IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re *inter partes* review of: U.S. Patent 7,093,086 to Hans van Rietschote.	Case Nos. IPR2013-00150
Filed: February 14, 2013	Atty. Docket: 2907.020IPR0
For: **Disaster Recovery and Backup using Virtual Machines**	

Declaration of Dr. Prashant Shenoy in Support of Petitioner's Reply

I, Prashant Shenoy, declare as follows:

1. I have been retained by Sterne, Kessler, Goldstein, and Fox PLLC on behalf of Veeam Software Corporation ("Veeam") for the above-captioned *inter partes* review proceedings. I understand that these proceedings involve U.S. Patent No. 7,093,086 ("the '086 Patent") entitled "Disaster Recovery and Backup using Virtual Machines," and that the '086 Patent is currently assigned to Symantec Corporation.

2. An updated version of my *Curriculum Vitae* is attached as Appendix A to this Declaration, which contains further details on my education, experience, publications, and other qualifications to render an expert opinion. My work on this

declaration is being billed at a rate of $435.00 per hour, with reimbursement for actual expenses. My compensation is not contingent upon the outcome of this inter partes review.

3. I understand that the Board instituted *inter partes* review of claims 1, 11, 12, and 22 of the ’086 Patent on 5 separate grounds. I have reviewed, and I am familiar with all of the prior art supporting those grounds, as well as the Board’s Decision on Institution. The grounds of rejection instituted by the Board include the following:

4. Claims 1, 11, 12, and 22 are anticipated by Lim et al (“Lim”) (provided at VEEAM 1004).

5. Claims 1, 11, 12, and 22 are anticipated by the “VMware ESX Server: User’s Manual” (“ESX”) (provided at VEEAM 1005).

6. Claims 1, 11, 12, and 22 are anticipated by “Getting Started Guide: VMware 2.0 for Linux,” (“GSG”) (provided at VEEAM 1006).

7. Claims 1, 11, 12, and 22 are anticipated “Checkpoint for Network Transferable Computer” by Suzaki (“Suzaki”) (English translation provided at VEEAM 1008).

8. Claims 11 and 22 are obvious over Suzaki in view of "Integrating Checkpointing and Transaction Processing," by Wang ("Wang") (provided at VEEAM 1010).

9. I also understand that the Patent Owner has submitted a response ("Response") in opposition to the petition filed in February 2013. I have reviewed the Response as well as Dr. Green's (Patent Owner's expert) declaration in support of the Response, including substitute exhibit C ("Green Dec.") and his deposition transcript ("Green Tr."). I have been asked to provide my technical review, analysis, and insight regarding the Response and corresponding opinions of Dr. Green.

The Capture of State While a Virtual Machine is Executing

10. I understand that both Patent Owner and Dr. Green contend that claims 1 and 12 require the virtual machine to be executing during the capturing step. (Response, p. 12; *see also* Green Dec., ¶¶ 47-49.) I also understand that the Dr. Green and Patent Owner further contend that state must include both the contents of memory, hardware state (including the processor state), and configuration information. (Response, p. 27; *see also* Green Dec., ¶ 40.) I don't agree that the broadest reasonable interpretation of claims 1 and 12 requires that

the captured state include the contents of the memory, hardware state (including processor state), and configuration information because the specification only describes that state "may" include these items, and the plain language of the claims only requires the state to include "at least one file." ('086 patent, 4:28-33; *see also* claims 1 and 12.) However, if the Board were to adopt these positions, it is my opinion that the virtual machine still must be suspended to capture state, at least temporarily, even for the embodiments in the '086 patent that involve capturing state while the virtual machine is executing.

11. Processors typically include a number of memory storage areas known as registers. (*See* Green Tr., 237:6-10.) The registers are responsible for holding data used for processing instructions on the processors, among other things. Thus, to capture state of a processor one must also capture the state of the registers. As Dr. Green explained during his cross-examination, processors have more than one register, and the registers are constantly being changed based on the current instruction or instructions that are being executed by the processor. (Green Tr., 254:13-18.) It naturally follows that to capture state one must stop, at least momentarily, a register from being updated while it is being "captured," or else the data could be inconsistent.

12. Instructions typically require at least one register in the processor for execution, and typically require many. Thus, if an instruction is being executed on the processor, one would have to capture numerous registers at once, or possibly all registers to ensure consistency of the processor state because it is very difficult to determine which registers are currently being used on the processor. This takes time, and in the meantime, the processor (i.e. virtual processor) would have to be suspended from execution until all the registers are copied.

13. During his cross-examination Dr. Green explained that one could capture the processor registers using similar techniques to those that the ’086 patent describes for capturing memory in some of its embodiments. (Green Tr., 255:3-17.) In other words, one could create a separate area to hold new updates to the registers. But, creating such an area also takes time, and in the meantime, updates must be suspended to the processor until the new “area” is created or an inconsistent state is created on the processor. Further even if such an area were preallocated (i.e. created before the capture began), the process of instructing the virtual machine to redirect its updates to the new area also takes a period of time, during which the virtual machine could not execute.

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.