

230

SNELL & WILMER LLP
Alan L. Sullivan (3152)
Todd M. Shaughnessy (6651)
15 West South Temple
Gateway Tower West
Salt Lake City, Utah 84101-1004
Telephone: (801) 257-1900
Facsimile: (801) 257-1800

CRAVATH, SWAINE & MOORE LLP
Evan R. Chesler (admitted pro hac vice)
Thomas G. Rafferty (admitted pro hac vice)
David R. Marriott (7572)
Worldwide Plaza
825 Eighth Avenue
New York, NY 10019
(212) 474-1000

**Attorneys for Defendant/Counterclaim-Plaintiff
International Business Machines Corporation**

**IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF UTAH**

THE SCO GROUP, INC.,

Plaintiff/Counterclaim-Defendant,

-against-

INTERNATIONAL BUSINESS MACHINES

CORPORATION,

Defendant/Counterclaim-Plaintiff.

Declaration of M. Douglas McIlroy

Civil No. 2:03CV-0294 DAK

Honorable Dale A. Kimball

Magistrate Judge Brooke C. Wells

DECLARATION OF M. DOUGLAS MCILROY

I, M. Douglas McIlroy, declare as follows:

1. I am an Adjunct Professor at Dartmouth College, where I teach advanced computer science courses, including "Advanced operating systems", "Information systems", "Logic of programming", and "Programming languages".
2. I received my undergraduate degree in Engineering Physics from Cornell University in 1954 and my Ph.D. in Applied Mathematics from the Massachusetts Institute of Technology in 1959.
3. Upon completing my Ph.D. work at MIT in 1958, I joined Bell Laboratories as a Member of Technical Staff in the Mathematics Research Department. At the time, Bell Labs was the research arm of AT&T. I moved to the Computer Science Research Center of Bell Labs when it was split off from Mathematics Research in the early 1960s. From 1965 to 1986, I headed Bell Labs' Computing Techniques Research Department in which Unix originated, and thereafter served as a Distinguished Member of Technical Staff until my retirement from Bell Labs in 1997.
4. In 2004, the Usenix Association (an association of Unix users) presented me its Lifetime Achievement Award "for over fifty years of elegant contributions to Unix and programming", and also its Software Tools User Group award.
5. In the 1960s, while I was the head of Bell Labs's Computing Technologies Research Department, Bell Labs collaborated with MIT and General Electric on a project known as Multics, to create a computer operating system that would allow for the simultaneous access by multiple users to a single computer.

6. Although Multics eventually resulted in an operating system that could accommodate several simultaneous users, at that time it supported very few users and was expensive to operate, and AT&T withdrew from the project.

7. One of the Bell Labs developers who worked on Multics, Ken Thompson, later drew upon his Multics experience to design an alternative operating system. Collaborating with others at Bell Labs, including Dennis Ritchie, Mr. Thompson developed an operating system that was later called Unix.

8. In the years that followed, AT&T developed numerous versions of Unix and made it widely available to universities and businesses, as well as the United States government, under license agreements. AT&T permitted licensees, including the University of California at Berkeley ("UC Berkeley"), to develop and add their own features to Unix and to distribute those features.

9. In 1975, more than 20 institutions were represented at the first user's group conference. Usenix, as the organization is now called, is still the primary organization for Unix developers. A more buttoned-down group called Uniforum, founded in 1981, also persists. Uniforum was instrumental in initiating standards work.

10. By the end of the 1970s, Unix had grown in popularity. Universities throughout the world, including UC Berkeley, began offering educational courses and sponsoring research projects involving Unix.

11. Numerous manuals, articles and papers were written about Unix, including several editions of the "UNIX PROGRAMMER'S MANUAL", distributed by Bell Labs along with its Unix operating systems, and Lions' Commentary on Unix 6th Edition, written by John Lions.

12. AT&T licensed Unix as-is for a nominal sum, partly because of consent-decree strictures that limited AT&T's potential for becoming a commercial software vendor, and partly to foster close and friendly relations with academia. Some commercial licenses were negotiated, but they were not a noticeable moneymaker; eventually the most important purpose was to achieve portability of software across the huge range of hardware that AT&T bought from many manufacturers.

13. Beginning in 1980, Unix product development migrated from the research area of Bell Labs to AT&T's Unix System Lab ("USL"). By now, Unix expertise was ubiquitous; customers demanded Unix on most every computing platform; and Unix or Unix-like development projects sprang up outside AT&T.

14. Knowledge of how to build systems based on the Unix model is widespread. The very clarity and intelligibility of the Unix model have led most major American universities to adopt Unix as a vehicle for computing and teaching and as testbed. Furthermore, universities helped advance the operating system by developing and adding methods, concepts and code to it. UNIX's popularity in academia, and the large pool of expertise there, made Unix the leading platform from which the Internet was developed.

15. AT&T licensed Unix source code as a trade secret, but very freely. Universities were charged only a nominal sum. The mechanisms of Unix, but not the literal code, could be studied in classes. Universities could authorize anyone to access the literal source code, provided the code was shared only with others who were collaborating in the research efforts. Moreover, since AT&T did not offer technical support, system administrators required access to Unix source code in order to maintain UNIX or adapt it to their own environments.

16. A complete manual of UNIX supplied detailed descriptions of important system data layouts, particularly those used in the file system. To an experienced programmer, the short, but thorough descriptions of standard system calls (now often called the API, or application programming interface) revealed the underlying architecture of the operating system.

17. Thus, the mechanisms of Unix have always been openly available and widely known. Indeed, had that not been so, academia would probably not have adopted Unix so enthusiastically, and Unix would probably have become just another among countless bit-player operating systems.

18. John Lions began to teach an operating systems course in Sydney from the actual text of Unix, and AT&T barely objected. To protect its interest in the license, AT&T asserted rights over the class notes. Far from suppressing Lions's work, though, the company adopted it for in-house use and, I believe, made it available to licensed customers. Lions was brought temporarily to Bell Labs to advance Unix documentation. AT&T did not object to the rampant circulation of samizdat copies of Lions notes. Some years later AT&T made his book, *Lions' Commentary on UNIX 6th Edition, with Source Code* (1976), available to the public, including the text of the Sixth Edition Unix, a direct ancestor of System V. The code is still accessible online.

19. One factor that led to AT&T's liberal licensing policy and lenient enforcement was that it was constrained by consent decrees from straying far from its core business as a communications provider. With no potential for significant licensing income, Unix was most profitable in building goodwill among and gaining the respect of the technical community.

20. Subsequent to Lions's book, many books about Unix internals have been written. Among the earliest was M. J. Bach, *The Design of the UNIX Operating System* (Englewood Cliffs, NJ: Prentice-Hall, 1986), which the Unix development team wrote. The availability of books has further encouraged the study of Unix as a classroom example, and has provided entrée for many others into the mechanisms of Unix.

21. Furthermore, variant UNIX material was widely discussed and disclosed through industry associations. The Usenix association has met regularly since its founding in 1975. Its annual technical conference ranks as a signal event for operating-system gurus and aficionados. Usenix goes well beyond sharing information about coping with system administration problems or announcing new features. Usenix members openly and regularly discuss the innards of the Unix system, and technical sessions frequently examine internal mechanisms, which are essential to master in order to improve real-life performance issues.

22. My understanding is that SCO alleges that it has the right to prevent disclosure of any code appearing in a modification or derivative work of SVR4, even when the code was not written by employees of SCO or its predecessors. I further understand that SCO believes it has the right to prevent disclosure of any "methods" and "concepts" of any derivative work of SVR4, even if such method or concept does not relate directly to SVR4. This theory is completely at odds with AT&T's view of its licensing agreements.

23. If SCO's theory of disclosure were correct, it would lead to the illogical conclusion that AT&T sought to prevent disclosure of the code, methods and concepts created by its licensees who were developing their own Unix-based operating systems.

24. SCO's theory of disclosure, if accepted by AT&T and the industry at the time in the 1980s when AT&T licensed UNIX, would have meant that no UNIX licensee would have been able to develop, market or service its operating system. Consequently, no one would have subscribed to Unix a license.

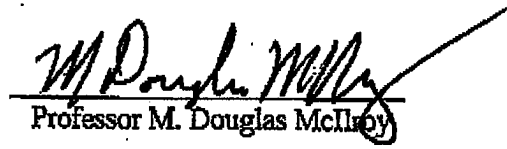
25. SCO further argues, as I understand it, that exposure to UNIX code, methods and concepts taints developers, who thereby cannot possibly independently develop other operating systems.

26. This part of SCO's theory of disclosure is, in my experience, novel in the industry (and possibly outside of the industry). If accepted, SCO's view would have startling consequences. In fact, SCO's theory of disclosure is so broad that it would effectively preclude any further development of any operating system. For example:

- Almost every computer science graduate has studied Unix.
- A large fraction of computer scientists has read various books about Unix that disclose code, methods and concepts and/or studied programs that contain either Unix code or homegrown material of a Unix derivative.

Therefore, under SCO's theory, no person who has majored in computer science or who has read any book about Unix or Unix-based code would be "clean" enough to develop code without infringing on SCO's ownership rights to Unix.

27. I declare under penalty of perjury that the foregoing is true and correct.


Professor M. Douglas McIlroy

Date: September 19 2006

Place: Hanover, NH