DL-TR-2007-001

Meaningful Benchmarks for Scientific Visualization

Richard Wain, Barry Searle, Miles Deegan, Martyn Guest, Christine Kitchen *and* **Igor Kozin**

February 2007

Council for the Central Laboratory of the Research Councils

Enquiries about copyright, reproduction and requests for additional copies of this report should be addressed to:

Library and Information Services
CCLRC Daresbury Laboratory
Daresbury Warrington
Cheshire WA4 4AD
UK
Tel: +44 (0)1925 603397
Fax: +44 (0)1925 603779
Email: library@dl.ac.uk

ISSN 1362-0207

Meaningful Benchmarks for Scientific Visualization

Richard Wain, Barry Searle, Miles Deegan, Martyn Guest, Christine Kitchen, Igor Kozin
Computational Science and Engineering Department
CCLRC Daresbury Laboratory
Warrington, WA4 4AD, UK

Abstract

This report presents results from benchmarking activities undertaken on an SGI Prism visualization system. The benchmarks used include the industry standard OpenGL rendering performance benchmark SPECViewPerf [1], *an in-house benchmark based on the DL Visualize software* [2] *and mpi-tile-io* [3], *an IO benchmark designed to simulate a typical visualization workload* [4].

1. Introduction

There is a general lack of well-defined scientific visualization benchmarks and therefore a lack of meaningful comparative data for high-end visualization systems. Many of the available benchmarks for graphics performance are targeted at the games industry (3Dmark [5]) or at professional CAD/CAM users (SPECViewPerf [1]). Generally, these benchmarks measure the peak rendering performance of a graphics card (for example in a games machine or a desktop PC or workstation). When benchmarking scientific visualization systems it is useful to be able to measure how well applications scale to multiple graphics pipes, and benchmarks such as SPECViewPerf do not currently allow for this. There have been some efforts to produce visualization benchmarks for systems with multiple graphics pipes (e.g. VMD and VRNav2 benchmarks from UCLA [6]). These efforts have not produced the standard benchmarks or standard result sets that would allow for the straightforward comparison of high-end visualization systems (as SPECViewPerf has done for graphics workstations). Given the lack of standard scientific visualization benchmarks, we have used one synthetic benchmark, one standard application benchmark and one in-house application benchmark in order to produce the results in this report. To conclude we discuss how 'useful' these results are in determining the relative merits of an SGI Prism system (or any other high-end visualization system).

2. Benchmarks

2.1 SPECViewPerf

The SPECViewPerf benchmark is a portable OpenGL performance benchmark endorsed by the Standard Performance Evaluation Corporation's OpenGL Performance Characterization (SPECopc) project group [1]. The benchmark comes bundled with a number of viewsets that characterise the OpenGL rendering functionality of visualization applications from a number of ISV's. We have used version 8.1 of the benchmark, which incorporates viewsets for the following applications (Source http://www.spec.org/gpc/opc.static/whatis_vp8.html):

- 3dsmax-03, based on SPECapc for 3ds max 3.1 configured with the Open GL driver; includes three models containing an average of 1.5 million vertices each, and tests performance of scenes with different levels of lighting.
- catia-01, based on Dassault's CATIA, with models containing up to two million vertices.
- ensight-01, based on CEI's EnSight engineering and scientific visualization application, covers both display-list and immediate-mode workloads.
- light-07, based on traces of Discreet's Lightscape radiosity application.

- maya-01, based on traces of Alias' Maya 5.
- proe-03, based on SPECapc for Pro/ENGINEER 2001, measures two models in three modes -- shaded, wireframe and hidden-line removal (HLR).
- sw-01, based traces of the Solidworks 2004 application from Dassault Systemes.
- ugs-04, based on SPECapc for Unigraphics V17, tests performance based on an engine model containing 4.1 million vertices.

2.2 DLV benchmark

DLV is a graphical user interface for use with a variety of materials simulation software. It is able to display and edit structures periodic in both 2 (surfaces) and 3 (crystals) dimensions. The current version (v2.5) is based on AVS/Express v7.0 and provides an interface to CRYSTAL03 (and CRYSTAL98), DL_EXCURV and GULP [2]. DLV benchmark results are obtained by measuring the frame rate when rendering five benchmark datasets. The datasets used in the DLV benchmark are:

- 1x2x2 crambin (~5000 atoms)
- 1x2x2 crambin + bonds
- 3x2x3 rusticyanin (~96000 atoms)
- crambin isosurface
- 1x2x2 isosurface

2.3 mpi-tile-io

MPI-Tile-IO [3] is a tile reading MPI-IO application. The application simulates a type of workload that exists in some visualization applications, namely, tiled access to a two-dimensional dense dataset [4]. While this benchmark is not testing visualization performance it is still useful in determining whether IO is likely to be a bottleneck when running parallel visualization applications.

3. Benchmark Results

3.1 SPECViewPerf 8.1

SPECViewPerf is concerned entirely with single GPU OpenGL rendering performance and this does not play into the hands of the SGI Prism system that is currently based on ATI FireGL cards that are no longer at the cutting edge. The results shown below cover desktop PCs, graphics workstations and an SGI Prism. These results provide some evidence that the SGI's ATI graphics cards are not competitive with the graphics cards in a modern workstation.

The following SPECViewPerf results are unofficial. All official SPECViewPerf results can be viewed on the SPEC website, here http://www.spec.org/gpc/opc.static/vp81results.html.

	3dsmax-03	**catia-01**	**ensight-01**	**light-07**	**Maya-01**	**proe-03**	**sw-01**	**ugs-04**
1 - GeForce FX5200/256MB, AthlonXP-1900 1.6GHz 1GB	2.137	1.628	2.179	1.332	2.968	1.998	1.639	2.097
2 - Quadro 980XGL/128MB, AthlonXP-2700 2.13GHz 1.5GB	10.48	8.269	5.538	10.08	19.91	9.231	6.713	11.16
3 - GeForce 4 MX440SE/64MB, Pentium4 2.8GHz 1GB	5.772	5.44	3.37	6.336	9.907	6.386	2.723	2.257
4 - GeForce 6200 TurboCache/ 64MB, pentium4 640 1GB	6.505	5.387	4.911	5.099	7.779	7.156	7.523	2.424
5 - QuadroFX 3450/256MB, 2xOpteron-252 2.6GHz, 6GB	39.78	33.85	29.63	27.35	58.82	47.48	29.89	30.8

6 - GeForce 6600/128MB, Pentium4 640 1GB	14.68	11.18	8.605	10.61	14.2	16.03	11.5	4.206
7 - SGI Prism, ATI X3 128MB PCI-X, Itanium 1.5GHz	8.991	6.015	7.326	7.518	5.735	6.47	4.375	10.7
8 - Dell P390, ATI V7200 256MB, Core2 Duo E6600 2.4GHz, 2GB	46.63	35.68	36.41	35.3	-	57.9	40.8	39.88

Table 1, SPECViewPerf data

SPECViewPerf results are plotted in appendix 1. These results clearly show the relatively poor performance of the ATI FireGL X3 card used in the SGI Prism. This poor performance can in part be attributed to the age of the graphics hardware.

3.2 DLV benchmark

	Nvidia Geforce 6600	**Quadro FX3450**	**SGI/ATI FireGL X3**	**Ati V7200**
1x2x2 crambin (~5000 atoms)	26.5	89	14.2	39
1x2x2 crambin + bonds	21	61	10.7	38.5
3x2x3 rusticyanin (~96000 atoms)	1.3	5.4	0.7	2.6
crambin isosurface	17.2	39	27	46.6
1x2x2 isosurface	3.6	8.5	3.7	4.2

Table 2, DLV Benchmark data

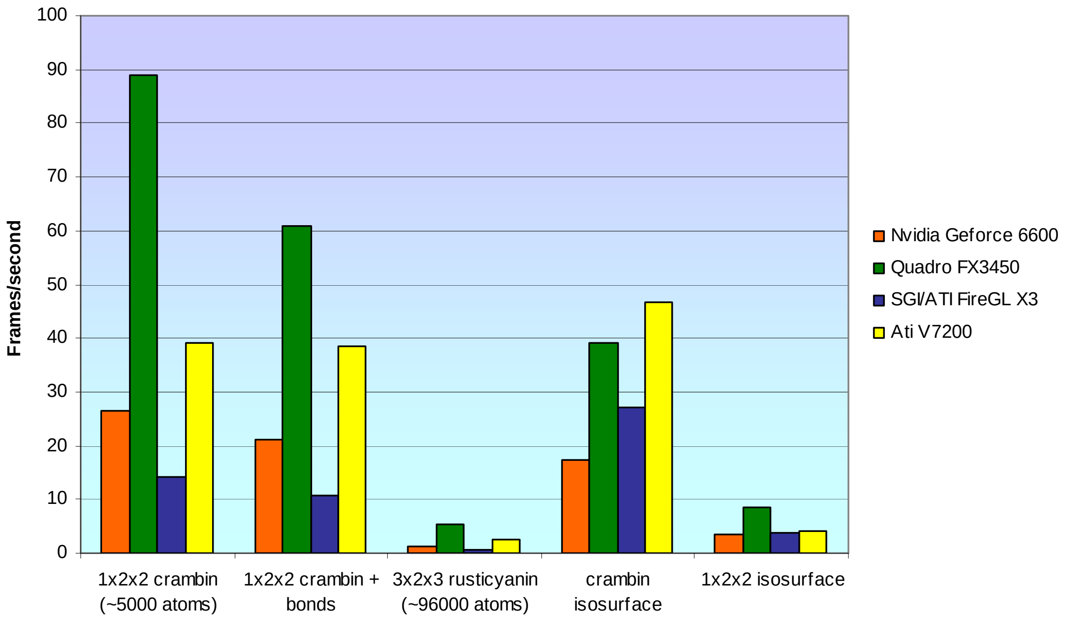


Figure 1, DLV Benchmark Performance

The results in table 2 and figure 1 again highlight the relatively poor rendering performance of a single pipe on the SGI Prism system when compared to a graphics workstation or recent desktop PC. While the SGI is competitive in both isosurface-rendering tests it is easily outscored in the other tests.

3.3 mpi-tile-io

n	M	cpus	Read (n x m) 512	Read (m x n) 512	Read (n x m) 1024	Read (m x n) 1024
1	**2**	**2**	11.187	529.436	5.67	534.048
2	**2**	**4**	1009.564	1004.82	1009.068	1014.605
3	**2**	**6**	1200.394	1411.442	1503.998	1430.236
4	**2**	**8**	1103.369	1844.598	1069.057	1866.168
5	**2**	**10**	1248.691	2182.052	1150.134	2197.431

Table 3, mpi-tile-io data

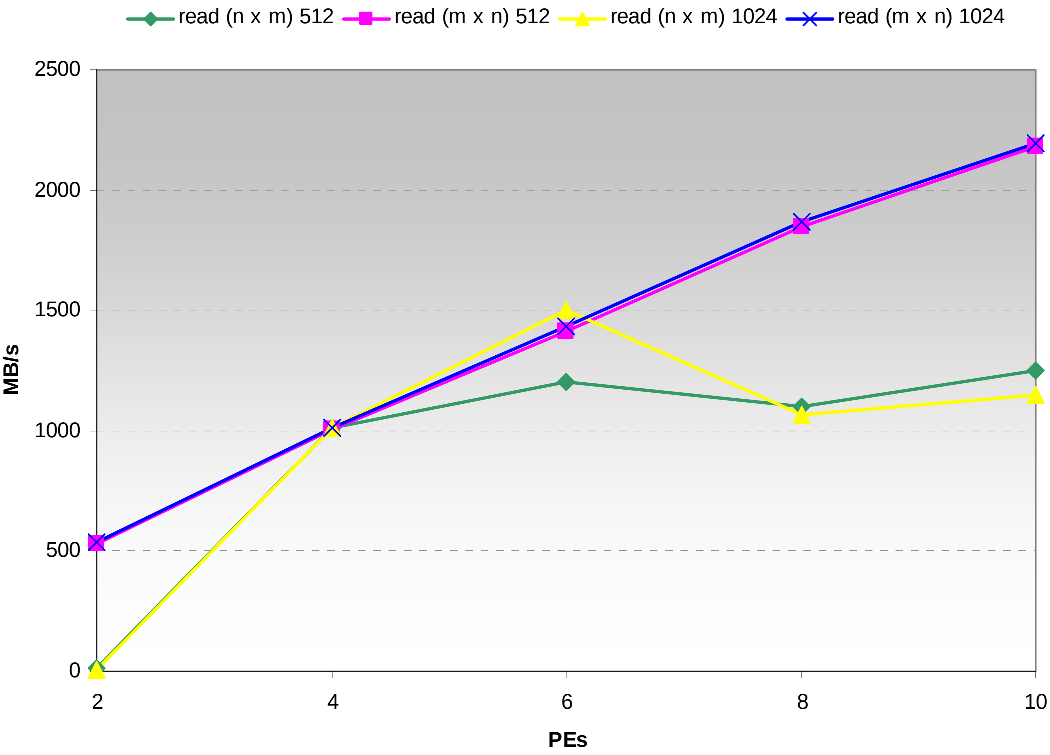


Figure 2, mpi-tile-io read performance on SGI-Prism

Read performance under mpi-tile-io can be seen to vary significantly depending on the organisation of the tiles. When arranged with ‘n’ rows and ‘m’ columns, read performance scales considerably better than in the same case with ‘m’ rows and ‘n’ columns. Segment size on the other hand seems to have little impact on the read performance measured here.

4. Conclusions

Results for both the DLV benchmark and SPECViewPerf indicate poor rendering performance for the SGI Prism system, but these benchmarks don’t take in to account many of the systems features. They do not capture performance data for multiple pipes or adequately test the large shared memory that is a unique selling point of the SGI Prism. It is possible to characterise certain elements of a visualization workload on a parallel system with synthetic benchmarks such as mpi-tile-io but such benchmarks tend to focus on one task (in this case MPI-IO) and are often not representative of real applications that are being run on a system.

High-end visualization systems are generally based on the same commodity GPU technologies found in games machines and PCs so running the benchmarks that we have run here on high-end systems is not entirely without merit, providing useful data on the raw performance of a single graphics pipe. The difficulty with visualization clusters is that they usually have multiple graphics pipes. The issue of importance with such systems is how efficiently a visualization workload scales to use the available number of graphics pipes. There are a wide range of scientific visualization applications and a broadening range of hardware vendors providing visualization cluster solutions for the HPC market. In order to accurately compare these products it is important that the scientific visualization community should develop a set of standard benchmarks that not only measure rendering performance for individual pipes under a relevant workload but that also demonstrate the scalability of scientific visualization applications on a given system.

5. References

[1] SPECViewPerf, http://spec.unipv.it/gpc/opc.static/vp81info.html

[2] DL Visualize, http://www.cse.clrc.ac.uk/cmg/DLV/

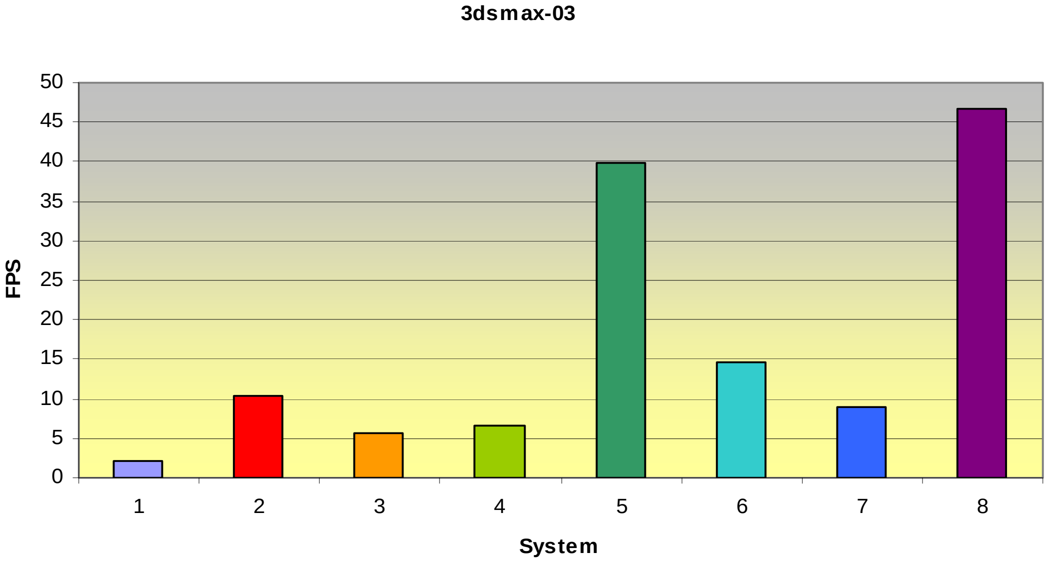
[3] Parallel i/o benchmarking consortium. http://www-unix.mcs.anl.gov/pio-benchmark/

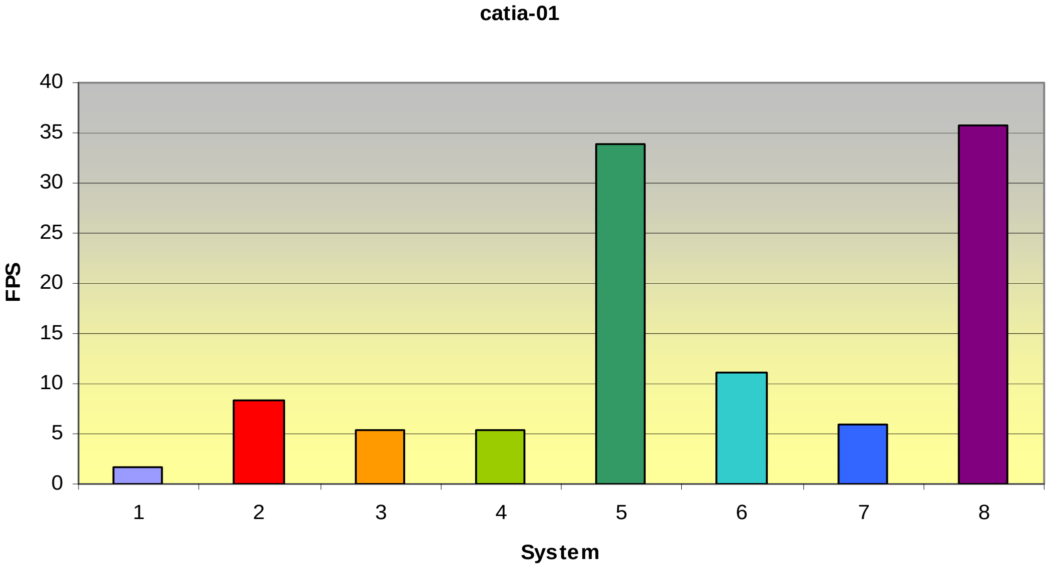
[4] Benefits of Quadrics Scatter/Gather to PVFS2 Noncontiguous IO _Weikuan Yu Dhabaleswar K. Panda, Network-Based Computing Lab, Dept. of Computer Science & Engineering, The Ohio State University {yuw,panda}@cse.ohio-state.edu, 2005

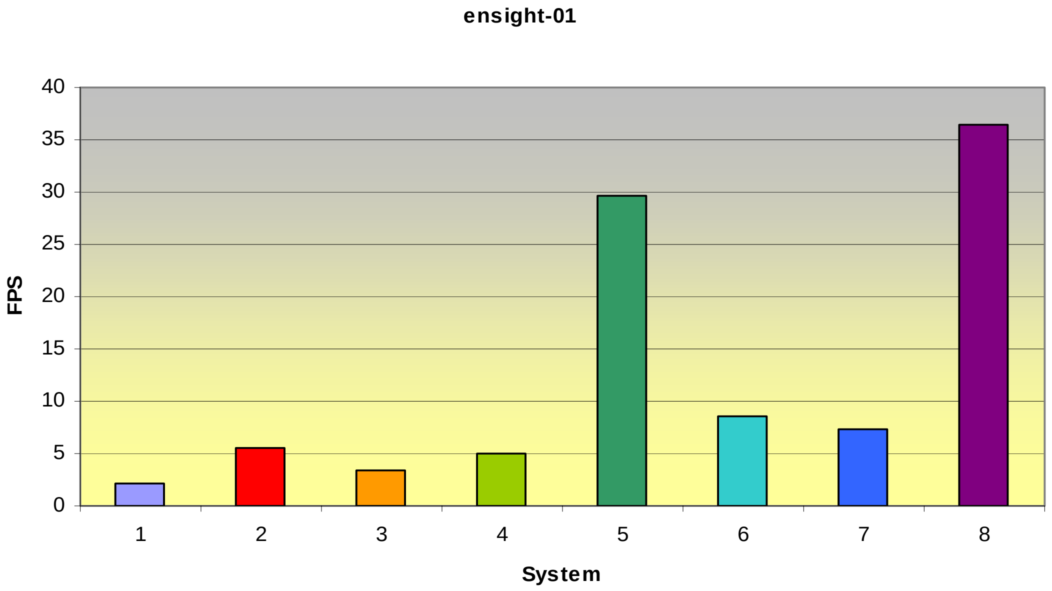
[5] 3DMark benchmark, http://www.futuremark.com/products/3dmark06/

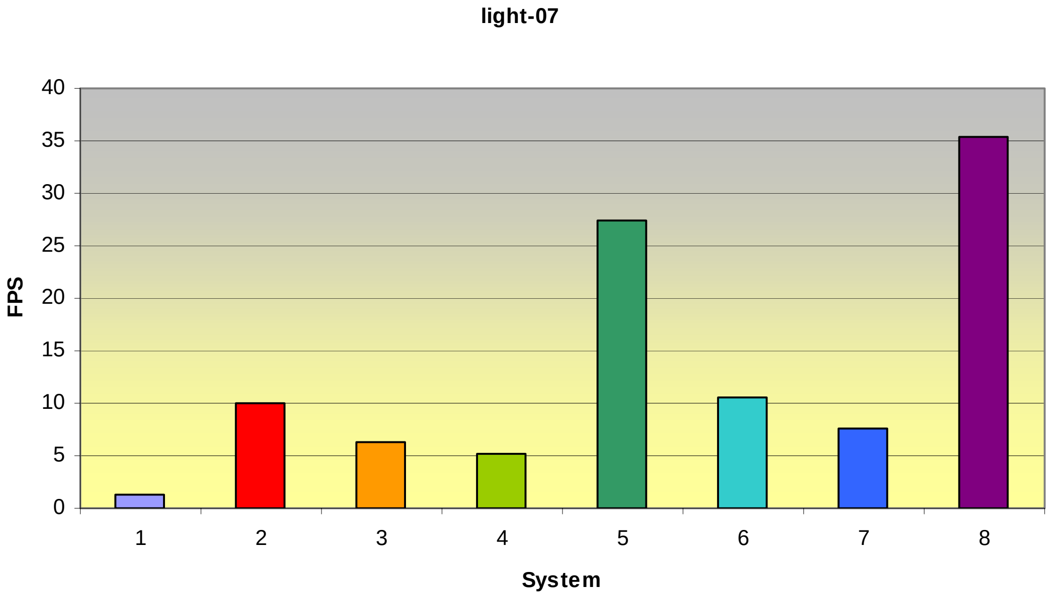
[6] Visualization benchmarking portal at the University of California http://www.ats.ucla.edu/Portal/research_activities/benchmarks/

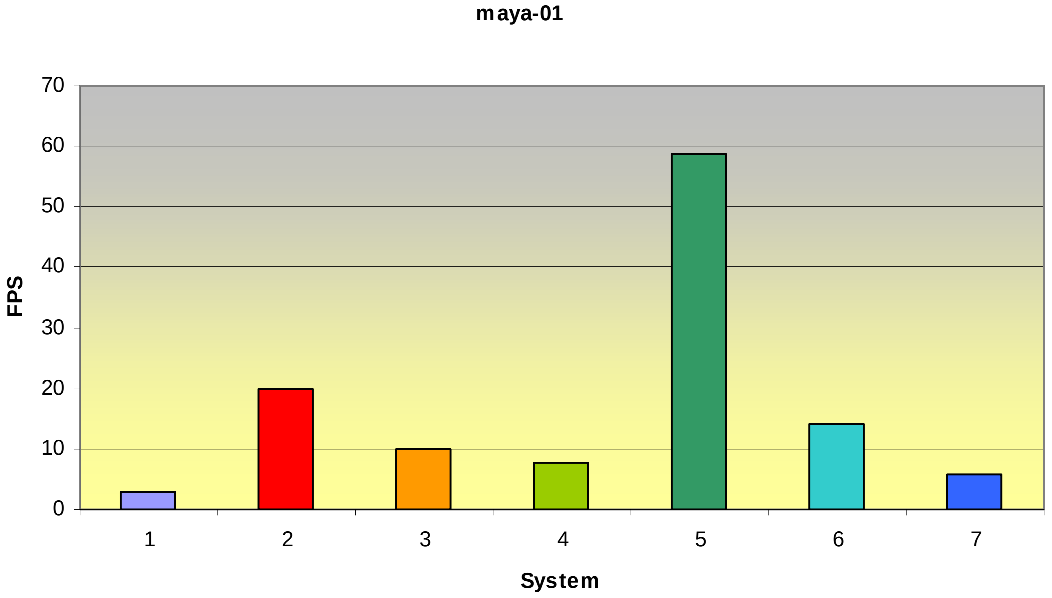
Appendix

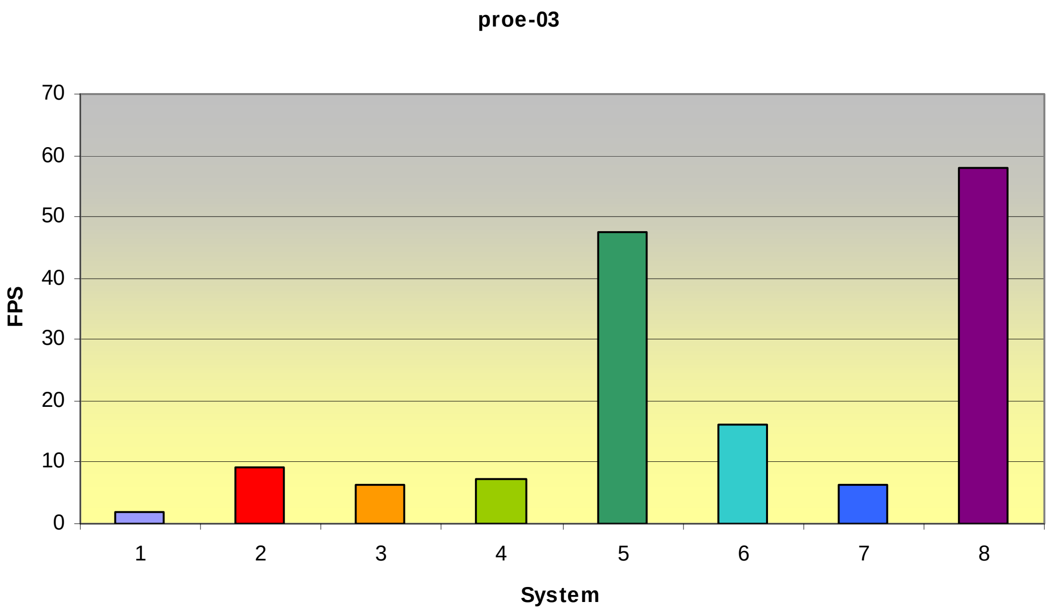

3dsmax-03
FPS
50
45
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
System

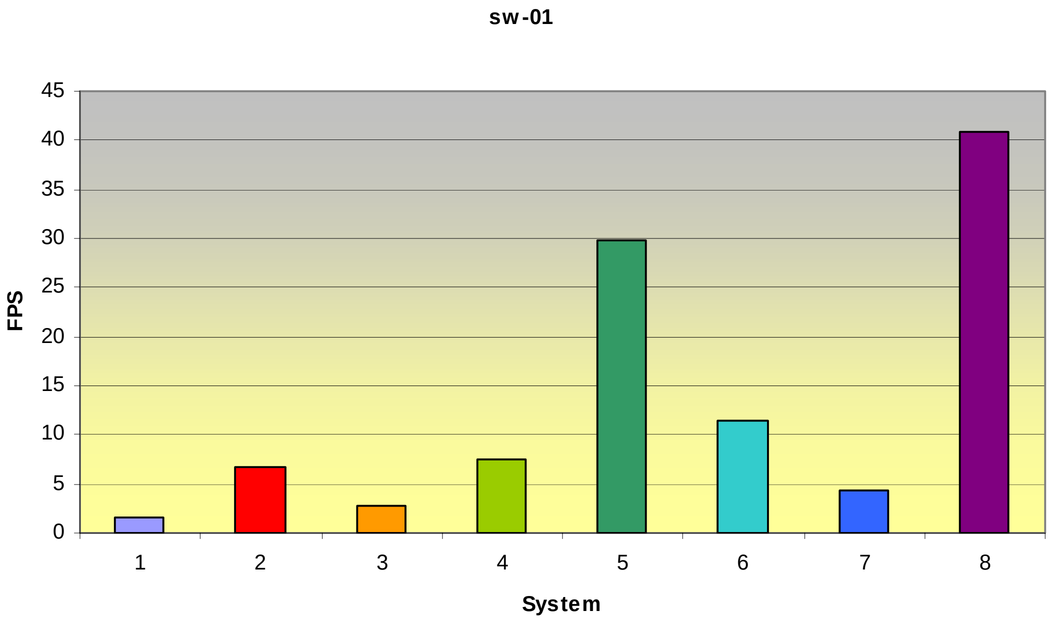

catia-01
FPS
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
System


ensight-01
FPS
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
System


light-07
FPS
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
System


maya-01
FPS
70
60
50
40
30
20
10
0
1
2
3
4
5
6
7
System


proe-03
FPS
70
60
50
40
30
20
10
0
1
2
3
4
5
6
7
8
System


sw-01
FPS
45
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
System

CCLRC

Council for the Central Laboratory of the Research Councils
Chilton, Didcot, Oxfordshire OX11 0QX, UK
Tel: +44 (0)1235 445000 Fax: +44 (0)1235 445808

CCLRC Rutherford Appleton Laboratory
Chilton, Didcot,
Oxfordshire OX11 0QX
UK
Tel: +44 (0)1235 445000
Fax: +44 (0)1235 44580

CCLRC Daresbury Laboratory
Keckwick Lane
Daresbury, Warrington
Cheshire WA4 4AD
UK
Tel: +44 (0)1925 603000
Fax: +44 (0)1925 603100

CCLRC Chilbolton Observatory
Drove Road
Chilbolton, Stockbridge
Hampshire SO20 6BJ
UK
Tel: +44 (0)1264 860391
Fax: +44 (0)1264 860142