SEEING THE FUTURE WITH THE SUN™ VISUALIZATION SYSTEM

Scaling, Sharing, and Scheduling Visualization Resources

White Paper
February 2007

Table of Contents

Executive Summary

Visualization technology is increasingly essential, letting professionals across a broad spectrum of commercial and scientific disciplines see the information behind their data. As trained specialists seek to make sense of their complex numerical data, effective visualization technology can make the critical difference, leading to new insights, breakthroughs, and understanding. Using an array of graphical methods, visualization technology uses the visual and spatial parts of the human brain to better understand and interpret information, ultimately leading to broad benefits from applied use of critical data.

Visualization technology is now considered essential in a wide variety of industrial and scientific disciplines including Manufacturing, Medical, Energy, Financial, Entertainment, and Design. In all of these areas, data sizes are growing exponentially and more data than ever awaits evaluation and interpretation. Advancing technologies such as grid computing and sensor proliferation are creating new and larger data sets, along with more detailed and complex designs generated by sophisticated computer-aided tools. Not only must visualization systems scale to meet these demands, but they must also be re-usable by others, and flexible enough to allow collaboration and sharing of results across the organization or around the globe.

With these pressures, visualization technology has changed in fundamental ways — responding to new challenges and new market realities. Large, dedicated, and proprietary systems have been replaced by flexible multidisplay systems and integrated visualization clusters comprised of computers, computer graphics hardware, display monitors, projectors, multimedia, and I/O technologies. Standard TCP/IP Ethernet networks now provide sufficient bandwidth to satisfy visual acuity requirements for high-framerate image transfer. High-speed, low-latency interconnects such as InfiniBand facilitate multisystem visualization clusters. The result of these developments is that visual applications and services are now being virtualized, and consolidated in the datacenter, where resources can be effectively controlled and managed, and where critical data can be protected.

These changes can present challenges to organizations that find themselves having to design, build, and deploy complex visualization systems, in addition to concentrating on their core missions. Based on 25 years of experience with network computing, datacenter design, systems, and visualization technology, Sun is ideally positioned to help organizations deploy effective visualization infrastructure. The Sun Visualization System — including *Sun Scalable Visualization software* and *Sun Shared Visualization software* — provides flexible visualization solutions that are fully integrated and supported by Sun. The combination of flexibility and support along with key innovations means that organizations can solve their unique visualization requirements and get even the largest and most complex visualization solutions right the first time.

Chapter 1
Redefining Visualization

Once defined by dedicated, large-scale proprietary systems, visualization technology is changing in response to new demands and technical opportunities. Standardization around key 3D interfaces such as OpenGL® has brought some stability to the marketplace. At the same time, technological innovation on the part of graphics vendors has drastically reduced the cost associated with high-quality, high-performance 3D graphics accelerators. Now new innovations are delivering the benefits of these developments along with massively scalable and sharable graphics resources to more organizations in scientific, research, and commercial endeavors.

Advances in technology along with consolidation around commodity graphics accelerators have meant that some traditional visualization vendors have fallen by the wayside, even as others have prospered and thrived. Making the most of these trends, the Sun Visualization System integrates workstations, servers, networking, interconnects, graphics, and innovative software to provide both scalable and shared visualization solutions. Building these systems requires new approaches that combine the best aspects of high-performance graphics technology with new networking and system architectures, while preserving investments in applications and infrastructure.

The changing world of visualization

Along with scalable performance from high performance computing (HPC) systems and clusters, visualization plays a valuable role in helping scientists, researchers and professionals from a wide range of disciplines understand their data in visual terms. Visualization technology is increasingly indispensable to the solution of the most important and complex problems. By simplifying and distilling information for human perception, visualization is aiding discovery and decision making in a host of scientific and engineering domains. Important new opportunities for visualization are also emerging in financial markets, national security, and public health.

Exploring and solving these new problems places new demands on visualization technology:

- ***From terabytes to petabytes***
  High performance systems together with technologies such as clustering in high performance computing are enabling more and larger simulations. Advances in sensor technology and high-resolution digital imagery are likewise resulting in enormous volumes of data. Models and data sets have grown, from gigabytes to terabytes, and they will grow to petabytes in the near future. Effective visualization technology is now essential to help scientists and others recognize features and patterns in these expanding data sets.

- ***Demand for more detail***
  Growing data sets, more complex models, and high-resolution imagery are also causing increasing demand for the ability to see more detail. Some need very high resolutions to visualize minute details, outstripping even the most advanced display technologies. Others need very high 3D graphics performance and the ability to interactively and realistically render large numbers of triangles and geometric detail, dwarfing the individual capabilities of even the fastest individual graphics accelerator.
- ***Essential multidisciplinary collaboration***
  With growing complexity of both research problems and product design, collaboration among multiple disciplines is vital. Fortunately, the proliferation of high speed data networks now allows sharing of vast data collections among ever-wider scientific and technical communities. Visualization systems are now available that support effective collaboration for research and development even as they allow the virtualization of graphics and visualization resources. Solutions must be client agnostic, and must address challenges such as interactivity, security, access, and protection of vital data.

These challenges are taking place at a time of considerable technology advancement.

- The visualization technology landscape has changed dramatically, with traditional large-scale visualization vendors being replaced by high-performance solutions available at lower costs.
- Fast, truly full-duplex system busses such as PCI-Express have enabled graphics solution vendors to bring high quality, high-performance graphics to lower cost systems.
- Industry-standard systems based on high-performance SPARC® and x64 processors are providing new levels of performance.
- High-speed, low-latency interconnects such as InfiniBand allow effective clustering of visualization systems to address scalability.
- Innovative software facilitates the scaling of standard OpenGL applications across multiple systems and graphics accelerators, and the sharing of those applications to multiple heterogeneous systems across the network.

Visualization markets

Visualization requirements are diverse, and often unique to the individual organization. The sections that follow define visualization needs in key markets:

- ***Manufacturing***
  Manufacturing organizations need to be able to visualize, analyze, and interpret their designs from anywhere using standard mechanical computer aided design (MCAD) tools. Manufacturers likewise need secure and interactive remote access for 3D clients to help enable collaboration with partners and vendors who may

need access to key design data. In general, manufacturers need to increase the availability of graphics resources for faster, more productive product development, while reducing the cost of large-scale legacy graphics systems and over-provisioned 3D client systems.

- ***Oil and gas (energy)***
  As they seek to evaluate and interpret large-scale earth models and seismic and reservoir simulation data sets, energy companies need powerful visualization tools that give them insights into large volumetric data. Making more informed drilling decisions can have a direct impact on success or failure. Most now seek to leverage industry-standard hardware and software as they migrate away from expensive legacy visualization systems. Being able to centralize and secure valuable proprietary data sets and reservoir models in the datacenter is vital.
- ***Life sciences and research***
  In order to work with large scientific data sets, researchers and life scientists need the best visualization tools to facilitate better, faster, and more informed decision making. Visualization also provides a useful sanity check and model verification function. Many organizations need visualization systems that integrate with HPC compute grids and clusters. These organizations also need state-of-the-art development environments and tools to develop new applications and algorithms.
- ***Defense and government***
  Increased security requirements mean that defense, government, and military organizations need visualization systems that can help them perform large-scale simulations and "what if" national defense and tactical scenarios in near real time. Given the risks involved, they also need to extend security from data to graphics and visualization. Industry-standard, commercial-off-the-shelf (COTS) components help ensure faster procurements, speed system validation, and extend useful project lifetimes.

Visualization requirements

To be effective, visualization systems need to provide:

- Ways to modularly *scale* both resolution and 3D graphics performance beyond the limits individual systems and components
- Ways to *share* both important data and powerful graphics acceleration technology across the network
- Ways to *schedule* and manage graphics resources for maximum effectiveness and utilization

In the Sun Visualization System, network computing and innovative software architecture provide the critical difference.

The Sun Visualization System

Addressing the evolving needs of diverse visualization markets and organizations requires a fundamentally new approach, one that combines the best aspects of standard and open source software with innovative architecture. The Sun Visualization System exploits key technology advances to deliver scalability, performance, and collaboration inside a fully integrated and supported framework.

Exploiting multiple transition points in visualization technology

Traditional visualization models dictated that graphics acceleration resources were located close to the user. Large-scale visualization resources were typically usable only in a specialized and dedicated facility. Smaller visualization systems deployed heavily-provisioned graphics systems (with large network bandwidth, increased memory, and increased graphics resources) but they too were typically available only for a few select users due to cost. For direct rendering, data had to be distributed to wherever the graphics devices were located, complicating security and data integrity. Older indirect rendering approaches also put graphics acceleration hardware on the client desktop and sent (typically un-encrypted) OpenGL/GLX protocol over the network.

Today's larger data sets and applications create new and larger complications:

- Increased rendering complexity can easily outpace individual graphics accelerators
- Demand for increased resolution can easily outpace individual displays
- The need to move large amounts of model data can quickly outpace available network bandwidth

For example, slicing through multigigavoxel data sets such as those provided by the Visible Human Project (*http://www.nlm.nih.gov/research/visible/visible_human.html*) implies textures of at least 3 MB each. These textures must be regenerated on every frame with no reuse of texture data from frame to frame. In this case, the bandwidth demands of a traditional distributed approach would quickly outstrip the resources of a gigabit network connection and an unsatisfactory and non-interactive frame rates would result.

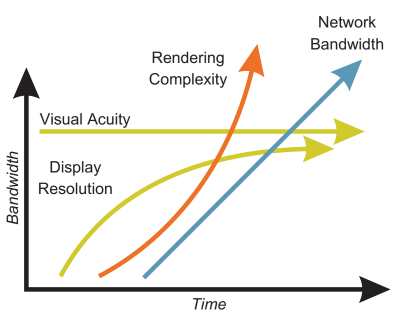

Figure 1. Graphics technology faces a transition point as network bandwidth scales to accommodate rendering complexity and resolution

Fortunately, complexity, resolution, network bandwidth, and cost are reaching a transition point that offers a new direction for scalable visualization solutions. It is now practical to transmit application images across a standard network for interactive display by a remote client at interactive rates, instead of sending actual model data across the network. As shown in Figure 1, even though single-display resolutions are not keeping pace with rendering complexity, network bandwidth is rapidly accelerating. Commonly-available networks and software codecs coupled with today's fast processors can now produce sufficient frame rates to satisfy people's basic ability to perceive visual information. Not only are networks and interconnects now delivering the raw speeds and low latencies to facilitate scalable multisystem visualization solutions, but open source technology such as VirtualGL is able to use both local and wide-area networking technology to provide remote visualization and collaboration.

Best of all, this model lets graphics servers, visualization clusters, and data remain in the datacenter where they can be secured and managed. Rendering is performed on the server side of the network, where there is a fast link between compute, graphics, and storage resources. With the 3D rendering occurring on the server side, only the resulting compressed or raw images must be sent to the client, resulting in a modest and constant bandwidth requirement no matter what the size of the 3D data used to generate them. This approach is possible thanks to system interconnects such as PCI-Express that provide the necessary read-back speeds from 3D accelerators installed in graphics servers, along with the latest multicore SPARC and AMD Opteron™ processors that provide sufficient processing to allow remote interactivity.

The attributes and benefits of Sun's approach

Unlike traditional approaches that dedicate graphics resources to each user, the Sun Visualization System increases productivity and potentially lowers costs by more effective use of available large-scale computational and visualization resources. Key aspects of this approach include:

- ***Centralized visualization resources***
  Keeping large data sets within the datacenter and providing remote access to centralized computational and visualization resources provides security benefits as well as better utilization of the shared multiprocessor systems and their attached graphics hardware. Centralizing these resources also decreases their administrative costs.

  Centralized graphical and visualization systems can be installed and configured once, and then used from anywhere. Users have convenient access to more resources than they would typically have installed on their desktops. Providing wider access to scalable visualization can increase insights and productivity for larger numbers of people.

Commercial, off-the-shelf (COTS) graphics hardware is rapidly gaining both in performance and very sophisticated capabilities. Sending parallel graphics streams to multiple commodity graphics accelerators is now preferable to designing special-purpose graphics hardware.

- ***Workstation and server-attached commercial, off-the-shelf (COTS) graphics***
  Scaling visualization performance to keep up with increasing data set size requires sufficient I/O and memory bandwidth to rapidly deliver data to the graphics accelerators, and also requires algorithms that will scale across many graphics pipelines. Sun has always strived for balanced and tuned system design, and integration of multiple graphics devices for scalable visualization can be accomplished either through large-scale server-attached graphics or through graphics clusters. Large-scale servers generally have better performance scalability due to high bandwidth and low latency from main memory to the graphics devices, along with a large shared memory that can cache more data. Graphics clusters are often cost-effective, and provide a higher degree of modularity and scalability by utilizing multiple low-cost systems connected by a high-bandwidth InfiniBand interconnect or on-board gigabit Ethernet.

  Sun Scalable Visualization software combines these two approaches as desired. Both workstations and servers can be utilized as render nodes combined with powerful NVIDIA graphics solutions. Workstations include the Sun Ultra™ 40 M2 workstation. Servers with both large numbers of processors, and high aggregate I/O bandwidth, such as the Sun Fire™ X4600 server can be connected to external high-performance PCI-Express graphics such as the NVIDIA Quadro Plex Visual Computing System (VCS) (*http://www.nvidia.com*). Both workstations and servers can, in turn, be used as render nodes in InfiniBand-connected graphics clusters. A flexible approach allows the mixing and matching of the right sized systems (in terms of processor count, memory, and cost) to high-performance graphics in various configurations. These configurations are built to customer specification by trained Sun visualization specialists and Sun Customer Ready Systems (CRS).

- ***Interactive remote access to graphical applications***
  Many visualization applications are still written for workstations, where the display is always local. Unfortunately, modifying individual graphics applications to add remote capabilities can be expensive and time consuming. Sun Shared Visualization software uses VirtualGL technology to enable users to run existing OpenGL applications interactively with interactive performance.

  Sun Shared Visualization software also provides infrastructure for interactive remote access to centralized graphics and visualization resources. Exhibiting some of the best performance in the industry in terms of rendered frame-rates, Sun Shared Visualization software is tightly integrated with Sun Scalable Visualization software, allowing visualization resources to be both scaled and shared across and between organizations. Support for a wide range of clients means that any authorized person can collaborate and view 3D data remotely and interactively.

- ***Controlled sharing and load balancing***
  Beyond accessing centralized graphics resources on an *ad hoc* basis, organizations want to be able to allocate graphics and visualization resources from a shared pool. By effectively time-sharing resources, the total cost of ownership (TCO) is further reduced, and utilization is increased, even as access to resources is carefully controlled. Authorization and authentication as well as secure data transfer are important features of any remotely accessible computing resource. and are also applicable to sharing visualization resources effectively.

  By extending grid computing concepts to graphical resources, the Sun Visualization System brings these capabilities to visualization infrastructure. This innovation lets multiple users and their applications share graphical resources effectively — sharing visualization servers and even individual graphics accelerators simultaneously. Device dedication and reservation for high priority sessions and device sharing for non-critical sessions are combined with special features such as advance reservation of visualization resources.

- ***A choice of platform and operating system***
  Organizations have their own unique reasons for selecting both computing platforms and the operating systems that run them. Legacy applications may have certain requirements or organizations may simply prefer particular software environments. In spite of a proprietary past, visualization technology must now be flexible and honor and protect existing investments in applications.

  The Sun Visualization System is largely based on open source software, and is designed to work with across multiple hardware architectures and operating systems. Graphics servers can select the advanced open source Solaris™ Operating System (OS) or Linux. Visualization clients can run the Solaris OS, Linux, UNIX, or Windows operating systems. With a range of innovative data-center features, the Solaris OS supports both SPARC and x64 architectures.

A fully integrated and supported visualization solution

Visualization systems can be complex to design and build. The necessary integration of networks, high-speed interconnects, computing systems, graphics accelerators, and display technologies along with software selection and configuration can be daunting and time consuming. Unfortunately, as the industry has consolidated around commercial off-the-shelf (COTS) graphics and open source software, much of the burden of designing and building visualization systems has been pushed back onto the organizations that use them. Sun's approach addresses these issues by offering:

- Tightly integrated software components built to work together
- Highly configurable solutions to fit a wide range of needs
- Integration, installation, and configuration of systems, graphics, networks, and software into pre-cabled racks that are ready to deploy
- Design and integration with other key visualization equipment such as projectors, switchers, and haptic devices by partnering with certified display integrators, Sun CRS, and Sun's professional services group

Sun has considerable visualization experience and expertise, having designed and deployed both scalable and shared visualization systems for a variety of diverse purposes around the world. For instance, Sun has implemented a grid-enabled interactive visualization system at the University of Texas, Texas Advanced Computing Center (TACC). Now the Sun Visualization System brings the expertise gained from that experience and others to help take the time and complexity out of deploying visualization solutions. The principal components of the Sun Visualization System are:

- *Sun Scalable Visualization software*, including tuned and supported versions of ParaView, Open Scene Graph, and Chromium for a wide variety of integrated multisystem visualization clusters constructed from multidisplay Sun x64 workstations and servers as well as single systems with multiple graphics cards
- *Sun Shared Visualization software*, providing tuned and supported versions of VirtualGL, Virtual Network Computing (TurboVNC), and extensive Sun N1™ Grid Engine software integration to aid with allocating and managing visualization resources on both UltraSPARC® and x64 systems
- *High-performance graphics accelerators*, such as NVIDIA graphics solutions for Sun x64 workstations and servers as well as Sun XVR graphics for Sun UltraSPARC based systems
- *Powerful Sun SPARC and x64 workstations and servers,* including the Sun Ultra 40 M2 workstation, and Sun Fire X2200 M2 and X4600 M2 servers based on the AMD Opteron processor as well as the Sun Ultra 45 workstation and Sun Fire V445 server based on the UltraSPARC processor
- *High-speed, low-latency InfiniBand interconnects* including support for the Sun IB Switch 9p InfiniBand switch as well as supported InfiniBand switches from third parties such as Voltaire and Cisco

Figure 2 illustrates a composite view of the Sun Visualization System software stack, depicting the integration of multiple hardware and software components. Not only are multiple hardware and software systems integrated to build scalable visualization systems, but those resources are then made sharable with a wide range of clients across the network. For example, users can schedule and start a job on a graphics cluster equipped with Sun Scalable Visualization software using Sun N1 Grid Engine software, while remotely viewing, interacting, and collaborating using Sun Shared Visualization software on a range of client systems.

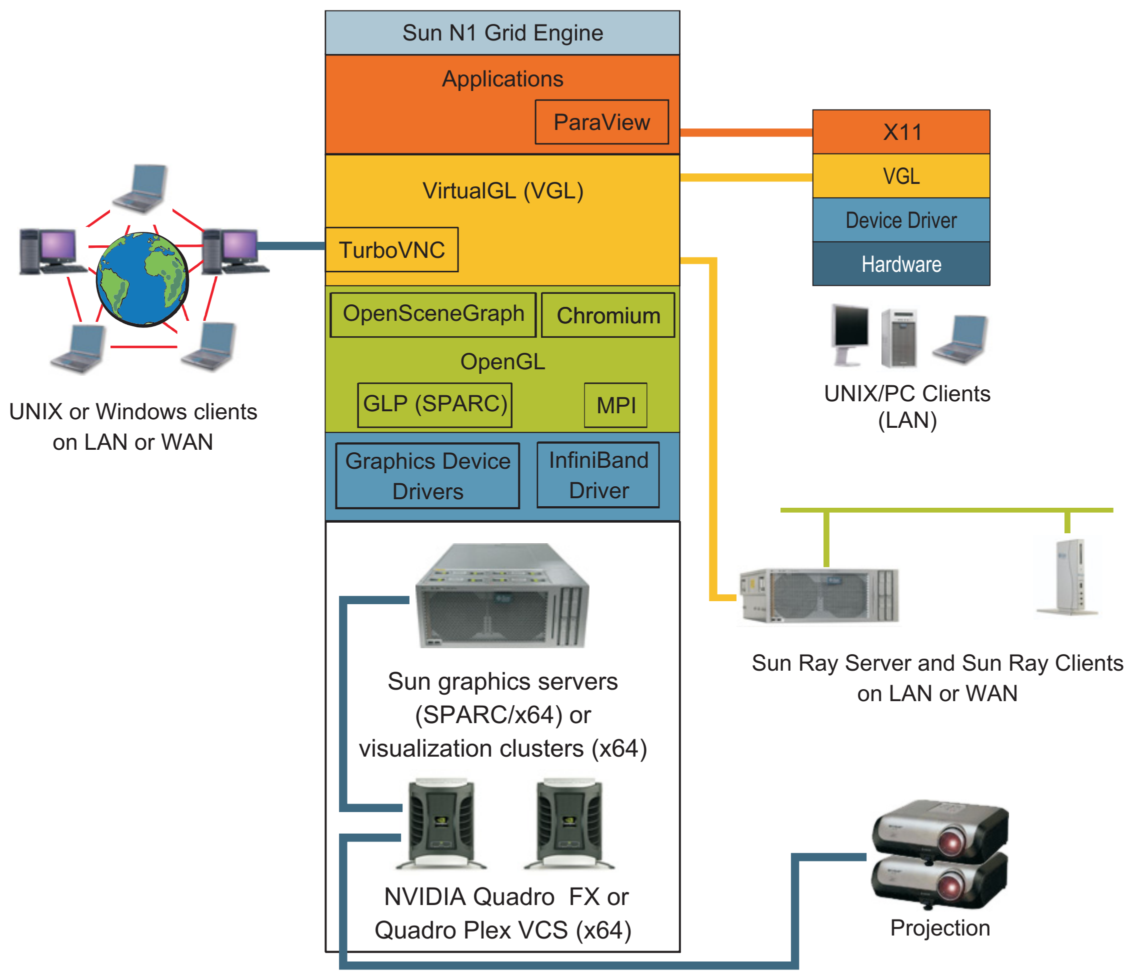

Figure 2. Sun visualization software stack (composite SPARC and x64, not all configurations supported)

To help ensure successful deployment of potentially complex visualization systems, Sun Scalable Visualization software is available through Sun CRS as part of an integrated hardware and software solution. Sun Shared Visualization software can be downloaded and used at no cost, with licensing and support plans available for purchase if desired. Elements of Sun Scalable Visualization software and Sun Shared Visualization software are described in the sections that follow.

Chapter 2
Sun Scalable Visualization Software

A variety of parallel rendering toolkits and open source clustering solutions have been available for some time, helping organizations to design and build scalable visualization solutions that expand beyond single graphics accelerators and individual systems. Unfortunately, these clustered visualization systems can be complex to implement and configure with multiple graphics accelerators, multiple systems, complex networking, and a long list of software components that need to work well together. In spite of their importance to the organization, critical tuning and support for these solutions was typically not available.

By providing an integrated and supported solution, Sun Scalable Visualization software provides a number of distinct advantages to organizations that use visualization technology:

- Seamless access to highly scalable visualization resources
- The ability to scale visualization power with the size of the data
- High-performance graphics for use with large shared memory and/or large CPU counts to maximize performance
- A choice between large and small visualization servers
- Transparent scaling of existing OpenGL visualization applications
- The ability to tailor multipipeline applications for best performance with a choice of toolkits such as ParaView, OpenSceneGraph, and Chromium

In addition to these benefits, security and total cost of ownership (TCO) can both be greatly improved by moving visualization resources and valuable data into the datacenter where they can be controlled and carefully managed.

Scaling mechanisms

The wealth of open source software dedicated to scaling standard OpenGL applications means that organizations can choose the approach that best fits their application requirements. Unlike some vendors, Sun offers a choice of open source scalable software support. In addition, Sun Scalable Visualization software is provided in the context of a tuned and supported platform that offers a range of scalable hardware and software solutions. The following sections discuss some popular options.

Chromium clusters

Chromium clusters (*http://www.chromium/sourceforge.net*) represent a successful middleware solution for scalable visualization in a clustered environment. A simple Chromium cluster is depicted in Figure 6. Chromium adopts a streaming model by organizing stream processing units (SPUs) into a directed acyclic graph. This architectural design provides a flexible method to scale data using sort-last render partitioning, or scale display using sort-first render task partitioning. Hybrid render task partitioning approaches are also supported. Special SPUs can be inserted in each subgraph to customize the rendering of some pipelines.

Chromium provides the following scalable rendering modes:

- *Tiled display (sort-first partitioning)* — The primary scalable rendering mode for Chromium is tiled display support, with or without Distributed Multiheaded X (DMX). This mode is useful for partitioning the screen space to enable building tiled displays and power walls from multiple displays. With these approaches, the display resolution can vastly exceed the capabilities of a single display device. With DMX, 3D windows and other application windows show on the same window system running on DMX. Without DMX, the 3D window is mapped to the tiled display, and the application windows are displayed on the application host's display. Render tasks are partitioned using a tile-sort SPU to perform sort-first partitioning. Software frame synchronization among the tiled screens is supported, so buffer swaps can occur at the same time on all tiles.
- *Sort-last composition* — When there is too much geometry data for a single graphics accelerator to process, Chromium also provides a parallel rendering API that allows developers to rewrite applications to partition their data sets in the object space. This approach distributes the workload to a cluster of chromium render nodes and and then composites the results of the several pipelines back together. A binary swap composition SPU composites *N* screens in *log(N)* composition steps. Both depth-buffer and alpha blending composition are supported. A compression SPU is also provided to minimize the impact of data transfer. For complicated applications, combining sort-first and sort-last is possible to gain performance and flexibility.
- *Sort-first-readback* — For fill-rate limited applications, a tile-sort SPU partitions rendering tasks into several rectangular regions, and the render results are read back from each pipeline using a readback SPU. This approach avoids the high performance cost of software composition.

The Chromium implementation provided with Sun Scalable Visualization software includes a Message Passing Interface (MPI) communication module to facilitate inter-node communication, independent of the choice of high-speed interconnect. In addition, Sun's implementation provides integrated configuration and support for multichannel, multipipe power walls using DMX. "Overlapping" support and mullion compensation support are provided to help configure and blend the overlap of multiple projectors and monitors. Bug fixes and tuning are provided to optimize performance on Sun x64 platforms.

ParaView

ParaView (*http://www.paraview.org*) is an open source parallel rendering application that includes a built-in scalable visualization paradigm. Both serial and Message Passing Interface (MPI) versions are supplied. When the serial version is used, a group of render servers and data servers can be started on different hosts with one client talking to them based on a configuration file. The MPI version can be started with a single `mpirun` command with each MPI process containing a chain of data filters, rendering, and composition. The MPI version of the data server and render server can also be started by separate `mpirun` commands, so that proper compute resource and render pipelines are allocated based on the problem size.

ParaView supports a tiled display mode (without data/object space partitioning) as well as object space partitioning using a tree composition. A group of user-configurable filters is employed to generate low-resolution geometry from large data sets to improve interaction. ParaView runs on both standalone Sun visualization servers with multiple graphics pipelines and can also run on larger Sun Scalable Visualization configurations. Sun Shared Visualization software can provide remote display of ParaView results.

OpenSceneGraph

Similar to SGI's Performer graphics library, OpenSceneGraph (*http://www.openscenegraph.org*) is a popular scene graph API for scalable rendering. OpenSceneGraph was designed for systems with multiple graphics devices, with each corresponding to a display, either a visualization server or a graphics cluster. With OpenSceneGraph, the rendering is divided into separate phases:

- APP (or pre-CULL) in which the dynamic user data is updated, including the position of the camera(s) as well as positional and attitudinal, and behavioral updates of moving objects
- CULL, in which the scene is sorted by objects visible in the viewing frustrum and display lists are built for the next phase
- DRAW, provides traversal of the display list and issuing of OpenGL calls.

The OpenSceneGraph API is multithreaded, with a thread each assigned to the APP phase, CULL phase, and DRAW phase, allowing rendering on multiple pipelines to run in parallel. Also, the next frame's APP and CULL phases execute in parallel with the execution of the DRAW phase of the current frame, helping to assure maximum parallelization. Sun Scalable Visualization software provides scripts for generating a camera configuration file to support "overlapping" when using projectors, or mullion compensation for power walls.

NVIDIA Scalable Link Interface (SLI) support

Developed by NVIDIA, the Scalable Link Interface (SLI) is a method for having two or more graphics accelerators share the load of graphics processing. Depending on the configuration, applications can benefit through increased screen real estate (resolution), increased geometric rendering performance, or increased quality. Using SLI and a special connector, multiple graphics accelerators can be "ganged" together to drive a single display. The resources of the linked graphics accelerators can then be combined to drive higher geometric performance (SLI Frame Rendering Mode) or to drive higher image quality (SLI full screen antialiasing (FSAA) mode). Though SLI support is provided on systems such as the Sun Ultra 40 M2 workstation, other approaches are available to provide higher levels of multisystem scalability.

Multidisplay graphics support in Sun x64 systems

Coupled with multidisplay visualization APIs and NVIDIA graphics, Sun x64 workstations and servers based on the AMD Opteron processor provide an ideal visualization platform. Individually, these systems can be configured with multiple displays to serve as powerful visualization servers that deliver both visualization and high-performance computational capabilities. For example, a single Sun Ultra 40 M2 workstation equipped with two NVIDIA Quadro FX 5500 graphics cards can drive a Sony 4K projector. These systems can also be combined into massively-scalable graphics clusters as a part of the Sun Visualization System as described in later in this chapter.

Sun x64 systems for scalable visualization

Select Sun x64 workstations and servers are available for configuration in visualization clusters powered by Sun Scalable Visualization software (Figure 3). Together these systems provide a broad range of performance and capabilities while each offers high performance, large memory support, and multiple high-speed PCI-Express graphics options. These systems are engineered with a balanced design to provide the essential performance and throughput so critical to HPC and visualization applications.

Figure 3. The Sun Ultra 40 M2 workstation along with Sun Fire X2200 M2 and X4600 M2 servers support multiple graphics accelerators and displays and are the key building blocks for Sun Scalable Visualization clusters.

Table 1 provides system details and multidisplay capabilities for supported Sun x64 workstations and servers, including NVIDIA graphics solutions available from Sun. These configurations can be used as rendering nodes in visualization clusters.

Table 1. Multidisplay support on select Sun x64 workstations and servers

System	Processors	Maximum Memory	PCI-Express slots	Supported NVIDIA Graphics Solutions	Maximum Displays
Ultra 40 M2 workstation	One or two Next Generation AMD Opteron 2200 Series dual-core processors	Up to 32 GB	Two x16 slots, two x4 slots	Up to two Quadro FX 5500	4
				Up to two Quadro Plex VCS	8[a]
Sun Fire X2200 M2 server	One or two Next Generation AMD Opteron 2000 Series dual-core processors	Up to 64 GB	Two x8 slots	One Quadro Plex VCS (in 8x or 16x PCI-Express slot)	4 or 8[a]
Sun Fire X4600 M2 server	Four to eight Next Generation AMD Opteron 8000 Series processors	Up to 128 GB	Four x8 slots, two x4 slots	Up to two Quadro Plex VCS	8[a]

a.Eight display support provided with a single NVIDIA Quadro Plex VCS Model II or with two Model I or III. Only a single NVIDIA Quadro Plex VCS Model II is supported per any one system.

NVIDIA Quadro FX graphics solutions

Sun x64 workstations such as the Sun Ultra 40 M2 workstation provide support for a host of NVIDIA graphics solutions from the NVIDIA NVS and Quadro FX product lines (including NVIDIA NVS 285 and Quadro FX 1500, 3500, and 5500 graphics cards). With a large (1.0 GB) texture memory, the NVIDIA Quadro FX 5500 graphics card (Figure 4) is particularly well suited to visualization applications supported is by the Sun Scalable Visualization software.

The NVIDIA Quadro FX 5500 is an ultra high-end 3D graphics card with dual display capability, ideal for CAD, DCC, and visualization applications. Armed with a radically new graphics architecture engineered to address the most demanding challenges, the NVIDIA Quadro FX 5500 delivers 33.6 GB/sec. memory bandwidth and a 256-bit memory interface, with support for 1024 MB of ultra-fast GDDR2 memory. Featuring two dual-link DVI connectors, the NVIDIA Quadro FX 5500 offers some of the industry's best image frame rates and quality at resolutions up to 3840 x 2400.

Figure 4. NVIDIA Quadro FX 5500 graphics card and back panel

The Sun Ultra 40 M2 workstation supports up to two NVIDIA Quadro FX 5500 graphics cards with support for NVIDIA SLI technology. NVIDIA Quadro G-Sync cards are used with multidisplay and multisystem graphics configurations to provide Genlock and Framelock synchronization.

NVIDIA Quadro Plex Visual Computing System (VCS)

While servers typically lack the power, cooling, and expansion profiles that are required by modern graphics accelerators, the advent of the NVIDIA Quadro Plex Visual Computing System (VCS) has meant that scalable server-based graphics is now a welcome reality. By providing an external, rack-mountable visual computing system, horizontally- and vertically-scalable Sun x64 servers can now host high-end graphics capabilities, and multiple graphics processing units (GPUs) can easily be added to servers via a 16x or 8x PCI-Express connection.

The NVIDIA Quadro Plex VCS is available in three models as shown in Table 2.

Table 2. Multidisplay support on select Sun x64 workstations and servers

	Model I	**Model II**	**Model III**
NVIDIA Quadro	NVIDIA Quadro FX 5500 (2)	NVIDIA Quadro FX 4500 X2 (2)	Dual NVIDIA Quadro FX 5500 SDI (2)
Number of Quadro GPUs total	2	4	2
Total frame buffer	2 GB (1 GB / GPU)	2 GB (512 MB / GPU)	2 GB (1 GB / GPU)
Option	NVIDIA Quadro G-Sync	NVIDIA Quadro G-Sync	NVIDIA Quadro SDI
Channels	4 dual-link DVI	8 dual-link DVI	2 dual-link DVI + 4 single-link HD SDI or 2 dual-link DVI + 2 dual-link HD SDI
Genlock/frame lock	Yes	Yes	Yes
Frame synchronization	Yes	Yes	Yes
HD SDI	No	No	Yes (4 dual link, fill or key)
Full Scene Antialiasing (FSAA, max per channel)	32x SLI FSAA	64x SLI FSAA	32x SLI FSAA
Maximum number supported in Sun x64 systems	2	1	2

Front and rear views of an NVIDIA Quadro Plex VCS Model III are provided in Figure 5. All NVIDIA Quadro Plex VCS models can also be configured with an optional rack-mount kit.

Two Quadro Plex are rack mounted side-by-side, requiring only 3U of rack space.

Figure 5. Front and rear view of an NVIDIA Quadro Plex Visual Computing System (VCS) (Model I shown)

Flexible and scalable x64 based visualization clusters

Scalable visualization configurations can be built from a range of powerful Sun x64 workstations and servers in myriad combinations. In addition, Sun Scalable Visualization software configurations are also supported by Sun Shared Visualization software. As a result, graphics resources including individual graphics servers and graphics clusters can be shared via VirtualGL on a fine-grained basis. For example, a Chromium master node can also use VirtualGL to distribute visual results via Sun Shared Visualization software. More information on Sun Shared Visualization software is provided in *Chapter 3*.

InfiniBand interconnects for visualization clusters

High-speed, low-latency interconnects are essential for building clustered visualization solutions that provide both scalability and high levels of interactivity. InfiniBand interconnects have proven themselves especially useful in this regard and Sun Scalable Visualization systems can be built using a choice of InfiniBand switches depending on scalability needs:

- ***Sun IB Switch 9p***
  The Sun IB Switch 9p switch supports clusters with up to eight rendering nodes and a single master node. Sun Visualization Systems with up to 64 displays can be built using a single Sun IB Switch 9p. Each switch provides nine 10/30 Gbps ports and a true non-blocking switch fabric with aggregate bandwidth of 540 Gbps and a fall-through latency of 86 nanoseconds.

- ***Voltaire Grid Switch ISR 9024***
  Visualization clusters require large data transfers at high speeds, and Voltaire's InfiniBand DDR solutions keep up with imaging data rates without sacrificing image quality. The Voltaire Grid Switch ISR 9024 supports clusters with up to 23 rendering nodes and a single master node, facilitating clusters with up to 184 displays based on a single switch. Each fully non-blocking switch provides twenty-four 10 Gbps or 20 Gbps ports in a 1 rack-unit (1U) chassis. Though not a Sun product, the Voltaire Grid Switch ISR 9024 can be configured through Sun Customer Ready Systems (CRS) as a part of a Sun Visualization System.

Massively scalable visualization cluster configurations

With a wealth of powerful x64 systems, a wide range of Sun Visualization Systems with Sun Scalable Visualization software can be designed and built to suit individual needs. For perspective, Table 3 provides a number of maximized Sun Visualization System configurations with various combinations of master nodes, render nodes, graphics accelerators, and InfiniBand switches.

Table 3. Sun Scalable Visualization software supports a wide range of x64 based cluster configurations based on a choice of NVIDIA graphics solutions and InfiniBand switches

Master Node (max. memory)	Render Node (max. memory)	Graphics Accelerators (max. displays/node)	Maximum Displays, Memory per Cluster
Sun Ultra 40 M2 workstation (32 GB)	Sun Ultra 40 M2 workstation (32 GB)	2 x NVIDIA Quadro FX 5500 graphics cards (4 displays)	Sun IB Switch 9p: 32 displays, 256 GB Voltaire Grid Switch 9024: 92 displays, 768 GB
		2 x NVIDIA Quadro Plex VCS (8 displays)	Sun IB Switch 9p: 64 displays, 256 GB Voltaire Grid Switch 9024: 184 displays, 768 GB
Sun Fire X2200 M2 server (64 GB)	Sun Ultra 40 workstation (32 GB)	2 x NVIDIA Quadro FX 5500 graphics cards (4 displays)	Sun IB Switch 9p: 32 displays, 320 GB
		2 x NVIDIA Quadro Plex VCS (8 displays)	Voltaire Grid Switch 9024: 184 displays, 800 GB
	Sun Fire X2200 M2 server (64 GB)	1 x NVIDIA Quadro Plex VCS (4 or 8 displays)[a]	Sun IB Switch 9p: 64 displays, 576 GB
Sun Fire X4600 M2 server (128 GB)	Sun Ultra 40 workstation (32 GB)	2 x NVIDIA FX 5500 graphics cards (4 displays)	Sun IB Switch 9p: 32 displays, 384 GB
		2 x NVIDIA Quadro Plex VCS (8 displays)	Voltaire Grid Switch 9024: 184 displays, 864 GB
	Sun Fire X4600 server (128 GB)	2 x NVIDIA Quadro Plex VCS (8 displays)	Sun IB Switch 9p: 64 displays, 1152 GB Voltaire Grid Switch 9024: 184 displays, 3072 GB

a.Eight display support provided with a single NVIDIA Quadro Plex VCS Model II

Visualization clusters can be built from workstations and servers, and from combinations of the two. For example, Sun x64 servers such as the Sun Fire X2200 M2 server and X4600 M2 server can be used as master nodes to drive visualization clusters of Sun Ultra 40 M2 workstation based render nodes equipped with NVIDIA graphics solutions. NVIDIA FX 5500 graphics solutions can be installed in the render nodes or multiple NVIDIA Quadro Plex VCS can be used to double the number of available displays with the same number of render nodes. Truly massive Sun Scalable Visualization clusters can also be built entirely from Sun x64 servers with Sun Fire X2200 or X4600 M2 servers acting as both master nodes and render nodes equipped with multiple NVIDIA Quadro Plex VCS. All Sun Scalable Visualization Systems are built, racked, configured and tested by Sun Customer Ready Services (CRS).

A Sun Scalable Visualization deployment example

A specific real-world example of a Sun Scalable Visualization system is shown in Figure 6. Deployed at the newly-opened London Centre of Nanotechnology (LCN), this Sun based visualization cluster is being used to harness the world-class expertise of researchers across the physical, engineering, and biomedical sciences, from University College London and Imperial College, to meet the needs of society and industry.

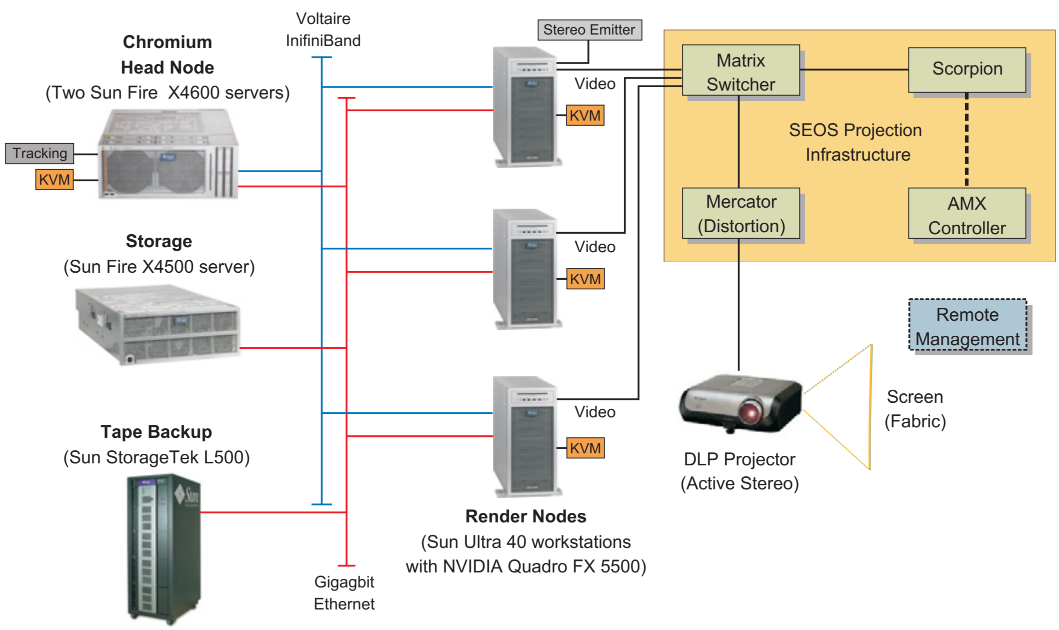

Figure 6. A Scalable Sun Visualization system deployed at the London Center for Nanotechnology, University College London and Imperial College

The use of leading edge modelling and visualization technologies is helping researchers from multiple disciplines to work together on a variety of projects, enabling people with diverse skills to collaborate in new ways. This collaboration allows healthcare (in the case of LCN) as well as others fields such as energy and information technology to tackle the challenge of understanding enormous new data sets.

The Sun Visualization System at LCN consists of powerful graphics clusters using Sun Scalable Visualization software, built with Sun Fire X4600 servers and Sun Ultra 40 workstations driving NVIDIA Quadro FX 5500 graphics cards. The resulting cluster provides high-quality graphics and exceptional rendering performance for 3D and textured surfaces. The digital video outputs from the visualization cluster are delivered via SEOS projection infrastructure to a stereo DLP projector. An integral SEOS Scorpion source compositor integrates multiple stereo input sources simultaneously.

Sun Shared Visualization software can also be used to distribute these results to authorized scientists and researchers, no matter what their location or client architecture.

Chapter 3
Sun Shared Visualization Software

Sun Shared Visualization software enables the use of remote graphics resources (as well as CPUs, memory, and storage) over the network in place of resources on the user's desktop. Graphics servers can be located in the datacenter where they can be effectively shared and managed. Storage, compute, and graphics processing can all be tightly coupled and secure in the datacenter or server room. The result is a configuration that can offer considerably more resources than the typical desktop system, often resulting in better performance than would be possible by running applications locally.

Sun Shared Visualization software also works together with Sun Scalable Visualization software, allowing centralized graphics resources to be dynamically allocated with fine granularity to serve the needs of the organization. For instance, the visual results of a Chromium cluster can be distributed when the Chromium head node is configured with Sun Shared Visualization software. Through this integration, graphics servers can now serve multiple clients serially or even simultaneously, aiding in collaboration and facilitating greater utilization of graphical resources including both systems and graphics accelerator hardware. Centralized graphical resources can be effectively divided, allocated, and re-allocated on demand, allowing users to build smaller or larger systems as their needs dictate. The sections that follow provide background and describe the details and architecture for Sun Shared Visualization software.

Traditional challenges: direct and remote graphics approaches

To appreciate the important innovations in Sun Shared Visualization software, it is helpful to understand how direct and remote graphics have been architected in the past. Historically, most applications were written to run in a workstation model, where the graphics acceleration and display hardware shared the same system with the application (Figure 7).

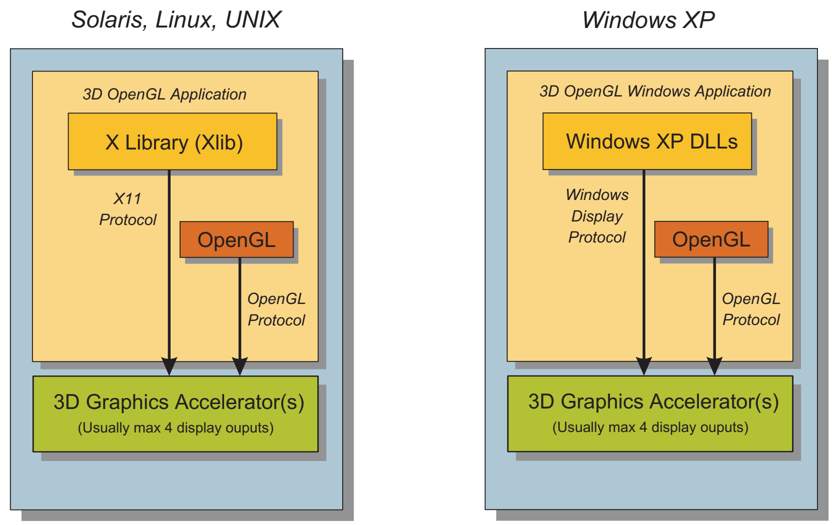

Figure 7. Local system based graphics

With tightly-integrated systems, this approach occasionally worked well, but had limited scalability and required dedicated and costly graphics resources at the desktop. Data, too, had to be available to the desktop as well as key applications. Unfortunately, dedicated resources were often limited to a few individuals and were typically under-utilized much of the time.

Modifying the dedicated approach only slightly, an application server model evolved where the application itself can run on a remote system (Figure 8). In this model, too, the graphics accelerator remains on the client system, close to the display hardware and the X11 server. Un-encrypted X protocol and its GLX graphics extensions move over the network between the application server and the client system.

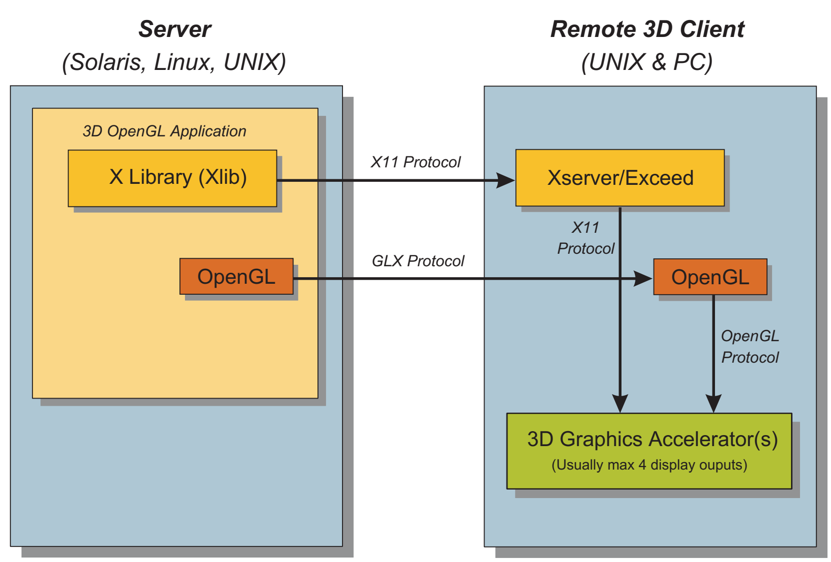

Figure 8. Traditional remote X server graphics

Unfortunately, this approach was ultimately limited, being useful only if the data being rendered were small and relatively static, if display lists were used, and if the network featured relatively high bandwidth and low latency. Furthermore, display list applications remained fairly rare since they were often not suitable for the particular type of rendering performed by the application. Rendering non display-list applications resulted in large amounts of network traffic as every vertex call had to be passed over the network on every frame as well as texture coordinates and the textures themselves. In addition, many OpenGL extensions assume direct access to graphics hardware, and thus did not function properly in an indirect rendering environment.

Sun Shared Visualization software overview

Exploiting key technology advances and innovative architecture, Sun Shared Visualization software effectively extends virtualization concepts to graphics and visualization applications and resources. Similar to successes with compute grids for computational workloads, the result is that graphics and visualization resources can be flexibly added to resource pools and used from almost anywhere. At the same time, effective shared visualization must:

- Support powerful graphics acceleration technology, including both multiaccelerator configurations as well as server-based graphics and multisystem clusters
- Serve the broadest range of clients with interactive performance, without arbitrary licensing or technology limitations
- Provide flexible resource management for graphical infrastructure

Sun Shared Visualization software addresses these challenges with a new approach and architecture that places essential components where they can provide the best performance without compromising utility for the organization or access for users. Based on the Sun sponsored open source VirtualGL project, Sun Shared Visualization software:

- Transparently plugs into any OpenGL application (Solaris OS or Linux) at run time
- Redirects 3D rendering into pixel buffers (Pbuffers) on the server's graphics accelerator
- Reads back complete images from the graphics accelerator, compresses the images in a platform-optimized fashion, and transmits the image to the client
- Decompresses and displays the images on the client display

Beyond merely distributing graphics, this approach also provides considerable investment protection in both applications and systems. For example, organizations can often migrate existing graphics systems into the datacenter to act as graphics servers while replacing desktop systems, such as Sun Ray™ ultra thin clients.

Sun Shared Visualization software components

Sun Shared Visualization software is tightly integrated with other key software elements to help enable visualization across a wide range of client systems as well as providing effective management of graphical resources in grid computing environments. Sun Shared Visualization features include:

- Integration with Sun N1 Grid Engine provides easy access, sharing, and management of centralized graphic resources as well as advance reservation
- Integration with Sun Ray ultra thin clients for low cost graphics display
- Integration with Turbo Virtual Network Computing (VNC) software with optimized compression for display to remote clients on slower or high-latency networks (such as the Internet)

VirtualGL

3D applications in thin client environments have traditionally been severely limited in terms of framerate and perceived lack of interactivity. Many implementations forced 3D applications to use slow software rendering. Others forced the transmission of 3D commands and data over the network to be rendered on the client system.

Open source VirtualGL (*http://www.virtualgl.org*) provides a better solution, bridging real hardware-accelerated 3D rendering capabilities with the benefits of thin clients. With VirtualGL, the 3D rendering commands from the application are intercepted at runtime, and redirected onto the server's local 3D graphics accelerator hardware. The resulting rendered 2D images are then read back from the hardware, compressed and optimized for network transmission and sent to the client for display. This approach produces a completely seamless shared 3D environment that allows thin clients to perform well enough for them to take the place of many types of 3D workstations.

VirtualGL offers distinct advantages, including:

- *Full-featured OpenGL support* — VirtualGL provides full OpenGL support (version 1.0 or later), including stereo, multisampling, overlays, and other features. Built-in support is provided for SSL, and VirtualGL can also work with SSH tunneling. VirtualGL supports a wide variety of servers and clients. The Solaris OS (SPARC and x86/x64 platforms) and Linux are supported on graphics servers while clients can run the Solaris OS, Linux, and Windows XP.
- *Integral image compression* — VirtualGL provides visually lossless JPEG compression for image transmission, natively accelerated using IPP (x86), Medialib (SPARC), or Quicktime (Mac)[1].
- *High levels of interactive performance* — Outpacing most competitive shared visualization products, VirtualGL can sustain 25-30 frames per second for a full-screen application on a standard x86 client over a 100mbps network (using 40-60 Mb/sec). Full MT-hot pipelined server and client software provides for effective scalability on multicore and multiprocessor systems.

1. A VirtualGL client for Apple Macintosh is provided by VirtualGL. Though expected to work, this client is not provided or supported with Sun Shared Visualization software and must be downloaded separately.

TurboVNC

For displaying to remote clients on slow or high-latency networks such as the Internet, Sun Shared Visualization software also includes Virtual Network Computing (VNC) software with optimized compression (known as TurboVNC). VirtualGL is integrated with TurboVNC, reading back images from the graphics accelerator and passing them uncompressed to TurboVNC's "proxy" X server on the graphics server host. The TurboVNC server then compresses the images for viewing by one or more TurboVNC clients. TurboVNC is completely backward compatible with other VNC distributions and can be installed onto the same system as another VNC distribution without interference.

TurboVNC is capable of sending nearly 20 megapixels per second over a 100 Mb/second local area network with perceptually lossless image quality. TurboVNC can deliver between 10 and 12 megapixels per second over a 3 Mb/second broadband connection at reduced (but usable) image quality. Throughput is quoted as the true measure rather than quoting frame rates with fixed resolutions since users can also change the window size to increase or decrease performance.

Multiple TurboVNC clients can share viewing and even interaction with the running program(s), helping to enable collaborative use of graphics applications. Users can take turns using their mouse to control the TurboVNC X server's cursor and entering keyboard input. Read-only clients can observe the TurboVNC session without providing input.

Sun N1™ Grid Engine software integration

To help with the location, allocation, and management of networked graphics resources, Sun Shared Visualization software provides graphical extensions to Sun N1 Grid Engine software (*http://www.sun.com/software/gridware*). The extensions allow Sun N1 Grid Engine software to allocate graphics resources in the same way that it allocates CPU and memory resources to computational jobs. In an environment that has multiple execution servers and/or multiple graphics accelerators on given hosts, Sun N1 Grid Engine software can select both suitable computational and graphics resources by load balancing, selecting least-recently-used resources. Jobs are also scheduled on servers based on user-assigned criteria such including architecture, desired numbers of processors or amounts of memory, and required graphics resources.

Sun N1 Grid Engine software also controls application startup, so users need not log into graphical servers at all. Job scripts can specify requirements and options to Sun N1 Grid Engine software. For example, these options could specify the number of required or requested graphics accelerators as well as specifying processor type(s) and operating system(s) required by the application.

Advance Reservation Server for Sun N1 Grid Engine software

To effectively use visualization resources in a shared fashion, computational resources must be coordinated to be available when people are ready to use them. Job queuing systems typically provide a capability known as Advance Reservations (AR), helping to coordinate systems and other required elements.

Provided with Sun Shared Visualization software, the Advance Reservation server adds this capability to Sun N1 Grid Engine software. This software helps ensure that reservations are not scheduled that would conflict with each other. Proper reservations avoid oversubscribing of available computational or graphics resources by multiple users and applications.

Industry-leading interactive performance

For locally rendered graphics, frame rate ultimately depends on the application, the complexity of the data to to be rendered, and the graphics acceleration hardware available. Even in a locally-rendered environment, graphics applications can have very low frame rates due to large complex models or insufficient graphics resources. Remote display of graphics often provides an advantage in that more graphics resources can be applied to the rendering task than would otherwise be available locally. Fortunately, 3D graphics and visualization applications place slightly lower demands on frame rates than do video applications.

In order for shared visualization software to prove useful, it must provide truly interactive levels of performance for users working from client systems. Remote graphics rendering performance with VirtualGL is subject to a host of factors, including compression of frames on the graphics server, network bandwidth, and client rendering performance. Under ideal circumstances, VirtualGL is capable of delivering 1280 x 1024 frames to a client over a high-speed network at 20-25 frames/second. This rate is typically fast enough to be perceived as interactive by the user. It is important to note that even on a 1600 x 1200 resolution desktop system with local 3D graphics resources, a 3D window is rarely as large as 1280 x 1024.

Among remote rendering solutions, VirtualGL compares quite favorably in terms of performance and interactivity. Sun Microsystems engineers have performed tests against a number of competing shared visualization solutions. In Sun's testing, VirtualGL and VirtualGL combined with TurboVNC provided superior frame rates than those of competing solutions.

Sun UltraSPARC and x64 systems for shared visualization

Sun Shared Visualization software supports both UltraSPARC and x86/x64 systems in the capacity of visualization servers for VirtualGL. Capabilities of Sun x64 based workstations, servers, and accelerators are described in Chapter 2 (Sun x64 systems for scalable visualization).

For customers with investments in UltraSPARC processors and the Solaris OS, Sun Ultra 45 workstations and Sun Fire V445 servers (Figure 9) provide scalability, continuity, and investment protection. With support for multiple UltraSPARC IIIi processors, large memories, and several Sun XVR-2500 graphics accelerators, these systems can support multiple displays and act as capable graphics servers running software releases including Solaris 8, 9, and 10.

Sun Ultra 45 workstation Sun Fire V445 server

Figure 9. The Sun Ultra 45 workstation along with Sun Fire V445 servers support multiple Sun XVR-2500 graphics accelerators and displays, sharable through Sun Shared Visualization software

Based on leading-edge architecture and technology, the Sun XVR-2500 graphics accelerator delivers high-quality graphics, provides a wide range of functionality, and offers exceptional rendering performance for 3D and textured surfaces. Features of the Sun XVR-2500 graphics accelerator include:

- 256 MB of GDDR3 unified memory
- Hardware-based texture-mapping and programmable shaders
- Support for two displays per Sun XVR-2500 graphics accelerator

With support for up to two PCI-Express x16 format graphics cards, both Sun Ultra 45 workstations and Sun Fire V445 servers can drive up to four displays for immersive visualization or multidisplay power walls. Of course, these systems can also act as VirtualGL graphics servers, allowing their resources to be shared via Sun Shared Visualization software to a range of available clients. Table 4 provides system details and multidisplay capabilities for Sun UltraSPARC based workstations and servers.

Table 4. Multidisplay support on select UltraSPARC workstations and servers

System	Processors	Maximum Memory	Supported Graphics Accelerators	Maximum Displays
Sun Ultra 45 workstation	One or two UltraSPARC IIIi processors	Up to 16 GB	Up to two Sun XVR-2500 graphics accelerators	4
Sun Fire V445 server	Two to four UltraSPARC IIIi processors	Up to 32 GB	Up to two Sun XVR-2500 graphics accelerators	4

In addition, Sun Shared Visualization software is also supported on other Sun graphics accelerators for UltraSPARC systems, including the XVR-600 and XVR-1200 graphics accelerators. This support means that organizations can retain their investments in both applications and systems, while displaying their existing applications on new client systems. Operating system and graphics accelerator support for both UltraSPARC and x86/x64 systems in Sun Shared Visualization software is shown in (Table 5).

Table 5. Supported server platforms and graphics accelerators

Processor architecture	Operating System releases	Graphics Accelerators
UltraSPARC	Solaris 8 OS and later	Sun XVR-600 Sun XVR-1200 Sun XVR-2500 (for mono or stereoscopic display)
x86 and x64	Solaris 10 OS Red Hat Enterprise Linux 3 and 4 SuSE 9 and 10	NVIDIA Quadro FX series NVIDIA Quadro Plex VCS series

Broad server support means flexibility and added longevity for existing applications without recompilation or changes to source code. For instance, applications running on UltraSPARC systems can be easily shared with Sun Ray ultra thin clients or x86/x64 clients running the Solaris OS, Linux, or Windows (Table 6). All client systems must support 24 bit (or 32 bit) per pixel true-color display. For stereographic support, a 3D graphics accelerator must be installed on the client.

Table 6. Supported client platforms

Processor architecture	Operating System	Supported OS releases
UltraSPARC	Solaris OS	Solaris 8 and later
x86 and x64	Solaris OS or Linux	Solaris 10 OS Red Hat Enterprise Linux 3 and 4 SuSE 9 and 10
x86	Windows	Windows XP with Exceed 2006 or later, or Exceed 3D 2006 or later for stereographic support

As an entry-level example, an x86 notebook computer equipped with an NVIDIA graphics chip, and running the Solaris OS or Linux can act as a Sun Shared Visualization server. This configuration is often used as a demonstration or proof of concept implementation.

VirtualGL: Designed to serve a broad range of client types

To satisfy the widest possible range of network and client scenarios, VirtualGL is designed to provide flexible operation through support of multiple operating modes. These modes determine whether compression is employed, and how resulting graphical images are distributed to visualization clients. Virtual GL modes include:

- *Direct Mode* for sending compressed images to a dedicated VirtualGL client
- *Proxy Mode* for sending uncompressed images to a proxy mechanism (such as TurboVNC)
- *Sun Ray Mode* for sending images directly to Sun Ray ultra thin clients

The details of how these modes work with with various clients are provided in the sections that follow. For simplification and clarity, the illustrations that follow show a single graphics accelerator. It is important to note that Sun Shared Visualization software works together with Sun Scalable Visualization software. As a result, the graphical results of multiple parallel graphics pipelines (such as a Chromium cluster) can be shared across the network in similar fashion.

VirtualGL Direct Mode — shared visualization across local area networks

In Direct Mode, the VirtualGL server component compresses the rendered output images from 3D applications directly on the 3D application server and sends the resulting JPEG format images directly to the client. Direct Mode implies use of a client application (`vglclient`) to decompress and display incoming images. VirtualGL works by:

- intercepting the 3D commands from the application
- rerouting the commands to the server's 3D graphics accelerator hardware
- reading back the rendered 3D images
- compressing the images using a high-speed image codec, and
- sending the compressed images to the client on a dedicated network socket

On the client side, Direct Mode requires an X server and the `vglclient` application to be running on the client machine. The X server processes 2D drawing commands from the application, renders the application's user interface, and feeds input events (key strokes, mouse movements, etc.) back to the application. The `vglclient` application provides decompression and composites the resulting images into the appropriate X window (Figure 10).

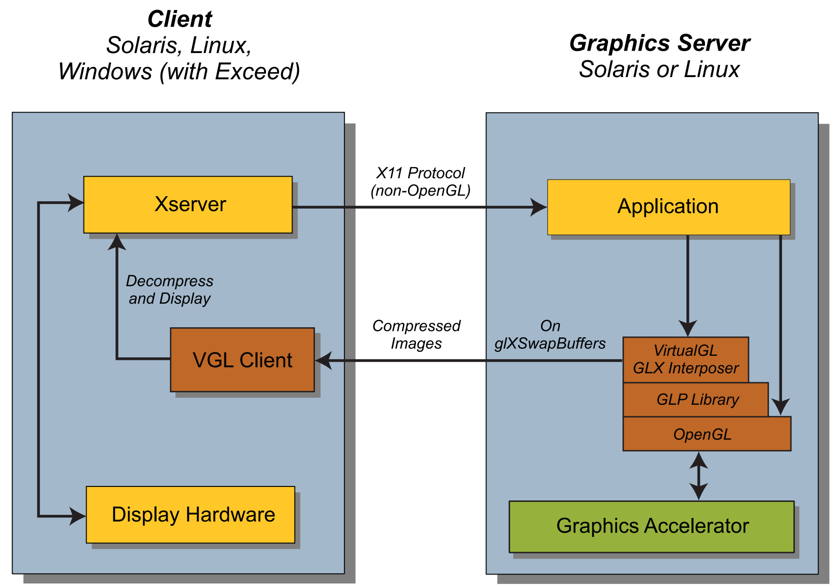

Figure 10. In VirtualGL Direct Mode rendered and compressed images are sent directly to the `vglclient` *application for decompression and display*

In general, VirtualGL Direct Mode is the fastest way to run VirtualGL on a local area network, and it provides a seamless end user experience that is indistinguishable from running the application locally. Direct Mode is typically used to run data-intensive 3D applications, while interacting with them remotely from a notebook computer or PC located elsewhere in the same building or campus.

VirtualGL Proxy Mode — visualization across wide area networks

VirtualGL Proxy Mode provides a solution for sending the results of 3D rendering across low-bandwidth networks. This mode allows the use of a variety of "X proxies" including VNC, Sun Secure Global Desktop, NoMachine NX, or Hummingbird Exceed, helping to enable a wide range of clients. In Raw or Proxy Mode, there is no need for an X server nor the `vglclient` program to run on the client machine.

As with Direct Mode, VirtualGL reroutes the 3D commands from the application to the server's 3D hardware for acceleration, and reads back the rendered images. However, in Raw Mode, Virtual GL does not perform its own image compression. Instead, the rendered 3D images are drawn into the X proxy as uncompressed bitmaps. The X proxy performs the job of compressing the images and sending them on to the appropriate client. Figure 11 illustrates the interaction of client and server elements in VirtualGL direct mode.

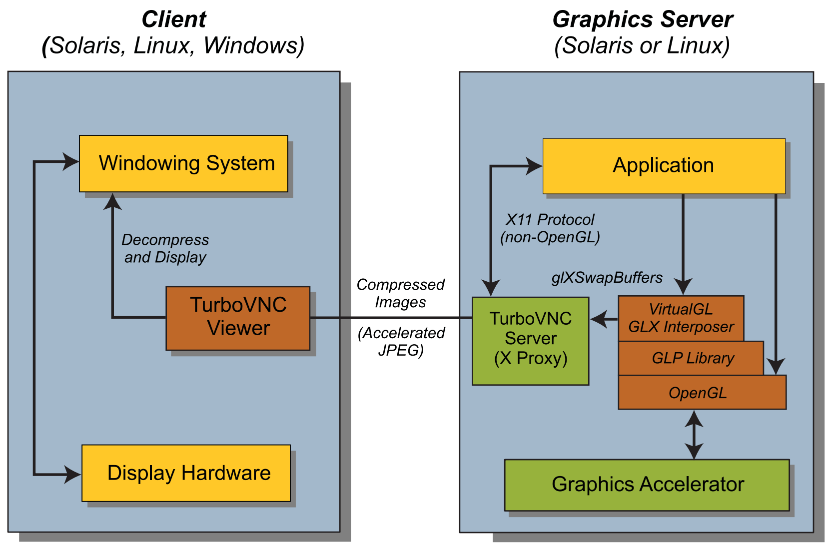

Figure 11. In VirtualGL Proxy Mode, rendered images are written as bitmaps to an "X proxy" such as TurboVNC, which delivers them to the client

Proxy Mode, in combination with TurboVNC, is the fastest solution for running VirtualGL on a wide area network (broadband, DSL, T1, etc.) In this fashion, server-based graphical applications can be viewed in geographically remote locations. Proxy Mode can be used to drive applications viewed in a Web browser environment through the Java™ technology based WebVNC viewer, or with the optimized and dedicated TurboVNC client viewer.

VirtualGL Sun Ray Mode — shared visualization to ultra thin Sun Ray clients

Many organizations use Sun Ray ultra thin clients, due to their centralized administration model and their security. With no storage devices, the stateless Sun Ray appliance provides no way for users to copy data onto or off of the server unless explicitly permitted to do so by a privileged administrator.

In Sun Ray Mode, VirtualGL compresses the rendered output images from 3D applications on the application server, and sends the resulting compressed images directly to the Sun Ray client device. Rather than rendering these images directly into an X proxy, VirtualGL instead compresses the images using the Sun Ray image codec and sends them directly to the Sun Ray hardware client (Figure 12)

.

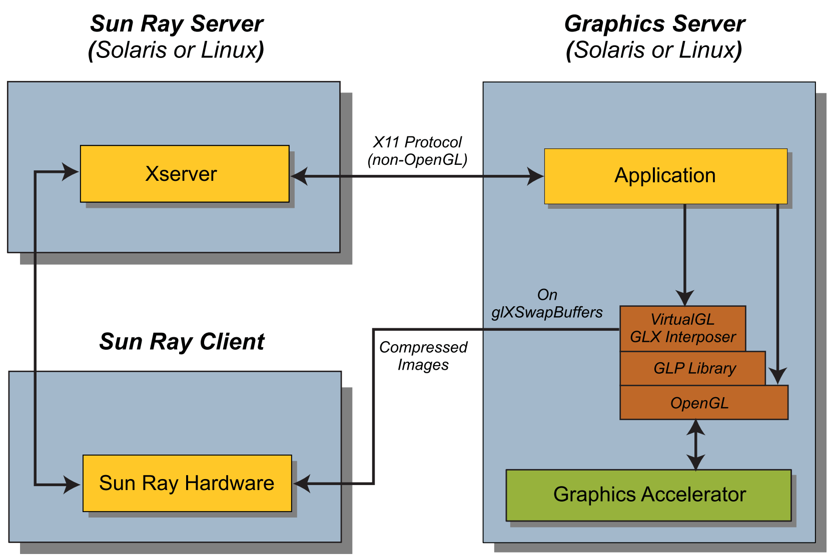

Figure 12. VirtualGL Sun Ray Mode compresses rendered images using the Sun Ray codec and sends the results directly to the Sun Ray ultra thin appliance

Running VirtualGL on Sun Ray ultra thin clients is a flexible approach, but there are several issues that should be considered. In Sun Ray environments, VirtualGL is typically not running on the same system as the Sun Ray server. For performance, it is beneficial to send images directly to the Sun Ray client rather than sending uncompressed images to the Sun Ray X server, as VirtualGL can generate considerable network traffic to sustain an acceptable frame rate.

At the same time, some Sun Ray networks are private, meaning that a remote 3D application server would not be able to contact the Sun Ray clients directly. In this situation, VirtualGL Raw Mode can be used to send images to the X server running on the Sun Ray server. However, though this configuration may be useful for light use, it is not usually recommended because of the network load that can be placed on the Sun Ray server and the associated Sun Ray network.

It is also possible to host the Sun Ray server and the 3D application server on the same system by installing graphics resources in the same system that runs the Sun Ray server software. This architecture works best when it is practical to perform graphics resource allocation at the time of Sun Ray session login, and 3D usage as a function of the number of users is fairly predictable.

For more information on operating Sun Shared Visualization software, please consult the S*un Shared Visualization 1.0 Software Client Administration Guide* and the *Sun Shared Visualization 1.0 Software Server Administration Guide*, available on *www.docs.sun.com*.

Chapter 4
Managing the Sun Visualization System

With key advances in graphics technology, visualization is now moving quickly into the datacenter where it can be effectively utilized, managed, and secured. Though this transition brings significant benefits in terms of scalability, collaboration, and resource utilization for the organization, it also poses new issues that affect the design and deployment of resources as well as their allocation to myriad users throughout the organization. Sun's experience with datacenter and high performance computing experience is ideal for helping organizations design, configure, and deploy effective visualization solutions.

Grid management of visualization resources

The integration of Sun Scalable Visualization software and Sun Shared Visualization software allows organizations to serve the diverse visualization needs of multiple users and groups. Grid management of visualization resources is key to both scalable and shared visualization infrastructure, offering fine-grained coordination and allocation of resources while maximizing the utilization of both graphics and computational investments.

In fact, scaling and sharing visualization resources poses many of the the same problems as general-purpose computing resources. However, unlike most grid computing jobs that require only a best-effort (batch) result, most visualization tasks are tied to user interaction of some sort. This key difference brings special requirements and expectations for performance and requires access to resources at particular times. For instance, users need to know:

- How to request the systems, applications, and storage that will fulfill their specific needs for graphics and computation without manual trial and error that takes them away from their task
- How to get unfettered, on-demand access to the resources they need without worrying about impacting others or being impacted by other applications
- How to reserve resources in advance for scheduled activities, or for systems that require a physical I/O or network storage component such as those that drive immersive CAVEs, theaters, or interactive powerwalls

To address these needs and to solve basic issues surrounding distributed visualization, the Sun Visualization System provides integration with Sun N1 Grid Engine software, extending key functionality to include graphics and visualization support as well as supplying advance reservation capabilities.

Sun N1 Grid Engine integration

Once a set of shared and scalable resources is configured and deployed, they can be easily allocated and managed through Sun N1 Grid Engine software. In particular, the Sun Shared Visualization software extends grid capabilities to allocate graphics resources in addition to, and in combination with computational resources. Based on user requests and criteria, Sun N1 Grid Engine software can select a suitable host for running an application and can allocate graphics resources as needed. A Sun N1 Grid Engine administrator can configure how many jobs can run simultaneously on a server or on an individual graphics accelerator. More information and in-depth configuration information can be found in the *Sun Shared Visualization Server Administrator Guide* and *Sun Shared Visualization Client Administrator Guide*.

Those using Sun Shared Visualization software have the choice of how they start their applications and the Shared Visualization server, including:
- Manual starting
- Sun N1 Grid Engine job scripts
- Sun N1 Grid Engine options from the command line

These options are listed in increasing order of complexity. Beyond simple experimentation, manual starting of the Shared Visualization server and self-selection of a graphics device in a shared environment is not generally advised. Within a shared environment, other users might be using, or about to use a certain device, causing contention. If a device is inadvertently selected that is in use by others, processes sharing the device could run out of resources. For example, memory on the graphics accelerator could become over-subscribed, causing processes to quit or become unreasonably slow. Sun N1 Grid Engine software is the recommended method for effectively sharing visualization resources. Figure 13 illustrates the integration of Sun Shared Visualization software with Sun N1 Grid Engine software.

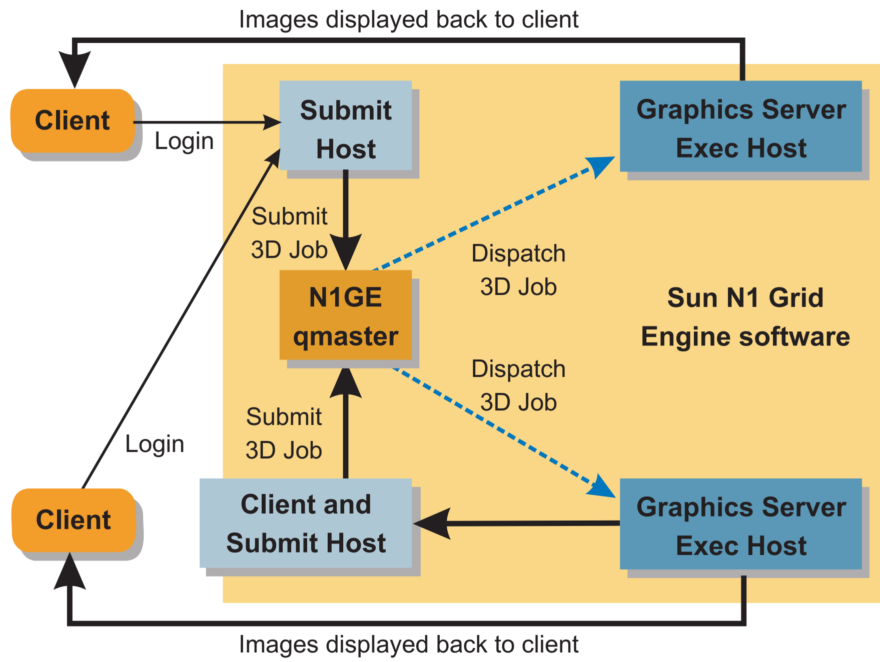

Figure 13. Sun Shared Visualization software integration into a Sun N1 Grid Engine software environment

Though details vary with the type of client being used, a typical set of steps for running an application using Sun Shared Visualization software in a Sun N1 Grid Engine environment would include:

1. Making sure that the graphics server has access to the X server (typically on the client, but also on the server in the case of TurboVNC)
2. Starting any necessary client viewer software such as the dedicated VirtualGL client or TurboVNC clients (Sun Ray Mode on a Sun Ray ultra thin client is automatic)
3. Running a script or submitting N1 Grid Engine commands such as `qsub` with the special VirtualGL `vglrun` command along with the name of a 3D application and its options, data sets, and runtime parameters

Once these steps are completed, Sun N1 Grid engine software will perform the following actions on the user's behalf:

- Select the appropriate graphics server based on criteria passed to Sun N1 Grid Engine software
- Select the appropriate graphics accelerator device
- Start the application on the graphics server under the control of VirtualGL

Controlling Graphics Resources

N1 Grid Engine administrators have complete fine-grained control over the resources on individual graphics servers. On any given graphics server under the control of Sun N1 Grid Engine, administrators can dictate which accelerators are available as resources, and can likewise determine how many jobs can share a given graphics accelerator. The number of concurrent jobs (computational and graphical) running on a server can also be controlled. For details, please see the *Sun Shared Visualization 1.0 Software Client and Server Administration Guides*.

As with typical Sun N1 Grid Engine operation, soft and hard limits for applications can be requested along with job submission, and jobs can be either queued (batch) or interactive. The heterogeneous nature of both Sun Shared Visualization software and Sun N1 Grid Engine software makes it easy to share and collaborate with different architectures of graphics server and clients.

Advance Reservation server for Sun N1 Grid Engine software

In addition to standard operation of VirtualGL within Sun N1 Grid Engine, Sun Shared Visualization software adds an Advance Reservation server that can specify resources at a given time and for a given duration. Once confirmed, resources are available to that user's Sun N1 Grid Engine jobs during that reservation period. Jobs intended to run during the reservation can be submitted right after the reservation is confirmed, or at any time before the end of the reserved period. Subsequent requests will be denied, until a new, approved reservation has been made. This access control feature is unique in the industry.

Implementing Advance Reservations outside of Sun N1 Grid Engine was accomplished by creating a dynamic Sun N1 Grid Engine queue to represent each confirmed reservation. As shown in Figure 14, this architecture consists of the following components:

- The Advance Reservation server helps ensure that applications consume only reserved resources. For each confirmed reservation, the server dynamically creates a Sun N1 Grid Engine queue which becomes active at the reservation time
- The Advance Reservation client is used to create, list, and delete reservations, communicating with the Advance Reservation server. The Advance Reservation client is available as a simple command-line program and as an easy-to-use graphical user interface (GUI).
- The reservation database serves to make reservations persistent
- Sun N1 Grid Engine software allocates resources to jobs, including jobs submitted against the reservation's queue

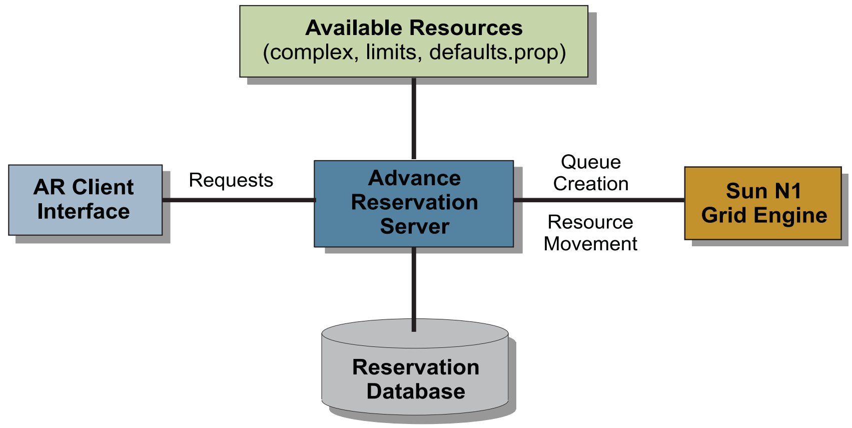

Figure 14. The Advance Reservation server creates a dynamic Sun N1 Grid Engine queue for each reservation

Putting the Sun Visualization System to work

Beyond system and graphics capabilities, Sun understands that designing and deploying effective scalable and shared visualization environments can be complex and time consuming. By integrating, tuning, and supporting the Sun Visualization System, Sun makes visualization easier and more rapid to deploy, letting organizations concentrate on their business or scientific endeavor.

Fully integrated and supported visualization systems from Sun

More than powerful systems and fast graphics, effective scalable and shared visualization solutions must focus on the whole environment with particular attention to needs of the critical applications that drive an organization's research or business objectives, and the needs of IT administrators. Because no two organizations are identical, Sun works with each firm or group to understand what they need to accomplish in order to reach success with their visualization deployment. Services provided can include:

- Systems engineering to design an appropriate visualization architecture
- Selecting an appropriate display integrator, speciality service provider, or supplier
- Selecting appropriate graphics servers and graphics accelerators
- Designing and configuring high-speed, low-latency interconnects
- Installing necessary software components, including Sun Visualization software and Sun N1 Grid Engine software

By providing a fully integrated and supported solution, Sun can help with the entire process.

Display Integrators and Sun's professional services group

Scalable visualization solutions and immersive visualization environments such as CAVEs and powerwalls often require specialized equipment including high-end projectors, screens, and synchronizing equipment, as well as motion tracking, force feedback, and new input devices. Sun works with leading display integrators to help ensure that visualization clusters are designed appropriately and deliver on their potential. Customers can also work with Sun's professional services group to design visualization solutions, if required.

Assembly and delivery through Sun Customer Ready Systems (CRS)

To help enable rapid and reliable delivery of HPC and visualization deployments backed up by Sun's support organization, the Sun Visualization System with Sun Scalable Visualization software is available only through the Sun Customer Ready Systems (CRS) program. This program offers factory-integration of Sun and complementary third-party hardware and software products. Sun CRS provides fully supported and integrated HPC and visualization systems, built in Sun factories and based on individual customer specifications.

Helping to take the time and complexity out of deployment, all Sun CRS systems are pre-installed, pre-tested, pre-configured, and interoperability tested in Sun's ISO 9002-certified factories. Customers avoid on-site assembly and integration problems and reap the rewards of faster deployment. With Sun CRS, 90- to 95-percent reductions in deployment times have been observed, coupled with up to 80-percent reductions in early-life system issues.

All Sun hardware and software products can be integrated by Sun CRS, including Sun Scalable Visualization systems. Organizations can work with display integrators or Sun's professional services group to design visualization systems that are ready-to-run upon delivery. Sun CRS will also incorporate selected third-party hardware and software as well as customer-specific applications and customized operating environment software.

Chapter 5
Conclusion

Solving today's important and urgent problems requires effective visualization technology and a comprehensive approach to HPC and visualization infrastructure. Scaling visualization infrastructure is often not enough if the results and resources cannot be effectively shared. Collaboration is not effective unless everyone has access. The most impressive resources are wasted if they are difficult to use and accessible to only a few.

Based on a wealth of leading systems, accelerators, and open source technology, the Sun Visualization System is dedicated to helping organizations see the information behind their data according to their own unique needs. Available through Sun CRS, Sun Scalable Visualization software lets organizations run the applications and APIs they need while deploying fully-integrated and supported visualization clusters. Sun's high-performance x64 systems, industry-leading graphics solutions from NVIDIA, and high-speed, low-latency InfiniBand interconnects help ensure that OpenGL based applications run to their best advantage.

Sun Shared Visualization software makes clustered and other server-based graphical and visualization resources available to others. This innovation not only lets organizations share graphics accelerators and systems, but aids collaboration across a broad range of clients, even as it drastically improves utilization. Just as grid computing helps many industries rapidly locate and utilize computational resources and launch applications, integration of the Sun Visualization System with Sun N1 Grid Engine software provides the same benefits to users of visualization technology.

Organizations and individuals can get started now by downloading Sun Scaled Visualization software at *www.sun.com/visualization*, or by contacting a Sun representative for information on designing a custom system for Sun Scalable Visualization software that is designed to fit their specific needs. Through the Sun Visualization System, Sun is committed to help organizations gain better insights, make better and more informed design decisions, and achieve the critical discoveries and breakthroughs that help improve the human condition — and the bottom line.

References

Sun Microsystems posts product information in the form of data sheets, specifications, and white papers on the internet at www.sun.com. Please look for abstracts for these and other Sun documents.

Sun Shared Visualization 1.0 Software Server Administrator Guide, January 2007

Sun Shared Visualization 1.0 Software Client Administration Guide, January 2007

Glossary

Anti-Aliasing
A method for getting rid of the “jaggies” that appear on screens when drawing diagonal lines or edges. Different methods are implemented in hardware to smooth out the jagged effects.

Bandwidth
The amount of data that can be transmitted over between two points.

CAVE (Computer-Augmented Visualization Environment)
A type of immersive reality environment that surrounds the user in the data set. Also referred to as an “immersive portal”, a CAVE is an environment where graphics can be applied in a room-like environment, on three or more surfaces.

Culling
Techniques that reduce computational and graphics processing requirements for 3D applications. *View frustum culling* renders only those objects within the current field of view. *Execution culling* invokes only the behaviors of relevant objects.

Chromium
Chromium (*http://chromium.sourceforge.net*) is a system for interactive rendering on clusters of graphics workstations and/or servers. Various parallel rendering techniques can be implemented using Chromium, including sort-first, sort-last, and hybrid approaches. A tuned and enhanced Chromium implementation is included with Sun Scalable Visualization software.

Codec
A device or program capable of performing encoding and decoding on a digital stream or signal.

Directed Acyclic Graph
A directed acyclic graph (DAG) is a directed graph with no path that starts and ends at the same vertex.

Distributed Multihead X (DMX) server
A distributed X11 server that allows tiled displays such as powerwalls to be built from multiple systems, each potentially with multiple displays. Applications connect to a single front-end server just as they would to a regular X server and a set of distributed back-end servers provide all of the visible rendering.

Fly-throughs
A capability of some immersive applications that allow a user to “fly” or “walk” through a data set. This capability enables users to visualize the interior of buildings, refineries, new products (including automobiles), and even human anatomy

Gigabyte, gigapixel
1 gigabyte = 1024 megabytes. 1 gigapixel = 1024 megapixels (typically 24-bit RGB with 8 bits per pixel)

GLX
OpenGL Extension to the X Window System

Haptic devices
Input peripherals that engage a user’s sense of touch.

HPC (High Performance Computing)
The use of systems to calculate complex results, typically those that need more compute power than is available on a user's desktop, notebook, or a single rack-mount server

Immersive reality
Visualization applications that create an environment in which users can "immerse" themselves in their data set. A CAVE is an example of an immersive reality environment.

Infiniband (IB)
A high speed interconnect between systems featuring low-latency that applications need to communicate with large amounts of data.

MCAD
Mechanical Computer Aided Design.

MCAE
Mechanical Computer Aided Engineering. Examples include Finite Element Analysis (FEA), and Computational Fluid Dynamics (CFD).

Mullion compensation
Mullion compensation allows shifting and expanding images across screens to compensate for the mullions present on flat panel displays or walls. The result is proper, optimized alignment from screen-to-screen to provide a contiguous image. Sun Scalable Visualization software provides scripts to simplify mullion compensation.

NVIDIA
A provider of computer graphics boards and chipsets. Provides Sun with the NVIDIA Quadro FX graphics cards and NVIDIA Quadro Plex Visual Computing System (VCS).

OpenGL
OpenGL (*http://www.opengl.org*) is an Application Programming Interface originally developed by Silicon Graphics, Inc. (SGI) that standardizes the programming for interactive 3D graphics.

OpenGL Extension to the X Window System (GLX)
X window system extension that provides integration between OpenGL and the X Window System, helping to enable OpenGL to operate within a window provided by the X window system.

OpenSceneGraph
OpenSceneGraph (*www.openscenegraph.org*) is an open source high performance 3D graphics toolkit that uses a scene graph rendering model. OpenSceneGraph is provided with Sun Scalable Visualization software.

Overlapping Support
Overlapping support helps integrators combine multiple projectors into video and data walls and other applications. Along with a projector's light intensity adjustments, overlapping and blending the images from individual projectors helps remove visible lines between displays. Sun Scalable Visualization software includes scripts to simplify overlapping configuration.

ParaView
Paraview (*http://www.paraview.org*) open source visualization application that runs on distributed and shared memory parallel as well as single-processor systems. ParaView is provided with Sun Scalable Visualization software.

Pixel
The smallest element on a computer graphics screen (Picture Element)

PCI-Express
A standard system interconnect expansion slot (bus and connector) within a computer system that provides both high bandwidth and full-duplex point-to-point architecture. PCI-Express bandwidth is defined in terms of "lanes", with each lane capable of transferring 250 MB/s. With a maximum of 32 lanes, PCI-Express allows for a total combined transfer rate of 8 GB/s.

Power wall
A collaborative visualization environment that enables joint design reviews. Power walls are typically composed of multiple frame buffers that drive multiple projectors and large-screen displays, and present a single logical image.

Resolution
The number of pixels that are used on the graphics board, or the number of pixels that the display device can show. Typically a PC or Workstation has a resolution of 1600 x 1200.

Scalable Link Interface (SLI)
A technology from NVIDIA which allows 2 or more graphics boards to share the load of graphics processing within a single workstation.

Scene graph
The type of rendering model used by OpenSceneGraph and Java 3D™ software. A scene graph is a hierarchical definition whereby components can be associated by their spacial and behavioral relationships. For example, an arm is composed of an upper arm, a lower arm, and a hand. A hand is composed of a palm and fingers, and so on. In a scene graph, moving the arm implies moving all of the subcomponents in a consistent fashion.

Secure Shell (SSH)
The Secure Shell is a set of standards and an associated network protocols that facilitate a secure channel between a local and a remote computer using public-key cryptography. SSH tunneling, or port forwarding, is a way to forward otherwise insecure TCP traffic through SSH.

Secure Sockets Layer (SSL)
A protocol developed by Netscape for transmitting private documents via the Internet. SSL uses a cryptographic system that uses two keys to encrypt data — a public key and a private key.

SLI
See Scalable Link Interface.

SSH
See Secure shell.

Stereoscopic (stereo) viewing
A capability that endows 3D imagery with the perception of depth. Stereo viewing usually requires specialized hardware, such as a stereo-capable frame buffer, display, and peripherals (e.g., stereo glasses), along with applications that can generate stereo images.

Sun Customer Ready Systems (CRS)
A service from Sun that can configure or build systems to the customer specifications. Sun Scalable Visualization systems are only available through Sun CRS.

Supersampling
A means of providing higher quality 2D and 3D rendering. Supersampling at a fine-grained resolution works by first rendering it at a large sample resolution and then sub-sampling it.

SSL
See Secure Sockets Layer

Terabytes
Trillions of bytes of data (1 terabyte = 1024 GB)

Texturing/texture mapping
A method employing an array of pixels that will be decal mapped or physically modeled onto a 3D surface to give more realistic physical appearance. Texture mapping is now commonly used to more accurately model materials and to detect discontinuities in surface models.

Three-dimensional (3D) modeling
A general class of visualization applications that use 3D geometry and surface modeling to represent three-dimensional data.

Virtual Network Client (VNC)
The Virtual Network Client (VNC) system allows one or more remote clients to access a "virtual" desktop in the main memory of a server host.

TurboVNC
TurboVNC is an enhanced VNC distribution. When used with VirtualGL, TurboVNC is the fastest solution for remotely displaying 3D applications across a wide area network.

VirtualGL
An open source package that provides hardware-accelerated rendering capabilities to thin clients. Information on VirtualGL can be found at *http://www.virtualgl.org*.

Visual simulation
A class of visualization applications that model aspects of behavior with an assumption of 60fps frame rates.

Visualization
A general term used to describe how applications display large, multidimensional data sets in a graphical way in order to facilitate better data comprehension.

Volume modeling
A visualization technique that treats 3D data as a solid volume, using voxels instead of pixels. Volume modeling applications often allow the user to "slice" through the volume.

Voxel
A volume element, the 3D version of the pixel.

Xinerama
A feature supported in the Solaris OS that provides a single logical screen driven by multiple homogenous frame buffers.

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA **Phone** 1-650-960-1300 or 1-800-555-9SUN (9786) **Web** sun.com

Printed in USA 02/07