

ProEngineer (Wildfire) Benchmark

Goals

Compare performance of ProE on different OS and configurations on the same hardware.

Test system

IBM T40p Thinkpad

1.6 GHz Pentium M (Centrino)

1 Gb RAM

ATI Mobility FireGL 9000, 64Mb (32bit graphics at 1400x1050)

OS's:

- Windows XP Professional, latest service pack and all patches. Graphics driver, latest from IBM (version 7.911-030701m1-010185C-IBM, ATI does not supply windows drivers for laptops directly) optimized for ProE. OS and ProE installed on NTFS partition, swap and benchmark files on separate FAT32 partitions.
- Redhat Linux 9.0, all latest patches (running vanilla, no customizations). Graphics driver, latest from ATI (version fglrx-glc22-4.3.0-2.9.13) with default settings. OS on ext3, benchmark and ProE install on FAT32*. Window managers, Redhat stock Gnome, KDE and Window Maker.

**I realize that this is not optimum but this is dual boot machine with limited space for install in ext3 partition (ReiserFS or XFS would of course be faster).*

Benchmark

Benchmark is thanks to Olaf Cortens (www.proesite.com) version 3.2. It should be noted that version 3.2 times on Wildfire do not compare to those on older versions of ProE. Benchmark was performed on freshly rebooted machine, networking off.

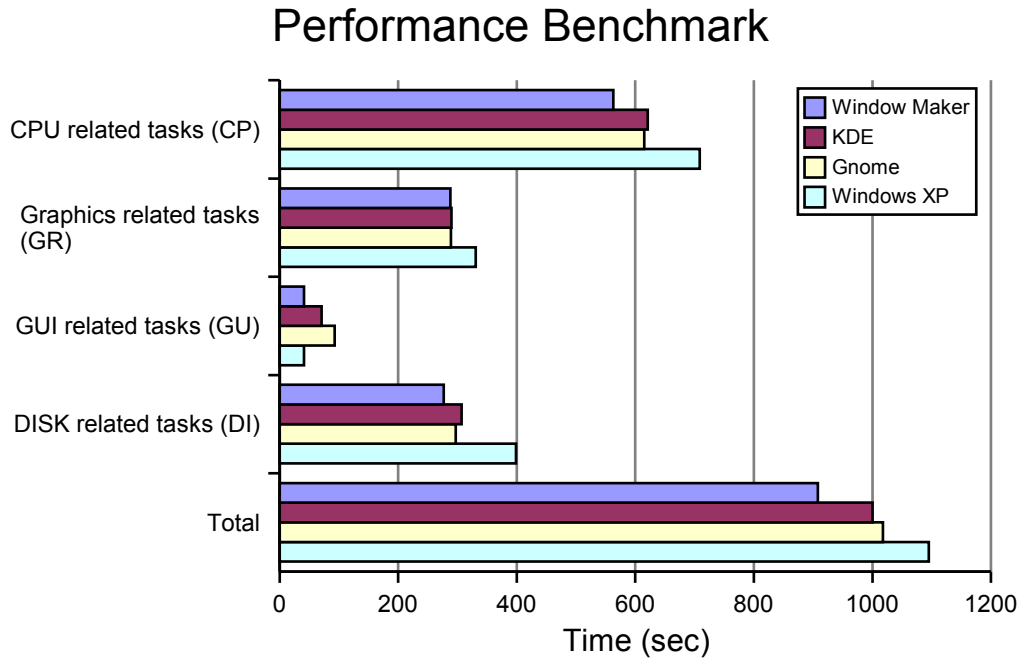
It should be noted that on the Windows XP runs the graphics display appears to leave artifacts behind in the window as the model moves. I was not able to reproduce this with any models other than the one's provided in the benchmark. I was not able to find a setting that eliminated this. I don't think this affected the timing though.

ProEngineer (Wildfire) Benchmark

Results

	Window Maker	KDE	Gnome	Windows XP
Retrieve Assembly (CP)	98	99	99	93
7 wireframe redraws (GR)	16	16	16	18
7 wireframe redraws with DTMS (GR)	19	19	19	20
2 hiddenline redraws (GR)	27	27	27	25
Turn on FastHLR (CP)	4	5	4	5
25 hiddenline redraws with FastHLR (GR)	18	17	18	18
20 shaded mouse spins (GR)	21	21	20	20
200 shaded redraws (GR)	32	33	33	39
4 shade calculations (CP)	19	19	19	20
60 wireframe mouse zooms (GR)	28	28	27	34
initiate advanced shaded mode (CP)	14	15	15	14
2 advanced shaded mouse zooms (GR)	14	13	13	13
300 menu redraws (GU)	12	26	33	14
300 dialogue box redraws (GU)	20	36	51	19
4 regenerations (CP)	15	15	15	14
60 Part Saves (DI)	16	18	21	14
10 Explode translations (GR)	28	29	29	27
6 Screen updates (GR)	21	22	23	23
4 Comp.Disp. translates (GR)	15	15	15	16
11 Model Tree Expansions (GU)	9	9	9	8
18 Automatic Regenerates (CP)	15	16	17	13
18 X-Section shows (GR)	49	50	49	78
30 Perspective views (CP)	15	17	18	15
1 Assy Mass Properties calculations (CP)	11	11	11	12
5 Global Interference checks (CP)	19	19	20	18
1 Assy IGES export (CP+DI)	22	35	31	38
Create Drawing (CP)	27	33	33	46
100 Show Dimensions (CP)	33	47	54	41
1 Regenerate Views NO HIDDEN (CP)	16	17	16	16
1 Regenerate Views HIDDEN LINE (CP)	16	19	18	17
1 Postscript file creation (CP+DI)	27	29	27	112
1 DXF file creation (CP+DI)	212	225	218	235
CPU related tasks (CP)	563	621	615	709
Graphics related tasks (GR)	288	290	289	331
GUI related tasks (GU)	41	71	93	41
DISK related tasks (DI)	277	307	297	399
Total	908	1000	1018	1095

ProEngineer (Wildfire) Benchmark



Comments

Overall the results show Linux (Redhat 9.0) appears to be significantly faster in all areas except GUI tasks. KDE and Gnome are both much slower the GUI operations than XP or Window Maker. It also appears that these KDE and Gnome add overhead to the disk and CPU. Drilling down into the numbers and you see that XP suffered in a few key areas that pushed it's score down significantly. In most other areas both OS's were very close.

Areas for improvement. All benchmarks were conducted on the same FAT32 partition fragmentation would be equal problem there and would be the source of most of the disk time. Alternate choices of file systems may improve numbers for all OS's (it should be noted that in my past benchmarking with Windows NT FAT partitions were significantly faster than NTFS partitions, particularly for swap, it is not clear if that holds for XP).