UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF FLORIDA

CASE NO. 11-20427-JORDAN

DISNEY ENTERPRISES, INC.,
TWENTIETH CENTURY FOX FILM CORPORATION,
UNIVERSAL CITY STUDIOS PRODUCTIONS LLLP,
COLUMBIA PICTURES INDUSTRIES, INC., and
WARNER BROS. ENTERTAINMENT INC.,

Plaintiffs,

v.

HOTFILE CORP., ANTON TITOV, and
DOES 1-10.

Defendants.
_______________________________________/

I, Ian Foster, hereby declare as follows:

1. My name is Ian Foster. I have previously submitted two declarations in this case, including a February 21, 2011 Declaration in support of Plaintiffs' Motion for Preservation Order and for Expedited Discovery and a February 28, 2011 Reply Declaration in support of the same motion. My qualifications and CV are provided in the February 21, 2011 Declaration.

2. I understand that the Plaintiffs in this litigation are seeking discovery of the "source code" for the Hotfile Website. The first purpose of this declaration is to explain what "source code" is and why it is relevant to understanding the computer system used to operate the Hotfile Website.

3. I also understand that the Defendants in this litigation have taken the position that it would be unduly burdensome to produce the Content Reference Data and Activity Data (which I have previously described in my initial declaration in this matter) for the Hotfile system. The

second purpose of this declaration is to offer my thoughts regarding the relative ease of copying and producing this data.

4. The observations and conclusions set forth below are based on my own observation and use of the live Hotfile site, as informed by my specialized knowledge, education, and expertise as applied to the facts and circumstances in this case.

BACKGROUND ON SOURCE CODE

5. Computers operate by executing sequences of instructions contained in computer programs. The "source code" for a computer program is a representation of those instructions in a human-readable form. Given the source code for any computer program, one can typically determine, in great detail, and with certainty, exactly what the program is telling the computer to do and therefore how the computer program works.

6. Source code can also include, in addition to executable instructions, "comments," which are text provided by the programmer that is not interpreted by the computer, but can provide explanations of why a particular approach was taken by the programmer to implementing a particular task.

7. Source code is useful for understanding computer programs because it provides a precise and objective representation of what a computer program does. Although it is possible to describe a computer program using means other than source code, such as documentation or narrative descriptions by the author(s) of the program, any such description of a computer program would always be a human interpretation of the program, and therefore subject to potential omissions, inaccuracies, and ambiguities. The source code itself, on the other hand, *is* the computer program – nothing more and nothing less.

IMPORTANCE OF SOURCE CODE IN THIS CASE

8. My previous declarations have been based in part on external observations of the Hotfile system, combined with statements that the Defendants have made about how the system works. However, the Defendants' own public statements may be incomplete, incorrect, or ambiguous, and external observation and use of the system will not permit analysis of functions that are invisible to the user. Access to source code would permit one to determine, without uncertainty or ambiguity, how the Hotfile system works, and how it performs the functions described on the Hotfile Website. It will also allow one to resolve any uncertainties regarding the details of its operations, and to verify the correctness of any statements made about the operation of the Hotfile system by Hotfile in its own online materials, website, and declarations. In addition, access to Hotfile's source code would also allow for clarification of details regarding the operation of the Hotfile System that are not clear from the materials provided.

9. Access to Hotfile's source code would allow for a precise understanding of the details of various features of Hotfile's features and operations that may not be apparent from external observation. For instance, consider the Hotfile feature that allows for the creation of multiple URLs for the same Content File. From external observation, it is not apparent whether this feature involves the creation of multiple URLs that each resolve to the same copy of the Content File, or whether it involves the creation of multiple copies of the same Content File. Access to the source code would unambiguously answer this question. Likewise, it is not apparent from external observation which steps Hotfile takes when it responds to a takedown notice that names one of multiple URLs created using this feature – whether it removes only the identified link (leaving other copies of the file or links to the file active) or removes others as well. In addition, when responding to notices of infringement more broadly, it is not apparent

from external observation alone whether Hotfile removes the underlying Content File(s) or merely disables the URL(s) that direct to those files, while retaining copies. Analysis of the source code would clearly answer these types of questions.

10. Hotfile's source code would also clearly show the circumstances under which copies of Content Files are made by the Hotfile system and the events or instructions that trigger the creation of such copies. It is not clear from external observation whether the Hotfile system retains and uses a single copy of a Content File uploaded by the uploading user, or whether it also creates additional copies under certain circumstances, and, if so, what those circumstances are (e.g. whether the system makes additional copies of all files, or only some subset). The source code for Hotfile's system would readily demonstrate the circumstances under which the system makes copies of Content Files and how the system uses those copies. It would make clear what "triggers" the creation of any additional copies of each Content File – for instance, whether Hotfile automatically creates backup copies of Content Files to guard against loss, or whether it creates copies of only a subset of Content Files in response to high download demand for those files. It is not uncommon for computer systems that distribute large numbers of electronic files to create additional copies of the files in order to facilitate the distribution. The source code for Hotfile's system would readily answer these questions with certainty.

11. Hotfile's source code would also help elucidate what information Hotfile maintains about the Content Files on the Hotfile system, how Hotfile utilizes such information in its everyday operations, and what abilities Hotfile has to search or query information regarding those files. I understand that the Defendants in this action have made various claims regarding their inability to monitor the nature of the Content Files hosted on their system, as well as claimed that they have a "filtering" system. Access to Hotfile's source code would show how

files are organized on Hotfile's system, how Hotfile uses that information, and precisely what steps Hotfile takes (or declines to take) to "filter" files designated for removal or blocking. This information, in turn, would show the extent to which Hotfile's system permits searching for particular content, what steps it takes to remove or block content designated for removal or blocking (or has declined to take), and how readily such functions could be implemented if they do not exist already.

12. Source code can also show the design choices and design history of a computer system. It is good engineering practice to maintain source code in source code management systems, and I therefore believe that it is likely that Defendants use such a source code management system for the Hotfile Website. A source code management system maintains the history of changes to the source code for a computer system. Therefore, access to Hotfile's source code would show when particular features were introduced and any changes that Hotfile made to its system over time.

SOURCE CODE IS THE BEST EVIDENCE OF HOW HOTFILE'S SYSTEMS WORK

13. I understand that Defendants in this Action have suggested that Plaintiffs obtain information about how Hotfile's system works from sources other than source code, such as taking the testimony of Hotfile's engineers. While other sources may be useful, they are ultimately only a proxy for the ultimate truth captured in the source code. Even with the best intentions, testimony by an engineer about the workings of a computer program may be incorrect or ambiguous, and is likely to be incomplete. Moreover, there is also the possibility that an engineer could deliberately misrepresent or mischaracterize how a computer program works. Some features of a computer program, for instance, might be "hidden" from external observation

and a person describing the program could simply decline to identify the feature. Because the source code *is* the computer program, it is not subject to these potential errors.

COMPETITIVE SENSITITY OF HOTFILE'S SOURCE CODE

14. I also understand that Defendants in this Action have taken the position that Hotfile's source code represents a trade secret that must be kept confidential. While I do not discount the possibility that some elements of Hotfile's source code may be competitively sensitive, it is important that such claims not be exaggerated.

15. There are numerous companies that operate online content hosting and distribution services, and several do so on a scale similar to Hotfile. The methods required to organize large numbers of Content Files and deliver them for download, therefore, are widely implemented. In general, from external observation the capabilities of the Hotfile system appear relatively straightforward and it appears unlikely that any particular innovations are necessary to implement them. Thus, while it may well be that there are some minor innovations in how the Hotfile system is implemented, these innovations are unlikely to have particularly substantial competitive value.

16. By analogy, the Hotfile system can be analogized to an automobile that is well-built but still uses a conventional combustion engine. It appears to have been built by combining existing ideas, probably with some minor innovations, but without involving any radically new approaches. To continue the automotive analogy, its value derives from good engineering overall, and not from an entirely new type of engine. Thus, while examining the design could be instructive to a competitor, it would be unlikely to provide any entirely new ideas.

LOGISTICS OF COPYING HOTFILE'S SOURCE CODE

17. Source code, by its very nature of being written by humans, is not enormous in scale. Based on my review of the Hotfile system and its relative complexity, I believe that the

source code for the Hotfile system is likely to be no larger than a few hundred megabytes and therefore fit easily on a single USB key drive, which commonly have capacities of several gigabytes (i.e. thousands of megabytes). In addition, source code is generally maintained at a single location, or very small number of locations, from which it can be readily copied. The use of a source code control system, which I believe it is likely that Hotfile uses, makes it a trivial task to obtain a copy of the current source code for the Hotfile system, as well as past/historical versions of the same.

LOGISTICS OF COPYING HOTFILE'S CONTENT REFERENCE AND ACTIVITY DATA

18. I understand that Defendants have taken the position in this litigation that it would be difficult to create and produce a copy of the Content Reference Data for the Hotfile System (as defined in my previous declarations), as well as server logs indicating uploading and downloading activity. I offer the Court my following observations regarding the relative ease of copying and producing such data.

19. I understand that Hotfile uses a MySQL database (a common database management system) to maintain its content reference data. This database system makes it straightforward to create a copy of the database's contents by using a "mysqldump" command, which is also routinely used to create backups of databases for administrative purposes or business continuity purposes.

20. I would expect the size of this database to be relatively manageable. Based on the types of data about each Content File that I expect Hotfile to maintain as described in my previous declarations, I would expect the Reference Data for each Content File to be no larger than a kilobyte. Therefore, if Hotfile had one hundred million files on its system, the Content Reference Database would be no larger than 100 Gigabytes (Hotfile appears to use a consecutive

numbering system, from which it appears that approximately a hundred million files have been uploaded to the system at some point in the site's history). By contrast, one can readily purchase consumer-grade hard drives with capacities as high as three terabytes at consumer electronic stores (such as Best Buy) for around one hundred and fifty dollars. Using a consumer-grade connection, such as USB 2.0, one hundred gigabytes of data should take only about forty minutes to transfer, and even a full terabyte of data should take no longer than seven hours. I would expect Hotfile to have access to commercial-grade connections that could accomplish such transfers much faster than consumer-grade connections.

21. Logging data reflecting uploading and downloading activity should be similarly manageable. A typical server log entry contains information such as the time of the request, the IP address of the request,[1] the nature of the request, the requested URL, and the name of the object to which the request refers. Even if Hotfile were logging some additional information in addition to these types of typical fields, I would not expect each log entry reflecting an upload or download to be any larger than two kilobytes. Based on publicly available reports regarding web traffic, Hotfile appears to receive roughly one hundred million pageviews per month (without access to Hotfile's own web traffic data, this figure is approximate, as there is some variation from month to month and among publicly available reports). Even if one half of those pageviews corresponded to uploads or downloads, then that is just fifty million uploads and downloads per month. Thus, the total logging data is unlikely to be any more than one hundred gigabytes per month, and I would not expect the total size of the data since mid-February 2011 to be any greater than 500 gigabytes (half a terabyte). In generating these estimates, moreover, I have used very conservative assumptions, and would expect the actual numbers to be yet smaller.

[1] A computer's IP address can generally be used to identify the geographic location of the computer.

22. In the context of Hotfile's regular daily operations, copying one or even two terabytes of data (which can readily fit on a consumer-grade hard drive) should not represent a substantial effort. Based on publicly available data regarding Hotfile's web traffic, the site receives around 100 million pageviews per month, or about three million pageviews per day. Assuming very conservatively that only one in ten pageviews represents a download, around three hundred thousand files are being downloaded from Hotfile every day. Assuming conservatively that the average size of a file downloaded from Hotfile is around twenty megabytes (which assumes a mix of audio, video, and other kinds of files), that would still represent around six terabytes of data downloaded from the site on a daily basis. Without access to Hotfile's logging or content reference data these are, by necessity, very rough estimates. However, it is clear from the general scope of its operations that copying one or two terabytes of data, for a site like Hotfile, should not represent an extraordinary effort.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

Executed on this 31st day of May 2011, at Chicago, IL

Ian Foster