EXHIBIT C-7

Microsoft contends that the asserted claims of the '132 Patent are invalid as obvious by Callaghan, "NFS Illustrated" (Addison-Wesley Ed. 2000) (Callaghan), Rodriguez-Martinez, "Automatic Deployment of Application-Specific Metadata and Code in MOCHA" ("Rodriguez-Martinez"), and Lomb, "Storage Management Solutions for Distributed Computing Environments" ("Lomb") prior art references under various subsections of 35 U.S.C. § 102 in view of other prior art references under 35 U.S.C. § 103 as set forth in Microsoft's invalidity contentions.

As Callaghan was published as early as 2000 and no later than 2001, Microsoft contends that it is prior art to the '132 Patent under at least pre-AIA 35 U.S.C. § 102(b).

As Rodriguez-Martinez was published in 1999, Microsoft contends that it is prior art to the '132 Patent under at least pre-AIA 35 U.S.C. § 102(b).

As Lomb was published in 1996, Microsoft contends that it is prior art to the '132 Patent under at least pre-AIA 35 U.S.C. § 102(b).

Patent No. 8,671,132 Claim Limitation(s)	**Disclosures**
"wherein the method further comprises translating the one or more attributes" (Claim element 22[a])	**Callaghan, "NFS Illustrated" (Addison-Wesley Ed. 2000) ("Callaghan")** Callaghan discloses translating the one or more attributes. For example, Callaghan states: "The NFS protocol allows files to be named by a sequence of names that make up a path. The protocol is careful not to require that pathnames be supported as entities within the protocol itself. A pathname is evaluated with a sequence of LOOKUP requests. Component-by-component evaluation make it unnecessary for the protocol to reserve a character to separate the components in a pathname. It is fortunate for PC-UNIX interoperability because UNIX uses a forward slash separator, a/b/c, whereas PC clients use a backslash, a\b\c. UNIX servers are *case-sensitive* and *case-preserving*. This means that a UNIX server sees ABC123 and abc123 as two different filenames – it is sensitive to differences in case. When a name is assigned to a new file, the UNIX server will preserve the cases of the characters in the filename; it will not map to uppercase or lowercase. DOS clients are case-insensitive and are not case-preserving. This means that a DOS client cannot distinguish the name ABC123 from abc123. If the file Abc123 is created, instead of preserving the cases, it will map all characters to uppercase—ABC123. A Windows client is *case-insensitive* and *case-preserving*. Like a DOS client, it cannot tell the difference between

Patent No. 8,671,132 Claim Limitation(s)	Disclosures
	ABC123 and abc123, but it will preserve the cases in a filename—it will not map lowercase to uppercase. The names within a UNIX or Windows path can be any sequence of up to 255 characters excluding the separator slash. DOS clients are restricted in their choice of names to an '8.3' format: the name is limited to 8 characters plus a dot and an extension of up to 3 characters. In addition, the characters .,+[]*?:\/;=<> are ruled illegal. These restrictions can create problems for DOS clients that access files on a UNIX or Windows server. How can long names be represented in 8.3 format and what can be done about the characters that are legal for UNIX but illegal for DOS or Windows? The DOS client uses an algorithm to map UNIX names to DOS equivalents, as follows:

Patent No. 8,671,132 Claim Limitation(s)	Disclosures

1. The special names . and .. are not mapped.
2. Lowercase characters are mapped to uppercase characters and uppercase to lowercase: `Abc123.txt` becomes `aBC123.TXT`.
3. If the name is now a legal DOS name and has no lowercase characters, then no further mapping is required.
4. If the NFS name has a legal DOS extension (up to 3 characters, none illegal), the extension is preserved.
5. The NFS name is truncated to leave just the first 5 characters, and illegal DOS characters are mapped to tilde (~). If the name is shorter than 5 characters, it is extended to a full 5 characters by the addition of tildes.
6. The 5-character name is extended to 8 characters by adding a tilde and two legal DOS characters that are chosen to make the name unique within the directory (shown as *XX* in Table 14.1).
7. If the NFS name had a valid DOS extension, it is appended following a dot.

TABLE 14.1 Mapping Table

NFS name	*DOS mapped name*	*Rules*
abc123.txt	ABC123.TXT	2, 3
lengthyname	LENGT~*XX*	2, 5
CORE	CORE	2, 3
core	CORE~~*XX*	2, 5, 6
.cshrc	~CSHR~*XX*	2, 5, 6
whitepaper.ps	WHITE~*XX*.PS	2, 4, 5, 6, 7
index.html	INDEX~*XX*	2, 5, 6

The making of a unique name in step 6 through addition of XX characters presents an interesting challenge. The two uniqueness characters must not be randomly chosen because the name must persist from one day to the next, even if the client is rebooted and the mapping table is lost. The Beame and Whiteside (now Hummingbird, Inc.) PCNFS client solved the problem neatly by using the READDIR cookie value associated with the directory entry as the basis for generating the XX characters.

The DOS-mapped name and the NFS name are stored in a mapping table. If a user types a DOS-mapped name, the client will translate it to the NFS name via the mapping table before sending a LOOKUP request to the server." Callaghan at 379-381.

Patent No. 8,671,132 Claim Limitation(s)	Disclosures

"**14.2 File Attributes**. DOS supports only a small set of file attributes compared with the POSIX set of 13 or so file attributes provided in the NFS fattr structure returned by the GETATTR request (Table 14.2). Some of the NFS attributes like file size and mtime have DOS equivalents. Others have no mapping or an indirect mapping."

TABLE 14.2 DOS File Attributes

DOS file attribute	*NFS mapping*
File size	`size`
Modification time	`mtime`
Read-only	Set if the write bit in the `mode` attribute is not set. The client needs to determine whether the user, group, or other write bit applies, depending on the file's owner and group. Server ACLs may make this determination unreliable. Reliable read-only indication can be returned by the NFS version 3 ACCESS procedure.
Hidden	If set, it indicates whether the file will appear in directory listings. The attribute is set if the NFS name begins with a dot, e.g., `.login`.
Archive	Used by DOS utilities to determine whether a file needs to be archived. The attribute is cleared (set to 0) when the file is archived and set back to 1 if the file is modified. Since there is no UNIX equivalent, it is set to 1 for all NFS files.
Directory	Set if the file `mode` attribute indicates that the file is a directory.
System	No NFS equivalent—not set.

Callaghan at 381-382.

Rodriguez-Martinez, "Automatic Deployment of Application-Specific Metadata and Code in MOCHA" ("Rodriguez-Martinez")

Rodriguez-Martinez discloses translating the one or more attributes. For example, Rodriguez-Martinez states:

Patent No. 8,671,132 Claim Limitation(s)	Disclosures
	"Database middleware systems, such as database gateways and mediator systems, are used to integrate heterogeneous data sources dispersed over a computer network. In order to achieve data integration, the middleware layer imposes a global data schema on top of the individual schema used by each source. Through this mechanism, the client applications been serviced by the middleware system are provided with a uniform view and uniform access interface to the data sets stored by each data source. The translation of the data items to the global schema is performed by either a wrapper or database gateway. Wrappers are used when integration is achieved through a mediator system, such as TSIMMIS [CGMH +94], DISCO [TRV96] or Garlic [RS97]. On the other hand, gateways are used when integration is realized by importing the data into a commercial DBMS, such as Oracle [Cor99] or Informix [Cor97]. Typically, these applications use a connectivity API such as ODBC or JDBC to extract the data from the sources. The wrapper or gateway can either be run on a machine near the data source (e.g. on the same Local Area Network) or at the site where the integration server runs." Rodriguez-Martinez at 1. "In order to access the wealth of information stored in a particular data source, the QPC connects to the Data Access Provider (DAP) associated with the source. The DAP is a server application which extracts data from a source on behalf of the QPC. For each data source, there is at least one DAP, and each DAP in the system can be located by QPC through a URL. There are two essential services provided by a DAP: a) data translation, and b) query execution. The DAP extracts requested items from the data source, and translates them from the local schema used by the source into the global schema used by QPC. Also, the DAP is capable of executing query operators that generate new abstractions from the data. In particular, the DAP is designed to execute those operators that filter out the data sets (e.g. a predicate) to produce smaller values. For this reason, the DAP should be run at the data source site or in close proximity to it (e.g. on another host in the same LAN). The QPC delivers all the code for the data types and operators used by each DAP. Similarly, all results produced by each DAP are sent to QPC for further processing until the final answer to the query is fabricated." Rodriguez-Martinez at 5. "The first resources that must be made available to MOCHA are the tables to be used by the application. For each table, metadata indicating its name, the database in which it is stored, the columns names and the middleware types needed to represent each column must be added to the

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.