assets are referred to here as “resources,” but in some cases, such as when resources used by one company are owned by another, it becomes more natural to talk about services. For our discussion of enterprise grid computing, the language of resources and services are interchangeable.

Virtualization and provisioning of infrastructure resources mean pooling resources together and allocating to the appropriate consumers based on policies. For example, one policy might be to dedicate enough processing power to a web server that it can always provide sub-second response time. That rule could be fulfilled in different ways by the provisioning software in order to balance the requests of all consumers.

Treating infrastructure resources as a single pool and allocating those resources on demand saves money by eliminating underutilized capacity and redundant capabilities. Managing hardware and software resources holistically reduces the cost of labor and the opportunity for human error.

Spreading computing capacity among many different computers and spreading storage capacity across multiple disks and disk groups removes single points of failure so that if any individual component fails, the system as a whole remains available. Grid computing also affords the option to use smaller individual hardware components, such as blade servers and low cost storage, which enables incremental scaling and reduces the cost of each individual component, giving companies more flexibility and lower cost.

Infrastructure is the dimension of grid computing that is most familiar and easy to understand, but the same concepts apply to applications and information.

Applications Grid

Application resources in the grid are the encodings of business logic and process flow within application software. These may be packaged applications or custom applications, written in any programming language, reflecting any level of complexity. For example, the software that takes an order from a customer and sends an acknowledgement, the process that prints payroll checks, and the logic that routes a particular customer call to a particular agent are all application resources.

Treating application logic as another resource in the grid enables better software reuse and creates applications that are easier to change.

Historically, application logic has been intertwined with user interface code, data management code, and process or page flow and has lacked well-defined interfaces, which has resulted in monolithic applications that are difficult to change and difficult to integrate.

Service-Oriented Architecture (SOA) has emerged as a superior model for building applications, and SOA concepts align exactly with the core tenets of grid computing. Virtualization and provisioning of application resources involves publishing application components as services for use by multiple consumers, which may be people or processes, then orchestrating those services into more powerful business flows.

In the same way that grid computing enables better reuse and more flexibility of IT infrastructure resources, grid computing also treats bits of application logic as a resource, and enables greater reuse of application functionality and more flexibility in changing and building new composite applications. Applications that are orchestrated from published services are capable of viewing activities in a business as a single whole, so that processes are standardized across geography and business units and processes are automated end-to-end. This generates more reliable business processes and lowers cost through increased automation and reduced variability.

Service Oriented Architecture

Building applications in the grid style of computing means using an SOA. The foundation of a SOA is a set of services – independent, well-defined encapsulations of software functionality that can be invoked over a network using heterogeneous platforms and execution environments. SOA, most simply, connects these independent services toward a larger purpose.

XML-based Web Services have emerged as the standard of choice for implementing Service-Oriented Architecture and, thus, for grid computing.

The recommended way to implement a SOA is using XML-based Web Services standards. Web Services will be successful where earlier distributed computing architectures have failed for three reasons: standards, broad adoption, and loose coupling.

First, the Web Services standards that define common interfaces (i.e. WSDL) and a common message construct (i.e. SOAP) are simple; also, standards for surrounding functionality, such as business flows and industry-specific data representations, are rapidly being defined.

Second, Web Services standards use underlying Internet standards (i.e. HTTP) that are already pervasive, which reduces the barrier to adoption. Web Services standards have been incorporated broadly into packaged software and adopted by companies across many industries, which reinforces the value of web service adoption.

Finally, XML enables Web Services to be more loosely coupled and less brittle in the face of change than alternative distributed architectures, such as distributed objects. The ideal Web Services implementation of SOA exposes relatively coarse-grained services and uses a more tightly coupled programming methodology for implementation and communication within a given service. XML-based Web Services enable loose coupling between services by facilitating a document-style exchange of information, which can be asynchronous and largely self-describing. XML further accommodates change and reduces brittleness by allowing content that is not understood by the service consumer to be ignored.

Combining Grid Computing for Infrastructure and Applications

Many of the vendors interested in grid computing have been focused on grid resource management as it applies to computers, storage, networking, and operating systems. Meanwhile, middleware vendors and applications vendors have been separately pursuing grid computing ideals in the domain of applications, under the designation of Web Services and SOA. These two worlds are now converging, and Web Services standards have emerged as the common basis to support language independent message delivery for all types of resources. New grid standards, defined by the Global Grid Forum, are building on the Web Services standards as defined by the W3C and OASIS standards bodies.

Information Grid

The third dimension to grid computing, after infrastructure and applications, is information. Today, information tends to be fragmented across a company, making it difficult to see the business as a whole or answer basic questions, such as, "Who are my customers? What products do they buy?" Information assets are underexploited.

In contrast, grid computing treats information holistically as a resource, similar to infrastructure and applications resources, extracting more of its latent value. Information grid resources include all data in the enterprise and all metadata required to make that data meaningful. This data may be structured, semi-structured, or unstructured, stored in any location, such as databases, local file systems, or email servers, and created by any application.

The core tenets of grid computing apply similarly to information as they do to infrastructure and applications. The infrastructure grid exploits the power of the network to allow multiple servers or storage devices to be combined toward a single task, then easily reconfigured as needs change. An SOA, or an applications grid, enables independently developed services, or application resources, to be combined into larger business processes, then adapted as needs change without breaking other parts of the composite application. Similarly, the information grid provides a way for information resources to be "joined" with related information resources to greater exploit the value of the inherent relationships among information, then for new connections to be made as situations change.

The relational database, for example, was an early information virtualization technology. Unlike its predecessors, the network database and hierarchical database models, in which all relationships between data had to be predetermined, relational databases enable flexible access to general-purpose information resources. Today, XML furthers information virtualization by providing a standard way to represent information along with metadata, which breaks the hard link between information and a specific application used to create and view that information.

Creating a true information grid means exploiting the latent semantic relationships among information resources in structure and unstructured forms within the enterprise.

Information provisioning technologies include message queuing, data propagation, replication, Extract-Transform-Load (ETL), as well as mapping and cleansing tools to ensure data quality. Data hubs, in which a central operational data store continually syncs with multiple live data sources, are emerging as a preferred model for establishing a single source of truth while maintaining the flexibility of distributed control.

While many of the building blocks required to build an information grid exist today, true information grids are still largely visionary. The significant missing link is the lack of explicitly managed information semantics (i.e. meanings). Gartner deems the practice "semantic reconciliation," which is the consistent and persistent understanding of data by all users and applications in an enterprise.[4] Without semantic reconciliation, for example, a business manager may see differing numbers for total headcount or repeatedly invest in rediscovering the meaning behind analytic reports as implicit knowledge departs in the minds of former employees.

The most expensive IT bungle ever – the Y2K bug – is also an example of failed semantic reconciliation.

Comprehensive management of data and metadata is the first step. Within an enterprise, most of the highest value information can be managed centrally using traditional means, such as relational databases and OLAP data stores, with semantic metadata embedded in ETL tools, business intelligence reporting tools, or enterprise application suites.

For some types of enterprise information that are less predictable, centralization and strict structure may be not only prove impractical but also undesirable. Email, instant messages, documents, and other less structured information assets should use the more flexible modeling of metadata provided through an XML-based repository to enable connections to be made across disparate sources, such as relating an email complaint from a particular customer, to a summary of outstanding service requests for that customer in a CRM system.

Some metadata describing information structure and semantics can and should be modeled explicitly, such as the structured fields in an email or document header information, or the definition of "headcount." Comprehensive modeling to accommodate any potential use of the information, however, will provide diminishing returns, so up-front semantic modeling should be supplemented with practical approaches to inferring meaning from indicators that already exist in information sources.

Google, for example, uses a series of heuristics such as analysis of referring hyperlinks, document structure, domain names, and time of update to draw reasonable conclusions about the relevance of web content in response to a given search request. Enterprise search engines in combination with classification techniques, semantic crawlers, and text mining also can infer meaning from enterprise information sources whether or not they are fully modeled.

What do relational databases, XML, and Google have in common? They all serve as base technologies for an information grid.

The goal for utilizing semantic technologies within the enterprise is the same as the goal of the Semantic Web. As defined by the W3C, "the Semantic Web provides a common framework that allows data to be *shared* and *reused* across application, enterprise, and community boundaries."[5] The primary missing link to effectively sharing and reusing data on the web is the lack of machine-readable standards for semantics of HTML-based content. Consequently, Semantic Web work focuses on establishing and promoting XML-based standards for representing generic relationships between information resources (i.e. Resource Description Framework - RDF) and for modeling the implicit knowledge associated with a particular domain which enables new inferences to be made (i.e. Web Ontology Language - OWL).

The future for semantic reconciliation within an enterprise will be a standards-based approach using XML-based standards such as RDF, OWL, and the Dublin Core Metadata Initiative. Until robust enterprise tools and applications support these standards, enterprises can begin deriving benefit from proprietary technologies by simply taking a deliberate approach to capturing and managing semantics.

Grid Resources Work Well Independently and Best Together

By managing any single IT resource – infrastructure, applications, or information – using grid computing, regardless of how the other resources are treated, enterprises can realize higher quality, more flexibility, and lower costs. For example, there is no need to rewrite applications to benefit from an infrastructure grid. It's also possible to deploy an applications grid, or an SOA, without changing the way information is managed or the way hardware is configured.

It's possible, however, to derive even greater benefit by using grid computing for *all* resources. For example, the applications grid becomes even more valuable when you can set policies regarding resource requirements at the level of individual services and have execution of different services in the same composite application handled differently by the infrastructure – something that can only be done by an application grid in combination with an infrastructure grid. In addition, building an information grid by integrating more information into a single source of truth becomes tenable only when the infrastructure is configured as a grid, so it can scale beyond the boundary of a single computer.

Oracle's Vision for Grid Computing

Oracle's product strategy is led by the vision of where we believe grid computing could lead in the future. Infrastructure resources managed in a grid will progress to the point that computing and storage capacity are delivered on demand like a utility. Applications in a grid will advance so that business and application logic are as massively connected and referenced as static web pages are on the Internet today, enabling frictionless, automated, global business between trading partners. Eventually, a global information grid will impart to every bit of digitally-represented information anywhere the same values we take for granted with relational databases; it will be as if all information resides in a single virtual database. All inherent relationships between information will be revealed, and anyone with appropriate authorization will have instantaneous access to all relevant information regardless of representation, location, or access method.

ORACLE 10*g* SOFTWARE

The "*g*" in Oracle 10*g* stands for grid. Oracle 10*g* was designed to help customers implement grid computing for infrastructure, applications, and information resources.

On the path toward this grand vision grid computing, companies need real solutions to support their incremental moves toward a more flexible and more productive IT architecture. The Oracle 10*g* family of software products implements much of the core grid technology to get companies started. And Oracle delivers this grid computing functionality in the context of holistic enterprise architecture, providing a robust security infrastructure, centralized management, intuitive, powerful development tools, and universal access. Oracle 10*g* includes:

- Oracle Database 10*g*;
- Oracle Application Server 10*g*;
- Oracle Enterprise Manager 10*g*;

- Oracle Collaboration Suite 10*g*.

White papers exist describing each of these products and each of the main feature areas, but summarized here are some of the key grid computing features for infrastructure, applications and information.

Infrastructure Grid

Server Virtualization

Oracle 10*g* software supports virtualization of all types of infrastructure resources.

Oracle Application Server 10*g* run-time services can be pooled and virtualized via application server clusters. Every service within the Oracle Application Server – HTTP Server, J2EE, Web Cache, Web Services, LDAP, Portal and others – can be distributed across multiple machines in a grid. Performance thresholds can be defined beyond which new application server instances can automatically be added and started (or relinquished) to process additional work on new nodes of a grid. Workload allocation across nodes can be controlled by resource consumption metrics, such as CPU or memory usage, or application-specific metrics, such as transaction throughput or JDBC connections, or workload can be provisioned based on schedules, such as peak times of day or end of quarter.

Oracle Real Application Clusters 10*g* enable a single database to run across multiple clustered nodes in a grid, pooling the processing resources of several standard machines. Oracle is uniquely flexible in its capability to provision workload across machines because it is the only database technology that does not require data to be partitioned and distributed along with the work. Oracle Database 10*g* Release 2 includes enhancements for balancing connections across cluster instances, based on policies.

Storage Virtualization

Oracle Automatic Storage Management (ASM), a feature of Oracle Database 10g, provides a virtualization layer between the database and storage so that multiple disks can be treated as a single disk group and disks can be dynamically added or removed while keeping databases online. Existing data will automatically be spread across available disks for performance and utilization optimization. In Oracle Database 10*g* Release 2, ASM supports multiple databases, which could be at different software version levels, accessing the same storage pool.

Grid Management

Because grid computing pools together multiple servers and disks and allocates them to multiple purposes, it becomes more important that individual resources are largely self-managing and that other management functions are centralized.

The Grid Control feature of Oracle Enterprise Manager 10*g* provides a single console to manage multiple systems together as a logical group. Grid Control manages provisioning of nodes in the grid with the appropriate full stack of software and enables configurations and security settings to be maintained centrally for groups of systems.

Another aspect to grid management is managing user identities in a way that is both highly secure and easy to maintain. Oracle Identity Management 10*g* includes an LDAP-compliant directory with delegated administration and in Release 2, federated identity management, so that single sign-on capabilities can be securely shared across security domains. Oracle Identity Management 10*g* closely adheres to grid principles by utilizing a central point for applications to authenticate users – the single sign-on server – while distributing control of identities via delegation and federation to optimize maintainability and overall operation of the system.

Applications Grid

Standard Web Services Support

Oracle Application Server 10*g* provides a cohesive SOA platform consisting of four main components. First, Oracle Containers for J2EE (OC4J) is a comprehensive, J2EE-certified service oriented architecture platform to develop and deploy simple and composite Web Services. Oracle 10*g* Release 2 includes support for J2EE 1.4, which includes multiple standards for implementing Web Services with Java.

Second, Oracle JDeveloper is the integrated J2EE development environment for modeling, developing, debugging, optimizing, and deploying Java applications and Web Services.

Third, the Oracle Application Development Framework (ADF), which is embodied by default in JDeveloper, provides an SOA framework based on the Model-View-Controller design pattern that will dramatically improve developer productivity.

Finally, Oracle TopLink provides an object-relational mapping solution and middle tier persistence layer that simplifies how J2EE applications map to and access relational and XML-oriented data.

Oracle Enterprise Manager 10*g* enhances Oracle's support for SOA by monitoring and managing Web Services and any other administrator-defined services, tracking end-to-end performance and performing root cause analysis of problems encountered.

Business Process Management

Business Process Execution Language (BPEL) provides a standard for orchestrating processes into complex business flows in a service oriented architecture. The Oracle BPEL Process Manager 10*g* is the industry's first native implementation of BPEL for modeling, deploying, and managing business flows in a standards-compliant way. It comprises an easy-to-use BPEL modeler, a scalable native BPEL engine, an extensible WSDL binding framework, a monitoring console, and a set of built-in integration services.

Oracle Business Activity Monitoring 10*g* (BAM) enhances Oracle's support for managing business processes by providing real-time visualization of business activities, alerts and notifications based on business-level thresholds, metrics and key performance indicators, and overall monitoring of business events.

Information Grid

Data Provisioning

Information starts with data, which must be provisioned wherever consumers need it. For example, users may be geographically distributed, and fast data access may be more important for these users than access to an identical resource. In these cases, data must be shared between systems, either in bulk or near real time. Oracle's bulk data movement technologies include Transportable Tablespaces and Data Pump.

For more fine-grained data sharing, the Oracle Streams feature of Oracle Database 10*g* captures database transaction changes and propagates them, keeping two or more database copies in sync as updates are applied. It also unifies traditionally distinct data sharing mechanisms – message queuing, replication, events, data warehouse loading, notifications, and publish/subscribe – all into a single technology.

Centralized Data Management

Oracle Database 10*g* manages all types of structured, semi-structured, and unstructured information, representing, maintaining and querying each in its own optimal way while providing common access to all via SQL and XML Query. Along with traditional relational database structures, Oracle natively implements OLAP cubes, standard XML structures, geographic spatial data, and unlimited sized file management, which virtualizes information representation. Combining these information types enables connections between disparate types of information to be made as readily as new connections are made with traditional relational data.

Oracle Collaboration Suite 10*g* takes advantage of the underlying file management capabilities in the database to offer secure content management and records management. Oracle 10*g* also exploits the combined OLAP and relational forms in the database by offering the first business intelligence tool, Oracle Discoverer 10*g*, with integrated OLAP and relational access. While business intelligence and collaboration applications are not part of the information grid *per se*, these capabilities are enhanced because Oracle 10*g* can significantly exploit more of the inherent relationships among various types of information, such as emails, calendar entries, and instant messages.

Metadata Management

Oracle Warehouse Builder is more than a traditional batch ETL tool for creating warehouses. It enforces rules to achieve data quality, does fuzzy matching to automatically overcome data inconsistency, and uses statistical analysis to infer data profiles. With Oracle Database 10*g* Release 2, its metadata management capabilities are extended from scheduled data pulls to handle a transaction-time data push from an Oracle Database implementing the Oracle Streams feature.

Oracle's series of enterprise data hubs (e.g. Oracle Customer Data Hub) provides real-time synchronization of operational information sources so that companies can have a single source of truth while retaining separate systems and separate applications, which may include a combination of packaged, legacy, and custom applications. In addition to the data cleansing and scheduling mechanisms, Oracle also provides a well-formed schema, established from years of experience building enterprise applications, for certain common types of information, such as customer, financial, and product information.

Metadata Inference

Joining the Oracle 10*g* software family in 2005 is the new Oracle Enterprise Search product. Oracle Enterprise Search 10*g* crawls all information sources in the enterprise, whether public or secure, including email servers, document management servers, file systems, web sites, databases, and applications, then returns information from all of the most relevant sources for a given search query. This crawl and index process uses a series of heuristics specific to each data source to infer metadata about all enterprise information that is used to return the most relevant results to any query.

Oracle's Commitment to Open Standards

Throughout the Oracle 10*g* product stack, Oracle implements open standards wherever those standards are defined. Oracle also works actively with standards bodies such as ANSI, W3C, OASIS, Global Grid Forum, Enterprise Grid Alliance, and the Java Community Process to promote standards creation. Oracle also works to innovate and introduce once those innovations into standards over time. Oracle's commitment to open standards helps protect and enhance customers' software investment.

Oracle also helps protect existing investment by supporting all popular hardware platforms and operating systems and by enabling incremental adoption of grid computing. Steps toward grid computing can begin within your existing environment, for example, by managing heterogeneous resources centrally or by integrating disparate information sources through a data hub.

CONCLUSION

Grid computing is the next generation model for enterprise computing based on the core tenets of virtualization and provisioning of every resource in IT. Grid computing delivers benefits of increased utilization and greater flexibility for infrastructure, applications, and information resources. Oracle 10*g* is the family of software products that supports grid computing and the software foundation that is enabling many companies and institutions to make the promise of grid computing a reality.

For more information, visit oracle.com/grid or call (800) 633-0615

REFERENCES

[1] Forrester, 2004. http://www.forrester.com/go?docid=34449

[2] IDC, 2004. http://www.oracle.com/technology/tech/grid/collateral/idc_oracle10g.pdf

[3] Mainstay Partners ROI Series, http://www.oracle.com/customers/studies/roi

[4] "Emergence of EIM Drives Semantic Reconciliation," Gartner, T-23-0615, 2004.

[5] W3C, 2004, http://www.w3c.org/2001/sw

"Grid's Place in the Service Oriented Architecture," Ian Foster, November 2004.

"What is Service Oriented Architecture," Hao He, April 2004.

"Architecture Patterns for Enterprise Semantic Web," Jeffrey Pollock, Network Inference, 2005.

"Semantic Information Architecture: Solving the Enterprise Data Problem," Zvi Schreiber, Unicorn, 2003.

"Evaluation and Market Report," M. Davis, D. Allemang, R. Coyne, TopQuadrant, 2004, http://wonderweb.semanticweb.org/deliverables/documents/D25.pdf

Web Services and Service Oriented Architectures: A Savvy Manager's Guide, Douglas K Barry, Elsevier Science, 2003.

Loosely Coupled: The Missing Pieces of Web Services, Doug Kaye, RDS Press, 2003.

The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management, M. Daconta, L. Obrst, K. Smith, Wiley Publishing, 2003.

Grid Computing with Oracle
March 2005
Author: Miranda Nash
Contributing Authors: Robert Shimp, Ken Jacobs
Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com