OSX Version 10.1 Swapfile Benchmarks

Preface:

With the recent release of OSX version 10.1, the majority of us have finally received the OS we've been waiting for. While the new package provides exceptional speed gains compared to previous versions, the question has once again arisen as to whether the use of a dedicated swapfile partition is beneficial. Since thereís only one way to find out, I have included benchmarks comparing OSX 10.0.4 to version 10.1. I think at this point that I can safely say, "every time I try to get out, they keep pulling me back in!" Actually, I volunteered for this one so I really shouldn't complain. In any event, here are the vital statistics.

Test Platform:

Power PC G4/466 Tangent (2001 Series)
768 Meg RAM (128-128-512 PC100/133)
OSX Version 10.1 (Build 5G64)
ATI Rage 128 Pro Video Card with 16MB VRAM
19î NEC FE 950+ Black set to Thousands of Colors
30 Gig Maxtor ATA/66 7200 RPM Internal Primary Drive
20 Gig Maxtor ATA/66 7200 RPM Internal Slave Drive

Methodology*

If you have read the previous copy of the OSX VM Swapfile Guide, youíll note that the benchmark format is mirrored for OSX version 10.1. Since I have a two-drive system with my swapfile located on the first partition of the slave drive, I used multiple arrangements to test application launch times. Included are the partition numbers for easy reference:

/dev/disk0s9: OSX System and Applications on Primary Drive
/dev/disk1s9: Dedicated 80 Meg Swap Partition on Secondary Drive
/dev/disk1s10: 10 Gig Empty Partition on Secondary Drive

Since my second partition on the slave drive was available, I installed another 10.1 system on it for these tests. Please note the following testing combinations:

Default: OSX System and swapfile located on /dev/diskOs9
Single Drive: OSX System located on /dev/disk1s10; swapfile located on /dev/disk1s9
Dual Drive: OSX System located on /dev/diskOs9; swapfile located on dev/disk1s9

The Benchmarks:

Once again, all application launches were performed in sequence following a system re-boot for 10 different programs in order to simulate an increasing processor load. None of the previous applications was quit prior to another one being launched, however I did use the command-h key combination to hide all other open windows. For the purposes of comparison, I have also included the previous benchmarks for system 10.0.4. All measurements are rounded to the nearest 1/10 of a second and were performed twice under the same conditions to minimize human error. The results are as follows:

	OSX 10.0.4			**OSX 10.1**		
	Default	**Single**	**Dual**	**Default**	**Single**	**Dual**
Internet Connect	4.3	3.4	2.9	4.9	3.6	2.8
OmniWeb	9.8	7.2	6.1	7.8	6.6	5.8
TextEdit Plus	3.1	2.9	2.1	2.1	2.1	1.8
Mail	4.6	3.4	2.9	3.3	2.9	2.1
iTunes	10.4	8.7	7.2	3.8	3.1	2.4
PhotoLine 32XL	10.8	8.8	7.3	6.2	5.3	4.2
System Preferences	4.7	3.8	3.0	2.6	2.3	1.9
Terminal	4.4	3.6	2.9	3.3	2.9	2.5
GraphicConverter 4.08	5.6	4.1	2.9	2.2	1.6	1.2
QuickTime Pro	10.4	8.2	6.9	3.1	2.8	2.3

Determination:

Whereas the placement of the swapfile on a dedicated partition made a large difference in application launch speed under 10.0.4, the revisions to 10.1 have apparently negated some of the gains (as predicted in the FAQ of the original guide). A possible explanation for this has to do with some very interesting observations I've made by running the top and vm_stat commands from within each operating system. In system 10.0.x, the swapfile was used to a greater degree with respect to pageins. Typically, my G4 would go through between eight to ten thousand pageins during boot sequence alone as libraries were transferred from the disk, via the swapfile, into core memory (RAM). As system uptime would increase, pageins would also rise. In the course of a 3-day span with no restart in between, my system would generally page in close to thirteen thousand times or more. Here's a terminal dump of the top header block running under 10.0.4 immediately after the system booted:

Processes: 29 total, 2 running, 27 sleeping... 70 threads 05:53:45
Load Avg: 0.14, 0.33, 0.30 CPU usage: 11.3% user, 13.0% sys, 75.7% idle
SharedLibs: num = 75, resident = 17.5M code, 1.16M data, 5.11M LinkEdit
MemRegions: num = 1543, resident = 29.0M + 4.64M private, 32.6M shared
PhysMem: 45.5M wired, 35.9M active, 95.4M inactive, 126M used, 662M free
VM: 781M + 40.9M 9078(0) pageins, 0(0) pageouts

In system 10.1, I've noticed that the system pages in much less frequently. I have run top or vm_stat immediately following boot sequence and the pageins only number in the low two thousand range. After running the machine for 4 days without a restart, the quantity of pageins only increased to four thousand. This indicates that some of the modifications to 10.1 include a reduced degree of dependence on the swapfile. As an example, here is a paste of my top command header block recorded under the same conditions as above (immediately after boot sequence):

Processes: 27 total, 2 running, 25 sleeping... 74 threads 17:49:55
Load Avg: 0.66, 0.30, 0.13 CPU usage: 47.7% user, 11.2% sys, 41.1% idle
SharedLibs: num = 79, resident = 19.1M code, 1.37M data, 4.89M LinkEdit
MemRegions: num = 1321, resident = 25.7M + 4.33M private, 29.9M shared
PhysMem: 45.1M wired, 41.9M active, 66.3M inactive, 133M used, 635M free
VM: 862M + 41.4M 2304(0) pageins, 0(0) pageouts

That's a HUGE reduction in the amount of work being done through the swapfile. This alone accounts for some of the performance gains that the OSX kernel development team built into version 10.1's revised VM scheme. Of interest is the level of improvement for some of the Apple native applications such as iTunes, Mail, System Preferences and QuickTime. Many of these are faster in version 10.1's default format than they were on a dual drive swap arrangement in 10.0.4. This could be due to changes in the m_alloc (memory allocation) schemes or possibly through the use of more commonly referenced libraries that alleviate the need to page in as much as information as their predecessors. Again, I'm not sure and this is pure speculation on my part.

Conclusions:

Once again, the data speaks for itself. Using a dedicated swapfile partition does provide speed gains, however due to the changes made to the structure of Aqua, the Finder and various applications, the gains are not as dramatic as they were in 10.0.4. At this time, it becomes more of an option use a swapfile partition and the decision should be based on what you, as a user, are comfortable with. I would once again strongly advise that you add as much physical RAM as possible to your machine to avoid system pageouts, which are disk based and slow the system down dramatically. If you're wondering whether I'm using a swapfile partition under 10.1, the answer is a resounding yes. If it worked well in 10.0.4, then it will continue to due so under 10.1 and any other OSX version Apple will release in the future.

Please feel free to e-mail me with questions, comments and feedback. I would once again like to thank the MacAddict forum members who never fail to support this work and contribute to it with their thoughts, ideas and probing questions. Among them are Michael Peters, Jason Robb, and fellow New Yorker John Dwight. Thanks are due also to the readers who continually provide me with feedback and suggestions from as far away as England, Belgium, Japan, Germany, France and Australia. This hack, in all of its various iterations, belongs to you.

Andy Moraitis
andymoraitis@mac.com
http://homepage.mac.com/andymoraitis

*Technical Reference Note:

A few of the software applications used in this test have since been upgraded by their respective manufacturers. As such, I was unable to test earlier versions of the software under 10.0.x and 10.1. These applications include OmniWeb, iTunes and QuickTime Pro. While I'm, not sure if the latest versions of these programs contain revisions that would markedly increase launch speed, the fact that they are revised is worth mentioning.

Sites That Host or Link to the OSX VM Swapfile Guide:

ResExcellence:	Michael Coyle's ultimate Mac Hack site (http://www.resexcellence.com)
MacOS Rumors:	Blake Patterson's informative forum and article site (http://www.macosrumors.com)
Darwin Friends:	Michael Peters' GNU/Darwin page with links to other OSX sites (http://24.5.29.77:10080/OSX-Darwin/Friends.php)