MARK A. LAUER (Bar No. 163756)
THOMAS W. LATHRAM (Bar No. 59639)
T. LESTER WALLACE (Bar No. 159967)
SILICON EDGE LAW GROUP, LLP
6601 Koll Center Parkway
Suite 245
Pleasanton, California 94566
Telephone: 925-621-2110
Facsimile: 925-621-2119

Attorneys for Plaintiff
Alacritech, Inc.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

ALACRITECH, INC.,	)	Case No.: C04-03284 JSW
	)	
Plaintiff,	)	**DECLARATION OF DR. KEVIN ALMEROTH IN SUPPORT OF ALACRITECH'S REPLY TO MICROSOFT'S OPPOSITION TO ALACRITECH'S MOTION FOR PRELIMINARY INJUNCTION**
v.	)	
MICROSOFT CORPORATION,	)	
Defendant.	)	
	)	Before the Honorable Jeffrey S. White Hearing Date: March 25, 2005 Hearing Time: 9:00 a.m. Place: Courtroom 2, 17th Floor

I, Dr. Kevin Almeroth, declare and state as follows:

Materials Reviewed

In addition to the materials reviewed as set forth in my prior declaration (Declaration Of Dr. Kevin Almeroth In Support Of Alacritech's Motion For Preliminary Injunction of

Microsoft's Infringement Of Claim 1 of U.S. Patent No. 6,697,868"), I have reviewed documents including the following:

a. Second Revised Exhibit A To Joint Claim Construction And Prehearing Statement Re Preliminary Injunction (the "Joint Claim Construction Statement").

b. Defendant/Counterclaimant Microsoft Corporation's Opposition To Alacritech's Motion For Preliminary Injunction ("Microsoft's Opposition").

c. Declaration of James Pinkerton In Support Of Microsoft's Opposition To Alacritech's Motion For Preliminary Injunction ("Chesson Declaration").

d. Declaration of Dr. Gregory L. Chesson In Support Of Microsoft's Opposition To Alacritech's Motion For Preliminary Injunction ("Chesson Declaration").

e. Exhibits A-Q to the Chesson Declaration.

f. The "Microsoft Press Computer Dictionary", third edition, pg. 256, copyright 1997, pages 254 - 256, 293, 307, 383 (Exhibit A to this Declaration).

g. U.S. Patent No. 6,141,705 to Anand, et al. (Exhibit B to this Declaration).

h. U.S. Patent No. 5,058,110 to Beach et al. (Exhibit C to this Declaration).

i. U.S. Patent No. 6,034,963 to Minami et al. (Exhibit D to this Declaration).

Construction of Claim 1 of the '868 Patent

1. I have reviewed Alacritech's proposed claim construction that is set forth in the Joint Claim Construction Statement and find that construction to be reasonable. In particular, I note that:

A. <u>Instructions</u>

2. The term "instruction" (an "instruction" of the type that is "executable on a processor") has a well-understood meaning in the electrical engineering and computer science arts. The '868 patent discloses a microprocessor as one example of a "processor," and more particularly identifies a Pentium as an example of a microprocessor. A Pentium is,

however, just one example of a microprocessor. There are many other examples of microprocessors. Each such microprocessor has its own "instruction set." The "instruction set" of a microprocessor defines at a very detailed level each particular "instruction" that the microprocessor can execute and what the microprocessor will do when it executes that instruction. An "instruction" of the type that can be executed on a processor as the term "instruction" is used in Claim 1 of the '868 patent therefore has a very clear and well-understood meaning to one of ordinary skill in the art. One of ordinary skill in the art, given the type of microprocessor, would know to consult the "instruction set" of the microprocessor. From that instruction set, one of ordinary skill would be able to explain in detail the exact form and effect of the "instructions" that are executable by the microprocessor.

3. The "set of instructions" of Claim 1 of the '868 patent is "executable on a processor." If a "set of instructions" is "executable on a processor," then the processor must have read the instructions or received the instructions from a processor-readable medium. It is impossible for a processor to "execute" an instruction if the instruction only exists in the abstract. The instruction has to be stored somewhere in order for the processor to be able to obtain it, decode it, and execute it. That place where the instruction is stored must be a real-world, processor-readable, tangible object, otherwise the processor would not be able to obtain the instruction. The "set of instructions" of Claim 1 of the '868 patent that is "executable on a processor" therefore is necessarily stored in a tangible media.

B. A TCP Connection

4. Microsoft proposes construing terms of Claim 1 in a manner that contradicts the evidence and leads to confusion. For example, Microsoft proposes construing "a TCP connection" to mean: "A logical communication path identified by a pair of sockets pursuant to the Transmission Control Protocol." Such a logical path would extend across a network and into a computer at both endpoints, each endpoint identified by an Internet Protocol (IP)

address that specifies the computer's location on the network and a TCP port within that computer. Such a logical path could not be offloaded from a processor to an intelligent TCP offload mechanism, as recited in Claim 1, because for that connection the endpoints would not change and so the logical path would not change. Moreover, a logical path would not involve a processor or an offload mechanism, because a logical path would not include specific physical devices such as a particular network, router, processor or offload mechanism. On the other hand, I find that Alacritech's construction of a TCP connection as a combination of information to be reasonable, because that body of information can be offloaded, i.e., transferred.

5. Such a logical communication path would also seem to describe User Datagram Protocol (UDP), which is connectionless. A UDP port, like a TCP port, is simply a number for an application that is using UDP or TCP, respectively. Microsoft's proposed construction of a logical path identified by endpoints thus ignores the main differentiation between TCP and UDP, that TCP guarantees reliable transfer of data by establishing a connection, i.e., a body of information that a processing mechanism uses to manage communication between applications over a network.

6. Another problem with Microsoft's proposed construction is that the term "sockets" has several meanings in the network communications world. "Sockets" or "Berkeley Sockets" is an application programming interface (API) for applications that use various network protocol services, such as TCP/IP, instead of being part of the TCP/IP protocol. I believe that Microsoft's proposed construction of TCP connection to include the word "sockets" leads to confusion rather than clarity. Even Microsoft's expert Dr. Chesson appears confused by these multiple meanings in his declaration, referring in ¶ ¶ 47, 48, 58, 62 and his claim chart in §6 of his Declaration to the functions socket() and connect(), which do not refer to IP addresses and TCP ports but instead refer to the sockets API that runs above TCP.

7. Moreover, Microsoft's proposed construction would render the term following

"a TCP connection" superfluous, as it is simply a restatement of "identified by a pair of sockets." In contrast, Alacritech's proposed construction of "a TCP connection" is supported by RFC 793 as well as the intrinsic evidence, as demonstrated by Alacritech's citations in the Second Revised Joint Claim Construction Chart.

C. Establishing a TCP Connection

8. Similarly, Microsoft's proposed construction of "establishing a TCP connection" points to RFC 793 for support, but the pages it cites from the RFC do not contain the words "establish" or "establishing," despite multiple references to establishing a TCP connection elsewhere in the same document. See, e.g., RFC 793, pages 4, 10, 11, 12, 21, 27, 30 and 31.

D. Offloading a TCP Connection from the Processor

9. Moreover, Microsoft's proposes to construe "offloading the TCP connection from the processor" to mean "allocating processing for the TCP connection from the processor," substituting "***processing*** for the TCP connection" in place of "***the TCP connection***." Neither intrinsic nor extrinsic evidence supports such a rewriting of this phrase. The disclosure cited by Alacritech in the Joint Claim Construction supports Alacritech's proposed construction of offloading a TCP connection.

10. Microsoft's construction also fails to consider that offloading of other TCP processing was known at both the filing date and issue date of the '868 patent. For example, U.S. Patent No. 6,141,705 to Anand, et al., which was considered during prosecution of the '868 patent, and discloses a peripheral hardware device and its driver that together can offload other TCP processing, states: "For instance, many NICs are capable of independently performing tasks otherwise performed by the CPU in software at an appropriate network layer, such as checksum calculation/verification; data encryption/decryption; message digest calculation; TCP segmentation; and others." U.S. Patent No. 6,141,705, column 2, lines 44-

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.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**