

Effect of PC configuration on Skype Internet telephony quality and performance

For Intel Corporation

Executive summary

Intel Corporation commissioned Principled Technologies to study the audio quality of a Skype VoIP conference call, along with the response time of additional applications on the same systems in a common business multitasking scenario. Our tests compared a new 2005 Intel® Pentium® D Processor-based business desktop platform with dual-core processing against a roughly three-year-old Intel Pentium 4 processor-based business desktop platform. On both of these platforms, we tested audio quality and system response time using a common enterprise scenario involving a Skype VoIP conference call and two simultaneous Microsoft Office 2003 applications, along with a virus scanning task running in the background. The three-year-old platform experienced significant audio and response time problems. The newer platform showed almost no audio problems and yielded much better application response time.

Key findings

- ❖ The Pentium® D Processor 840-based system with dual-core processing delivered wide-band Skype VoIP audio that we found to be nearly flawless despite heavy system loads.
- ❖ By contrast, the three-year-old Pentium® 4 processor 2.0 GHz-based system experienced a significant number and large total duration of audio glitches under the same workload.
- ❖ The Pentium D Processor 840-based system with dual-core processing delivered much better application response time when facing the sustained load of a Skype VoIP conference call and multiple applications running simultaneously.

We performed the tests using two different Intel processor and platform combinations:

- A new Intel 2005 High Performance Platform: Intel® Pentium® D Processor 840 (3.2-GHz dual-core processor) on the Intel® D955XBK desktop board with the Intel® 955X Express chipset
- A roughly three-year-old platform: Intel® Pentium® 4 processor 2.0 GHz on the Intel® D845GVFN desktop board with the Intel® 845G chipset

We selected a scenario and created a script that allowed us to automate performance testing. The scenario involved a real-time VoIP conference call, a virus scan, and activities using Excel and Word. It simulated the type of workload a desktop PC would face in common multitasking scenarios while under the sustained load of a Skype VoIP conference call. We ran the scenario with both a female voice and a male voice providing the sound for the conference call.

Rather than attempting to quantify the voice problems, we instead looked at them as a typical user on a call might. We counted the number of audio problems, or “glitches,” and categorized those problems for ease of reference (we define the glitch types in the Test results and analysis section, below.) We also recorded the length of each glitch. Figure 1 details the audio problems the three-year-old system encountered on each of the five test runs we conducted with each voice type. As you can see, this older system encountered a noticeable number of problems on every test run, with significantly more problems on the female voice sample test than the male voice test. The glitches also lasted significant amounts of time, particularly on the calls with the female voice sample. As we show in the “Test results and analysis” section, the newer systems, by contrast, had almost no problems across all five runs with either voice type.

Voice sample/Run	Counts of types of audio glitches					Total glitches	Total time involving glitches (mm:ss.ss)	Total voice sample time (mm:ss:ss)
	Pause	Stutter	Drop-out	Boing	Stretch			
Female voice: Run 1	0	4	2	1	1	8	01:00.21	11:06.20
Female voice: Run 2	0	2	4	1	4	11	01:28.18	11:24.05
Female voice: Run 3	2	2	0	4	0	8	00:22.50	11:24.25
Female voice: Run 4	1	8	0	5	0	14	00:59.03	11:35.95
Female voice: Run 5	1	4	0	1	0	6	00:10.60	10:48.00
Male voice: Run 1	1	0	0	1	0	2	00:00.48	10:49.30
Male voice: Run 2	2	0	0	1	0	3	00:02.15	11:25.00
Male voice: Run 3	1	0	0	1	2	4	00:17.49	10:49.15
Male voice: Run 4	2	2	1	2	2	9	00:02.50	11:10.15
Male voice: Run 5	1	1	1	0	0	3	00:01.78	10:48.20

Figure 1: The number, type, and total duration of audio glitches on the Pentium 4 processor 2.0GHz-based system for five runs each of female and male voices with the system under heavy load.

We coined the terms we used to describe each type of audio glitch, and we explain these terms in detail below and show samples of each. Briefly, a pause is when the voice stops and later resumes at the same point. A stutter is when part of the beginning of a word is repeats. A drop-out is when the voice stops and later resumes, omitting part of the sample. A boing is a brief mechanical-sounding noise. A stretch is when the end of a word elongates.

As Figure 2 shows, we found that the newer business platform yielded significantly better application response time than the older system. On all timed tasks, the system with the Intel Pentium D Processor 840 with dual-core processing significantly outperformed the older system with the Intel Pentium 4 processor 2.0 GHz, scoring from 296 percent to 595 percent faster on different applications and tasks. Many of these performance improvements translated into multi-second time savings that would be noticeable to users. In one case, for example, a task that completed in 15.7 seconds on the Intel Pentium D Processor 840-based system with dual-core processing took over 63.7 seconds on the system with the Intel Pentium 4 processor 2.0 GHz.

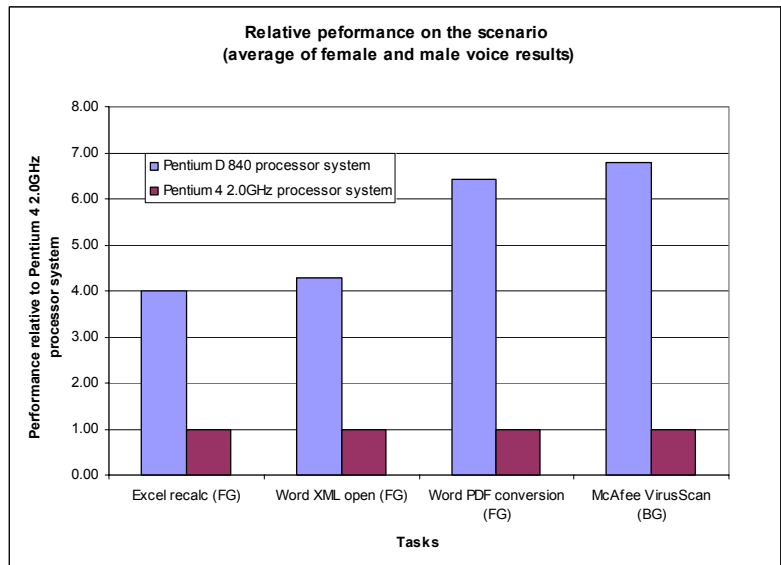


Figure 2: Overall average performance, relative to the Intel Pentium 4 Processor 2.0 GHz processor-based system on all scenarios' foreground and background tasks for both systems. Taller bars indicate better performance.

To make comparisons easier, we always show normalized results in which we divided each system's time by the time it took the system with the Intel Pentium 4 processor 2.0 GHz to do the task at hand. The result for the Intel Pentium 4 processor 2.0 GHz-based system is thus always 1.00, because it's the basis of comparison. Results higher than 1.00 indicate how much faster the newer system is than the older one. Because of the normalization, larger results are better. So, for example, a result of 1.80 means that the newer system is 80 percent faster than the system with the Intel Pentium 4 processor 2.0 GHz.

We used the following common enterprise business applications in our test scenario:

- Adobe Acrobat* 7.0.3 Standard
- Microsoft* Office Excel 2003 (Service Pack 1)
- Microsoft Office Word 2003 (Service Pack 1)
- Network Associates' McAfee VirusScan* Enterprise version 8.0i
- Skype Limited Skype Version 1.4.0.71

So that we could make the test systems as similar as reasonably possible, while varying the processors and motherboards, we built the systems rather than purchasing them from vendors. Intel specified the general system types and supplied processor and motherboard for the Pentium D Processor 840-based dual-core processing system. We purchased the older motherboard and processor, as well as such common components as RAM, hard disks, and optical drives. Each test system had the following basic components (see Appendix A for in-depth configuration details):

- 512MB of the fastest RAM its motherboard would support
- 80GB hard disk with an 8MB buffer (SATA or EIDE depending on what the system would support)
- DVD-RW optical drive
- Windows XP Professional Service Pack 2

To make it easier to test scenarios repeatedly and accurately, we automated the application functions using scripts we wrote in IBM's Rational Visual Test* 6.5. In our analyses we used the times from our tests of the scripts.

In the following sections we look more closely at the test scenarios (Application scenario), examine the results of our tests (Test results and analysis), and give detailed information about how we actually performed the tests (Test methodology). In the appendices, we present the configurations of the test systems, explain how to manually execute the application functions in our scenarios, and discuss some issues in the development of the test scripts.

Application scenario

To demonstrate the types of Skype VoIP improvements that 2005 business platforms based on the Intel Pentium D Processor with dual-core processing could deliver, we developed a common business multitasking scenario involving a VoIP conference call that takes place while typical office applications run at the same time. This scenario includes a real-time conference call using Skype Limited's Skype and business functions using Microsoft Office 2003 and Network Associates McAfee VirusScan. We examined these processor-intensive functions because they often cause users to wait. Users are therefore likely to appreciate performance improvements in these areas.

In this scenario, Pat, a marketing manager, is hosting a Skype conference call with three other team members. During the call, an automatic virus scan the IT department has scheduled begins. As the scan runs, team members request a PDF of an XML document that Pat has referenced. Pat first opens an Excel spreadsheet that she will use during the call. Because she isn't sure that the calculations on the spreadsheet are current, she starts a manual recalculation as soon as the spreadsheet opens. When the calculation ends, she opens the XML file and starts the PDF conversion. The virus scan is running the entire time. Pat and the team wrap up the call after the virus scan finishes.

For more details on how we executed and measured this scenario, our specific test functions, and the files the scenario involved, see the Test methodology section and Appendix B.

Test results and analysis

During this scenario, a number of tasks are underway. For four of these tasks, the performance measurement that matters is response time. Therefore, for these tasks—the virus scan, the Excel recalculation, the XML file open, and the PDF conversion—we timed how long it took each application to complete the task, and we

report those times. For the Skype conference call that the test machine hosts, what matters is sound quality. For that task, we report the total number of audio problems, or “glitches,” that occurred and the total number of seconds with audio problems. To accommodate any differences between female and male voice samples, we ran the test with one of each.

Figures 3 and 4 detail the audio problems we encountered with the female and male voice samples, respectively, on all five runs of the test on both of the test systems.

Run	Counts of types of audio glitches					Total glitches	Total time involving glitches (mm:ss.ss)	Total voice sample time (mm:ss:ss)
	Pause	Stutter	Drop-out	Boing	Stretch			
Pentium® D Processor 840-based system								
1	0	0	1	1	0	2	00:04.03	02:01.00
2	0	0	0	0	0	0	00:00.00	01:59.25
3	0	0	0	0	0	0	00:00.00	01:59.30
4	0	0	0	1	0	1	00:00.13	01:57.75
5	0	0	0	0	0	0	00:00.00	01:58.95
Pentium® 4 processor 2.0GHz-based system								
1	0	4	2	1	1	8	01:00.21	11:06.20
2	0	2	4	1	4	11	01:28.18	11:24.05
3	2	2	0	4	0	8	00:22.50	11:24.25
4	1	8	0	5	0	14	00:59.03	11:35.95
5	1	4	0	1	0	6	00:10.60	10:48.00

Figure 3: The number, type, and total duration of audio glitches on the two systems for five runs of the scenario using the female voice sample with the systems under heavy load.

We look at representative samples of these glitches in Figures 6 through 10 and describe them in more detail later in this section.

The last two columns of Figures 3 and 4 record the total elapsed time of all of the glitches and how long the sample ran. On the Pentium 4 processor 2.0 GHz-based system, the total voice sample time is significantly longer than on the other system because that system took so much longer to complete the other tasks. The total amount of glitch time, however, is still a much higher percentage of the total sample time on that system than on the newer system.

As you can see, the Pentium D Processor 840-based system experienced very few glitches with the female voice, and those that occurred were short in duration. On the majority of tests, that system experienced no glitches. On the male voice runs, the Pentium D Processor 840-based system exhibited no glitches at all.

Run	Counts of types of audio glitches					Total glitches	Total time involving glitches (mm:ss.ss)	Total voice sample time (mm:ss:ss)
	Pause	Stutter	Drop-out	Boing	Stretch			
Pentium® D Processor 840-based system								
1	0	0	0	0	0	0	00:00.00	01:56.50
2	0	0	0	0	0	0	00:00.00	02:00.25
3	0	0	0	0	0	0	00:00.00	02:00.40
4	0	0	0	0	0	0	00:00.00	01:59.05
5	0	0	0	0	0	0	00:00.00	02:01.70
Pentium® 4 processor 2.0GHz-based system								
1	1	0	0	1	0	2	00:00.48	10:49.30
2	2	0	0	1	0	3	00:02.15	11:25.00
3	1	0	0	1	2	4	00:17.49	10:49.15
4	2	2	1	2	2	9	00:02.50	11:10.15
5	1	1	1	0	0	3	00:01.78	10:48.20

Figure 4: The number, type, and total duration of audio glitches on the two systems for five runs of the scenario using the male voice sample with the systems under heavy load.

Figure 5 shows the median times (in seconds) that each of the test systems took to complete each of the application tasks we timed.

System	Intel® Pentium® D Processor 840 (3.2-GHz dual-core) on Intel D955XBK	Intel® Pentium® 4 processor (2.0 GHz) on Intel D845GVFN
Female voice		
Excel recalc (FG)	15.718	63.657
Word XML open (FG)	8.079	33.891
Word PDF conversion (FG)	57.047	368.094
McAfee VirusScan (BG)	95.265	661.891
Male voice		
Excel recalc (FG)	15.891	62.875
Word XML open (FG)	7.547	33.172
Word PDF conversion (FG)	57.406	365.531
McAfee VirusScan (BG)	95.907	634.172

Figure 5: Median results for both test systems on the scenario with both female and male voices. The medians are of five runs. Times are in seconds. FG indicates a foreground task. BG indicates a background task.

The following sub-sections explore these results in more detail.

Application scenario (with the female voice sample)

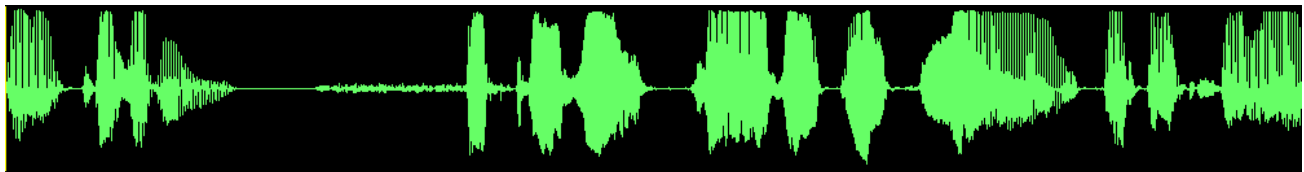
In this scenario, Excel recalculates a 2.6MB spreadsheet (the first foreground task); Word opens a 1.3MB XML file (the second foreground task) and then converts the file to PDF format (the third foreground task); and while the foreground tasks are executing, Network Associates McAfee VirusScan scans for viruses (the background task). The entire time these tasks are running, Skype is running a VoIP conference call. We used a high-quality female voice (www.voiceoverdirectory.com/female_voiceover_melanie_haynes.html) during this scenario. (See Figure 20 for a diagram of the systems involved in this test.) Because the results for the female voice sample showed the most glitches, we will look at the specific glitches in depth here, then summarize the male voice results in the next sub-section.

To understand the individual types of glitches, it's useful to look at how their waveforms compare to those of glitch-free samples. In the figures that follow we compare, for each glitch type, the waveform of the voice

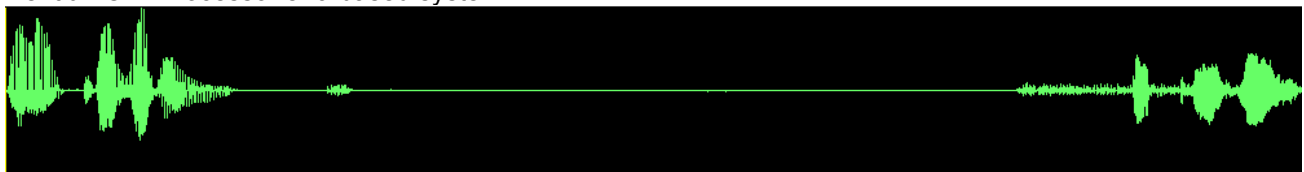
sample on the Pentium D Processor 840-based system with that of the glitch on the Pentium 4 processor 2.0GHz-based system.

As the waveform figures show, on the three-year-old system not only do glitches occur, the waveforms also often have much lower amplitudes—an indication of a loss of audio range and thus quality—compared to the newer system.

The first type of glitch is a pause. A pause occurs when the recording contains a segment of silence where there should be speech—but the recording does not lose any part of the speech, as if the recording device stopped to synchronize with the original broadcast before continuing to the next word. Pauses differ from drop-outs in that all the content of the speech made it through, but with distracting delays between words or sounds. For example, in the recorded segment of the female voice from the Pentium 4 processor 2.0GHz-based system in Figure 6, we experienced approximately 2.5 seconds of silence between sentences; the same interval in the original speech was over 200 milliseconds.



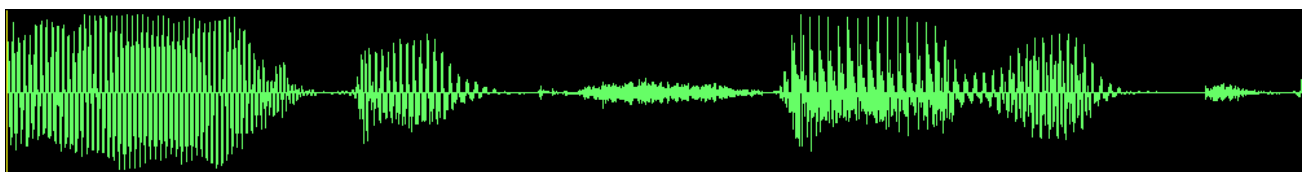
Pentium® D Processor 840-based system



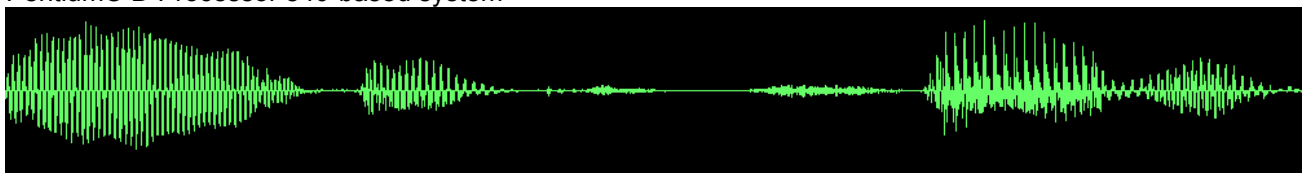
Pentium® 4 processor 2.0GHz-based system

Figure 6: Pause.

We refer to the second type of glitch as a stutter. A stutter occurs when there is a distinct repeating of a syllable within a word. A stutter differs from a stretch in that the stutter repeats the syllable, while the stretch elongates it. In the example using the female voice in Figure 7, we should hear her say “two excitatory”. By contrast, when the stutter occurred on the older system, the voice repeated the second syllable—sounding as if it had said, “two excicitary.”



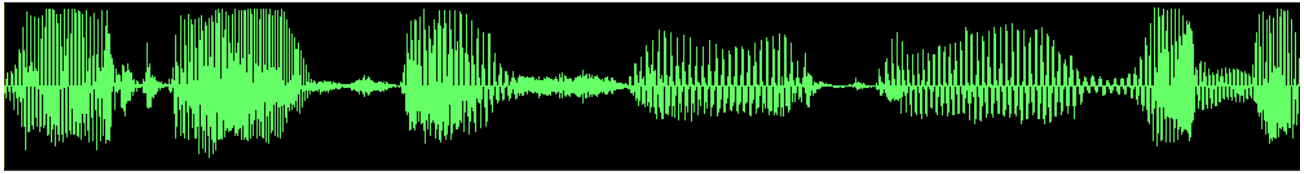
Pentium® D Processor 840-based system



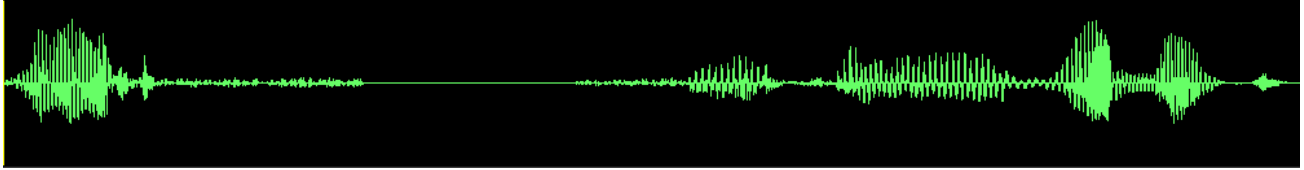
Pentium® 4 processor 2.0GHz-based system

Figure 7: Stutter.

The third type of glitch is a drop-out. A drop-out occurs when we should be hearing speech, but hear silence instead. Drop-outs differ from pauses in that the speech continues, but the sound becomes inaudible. For example, in the recording using the female voice in Figure 8, the voice on the two newer systems said, “We also see an inhibitor...,” while with the drop-outs the voice appears to say, “We ...[silence]... an inhibitor....” Missing key words can obviously create serious problems in any call.



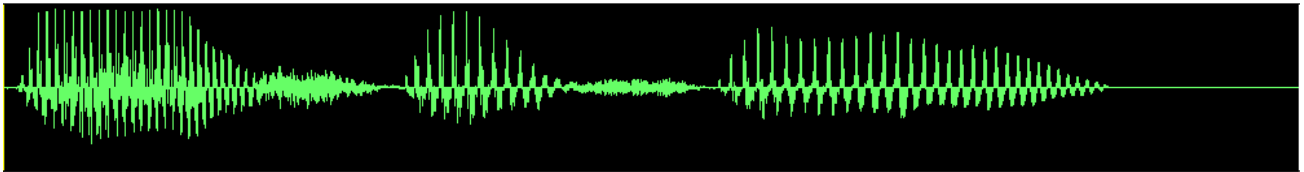
Pentium® D Processor 840-based system



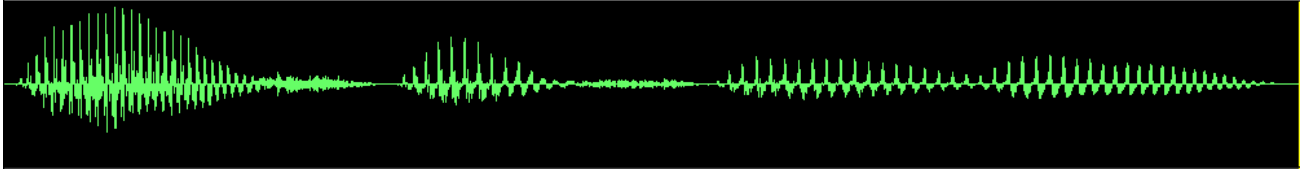
Pentium® 4 processor 2.0GHz-based system

Figure 8: Drop out.

Figure 9 shows the next type of glitch, a boing. A boing occurs when a compression in the sound of the voice causes a brief, artificial, mechanical sound within a word. A boing can be part of a stutter or a stretch, but it always carries the distinctive, robotic tone. In the example of the recording using the female voice below, the voice said, “processing”. With the glitch, however, a boing occurs in the last syllable, so that syllable sounds bad and also stretches.



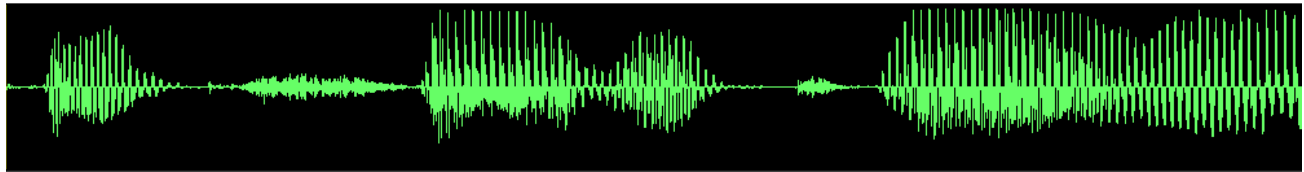
Pentium® D Processor 840-based system



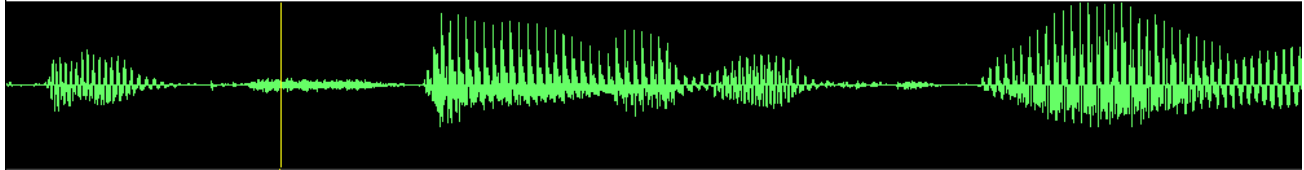
Pentium® 4 processor 2.0GHz-based system

Figure 9: Boing.

The final type of glitch is a stretch. A stretch occurs when the system elongates single words, as if the speaker is enunciating a certain syllable longer than normal. A stretch differs from a stutter in that a stretch elongates a syllable within a word, while a stutter repeats it. In the example of a stretch error in Figure 10, the speaker seems to accentuate and slightly elongate the second syllable of a word.



Pentium® D Processor 840-based system



Pentium® 4 processor 2.0GHz-based system

Figure 10: Stretch.

Not only did the Pentium 4 processor 2.0GHz-based system experience a significant amount of voice glitches, its response time on the application tasks was also much slower than that of the newer system.

Figure 11 shows the large performance improvement of the newer system: the Pentium D Processor 840-based system was 305 percent to 545 percent faster on the foreground tasks and 595 percent faster on the background task. These large percentage improvements represent absolute time improvements ranging from 25 seconds on the XML open foreground task to over nine minutes on the VirusScan background task—all time savings that users would be happy to realize.

To understand why the newer system completed the tasks so much more quickly, it's useful to examine the loads their processors faced. We captured the processor utilization curves in Figures 12 through 16 with Windows XP's Perfmon utility. Each of these curves illustrates how demanding each task would be if run alone on a modern system. In multitasking scenarios, the system faces each of these same demands, but all at the same time from all the tasks that are running simultaneously. When all the simultaneous tasks have minimal processor requirements, keeping them all executing smoothly in parallel may not be difficult. When, by contrast, one or more of the tasks has significant processor demands, meeting those demands at the same time can be extremely difficult for a processor without the added execution ability of a Pentium D Processor with dual-core processing.

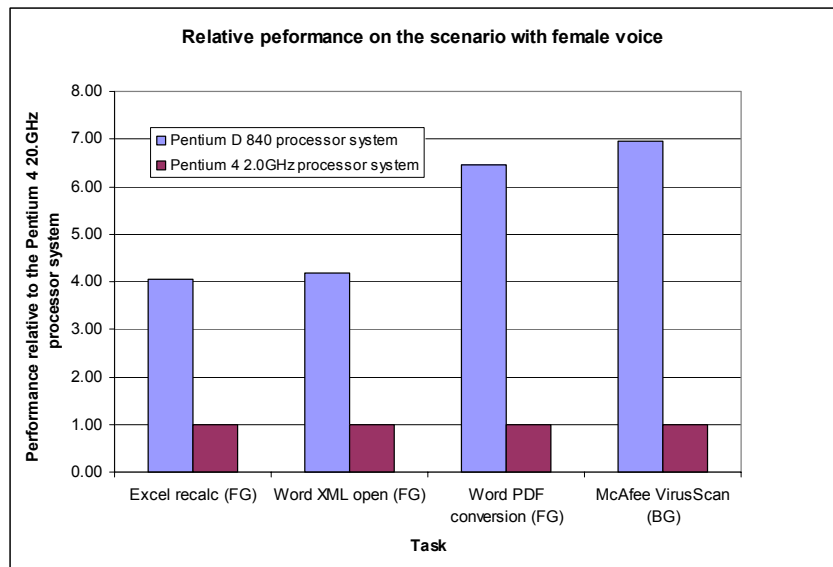


Figure 11: Performance, relative to the Pentium® 4 2.0 GHz-based system, of both of the test systems on a set of concurrent tasks. Taller bars indicate better performance.

We captured each task's processor-demand curve on a Pentium 4 Processor 640-based system with HT Technology turned off. We chose this system because it is a current platform with a single core; had we captured the processor utilization on the older system in this report, we might have given the impression of overstating the processor demands of individual tasks. We also chose to turn off HT Technology to make it easy to see the processor demands; had we captured the processor utilization with HT Technology enabled or on a dual-core system, Perfmon would have capped each logical (or physical) core at 50 percent utilization.

In all the processor utilization charts, we use a blue line for each foreground task and a red line for each background task.

In our test scenario, the Skype VoIP conference call puts modest but steady load on the processor. The background McAfee VirusScan requires almost all of the processor. Each of the three foreground tasks—the Excel recalculation, the XML open, and the PDF conversion tasks—demands a lot from the processor. The need to run three tasks at the same time clearly shows how the second core in the Pentium D Processor 840-based system can markedly improve performance over the Pentium 4 processor 2.0GHz-based system.

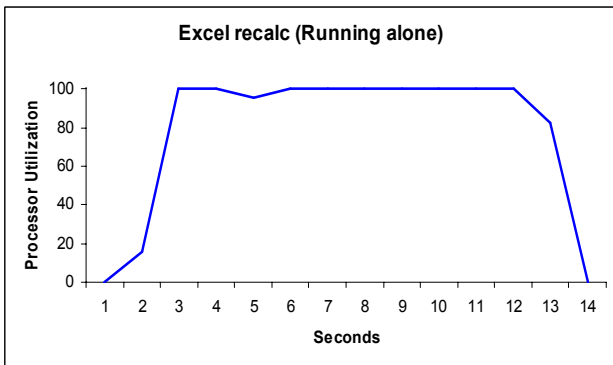


Figure 12: System processor utilization of the first foreground task (recalculating an Excel spreadsheet) running alone.

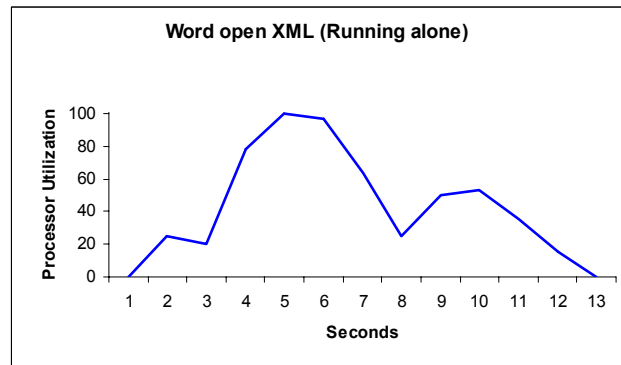


Figure 13: System processor utilization of the second foreground task (opening an XML file in Word) running alone.

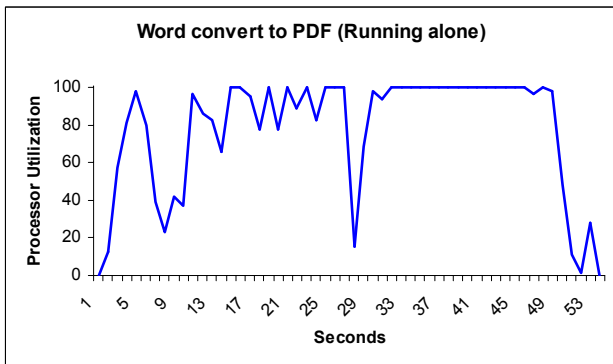


Figure 14: System processor utilization of the third foreground task (converting a document to PDF in Word) running alone.

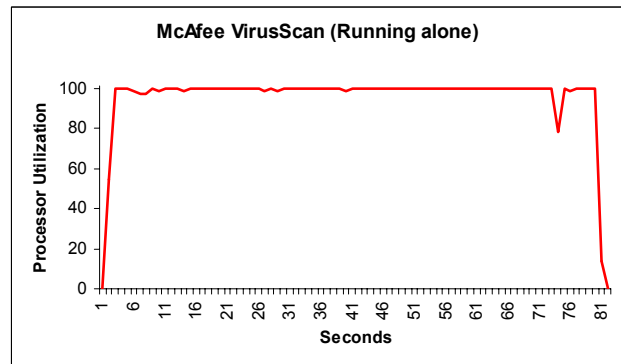


Figure 15: System processor utilization of the background task (scanning for viruses in McAfee VirusScan) running alone.

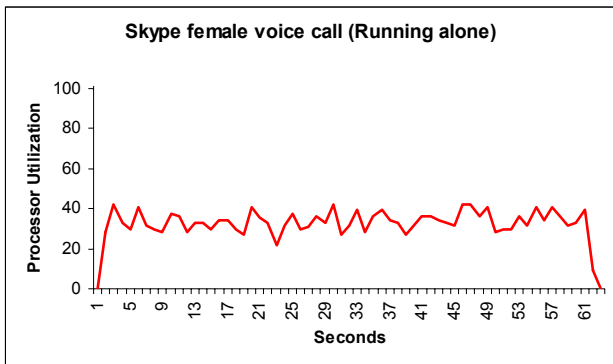


Figure 16: System processor utilization of the VoIP conference call using a female voice in Skype running alone.

Application scenario (male voice)

This scenario is identical to the one with the female voice except that we used a different (male), high-quality voice

(www.voiceoverdirectory.com/male_voiceover_Rick_Adamson.html). We summarized the glitches and showed examples of them above, so here we will discuss only the application response times.

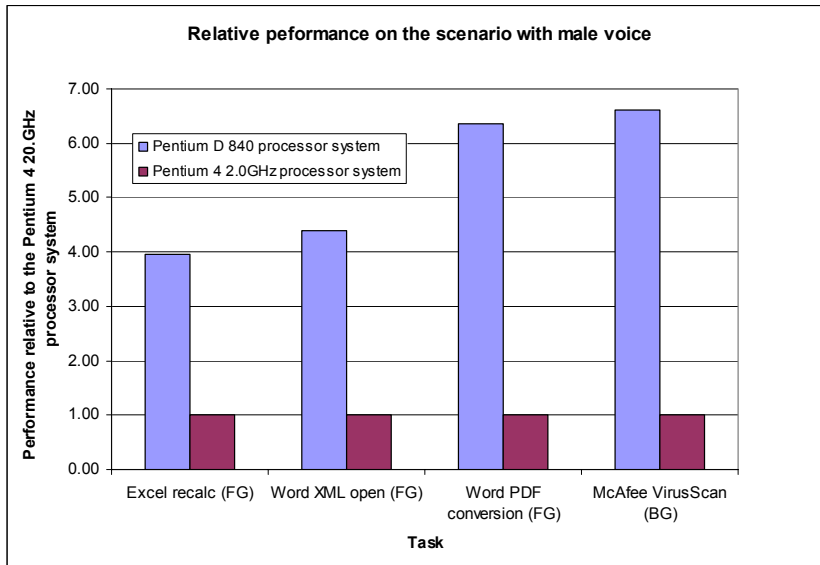


Figure 17: Performance, relative to the Pentium® 4 2.0GHz-based system, of both of the test systems on a set of concurrent tasks. Taller bars indicate better performance.

Figure 17 shows the large performance improvement for the newer system. The Pentium D Processor-based system with dual-core processing was 296 percent to 537 percent faster on the foreground tasks and 561 percent faster on the background task. These large percentage improvements represent absolute time improvements ranging from 25 seconds on the second foreground task to almost nine minutes on the VirusScan task.

Figure 18 shows the processor utilization for the Skype conference call with the male voice. Though different from that of the female voice, this demand curve does not have a significantly different peak processor utilization percentage.

These application scenario results show that the Pentium D Processor with dual-core processing is likely to yield significant performance improvements compared to the three-year-old systems typical of those in use by average business users when running multiple processor-intensive tasks in the face of the sustained load of a Skype conference call.

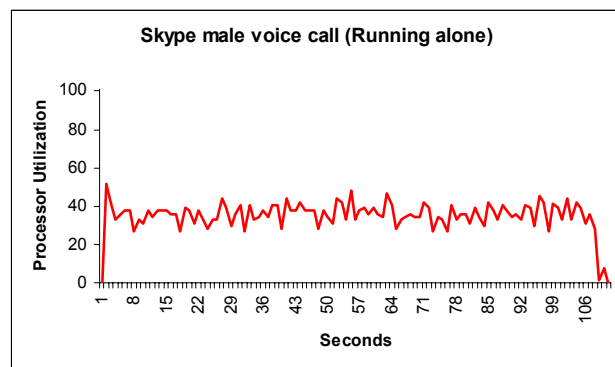


Figure 18: System processor utilization of the VoIP conference call using a male voice in Skype running alone.

Test methodology

We evaluated the performance of the application scenario both by hand and with an automated test script, which we developed with IBM's Visual Test 6.5. Appendix B details the steps we followed when we hand-timed the scenarios. In this paper, we concentrate our discussions on the results of the automated script, because those results are generally more repeatable than hand timings.

We collected results for ten runs of the script—five with the female voice sample and five with the male sample—on each of the test systems. We refer in this paper only to the median results of those runs on each system. The script produces times in milliseconds, where lower times to perform a given function are better.

Appendix A provides detailed configuration information on the test systems. We set up each of those systems using the following process:

- Install Microsoft Windows XP Professional.
- Using the standard Microsoft Windows Update Web site, apply all current (as of September 28, 2005) Windows XP critical updates, including Windows XP Service Pack 2—but not including optional updates, such as Windows Media Player 10 or Windows Media Connect, that were unrelated to the goal of this paper.
- Install Microsoft Office 2003.
- Using the standard Microsoft Office Update Web site, apply all current (as of September 28, 2005) Office 2003 updates.
- Turn off Windows Service Pack 2 Security Center Pop-up Alerts. Doing so prevents such alerts from occurring during testing and affecting results.
 - Open the system Control Panel.
 - Choose Security Center.
 - Click “Change the way Security Center Alerts me” on the left.
 - Uncheck Firewall, Automatic Updates and Virus Protection.
- Turn off Windows Automatic Updates. Doing so prevents such updates from occurring during testing and affecting results.
 - Open the system Control Panel.
 - Choose Automatic Updates.
 - Select “Turn off Automatic Updates.”
- To ensure as consistent a starting point as possible for the performance measurements, defragment the hard disk of each system.
- Using Symantec's Ghost utility, make an image of each system's hard disk. (This image lets us return to a clean and consistent starting point whenever necessary.)
- Install the additional software (versions current as of September 28, 2005) that the scenario requires:
 - Adobe Acrobat 7.0.3 Standard
 - Microsoft Office Excel 2003 (Service Pack 1)
 - Microsoft Office Word 2003 (Service Pack 1)
 - Network Associates' McAfee VirusScan Enterprise version 8.0i
 - Skype Limited Skype Version 1.4.0.71

To obtain the results for the application scenario, we followed the same basic process in each test run:

1. Reboot the system under test.
2. Wait ten seconds after the Windows hourglass disappears and Windows XP has completed its startup sequence. This delay ensures the system is in a consistent starting state.
3. Run the test script (or hand-timed application functions, as appropriate).
4. Record the results.
5. Repeat the above process five times. (If any test failed, we discarded that test's results and ran the test again.)
6. Take the median result of the five iterations of each task as the representative result of that task.

Appendix A: Test system configuration information

In this appendix, we provide in-depth configuration information for the two test systems.

We installed the applications necessary to run the scenarios on each system using the same steps. Because IT organizations often employ standard configurations—and to make the systems as comparable as reasonably possible—we installed on each system only those applications we needed for our testing. We applied all Windows critical updates, including SP2, that were available as of September 28, 2005, except those optional updates unrelated to enterprise systems, such as Windows Media Player 10 or Windows Media Connect. We applied all the application updates, such as those for Microsoft Office 2003, that were available as of the same date. We then defragmented the hard disk of each system and saved the resulting disk image. We performed all tests on these “clean” disk images.

System	Intel® Pentium® D Processor 840 (3.2 GHz Dual-core) on Intel D955XBK	Intel® Pentium® 4 processor 2.0 GHz on Intel D845GVFN
General		
Processor and OS kernel: (physical, core, logical) / (UP, MP)	1P2C2L / MP	1P1C1L / UP
Number of physical processors	1	1
Single or dual-core processor	Dual	Single
Processor Hyper-Threading Technology status	N/A	N/A
System power management policy (AC/AlwaysOn, AC/Adaptive, DC/AlwaysOn, DC/Adaptive, Other)	AC/AlwaysOn	AC/AlwaysOn
CPU		
Segment (server, desktop, mobile)	Desktop	Desktop
Vendor	Intel	Intel
Model number	Pentium D Processor 840	Pentium 4 processor
Stepping	4	9
Socket type and number of pins	Socket T/ LGA775	Socket 478/ mPGA-478
Core frequency	3200	2000
Front-side bus frequency	800	400
L1 cache	16KB + 12KB ⁽¹⁾	8KB + 12KB
L2 cache	1MB ⁽¹⁾	512KB
Platform		
Vendor and model number	Intel D955XBK	Intel D845GVFN
Motherboard model number	D955XBK	D845GVFN
Motherboard chipset	I955XE	i845GV
Motherboard revision number	AAC96732-405	AAC77646-105
Motherboard serial number	ABBK52000110	FCFN44613282
BIOS name and version	Intel BK95510J.86A.1959.2005.0826.17 23	Intel FN84510A.86A.0020.P06.0503 092001
BIOS settings	Setup default	Setup default
Chipset INF driver	7.0.0.1025	4.04.1007
Memory module(s)		
Vendor and model number	Micron Technology CM32M6408U26AD2F-3VC	Spectek s80032vmctl
Type	PC5300 DDR2-SDRAM	PC2700 DDR-SDRAM

System	Intel® Pentium® D Processor 840 (3.2 GHz Dual-core) on Intel D955XBK	Intel® Pentium® 4 processor 2.0 GHz on Intel D845GVFN
Speed in the system currently running (MHz)	533	266
Timing/latency (tCL-tRCD-tRP-tRASmin)	4-4-4-12	2-3-3-6
Size	512MB	512MB
Number of sticks	2 x 256MB	2 x 256MB
Chip organization (single-sided, double-sided)	Single-sided	Single-sided
Channel (single/dual)	Dual	Dual
Hard disk		
Vendor and model number	Western Digital WDC WD800JD-75JNA0	Western Digital WDC WD800JB-00JAA0
Size	80GB	80GB
Buffer size	8MB	8MB
RPM	7200	7200
Type	Serial ATA 1.5	ATA-100
Controller	Intel 82801GB (ICH7)	Intel 82801DB (ICH4)
Windows XP Microsoft hard disk driver version number	5.1.2535.0	5.1.2535.0
Operating system		
Name	Microsoft Windows XP Professional	Microsoft Windows XP Professional
Build number	2600	2600
Service pack	SP2	SP2
File system	NTFS	NTFS
Kernel	ACPI Multiprocessor PC	ACPI Uniprocessor PC
Language	English	English
Microsoft DirectX version	9.0c	9.0c
Graphics		
Vendor and model number	nVidia GeForce 6600	Intel 82845GV
Chipset	GeForce 6600	Intel 82915GV Express
BIOS version	5.43.02.16.00	Intel Video BIOS 3364
Type (PCI Express, AGP 8X, integrated)	PCI Express	Integrated
Memory size	256MB	64MB shared
Resolution	1024 x 768 x 32-bit color	1024 x 768 x 32-bit color
Driver	nVidia 7.8.0.1 8/2/2005	Windows XP driver: Intel 6.14.10.3762
Sound card/subsystem		
Vendor and model number	SigmaTel High Definition Audio CODEC	Realtek AC'97 Audio
Type (PCI, integrated)	Integrated	Integrated
Driver	SigmaTel 5.10.4647.0	Realtek 5.10.0.5870
Network card/subsystem		
Vendor and model number	Intel PRO/1000 PM Network Adapter	Intel Pro/100 VE Network Adapter
Type (PCI, integrated)	Integrated	Integrated
Driver	Intel 9.1.30.0	Intel 6.4.14.0
Optical drive		

System	Intel® Pentium® D Processor 840 (3.2 GHz Dual-core) on Intel D955XBK	Intel® Pentium® 4 processor 2.0 GHz on Intel D845GVFN
Vendor and model number	Lite-On DVDRW SHOW-1673S (CD 48X Rd, 48X Wr) (DVD 6X Rd, 6X Wr)	Lite-On DVDRW SHOW-16735 (CD 48X Rd, 48X Wr) (DVD 6X Rd, 6X Wr)
Type (CDROM, CDRW, DVD-ROM, DVD-W)	DVD-W	DVD-W
Interface (internal, external)	Internal	Internal
USB ports		
Number of ports	5	4
Type of ports (USB1.1, USB2)	4 - 1.10 and 1 - 2.00	3 - 1.10 and 1 - 2.00
Monitor		
CRT/LCD type (plug & play, TFT, reflective)	Plug & play	Plug & play
CRT/LCD refresh rate	75Hz	75Hz

Figure 19: System configuration information for both test systems.

⁽¹⁾ These cache sizes are per core for each of the two cores.

Appendix B: Instructions for running the application scenario

This appendix explains how we manually tested and timed each of the application scenarios. Though our discussions focus on the results of the automated tests, we verified that manually performing the same functions yielded results similar to those of the automated scripts.

As the instructions below reflect, to get the most consistent possible timings and to make our hand-timed actions more like the ones the automated scripts perform, we sometimes chose to follow procedures for launching applications that were different from those typical users would follow. (See Appendix C for additional information on scripting issues.) When we made such choices, we also independently verified that the typical user procedures would still show similar results. Consequently, we are confident that the benefits dual-core processor technology delivered in these scenarios are benefits that users can expect to realize in real work situations, and are not artifacts of the measurement or scripting technology.

We ran all application scenarios and voice combinations five times on both of the systems under test, and we reported the median of those runs.

This scenario uses the following files and directories:

- Excel2MinMacro.xls, a formula-rich 2.56MB Excel file.
- Digital Home_SC1.XML, a 1.27MB XML document with pictures and graphics.
- C:\Windows\Driver Cache\i386 folder. To cut down on the length of the background VirusScan, we scan only this one 86.8MB folder. The folder contains cabinet files, applications, and system files.
- Separate high-quality recordings of a male voice and a female voice. To accommodate any potential differences between different voice types, we ran each test once with a female voice sample (www.voiceoverdirectory.com/female_voiceover_melanie_haynes.html) and once with a male voice sample (www.voiceoverdirectory.com/male_voiceover_Rick_Adamson.html) during this scenario.

The scenario requires these applications:

- Adobe Acrobat 7.0.3 Standard
- Microsoft Office Excel 2003 (Service Pack 1)
- Microsoft Office Word 2003 (Service Pack 1)
- Network Associates McAfee VirusScan Enterprise version 8.0i
- Skype Limited Skype Version 1.4.0.71

We used the following applications on the audio monitoring system:

- DigitByte Studio Super Sound Joiner*
- Yoshimasa Electronic DSSF3*

Figure 20 shows the system set-up for the conference call among four client systems running Skype. The test requires several different systems:

- The system under test.
- Three systems to serve as conference call participants. We used the other test system plus third and fourth systems not otherwise in the test. We set up the systems under test and the conference participant systems identically.
- A monitoring system. This system sends the voice through the conference call, and it also captures and monitors the audio that the other participants of the call receive.

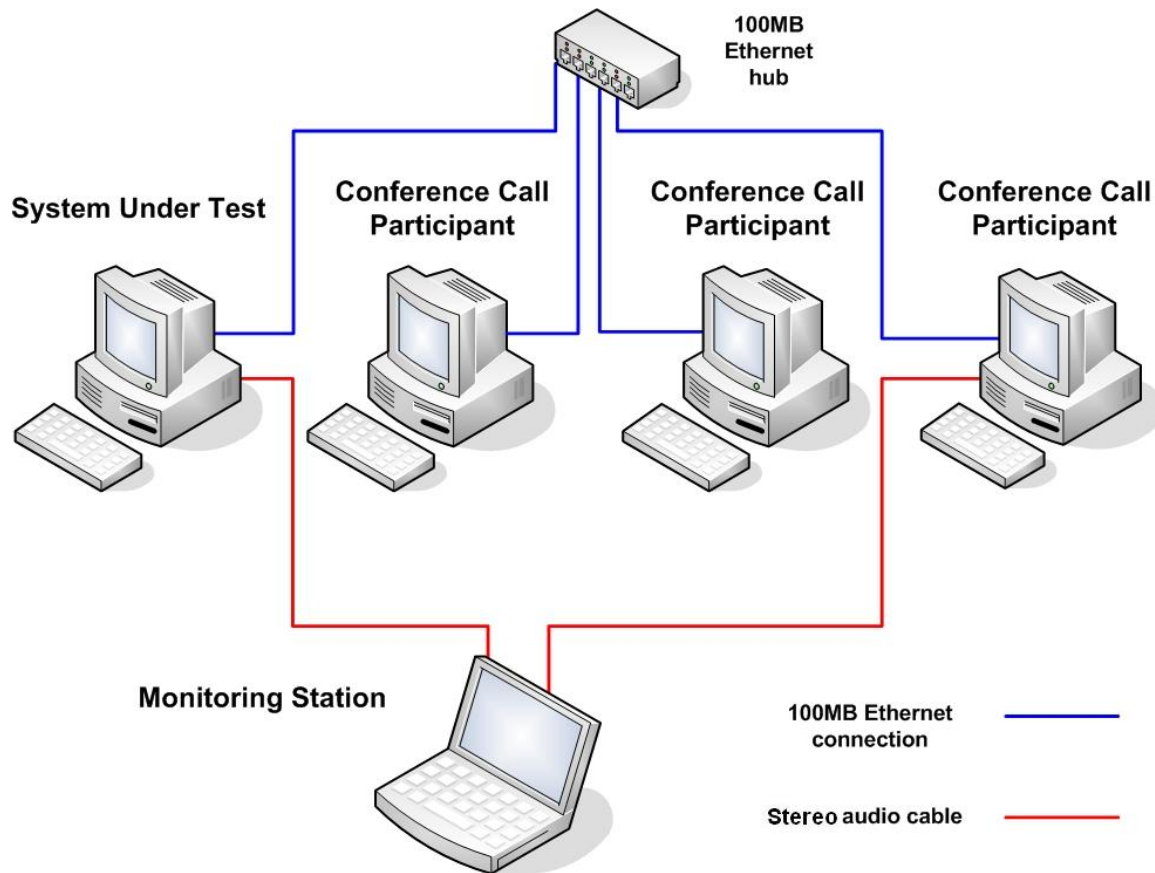


Figure 20: System configuration for the Skype conference voice call test.

We set up the monitoring system as follows.

1. Load the male and female test voice files on its hard disk. Download the male voice sample, a medical/technical narration, from www.voiceoverdirectory.com/male_voiceover_Rick_Adamson.html. Download the female voice sample, an IT/medical tutorial, from www.voiceoverdirectory.com/female_voiceover_melanie_haynes.html.
2. Use Super Sound Joiner to create longer voice sample .wav files by joining each of the above files multiple times to itself. Join the male sample seven times to create a file 11:51 long, and the female sample twelve times to create a file 11:08 long.
3. Install the DSSF3 oscilloscope program using the default options.
4. Create a desktop shortcut for the Realtime Analyzer.
5. Open Realtime Analyzer, and select the Oscilloscope button. In the Oscilloscope window, select the "1ch" Radio button at the bottom right of the screen.
6. Connect the monitoring system to the system under test with a stereo audio cable. The audio cable should run from the speaker out port of this system to the microphone input on the system under test. This enables the monitoring station to mimic someone speaking into a microphone during the Skype voice call.
7. The monitoring system will also capture and monitor the audio a participant of the call receives. To enable it to do this, connect a second stereo cable from the speaker out port of a system receiving the call to the line in port of the monitoring system.

Set up the system under test and the conference participant systems to run the Skype conference call by doing the following steps, which are necessary only once. This is the minimum setup required for the participant systems. Additional setup instructions for the system under test follow this set.

1. Set up accounts with Skype to use to connect to the Skype server. To create the accounts, go to www.skype.com and follow the account creation instructions there.
2. Install Skype, accepting all defaults.
3. After the installation, Skype prompts for user account information. Enter the accounts that you created on Skype's Web site.
4. Skype logs onto the network.
5. From the Tools drop-down menu, select Options.
6. In the Options window that appears, select the Advanced menu on the left.
7. In the Advanced option, place a check in the box next to "automatically answer call". This setting causes the clients to automatically answer the call, thus avoiding prompts to answer each call.

Set up the system under test to run the scenario by doing the following steps, which are necessary only once.

1. Set up Network Associates McAfee VirusScan Enterprise version 8.0i.
 - a. Start McAfee VirusScan Console from the Start menu.
 - b. Select Task > New On-Demand Scan Task.
 - c. Name the task DriverCache.
 - d. The VirusScan On-Demand Scan Properties – DriverCache dialog appears. If it does not, right-click on the DriverCache task name.
 - e. Click Remove twice to remove the two item names in the list box on that dialog, and click Yes on the Confirm dialog that appears after each deletion.
 - f. Click Add and scroll down to Drive or Folder.
 - g. Click Browse. On the Browse for Folder dialog, browse to the C:\Windows\Driver Cache\i386 folder.
 - h. Click OK to close the Browse for Folder dialog, and again to close the Add Scan Item dialog.
 - i. Click the Detection tab.
 - j. Check the two checkboxes under Compressed Files.
 - k. Click OK.
2. Set up Microsoft Word 2003.
 - a. Start Microsoft Office Word 2003.
 - b. Put Word into windowed mode, if necessary, by clicking the Restore Down button in the upper right.
3. Set up Microsoft Excel 2003.
 - a. Start Microsoft Office Excel 2003.
 - b. Put Excel into windowed mode, if necessary, by clicking the Restore Down button in the upper right.
4. Set up Adobe Acrobat 7.0 Standard.
 - a. Start Adobe Acrobat.
 - b. Put Acrobat into windowed mode, if necessary, by clicking the Restore Down button in the upper right.
5. Set up Skype Version 1.4.0.71.
 - a. Start Skype from its desktop icon.
 - b. Create a user account on each of four systems involved in the conference call, and enter all the required information on the account screen.
 - c. Set up each client to automatically answer calls.
 - d. Set the Away and Not Available icons to the highest option for each.
 - i. To change these settings, select Options from the Tools drop-down menu.
 - ii. In the Options window, the General Options should be visible.
 - iii. Change the Show Me as 'Away' when Inactive For and Show Me as 'Not Available' when Inactive For settings each to 999 minutes.
 - e. Select Advanced in the left-hand window.
 - f. Select Automatically Answer Call by placing a check in the box.
 - g. Select Save.
 - h. Close the Options menu.
 - i. Select Tools > Add a Contact.
 - j. Add the Skype accounts of the other three systems as contacts on each system.

- k. Exit Skype.
- 6. Organize the desktop.
 - a. Create one desktop shortcut for each of the two test data files.
 - b. Create a desktop icon for the McAfee VirusScan Console.
 - c. Size and position the Acrobat, VirusScan, Excel, and Word windows so that each takes up less than a quarter of the screen and they do not overlap each other or the desktop shortcuts you created. It is OK for them to overlap the Skype window.
 - d. Close Acrobat, VirusScan, Excel, and Word.

You will need to time multiple different tasks, so you'll either need to do multiple sets of runs to get the sets of times or use multiple people and stopwatches; the latter approach is preferable. To run each timed test, we did the following:

1. Reboot the system under test.
2. Wait ten seconds after the hourglass disappears so the system is in a consistent state.
3. Run the Skype conference call background task.
 - a. Open the Skype console window from its shortcut.
 - b. From the Skype window select Tools > Create Conference...
 - c. In the Conference Call window, select the contacts for the other three systems and click Add.
 - d. Click on Start to begin the conference call. We do not time this task because its load is constant throughout the test.
 - e. On the monitoring system, open Realtime Analyzer.
 - i. Click on the Oscilloscope and Recorder buttons along the bottom of the application window.
 - ii. In the Oscilloscope window click Start.
 - iii. In the Recorder window, check the Recording box.
 - f. Begin playing the voice sample you are testing (male or female) through Windows Media Player so the voice sample will play through the call.
4. Run the VirusScan background task.
 - a. Open the VirusScan console from its shortcut.
 - b. Select the DriverCache scan task.
 - c. Click Start. You do not need a stopwatch for this task because VirusScan reports the elapsed time for each scan.
 - d. Move the Scan Progress window to overlap the VirusScan Console and wait until the scan time is at five seconds (the time displays in the status bar at bottom of the Scan Progress window). This delay mimics the time a typical user might wait after confirming that one task has started before moving to another.
5. Run the Excel calculation foreground task.
 - a. Wait 5 seconds. This delay mimics the time a typical user might wait after confirming that one task has started before moving to another.
 - b. Open the Excel file from its shortcut.
 - c. Wait 5 seconds after Excel completes loading the spreadsheet. This delay mimics the time a typical user might wait to make sure the relatively slow-loading application is fully available.
 - d. Start the macro to run the calculation by pressing Ctrl-t, and start the stopwatch. (If you prefer not to time the task with a stopwatch, press Ctrl-a to run an alternate macro that runs the same task and shows the elapsed time on the Excel status bar.)
 - e. If you're timing the task, stop the stopwatch when the word "Done" displays on the status bar. Otherwise, the task is done when "Finished" and the elapsed time display on the status bar.
6. Run the XML open foreground task.
 - a. Wait 5 seconds. This delay mimics the time a typical user might wait after confirming that one task has finished before moving to another.
 - b. Open the XML document from its shortcut, and start a stopwatch for this task.
 - c. Stop the stopwatch for the XML open task when the file displays in Word and the page counter in the Word status bar shows 1/10.
7. Run the PDF conversion foreground task.

- a. Wait 5 seconds. This delay mimics the time a typical user might wait after confirming that one task has finished before moving to another.
 - b. Select Adobe PDF > Convert to Adobe PDF from the Word menu.
 - c. When the Save Adobe PDF File As dialog displays, select Save.
 - d. Start a stopwatch for this task.
 - e. Stop the stopwatch for the Acrobat conversion task when Adobe Acrobat opens and displays the PDF file.
8. Finish the background tasks.
 - a. Wait for the VirusScan to finish.
 - b. Get the time for the VirusScan task from the status box at the bottom of its screen.
 - c. Stop the recording on the monitoring system.
 9. Uncheck the Recording box on the monitoring system.
 10. Save the recorded .wav file.
 11. Clean up.
 - a. Exit Acrobat, Excel, Skype, Word, and VirusScan on the test system.
 - b. Delete the PDF file you created.

Appendix C: Issues in script development

To the best of our knowledge, despite its age IBM's Visual Test 6.5 remains the tool most widely used today for constructing application-based benchmarks and performance tests for PCs running various versions of Microsoft Windows. We have used this product (and previous versions of it) for many years to build performance tests. The tool does, however, have some stated limitations that unavoidably affect the way one develops performance tests with it.

First, the product's own documentation notes that its primary goal is to be a tool for automating application testing, not a benchmark development system. Consequently, the granularity of some of its functions and the way some of its functions behave are not ideal for benchmark development.

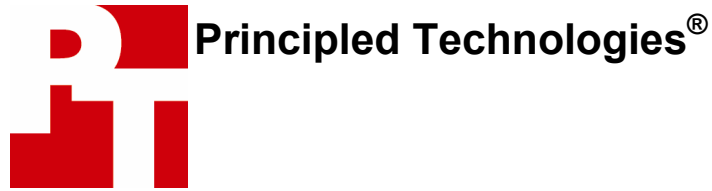
IBM also does not officially support Visual Test 6.5 for the Windows XP operating system. Because Windows XP is the leading and most current desktop version of Windows today, we nonetheless felt it was essential to use that operating system in our tests.

The presence of any scripting tool has the potential to affect the performance of a system. The tool unavoidably must, for example, occupy some memory and consume some processing power. Consequently, developing a performance-measurement script with such a tool involves maintaining a delicate balance between using the tool to automate typical real user behavior and minimizing the effects of the tool on system performance. To make sure the results of our scripts were accurate, we also hand-timed each of the functions we scripted.

To minimize these limitations and problems, we sometimes had to use scripting techniques that would achieve the same results as typical user behavior but not exactly mirror that behavior. Such techniques include inserting delays to mimic user think time and launching applications with a click on the OK button of a pre-filled Run command line. The hand timing instructions we provide in Appendix B reflect those techniques, so following those instructions should yield results similar to those the scripts produce. Whenever we had to use one of these alternative techniques, we manually verified that doing so did not materially alter the way the system behaved and that real users performing the same actions in more typical ways should see the type of dual-core processor technology benefits that we describe.

The timings the scripts produce also inevitably contain some variability. This variability is a result of the combination of the tool's limitations and the generally asynchronous nature of the many processes Windows XP and other modern operating systems have running at any given time.

Finally, though one of the goals of this effort was to produce reliable scripts, we were not trying to build bulletproof benchmarks for wide distribution and use. We developed the scripts to mimic user behavior on our specific test systems; on different systems the scripts might show different levels of benefit from a dual-core processor or even fail to work. So, although the scripts are as reliable, self-contained, and free of system dependencies as we could reasonably achieve within the project's timeframe, they do sometimes fail or encounter problems. Should a problem occur, rebooting the system and running the script again will generally yield a good result.



Principled Technologies, Inc.
4813 Emperor Blvd., Suite 100
Durham, NC 27703
www.principledtechnologies.com
info@principledtechnologies.com

From Intel Corporation:

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit <http://www.intel.com/performance/resources/limits.htm> or call (U.S.) 1-800-628-8686 or 1-916-356-3104.

From Principled Technologies, Inc.

© 2005 Principled Technologies. All rights reserved.

Principled Technologies is a registered trademark of Principled Technologies, Inc. Intel and Pentium are registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Disclaimer of Warranties; Limitation of Liability:

PRINCIPLED TECHNOLOGIES, INC. HAS MADE REASONABLE EFFORTS TO ENSURE THE ACCURACY AND VALIDITY OF ITS TESTING, HOWEVER, PRINCIPLED TECHNOLOGIES, INC. SPECIFICALLY DISCLAIMS ANY WARRANTY, EXPRESSED OR IMPLIED, RELATING TO THE TEST RESULTS AND ANALYSIS, THEIR ACCURACY, COMPLETENESS OR QUALITY, INCLUDING ANY IMPLIED WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE. ALL PERSONS OR ENTITIES RELYING ON THE RESULTS OF ANY TESTING DO SO AT THEIR OWN RISK, AND AGREE THAT PRINCIPLED TECHNOLOGIES, INC., ITS EMPLOYEES AND ITS SUBCONTRACTORS SHALL HAVE NO LIABILITY WHATSOEVER FROM ANY CLAIM OF LOSS OR DAMAGE ON ACCOUNT OF ANY ALLEGED ERROR OR DEFECT IN ANY TESTING PROCEDURE OR RESULT.

IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC. BE LIABLE FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH ITS TESTING, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC.'S LIABILITY, INCLUDING FOR DIRECT DAMAGES, EXCEED THE AMOUNTS PAID IN CONNECTION WITH PRINCIPLED TECHNOLOGIES, INC.'S TESTING. CUSTOMER'S SOLE AND EXCLUSIVE REMEDIES ARE AS SET FORTH HEREIN.