

EnSight's Parallel Processing Changes the Performance Equation

by Kent Misegades, CEI president

Parallel processing for scientific computation is certainly not a new concept. For decades, performance research has focused on reducing the time it takes to execute floating-point and other operations related to solving numerically intensive algorithms used in such fields as structural mechanics and fluid dynamics. Over the last few years, CEI has aggressively expanded that research into the realm of high-end visualization and VR display.

CEI is no stranger to the high-performance computing (HPC) realm. The company evolved from the visualization activities of the late 1980s and early 1990s at the well-known supercomputer manufacturer Cray Research. Through the years, CEI's work has been typified by complex simulations requiring the most advanced computer architectures.

At the forefront of recent HPC research is the U.S. Department of Energy's ASCI (Accelerated Strategic Computing Initiative) project. CEI has played a key role in this effort, providing EnSight Gold software to visualize models containing more than 1 billion cells and hundreds of time steps. Processing these models has required research in a number of different areas, the most important one being parallel processing. Other aspects involve efficient data formats, improved means of displaying millions of polygons interactively, and support for virtual reality.

Three distinct areas of parallel processing are supported in EnSight:

- server-side functions -- numerically intensive operations in the EnSight server process
- client-side functions -- numerically intensive operations in the EnSight client process
- object rendering -- dynamic display of an object

This article describes the opportunities for parallel processing that CEI has discovered, and the performance improvements that have been gained using conventional and emerging computer architectures. With the exception of parallel rendering, all parallel-processing functions described in this article are available in both EnSight and EnSight Gold. EnSight users are limited to two processors; for EnSight Gold there are no such limits.

Server-Side Parallel Processing -- An Overview

Early attempts to visualize ASCI-class datasets quickly revealed compute-intensive bottlenecks in the EnSight server process, which is responsible for most data I/O and numerical operations. These areas include:

- clip plane and isosurface extraction
- particle trace calculation
- bounding box determination
- histogram computation
- I/O

Parallel processing and Amdahl's Law -- Performance improvements through parallel processing generally follow Amdahl's Law, attributed to Gene Amdahl, one of the original architects of the IBM System/360 computer line and founder of the Amdahl Corp.:

$S_{eff} = S_f / (S_f (1-f) + f)$, where

f is the fraction of the program that can be improved

S_{eff} is the overall speedup factor

S_f is the speedup factor on this fraction, ideally equal to the number of processors used

Figure 1 depicts the overall speedup factor using eight processors, assuming an ideal **S_f**. It is clear from this diagram that a high level of parallelization must be reached to realize any significant gains from parallel processing. Even at 90-percent parallel the speedup factor is only slightly better than half the optimum of eight.

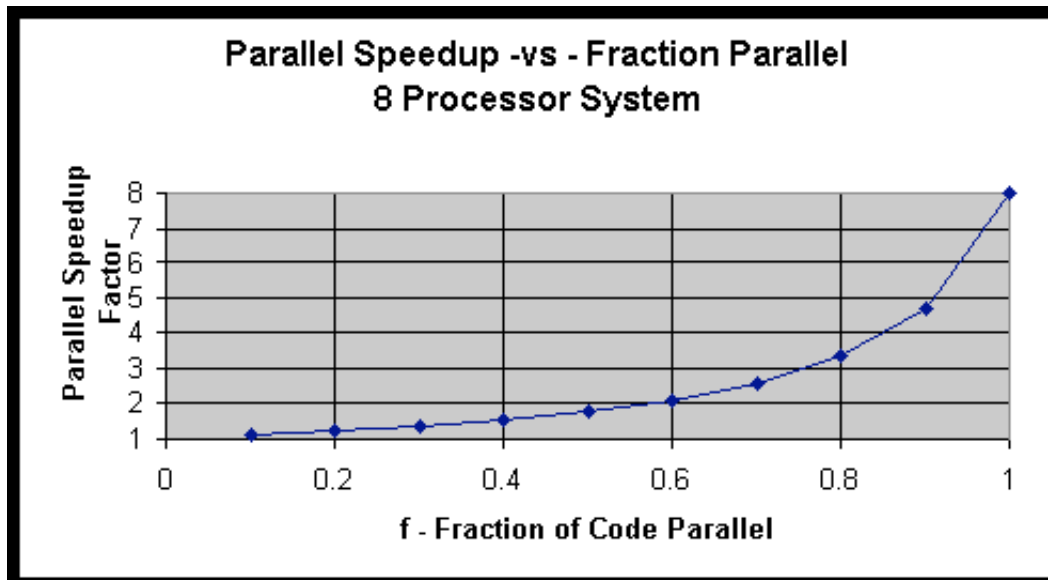


Figure 1 -- Overall parallel speedup factor versus fraction of code parallelized.

Parallel simulation versus parallel visualization -- In contrast to modern simulation software in the fields of FEA and CFD, in which nearly 100 percent of time-consuming computations can be run in parallel, visualization techniques have until the recent past remained mostly serial operations. It is only when models become large that parallelizing functions such as those listed above become attractive. Even for today's most demanding industrial-class problems (20-50 million cells), the functions investigated here do not exceed parallelization of more than 90 percent, implying from Amdahl's Law that speedup factors greater than six for an eight-processor system are unlikely.

SMP versus DMP -- In the discussion above, it is assumed that parallel processing is being applied in the classical sense, i.e. a single operation is executed in parallel over the entire model. In computing a number of particle paths through a given flow field with each path independent of the other, for example, the total effort is divided equally among the available processors. If, as is the case for most desktop parallel-computing systems today, the entire memory is shared among the processors, it is called a shared-memory parallel architecture, or SMP. SMP systems remain the most prevalent today, even though it has been shown that parallel speedups for industrial applications plateau after eight to 16 processors.

To overcome the restrictions of SMP systems, researchers break up data sets into a number of blocks. Each block is processed independently, then combined in a compositing step prior to display. This is known as distributed-memory parallel processing or DMP.

DMP-based systems come in many varieties. The most popular are tightly integrated Windows or Linux clusters (also known as Beowulf clusters), each node of which might be an SMP system containing a number of processors sharing a common memory. If each block of a decomposed model is the same size, resulting in a good load balance on a given system, the highest possible speedup can be expected. This fact has made DMP systems the architecture of choice for those with the greatest need for computing power.

CEI provides parallel processing capabilities for both SMP and DMP systems. For DMP systems with SMP multiprocessor nodes, the SMP and DMP features can be combined, resulting in a two-tiered approach to

parallelization. Let's look at results of recent benchmarks of EnSight server-side computations on common SMP architectures and modern DMP systems.

Server-Side Parallel Processing - SMP Systems

While DMP systems promise the greatest potential for acceleration, there are at least three reasons why parallelization for SMP systems remains important:

1. The most common type of computers in use today remain SMP systems running under Unix or Windows and containing from one to eight processors and several gigabytes of shared memory.
2. Most computations today are performed using commercial software. While vendors of popular packages such as LS-Dyna, MSC.Nastran, Fluent and STAR-CD offer versions of their products for DMP systems, results are invariably provided as monolithic datasets that are no longer decomposed.
3. No standard format for decomposed data exists today, impeding the transfer of data from DMP-parallel solvers to post-processors.

Based on these obstacles, the visualization of simulation results on SMP systems will remain standard industrial practice for the foreseeable future.

A number of performance tests were recently performed at CEI's offices to measure EnSight Gold's parallel speedup on three common SMP systems:

- Compaq ES40 6/667, with four processors and 1 GB of memory, running Tru64 v5.1
- SGI Onyx 2000, with eight R10000 processors, 10,240 GB of memory, running Irix 6.5
- Sun E450, with four 480-MHz processors and 3 GB of memory, running Solaris 8

Results of airflow predictions around the Lockheed SR-71 aircraft from a modern CFD package, Cobalt (www.cobaltcf.com), were used for the tests. The mesh consisted of 6,989,123 tetrahedral cells and 1,215,026 nodes. EnSight Gold version 7.3.2 was used in all cases.

For each of the following four functions, compute time was measured, but not the time it took to render (display) the result. In EnSight, the same algorithm used to compute clip planes is used to extract isosurfaces; therefore only the clip-plane function was measured.

- Clip-plane extraction -- time to compute a clip plane at $Z=0$, resulting in 201,000 new polygons
- Particle-trace calculation -- time to compute a rake of 9,999 particle traces through the model
- Bounding-box determination -- time to find the external boundary of the fluid region
- Histogram computation -- time to create a histogram of the pressure field

While it's relatively easy to recognize the importance of computing clips, isosurfaces and particles in parallel, it might not be as obvious why it is important to benchmark performance for the other two functions. The bounding box computation occurs frequently in the course of visualization, although this is normally transparent to the user. Histograms are computed whenever the color map editor is used for a given variable. Although both functions might account for only a small percentage of the overall computation effort, they become significant when visualizing transient data, where a given visual might be repeated for each of hundreds or thousands of time steps.

Figures 2 through 5 depict the parallel speedup measured for each of the four functions running EnSight on the three SMP systems.

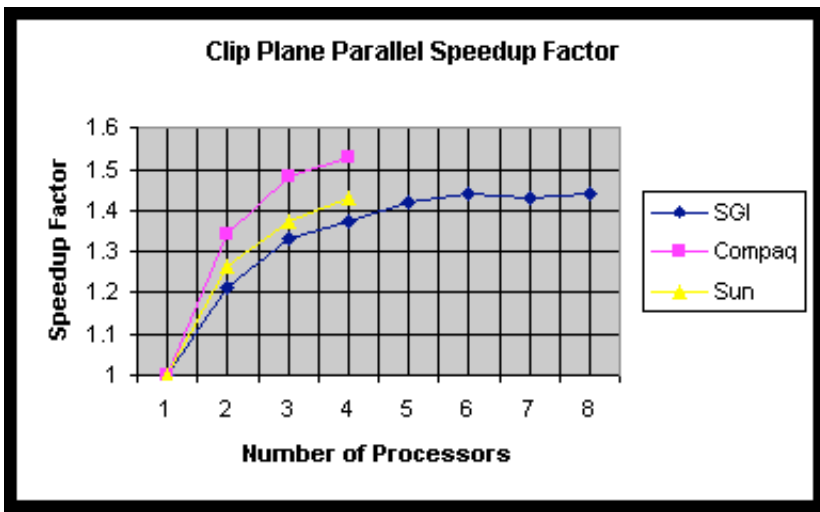


Figure 2 -- Server-side parallel speedup on SMP systems, clip-plane extraction

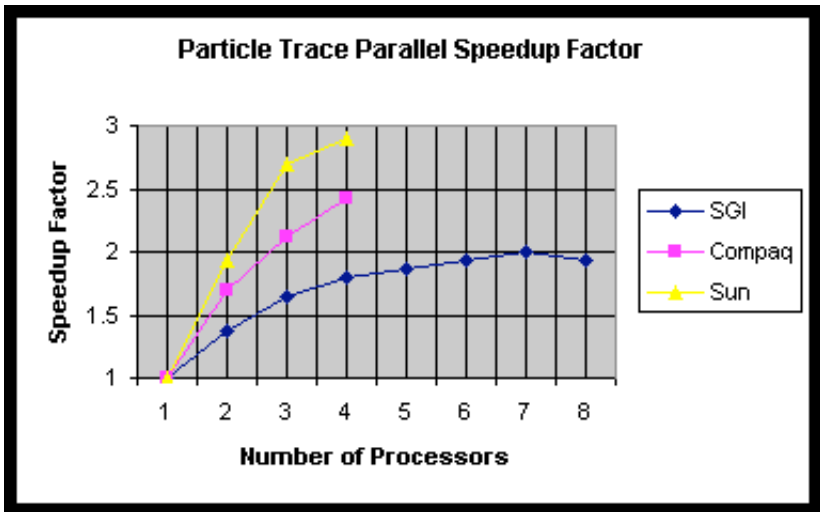


Figure 3 -- Server-side parallel speedup on SMP systems, particle traces

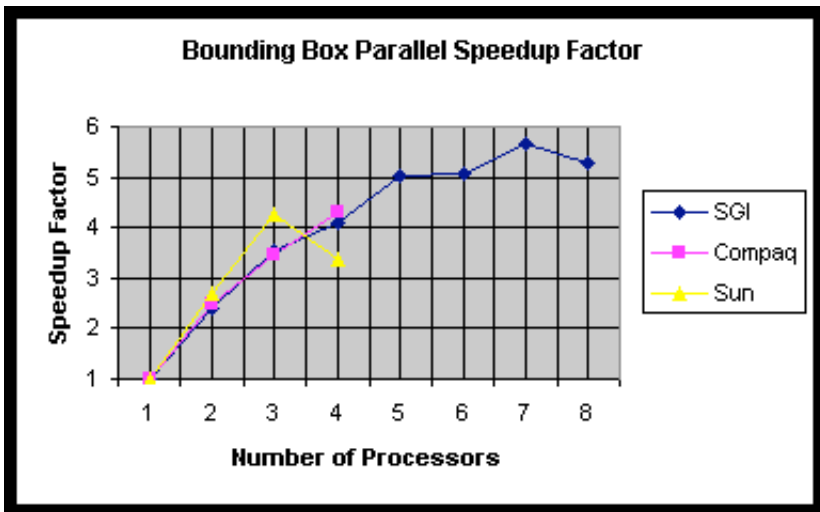


Figure 4 -- Server-side parallel speedup on SMP systems, bounding box

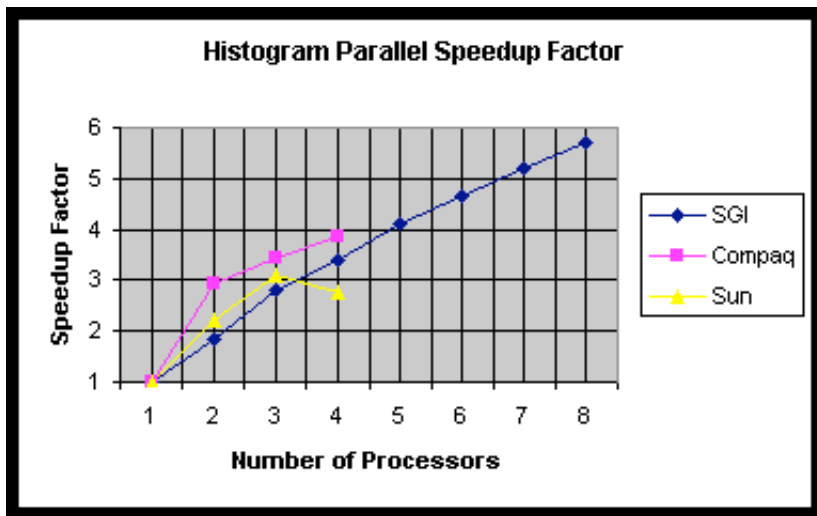


Figure 5 -- Server-side parallel speedup on SMP systems, histogram

One note regarding times for parallel particle tracing: It was not possible to separate the time to compute the traces from the rendering, thus the results shown in figure 3 only partially reflect the effect of parallel processing. Further tests were carried out on CEI's eight-processor Onyx2 using a development version of EnSight. In these tests, speedup factors of 1.8x, 3.3x, and 5x were measured for only the particle trace calculation when using two, four and eight processors, respectively.

Parallel I/O -- EnSight Gold takes advantage of special features of SGI Onyx systems running under Irix 6.5 that permit parallel I/O. When loading large EnSight Gold format datasets, N processors, or threads, are used to read the data in parallel. Each thread reads 1/Nth of the data into the proper location of the data structure. Currently, N is set to four, which after extensive testing has been found to be an optimal number of threads.

Many versions of the Unix operating system support automatic caching files if enough memory is available. This can be beneficial for successive reads of these files, since the system will load the data from cache memory instead of directly from disk. Since EnSight doesn't typically reread a file, this automatic caching tends to have little benefit. EnSight Gold takes advantage of this caching, however, by using a background thread to pre-read the next step in a time series. Specifically, if the user is processing time step N, then a background thread will pre-read step n+1. If the computer has enough free memory, then the file will reside in cache by the time the user advances the clock to step N+1.

While it is difficult to predict the performance improvement one will gain through these parallel I/O features when using EnSight Gold on SGI Onyx systems, reductions in I/O times from 15 to 70 percent have been measured. The most significant gains will occur when loading many large time steps, such as one might do when creating flipbook animations.

User Considerations -- With the release EnSight 7.3, the SMP parallel functionality described above is supported on all SMP hardware platforms with the exception of PCs under the Windows operating system. By default, only one processor will be used. To use more, the environment variable ENSIGHT7_MAX_THREADS is set to the maximum number of processors desired.

Thus, users should modify their .cshrc file to include the following line:

```
setenv ENSIGHT7_MAX_THREADS 8
```

or the .profile file for Bourne shell users:

```
ENSIGHT7_MAX_THREADS =8
export ENSIGHT7_MAX_THREADS
```

in order to use a maximum of eight processors.

Server-Side Parallel Processing -- DMP Systems

Parallel processing on distributed-memory parallel systems is made possible through EnSight's client-server architecture. In contrast to SMP systems, where a single client process is linked to a single server process using multiple processors, on DMP systems a new, server-of-server (SOS) process is introduced. The SOS acts as a router of information from the EnSight client to a number of EnSight servers, one per computational node. As mentioned before, each computational node can itself be an SMP system.

Figure 6 compares EnSight's implementation on SMP and DMP systems. The SMP system has four CPUs. The DMP system consists of four computational nodes, each with four CPUs.

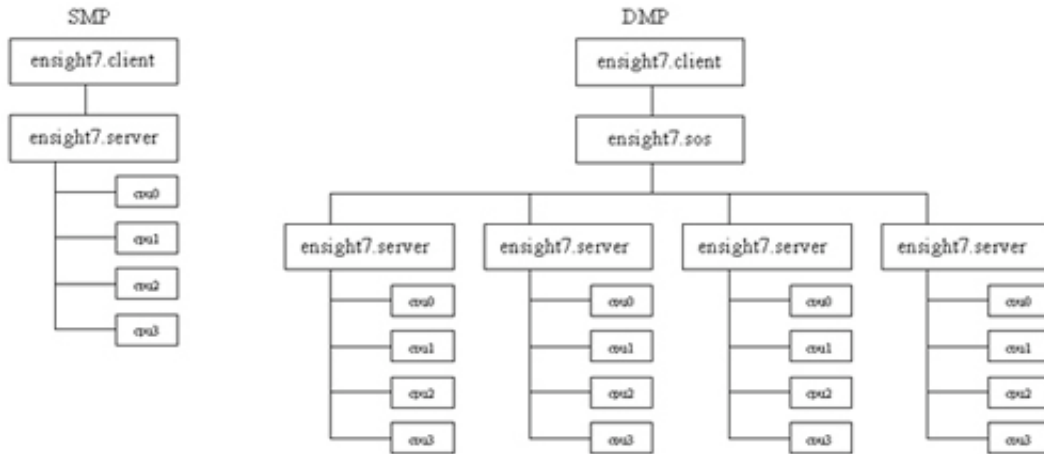


Figure 6 -- EnSight processes on SMP and DMP architectures

User Considerations -- To use the SOS, one must manually start a client process (*ensight7.client*) and an SOS process (*ensight7.sos*). Data must exist as a set of independent CASE filesets. A new type of CASE file has been developed to describe the location and name of the decomposed data. Currently, each CASE fileset must contain the same part structure and the same number of cells and nodes. A sample SOS CASE file for a model distributed across four compute nodes is shown below. Each individual case file -- *quad001.case*, *quad002.case*, etc. -- contains the usual information describing the model geometry, time information, and variables.

FORMAT

```
type: master_server gold
```

SERVERS

```
number of servers: 4
```

#Server 1

```
machine id: node0
```

```
executable: /home/users/ensight7/server/ensight7.server
```

```
casefile: /home/users/jesse/data/one/quad001.case
```

#Server 2

```
machine id: node1
```

```
executable: /home/users/ensight7/server/ensight7.server
```

```
casefile: /home/users/jesse/data/two/quad002.case
```

#Server 3

```
machine id: node2
```

```
executable: /home/users/ensight7/server/ensight7.server
```

```
casefile: /home/users/jesse/data/three/quad003.case
```

#Server 4
machine id: node3
executable: /home/users/ensight7/server/ensight7.server
casefile: /home/users/jesse/data/four/quad004.case

Limitations -- All server features with the exception of particle tracing and new variable calculation are supported by the SOS process. The two remaining features will be supported in a future version of EnSight.

ASCI Benchmark -- The parallel performance of the EnSight SOS process has been tested on some of the world's most powerful computing systems. One of these, the so-called ASCI Blue Mountain system at Los Alamos National Laboratory, consists of 48 SGI Origin 2000 servers, each containing 128 MIPS R1000 processors and 1 terabyte of memory.

A test model was created containing the data structure and variables representative of ASCI production simulations. Decomposed in 30 blocks of equal size and distributed on 30 of the 48 Origin servers, the model used an unstructured mesh containing 11.5 billion hexahedral cells. The benchmark measured the clock time to load the model, perform a plane clip and isosurface extraction through the entire domain, and render the resulting surfaces onto a display device. All 128 SMP processors on each of the 30 compute nodes were used, giving a total processor count for the test of 3,840. The rendered surfaces contained approximately 96 million polygons.

Using EnSight's SOS features, the entire operation was completed in approximately 15 minutes. Tests of smaller models on the same system have shown excellent scaling of SOS parallel performance with increasing model size.

Client-Side Parallel Processing

Recent research at CEI has focused on parallel processing in the client process. Multiple threads are used each time a new part comes from the server or when a part representation changes (line to shaded, shaded to colored, etc). They are also used to generate the "displayable object" from the internal data structures, including normal generation, scaling values to the color map, and other operations.

A benchmark has been performed for an isosurface part consisting of 11 million triangles. The time required by the client to display the isosurface as a colored solid object is shown in figure 7. Additional tests have been performed on SGI systems using 32 and 64 processors, achieving speedup factors of 27x and 45x, respectively. Results are shown in figure 8.

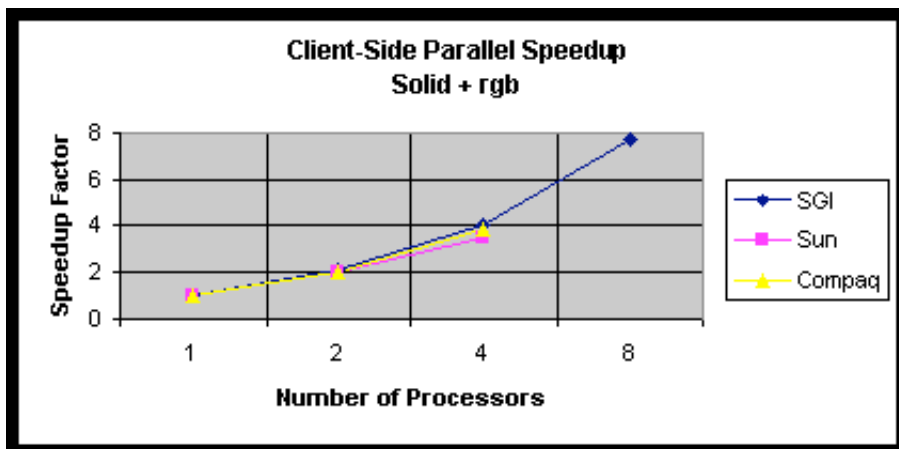


Figure 7 -- Client-side parallel speedup, three different hardware systems

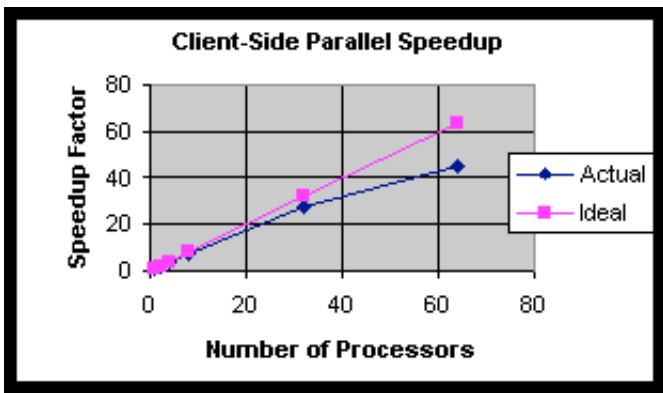


Figure 8 -- Client-side parallel speedup, actual versus ideal

Parallel Rendering

Just as parallel CPUs accelerate numerical computations, multiple graphical processors accelerate the rendering of a given scene. The user senses this through quicker model transformations (rotate, translate, zoom). Until recently, only the multi-pipe Onyx systems from SGI provided the multiple graphical processors necessary for parallel rendering. CEI, working closely with SGI, has optimized EnSight Gold for operation on Onyx systems containing more than one graphical pipeline (so-called "monster" mode on SGI systems). This feature can be used in conjunction with stereo and multi-frustum viewing. If a system contains four pipes and a two-panel display system is used, each display would make use of two pipes. EnSight Gold users have complete control of Onyx pipeline allocation.

Parallel rendering performance has been tested on an SGI Onyx2 system with eight Infinite Reality graphical pipelines. The test measured the time to render a 22-frame animation of an isosurface containing 11 million triangles. Figure 9 displays the results.

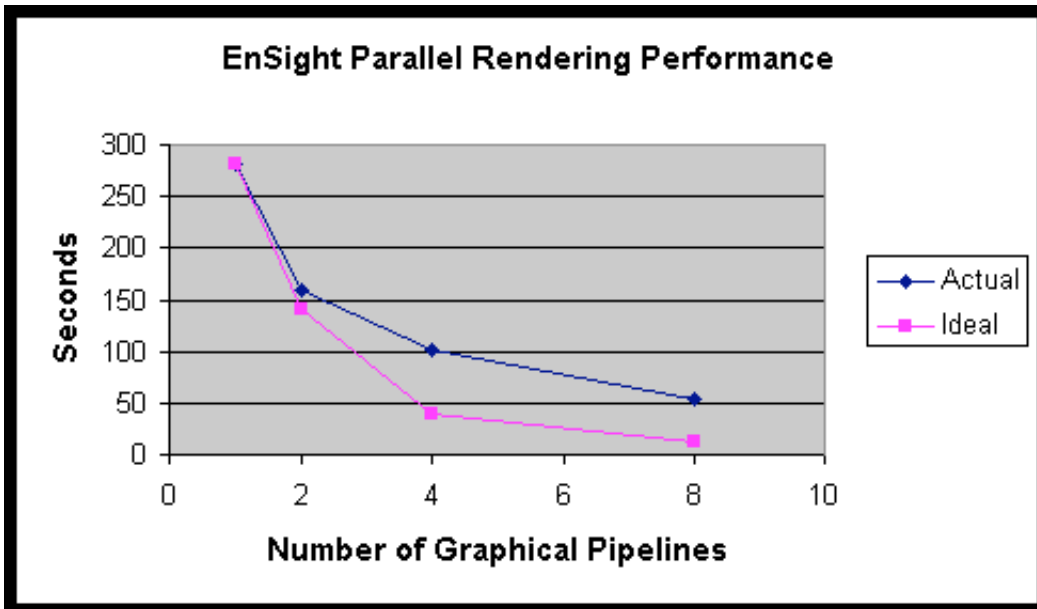


Figure 9 -- Parallel rendering performance gains

Recently, other hardware suppliers such as Sun Microsystems have released products with multiple graphical pipes. EnSight Gold also takes advantage of these systems. HP, SGI and Sun all have "transparent" multi-pipe systems that require no additional optimization to realize parallel rendering. Performance gains for such transparent solutions are less than for optimized implementations, however.

Parallel rendering performance improvements will not, in general, be realized when using multiple graphics pipelines for multi-frustum applications such as CAVEs. For details on configuring EnSight Gold for use on

multi-pipe systems, contact CEI or its distributors.

Parallel Processing Research at CEI

As this paper has shown, very significant performance gains have been achieved by CEI through the use of parallel processing in its many forms. Nevertheless, the use of multiple processors, compute nodes, networks, clusters, graphical pipelines and other parallel processing methods remains an area of groundbreaking research when applied to visualization. CEI is currently investigating a number of related topics including:

- further improvements to SMP parallel processing
- further improvements to DMP parallel processing
- Wire GL (Chromium) -- transparent parallel rendering for OpenGL applications
- parallel rendering on DMP systems with commodity graphics processors
- collaboration

For further details, contact CEI or one of its authorized distributors.

copyright © 2002 CEI. All rights reserved.