United States Patent and Trademark Office

Public Hearing on Use of the Patent System to Protect Software-Related Inventions

Transcript of Proceedings

Wednesday, January 26, 1994
Thursday, January 27, 1994
9:00 a.m. to 5:00 p.m.

Before

Bruce A. Lehman
Assistant Secretary of Commerce and
Commissioner of Patents and Trademarks

Location:
San Jose Convention Center
408 Almaden Avenue
San Jose, California

Table of Participants

Before: Bruce A. Lehman
               Assistant Secretary of Commerce and
               Commissioner of Patents and Trademarks
               United States Patent and Trademark Office

The Panel: Ginger Lew
               General Counsel-Designate
               United States Department of Commerce

               Lawrence Goffney
               Assistant Commissioner for Patents-Designate
               United States Patent and Trademark Office

               Micheal K. Kirk
               Assistant Commissioner for External Affairs
               United States Patent and Trademark Office

               Jeffrey P. Kushan
               Attorney-Advisor
               United States Patent and Trademark Office

Recording
Technicians: Karl Henderscheid
               Support Office Services
               52 Second Street, Third Floor
               San Francisco, CA 94104
               (415) 391-4578

Trascriber: Milton Hare
               Rogershare Transcribers
               541 Maud Avenue
               San Leandro, CA 94577
               (510) 357-8220

UNITED STATES PATENT AND TRADEMARK OFFICE
Public Hearing on Patent Protection for Software-Related Inventions
San Jose, California -- January 26-27, 1994

Witnesses

January 26, 1994

Mr. Clark
VideoDiscovery

Mr. Poppa
StorageTek

Mr. Ryan
Intellectual Property Owners, Inc.

Mr. LeFaivre
Apple Computer
Computer and Business Equipment Manufacturing
Association

Mr. Lopez
Interactive Multimedia Association

Mr. Heckel
Abraham Lincoln Patent Holders Association

Mr. Kohn
Borland International, Inc.

Mr. Brotz
Adobe Systems Incorporated

Mr. Troesch
Fish and Richardson

Mr. Sabath
World Intellectual Property and Trade Forum

Mr. Benman
Benman & Collins

Mr. Baker
Oracle Corporation

Mr. Silverman
Intel Corporation

Ms. Caldwell
Software Entepreneurs Forum

Mr. Chiddix
Time Warner Cable

Mr. Fernandez
Fenwick and West

Mr. Antoniak
Solar Systems Software

Prof. Hollaar
University of Utah

Mr. Henry
Wolf, Greenfield & Sacks, P.C
Boston Patent Law Association

Mr. Cassamassima
Exxon Production Research Company

Mr. May
Iconik Corporation

Mr. Brown

Mr. Irlam
League for Programming Freedom

Mr. Yoches
Finnegan, Henderson, Farabow, Garrett & Dunner

Mr. Shay
Morrison & Foerster

January 27, 1994

Mr. Fiddler
Wind River Systems

Mr. Warren
Autodesk, Inc.

Ms. O'Hare
Mr. Glenn
Intellectual Property Section of the State Bar of California

Mr. Lippe
Synopsys

Mr. Boyle
Multimedia Development Group

Mr. Laurie
Weil, Gotshal & Manges

Mr. Patch
Sun Microsystems, Inc.

Mr. Byrne
American Committee for Interoperable Systems

Mr. Gimlan
Fliesler, Dubb, Meyer & Lovejoy

Mr. Cronan
Taligent, Inc.

Mr. Neukom
Microsoft Corporation

Mr. Morgan
The Prudential Insurance Company of America

Mr. Stallman

Mr. Casey
Silicon Graphics, Inc.

Mr. Sterne
Sterne, Kessler, Goldstein and Fox

Mr. Siber
IBM Corporation

Mr. Lachuck
Poms, Smith, Lande & Rose

Mr. Aharonian
Source Translation and Optimization

Mr. Cole
Mr. Graham
International Federation of Industrial Property Attorneys

Mr. Lemon
Network Computing Devices

Mr. Schlafly
Real Software

Mr. Brand
Reasonings Systems, Inc.

Mr. Judd
Mentrix Corporation

Mr. Higgins
Cooley, Godward, Castro, Huddleson & Tatum

Mr. Grace
Tetrasoft International

--oOo--

JEFF KUSHAN: We're ready to begin the hearings today. We're very pleased to be out here on the West Coast. What I'd like to do is introduce the Vice-Mayor of San Jose, Blanc Alvarado, so we could begin the program.

--oOo--

VICE-MAYOR BLANC OALVARADO

VICE-MAYOR, CITY OF SAN JOSE, CALIFORNIA

VICE-MAYOR ALVARADO: Well, good morning to all of you. It's wonderful to see as many of you here today. Hopefully more people will come in during the course of the hearings, because indeed what is happening here in our city today is very very important not only to the software industry, but certainly to the nation and to the country as a whole.

Commissioner LEHMAN, distinguished officials from the Patent and Trademark Office, ladies and gentlemen, I am Blanc Alvarado, Vice-Mayor, City of San Jose and it is my pleasure to welcome you to our city known as the capitol of Silicon Valley for hearings on patents for software-related inventions.

As you all know, for the past decade the computer software industry has evolved into one of the nation's fastest-growing industries, and today U.S. software firms lead the world in innovation and market share. Silicon Valley firms develop one-fourth of the world's software, and to maintain leadership in this vital industry, a commitment to innovation on the part of each and every one of us is essential.

It is quite a privilege for San Jose to host the first landmark public hearing on patent protection held outside of Washington, D.C. This is the first in a series of public hearings which will result with your input and good advice in a revamping and updating of a system that at best I would say is somewhat out of date. We commend the Patent and Trademark Office for recognizing Silicon Valley's critical role in the world software industry, and I would also like to recognize the efforts of our own Office of Economic Development in bringing these hearings to San Jose.

The dialogue that will take place today and tomorrow and throughout the public hearings in other parts of the country is as you know extremely important to the software industry, to our regional economy, and certainly to the nation's economy. We wish each and every one of you a very very successful public hearing, and we encourage you to speak candidly, to give us your best advice so that those that are here from the Patent Office will be able to take your recommendations, your input in the most serious step further up ahead, to look at how the system for patenting inventions can be improved. We wish you success, encourage you to stay for the two days, and also encourage you to invite other people who are not here presently to attend as well. Thank you very much for being here.

COMMISSIONER LEHMAN: Thank you very much.

--oOo--

COMMISSIONER BRUCE LEHMAN

COMMISSIONER OF PATENTS AND TRADEMARKS

COMMISSIONER LEHMAN: On behalf of President Clinton and Secretary of Commerce Ron Brown and all of us at the Department of Commerce and the Patent and Trademark Office, I'd like to thank the Vice-Mayor and the City of San Jose for providing this great spot to have these hearings and providing the technical and physical assistance that we really needed to come here from the political and legal capitol of the United States to the technological capitol of the United States which really is right here in Silicon Valley. And I think it emphasizes what we would like to think is a growing cooperation between the political capitol and this technological capitol, and particularly the President's commitment to work with California to make certain that California can be everything that it possibly can be.

President Clinton has made the development and competitiveness of America's high-tech industries the cornerstone of his economic program, and he talked about that last night in his State of the Union Address. Promoting these industries will lead to the highway to high-tech jobs for Americans, more high-wage, high-tech jobs, and will insure continued competitiveness for our industries into the future.

In fact, you know, we're not far from the mountains of the Gold Rush. In the Nineteenth Century it was the wealth in the ground that created the wealth of California and the nation. As we move into the 21st Century, it's the wealth of the human mind which is going to be our most precious natural resource. That wealth doesn't mean a whole lot if it doesn't have a conception of a modern up-to-date legal system which defines the rights that individuals have in their creations. That system needs constant revising and constant modification, and that's why we're here, right here in the heart of Silicon Valley, where we see a tremendous potential for pursuit of the Clinton Administration's goals.

Much recent concern has been expressed over the patent system to protect software-related inventions. These concerns range from claims that the patent system is incompatible with software development to skepticism over the ability of the Patent and Trademark Office to accurately gauge innovations in this field of technology. However, to date, there has not been a forum in which those having concerns could air them with the hope that they would be heard, evaluated, and used to develop future policy. These hearings are intended to provide that forum. We're really quite serious about having this opportunity to really hear from people in the business about what they think about the current state of the intellectual property system, what they think needs to be done to improve it.

Before we begin I'd like to provide you with a little information about the Patent and Trademark Office, our operations and our plans for the future. The Patent and Trademark Office is an agency of the Department of Commerce. It's a very old agency; it was founded in 1790. It's one of the first agencies of the Federal Government. It's a part of the team that the President has assembled to promote technological innovation and exploitation to increase our exports and to enhance the overall competitiveness of U.S. industry. The Commerce Department is also leading the Administration's initiative to accelerate the development of our nationwide electronic superhighway. The President also talked about that last night in his State of the Union Address. This Information Superhighway, and of course Silicon Valley and the intellectual property system is very much a part of that, will

be the basis for our information-based high-tech economy of the 21st Century.

Our office plays an integral role in this team, and Secretary of Commerce Ron Brown has given us the following mission:

First, we're to administer the laws relating to trademarks in order to promote industrial and technological progress in the United States and strengthen our national economy. Secondly, we are to develop and advise the Secretary and the President on intellectual property policy, including copyright matters, and of course that's part of what we're here about today. And finally, in cooperation with other trade agencies of the government, the International Trade Administration in the Department of Commerce, we are to advise the Secretary and other agencies of the Government such as the United States Trade Representative on the trade-related aspects of intellectual property, and we just finished, as you probably know, a very successful effort that culminated in Geneva last month to provide a new trading regime for the world which will very much benefit high-tech industries, particularly the computer software industry.

The new focus of the President and the Secretary of Commerce on technology-based economic growth makes this a very exciting time for me to be leading the Patent and Trademark Office as its commissioner. It also places a serious obligation on us, however, to ensure the proper functioning of the patent system, especially in the rapidly-developing areas of technology. We have devoted a substantial amount of effort and resources to improving quality of examination for our software-related inventions, though I must say there are some inherent problems there which are very difficult to address, and just to point out what one of them is, particularly in this area of computer software-related inventions is that a lot of what is known in this area is in the area of trade secrecy. It's not written down anyplace. It's not even in prior patent applications, and so we have a very difficult time sometimes making determinations -- what is the existing state of technology? And as most of you who are here know, that's a critical requirement in order to be able to issue a patent, to know what is in fact new, what is a new innovation that should warrant protection.

Our Computer Systems and Applications Examining Group headed by Gerry Goldberg, who is here with us today, has been at the center of this effort. This group currently employs over a hundred and sixty examiners who bring with them a wide range of expertise and experience. We use stringent hiring standards to ensure that our examiners come into our office with the proper background, and then provide training from experts in the field to ensure that they keep abreast, not only of the state-of-the-art technology, but also of current legal standards. These examiners are responsible for the examination of a steadily-increasing number of patent applications being filed by inventors in this field. In 1991 we had 6,600 applications. In 1992 we had 7,500, and last year we had over 8,300. That's out of a total of about 190,000 patent applications in all fields of technology in our office.

We have also worked hard to remain receptive to public and industry concerns, and I think this can be seen through our extensive efforts to improve our prior art collections, conduct training for our examiners and recently to respond to intense industry concern over the issue of patents.

We will soon forward to Congress a legislative package. We'll make our reexamination process more open to third-party participation. These changes will make reexamination a more attractive option for those having reasons to question the validity of any particular patent. We will also be changing the patent term to run from twenty years from the date of filing rather than seventeen years from the date of grant. This is a change that is good for the United States. We believe that it will prevent the disruptive effect of patents that are issued long after they have been filed, due to administrative delays in the processing of the patent applications, delays that are often deliberately arranged, sometimes by patent applicants who want to extend ultimately the reach of their patent further than it ought to go.

We recognized this benefit when we agreed to a change recently in the GATT-TRIPS context where we agreed to an international standard of twenty years from filing as the standard that all countries would attempt to achieve. But as an extra bonus, by making this change, we have been able to convince -- and I just back from Tokyo last week -- we've been able to convince the government of Japan to loosen their rules regarding filing patent applications. This will greatly assist U.S. inventors in their efforts to gain patent protection in Japan. And I should add that intellectual property protection in foreign markets like Japan is a very vital part of our effort to encourage and promote U.S. exports, because what we have to promote is often the technology itself, which is not going to be very valuable unless it's protected by an internationally-recognized regime of intellectual property laws.

Finally, our hearings today, I would like to think, emphasize our desire to remain receptive to the needs of our users and our public. Of course we, I think, will find out in the next few hours that sometimes those recommendations that we get from users in the public aren't always in harmony; I'm sure that we're going to hear differences of opinion, and we are going to have the task of sorting through those differences and coming up with a policy that works.

I'd like to just make an observation about the nature of the intellectual property system before we proceed, and that is that I don't think there's any question about it, that intellectual property protection, patents and copyrights, have been a major part of the economic growth of America from the very beginning. When I walk into my office every morning, I see the patent model of Thomas Edison's light bulb sitting there, greeting me as I walk in the door, and out of that, great industries have been built. I see mid-Nineteenth Century inventions, inventors like Eli Whitney. That innovation is increasing in the United States, and it's always been protected by the patent system and encouraged by the patent system. The purpose of the patent system is to incent innovation, not to disincent it. And patent law is a very delicate instrument. If it doesn't work right, if the threshold of patentability is too low, there's too much confusion in the system, it may actually end up disincenting innovation rather than incenting it. The best kinds of patent systems is a patent system that is clear, that everyone understands, that requires very little litigation to use it, and it is our concern that we're not quite seeing that

kind of a patent system, particularly in the software-related inventions area, that has caused us to be here in Silicon Valley today, and we're going to do everything that we can possibly do to try to make this the best system that we can.

Finally, before we call our first witnesses, I'd like to introduce the people who are here with me at this table this morning, and I'd like to start at my far right with Michael Kirk. Mike is currently our Assistant Commissioner of Patents for External Affairs. He has been responsible for some time for all of the policy and legislation in international matters in the office. President Clinton has nominated him to be the Deputy Commissioner of Patents, and as such he's going to lead an effort to deal with these policy problems and make our intellectual property system all that it can be.

Next I'd like to introduce Ginger Lew, to my immediate right. Ginger Lew is our new General Counsel for the entire Department of Commerce. She works directly for Secretary of Commerce Ron Brown. Ginger is from the Bay Area. She was a practicing lawyer out here, has worked in this area of law herself, and I think is an illustration of the kind of technology-oriented, California-oriented people that we have in our administration who I think are going to be very receptive to the concerns of people out here.

Next I'd like to introduce Larry Goffney. Larry is our new Assistant Commissioner for Patents. His responsibility will be to supervise the patent examining corps. He'll have about three thousand people working for him, including all the examiners and support staff who examine the software-related inventions. Larry has taught law around the United States, he has a an engineering and a law degree, he was a patent lawyer and a partner in a very prominent law firm before coming to our office, and we're very excited about the prospects that he brings to our office and his understanding of these issues. And finally you have already been introduced to Jeff Kushan who is on the staff of our Office of Legislation and International Affairs who has been the sparkplug who's really pulled all of this together, and I know that this room I think seats about a thousand people, so -- I think the Vice-Mayor was a little concerned that we didn't have a good turnout, but I think that when you think of how many people the room seats that we're having a pretty good turnout, so it indicates that Jeff has done his job of getting the message out so that all of you know that we're here to listen to your concerns.

The people who will be testifying over the next two days should have received a schedule indicating the approximate time that they have been assigned to give their remarks. A final list is available at the entrance to the room. I imagine most of you got it, and I would encourage all of the people who are scheduled to testify here to be here at least twenty minutes before your assigned time slot. We're going to try to keep to this schedule as much as we can, but hopefully you'll give us a little bit of leeway either way. Each person will have eleven minutes to speak, and the computer monitor in front of this podium here, this computer monitor that you see sitting in front of Jeff, will display a green screen for nine minutes, and then it will turn yellow during your last two minutes, so that you'll know that you'll have to start wrapping up, and then when the screen turns red the time will be up. I would encourage everybody to try to cooperate with us and stick to these time limits so that we

can be fair to everybody. Unfortunately if one person goes over too much then that really is at the expense of others, and we're not going to have a very balanced hearing. I think eleven minutes ought to be pretty good time for people to get most of their comments in, but of course we're not limited to these oral comments, and I would encourage anyone who has further additional written comments to give them to us and anybody who's not here, anybody that anybody here knows is not here who has views about this should certainly feel free to send in their written comments and they can consult the Federal Register notice which was published on December 20th, 1993 for more information about this. We may have copies of that out here for those of you who don't have it, and Jeff can certainly get that for you if we don't.

The notice has been widely circulated through the Internet, and it can be retrieved at our ftp site, which is comments.uspto.gov. The transcripts for these hearings will be available after February 7th, 1994, and paper copies will be available from our office for a charge of $30.00. I'm sorry we have to charge that, but it's unfair to ask our patent applicants to pay that fee, so there will be a $30.00 fee which is basically our charge of reproducing them, and the transcripts will also be available through our ftp site that I just gave to you.

Once again I'd like to welcome everybody here today. It's a real pleasure for us to be out here, and I think this is going to be a very productive exercise, that we are going to in fact learn much information that will help us turn those dials of law and policy in Washington so that they're totally finally tuned, and we can provide all of you out here we're looking to to create the wealth of America for the next century, with the kind of intellectual property system that you need to do your job properly.

In calling our first witness, I'd like to say even though everybody has eleven minutes to speak you don't need to use all the eleven minutes, don't feel that you have to do that. To the extent that you leave us a little more time, we might be able to ask a question or two if we feel motivated to do that. So with that explanation, I'd like to call on our first witness, who is Joe Clark, the Chairman and CEO of VideoDiscovery, and welcome, Mr. Clark.

--o0o--

JOE CLARK

CHAIRMAN AND CEO, VIDEODISCOVERY

MR. CLARK: Thank you very much, Commissioner LEHMAN. My clock says 9:28, and that's what it says on the sheet, so I congratulate you the good timing of your remarks, and I'll try to do the same.

I really appreciate the opportunity to come and talk to this group. I have a publishing company in Seattle, Washington that publishes multimedia products for science education, and we're pretty well-known at this time throughout the country, and we're totally frustrated by our experiences with the patent system. And so what I would like to do during my time is kind of share with you personal experiences that we had in order to illustrate some of the problems that I think are inherent in the current patenting process, and then I'd like to recommend two or three different solutions.

Now, you've sort of preempted my remarks, and I'm really

appreciative of it. It's the first time that I've heard that the Patent Office was drafting legislation that would take care of most of these, but there's one part of it that I want to try to deal with and maybe ask you a question about.

On the plane on the way down I thought it was in some way almost poetic that I was headed down to California. I was a little bit anxious about earthquake country, you know, in fact I didn't sleep very well last night. But I thought about the earthquake thing as not being too dissimilar from the situation we find ourselves in with the patent system, that is to say, in the last year and a half we've heard some rumblings coming from the Patent Office with patents like the Grass Valley patent and the Optical Data patent and then the Compton patent which sort of got a 7.3 on the Richter scale as far as I could figure out, and my big worry is that there's no end in sight, that is to say, given the secrecy in the Patent Office, we don't know what to expect tomorrow, next week, next month, and so on.

In my case, I experienced a situation where a patent was called to my attention that literally could drive me out of business. I started VideoDiscovery in 1983. I worked very hard for ten years to get it to the point it is at, with sixty people, and for somebody to have a patent that was a club that could just close down my operation didn't set well with me. I didn't know much about the patent system until about a year ago when I got a letter from my competitor that called attention to the award of two patents, and they indicated that they thought we might be infringing on their patents and suggested that I contact their technical person to see about licensing the technology.

Well, they had two patents. One of them was for a method of instruction that we lovingly call the Socratic Method, and the other patent was for a method to customize a curriculum, using a computer to do it, and we thought that that had been a process that had been going on for a long, long time.

So my immediate reaction was confusion. In the first place I couldn't believe that anybody would patent these things. If I could have the Number 3 slide, please. I just want to describe this one patent for you. The patent on the Socratic Method, as I say -- this is not the way it's described, it's described as a method of instruction -- was composed of a trilog, and the trilog had three components. One was a random-accessible reservoir of information like a video disk player -- could have been a CD-ROM or a hard disk, I guess, or maybe a textbook, a teacher was the second component, and the students were the third component. The way this system worked was that the teacher was given instructions where to go on the random-accessible reservoir of information, withdrew the information, passed it on to the students, the students responded, and then they would go back to this system.

Now, they got a patent for this. I couldn't believe it. And my first reaction was anger, disgust, you know, disbelief, and so on. More recently I've modified my position where I believe that the, the Patent Office is indeed not the culprit, but the victim in a system. Any system that operates in secrecy like that where they can't confide or consult with the members of an emerging industry, for which there's no prior art, no experience, no informed judgment on the part of the examiners, is ultimately a victim to this thing where they

have to -- they, they can't distinguish obvious from unique and are almost obliged to issue the patent.

So the point there is that I think that if we would remove the veil of secrecy, which I hope is part of the legislation package, and I'm not sure that is attached to, you know, ultimately or necessarily to first-to-file process, so I'd like a clarification about that, but that is really critical, to support change in our industry. At one point it was the computer industry, it was the biotechnology industry, it was the software industry, the multimedia industry, what's going to happen next week, next month and so on, but whatever the new, emerging industry is, unless that law is changed so that there's some better process, we're going to face the same kind of dilemma.

Let me come back to the history on my case. As soon as we got that indication, as a small company I did not have many attractive alternatives. One was litigation, which was very very expensive, and I could literally not afford that, and the other one was the reexamination process. In talking to no less than ten patent attorneys over the course of last year, I never got one that recommended reexamination process, and so I'm very happy that the Patent Office recognizes the problem with that and is going to include more third-party involvement in that reexamination process. But that was one of my recommendations.

As it happens, when we filed in Court in August to ask the Court to find these patents invalid based on the obviousness of the patent and the existence of prior art, the other party had sixty days to respond; and on the sixtieth day they decided to donate the Socratic Method back to the public where it belongs. The second patent, however, they asked for reexamination in the Patent Office. That puts us in a terrible position. We know that the inventor has a tremendous advantage through the reexamination process as it exists. We also have some trumps, some prior art that we know that will knock that out, and we don't know whether to provide it to the Patent Office or to save it for litigation if it emerges from the Patent Office. So you can kind of see our dilemma on that thing.

So a recommendation that I would have is to change the process so it's published prior to award. That would solve the problem and serve existing, you know, changes and so on (sic). Second recommendation would be improve the examination process. I think that that's being considered, and so on. But a compromise position, and something that I would like to see for immediate relief -- I'm talking we need Federal aid now, an immediate relief to get us out from under this anxiety, is either a hiatus on any more patents coming out for multimedia -- that would be preferable -- if you can't do that, then I'd like to see the Commissioner empowered to constitute a commission and do peer review of any patents that have come up. The Commissioner would identify new emerging industries, all right? And when they saw that, because there's no prior art, because there's no experience with the examiners and so on, that they work with the industry leaders in that particular industry, to develop the prior art collection, to set the guidelines for what's patentable, to review patents as they come out.

So I hope that you can sympathize with the position of a small business person who has gone through the fear and the anxiety of having something like that ripped out from under

them. I feel I've been extremely lucky to get my resources together and to be able to operate in this system.

Thanks very much for your attention; I'd be happy to answer questions.

COMMISSIONER LEHMAN: Thank you very much, Mr. Clark. In terms of this reexamination process that you're concerned with right now, I think one of the problems that we have in reexamination right now is that generally speaking we limit the prior-art references in a reexamination to those that are submitted by the person who is seeking reexamination, and that creates a problem for third parties.

MR. CLARK: Aha!

COMMISSIONER LEHMAN: So in this situation it sounds to me like the patentee kind of got the leg up on you by being the first to request reexamination. Is that problem.

MR. CLARK: Yeah, I don't disagree. Yeah. I don't disagree that I was outmaneuvered, you know, but -- yeah, you're right. You're right.

COMMISSIONER LEHMAN: I really appreciate that. Does anybody have any other comments? If not, thank you very much for coming down here and sharing these thoughts with us.

MR. CLARK: Thank you.

COMMISSIONER LEHMAN: Next I'd like to call Mr. Ryal Poppa, the Chairman and CEO of Storage Tek.

--o0o--

RYAL POPPA

STORAGE TEK

MR. POPPA: Good morning, and thank you very much. I'm going to be answering principally the Question 4 on whether the circumstance or the framework of the current patent law preserves competition.

Quick background on Storage Tek, we are a twenty-five-year-old company, start-up, much like you see here in the Valley. We do about one and a half billion, have ten thousand employees. We belong to many organizations, the CCIA, Computer and Communications Industry Association; ESIS and ASIS, the European and the American Committees on Interoperable Systems; AEA, American Electronics Association; and the IEEE, the Institute of Electronic and Electrical Engineers. All of these are very pro-competitive in their attitudes, and that's the principal reason we belong to them.

For the record, we have over three hundred programmers cranking out programs all the time, we have lots of patents and we have the same risk everybody else does. In the industry though we do build principally data storage devices that are attached to all the major mainframe manufacturers of the world, eighteen at this time, and many networks. We do support patents in every sense, as applied to the area of software, with one exception. We believe we have to have continued decompilation rights to maintain interoperability when needed. Modern APIs or mandatory APIs would do the job -- application program interface -- but frankly most companies do not want to do this, but that would be a solution in addition to decompilation.

In supporting the thesis of "pro competition" I want to give three illustrations.

One is old Ma Bell; we all know what it was like in the old days, very bureaucratic, very successful, it was the stock of widows and orphans, but it was a stifling, stultifying, noncompetitive environment. The new Bell of course is growing, marvelously. It is competitive. The competitive juices are flowing. And think of all the new services in the last twenty years that we have become accustomed to. Auto-answer, FAX, networks, fiber-optics, cellular, ISDN, et cetera, et cetera. New companies; MCI, the WilTel's, the Sprints, et cetera, and new related industries defined by companies such as Novell with software, and NSC for high-speed data transmission, McCaw for cellular, Hayes for modems and the list goes on and on and on as you well know. But this flowed out of a pro-competitive stance, a break-up of a company, and it led ultimately to [DarvdaNet], InterNet, and the information highway. That was the genesis of that kind of program. Today we've globalized that communications capability all over the world and we continue to do so by freeing our telecommmunications capability.

A second illustration is, in my principal industry, computers, leader of the industry, IBM, fell upon hard times. We all are aware of that. They are correcting that problem and will clearly return to health. But they fell into a pattern of being in the sixties and seventies highly competitive, and then they went into a protect mode, where they were trying to hold onto their base, hold onto their customers, and as a consequence, they fell behind, they became anticompetitive, and as a consequence, they began to lose the competitive juice, and even today some of the comments made by their own officers saying that we have lost the will to compete. They're getting it back, they will return, but it was because they lost their desire to win. Part of that was the intense, massive fight over copyrights and patents, protecting in the court, fighting in the courts rather than fighting in the customer arena to save the customers and keeping them happy. No new ideas were allowed in the seventies and eighties. Where did the PC come from? or the mini? or the workstation? Not out of the big companies. They came out of the start-ups, where they could get the proper patent and copyright protection. But oftentimes decompilation is part of that because you have to maintain interoperability -- very fundamental issue.

I sincerely hope that Lou Gershner understands that there's a smothering effect of too extensive use of patents and copyrights. It destroys the design capability of the company because they don't look to get their products to market quickly, they design poor patents and protection, and therefore it doesn't really work to their great benefit. In the last six years, we along with many of our colleagues, have opposed the forces in Europe that have been seeking to use copyright and patent protection to limit competition, and as you well know, the EC finally came down on the point that decompilation is legal, desirable, when necessary for interoperability, and that's the only thing we look for -- no piracy, no cloning, no copying, no replacement of the program, but interoperability. That must be preserved.

The last example is the information highway. This is not an economic debate in the way it was between Bell and IBM, but rather it is a competition of ideas, of interchange, of the exchange of relationships over the Internet, as you -- if you

follow it and see what goes on, it is used in many, many ways.

I would add a challenge to you, because of that Internet. Looking for data the way you are, and in your opening statement you said you're actively looking for ideas from our society, put out on the Internet a question. Should decompilation be allowed for the purpose of interoperability, but clearly not for piracy, cloning or replacement of the basic software? I think you'll get a very strong eighty-to-ninety-percent positive response that it should be. The reasons are you have organizations like the IEEE with two hundred and forty thousand members who have voted and publicly stated they support decompilation when required. The same with ASIS, the same with CCIA. In the case of CCIA we have sixty-seven companies -- I happen to be Chairman of that -- most of the RBOCS, all but one, AT&T, Univac, Amdahl, Storage Tek, and we have approximately a million employees represented, and they would vote for decompilation and maintaining it as a, a necessary option. Remember the alternative is a mandatory API. That could be something that could be legislated, but if not --

COMMISSIONER LEHMAN: Could you explain that?

MR. POPPA: Applications Program Interface?

COMMISSIONER LEHMAN: Right. Well, API, but what would be a mandatory API?

MR. POPPA: Well, literally in law saying that a second vendor or a third vendor would have rights of access to the base code so that they could understand it, so that they could build their code to interact with it and interoperate with it. Today most companies are locking that up on an object-code-only basis, and not allowing us to see it.

COMMISSIONER LEHMAN: To some degree this is an antitrust issue, and my colleague, Ann Binghamen, who is the Assistant Attorney General for Antitrust over in the Justice Department has indicated that she's going to put together a task force to start working on these problems, and it may well be that this sort of mandatory API idea should be explored in that context. In a sense that's more of a, I think, an antitrust kind of a solution to this problem as opposed to an intellectual property solution.

MR. POPPA: We think that's a good alternative. We would also hope that within the structure of the PTO we could get a clear distinction, as they have done in the EC, but I don't think they did it as clearly as they should. They left it a little unclear, such that some people are going to be able to say, Well, Gee, for interoperability I can really replace the program. We are not in any way supporting that position; only that we should be able to access the program so that we can look at it and make sure that we can make our programs talk to each other. Nothing beyond that.

My last point, and it's really a picture worth a thousand words: a terminal device we're all well-familiar with. It is a very standard device, but it also comes with an information network, with a very standard interface. It's a telephone. But when you take patents and copyrights and you tighten them down such that you could not in any way see the inside of the machine, what we do is we cover up the access port, we take the network, we cut off this end so you can't see the interface, and we say, Now, Mr. Engineer, make them interact. And that doesn't make any sense, because it isn't just a telephone, it's a computer with three million instructions in it, and this network has on the line hundreds of terminals with other millions of instructions, and therefore decompilation and interactivity must be maintained. Please.

Thank you very much.

COMMISSIONER LEHMAN: Thank you very much. Do any of my colleagues have anything they want to add? Thank you very much. Thanks for coming over.

MR. POPPA: You bet.

COMMISSIONER LEHMAN: Next I'd like to call William Ryan who is representing the Intellectual Property Owners Incorporated, but is directly with AT&T.

--o0o--

WILLIAM RYAN

INTELLECTUAL PROPERTY OWNERS, INC.

MR. RYAN: Good morning, Commissioner LEHMAN, and members of the Panel. My name is William Ryan, I'm a general attorney at AT&T. I'm here, however, as you say, representing the Intellectual Property Owners. They are a nonprofit association located in Washington whose members are companies, including AT&T, other companies as well as universities and individuals who own and are interested in patents, trademarks, copyrights and trade secrets. IPO presents these remarks in support of the continued strong patent protection for computer program-related subject matter.

There can be little doubt that the computer software industry is an important and growing economic force in this economy, in fact for year 1992 it was estimated that just the packaged software industry accounted for some seventeen billion dollars in revenue, and importantly about half of that was from sales oversees. However, the amount of revenue in the software industry is not limited just to the packaged software. There is indeed a great deal of other software that goes on that is less visible but nonetheless very very important. I make reference to control software for manufacturing systems, for controlling the telephone network, for the ubiquitous microwave ovens and the VCRs and intelligent-talk telephones and many many other applications.

The sales and revenues for devices or the software component of these devices and systems by many estimates well-exceeds a hundred billion dollars a year. Even one level higher in the scheme of things, the services provided by both the equipment and many of the underlying processes, again referring to such things as the financial systems, the telephone network and the entertainment business, are increasingly based on software infrastructure underpinning it. Therefore, the effect on the economy of decisions made relating to software have implications much beyond the kinds of software sales that might be accomplished in local over-the-counter sales.

We don't come by this cheaply, though. Software research and development is very expensive. Among the hundred top packaged software companies it's been estimated that an average of seventeen percent of revenues flows through to support continued R&D. And many companies not

traditionally thought of as software companies -- and again only by example I'll refer to AT&T, we have an R&D budget of approximately three billion dollars per year, and some sixty percent of that is devoted to software efforts, and we are not exactly thought of as a software company.

With this kind of economic importance at stake, it's clear that the managers of our business have to seek ways in which these large investments can be protected. One way that it's been adopted of course is the well-known copyright as it has been applied for example to the mass-market software, and it's been done very successfully by and large. Less publicized though is the need to protect and the vulnerability in fact of the other software investments, some of which I referred to before, the large systems software and the embedded software in many, many applications.

So then the question is asked, Why isn't copyright protection sufficient to protect the software investment? Well, it should be clear from the very words of the Act itself that the copyright is intended to protect the expression. It does in no event protect the systems, the methods, and other aspects of functionality within the software. Importantly too, copyright suffers from a limitation that it is not as precisely defined as patents, although the arcane language we practitioners use to define our inventions in patents is sometimes criticized, nevertheless it is precise, and with copyright protection there is no such precision. It's a very much looser judgment that business people have to make as to where the boundary is for their protection.

Moreover, the Supreme Court has made it very clear that patents are the preferred mode for protecting the functionality, the implementations of ideas as against the expression of the ideas, which the Copyright Law can protect. We think that computer program-related subject matter is, and has been, protected well in the past, by patents.

Many people think that software protection for software commenced only perhaps in 1980 or '81 with the Diehr decision, but that is not true. We have been filing applications in the computer-related industries for decades, going back at least until the 1950s. And it's proved very effective. One of the questions in the Notice of Hearing was, what experiences have we had, and by and large the experience has been good. We've been treating software inventions for some time in precisely the same way as we protect and treat inventions in other areas. In fact, in many ways it's hard to tell whether an invention is a software invention or not. I'll deal with that just a little bit later.

Why are the patents for software-related inventions particularly important? Well, they're not particularly important for software any more than any other inventions, for example, investors seeking to sponsor a start-up organization or a new enterprise within a larger company would like to have some certitude about what it is that they can hope to have some protection for and where their investment, how their investments can be protected.

Also, importantly, is the disclosure aspects of patents. One of the functions served by patents is to disclose to the public. Before an allusion was mentioned to the secrecy that often attends patents. Well, in many ways the contrary is true. Patents themselves of course contain disclosure, but also in an organization like mine again, we encourage publication of technical ideas, in fact last year we published some forty-four hundred technical articles. Many of these would not have been published if we could not also have concurrently filed patent applications so that the publication of the technical papers would not compromise the value of our inventions included in the disclosures.

Patents are important in many other ways, one of which is the -- they provide a vehicle for developing of the ubiquitous alliances that are present in the software and hardware industries. They provide a medium, in fact, for people to come together and exchange value so that they can work together to get a cooperative result. Often this helps people and companies get into new markets and establish businesses that would not otherwise exist.

Again, some people have said that software inventions should be treated differently. We think not. In some respects, in fact to treat the software industry as an industry raises more questions than it answers. As I mentioned before it's much of an enabling technology in many domains, in telecommunications, in entertainment, in finance, in manufacturing and process control, software is often a common denominator. And the earlier-mentioned national information infrastructure, the highway likewise is largely a software development, and it's been going on for some time.

Some people suggest that a solution is that we should have a new statute, one especially tailored to software problems. We think not. These sui generis proposals have arisen in the past and have been bandied about for some time and in one case applied to a different subject, the Chip Protection Act is an example. We think these raise more problems than they solve as well. We think the existing scheme is workable and is fair and appropriate. If we were to adopt the sui generis scheme we'd have to live through many of the uncertainties that we've had in both the copyright and patent realm for these last twenty years. We may have to live through the same issues tried from a different perspective. Moreover, the many other of our industrialized countries, the European countries for example and Japan, have statutory schemes that are roughly equivalent to our present protection for software-related inventions. Patents are available there to about the same extent they are here.

The recently-enacted NAFTA and GATT treaties also have dictates in them that would suggest that we cannot get too far out of whack. We must have a level of protection consistent with what is currently provided by the patent statute.

The questions posed in the Notice of Hearing, in some cases have been dealt with; in other cases they deal with details of claim formats, which I'm not prepared to deal with now, but which we will respond to in our extended remarks.

In concluding then, we'd like to say that consistent with the findings of the final report of the Advisory Commission on Patent Law Reform, to the Secretary in 1992, the current statutory regime for the protection of rights, both copyrights and patents and other matters as well, is adequate; it is working and it is working well, but not perfectly. If the experience of the last ten years teaches us anything, it is that we can't predict with any certainty what directions the information processing industry will follow in response to new technology and new global political and business trends. This vast changing environment requires

the U.S. patent laws and implementing procedures to be technologically neutral and flexible enough to avoid major discontinuities.

Thank you.

--oOo--

COMMISSIONER LEHMAN: Thank you. I have a question, and that is that, you mentioned that the exclusivity provided by the patent system encourages you to publish because obviously you can make the information available and then you can be certain you'll be protected, and of course that's been a traditional trade-off in the patent system; you disclose, and you get protection.

One of the suggestions which was made by Mr. Clark at the very beginning of the hearing and that's floating out there is that in this particular area where it's very hard to keep up with the technology, where a lot of the prior art is not easily findable, is that we have some kind of prepublication prior to the issuance of the patent. That, if we were to do that, that obviously to some degree abrogates that traditional deal, because certainly you know that you might get the patent but you're not certain that you're going to get it prior to the disclosure, and I'm wondering if you have a reaction to that.

MR. RYAN: Well, that's a risk that the parties would have to take, of course, knowing when they file their application that there's a possibility they will have in effect given away the genesis of the invention and for one reason or another are not able to get a patent. That's a risk the filing party would have to take.

COMMISSIONER LEHMAN: Do you think that the advantages though of prepublication would outweigh the negatives of that risk?

MR. RYAN: It would have to be evaluated on an individual basis, but I think in many cases that's the case. We publish much more broadly in the technical literature than we do in the patent literature.

COMMISSIONER LEHMAN: Thank you very much.

Next I'd like to call Richard LeFaivre, the Vice-President of the Advanced Technology Group of Apple Computer who will be representing the Computer and Business Equipment Manufacturing Association CBMA.

--oOo--

RICHARD LeFAIVRE

APPLE COMPUTER, AND,

COMPUTER AND BUSINESS EQUIPMENT MANUFACTURING ASSOCIATION

MR. LeFAIVRE: Thank you, good morning. My name is Rick LeFaivre from Apple Computer and today I'm actually wearing four hats, first as a computer scientist with twenty-five years' of experience in software technology as a researcher, a professor and an R&D director, second as Vice-President of Advanced Technology at Apple Computer. My organization is responsible for a large percentage of the patents that are granted to Apple, and the protection of the innovation that we do is very important to me. In particular over the years we've seen a marked shift in our innovation focus from hardware to software, and so I'm very interested in the topic of these hearings in particular.

Third, I'm the founding member of the Executive Committee

of the Software Patent Institute. As you may be aware the SPI was founded to provide training in software technology and access to prior art, to help insure that those software patents that are granted are of high quality, and we're working very closely with Gerry Goldberg in that task. I should point out that the Software Patent Institute has chosen to take a neutral stance on the broad issue of the patentability of software so the views I'm about to express do not necessarily reflect those of the SPI.

Finally and most importantly I will be testifying today on behalf of the Computer Business and Manufacturer's Association CBMA, and let me give you a little background of this group. CBMA is a trade association whose members represent the leading edge of high technology companies in the computer business equipment and telecommunications industries in the U.S. In 1992 CBMA's twenty-six members had a combined estimated sales of more than two hundred seventy billion dollars, which represents about four and a half percent of the U.S. gross national product. CBMA member companies employed approximately a million workers in the U.S. in this past year.

The computer industry performs about twenty percent of the total private-sector R&D investment in the U.S. That figure is about five times the investment of the aerospace industry, three times the investment of the health care industry, and four times that of the chemical industry. This investment allows our members to rapidly advance the capabilities of their products and to get access to, and compete successfully in, a very tough international marketplace. It also results in significant numbers of jobs just within R&D alone. I'm here today because patent protection for new computer functions is absolutely crucial to all our members. Software-related inventions fit within our present patent system and patents issued under a sound application examination process support the Constitutional mandate of promoting the useful arts and sciences. CBMA members file for and obtain patents for software-related inventions. They also enter into agreements to utilize such patents held by others. Because our companies typically have broad product lines, they address patent issues in many areas of technology. They see no reason to treat software-related patents differently from patents related to other technologies.

In the first question set forth in the hearing notice, there are a number of subparts relating to claim subject matter and claim formats. CBMA's response to this question is simply that if the claim is drawn to the solution of a real-world commercial problem, and the claim functional steps or elements as a whole meet the strict legal requirement to be new, nonobvious and useful, then a patent should issue. The function claimed, not the format, is what is important. It shouldn't matter whether new, nonobvious and useful process steps are claimed in the context of a program or a disk or claimed in a hardware or method format, or in the context of a semiconductor chip. Software-related inventions are valuable to the purchaser not for what they communicate, but for the functions they perform. The functions are what are important and what should be assessed for novelty and nonobviousness.

Relative to Question 2, our members have integrated their software-related patents into their overall patent portfolios

and practices so that separating out their impact is quite difficult. However, this integration itself demonstrate that these patents are just like all others. They are sought when the inventor or his or her employer believe that the investment in obtaining the patent will be returned. Conversely, CBMA members often must respect the software-related patents of others, which they do in the same manner as further technologies.

Regarding Question 3, the standard for patent eligibility for software-related inventions should be maintained at the same level as for all other technologies. An alteration in that standard would negatively impact investment in our industry. If the standard were to be restricted severely it would disarm CBMA member companies in their dealings with foreign competitors because licenses under U.S. patents are used to negotiate access to foreign markets and foreign technology. Obtaining patents for software-related inventions in our principal competitor countries is generally equivalent to that of the U.S.

Software-related technology will be one of the leading technologies of the 21st Century. Discrimination against this technology would set a terrible example sure to be rapidly adopted by the developing world. To now have the leading country in software creation and patents declare that such inventions are excluded from the statute, despite falling within the terms of statutory subject matter, or are to be treated differently from patents involving other technologies, would reverse much of the hard-fought progress that has been made over the last decade in improving intellectual property protection throughout the world.

Relative to Question 4, patents provide the relatively broad protection necessary to bring in risk capital for new and useful inventive functions that are generally defined in terms of processes or methods of operation. This protection should be afforded only after a detailed examination to insure that the claimed functions are truly novel and nonobvious. This, by the way, is one of the places where the SPI is trying to work with the Patent Office to make that process more efficient. In contrast, copyright protects only the expression contained in the computer program, as it does for other literary works. High-level functional processes are expressly excluded from protection by statute. Thirdly, trade secrets provide the necessary protection to facilitate the disclosure of confidential software-related designs to employees, joint venture partners and others within the structure supporting that confidentiality.

Thus, each protects different aspects of the intellectual property. The inventor, who may not wish to or be able to author a complete software product, deserves protection. The author of a program deserves protection from piracy and plagiarism. Those with confidential information, willing and able to keep it confidential, should be able to protect that value against those from which it has a fiduciary relationship.

Finally, with regard to Question 5, CBMA supports continued reliance on the tested, well-developed protection of patents, copyrights and trade secrets. We strongly support continued improvement in the patenting process for software-related inventions. But nothing suggests the need to treat software differently. A new and untested regime would fail to provide inventors and authors with any certainty of protection for an extended period of time while judicial precedent was developed to determine the scope of the law.

Additionally, international protection for our software research and development is critical. There is no certainty that a new protection system could be implemented worldwide, whether through multi- or bilateral negotiations. The hard-fought protections in the GATT, TRIPS and NAFTA treaties regarding literary work protection for programs and the issuance of patents without discrimination based on technology were just obtained last year. It is inconceivable that such protections would now be abrogated with the ink hardly dry on these provisions by the adoption of a sui generis protection.

In closing, our message to you is this: Don't cut back on patent protection for software-related inventions because some invalid patents may have been issued. The current reexamination process and the Federal Court system do provide mechanisms for the removal of these mistakes. We believe that further training for examiners, and access to a larger library of prior art can and will reduce the possibility of future mistakes. Overall, the system is working and should be improved, not abandoned. If the standard for patentability is changed for software-related inventions, or if patent protection is dropped in favor of some new form of protection, it will severely and negatively impact CBMA members, our industry and the country.

Thank you for letting me submit these remarks and we look forward to continuing to work with the Patent Office on these issues.

COMMISSIONER LEHMAN: Thank you very much. I have one question if you have a moment, and that is that, there's obviously a difference of opinion about the application of the patent system to the software industry that is represented in the room, we've already heard it this morning and I think we're going to hear more testimony about it. Apple certainly is a company, and I gather that CBMA is a company now that very much favors patent protection for software, and Apple's certainly a very important, successful part of American enterprise today.

One question that I have is that obviously the purpose of patents is to incent people to invent and to make investments. And can you point in your own experience to an example where that has happened? Has the patent system actually been a factor in a decision to go into a new technology, the fact that it might be patentable? Has it been a factor in getting financing from capital markets?

MR. LeFAIVRE: Yeah, that's a good question. Apple thinks a lot about patentability of any technology, software or any other, in looking at some of our innovations. We do feel that there has been a lot of investment made in technologies that, to be quite honest have been appropriated, copied, whatever, by other companies, that have not helped our situation in the marketplace, I think it's fair to say, and so we certainly are interested in trying to evaluate the patent potential of different technologies as we develop them, so I wouldn't't point to any particular issues or topics, but yes, we certainly take that into effect when we're looking at technology investments.

COMMISSIONER LEHMAN: So that is an important part of Apple's decision-making process.

MR. LeFAIVRE: Yes. I think that's probably true for all companies now.

COMMISSIONER LEHMAN: Thank you very much.

MR. LeFAIVRE: Okay? Thank you.

COMMISSIONER LEHMAN: Next I'd like to call Mr. Tom Lopez who is President of the Interactive Multimedia Association.

--o0o--

TOM LOPEZ

INTERACTIVE MULTIMEDIA ASSOCIATION

MR. LOPEZ: Thank you. Mr. Commissioner, my name is Tom Lopez. I'm Chairman of Mammoth Microproductions of Seattle, Washington, a multimedia development company. I'm also President of the Interactive Multimedia Association. The Association's general counsel, Brian Cann, who also directs our Intellectual Property Project, is with me today to help answer any questions that you may have.

The Interactive Multimedia Association is a 290-member trade association headquartered in Annapolis. We are here today because the patent system has cast a cloud over our emerging industry, an ambitious and motivated industry which seeks to transform the way we play, learn, work, think and communicate. We're specifically concerned about the impact of patents on the flow of information and fundamental principles of free expression, on the impact of patents on enabling environments, in particular the development of the national information infrastructure, the need for a patent system that is publicly accountable and open to industry input specifically through pre-grant publication and peer review, knowledgeable and informed about its operation and its economic and social impact, and sensitive to competing values and policies.

We bring a unique perspective, because our membership spans the whole of the multimedia industry, from large computer companies to small publishers and developers. This makes it impossible for the IMA to take positions on issues such as the merits and proper scope of software patents where we encompass many different views. However, we have historically been especially concerned with needs and perspectives of developers of multimedia also known as content-driven software. Multimedia developers provide the creative spark that is driving multimedia into homes, schools and businesses. Therefore, we do not address competition within the software industry, we address the impact of patents on content, on the organization, expression and communication of information.

Multimedia developers depend upon computers, networks and operating systems, authoring tools and other software environments. They build on technological platforms developed by others. Like traditional publishers, they add value through research, selection, organization and coordination, by aggregating rights, by creating original material and by expressing whatever ideas they believe will move the market, the body-politic or the soul. They use interactivity as their grammar. It is how computers speak to people, it is how people speak to computers. It is how people speak to other people through computers.

Historically, copyright law has provided a level of protection to the software developer. Unlike copyright, patents control the private use of patented processes. Unlike copyright, independent creation is not a defense to patent infringement. Patents therefore control not only original implementations, but also the users of such original implementations. Patents even control the use of products of the process. The extraordinary power of patents resonates across an increasingly-integrated and interdependent digital environment, putting everyone downstream of the underlying technology at risk. Content-integrators, publishers, distributors, even users. Indeed, typically end-users are the direct infringers. The upstream providers are technically only contributory infringers. For example, in the recent case of the Optical Data patent, interactive method for the effective conveyance of information in the form of visual images, the direct infringers were the hundreds of thousands of teachers in classroooms, and by extension the local school districts and all of us as taxpayers.

The Computer and Business Equipment Manufacturers Association described this problem sixteen years ago in arguing against patents for algorithms. Quote. The computer has become the engine which assists in running our society and in the future will assist man in numerous areas totally unrelated to the usual application of today's computers. These applications and computer uses should not be clouded by problems resulting from unwitting infringement by computer users. End quote.

Looking at the list of speakers today, it is clear that users are not represented at this hearing except for multimedia developers. Multimedia developers are on the front lines of the user community, because they're developing content-driven product and services. To the extent that they are successful, they become targets for patentees. Content-oriented developers get protection from copyright, not from patents. They need protection against patents. How do they get it?

For the first time, errors and omissions insurance to cover patent infringement is available from the American International Group for multimedia products. The cost is fifty thousand dollars per product, with a fifty thousand dollar deductible. That's a formidable barrier for an independent developer, a regressive tax on interactive expression. Such insurance, costly as it is, does not cover patents of which you are aware you may be infringing. This unfortunately is another good reason, along with the threat of triple damages for wilful infringement, to avoid reading patents entirely, a sad comment on a system originally intended to spread technical knowledge.

And while we are concerned with the impact of patents on publishing and First Amendment values, we share with others a concern for the related problem of patents on broad abstract processes. Such patents are extremely difficult to interpret. They often purport to preempt basic functionality so as to preclude others from designing around the patent. Usually their claims are over-broad, but can be narrowed only at great expense to those who would challenge them.

These abstract system-level patents threaten the development of common standards, specifications and architectures, including our Association's own work on

cross-platform compatibility. They create information bottlenecks, or tollbooths, in the vision of a national information infrastructure. Mindful of the history of blocking patents in the development of the radio and the aircraft industries, we note that the highly complex and interrelated nature of the information infrastructure makes it very vulnerable. Broad patents are especially suspect in the case of software where functions can be implemented in a wide variety of ways and where independent creation is commonplace.

This problem, along with the threat that patents still pending may be inadvertently incorporated in standards or infrastructional systems, would be greatly alleviated by pre-grant publication. The secrecy of the present application process is an anachronism, and a primary cause of the present uncertainty and insecurity. We plan to address this issue further in the February hearings on examination processes.

We're grateful to the Commissioner for holding this public hearing and dealing openly with the issues as a matter of public policy. We're also pleased to see a serious effort to develop patent policy within the larger context of economic development and the Administration's vision of a national information infrastructure.

We would like to close by expressing our support for a strong patent system. By that of course we do not mean a system that in the name of incenting novelty oozes uncontrollably into every corner of human life. We mean a system that knows its limits, that functions spectacularly within those limits and that does not debase the concept of intellectual property by incenting gaming and speculation. We mean a system that works in the real world, that acknowledges its regulatory nature and is tailored to the economic characteristics of the operating environment. We mean a system that operates proudly in public view, that is understood and acclaimed not only by patentees, their agents and their attorneys, but by the tens of millions who also contribute to our economy and society and face tough competition unarmed by patent monopolies.

Thank you very much.

COMMISSIONER LEHMAN: Thanks. You said that you felt that with the exception of the Interactive Multimedia Association that we didn't have on our witness list real representatives of users of the system. Who are we missing? Who would you count in that category of users?

MR. LOPEZ: For instance, teachers who use the products of multimedia developers, people who are users in the home who I think would be very upset to find out that they perhaps are infringing on patents without knowing about it.

COMMISSIONER LEHMAN: How might they be infringing on, on --

MR. LOPEZ: If they are actually the people who are taking the actions which would be against the patent, as teachers would be in using the products that would violate the Optical Data patent as an example.

COMMISSIONER LEHMAN: Thank you. I also wanted to ask, just to clarify your position a little bit more, is your position that you would not be in favor of such a drastic step as doing away with patentability of software-related inventions.

MR. LOPEZ: No.

COMMISSIONER LEHMAN: So you really feel what we need to do is to reform the system to make certain that, that we have a clearer scope of patentability and that we have better procedures, primarily pre-grant publication for making sure we capture the prior art.

MR. LOPEZ: Exactly. I think as Mr. Clark has indicated in his testimony, one of the greatest problems for multimedia developers today is the uncertainty that exists, and when this uncertainty exists it inhibits the investment and the -- not only of intellectual energy, but also of capital.

COMMISSIONER LEHMAN: My colleague, Ginger Lew, who's our General Counsel for the Department of Commerce and on Assistant Secretary of Commerce has a question.

GENERAL COUNSEL LEW: In Mr. Poppa's testimony he mentioned the possibility for the need of mandatory API -- ADI, and I wanted to know if the Association had any position on that.

MR. LOPEZ: The question is does the Association have any position on mandatory APIs regulated by law? Brian.

UNMIKED VOICE: No.

MR. LOPEZ: No, we do not at this point.

COMMISSIONER LEHMAN: One comment I'd just like to -- maybe it's more of a comment; we have a little more time here, and you may have a response to it. We haven't focused very much on that, and I don't think our questions did, but a previous witness I believe it was indicated that they thought it was very important to have the Courts to flesh out the patent system. There are elements in patent law at the moment, for example, the fact that, even assuming we spruce up our examination process, right now you can go to the Court of Appeals through the Federal Circuit, and basically get de novo review of our Patent Office decisions. The Court can second-guess the patent examiner, judges who are not even remotely experts in a given technology.

Secondly, we have a number of legal doctrines, like the Doctrine of Equivalence, which some have argued cloud the certainty in the patent system, and I'm wondering if you have any comment about the impact of those on your industry. Do those kinds of problems that occur in enforcing -- understanding how the Courts will interpret a patent, do they create uncertainty that creates problems for you?

MR. KAHIN: We really haven't addressed the problems, the technical problems you've described, at that level. I think the concern in the judicial evaluation is more focused on a very high presumption of validity of the patent examiner's determination, and -- that carries over into the judicial system. So that once that determination of nonobviousness based on the referenced prior art is made, it can't be overcome except by clear and convincing evidence. So that that high presumption is a disincentive to challenging the patent in Court.

COMMISSIONER LEHMAN: Actually I think the situation is a little different than that. I think we have maybe a little bit the opposite problem with the Court of Appeals. The Court of Appeals for the 12th Circuit has de novo right to review the patent and I'm not sure that they always do use that clear and convincing evidence standard that you

discussed.

MR. KAHIN: But it's very expensive to get that far, to get to the Court of Appeals for the Federal Circuit. Most of our members have enough trouble getting to a patent attorney, let alone filing suit in a District Court.

COMMISSIONER LEHMAN: Thank you very much.

MR. KAHIN: Thank you.

COMMISSIONER LEHMAN: Next I'd like to call Mr. Paul Heckel, Acting President of Abraham Lincoln Patent Holders Association who is from Hyperracks, Incorporated.

--o0o--

PAUL HECKEL

ABRAHAM LINCOLN PATENT HOLDERS ASSOCIATION

MR. HECKEL: Thank you. If somebody can display these slides?

COMMISSIONER LEHMAN: Charlie Van Horn from our office will.

MR. HECKEL: Thank you.

COMMISSIONER LEHMAN: I think you testified at our hearings on harmonization.

MR. HECKEL: Yes, I did. And --

COMMISSIONER LEHMAN: By the way, for those of you who don't know, Secretary Brown issued a statement yesterday, or on Monday, I guess this is Wednesday already, in which he indicated that we would not at this time proceed with international negotiations which would require the United States to change to a first-to-file system. I think it's important to bring this out at this point because these, this process of obtaining public input does make a difference. There are some people who don't think it makes a difference, but we had hearings on the question of patent harmonization, and we heard public testimony, Mr. Heckel testified, he had a very strong position on that, which I recall was somewhat consistent with the position that the Secretary has taken here on our recommendation, and so we have changed our policy, and so these hearings can make a big difference.

I should add, just as a footnote on that, that that doesn't necessarily mean that we won't at some point reconsider the question of a change in our system, but we concluded on the basis of those hearings that we had that we weren't really getting a good deal, that we weren't getting harmonization, and that the disadvantages to the U.S. creative community were not outweighed by the comparable advantages that we would receive as the proposed harmonization treaty was presently constituted. So this is a serious exercise, and thanks for joining us again, Mr. Heckel.

MR. HECKEL: Thank you, Commissioner LEHMAN, and I was there, and I felt at the time that it was very useful to get input from a lot of different people, and I feel that it's very good that you hold these hearings too. I think an awful lot of what has been spoken is really not supported by the facts, and I think it's useful for people to come there and to provide a reasonable basis for their opinions, because I think a lot of the time it doesn't stand.

Well, I'm Paul Heckel, and I'm here basically as Acting President of an organization called ALPHA, which is an organization of software patent holders. We only have about twenty members, but fourteen of our members are patent holders. I think ten or twelve of those actually had founded companies based on their patents. Two of our members were on the board of directors of the Software Publishers Association. Three of our members had their patents attacked by the League for Programming Freedom in several of their publications, me being one of those people. In fact, it was those attacks that really started to bring me in, to get interested in the issue, and as I suspect the Commissioner may know, I wrote two articles, one on the Communications of the ACM and one in Computer Lawyer on the software patent issue to try to bring out some of those facts and I'll bring out some of those facts later.

Clearly ALPHA strongly supports software patentability in pretty much all the forms that are there. We've also had an opportunity to look at the statements of the American Bar Association, the Software Entrepreneurs Forum and the Intellectual Property Section of the California State Bar and we concur in their positions as well.

Basically we feel we should have software patents. Inventiveness should be judged by the content of the invention and not by the color of the technology, as a variant on Martin Luther King's famous quote. We believe, by the way, that the quality of the examiner's position should be a more high-status position. We believe that trying to increase the pay and increase the professionalism of examiners is desirable. We all want a system which will make it clearer and less uncertain for everybody. Nobody, patent-holder or potential infringer alike, gets any advantage out of infringement.

Now I want to talk a little bit about some of our members because I know you're interested in personal experience.

For example, Mike and Susan Morgan found a company called MacInTax, developed a couple of products. Because they had patents on them they told me that, as Susan told me, she said, with her venture capitalists, when the venture capitalists asked us how we could protect ourselves against say Microsoft coming out with a competitive product and stealing our market, the fact that we had applied for patents put the problem to bed. It made the VCs feel much more comfortable, and that's a big difference. They have since sold out, they started another company.

Reed Hastings is another person. He founded a company here in Silicon Valley called Purer Software. He started, he made it profitable, he raised a couple million dollars from venture capitalists, the fact that he had patents made that possible; certainly it helped him a great deal. Currently he's facing a potential litigation problem with one of -- somebody in his market said, "Why don't we add his patented feature to our product?" and so he's having to deal with those problems.

Another is Dr. Marcusson who is a patent-holder and a physician. His -- when Oracle recently announced its product for the Information Superhighway they used something that he had designed for teaching medicine. It was called "Salvaging a Patient with a Stab Wound to the Heart. It was running on a Hypercard-like environment. He's had a lot of experience with inventors. He's a specialist in repetitive-strain injury, so he's familiar with that controversy which is going on. But what he has said is that, "I have seen first-hand emotional and financial damage done to

independent inventors whose inventions are ripped off by big companies," and he said he "fears that many small inventors will be the roadkill for the Information Superhighway," which is the talk that I take, and having heard the previous talk I'm concerned that if their position is taken that it could very well happen that way.

Hal Nesley is actually an investor, but he's invested in four start-ups which have software involved, one has a patent, the other three are in the process of getting patents.

So those are some examples. In my own case, I started a company relying on patents. It gave me more confidence to start the company since I had the patent or was going to get the patent, I thought, and it gave my investors confidence. They told me it was one of the reasons they decided to put money into it. We brought products to market, as did the other people that I have been talking about, and we then found out we were infringed by Apple, we got in some litigation and I've described it in my book that some people here I'm sure are aware of, and we settled and they took out a license. Then we got involved with IBM; that situation still is not clarified.

But I guess I'd like to go into what I really found out when I examined the patents that Mr. Stahlman attacked in some of his articles. And I particularly refer to the ACM article. If we can have that slide now. I went and I called. They gave an example of nine patents, and I went and I called the patent-holders on each of those nine patents. I found out some interesting things. I want to refer specifically to it. That chart is in Computer Lawyer, and this afternoon I'll have copies of Computer Lawyer out there for people to look at so they can see the chart. But based on that chart we have some conclusions.

One. All nine patents protected commercial products. Every one that they brought up that they said is an example of a bad patent and absurd patent. Two. Software stimulated new business formation. Four of those nine patents were held by companies that were started precisely to develop the technology that was in the patents, and a fifth company had only been in business for two years when it filed the patent. So five of the nine companies really were independent small start-up phase companies that were using patents. Okay? I think that that's strong evidence, based on a sample selected by the people who are condemning patents that software does stimulate new businesses.

Second, I would argue that they stimulate the introduction of fundamental technology. I think three of those patents introduce technology that was fundamental, at least in the sense that it was widely seen throughout the industry, and I'll talk about one of those later. By the way, I've talked to several inventors in different technologies, and I referred to a lot of that in my Harmonization testimony, and I found out in many ways that the problems faced by software developers or software inventors are very similar to the problems faced by inventors in other technologies. They're made more severe by the prior art problem and the newness of the technology, but fundamentally they're very very similar problems, and the way the trade system treats them is very similar.

My last point is an interesting one. Can I see the next slide, please? Small entities are exceptionally cost-effective in encouraging innovation, especially compared to Federal funding, and I will give you the example. It might be a little hard to see there, but if you look down the first column we have the number of commercial products. The first item is four for large entities. Next is five for small entities, and below that we have zero for Federally-funded. None of the nine patents cited a product that had a Federal patent behind it, and as you know, if you develop something under Federal law funding, you can get patents on it, you do have rights to use those patents in the commercial marketplace.

Now I looked at what I call the efficacy of the invention, and I used the fact that somebody has asserted a patent as a measure of efficacy, because a lot of patents aren't asserted, and I found two of the large company patents were submitted, and all five of the small-entity patents were asserted. So I use that as a measure of effectiveness, because we're going to look at taxpayer cost effectiveness.

Now if you look at Federally-funded we gave one there just so you don't have a number of zeroes, so the numbers work out in some sense.

Now we looked at the cost, and in 1989 the Federal funding of the Patent office was two million dollars, and so we allocated those costs and we got thirty, fifty thousand -- I can't quite read those numbers there, for those numbers, and then we divided to get the efficacy. By the way, the Federal funding of computer science in that year was four hundred and eighty-seven million dollars. So if we look at the cost-effectiveness of it, and the large entities had a cost-effectiveness thirty-three thousand, the small entities had a cost-effectiveness of two hundred and fifty thousand and Federally-funded had a cost-effectiveness of one point o three. Which says that a dollar spent in the tax -- to help the Patent Office really brings back more innovation. Now clearly if the Patent Office was clearly funded, the numbers would probably knock down to something like thirty-three and two hundred and fifty, which is still a very large number compared to one.

Now I fully recognize that this is only nine numbers. It's a very small sample, but remember, these numbers were picked by Mr. Stahlman and the League for Programming Freedom to say that it's bad for innovation, and there's a very very strong prima facie argument that it does encourage innovation. So those were the results of those numbers.

I want to talk now about a specific patent, which is the spreadsheet patent that I'm sure a lot of people have heard about. It's been described as the automatic recalculation patent, and when first suit was filed on it in 1989 it was attacked widely in the press as obvious and it was well-known in the prior art and stuff like that. By the way, I called the inventor, I got a copy of the patent, and I said, Who's talked to you? Nobody in the American press had even called this person although widely his patent was attacked in the press, and it was clearly easy to find him as all you had to was get a copy of the patent. So it doesn't give me a great deal of confidence when I hear these press stories about these horrible patents.

So since then I've learned a certain amount about the patent. In my opinion, that patent is to the modern computer spreadsheet what the Wright Brothers' invention was to the airplane. It might not have had a visual display; they used a teletype terminal. They started out with a concept of Basic, and instead of executing the statements in the numbered

order, they said, Why not take the statement numbers, break them in two, use both halves as indexes into an array, and then calculate the formulas in the order which is natural, and use it to solve business problems. That seems quite clear from reading the patent. They developed a product, they brought a product into the marketplace, and they had real users; okay? But they had a problem with the patent system. By the way, they filed the patent in 1970, twenty-four years ago, they have yet to see dime one for an invention which is in many ways responsible for the success of Apple, because VisiCalc helps out Apple Computers, the success of Lotus. They have yet to receive dime one.

This is what happened to them. They got a Notice of Allowance from the Patent Office. Then the Benson decision came down, and then the Patent Office took their patent away from them, because of the Benson decision. They then appealed it, pro se, through the Courts, and got a decision at the CCPA called in re parto, which says that just because the inventiveness is in an algorithm or in the software does not mean it's not patentable -- that's an important decision as I'm sure everybody involved with software patent knows, it was done pro se without an attorney by those inventors. Now they are in trial, and in July I went to hear a one-day --

COMMISSIONER LEHMAN: Yeah, I think, Mr. Heckel, I think we're going to have to --

MR. HECKEL: Turn me off?

COMMISSIONER LEHMAN: Yes.

MR. HECKEL: Okay, I'm sorry, can I just briefly --

COMMISSIONER LEHMAN: Is our machine on? I'm not sure if it's working right.

VOICE: It was on but we gave you a few more minutes because we tied up in the beginning.

MR. HECKEL: I'm sorry. I just want to say that I saw in Court their patent attorney in my opinion perjure himself on the stand to testify against his clients, to save himself from a malpractice suit. I saw that in July. The decision hasn't come down. I hope when the decision comes down you read it, Commissioner, Examiner, and look at that patent lawyer and consider whether or not this is what you want to have representing clients out there in the field.

Thank you very much, Commissioner.

COMMISSIONER LEHMAN: Thank you very much, Mr. Heckel. You know, we do have a procedure in the Patent Office for hearing complaints against people for not carrying out their professional responsibilities, so it's certainly available to people if they wish to use it.

Next I'd like to call Mr. Robert Kohn, the Vice-President and General Counsel of Borland International.

--o0o--

ROBERT H. KOHN

BORLAND INTERNATIONAL, INC.

MR. KOHN: Thank you, Commissioner LEHMAN, for the opportunity to testify today. I'm Bob Kohn, Vice-President of Corporate Affairs of Borland International, a leading developer and marketer of desktop and client-server computer software including D-Base, QuattroPro, Paradox, InterBase and Borland C++. I worked in the entertainment

and computer software industries my entire career. My experience in the software industry includes many types of application, utility software for both mainframe and desktop computers. After a brief period of private practice and as Associate Editor of the Entertainment Law Reporter, I joined the legal department of Ashton-Tate Corporation in 1983. Until its acquisition by Borland in 1991, Ashton-Tate was one of the world's largest computer software companies. In 1985 I left Ashton-Tate to become Associate General Counsel to Kandell Corporation, a leading supplier of IBM mainframe software, and in 1987 I joined Borland as General Counsel.

I want to emphasize that I am sensitive to the need for the intellectual property protection on both a professional and personal level. My professional career is focused on protecting the valuable intellectual property assets of software companies. I'm also an author myself, having recently written a reference book on music licensing that was published by Prentice-Hall. To call order 1-800-223-0231. So I can certainly appreciate the need to protect intellectual property. And I hope I've made my point. If you need the number again I'll have it available.

COMMISSIONER LEHMAN: Be careful, you know, works of the United States Government are not copyrightable, so if you get your stuff involved with ours you might have a problem.

MR. KOHN: I'll try not to read my -- I'll try not to read my book into the record.

I'm testifying today in my capacity as Vice-President and General Counsel of Borland, a publicly-traded Silicon Valley company. On behalf of Borland I want to comment specifically on Question 4 in the Hearing Notice, and if time permits more generally on questions regarding the scope of protection for visual aspects of software programs.

Question 4 asks whether the present framework of patent, copyright, trademark and trade secret law effectively promotes innovation in the field of software. Like all other software companies, Borland invests heavily in both the creation and acquisition of new software products, and like other companies Borland needs strong government enforcement of existing intellectual property rights, especially in foreign markets, in order to protect its investments.

But it is particularly unproductive at these hearings and at other forms for public debate on these issues to hear two extreme views espouse. One group, generally small companies, argue for no protection or perhaps at best very weak protection. A second group, generally very large companies, addresses the issue of scope rather than enforcement, arguing that broader protection for software is necessary, and indeed the broader the protection the better. We believe that much of the polarization you have heard and will hear is the result of a confusion of what is being debated.

Protectionist interests in particular confuse enforcement of what is an undisputed intellectual property right with the underlying scope of intellectual property protection. We in the industry all understand that software as a product is particularly susceptible to unauthorized duplication. We therefore need strong enforcement of existing intellectual

property rights to make sure that we are protected against the pirating of our software. But issues concerning the enforcement of intellectual property rights must not be confused with issues concerning the scope of intellectual property protection. It is too easy to wrap oneself in the proverbial American flag of antipiracy and anticonterfeiting enforcement. There is no dispute that strong antipiracy enforcement is required to promote the resources necessary for research and innovation. But it does not follow that because strong enforcement of intellectual property promotes innovation, a broader scope of intellectual property protection will also.

We should understand that many of those who very responsibly argue for limitations on the scope of intellectual property protection are not trying to defend pirates. They are, rather, trying to make a medium under which the proper scope of intellectual property protection as established by Congress and the Courts is respected and strongly enforced by the Administrative Branch of government.

This distinction between the enforcement of existing rights and a broadening of the underlying scope of protection was recently addressed at the 1993 Berkeley Roundtable on the International Economy in which the Vice-President, the Commerce Secretary and the Commissioner all participated.

The Report of the Roundtable on Maintaining Leadership in Software states the distinction between enforcement and scope very clearly. I'll include a block quote in my written testimony which begins with the following sentence: "Industry representatives argue that the importance of protecting intellectual property from theft by commercial counterfeiters and unscrupulous users must be distinguished from issues concerning the proper scope of intellectual property protection."

Unfortunately Question 4 in the Hearing Notice, in our view, heightens rather than diminishes the confusion and polarization. Question 4 seems to be based on a premise that strong protection for existing intellectual property rights necessarily implies a greater scope of intellectual property protection, and further that a greater scope of intellectual property implies a greater amount of innovation.

Implicitly, Question 4 neglects the important role that competition plays in encouraging innovation. We believe that the implication inherent in Question 4 should be the subject of much greater scrutiny and analysis. Within the industry we all, or at least most of us, agree that greater enforcement of intellectual property is necessary. What has fractionalized the industry is the attempt by some to use the need for greater enforcement to attempt to expand the scope of underlying intellectual property rights, particularly within the copyright area. As the Commissioner is aware, just two weeks ago the head of the Antitrust Division of the Department of Justice, Assistant Attorney General Ann Binghamen gave a major speech on the occasion of the sixtieth anniversary of the founding of the Antitrust Division. The Assistant Attorney General recognized the polarization within the industry that has been caused by attempts to increase the scope of intellectual property protection.

She said, "The substantive reach of the exclusive rights granted under the intellectual property laws also has been a matter of particular concern and ferment in the software industry. The Courts and the agencies have been faced with difficult decisions about the scope of both patents and copyrights in this field, as is clear to anyone who has paid attention to the long series of important court decisions on computer software copyrights, including Whelan, Altai, and the recent decisions in Lotus v. Borland, now under review in the 1st Circuit. The scope of copyright protection for computer software has we believe important competitive implications as well as important implications for the incentives to innovate."

We are particularly heartened to hear Assistant Attorney General most eloquently state her concern about attempts to increase the underlying scope of intellectual property protection. Again, please permit me to quote what she had to say. "Given my strong belief in competition, I think the courts should be hesitant to read the statutory grant provisions expansively, but should recognize the anticompetitive potential of restrictive practices at or beyond the borders of clearly-conveyed statutory rights."

While the Assistant Attorney General was directly addressing only the courts, we believe the same cautions should apply to the Administrative Branch of government as well.

Many questions to be addressed at these hearings deal with the visual aspects of computer screen displays. In evaluating the proper scope for protection for the individual aspects of computer programs, we believe that the Patent and Trademark Office would do well to consider the analytical framework employed by the engineers and computer scientists as opposed to the lawyers and judges in the software industry.

As the Commissioner is aware, much of the original and seminal work in graphical user interface analysis and design was done at Xerox Corporation in the 1970s. The research at Xerox formed a wealth of user interfaces far beyond just those of Xerox's products. Apple's MacIntosh and Lisa, Hewlett-Packard's New Wave, Microsoft's Windows, X-Windows, IBM's Office Vision and OS/2 to name just a few. Much of the research at Xerox was published in scholarly papers for distribution both inside and outside of Xerox. The most famous of those papers, entitled "A Methodology for User Interface Design," was published by Xerox Palo Alto Research Center in January of 1977. Because of the enormous importance of this paper, I'm going to attach it to Borland's written comments and ask that you consider it as part of these proceedings.

The Xerox research produced a methodology of interface design that is based upon what Xerox researchers called a taxonomy or classification for user interface analysis. This taxonomy is designed to permit analysis and evaluation of what each aspect or component of a user interface does. The taxonomy was created for software analysis and not for any legal purpose, but remarkably it dovetailed seamlessly with the overall intellectual property scheme of patents, copyrights and trade secrets established by Congress.

As the Xerox research concluded, every user interface has three separable components; one, the user's conceptual model; two, the control mechanism or command invocation of the product; and three, the visuals, or the information display. The user's conceptual model is the abstraction selected by the software developer which users can relate to

the task they are trying to perform.

For example, the spreadsheet metaphor is the conceptual model that underlies Borland's QuattroPro line of products. Under our intellectual property scheme, the conceptual model of a particular piece of software would not be protectable at all except of course insofar as it may be protected by trade secret or under the terms of a contract or confidential relationship. The command invocation or control mechanism of the user interface is the mechanism that extracts the functionality built into the software. It is a set of actions and results defined in particular relationships to one another. Menu items and keystrokes are part of the control mechanism and were clearly identified as such by the Xerox research published in the mid-seventies. Indeed, the control component was originally called the command language.

Under the intellectual property scheme established by Congress, the control mechanism of the software product falls within the ambit of patent law, specifically utility patents. In order to secure utility patent protection over a control mechanism, an inventor should be required to satisfy the statutory requirements of novelty, advancement over the prior art and so forth. For example, if the user entered a database by first clicking on the picture of the door to simulate knocking, and then clicking on the picture of the door-knob to simulate turning it, the sequence of steps would be part of the control mechanism and must satisfy the rigors of patent examination if it is to be protected. If the command mechanism does not meet the rigors of patent protection, it should not be protected by any other form of intellectual property protection such as copyright.

Finally, in Xerox's terminology, there are programs of visuals. The screen displays of many sophisticated user interfaces have a truly separable visual or expressive component. Images that can be manipulated through animation techniques.

The Congressional scheme provides for protection of these visuals, and under both statute and the case law, the visual display of the computer program may be protected by copyright law if and only to the extent its artistic features can be identified separately from and are capable of existing independently of the utilitarian aspects of the software program. Note that the definition of computer program under copyright law is a set of statements or instructions to be used directly in a computer in order to bring about a certain result. The screen display is a certain result of the set of statements or instructions that comprise the underlying computer program and must therefore independently qualify as a work of authorship.

Those are my two paragraphs. Thank you for the opportunity to appear here today and I would be happy to answer your questions.

COMMISSIONER LEHMAN: Thank you very much, Mr. Kohn. I just note you refer to Question 4 in our Federal Register Notice which states that -- which asks the question, Does the present framework of patent, copyright and trade secret law, A, effectively promote innovation in the field of software, and, B, provide the appropriate level of protection for software-related inventions. I don't read those as implying that we should raise the level of protection; in fact I read those as an open-ended question as, What is the

appropriate level? and that may well be a lower level. It may be no level at all, and I think the questions we've asked would suggest that we do have an open mind about that.

MR. KOHN: I'm glad to hear that the Commission has an open mind about these issues. I think that, looking at the background section of the hearings, I don't have it in front of me, specifically emphasizes the innovation that's promoted by protecting intellectual property, and the point that I made is that there is absolutely no reference whatsoever to the importance of competition in promoting innovation, and you mentioned earlier, to an earlier witness, that -- you suggest that the competition issues might be more appropriately addressed under Antitrust provisions, but it is an intellectual property issue, and that's precisely what Ann Binghamen had said in her speech. It is an intellectual property program, we are after all talking about government-granted monopolies.

COMMISSIONER LEHMAN: Thank you very much. I'd like to take a five-minute recess before we reconvene for the rest of the morning's hearings, and our next witness, when we come back will be, I believe, Douglas Brotz from Adobe Systems.

(Recess)

COMMISSIONER LEHMAN: Next I'd like to call Douglas K. Brotz who is the Principal Scientist of Adobe Systems, Incorporated right here in the Valley.

--oOo--

DOUGLAS BROTZ

ADOBE SYSTEMS, INC.

MR. BROTZ: Good morning, Mr. Secretary and members of the Panel. My name is Douglas Brotz. I'm Principal Scientist at Adobe Systems, Incorporated, and I am representing the views of Adobe Systems as well as my own. Adobe is a software company based in Mountain View, California. We are most well-known for our PostScript language and interpreter which provides foundation for desktop and electronic publishing.

Although I am a computer scientist, I became involved in patents when Adobe was contacted by another company regarding Adobe's possible infringement of a patent. I'm currently Adobe's technical advisor to our patent attorneys.

Let me make my position on the patentability of software clear. I believe that software per se should not be allowed patent protection. I take this position as the creator of software and as the beneficiary of the rewards that innovative software can bring in the marketplace. I do not take this position because I or my company are eager to steal the ideas of others in our industry. Adobe has built its business by creating new markets with new software. We take this position because it is the best policy for maintaining a healthy software industry, where innovation can prosper.

The problems inherent in certain aspects of the patent process for software-related inventions are well-known, the difficulties of finding and citing prior art, the problems of obviousness, the difficulties of adequate specifications for software are a few of those problems. However, I argue that software should not be patented, not because it is difficult to do so, but because it is wrong to do so.

The software marketplace requires constant innovation regardless of whether the computer programs can be

patented or not. Indeed, the fundamental computer programs and concepts on which the entire industry is based were conceived in an era when software was considered to be unpatentable.

For example, when we at Adobe founded a company on the concept of software to revolutionize the world of printing, we believed that there was no possibility of patenting our work. That belief did not stop us from creating that software, nor did it deter the savvy venture capitalists who helped us with the early investment. We have done very well despite our having no patents on our original work.

On the other hand, the emergence in recent years of patents on software has hurt Adobe and the industry. A "patent litigation tax" is one impediment to our financial health that our industry can ill-afford. Resources that could have been used to further innovation have been diverted to the patent problem. Engineers and scientists such as myself who could have been creating new software instead are working on analyzing patents, applying for patents and preparing defenses. Revenues are being sunk into legal costs instead of into research and development. It is clear to me that the Constitutional mandate to promote progress in the useful arts is not served by the issuance of patents on software.

Let me illustrate this burden with some figures. The case Information International Incorporated v. Adobe, et al., was filed five years ago. Last year the trial court ruled for Adobe, finding no infringement. In December the Appeals Court for the Federal Circuit unanimously affirmed that judgment. Yet, in that time, it has cost Adobe over four and a half million dollars in legal fees and expenses. I myself have spent over three thousand five hundred hours of my time -- that's equivalent to almost two years of working time -- and at least another thousand hours was spent by others at Adobe. The Chairman of the Board spent a month at the trial. This type of company behavior would not be high on anyone's list of ways to promote progress.

This state of affairs might be acceptable if there were a corresponding benefit for patents in the software industry. However, I see none. Companies that have trumpeted their fundamental software patents are not leaders in software innovation. Conferring monopoly positions in an industry that was already the most innovative of all will promote stagnation rather than increased innovation. When companies turn from competing by offering the best products to earning money by the threat of patent litigation, we will see our best hope for job creation in this country disappear. An industry that still generates tremendous job growth through the start-ups of two guys in a garage will not continue to grow when a room for a third person, a patent attorney, needs to be made in that garage.

There does exist a perfectly adequate vehicle to protect creator's rights in this industry, the Copyright Law. The nature of software is that it is a writing, an expression of mathematical ideas. The copyright law protects this expression, and it does so without requiring costly and time-consuming proceedings. For people working in the fast-paced software industry, the way a copyright is created is idea. While feverishly working to meet deadlines, there is no need to explain what you've done to a government agency. The very act of writing the software confers the copyright on it.

Furthermore, the copyright law confers the correct level of protection on computer software. Regardless of what current regulations may say, the fact is that all computer programs express mathematical algorithms. Every part of every computer program manipulates numbers with logic. Any software that performs any task does so through mathematics. It is inconsistent to hold that mathematic algorithms are unpatentable while granting patents on systems composed of software.

If the Patent Office were truly following the law it would recognize the inherent mathematical nature of software and it would not grant patents to software-based inventions. In the last decade the Patent Office has been granting patents on software and algorithms regardless of superficial attempts to cast claims as systems methods or processes. The Supreme Court did not say in Diamond v. Diehr that pure software inventions are patentable. By adopting this position in its recent practice, the Patent Office has made a dangerous step that could decimate the very industry it wishes to protect.

Whenever the Patent Office grants a software patent, it grants a right to the patent-holder to devastate innocent businesses. Due to the arcane nature of this technology, our courts find it very difficult to distinguish frivolous software patent lawsuits from legitimate ones. As a result, a frivolous plaintiff is in a very strong blackmailing position, where a defendant can look forward either to an extortionate settlement or enormous legal costs. An excellent remedy would be to change our law to allow a successful defendant to recoup legal costs in patent cases. Until that day arrives, at least our Patent Office can refrain from granting these dubious patents.

We have heard today from proponents of software patents who will claim that these patents can protect the independent inventor. This belief is a delusion. The expensive patent process protects large, methodical corporations that can afford to apply for scores of patents much more than it protects the poorly-capitalized lone inventor, and when that inventor tries to produce his invention he may well find that those large corporations can ruin his own business with their large software patent portfolios.

In summary, these are my main points:

The software industry thrived without patents, creating its fundamental base in an era of no software patents; software patents harm the industry, with no corresponding benefit; software embodies mathematical algorithms; the law, starting with the Constitution, argues against patents for software-related inventions; and last, the proper form of protection for software is copyright.

As a postscript to the figures on the patent lawsuit that I discussed before, the final figure is actually not in. Although Adobe has been successful twice already, the plaintiffs are asking for reconsideration of the unanimous appeal judgment against them. These kinds of festering sores are what our country can ill-afford when we are trying to lead the world in creative industry.

Thank you.

COMMISSIONER LEHMAN: Thank you very much, Dr. Brotz.

You've indicated that you think that the copyright system works very well to protect software. An earlier witness, Mr. Kohn from Borland, indicated that he felt that there were serious problems with the existing copyright system, and in particular he felt that it shouldn't protect screen displays, for example.

Other witnesses have indicated that they're very concerned about, I believe, the witness from Storage Technology indicated that he was very concerned about the decompilation issue. He very much believed that one should be able in effect to copy software in the decompilation process in order to produce interoperable works. I'm wondering, since you really believe that we should focus on copyright, if you have views on either of those two issues.

MR. BROTZ: Yes. I certainly do. I agree with Mr. Kohn that we should not confuse strong enforcement of copyright rights with broadened scope of copyright rights. I agree that some plaintiffs have tried to stretch the scope of copyright beyond where it ought to go. I firmly support his position, in fact, that copyright law should protect us against piracy and the kinds of threats that copyright law was intended to protect us against.

In answer to your other question about decompilation and interoperability considerations, I believe that the evidence always cited for the importance of interoperability is that companies that do not provide for interoperability fall of their own weight. I do not see that as an argument for insisting that companies therefore make themselves interoperable. If strong rights are granted to all aspects of the written computer software, then a company could choose what level of protection it wanted and how far to assert its rights and whether they wanted to open their interface or not. If they make a wise decision and offer enough interoperability, they'll do well; if they make an unwise decision, they won't, and it's up to them to decide whether they want to succeed or not.

COMMISSIONER LEHMAN: In other words, your view is that the licensing system deals with this problem, that if people don't adopt intelligent licensing processes, then they will suffer the economic consequences which will be negative and will encourage basically licensing that creates more open systems.

MR. BROTZ: That's right. And I would oppose having a law that straitjackets the way in which these kinds of licenses or accesses must be made.

COMMISSIONER LEHMAN: Thank you very much. Does anybody else have any questions? Thank you.

Next I'd like to call Hans Troesch, partner in the law firm of Fish and Richardson.

--oOo--

HANS TROESCH

FISH AND RICHARDSON

MR. TROESCH: Good morning, Mr. Commissioner, distinguished Panel. My name is Hans Troesch. I'm here speaking on my own behalf. My partners have reminded me of that.

Many years ago I earned a Masters Degree in Computer Science at the University of Michigan, and for close to ten years, regardless of my various and more fancy job titles, I considered myself principally to be a computer programmer. Today I'm an attorney and a member of the patent bar. I practice patent, copyright and trade secrets law, as we already mentioned, with the law firm of Fish and Richardson.

I'm here today because I would like to offer my own views on a few of the questions that the Patent Office has invited the public to address at these hearings. As a preliminary matter, I must confess my own deep concerns about the present fate of software inventions in the Patent Office. I believe that the logical, almost musical nature of software technology provides unique opportunities for advocacy and for confusion in a system that is based on a more structural, may I say more sculptural view of the world, but that is a topic for another day.

Today I will merely state my hope and belief that the Patent Office will rise to the challenge of finding and keeping qualified examiners, securing access to the vast body of software-related prior art that is not of record in the Patent Office, and of developing delimiting doctrines of novelty, obviousness and enablement in ways appropriate to the peculiarly flexible genius of software technology.

I would like to address Question 1 at this point in the Office Notice, and to state my view that a computer program, that is to say, a set of instructions that is executable on a computer, to achieve a result, should be considered a machine within the meaning of Section 101 of the Patent Statute, and should therefore be eligible for patent protection, without resort to the additional and often redundant limitations to computer processors, read-only memories or data input-output devices.

On the issue of eligibility for patent protection, I dare say such a change in the form of the law would not greatly expand the scope of protection available to inventors, at least not to those inventors who can afford the kind of legal talent testifying at these hearings.

Those of us who know what we are doing can get computer program machines covered. The process we have to go through may be painful to watch, may be expensive, but we can do it. For that reason I would promote my suggestion principally as one that will improve the quality of the analysis of software-related ventures, and the doctrine under which those inventions are examined.

On the issue of infringement we would have to be a bit more subtle. If we allow claims to be made to computer programs per se we must be careful not to create a risk of infringement by traditional print media and their successors in electronic publishing. The publication for human readers, whether or not on paper, for the patent-protected computer program, should not by itself be any kind of infringement of the patent.

I would like to turn now to one of the specific questions raised for today's hearing. What aspect of a mathematical algorithm implemented on the general-purpose computer should or should not be protectable through the patent system? I believe that a machine made up of computer program instructions that usefully transforms data or information should be protectable under the patent laws in all its novel and nonobvious aspects. Given the importance of information processing to our economy, it would be perverse for us to continue to deny direct protection to a

technology that is so important to our information processing prosperity.

This leads me unavoidably to the question of what is novel and unobvious in a computer program. I believe that our greatest challenge lies in these two questions, regardless of how we answer the question previously posed.

For myself, I would not consider novel and nonobvious merely to transpose to a computer something previously known to be done by hand, or in one person's head, or collectively by a group of people. But if the method for transforming information is truly new, then the doctrines that limit or preclude protection solely because the method is a computer program seems unwarranted.

It has been suggested that allowing mathematical algorithms to be protected would remove laws of nature from the public domain and give an unwarranted universal scope of protection to a patented technique. Personally I find those rationales peculiar. Taking the computer programmer's informal definition that an algorithm is a predetermined set of steps to perform a function, and that a mathematical algorithm is one that operates on mathematical objects, such as numbers, triangles, continuously differentiable functions, then granting protection for a new, previously-unknown and nonobvious set of steps withdraws nothing from the public. And if the patent reaches over a broad range of applications, that would merely correspond to the broad usefulness of the new algorithm.

In any other technology this would be grounds for praising the inventor, not for denying protection. I would submit to you that if someone were to object to a patent on the transistor on the ground that it would have too many uses, you would find that objection incomprehensible.

One final point that might be kept in mind before whence it comes to an alarm about the potential breadth of claims to mathematical algorithms; a naked mathematical algorithm claim would seem to be the ultimate engineering claims, and therefore particularly susceptible to being rejected or invalidated, because any prior art that shows the algorithm steps being applied in any context would invalidate the claim. Personally I would be surprised if any patent practitioner would ever rely solely on a naked algorithm to protect his client's interests.

The Patent Office also poses the variant of this question, limiting it to the implementation of the algorithm to a special purpose rather than a general purpose computer. If the problem is bad patents, this does not seem to be a solution. If one begins with a computer program that should be unpatentable because it is not new, or because it is obvious, one should not in my view be able to achieve patentability merely by attaching to the program the input-output devices that are conventional for the process that the computer performs.

In other words, one should not be able to save an old or obvious bread-baking program merely by attaching a digital thermometer to it. Conversely, if the program is new and not obvious, then the conventional addition of necessary computer hardware and other devices is redundant to the claim, at least insofar as patentability is concerned. Such additional limitations would not in fact limit the scope of protection available to the inventor, unless parenthetically

the claims are poorly drafted. But such a redundant edition of apparatus to the program claim does create a potentially substantial distraction for the patent examiner who, in the terms of my example, in searching the art of digital thermometers, may completely miss the point about bread-baking.

Finally, I would like to say a word about whether we should replace patents with a new form of protection for computer software. My one-word answer is no. Patent law can deliver predictability, definiteness and uniformity. Under copyright law we cannot protect your ideas, at least not without doing some violence to traditional copyright principles, and we are subject to forum shopping in thirteen circuit courts of appeal. Under trade secrets law, we can protect our ideas, but are subject to the law as developed in any of fifty different state courts and their Federal counterparts. We're never quite sure what the protected ideas are and are at risk of having someone rediscover or reverse-engineer our inventions out from under us. But under the patent laws we can get warning about what is protected expressed with reasonable clarity and applied with national uniformity. It would be unfortunate if such a sound concept were to be crippled because we were too slow in learning to apply its fundamental principles to the challenges of software-related inventions.

Thank you.

COMMISSIONER LEHMAN: Thank you very much, Mr. Troesch. That was very helpful.

The next person on our list is Brett Glass, but we're not sure that Brett Glass is here. If you are, will you please stand up and identify yourself and come forward? If not, we will move on to Robert Sabath, President of the World Intellectual Property and Trade Forum.

--o0o--

ROBERT SABATH

WORLD INTELLECTUAL PROPERTY AND TRADE FORUM

MR. SABATH: Mr. Commissioner, distinguished Panel. My name is Robert Sabath. I speak today in both my capacity as President of the World Intellectual Property and Trade Forum and as a solo practitioner in the patent field. I'm also on the Executive Committee of the California State Bar's Intellectual Property Section, but as you know, Mary O'Hara and Michael Glenn will testify at these hearings on behalf of the State Bar. Additionally I speak as Legal Issues Editor of QuickTime Forum, a multimedia developer's publication.

The primary topic today is the use of the patent system for the protection of software-related inventions. The central objective of my remarks is to encourage greater flexibility within the framework of the law in promoting the patenting of software-related inventions as well as pure software inventions.

Patents themselves are the best prior art against subsequent applications for a patent grant. Anything that artificially limits the development of the body of prior art relied upon by the patent and trademark office has the effect of slowing the progress of technology in critical fields. Software is clearly a key and strategic industry for the United States. It's no secret that software itself in the development of the industry were not caused by the patent incentives, but still, the patent

system is part of the incentive structure which is necessary to the continued development of many software firms.

Moreover, the efforts of the United States Government to promote U.S. trade interests abroad and even to advocate changes in the intellectual property laws of other countries are severely undermined if the U.S. intellectual property laws and regulations fail to encourage successes of key U.S. industries at home. One such key industry is clearly software.

I do ask for your indulgence, Honorable Commissioner, in addressing a slightly broader question than the primary topic indicated above. As a sole practitioner I've come close to the plight of the solo inventor affected by the substantially increased PTO fees promulgated by prior administrations. Particularly the maintenance fees are believed to be a disincentive which may dissuade individuals from even initiating the process of obtaining patent protection.

But the cost of patenting which is born today by inventors and companies is softened by the Silicon Valley spirit of self-help which has characterized the American spirit since the days of George Washington and Thomas Jefferson.

An example of this self-help is the Sunnyvale Patent Information Clearinghouse. Self-help, and necessity, have additionally spawned in Silicon Valley a substantial venture capital community which is selectively supportive of the efforts of individual inventors. This spirit of self-help is additionally shown by many local firms and companies which have opened offices in Washington, D.C., and its surrounding communities of Virginia and Maryland.

We do salute you, Mr. Commissioner, and distinguished Panel for coming here to California. We clearly need your help, not just with regard to improving the laws and the regulatory environment as it relates to patents, but also with regard to the infrastructure in which patent and invention processes play themselves out in the United States.

The U.S. Government has facilities, buildings and courthouses throughout the nation. These facilities and buildings have many purposes. Federal courthouses now hear patent lawsuits in San Jose as well as in San Francisco, and in cities throughout the country.

It is clear that our country has developed elaborate mechanisms for facilitating and resolving disputes between litigants and patent lawsuits. However, we have done pitifully little at the Federal level to enable the solo inventor to search for prior art and effectively to limit the scope of claims to his fields or her fields of rightful entitlement. Mr. Commissioner, accordingly we're very happy to have you here today in this convention center.

We believe as a minimum the West Coast deserves a branch of the USPTO having at least search facilities to support the software, the semiconductor and the electronics industries that have developed the infrastructure of the American West so extensively. Perhaps the availability of public search rooms for inventors is not a matter for the Department of Commerce, but rather for the Department of Education. But whether the Commissioner of Patents and Trademarks takes the initiative or whether another Federal agency takes the lead, it is clear that many communities in our country need access to the technical collections and patents of the Federal Government.

The facilities for obtaining prior art in Sunnyvale are clearly needed. But in most communities of America, such facilities are nonexistent. Moreover, solo inventors are seldom in a position to invest in a state-of-the-art CD-ROM system or computer search services in view of their high cost. The Information Superhighway offers a bright vision of a technological future. Will there be facilities to provide public access to information carried in this superhighway?

The Patent Office can provide such facilities to bring the fruits of this superhighway of information to our inventors, to the young in America who thirst for knowledge and progress, and to the public at large.

The physical facilities of the Department of Commerce and the United States Patent and Trademark Office are needed in our local communities to implement the purposes of the intellectual property laws of our country. America wants to build its future by educating the inventors of the future. We need public search facilities for the electronic arts wherever major electronics developments are being made, in California, in Austin, Texas, in Dallas, in Colorado, along with many other communities across America.

We need biotechnology search facilities in Emeryville, California, and in Cambridge, Massachusetts, and in other communities of the nation, and we need software search facilities in the nation's software development centers including but not limited to such areas as Seattle and Silicon Valley. The German example for one shows that search facilities and examination facilities need not necessarily be located in the same cities.

The resources and the facilities of the PTO should be distributed at various locations across America to provide public access to the prior art regarding technological developments which have already become known. We salute you, Mr. Commissioner, for coming to California to address the vital subject of patenting software-related inventions in this public forum. The California economy is improving, but it remains disastrously understimulated. Because of the size of California's economy it can either drive or hamstring recovery on the national scale.

The questions raised at this public hearing have a direct and vital bearing on the economic well-being of California. We thus appreciate your coming to guide these hearings.

To focus more definitely on the subject of patenting software-related inventions, it is my belief and that of many participants in the World Intellectual Property and Trade Forum that there is no substitute for the development of an increased body of software art available to patent examiners. With a properly classified and complete body of prior art the searching and the examination of new patent applications will be enhanced.

The World Intellectual Property and Trade Forum salutes the corrective action of the Commissioner in connection with the reexamination of Compton's multimedia patent. This reexamination process clearly shows that even though applicable prior art was not initially available to the examiner, there are mechanisms for addressing questions of patentability even after grant of a patent, but certainly it would be optimal if the applicable art had been found and addressed during actual examination.

One way to ensure an effective and complete body of prior

art in the field of software patents is to relax the policies of the PTO with respect to the patentability of mathematical algorithms. Considerable room for relaxation is available even within the bounds of current case law on the subject. Many new inventions beneficial to the developing field of software may currently not even be the subject of patent applications because of the chilling effects of the PTO's restrictive approach to the patentability of software.

Insufficient software prior art limits the ability of the PTO to examine effectively future software-related patent applications, including pure software patent applications. The distinction between hardware and software approaches to the same problems has blurred technologically. This distinction should be blurred and eliminated in the bureaucratic spaces of the Patent and Trademark Office as well.

Patent examiners should rely less heavily on the Section 101 as a basis for rejecting software-related inventions. With an increasing body of prior art established by greater flexibility in allowing software-related patents, examiners will be encouraged to make substantive office actions based upon technical art rather than merely implementing policy articulated by agency representatives.

One object of the patent system is to encourage progress in the arts by publication of inventions. The effect of patent grant is to add to the body of detailed technical information comprised in issued patent documents. When a Section 101 rejection is successfully asserted by the PTO, the practical effect is to deny future software developers of the benefit of full patent disclosure. This hampers the development of new ideas in many technical fields, including multimedia software generally and even biotechnology which is to an extent dependent on progress in the field of data systems for its instrumentation to be effective.

We thank you again, Honorable Commissioner, for coming here and conducting these hearings.

COMMISSIONER LEHMAN: Thank you very much, Mr. Sabath. I just point out that we do have patent depository libraries all over the United States and I think there are several on this area, probably one at Stanford and Berkeley, and they actually have everything we have in the Patent Office there, and that is available to the public. In addition, we're automating the Patent and Trademark Office, and in fact right now if you're in Arlington you can go into the -- in fact, Group 2300 is fully automated already, and if you go into our patent search room in Washington, we've got a facility there where you can actually get computer retrieval of the patent documentation and we have plans to extend that to the patent depository library so that you'll be able to come out here and do the same thing. And eventually, hopefully in a few more years, we'll actually have this service available through the Internet. We're not quite there technologically yet, but every engineer and computer scientist in Silicon Valley will just be able to, in a few keystrokes, get access to our patent database. We think that's not only going to help people understand what the patent prior art is but hopefully it will give them access to some of the technology more easily than they otherwise would have, so we are indeed, I think, making some progress on that problem.

Thank you very much.

MR. SABATH: Thank you.

COMMISSIONER LEHMAN: Next I'd like to call Mr. William Benman of Benman & Collins. And Mr. Benman is the last speaker for this morning. We're pretty much right on target on our time. Thank you for joining us.

--o0o--

WILLIAM BENMAN

BENMAN & COLLINS

MR. BENMAN: And as someone that hasn't had breakfast, I assure you that my testimony will be brief!

Commissioner LEHMAN, thank you for hearing me on the question of the patenting of software-related inventions. I commend you for taking testimony on this important topic.

I'm a partner with the intellectual property law firm of Benman, Collins and Sawyer, with offices in Palo Alto, Los Angeles and Tucson. We have prepared and prosecuted numerous applications on software-related inventions, and we offer this testimony in the hope that our experience in this regard maybe of some value in connection with your effort to sample public opinion on this topic.

Since the early 1980s, we in the community of patent lawyers who are preparing and prosecuting patent applications on software-related inventions observed two thrusts, one in the direction of copyright protection as the primary if not sole form of legal protection for software, and the other in the direction of patents where patent protection was available. We notice that those companies that were patent-conscious were most likely to exploit the patent option. Many of the companies which did not utilize the patent option were often new or young software companies which were not patent-conscious per se. The plight of these companies was compounded by their reliance on counsel from lawyers that were typically not patent lawyers with experience in the field. These lawyers held themselves out as experts in computer law and high-tech law, but typically held a mistaken notion that software was not patentable, although patents had already been issued for software-related inventions. To this day, many uninformed attorneys either advise their clients that patent protection is not available for software, or they criticize the patent system as too costly and cumbersome for the protection of software. As a result these attorneys have either generally steered their clients away from patents as an appropriate form of protection or neglected to advise their clients to seek counsel from an attorney having experience in the patenting of software. Hence, for many years now, many software publishers have been lured by the ease and low cost of copyright protection without being properly advised of the shortcomings of same nor the availability and advantages of patent protection.

As early as ten years ago we began to suspect that the copyright holders would someday become aware of the limitations of copyright protection when it became necessary to enforce the copyrights. We expected that they would seek a more substantive form of protection such as that currently afforded by the patent system. This has now come to pass as the software industry has become aware of the shortcomings of copyright protection often as a result of painful and costly court battles. The lawyers guiding these firms away from patents have now essentially painted

themselves into a corner, having failed to stretch the scope of copyright protection to cover substantially more than the form of the software and not the underlying inventive aspects thereof. These attorneys have now marshalled some software publishers behind their effort to escape from the situation by criticizing the current patent system as unwieldy for software, advocating instead some new form of protection for software that would protect the underlying concepts without a rigorous examination process. They are supported in their efforts by recent admissions by the Patent Office of a shortage of prior art in PTO files to facilitate the examination of software-based applications. It would appear that many now desire the ease and low cost of the copyright system with the substantive protection afforded by the patent system.

It would clearly be adverse to the public interest, in our view, to grant such monopolies without a substantive examination of the application. Yet, in our view, any system which involved a substantive examination of such applications would be essentially equivalent to the present system. Hence we seek to speak for those attorneys who are registered to practice before the Patent and Trademark Office and experienced in the protection of software in expressing our view that the present system of examination for software-related inventions and innovations is, though not perfect, adequate and appropriate for the protection of software.

First we feel compelled to mention that many of the concerns in the industry would be adequately addressed by educating the industry about the patent process by those qualified to do so. We believe that it is unethical for any attorney to advise a client not to seek patent protection for inventions whether software-related or not, if the attorney giving the advice is not skilled in the field.

The process by which attorneys are admitted to practice before the Office is intended to protect the public interest by ensuring that those that hold themselves out as qualified to practice before the office are indeed so-qualified. Unfortunately the software industry has grown up on an unhealthy of poor counsel with respect to intellectual property issues pertaining to software. It is no wonder then that it now has a bit of indigestion and needs some relief. We feel that the present system of examination if properly administered can provide that relief, and allow me to briefly address some of the perceived problems with the present system.

One of the perceived problems is that the Patent Office will issue patents on inventions already in the public domain. That's been touched on by some of the speakers earlier this morning. Another problem is that the patent process is costly and therefore available only to big companies, and a third often-heard complaint is that the patent process is too slow.

First, with respect to the concern that the Patent Office will issue patents on innovations that are known and used by others in the industry, let me say that to the extent that this is a real problem, it is no different for the software industry than for other areas of technology. There are at least three levels of safeguards by which this perceived problem may be addressed by the current system. First, the Patent Office is currently in the process, as I understand, of providing the

examining corps with the capability to search beyond PTO files to the on-line computer-searchable databases, such as those accessed through the Dialog system. This will facilitate a more thorough examination of the prior art including technical literature, new-product announcements and et cetera, minimizing ab initio the probability of a patent issuing on an innovation which lacks novelty or is obvious in view of such art.

To the extent that the innovation is known by others but is not published and therefore inaccessible to this approach, the next level of safeguard should be considered. However, it should be noted again that software-related applications do not differ from other applications in this regard. The second safeguard is afforded by the current reexamination process. Those who are aware of prior art which might render a patent issued on a software-based application invalid may initiate a reexamination of the patent, and I commend you for your intent to improve that process. A third safeguard results from the practical realities of patent enforcement. The cost of patent litigation are sufficiently high that one holding a patent of questionable validity on software would think twice before instituting a court battle when advised of prior art which would invalidate the patent. Hence there are clearly several current safeguards to address this first concern.

The second concern, that the patent process is costly and therefore only available to big companies must be considered in light of the fact that in other areas of technology, patent filings by small companies and individuals are quite high. In recognition of the value of innovations provided by small entities and individuals, those that qualify are entitled to pay reduced filing fees. In addition, it will soon be appreciated that in the software industry, as in other industries, the total cost of procuring a patent, typically somewhere between five thousand fifteen thousand dollars depending upon what part of the country you're in, is small compared to the value of patents and certainly small compared to the cost of copyright litigation. While patents are more expensive to acquire than copyrights, the old axiom, "You get what you pay for," comes to mind.

Finally with respect to the concern that the patent process is too slow, we have a suggestion. You might consider an expedited examination on the basis of a higher filing fee for those that would like to see their patents issued quickly or be prosecuted quickly.

Our concerns as practitioners with the patent system relate to the manner by which the examiners are currently applying the PTO test, and the extent to which the current PTO test is out of conformity with the position of the Court of Appeals for the Federal Circuit, and in that regard I'd simply like to complete my remarks by saying that we're very much encouraged by the position taken by the Board with respect to the Veldhuis opinion, to the extent that that is in conformity with the Arrhythmia decision of the Court of Appeals for the Federal Circuit.

So to conclude, it's our opinion that what is needed in the system is an adoption by the patent office of a test that's consistent with the case law, a cadre of examiners that understand how to apply this test, and in this regard I should note that I think Mr. Goldberg has done an excellent job of training the examiners, but -- they're coming along, but with

a little more help from us practitioners on the outside I think they'll get it right at some point. But I would advise any company or individual with a software-related invention to consult with a qualified attorney, experienced in the preparation and prosecution of software-related patent applications with respect to the advantages and disadvantages of the various forms of protection.

With these elements in place, all concerned will recognize the viability of the current system of examination with respect to software-related inventions and innovations.

Thank you.

COMMISSIONER LEHMAN: Basically it's your position that we don't have much of a problem, right now, as I understand it.

MR. BENMAN: That's basically my position. I favor the current system. I think that when we get the bugs worked out, when you access the other sources of prior art, everybody will find that the system works pretty well.

COMMISSIONER LEHMAN: Do any of my colleagues have any questions or concluding remarks for this morning? If not, we will reconvene at 2:00 o'clock, at which time we will hear from Mr. Jerry Baker, Vice-President of the Oracle Corporation.

(Noon recess)

--oOo--

JANUARY 26, 1994
AFTERNOON SESSION

--oOo--

COMMISSIONER LEHMAN: Looks like we have a somewhat-dwindled group, but still an impressive audience for this afternoon. Thanks for sticking with us.

Our next witness, to start us off this afternoon, is going to be Jerry Baker, Senior Vice-President of the Oracle Corporation, and please accept our apologies, Mr. Baker, for our starting a couple minutes late. Do you want to come forward?

--oOo--

JERRY BAKER

ORACLE CORPORATION

MR. BAKER: Good afternoon, distinguished representatives of the United States Patent and Trademark Office, and members of the public. I am Jerry Baker, Senior Vice-President of Oracle Corporation and head of the company's product line development organization.

Oracle is now a one and one half billion dollar company employing over eleven thousand people worldwide. At Oracle we believe that patents are inappropriate means for protecting software and are concerned that the patent system is on the brink of having a devastating impact on the software industry. In our opinion, copyright and trade secret law is satisfactory to protect the developer's rights in software and to promote innovation in our industry.

I commend you, Commissioner LEHMAN, for the foresightedness to recognize this imminent threat, and to hold these hearings. This Administration has shown tremendous strength of character by raising such fundamental questions about its mission and objectives, and I applaud you for doing so. As we proceed through these hearings let us always keep sight of the U.S. Constitutional mandate for the patent system, to promote the progress of science and useful arts. I cannot find any evidence that patents for software will tend to achieve this purpose. indeed, every indication is to the contrary.

I will attempt to explain Oracle's thesis within the framework of the questions the PTO has propounded for this hearing. First, you ask, "What aspects of software-related invention should or should not be protectable through the patent system?" The examples specified in the question illustrate part of the problem. Software is fundamentally different from what the PTO is used to seeing. In many other industries the policy rationale for patent protection is understandable. In exchange for making their inventions available to the public, patent holders are rewarded with a seventeen-year monopoly, giving them exclusive right to this new technology. In cases where an inventor has committed substantial capital resources to the invention, this opportunity to monopolize the commercial application of the invention is justified not simply as a reward but as an incentive to motivate the developer to dedicate time and money necessary for innovation, design, production, marketing and distribution.

This policy, however, does not fit well with the software industry. Unlike many manufacturing-intensive industries, innovation and development of software products is very

rapid. Although there may be substantial development expenditure, there is an absence of tooling and production is accomplished almost instantaneously. As a result, software improvements are quickly incorporated into new versions, making product cycles very short. Because a patent takes two or more years from application to issuance, well into a product's projected life cycle, patents do not motivate companies to invest in the development, design, production and distribution of their products. In this environment a seventeen-year monopoly is completely out of context with industry reality.

Software varies from manufacturing in another key aspect. The engineering and mechanical inventions for which patent protection was devised are often characterized by large building-block inventions that can revolutionize a given mechanical process. Software seldom includes substantial leaps in technology, but rather consists of adept combinations of several ideas. A complex program may contain numerous established concepts and algorithms as well as a multitude of innovative ideas. Whether a software program is a good one does not generally depend as much on the newness of each specific technique, but instead depends on how well these are incorporated into the unique combination of known algorithms and methods. Patents simply should not protect such a technology.

The scope of what is protectable is a core issue with tremendous impact to anyone in the software industry. Oracle's answer to your question is that none of the cited examples should be protectable with the possible exception of Example F, which is not truly a software innovation, but rather an otherwise-patentable invention that just happens to be implemented on a computer.

Next, although Oracle has not yet been a defendant in a patent infringement suit, it is probably just a matter of time before we are. Our engineers and patent counsel have advised me that it may be virtually impossible to develop a complicated software product today without infringing numerous broad existing patents. Since the validity of many issued software patents is highly questionable and because Oracle is a company with sizeable resources with which to defend a lawsuit, many patent holders must be reticent to litigate an infringement action against us. Further, as a defensive strategy, Oracle has expended substantial money and effort to protect itself by selectively applying for patents which will present the best opportunities for cross-licensing between Oracle and other companies who may allege patent infringement. If such a claimant is also a software developer and marketer, we would hope to be able to use our pending patent applications to cross-license and continue our business unchanged.

But not all infringement plaintiffs are in the software business, and we would be forced to either pay royalties or risk an expensive lawsuit. Thus, to answer your next question, only if patent eligibility standards were dramatically limited could we expect to see a positive implication in the industry. And most positive would be for no software to be patentable at all.

Your next question takes us back to the Constitutional issue. Do software patents promote innovation in the field of software?

The U.S. software industry has evolved to a multibillion dollar industry that leads the world in productivity and accounts for a substantial portion of the U.S. GDP. The software industry has advanced the efficiency of other industries through the proliferation of computing and computer-controlled processes. All of these gains have come prior to the application of the patent process to software, and consequently without patent protection for software. Software companies succeed only because they continue to be innovative in bringing new and better products to the market, and these very market forces will continue to drive the software industry without patenting of software.

Finally, you asked whether a new form of protection for computer programs is needed. We do not believe one is necessary. Existing copyright law and available trade secret protections have proved very well suited to protecting computer software and they have done so in a manner that is not disruptive to software development. Copyright protects software as soon as it is written, without the expenditure of time and money on prior art searches and registration. Since computer software is considered a work of authorship under copyright law, the entire software program including each portion of code as well as the derivatives thereof are protected from copying. Developers may write software code without fear of infringing the rights of others, so long as they do not copy other developer's works. Copyright law encourages innovation since it allows everyone to take advantage of improvements in technology while protecting developers from having their specific works copied or appropriated.

At the same time, trade secret law protects developments that have not been disclosed beyond the development team. Many companies are successful in using trade secrets to establish market prominence, while the competition hurries to catch up.

Oracle has recommended that patent protection be eliminated for computer software and computer software algorithms because software patents are failing to achieve the Constitutional mandate of promoting innovation and indeed are having a chilling effect on innovative activity in our industry and because software is fundamentally different from manufactured products and these differences justify different treatment under the law.

Nevertheless, if patent law continues to apply to software, we believe that fundamental changes must be made in patent policy and procedure. Our recommendations in no way endorse the use of patents for protecting software, but the recommended changes could serve to assuage the existing problems if patents must ultimately affect software development.

However, we believe that making the necessary changes to the patent system will prove to be highly difficult to achieve. Patent law must be consistent throughout the world, and if it is to be applicable to software, it should encompass much shorter periods of protection than exist now, unified prior art searching capabilities, equal standards of novelty, the elimination of patent rules that allow patent flooding, and identical standards for prior-use restriction.

Because the evolution of software moves very quickly, the term of software protection should be cut back accordingly from the current seventeen years from grant date to three

years from the application date, that is, the application period must be dramatically reduced. A balance of fifty years protection for direct copying of code would continue to be provided by copyright law.

Also key to the success of the patent system for the software industry are the following changes. First, the prior art capabilities of PTO workers must be vastly improved to conform effectively the novelty and nonobviousness of the software patent that is the subject of applications. New classifications as well as an effort to record the current state of prior art would be necessary. This is conceptually a daunting task. Most software innovation is not recorded for public availability. Instead it is held as trade secrets.

The Software Patent Institute has been formed to build a database to assist the PTO with finding prior art, and while the SPI's intentions are admirable, it is inconceivable that developers, small and large, will be willing to give up their trade secrets or even to devote the substantial time needed to evaluate, draft and submit evidence of existing art to the SPI database.

Second, because the unusual speed with which software innovations are incorporated in products, the PTO's patent review process must be made more efficient. It should take no more than six months from application to registration. In the software industry where a patent application typically takes two or more years to process, the patented invention is frequently either widely used or obsolete by the time the registration is issued and the public discovers it is protected by a patent.

Third, examiners skilled in computer science and software programming must be trained on the nature of software inventions and the state of existing art. Many more qualified examiners must be employed at the PTO. Compensation rates equal to those provided by the industry are essential to recruit qualified personnel and to retain them at the PTO.

Fourth, the PTO in conjunction with industry must establish additional committees to clearly delineate the standards of novelty and nonobviousness that will be required for software inventions to receive patents.

Thank you for affording me the opportunity to speak today. I again commend the PTO for its willingness to face this very difficult but extremely important issue.

COMMISSIONER LEHMAN: Thank you very much, Mr. Baker, I really appreciate your coming to us. I would love to ask a bunch of questions, but I think since we got a little bit of a late start we move on, so, thank you.

Next I'd like to call Carl Silverman, Chief Counsel at Intel Corporation.

--oOo--

CARL SILVERMAN

INTEL CORPORATION

MR. SILVERMAN: I thank you. Good afternoon. My name is Carl Silverman, Chief Counsel, Intellectual Property for Intel Corporation. We understand that the Patent and Trademark Office is interested in obtaining public input on issues associated with the patenting of software-related inventions, and we're pleased that the Patent and Trademark Office invited us here today to briefly testify.

Software technology has become an integral part of virtually all of U.S. industry, as innovators strive to develop new and improved products in today's competitive, worldwide marketplace. Now, this technology includes pure software, and software which is combined with hardware, so for example, Intel Corporation, like other successful high-technology companies, invests the efforts of its engineers and large sums of money, the shareholders' money, to develop software-related technology. In 1993 alone, Intel Corporation invested nearly one billion dollars in research and development, including a substantial amount in software-related technology. We also, we, Intel Corporation, also invested nearly two billion dollars in capital to build factories so that we can build these advanced products.

These advanced products include products such as our Pentium processor. This is a microprocessor with more than three million transistors on a single chip. This microprocessor product includes software technology in the form of microcode and other computer programs.

Now, to protect and encourage this kind of vast U.S. investment, and I'm referring to both the technical as well as the financial aspects, and, to promote the development of new and improved products, we at Intel believe that software-related technology should continue to be afforded the opportunity to obtain patent protection.

The patent system has consistently provided an incentive to expend the kind of technical and financial efforts previously testified to to develop new technology, including software-related technology. In the United States the Patent and Trademark Office carefully examines every patent application against prior art to insure that only the novel and nonobvious inventions obtain patent protection. Software-related technology is no different.

We support the current statutory law concerning patents as well as its interpretation by the courts as relating to software-related inventions. We are currently aware of no alternative to the patenting of software-related inventions that will better-serve our industry than the current patent laws.

Further, we believe it would be a mistake to treat the patenting of software-related inventions differently than the patenting of other utility inventions. In this regard, Patent and Trademark Office and the courts should be left free to develop the extent of patent protection for software-related inventions and its enforceability on a case-by-case basis until such time as it is apparent that the courts are not up to the task. This time is not at hand, rather, the courts for the most part are both interested and concerned about protecting innovative technologies such as software-related inventions.

Now, this is not to say that the current system for patenting software-related inventions is not without opportunity for improvement. For example: We understand that the Patent and Trademark Office is working to improve its library of software prior art so as to improve its ability to examine patent applications in this area. We support this effort. We urge the Patent and Trademark Office to increase its capability to examine software-related patent applications by taking whatever steps necessary to establish the best prior art software library and to increase and-or redeploy the number of patent examiners who are knowledgeable in this

crucial area.

On behalf of Intel Corporation we thank you for providing us with the opportunity to present our views on this subject, and we are delighted that the Patent and Trademark Office has encouraged this free flow of ideas so that we as a country can do the right thing here.

Thank you.

COMMISSIONER LEHMAN: Thank you very much, Mr. Silverman.

What's your view on the idea of prepublication that has been mentioned several times this morning here, that that would be one way of making certain that we wouldn't overlook some priority that we might already have missed.

MR. SILVERMAN: I apologize, Mr. Commissioner, for not being here this morning; I know generally the subject of publication, and oftentimes patent applications end up being published anyway as they're filed in non-U.S. jurisdictions, and I think that is perhaps a vehicle which would simplify some of the issues that are involved here. So I think we're open on that one.

COMMISSIONER LEHMAN: So you don't find that inherently offensive at Intel.

MR. SILVERMAN: No, I don't.

COMMISSIONER LEHMAN: One other question, too, and that is, you know, I remember the last time that I was in San Jose actually was sixteen years ago, and we were at the other end of downtown here at the old Santa Clara County Courthouse, and we had the first set of hearings. At that time I was Counsel of the House Judiciary Committee, and we had our first set of hearings that led ultimately to the legislation that became the Semiconductor Chip Protection Act of 1984, and Intel was a primary component of that. You were having problems with unauthorized reproduction of your semiconductor chips way back then, and ultimately Congress responded by creating a new form of intellectual property protection, and I'm wondering if you have any thoughts as to how that system is working and is the existing patent, copyright and mass works protection regime adequate, or in your particular area do you feel a need for either a strengthening of the mass works legislation or some alternative to that?

MR. SILVERMAN: The Mask Works Act, I think, was very necessary at the time in which it was created, and I think it's been successful with regard to those who would copy other person's or company's products. I think it's been effective there. I think these days, however, it forms a piece of the overall intellectual property protection available in this country. I think it made a lot of sense to do that then. I don't think it makes any sense to do a similar type of protection mechanism for software-related inventions.

COMMISSIONER LEHMAN: Basically you feel that you have an existing intellectual property regime with these three forms of protections that meets your needs, and it's really a question of tightening up on it, and I know you have strong international concerns, but it's really a question of enforcement and tightening up; there's no really fundamental problem with it.

MR. SILVERMAN: That's correct.

COMMISSIONER LEHMAN: Okay. Thank you very much.

Next I'd like to call Kaye Caldwell, President of the Software Entrepreneurs Forum. Maybe you can tell us a little bit about what your forum is, too. It seems to me you were at the Bree conferences before, weren't you?

--oOo--

KAYE CALDWELL

SOFTWARE ENTREPRENEURS FORUM

MS. CALDWELL: Yes. I work with a lot of different software organizations and I'm speaking for the Software Entrepreneurs Forum today. I'm the President of that organization. I'm also the Legislative Awareness Director. SEF, as we this Software Entrepreneurs Forum, is a ten-year-old nonprofit organization of over one thousand present and future software developers. Nearly all of them are located in the Silicon Valley. We have monthly dinner meetings which attract over two hundred people and we've had as many as six hundred people. We also have eleven special-interest groups that each hold monthly meetings on a variety of technical and business topics. Our members are mostly small companies, and I think our idea of small is probably not the same as your idea of small. By small I mean one to five people in the company, very small.

The U.S. software industry is unique in that it includes a large number of very small companies. We have a thousand members just in the San Francisco Bay Area. Many of these small companies are responsible for the most creative new software developments. In the early days of microcomputer software development, nearly all software was created by individuals working on their own with one or two associates; yet these types of development environments were responsible for early word processors, spreadsheets and accounting software. In earlier generations of hardware, I'm told that it was also the case that one- or two-person companies were a major factor in leading innovations. Even today much software marketed by large software companies is initially developed by small, independent software developers.

SEF's mission is to help these small companies succeed. While SEF is local to Silicon Valley, we feel that SEF members are representative of the thousands of software entrepreneurs throughout the United States. Our members have different opinions on software patentability, indeed you're hearing from at least two of our members at other points during this hearing. Some of our members are patent holders and are strongly in favor of broad patent software protection. Other members are against software patents entirely. However, there are issues on which SEF members are in general agreement. Our members feel that the patent system favors large companies over small companies, and we feel that it's important that the patent system both in theory and in practice should not give big companies an advantage over small ones.

Most of our concerns have to do with Patent Office practices, and we understand that this subject is scheduled for hearing in February. We do appreciate the opportunity to speak on these issues today.

To the degree that software is patentable, SEF members want the patent system to produce good, clear patents, especially where the patents are important. We want patents to be issued and infringement issues resolved

expeditiously.

Uncertainty as to the validity or scope of a patent hurts patentee and possible infringer alike. It makes it difficult to make decisions, to raise capital, to develop business plans and to make business decisions. Patents of uncertain scope or validity are much more damaging to small companies than large ones. Small companies can rarely afford a good legal analysis on a patent's scope and validity, and an uncertain situation can often put a major part of the small company's net worth at risk. The current patent system seems to encourage litigation. We feel that it's important to improve the patent system so that it becomes more self-enforcing.

While we realize that much progress has been made recently in the ability of the Patent Office to deal with software patents, initial progress needs to be made. In proposing the following recommendations for consideration we're less concerned with the specific suggestions than we are in highlighting what we see as problems and in stimulating the Patent Office to solve them.

Our first recommendation would be that the Patent Office should continue to improve its prior art database by adding to it textbooks, scholarly articles, user manuals of commercial products and nonpatent prior art cited in existing and pending software-related patents. This would be particularly effective if such art could be added to the PTO's computerized database, but also be useful to review the trade publications for the last ten years to identify significant software products or product enhancements so that their manuals could be included in the database.

Our second recommendation is that we realize that patent examiners must have both technical ability and a knowledge of how to apply legal principles to determine patentability. We encourage the Patent Office's recruiting of examiners with computer science knowledge. We encourage the Patent Office to continue to improve the quality and expertise of its patent examiners and software. We particularly suggest increasing the pay and professional stature of examiners so that more examiners see it as a professional career rather than just a stepping stone to private practice; I think you've heard those suggestions earlier also.

We also suggest putting a high priority on identifying and expeditiously examining patents which are likely to be asserted. We feel that it's important to identify crucial patents and to focus patent office resources on them as the place that will have the most real-world effect. The accelerated patent examination appears to be a mechanism for achieving this, but in practice the accelerated examination does not seem to be having the desired effect. We believe the criteria for making the accelerated examination also serve to select those patents which are likely to be asserted, those patents being reexamined, reissued, or where the patent holder says there's a suspected infringer.

The performance criteria for the Patent Office should give more weight to the examining of high-priority cases rather than simply counting numbers of patents examined. We suggest that expediters be responsible for getting accelerated patents through the system so they don't get stuck on individual desks. This and other problems could be reduced by tracking patents based on the length of time

they've been in the Patent Office rather than their length of time on a particular desk. In brief we believe it's good public policy to identify those patents that are likely to be asserted and examine them promptly and thoroughly so as to reduce the uncertainty of the scope of those patents. This way the patent can take a place in the free-enterprise system as a negotiable commodity of reasonably-certain scope.

We understand that the Patent Office is trying out a preexamination interview. As we understand the way it works, prior to the examination the examiner, the patentee's lawyer and possibly the patentee have an interview where they attempt to convey what the invention is and to identify where relevant nonpatent prior art might be found. We commend this idea, which we think has potential to both speed the examination process and create a better-quality patent. We also commend the Patent Office for trying out new ideas on an experimental basis to try to improve the patent process.

We understand that the level of skill and the art needed to determine obviousness should be supported by printed publications. We also understand that the determination of obviousness is a legal question. However, we encourage the patent office to try to use software professionals and academics to help locate relevant printed publications which would document the level of skill in the art.

We also encourage the Patent Office to provide further education for patent applicants, which includes actual case examples illustrating how applicants can pursue the question of nonobviousness. Particularly important would be actual examples outlining the examiner's reasoning in determining nonobviousness.

We feel that there's a need for better education of the public on patents in general and software patents in particular. The reexamination process should be highlighted as a normal part of the process. The role of prior art in the reexamination process should be made known to the public and to the press in order to reduce the concerns of possible infringers.

The Commissioner has ordered a reexamination of the Compton's Multimedia patent. We applaud this action as it shows a respect for the legitimate concerns of possible infringers, especially small ones. We propose that the Commissioner, as a standard policy, order reexaminations of patents at no cost on request by small entities which both present evidence that they've been given notice on the patent and produce prior art or other evidence of invalidity.

In the interests of reducing the time and expense it takes to determine the validity and scope of patents, we propose that the law be changed in the following ways. A, have Federal judges remand all validity issues to the Patent Office for reconsideration. The courts should still be able to review such actions. B, require that anyone representing a possible infringer who has prior art on a patent send that prior art to the patent office for submission into the patent's file wrapper. The penalty for not doing so would be that the possible infringer could not use such prior art to challenge the validity of the patent. C, limit the number of reexaminations on any one patent to two except under extremely unusual cases, in order to bring forth prior art at an early point; require rather than allow that the Patent Office consider all previously-unconsidered art in a file

wrapper at any reexamination, provided such prior art is filed at some time prior to three months after the reexamination notice is published.

Finally, we would encourage the Patent Office to take full advantage of public participation in the patent process by making their internal prior art database available on electronic form via the Internet as well as placing notices of reexamination on the Internet. You spoke about this earlier today, and mentioned that this was a goal to be achieved several years down the road. Here in California there was a law passed last year that went into effect January 1st to put all pending legislation on the Internet. Last Friday that system went on-line. It took them three weeks. You might want to speak a little bit to Jim Warren who's testifying tomorrow morning. He was very much involved in getting this legislation passed and in getting this system implemented. So I think he could probably tell you something about that process.

We expect that the effects of these suggestions would be to force out prior art early on so as to more quickly determine the scope and validity of the patents.

Commissioner LEHMAN, I'd like to thank you for holding these hearings here in Silicon Valley and for giving us the opportunity to speak. Thank you.

COMMISSIONER LEHMAN: Thank you very much. I want to commend you for a very interesting catalogue of suggestions, and I think it's very gratifying how a group of individual inventors like yours, not a big corporation, can really give so much thought to something like this and come up with so many very intriguing recommendations that we're going to be looking at, and on that question of the -- putting our system on the Internet, I would just make an observation that the quantity of data in our files is a little bit larger than the legislation currently pending before the California Legislature, and there are a few more technical problems, but one of the things maybe if I can put in an advertisement, we have openings for two positions now in the Patent and Trademark Office, basically the two top people who ran our information systems program for the last seven years have retired. So we're recruiting for new people to take this over.

Obviously I think one of the problems we have now, there was a suggestion earlier that maybe we ought to move the Patent Office out here, and maybe we should move at least Group 2300 out here. I don't know. I see Gerry Goldberg is saying that wouldn't be such a bad idea, after all we went through in Washington last week. But this certainly is where the talent is, so I think we'll be publishing these openings pretty soon, and I think we'll do some aggressive recruiting out in this part of the world; and maybe you can help us to get ourselves up to snuff technologically. Maybe it won't be three weeks, but maybe it won't have to be the years that it's taken us thus far to make these improvements.

Anyway, I wanted to thank you very much for your excellent suggestions that are really appreciated. Thanks.

Next I'd like to ask Mr. James Chiddix to come forward, who is Senior Vice-President for Engineering and Technology of Time-Warner Cable.

--oOo--

JAMES CHIDDIX

TIME WARNER CABLE

MR. CHIDDIX: Good afternoon. My name is Jim Chiddix. I'm Senior Vice-President for Engineering and Technology at Time Warner Cable. Time Warner Cable is the second-largest cable operator in the United States.

COMMISSIONER LEHMAN: Can I ask, are you headquartered in New York or here?

MR. CHIDDIX: Actually in Stamford, Connecticut.

COMMISSIONER LEHMAN: Oh, really.

MR. CHIDDIX: We serve more than seven million subscribers in thirty-six states. Our parent company, Time-Warner, is the largest owner and distributor of copyrighted material in the world, and intellectual property rights are something for which we have great respect.

Two years ago we built the first one hundred and fifty-channel cable system in Queens, New York, and that remains the most advanced cable system in the world today. Currently we're building the country's first electronic superhighway, which we call the full-service network, in Orlando, Florida, and there we'll offer a host of high-speed two-way interactive services including video on demand, interactive shopping, and distance learning. Time Warner Cable plans to spend more than five billion dollars over the next five years to deploy full-service networks in the majority of our service areas across the country.

The Administration and members of Congress have indicated that building such networks is a national priority. In our experience, however, the current patent system is working against the development of an advanced communications infrastructure. Ever since we announced our full-service network plans, we and our suppliers have received a number of inquiries from individuals and companies who purport to have patent rights that cover basic but to us obvious elements of the information superhighway as well as traditional cable systems.

I'd like to describe for you two of these patents. The point is not whether these patents are valid or invalid, or whether any particular use is infringing or not infringing, although we firmly believe that nothing we are doing infringes on any valid patent. They do serve to illustrate the current patent system is out of balance and that rather than promoting the progress of science and useful arts, that system is stifling such progress.

My first example involves the Cutler patent. The Cutler patent purports to cover many uses of optical fiber to transmit television signals to receivers in the home. This patent was granted in 1979 and will expire in 1996. The use of fiber, of course, is basic to the electronic superhighway and has also been used for many years in traditional cable systems. Indeed, Time Warner Cable has been a pioneer in the deployment of broad-band optical fiber in cable systems. The inventor of the Cutler patent did not invent optical fiber.

Rather he merely filed a patent for using fiber to transport video signals to the home. The patent statute says that patents are not to be granted if the subject matter as a whole would have been obvious at the time the invention

was made to a person having ordinary skill in the art to which subject matter pertains. I'm not a lawyer, but it seems to me that the idea of using optical fiber to transport video signals to television sets is not only obvious, but also inherent in the fiber optic medium itself, which was conceived as an information conduit. If such a patent were valid, I would think it would also have been possible to obtain patents after the invention of television for using the medium to transmit drama, sports or news programming. These, however, are merely self-evident uses that are inherent in the medium of television, just like video transport is inherent in the medium of optical fiber.

My second example involves the Starside patents which purport to cover a wide variety of features used in connection with electronic program guides. Electronic program guides are on-screen guides that provide program listings for channels that are broadcast or provided by a cable system. Starside is a number of patents, but the features I discuss here are purportedly covered by a patent granted in 1987 and another that is currently pending before the Patent Office.

Pursuant to this patented application, Starside apparently claims and seeks protection for the following electronic program guide features. First, the ability to move a cursor of automatically-varying size about on an onscreen program guide, to highlight a particular program on the schedule and then press a button on a remote control to tune the channel on which that program is being transmitted. Second, the ability to combine two or more criteria, such as sports and football, to obtain a listing of the times and channels on which programmings filling those criteria will be telecast.

Again, to me these features seem obvious and inherent in the technology that provides them. Daily newspapers have long provided channel listings, often using a grid format that shows what programs are on what channel at what time. In addition, individual broadcast channels and cable systems have long-telecast on-screen programming schedules. When a television viewer uses such a schedule he finds a program of interest, identifies the channel, and punches the number into the remote. The Starside system merely does this tuning process automatically through a straightforward transfer of the process to a computer. Similarly, when I want to watch football games on television, I simply scan the program schedule for such programs. It would be a simple but somewhat time-consuming task to write out a list of such programs, but again, preparing lists from data based on multiple criteria is a simple, straightforward and obvious computer application.

Under existing law, patents for what I've just described may be valid or invalid. As I said at the beginning of my remarks, however, in either case, such patents present impediments to progress. If such patents are found to be valid, surely the patent system has gone too far in providing protection for what would seem obvious to a layperson, let alone to a person have ordinary skill in the art. The result of awarding such patents at best results in added costs for no added value, if a license is obtained, and at worst prevents consumers from fully realizing the benefits of technology if a license cannot be obtained at a reasonable price.

If such patents would ultimately be found to be invalid, however, the patent system would still not be working

properly. Some of the Starside patents are currently being challenged in court. Business, however, cannot come to a halt in the meantime. Also, litigation is costly, slow, and never free from risk.

Rather than expend time and money on litigation, many prudent business people will choose to avoid the problem. Indeed, one of our suppliers of set-top boxes has informed us that rather than challenge the Starside patents, they will instead defeature the boxes they are making for one of our cable systems, removing ability of those boxes to provide some of the features that Starside claims are covered by its patents. This is not an uncommon or irrational decision. This supplier will be spending many millions of dollars to manufacture these new boxes. Even though they believe that Starside patents are not valid, it is simply not worth the risk and the cost of fighting them in court. Of course our supplier can always attempt to obtain a license for these features, but again this would result in added cost for what in our view provides no real added value.

So in our view, the present system of patent protections is not optimally promoting innovation in the field of software-related inventions. Rather, the current system is in some important instances stifling innovation, increasing costs and leading to defeaturing rather than fostering the development of new and better products and services. However, it is not the framework of the system that is the problem. The statutory tests of obviousness, and the person of ordinary skills standard, in themselves strike the proper balance.

What is needed then is not a new framework for patent protection for software-related inventions, but a more rigorous application of the present standards. For one thing, obviousness should include routine applications of a given technology regardless of whether there is prior art showing that particular application. For another, any invention that merely transfers a series of routine tasks to a computer should also be viewed as obvious.

As your Notice for these hearings states, the computer software industry has evolved into a critical component of the U.S. economy. Indeed, the importance of this component is growing greatly every day as the computer, cable and telephone industries continue to converge. If the United States is going to continue to be a the forefront of these crucial industries, it is imperative that the patent system be restored to its proper balance so that it can properly foster rather than frustrate innovation.

Thank you.

COMMISSIONER LEHMAN: Thank you very much, Mr. Chiddix. Time-Warner's certainly a company, unlike some of the other people who have testified, who is well-able to use every legal technique at its disposal to protect its rights, and does so if it has difficulties. I'm, I -- it's interesting to me that you haven't -- you have never apparently used the reexamination system to attack some of these patents that you disagree with. Is that because you did not feel the problem was in the prior art that was examined, that it was more the legal standard that was applied by the patent examiner, or is there some other reason why you failed to use the existing examination system?

MR. CHIDDIX: These are both very current cases, and I'm

not sure that all avenues have been explored. The obviousness argument is one though that even reexamination may not be fully armed to deal with.

COMMISSIONER LEHMAN: I take it you don't share the view of some of the witnesses that we should completely eliminate software patents, rather, we should tighten up on the legal standard of patentability for software patents.

MR. CHIDDIX: Yes. That's correct.

COMMISSIONER LEHMAN: Thank you very much. Are there any other questions? If not, thank you very much.

Next I'd like to ask Wallace Judd, the President of Mentrix Corporation to come forward if he would please.

Is he not here? In that case I think we'll move on to Robert May, Ikonic Corporation. Is he here?

Okay, well, then we're -- this is why we ask people to be here at least twenty minutes ahead of time, the scheduled time, because we can see what happens.

The next person on my list is Pete Antoniak of Solar Systems Software. He's not here? Mr. Antoniak is not here? Is Professor Hollaar here? Good. Since you had to come all the way from Utah, you're --

MR. HOLLAAR: Early.

COMMISSIONER LEHMAN: You're early. So you'd help us out and maybe some of these other people will arrive. Thank you very much, Professor, for joining us.

--oOo--

LEE HOLLAAR

UNIVERSITY OF UTAH

MR. HOLLAAR: My name is Lee Hollaar, I'm a Professor of Computer Science at the University of Utah where I teach the Senior Software Development Laboratory and also teach computer intellectual property law within the Department of Computer Science. I also conduct research into information retrieval systems. I've been involved with computers for almost three decades and received my Ph.D. in Computer Science in 1975 from the University at Illinois. I'm also a registered patent agent working with the Salt Lake law firm of Van Cott Bagley Cornwall and McCarthy primarily with computer-related inventions. I hold one United States patent and I have another patent pending. The views I'll be expressing are my own and not necessarily those of the University of Utah or any other organization.

First I'd like to thank the Commissioner's staff for the opportunity to testify regarding these important matters and to commend them on holding these hearings. I'd also like to congratulate the office for making their decision to accept comments electronically and to make the comments and transcripts of these hearings available on the Internet.

Today what I'd like to do is cover a few points from my dual perspective as a computer scientist and also a patent practitioner. I'll be following up this testimony with written comments.

One of the things I would like to mention is that there are a number of assumptions that seem to be taken for granted about the differences of software and its patentability that may not actually be true, and should be examined. One is that there's the assumption that computer software is a fast-moving technology and that therefore a lesser patent

term -- I've heard three years suggested -- may be appropriate. Interestingly enough, about two weeks ago Butler Lamson gave an address at the University of Utah, he's one of the inventors of the Alto Computer and a number of other innovations from Xerox Park, and he indicated that from at least his point of view much of the innovation going on in computers is not the result of computer software, but the computer hardware now available at much faster speeds with more memories, enabling techniques that were known in the laboratories many years ago to be possible now and to be available to the masses. In fact he made the statement which many people disagreed with, but it certainly caught people's attention, that from his point of view with the exception of spreadsheets there's been no surprises in computer science since 1975. It's interesting to note that anything which you filed an application on in 1975 and a patent would have issued, the patent would have been expiring about this time. It's also interesting to note that the fundamental books on computer algorithms, written by Donald Knuth came out when I was a graduate student approximately two decades ago. Again, if everything in those books was patented, the patents would be expired by this time.

Much of computer science I see, my students and so forth, consists of reinventing wheels. A large amount of that is because people don't check prior art and a large amount of that is because there's no good prior art collections to check, and I think -- and I will comment on this -- that this is one of the problems that has been caused by the two decades of the Patent Office having at best ambivalent attitudes toward the patentability of computer software and not using the patent system to draw the trade secrets and the other art into the printed publications of U.S. patents. I was involved with developing a computer system in 1969 which is still running. If I had patented every technique in that computer system, and it still represents the basis for a state-of-the-art system, those patents would have expired five or six years ago on it.

Comment has been made that patents can restrict developers, and that's certainly the case, but that's true on every patent in every area of art. It's not particularly true for computer science in any respect. The comments have been made about bad patents, patents which have prior art problems, poor examination, again these exist in every art unit. Perhaps it may be worse in the computer art area because there was this period of time when the Patent Office was denying computer patents and therefore not building their prior art collection, not encouraging the submission of patent applications which would mature to patents and go into the prior art collection on it.

In my recent practice I've found the inexperience that people have talked about with the examiners to be diminishing and I've found that the examiner's doing quite a good job given the constraints of trying to examine very complex art in a wide variety of areas. I think also that the patent office has received bad press due to both people who should know better and also the people providing the information not knowing enough about the patent system. Too often comments are based on the title or at best the abstract of the patent, and not the claims, which indicate what the true invention is. Often this is compounded by press releases

from patentees trying to make their patents seem more important than it really may be.

There have been a number of problems caused by the past PTO position on patents. One, as I mentioned, is the poor prior art collection, which has been difficult, because in many cases computer inventions aren't as self-revealing as mechanical invention. When you sit down and sue a spreadsheet or another computer, you're aware that you're using a spreadsheet, but you may not be aware of the order that cells are being recalculated or the particular algorithm that is being used for justifying the text. The patent system would have eliminated this problem by forcing inventors, in trade for the patent monopoly, to reveal their trade secrets, to reveal how this would work, thereby giving a prior art collection for people who want to develop follow-on systems, and because of the distinctive claiming of patents as opposed to copyright, providing an indication of what an inventor can do to avoid the patent and yet produce an improved product on it.

Computer scientists suffer from a poor tradition of publishing their algorithms, especially those involved in industry, and also in looking at prior art. I often think that many of my students wouldn't know where the library is on campus if you ask them, and when they start out on a project the idea of going to the library is not the first thing that crosses their mind, and certainly the idea of doing a patent search to see what's in the prior art and see how to accomplish things is far from their mind.

I think that while the Software Patent Institute should be congratulated for what they're doing, their idea of collecting prior art is doomed to failure because especially for old inventions, it is very hard to determine what is prior art or what are the novel things. In the typeset system I wrote many years ago there were probably hundreds of things which may be novel or may at least be prior art for future inventions, and it would be hard to enumerate them.

Other problems caused by the past PTO position is both in the disclosure and in particular in the claims' obtuse language, unclear claims, as attempts to avoid a perceived perception of the PTO position on Section 101, claiming things which are clearly software programs as computer building blocks. I think of one patent issued to Thompson where not only is the source code listed, but they tell you how to build it out of computer modules from Digital Equipment, out of flipflops for this searching technique. No one believes that that's how they intended to implement the invention. More importantly, the Section 101 babble often takes the steam out of the examiners. After fighting the 101 fight there may be very little fight left in them for the proper 102 and 103 questions. I recently reviewed the file wrappers of two patents for a client who wanted to know their scope, and was shocked by how little prosecution history there was after the 101 arguments had been resolved, even though one patent had a continuation application filed and had been pending for a number of years on it, virtually no 102, 103 arguments after the initial 101.

I think the patent system has produced distortions in copyright laws, courts have held it necessary to provide protection beyond literal copying so that we have decisions like Whelan or the current Lotus against Borland decision, trying to extend copyright perhaps too far from its intended

purpose, and as I said we have the loss of past disclosures because the Patent Office wasn't accepting patent applications, which has caused much reinventing of past techniques.

The solutions? I think one is to eliminate much of the current 101 confusion, and in fact go back to the basic principles of inventorship, that patents are, if there is something useful being produced, should probably meet the 101 test, and the real battle of whether the patents should be issued should be tried on 102 and 103 issues.

The comments have been made about laying open files during prosecution, and I strongly support that. I think it's the one hope we have for getting the prior art. Both laying it open at some fixed period of time such as eighteen months has been suggested to eliminate the so-called submarine patents, and also laying it open sometime after initial examination by the Patent Office. I would make sure that the file isn't laid open before the first office action such that the applicant has a chance to withdraw the application based on the position of the Patent Office and the prior art that's been found, and make an intelligent decision of whether to keep it as trade secret.

I would recommend that there be a period after the office determines that a patent is allowable that people can submit prior art for consideration by the examiner. This could be done by using technology to widely distribute the notice that this patent is about to issue and a representative claim and has some method for retrieving of the application, perhaps tied in with the project ongoing for the electronic filing of applications. This shouldn't be an advisory procedure, but a way of having the prior art brought to the attention of the examiner. I think in my case of the old system that I have, I certainly could see if there was an application and claims whether I had prior art on that and submit that to the attention of the Office much easier than I could go through a very complex system and identify every piece of prior art such that it could be searchable by the Patent Office.

That concludes my remark, thank you.

COMMISSIONER LEHMAN: Thank you very much, Professor, that was some really very helpful testimony. Appreciate you coming over here to California to talk with us. Thank you.

Next I'd like to ask Dennis Fernandez. Is he here? For Fenwick and West? Okay.

Welcome.

--oOo--

DENNIS FERNANDEZ

FENWICK AND WEST

MR. FERNANDEZ: Good afternoon, ladies and gentlemen, my name is Dennis Fernandez. I am a registered patent attorney at the Silicon Valley-based technology law firm of Fenwick and West. I am speaking today on my behalf.

Before practicing law I was an electrical engineer and a technical manager for several years at various companies in the computer and electronics industry including NCR, AT&T, Digital Equipment Corporation, Raytheon, RayCal Limited, where I did semiconductor chip design and processing as well as managed marketing and distribution of semiconductor chip products. Currently I specialize in patent

prosecution and litigation in the semiconductor chip industry.

As you know, in this highly competitive electronics industry, intellectual property protection has become a very important part of doing business. This afternoon, I wish respectfully to share with you a few brief comments based on my practical experience in the semiconductor industry. As an electronics engineer and also as a legal counsel I believe I can provide many comments on how software patents have already or might in the future impact the business interests of semiconductor and software companies in the Valley, and because of the limited time which has been allotted for speaking I direct my comments only to a few points which are relevant to chip design software.

I believe these comments may reflect the interest of companies who do business in the area known as electronic design automation, or, EDA. EDA products refer generally to sophisticated software written for automating the highly complex process for designing and testing semiconductor chips and related system boards.

In this EDA context, I offer comments on the following three areas, number one, special technical need for software patent protection; number two, practical timing problems in U.S. patents, and number three, apparent effect of software patents on innovation.

First point, which is special technical need for software patent protection. Because of the highly functional nature of technical innovations that are developed for EDA software products, patenting seems to be an appropriate way for legal protection. For example, EDA software typically includes software programs for synthesizing logic circuits, generating test program vectors or simulating digital and analog system components. These software functions involve fairly abstract ideas, which would not be protectable ordinarily under copyright law, but would be protectable under patent method or apparatus claims if sufficiently inventive. Thus, due to the largely functional nature of innovations in EDA software, I believe that patent protection is appropriate.

Second point. There are practical timing problems in U.S. patents. There appear to be two practical problems which may apply to EDA companies with respect to timing related to U.S. patent applications. First, the seventeen-year patent duration may be too long. In the context of the EDA industry, where software products typically have product lives that are less than half this duration, it might be more appropriate to provide a shorter period of exclusivity. Second, the current two- to three-year backlog in the United States Patent and Trademark Office, especially in the electronic and software arts, may pose some problems to companies in the EDA industry, both for those companies who wish to enforce their patents during the market window available for their software products, and also for those companies who wish to learn about the existence of relevant patents and thereby avoid them.

Third and last point, the apparent effect of software patents on innovation. In the highly competitive EDA business, particularly in the Silicon Valley, it has been my experience in a number of recent cases that the presence of relevant software patents do not necessarily serve to impede or deter competitive product development. Typically, clients in this business tend to be fairly sophisticated in our understanding of patent enforcement matters. Furthermore, these individuals are aggressive entrepreneurs who are doing pioneering technical work in product development and are typically backed financially by venture capitalist institutions. These individuals often find ways to design around even what appear initially to be fairly broad patent claims. Also, it has been my experience that such EDA clients are often able to obtain reasonable licensing terms or raise reasonable arguments for invalidity based on relevant prior art, thereby providing themselves with opportunities to make, sell or use their EDA products, possibly without legal liability. It does not seem to me, therefore, that software patents have necessarily stifled competition, at least in the electronic design automation industry at this time.

This concludes my prepared comments. I thank you very much for your consideration.

COMMISSIONER LEHMAN: Thank you very much, Mr. Fernandez. So basically the bottom line is that you really don't think we need fundamental changes in the system, but you think we might want to deal with certain issues regarding the EDA industry like the appropriateness of length of terms.

MR. FERNANDEZ: Term limits. Yes.

COMMISSIONER LEHMAN: Yes. Thank you very much.

Next I believe that Pete Antoniak has arrived now from Solar Systems Software, so we can put him back on the agenda. Come forward, please.

--o0o--

PETE ANTONIAK

SOLAR SYSTEMS SOFTWARE

MR. ANTONIAK: My name is Pete Antoniak and I'm a professional engineer. I'm looking at the agenda today and see a lot of CEOs, chairmans of boards, and lawyers, patent agents, et cetera. I see very few engineers and software developers.

COMMISSIONER LEHMAN: You can probably help us a little bit if you just make sure that that mike you're talking into --

MR. ANTONIAK: Can you hear me now? Everybody? Good. You see very few engineers, software developers. I develop educational game software. I also teach, consult and program for others in order to supply this software habit of mine. For the past ten years I've been making about a third as much of money as I was making as an executive for GTE Sprint, and I do this because I'm looking for the big payoff, developing the great American software.

Approximately six years ago I started development of an educational game concept that I thought was quite unique and I was totally aware that nobody else had done anything like I was doing. I attended a seminar up in San Francisco, I believe by Prentice-Hall, in which I believe somebody from this panel or somebody from the Patent Office gave a talk and encouraged people like me to go ahead and get patents on our software. I was sent some brochures and literature and it seemed like a fairly friendly environment, I said, By golly, I'll do it. I'm the type of guy that repairs my own car and so I went to it.

When I developed my program, and I can understand from people out there that, you know, I'm a little naive in this, I

wanted a program that could make an educational game out of any type of material, any subject, any grade level and any language. Particularly I wanted a game that did not require the need of a keyboard, no typing required. And I came up with a very interesting concept. My game used objects on the computer screen that represent abstract ideas and concepts. The player moves the objects around on the screen to represent their relationships. The concept I developed is simple, compelling and fun, and not only was there no prior art, but even to this day, and this is six years after development there still is nobody even doing anything like I'm doing. I have essentially no infringers.

I purchased and read the book called Patent It Yourself by David Pressman, which is kind of a bible in this industry of people who are inventors like me. I also learned that David was a fellow-member of Mensa, lived in San Francisco and for a fee of seventy-five dollars a review, now one hundred, would review my application. I spent six months developing an application, writing and rewriting it. He had about three or four times to review it and I sent it in as a patent application.

My first office action came about six months later when all my claims were rejected on the basis of prior art. Now anybody in this business understands that this is very common. I was upset at the time, extremely upset, but have since come to understand that this is normal. I can't complain about this as I was told by somebody in the Patent Office, this is the way the game was played.

However, what I can complain about is the fact that none of the prior art had anything to do with computers. I was confronted with such things as jigsaw puzzles, board games, card games, and classroom wall charts. One of the prior art patents was a few months short of one hundred years old. I spent a great deal of time writing responses, and it was very frustrating. It was unbelievable to me that anyone could even connect my program with the prior art that was being used. I felt that there was some kind of a logic gap between me and the examiner. In the end the examiner maintained that my arguments were not persuasive enough and I got a final rejection.

I phoned the examiner, and during the interview it came out that she was a mechanical engineer, did not have a computer and didn't know much about computers in general. She told me that her expertise was in games. I presented a logical set of arguments to get her to admit over the phone that the prior art she was using was absolutely not applicable to my claims.

She then stated that she was sure that there was something out there, perhaps in child development books or something that duplicated my computer game on a table, with three-by-five cards, a pencil and perhaps a teacher to look over as a referee or a judge. If she had the time she would find it. I said that a table was different from a computer in that a table could not know where those three-by-five cards were, whereas the computer screen did and could do instant evaluation well beyond the capabilities of a teacher. She said that I should explain that if I was to reapply. Well, essentially what I was doing defending an invention that didn't exist but only in her mind and which she didn't really tell me about.

She finally requested a different examiner. David Pressman also advised me to request a different examiner. He even made a few derogatory comments about my examiner being young and experienced. This was the luck of the draw and I just had to live with it.

I resubmitted, respectfully requesting a different examiner, waited six months, and the first Office action came back, and you guessed it, I had the same examiner. I went through another frustrating round going back and forth. I felt sometimes like I was arguing with a brick wall and there was some sort of a hidden agenda that no matter what I said that I wasn't going to get a patent.

After the final rejection the second time, I traveled to Crystal City, had an interview with the examiner and her supervisor. I arranged to use a computer store across the street from the Patent Office and demonstrated my software. I quickly became aware, to my dismay, from the examiner's reaction to the program, that she really didn't even understand what it was that I had submitted, and more importantly how it worked. This was moot, however, because just prior to the demonstration her supervisor had promised that if I were to resubmit again that I would get a different examiner.

I resubmitted a third time, and again was rejected in the initial office action. However, the tone of the rejection is different. It was obvious that the new examiner had read the application and understood it. He pointed out discrepancies in grammar between the original application and the claims. He pointed out some tactical errors in the claims and he recommended ways to correct things. The prior art he introduced seemed more pertinent to the claim and the invention. I almost cried, not that I had been rejected a third time, but because at least I had someone whose attitude was not, "How do we get this guy out of here," but, "How do we get this application in proper order to be patentable."

In the next two weeks I will submit a file wrapper continuation and start Round 4. The last thing my new examiner told me, however, when I called him up, was that he is now under pressure because of the Compton's fiasco, whatever you want to call it, and that he'd probably be throwing a lot more prior art at me. I'd rather take a stoic attitude about my experience with my Patent Office. I'd like to say that an easy patent that has not been exposed to a lot of prior art is not a patent that would stand up to a challenge; this has often been said, that if you get a patent too easy you may have a problem with it later on. However, I believe that the type of prior art used in my case was so nonapplicable as we go to the background, the experience of the examiner, that for all intents and purposes I am starting over, having wasted four years, many thousands of dollars in fees, approximately eight man months of my time that I could have been using in development, in marketing a product.

My claims are not earth-shaking. They're not controversial. They only protect my program. I will note that even though, and I'd mentioned before, I haven't shown it to a lot of people, I'm always aware of what's going out there, I've yet to find anybody that would be considered an infringer of my product. I'm a little guy; I'm the inventor. I'm the software-equivalent of the inventor, and my bottom line in all this is that if the Patent Office is doing a good job -- I'm not recommending a lot of changes to the Patent Office --

it's just if they're doing a good job, you know, I would -- I would probably have a patent right now, or at least I would know I couldn't get a patent right now. Right now I've spent a lot of time on absolutely, you know, nonsensible things. And I kind of consider, like when I took Latin in high school, you know, it's one of those things to give you some discipline. I can certainly write a claim. Jeez, you know, I can do very well. Matter of fact, as a member of the Software Entrepreneurs Forum I even had a conference about a year and a half ago in which we had the whole afternoon devoted to doing patents. I brought in a patent agent and we discussed these things. I'm pretty much up on it. However, I'd rather have a patent, rather have the, you know, the rights thereto, that I may attain to.

Anyway, that's it, and that's my bottom line, and if I have any time left, I can entertain any questions.

COMMISSIONER LEHMAN: Thank you very much, Mr. Antoniak. I would just say that the patent system is not set up to be a pro se system, and we have a lot of problems in that and we're here to help solve those. But part of being a good patent lawyer is to understand the art of lawyering, and oftentimes that involves making certain that your claims are drawn in such a way that you get to the right patent-examining group -- doesn't sound like your patent originally went to Group 2300 -- and that it get to the right examiner, and I'm very sympathetic with your difficulties that you had, but I think that when one attempts to bootstrap their case and you try to do brain surgery self-taught that you're inevitably going to run into some difficulties, that are hard for us to try to address in the system.

MR. ANTONIAK: Let me add that all along the way I was using a patent agent to review everything I set in. First it was Pressman and then it was a registered patent agent here in the Valley who has done software for the last twenty years. I looked to the Patent Office like a pro se inventor. In reality, though, everything was checked and rechecked and gone over by a patent agent who gave sound advice. All along the line they were saying, "Yes. This is definitely patentable. Matter of fact, what's somewhat ironic is that when I submitted my thing to Dave Pressman the first time around, he said somebody else had also submitted something that he said was absolutely not patentable, and what he was surprised was he sent it in and got a patent right off the bat with very little problems. I've run into that individual because it turned out I know him anyway, and Dave was surprised that mine had such a hard time getting a patent. Again, the wording was right, everything was pretty much right, my logic was right.

I think that the term pro se inventor, you know, have said, "Okay, here's a category, we don't have this guy the, you know, the professionalism that you'd give perhaps a patent attorney who can call us on it." Now that's my, that's my personal judgment on this, and I could well be wrong, but this is an experience; a lot of emotions and a lot of energy's gone into this, and -- you know, what can I say? I've been -- I've been to Vietnam and I've learned a lot from that, too. So.

COMMISSIONER LEHMAN: Well, I'm sorry you had that difficulty and I'm hopeful as a result of these hearings we'll get the problem fixed. Thank you very much.

MR. ANTONIAK: Thank you.

COMMISSIONER LEHMAN: Next I'd like to call Mr. Robert May from Ikonix Interactive, I believe.

--o0o--

ROBERT MAY

IKONIC INTERACTIVE

MR. MAY: I want to thank you very much for coming out to Silicon Valley and also for slipping me in unannounced. I'd expected to be out of town today and at the last moment my travel plans changed.

I want to give a quick perspective on the lay of the land from Ikonic Interactive. We're a software developer located in San Francisco. We're a multimedia developer with about nine years of experience in this business. Current clients include Time Warner for whom we're designing the user interface and software for the full-service network in Orlando, Florida; Dow Jones for whom we just recently completed the redesign of the Wall Street Journal for PDAs; a variety of other projects. So we're intimately acquainted with some of these issues and I'd like to just give you a snapshot of some of our perspective.

I spent my morning on the phone with one of our clients negotiating contracts, and I should say, number one, I am not an attorney, and it's only through the tutelage of Kate Spellman up at Steinhart and Falconer, our IP attorney, and David Hayes down here at Fenwick and West that I know just enough to be dangerous, but notwithstanding that, I often rush in where angels fear to tread, and I wanted to discuss two key issues that we face every day, and just to give you some data with which to make some decisions.

I should also say in the spirit of full disclosure, we do have a software patent application under way, another one that we're considering, and I come here as a supporter of the notion of software patents, and more specifically, interface patents. Notwithstanding that, a single biggest problem I'd say from a business exposure standpoint is that we are often asked by our clients to indemnify them, that we have not incorporated prior art or other patents in our work. Given the way that the prior art search has to be conducted at this point it's very very difficult for us to indemnify our clients to that, and I would respectfully suggest two possible solutions to that.

And the first would be that in my midnight reading of patents and patent law, which I've been doing the last year or so, I've learned that things like the Compton's patent have many many many claims attached to them, and it's very very difficult to understand, let alone plan for the implications of those claims. Originally as I understand it, patent law was designed to address inventions that were reduced to practice, and it seems to me that it might be a useful distinction to separate claims very specifically to those that are actually reduced to practice and are shown to be reduced to practice and those that are speculative and looking for future technologies. And as a nonattorney I've got a very difficult time when I'm faced with trying to judge that and then promise in a contract that I will in fact not -- not infringe on those claims.

Number two, recently the FCC decided to free up 10(k)s and other public filing information and make that available on the Internet; previously it was available on Meade and I believe maybe Lexis and Nexis. Currently I'm not aware of a

way to research patent information by public, without signing onto Lexis, undergoing quite an expensive search process, and insofar as it is public information I'd like to urge you to make that information available on the Internet and make it available publicly.

Second issue that we're confronted with very frequently, and this wraps around both patent issues and copyrights and others --

COMMISSIONER LEHMAN: Can I ask you a question. What kind of information for somebody like you would be useful? You know, there's everything from the full text of prior-issue patents and all of the company technical drawings to, you know, abstracts of the patent. What kind of -- when you're talking about making things available on the Internet, what kind of information of that type would be useful to you?

MR. MAY: Yes, sir. Given the example that I just gave, from a business person's perspective, I'd like to see the whole thing, because I'm being asked to indemnify my client against all claims. A helpful start would be the abstract, but lamentably, I've got to be familiar with the art to the extent that I can be. It would help me very much if there were to be drawings, et al. Does that answer your question?

COMMISSIONER LEHMAN: Yes. Yes, it does.

MR. MAY: Okay. The second area that we experience in day-to-day is again as I said, broadly a problem across both patent law, copyright law, and I'd like to just raise it in the context of patent law here today, and that is the difficulty between what current practice is and what we'd like to see practice moved to, and that is the concept of work made for hire wherein typically a small company like ours is doing work for a larger client, who attempts to get us to engage in that work under work made for hire, which means that, as you know, they own all patent rights and copyrights and trade drafts, et cetera. Very difficult to conduct business in this way and to grow a business in this way.

So we've been successful ourselves and I urge other folks out there in our business to move to a license strategy where in fact we retain the rights to underlying key concepts and intellectual property that we develop and license that in perpetuity on a royalty-free basis to our client.

One of the key problems with that approach is the ambiguity in copyright law between what's called look and what's called feel, and I would urge you and the folks you work with to turn your attention to that ambiguity and try to address that, and the specific case in point is that when we're faced with producing a project for say Time News On Demand for Time Warner, it's one thing to grant them the rights to look at that program and morally and ethically and by all other business means, I'm absolutely committed not to producing work for another client that copies and looks the same as the work that I do for my initial client. At the same time, in the pursuit of my business, we often enjoy the discovery of elements that help us do the job better for the next person, and the current ambiguity in copyright law makes it very difficult to parse out, to separate out what is look from what is feel, and feel, as you know, in the Apple Microsoft Case has been pulled out to mean basically menu command structures, things like that.

So that's the underlying structure that we need to have in

what we call our multimedia toolbox in order to ply our trade, and to the extent that we are forced by circumstance or by the size of our business in the marketplace to give up those rights in a work-made-for-hire scenario, it's very difficult for us to ply our trade in the future. Like the carpenter being told that they can't use a particular jig that makes them drill holes faster, it cuts down on our efficiency and our ability to carry on our business.

So I wanted to bring these two points to mind just to give you a snapshot of what it's like out here on the frontier of the Information Superhighway, but these are key issues for us and I'd welcome your attention to those.

COMMISSIONER LEHMAN: Part of that problem that you have, and I don't mean in any sense to suggest there isn't merit to your substantive ideas about scope of copyright protection, but is part of the problem there that in your situation when you mentioned Time Warner, they're a client, you're a small company, a small entrepreneur, that you just don't have the marketing power basically to avoid being, you know, strong-armed into signing work-made-for-hire agreements that would mean that you have to give away more than you'd like to give away. I mean is that one of the reasons this becomes really acute? You just -- Ideally, you know, you don't have to -- you work under conditions that you want to work under and you can say, Well, I'm sorry, I'm not going to give away some of my techniques that I would otherwise give away, but you just can't do that because you don't have the market clout. Is that a problem?

MR. MAY: Well, I think it's tempting to paint the big company as the bad guy and the little company as the Don Quixote. I would suggest from personal experience that large companies in point of fact, once this distinction, this difficulty is raised, they're willing to look at solutions that work for both sides. There is a reflexive tendency to turn to work made for hire as something that quote has been done in the past, and it's always been good enough. Happily we've been able to negotiate positions with our clients that enables us to move forward, but yes, I think in some instances you're correct, if you're a smaller company, if you haven't been able to build your multimedia tool kit and you come in with just a hammer and saw sometimes it's easy to be bulldozed, to mix my metaphors.

COMMISSIONER LEHMAN: Thank you very much.

MR. MAY: Thank you. Appreciate it.

COMMISSIONER LEHMAN: Next I'd like to, back on our regular schedule here, and I'd like to ask Mr. Steven Henry from Wolf, Greenfield & Sacks, came all the way from Boston.

--oOo--

STEVEN HENRY

WOLF, GREENFIELD & SACKS, P.C.

MR. HENRY: Thank you, Mr. Commissioner and distinguished panel members. I'd like to begin my comments just by stating who I am. I'm a patent attorney in a large intellectual property firm, approximately forty-five professionals, about half of whom deal with the computer industry, hardware and software. We have considerable experience in our client's experiences on all sides of these matters.

To back into my remarks, I am a strong advocate of the patent system and I have seen it work time and again in the software industry as well as other industries. I have seen no fundamental differences in the software industry other than tentativeness in applying the existing rules, and the problems that other speakers have addressed with respect to the ability of examiners to get at the prior art, which is indeed a serious problem. I don't believe the software industry operates under substantially different economic principals than any other industry, or that the people in that industry are driven by a different human nature.

Professor Hollaar addressed many of the points, made many of the recommendations that I would like to make to this body, and I certainly endorse what he said. I'd like to, before proceeding, go one step further and address a topic or two that he did not address, and specifically the issue of reexamination as a cure for defective examination in the first place. If one looks at the statistical studies that have been done of reexamination, and one takes into account the kind of anecdotal experiences that we have had, reexamination is tilted in favor of supporting the conclusions originally reached by the Patent and Trademark Office, not through any intentional bias, but that's what the statistics indicate; and number two, it is severely limited and was intentionally limited when it was fashioned, limited to consideration of patents and printed publications. The problems of examiners not understanding what they're looking at not addressed, the opportunities for testimony are not provided. If one has an initially-weak examination and it is then reinforced by a faulted reexamination system, we've compounded the problem; we haven't addressed the problem. Though it takes money principally to free up manpower to hold hearings and to broaden proceedings, I believe that there is no cure for the problem other than the money, the time and the increased training.

In written remarks we will address the overall legal and theoretical issues raised in your Notice. I'd like to take a few minutes to talk about some practical, anecdotal experience.

COMMISSIONER LEHMAN: Can I ask a question? You know, you're talking about the money that would be involved and the change of procedure that would permit us and maybe encourage us to take oral testimony and to get at nonwritten prior art, but to some degree -- life is not, you know, totally fair, but to some degree, and I assume that would partly be on the motion of the parties seeking reexamination if you wanted to have reexamination just on the basis if you couldn't afford, for example, to support coming to Washington, getting witnesses there and so on and so forth, you could still go forward with the written record. I mean it's not automatically implying a greater burden, financial burden for everybody.

MR. HENRY: Certainly the requester could go on a written record if the requester so desired. It may well be that the Commissioner should consider some way of developing a fund wherein if the examining group thought it would be appropriate to have a hearing of some sort, and the requester is not able to bear that expense, that there may be other resources brought to bear to be able to fly appropriate witnesses in. Because I think faith in the system is something that's extremely important and right now that's what's lacking. It's lacking in part because of media attention on a few glaring mishaps in the system, they're not the rule, they are the exception, but it so happens that the exception gets the attention.

To turn to some of the times we've seen the system work, I'll try to give a synopsis of a few experiences, hopefully without identifying the companies. In our first case I have a client that's a small software company on the West Coasts, initially financed through the founder's own resources. This is a utility type of software, improving hardware performance and reliability. They filed a patent application; a hardware company that they were working with decided to flex its muscles a bit and threatened to design their own product, notwithstanding the patent application. However, once we had an indication of reasonable allowable claims we were able to negotiate them back into the fold.

A few months later, despite the success of the product, as we all know, it's extremely expensive to get software into the marketplace and marketing expenses were just eating up the company's cash.

The company went to look for investors. Every single investor refused to get actively involved until knowing that there would be strong patent protection, because the one thing that makes software unique is how easy it is to copy. And I'm not using copy necessarily in the copyright sense, but analyzing it and taking what's there.

This was a situation where fortunately the system and some public servants in the patent and trademark office, very sensitive to issues such as this, responded and dealt expeditiously with the response we had filed to an outstanding action, and indeed allowed very broad claims, and our client is at this point closing the financing which was the difference between life and death for the company.

We represent university clients also. Universities will generally not be able to license their technology unless they have chances of protecting it. They are not known to be litigious; it is out of respect for the patent system and access to future technology generally that a licensee signs up. We've seen a number of instances where software developed at universities was licensed by the very developers who knew the potential, went out, formed their own companies, and that was a revenue stream that was formed back to universities; and that revenue stream is very important.

We have investors come to us, any number of times, thinking of investing in software-related companies, and their question again is, "Is this protectable? If I'm going to put in my millions and millions of dollars and all of my effort, will someone else be able to come along and walk off with it?"

In those situations where our own investigations of prior art or the Patent Office investigations of prior art make it questionable that strong protection is available, generally an investment is not made. Where, however, it appears that protection is available, an investment often is made. We don't want investors to start getting gun-shy about investing because subsequently we find out that the examination -- search process in particular -- is defective. The best thing we can do at this point is everything reasonably possible to beef up that process.

That will have carryover effect, as Professor Hollaar mentioned, with respect to the copyright system. The

copyright system is drawing a great deal of fire because of the look and feel and its progeny and uncertainty. Investors and business people look for certainty, and it's our job to move the system in the direction where they feel a lot more comfortable with it.

Thank you.

COMMISSIONER LEHMAN: Thank you very much.

Basically it's been your testimony, and it's very strong, that in your experience representing clients you've seen a number of very specific examples where investment in innovation would not have occurred had it not been for the patent incentive.

MR. HENRY: Absolutely.

COMMISSIONER LEHMAN: Thank you very much.

Next I'd like to call Sal Cassamassima, General Counsel of the Exxon Production Research Company.

--o0o--

SAL CASSAMASSIMA

EXXON PRODUCTION RESEARCH COMPANY

MR. CASSAMASSIMA: Thank you, Commissioner LEHMAN, and thank you particularly for pronouncing my name correctly. I know it's a struggle to get that one right the first time. My name is Sal Cassamassima, and I'm the General Counsel of Exxon Production Research located in Houston, Texas. I'm here to present testimony on behalf of Exxon Production Research, which for the sake of brevity I'll refer to as EPR.

I'm going to address the subject of patent protection for software-related inventions and more particularly on the patentability of inventions containing mathematical algorithms. This is a very important subject to EPR, and we appreciate this opportunity to present our comments on the subject. We will also submit more detailed comments, written comments for the record by the March 15th deadline.

Our comments will focus on the subject of inventions containing mathematical algorithms, and we will recommend that the Patent and Trademark Office clarify and liberalize its guidance on the subject to better align with the views of the Federal Circuit.

Let me tell you first a little bit about EPR and why the subjects addressed here today are important to us. We are a wholly-owned affiliate of Exxon Corporation and we're engaged in basic and applied research related to oil and gas exploration and production technology.

Our company and many other companies in the oil and gas industry are among the most intensive users of advanced computer technologies and applications. For example, in exploring for oil and gas, there is an increasing need for highly accurate representations of subsurface formations, particularly in geologically-complex areas. Meeting that need in recent years has been an enormous challenge for industry, particularly with oil hovering around fifteen dollars a barrel.

That challenge is being met by rapid advances in the application of leading edge computer technology and the processing of geophysical data which you probably know is obtained from seismic surveys. For example, so-called three-dimensional or 3-D seismic provides much more accurate depictions of complex geologic formations than was ever possible with more conventional two-dimensional, 2-D seismic. Advanced computer applications are also being used on the production side of our business to predict oil and gas reservoir drainage and to model enhanced oil recovery techniques by computer simulations.

In pushing the frontiers of this new technology, EPR is constantly challenged in several areas that are computer-related.

First is the hardware itself. Seismic surveying and the ensuing data processing require enormously powerful computers such as supercomputers and massively parallel computers. To obtain these various depictions of 3-D seismic you really have to have an enormous amount of number-crunching capability. In fact, we think our industry is second only to the Defense Department in the use of MPPs and supercomputers.

Secondly, the industry must develop the software to both process the data and convert it into readily-analyzable forms; and this is where we come in. Closely related to hardware and software development is the ongoing challenge to develop sophisticated mathematical algorithms which are the key to enhancing analysis of seismic or reservoir data. The algorithms we develop do many critical things in analyzing seismic and reservoir data.

For example, the algorithms enable the computer to process data more efficiently. They compress data or rearrange data to make it more readily processible, and they enable the processing of poorly-conditioned data by removing noise or other irrelevant signals.

Development of these algorithms and their integration into the hardware and software usually involves very major investments in both time and money. The synergistic combination of more powerful computers and software enhanced by mathematical algorithms has resulted in a quantum leap in oil and gas exploration capabilities. Many companies are now reexploring mature producing areas such as the Gulf of Mexico because the new technology enables the discovery of reservoirs that were heretofore unknown because of their complex subsurface geology.

The type of inventions we seek to protect generally relate to methods for analyzing seismic or reservoir data using mathematical algorithms which yield a desired output, such as an accurate 3-D depiction of the subsurface. For example, these methods may accurately identify salt domes, highly faulted formations, and other complex subsurface anomalies which reveal oil and gas deposits. Patent applications claiming methods for analyzing seismic data using mathematical algorithms have always been among the most perplexing cases for the Patent and Trademark Office to review. In some cases, the Office in applying the so-called Freedom Walter Abel test, two-part test, has held such claims to be patentable subject matter under Section 101. In other cases, very similar cases, the Patent and Trademark Office has found the test not to be satisfied.

It was thought by many in the industry that some clear guidance would be forthcoming from the Patent Office when the Federal Circuit issued its opinion in the Arrythmia Research v. Corazonix case which some other speakers mentioned today. However, we have been disappointed that

the Patent and Trademark Office often does not appear to follow the reasoning of Arrythmia, thereby creating a great deal of uncertainty in this important area of the law, and I realize that consistency is the hobgoblin of little minds, but in the area of Section 101, the threshold of patentability, consistency is very important.

COMMISSIONER LEHMAN: I think that consistency in the Patent Office is an extremely important part of customer service, so I don't think that cliche is applicable to us at all, and I think this is a very good point that you're making.

MR. CASSAMASSIMA: The types of inventions that are the subject of this controversy all have one thing in common. They deal with algorithms which yield a useful result. For example, in many of our inventions seismic signals are analyzed to accurately depict subsurface geology. In the arrythmia case, electrocardiograph signals were analyzed to detect the susceptibility to excessively rapid heartbeat, which is known as tachycardia, a very life-threatening illness. In other cases, we might see techniques for mathematically analyzing molecular structure to screen for chemotherapy agents. But let me give you a hypothetical example that may be more meaningful, given recent events. I'll describe this hypothetical invention: A method of analyzing seismic data to predict earthquakes, said method comprising combining seismic signals using Formula X, mapping said combined said combined seismic signals using Formula Y, comparing said map seismic signals using Formula Z with a database of historically-mapped signals to determine the probability of an earthquake.

Now of course I have no idea what X, Y and Z formulas might constitute, but needless to say, that would be a rather dramatic invention.

Question: Is the method I just described patentable subject matter? And I would say that under current Patent and Trademark Office policy, the answer is hard to concern. Should it be? In my opinion, absolutely yes. The method that I just described, claimed, is not an abstract idea of a law of nature. It is a process for analyzing real physical data to yield a highly-useful life-saving result, the prediction of earthquakes. The earthquake claim does not seek to protect a generic technique for analyzing data. It does not claim a purely mathematical method of identifying a data anomaly by comparing sample data to generic databases. What it seeks to protect is a process for determining the probability of an earthquake by comparing in a quantifiable manner actual seismic data with reference databases. By grounding the claim in seismic signal analysis the claim comes to life as a patentable process, and it's protected by the Patent Act. It does not matter whether there is a discernable physical component to the claim itself.

Inventions, such as the ones I have described, our type of inventions, the Arrythmia case or the hypothetical, all have the requisites of patentable subject matter. They are new and useful, and granting patent protection to them would foster innovation and technology development, the very purpose of the patent law. Absent patent protection, the time and the resources to develop such new and needed technology might not be forthcoming.

We therefore recommend that the Patent and Trademark Office dispense with the two-part Freeman-Walter-Abel test and issue new guidance that embraces a statutory basis for determining patentable subject matter under Section 101. The guidance should be simple and as broad as the statute and judicial precedent permit. We recommend that the word "process" be given its literal meaning and let the guidance dispense with the notion that a method claim containing a mathematical algorithm cite some physical step. Only algorithms which solve abstract or generic math problems should be deemed nonstatutory. The guidance should also direct that the algorithm be viewed in the context of the specification as a whole, in the claims preamble. With that approach, method claims of the type contained in arrythmia and the ones I've described would be statutory subject matter.

Finally, the guidance should make clear that in the absence of legislative limits by Congress, the Patent and Trademark Office will not impose nonstatutory or policy constraints on what processes are worthy of patent protection. Such guidance would eliminate the present uncertainty and would provide a boost to the type of technology that will come to dominate the Information Age in the years ahead.

Thank you very much.

COMMISSIONER LEHMAN: Thanks very much, Mr. Cassamassima, I really appreciate all your very specific recommendations. We'll examine those.

Next I'd like to call Christopher Palermo, another Fish and Richardson attorney. Is he here? I guess not. In that case we'll move on to Neil Brown. Mr. Brown? We're ahead of schedule now.

MR. BROWN: Sorry about that. I was expecting twenty minutes, but --

COMMISSIONER LEHMAN: I know; we're a little further ahead in the schedule than we thought, mainly because the preceding witness wasn't here.

--oOo--

NEIL BROWN

INDEPENDENT SOFTWARE ENGINEER

MR. BROWN: Well, I have a lot to say. First of all, I'd like to thank you all for being here. I have a lot to say and I appreciate the opportunity to say it to you directly. It's been a long time since I've been on stage; sorry. My name is Neil Brown. I'm an independent software engineer. I work as a contractor for software development companies and I do development of my own. I did programming for fifteen years and I've been getting paid for it only in the last six.

Although I can't speak for all of us, I can speak for some of us, and I can certainly speak for all the friends I've talked to who feel very similar to the way I do. I represent the ultimate source of all revenue for every person who profits from the software industry, however indirectly, the developers; if somebody wasn't writing the software, there wouldn't be a single software patent lawyer that could possibly make a penny. This is why I -- what I, what the League for Programming Freedom and what the Free Software Foundation, neither of which I'm a member, have to say is important. We power this industry; we're the dynamo which causes it to exist, and in order to keep our jobs, in order to continue marketing software, we must make clear what we need before its too late.

I'd like to address the questions that you've asked, as I

indicated from not listing affiliations, I'm not speaking on behalf of anyone other than myself and those in the software industry whom I have found to agree with me.

On the question, Topic A, Question 1, Example A, "What part of mathematical algorithm implemented on a general purpose computer can be patented?" My response: None whatsoever.

The technique of long division where one writes the number to be divided down, puts a little bar under it, and writes the number to divide into it, next to it, writes notations and partial answers, gradually arriving at a more complete answer, can clearly be described in a fashion executable by a computer. It's not at all hard for most developers to write a graphical front end for long division, so this is a useful, I mean the question of whether or not division is useful is, is not worth debating. It's a useful tool to accomplish a useful goal, and real money is made from using it, but what would happen to your education? How would you have learned division if the school that was trying to teach you that suddenly found itself being attacked by another school who claimed to own that very method?

A mathematical algorithm performed on a special-purpose computer; can you patent a calculator? I do seem to remember that the beginning of the digital revolution was really noticed when pocket calculators starting causing slide rules to disappear. If only one company had been able to produce calculators, would the price have dropped from four hundred dollars to fifteen in only a couple of years? If only one company could own the legal right to build a machine to perform mathematical calculations, where would the software industry have gotten its start?

Topic A, Question 1, Example B, sorry, Example C and C-2. "Can you patent the disk on which a computer program is stored?" Again, with the calculator, if it were possible to patent the concept of a calculator, if it were possible to patent the calculator that has the ability to execute more than one program, where would the software industry be today? How do you define what a program is? Do you define it as being able to push the sine key and get the sine of the number? Do you define a separate program as one in which you can press the cosine key and get the cosine of that number?

Question 2. "What impact, negative or positive, have you or your organization experienced from patent issues on software-related inventions? On several occasions I have found myself being unsure of whether or not I was able to use a particular algorithm, specifically the compression algorithm embedded in the program known as Compress, within software. There have been many questions raised and lots of time spent chasing after whether or not the company could somehow use this and escape any royalty obligations.

"What implications, positive or negative, can you foresee in maintaining or altering the standards for patent eligibility?" This is Question 3. Well, I see small guys getting squeezed out. I see innovation becoming more and more difficult, because every one of the hundreds of ideas that the developer goes through while writing an application, every one of the little techniques that he goes to use or goes to put together with another, he has to go and call up Legal to find out if that's been patented or if it might be covered by a patent. The software industry is not going to progress very

rapidly if people like me spend all their time on the phone to Legal asking if we can do this.

"Does the framework of patent, copyright or trade secret law," Question 4, "effectively promote innovation in the field of software?" Yes. "Does it provide the appropriate level of protection?" A qualified yes. The qualification is that it provides too much protection, potentially.

Question 5. "Do you believe a new form of protection for computer programs is needed?" My answer is no. The water is muddy enough.

On Topic B. I agree with all concerns that access to prior art is difficult, or is outmoded. The difficulty of determining whether or not two programs are equivalent or similar is extremely difficult. I deem it intractable. There are so many languages out there, there are so many sophisticated ways of expressing algorithms that it's hard enough just to understand one, but comparing two? Doing this for every program out there that seems to possibly be related to an application can take forever. The very concept of, is there anything out there at all that does what this does, is extremely difficult to solve, and I deem it to be intractable for software in general.

Topic B, Question 1. No, I don't think that the patents and printed publications provide examiners with sufficient collection of prior art, and as I said before, it can't. The work on software interface patents, if you allow patenting of the idea of having a hammer on a desk and deem it a different invention if the hammer is the drawer of the desk, how is somebody using that desk going to be able to get their job done? How is somebody going to be able to design -- How is anyone going to be able to get the job done if their job is to design a new desk and they have to go and find everyone that has similar functionality available at the top of their desk? The patent which has control information such as page numbering and position on the page and document being edited available on the screen, the very idea of having many pieces of information available for manipulation of the information is the whole idea of an interface. You want to be able to provide as much ability for the user to manipulate the raw material they're working with as apparently is possible. You want all of these tools to be easy to get to and easy to work with, and if someone comes with a formalism for making all of these things available, then if any means of providing that same functionality is deemed equivalent, then how can progress possibly happen?

I do have lots more to say, as I said, but --

COMMISSIONER LEHMAN: Well, maybe I can help you to wrap up in just asking you, I have a little confusion in your statement to us; in the answer to Question 4 you basically seem to say that the present framework is okay, but I have the impression that you basically don't think that -- that you think there are a lot of problems with the patentability of software just generally. Is it your position that software should not be patentable?

MR. BROWN: Software patents are a blight. They're a problem. They get worse.

COMMISSIONER LEHMAN: So basically you think that copyright protection is --

MR. BROWN: Copyrights and trade secrets.

COMMISSIONER LEHMAN: -- is okay, and trade secrets,

but that basically the difficulties from trying to work with the patent system applied to this industry are so great that it virtually makes it impossible to use it as an effective technique for protection that developers like yourself can really work with.

MR. BROWN: Yes. How many houses would you build if every time a carpenter went to build a house, every time a carpenter went to take up a tool he had to pay a point one percent royalty on the gross profit on that house, or the gross revenue on that house?

COMMISSIONER LEHMAN: We appreciate your coming and sharing these comments. Thank you.

MR. BROWN: Thank you very much.

COMMISSIONER LEHMAN: Next I'd like to ask, if he's here, Gordon Irlam, representing the League for Programming Freedom, which our previous witness referred to in his statement.

--o0o--

GORDON IRLAM

LEAGUE FOR PROGRAMMING FREEDOM

MR. IRLAM: Good day. The League for Programming Freedom's an organization of roughly six hundred people within the software industry. It's a combination of software developers employed by various companies and small people who own their own business which typically are anywhere between like one or two people up to fifty or a hundred people, and they also have a number of members that are either academics, researchers or students. The League for Programming Freedom has two main policy areas it's concerned with. It has a belief in doing software development on the basis of competition and between different implementations of what could be the same technology, and we believe by doing that that such software is going to be in a sense like economically more efficient impact, if you can get multiple products that will represent the same technology; and prices -- and efficiencies are going to be driven up by the competitive ventures involved.

So to do that we believe in what you might term a traditional literal aspects doctrine of copyright. That's very useful for software developers. It basically means if you copy code, you know, you can't do that, but if you wrote the code yourself, that's all you need to know. You can then go out and sell it, and you're safe in the knowledge that you can, and you won't be sued later on. You know you owned it.

So based on this belief the League has two positions. We're opposed to the look and feel copyright, and extensions that seem to have been happening over the past few years, and we're also opposed to software patents. And we take various actions to try and raise these issues and submit these to courts and so on.

Fundamentally I think this whole question of software patents is being approached in what might be the wrong direction. In fact I think the whole issue has to be looked at as one of economics, and not either based on, you know, the direct, uh, interests of any party or, uh, you know, you've got to stand back a bit and take a big picture view of, you know, what effect do software patents have on competition and market structure. So therefore I'm rather disappointed, but as far as I know, there hasn't been anybody with any

economic background that's been speaking on these matters. And you know, I think it would be really important if the Commission or whatever could like seek out people with an economic background that can provide the input on these matters. And so I think, you know, if you start analyzing patents and in particular software patents from an economic standpoint, then you'll get a lot of benefit. But I think it should be obvious that every industry has different economic characteristics. This is both like market size, market structure, with availability of market information, extent of competition, and there's just a huge range of economic parameters that differentiate, for instance, the pharmaceutical industry from the software industry, and you know, because of this, the application of patents to one industry might be sound public policy, but the application of the same rules to another industry will have an adverse effect on the overall public welfare.

So for instance, one of the big things about computer software is it's protected by copyright in a way that pharmaceuticals, for instance, aren't. And that provides a good basis for allowing different people to develop different products and compete, so you know, the important thing is if you look at it from an economic standpoint, every industry's different, and so the impact of patents will be different on every industry; and I contend that in the software industry, patents are harmful.

And I believe it's because, based on my experience, the software industry appears to be a highly competitive industry, and I feel software patents have potential to stifle this competition in terms of they can take away profits from, you know, some of the firms, perhaps significant value to the industry, and they'll be transferred to firms that don't add a lot of economic value.

I think the important thing about software is it's not so much new ideas that are important; it's building real products that solve customer problems, and I'm a software engineer in my regular employment, and my job doesn't consist of trying to come up with new ideas all the time. There's just hundreds of ideas. It's trying to implement those ideas, build real products that solve real customer problems and, you know, if they solve them well, that both they integrate well with other products and are -- and easily usable. I think if you look at the last ten or twenty years with like the microcomputer revolution, that's what it's really all been about, you know, the firms that have been successful are the firms that have been able to take ideas and turn them into something that's useful for end-users, and add value that way. So you know, the successful firms are companies like Microsoft, Novell, Borland, Adobe, and you know, the list goes on, all these new companies that have, you know, typically been started some time in the last ten to fifteen years.

And so when you start looking at software patents you'll find that it's quite disturbing because if these are the companies you want to attract, if you actually start to count which companies have how many software patents, you'll find things that are quite alarming in terms of the companies that you'd imagine should be being rewarded by the patent system for developing real economic value hardly hold any patents. And indeed, many of them don't even hold any patents at all. And even the large company such as

Microsoft count very few patents. And on the other hand, if you look at companies like Hitachi and IBM and AT&T which people within the software industry will tell you have been more or less totally inept at bringing their software ideas to market, they can come up with ideas but they just can't implement them successfully in a way that fulfills real customer needs.

And so I think if the software patent system continues, then it's going to have a very adverse effect in terms of resources from the economy are going to be diverted away from the firms that are adding real value and put towards these very large conglomerates which are like multifactored concerns that have large patent hierarchies.

And so because of this I think, you know, the patent system's got to be analyzed from an economic point of view and, you know, in the case of software I believe that that's -- the patents are harmful to the software industry and I think the best solution will be to make software nonpatentable.

You know, I'd like to just mention briefly on the issue of economics, there seems to be very little real research into like the fundamental functioning of the patent system in terms of I know there was a study way back in I think 1958 by Fritz Matlock (phonetic) that, you know, did an economic evaluation for the Senate, and raised an awful lot of questions about the patent system, which I think was good, but then they tended to be just left off, and I feel, you know, if someone had taken those questions and started evaluating them and gathering real data, we'd be in a much better position today to actually know what the state of the patent system is, exactly how it works on deciding important public policy issues such as this.

All right. In the remaining time I'd like to just read, if I may, one or two things by some of our members who haven't been able to attend today, have sent. Okay.

So this is from James Hellman (phonetic) who's a member of the League for Programming Freedom, and he says, "A couple of years ago I was involved in a start-up that was shut down by a bogus software patent. We were well on our way to having several hundred thousand dollars of private-placement venture capital. Out of the blue another company was awarded an extremely broad software systems patent for an obvious concept through substantial existing prior art. We received a cease and desist letter and our funding evaporated. The sad thing is that the company that received the patent were so incompetent that they went out of business shortly afterwards. Business success should be determined by who has the best ideas, best implementations and best marketing.

Okay. Have you got any questions?

COMMISSIONER LEHMAN: No, other than to say that I think you make a very good point about lack of really effective economic analysis about how the patent system works. It is something that is lacking. I know we've attempted to survey the literature on that subject, and there really isn't any. It's interesting that certainly this hearing was made available to everybody; we're within a few miles of one of the great research institutions in this area of the country. Nobody, no professor, no great economist saw this subject as worthy of advising us on and it's a problem we have, frankly, it's a problem, and unless we go out and commission

the work to be done I'm not sure that we're going to get that kind of information, but it does put us at some kind of disadvantage so we have to do the next best thing and that is get the kind of anecdotal information that we're getting here today from people like you and other witnesses. The difficulty with that is that we're hearing exactly the opposite thing from several different witnesses. We heard people earlier who testified that it's an absolute fact that there would not have been investments made in innovation, companies would not have been formed if they had not had the patent incentive. You've given us in your letter that you just read to us an assertion of an exact opposite situation. So we'll have to sort through all of this.

MR. IRLAM: I believe there will be value if, you know, longer term, the Patent Office was to maybe develop some of its own skills at doing like economic analysis for its like policy section or whatever that may exist.

COMMISSIONER LEHMAN: Well, we may well have to do that. Thank you very much.

Next I'd like to ask Mr. Robert Yoches to step forward, who has, I assume, come all the way out there from Washington, Finnegan, Henderson, Farabow, Garrett & Dunner.

--oOo--

ROBERT YOCHES

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER

MR. YOCHES: Thank you, Commissioner LEHMAN, it is indeed a pleasure not to be in Washington, D.C. today because of the weather, and also a pleasure to be before this distinguished Panel. For the record my name is Bob Yoches and I am a partner of Finnegan, Henderson, Farabow, Garrett & Dunner, although I speak today not as a representative of that firm, not as a representative of my partners, and not as a representative of any of the clients of the firm. Instead I offer my own views based upon having practiced in the area of intellectual property for patents for fourteen years, and in that capacity I've been before the Patent Office in prosecution, I've litigated patents in the software and computer area, I've licensed patents in those areas, licensed in and out technology, I've been involved in copyright registrations, copyright licensing and litigation, and I've been involved in trade secret litigation. I've represented both large companies that have been well-established, small companies and start-up companies. I've also represented domestic companies and foreign companies, and I've represented those that had intellectual copyrights and those that were concerned with other parties' intellectual property rights. And based upon that experience I'd like to offer some observations about the applicability of the patents and the patent law to the software-related inventions. I'm going to restrict my remarks to perhaps unique aspects of software that make patents appropriate or inappropriate, as opposed to addressing any of the general attacks on the patent system itself. I'm not under the impression that there's any large-scale movement to rid ourselves of the patent system, so let me address myself to the specific aspects of software and the specific aspects of how the patent system impacts software.

I note that in the discussions today and the testimony given there are three characteristics of software that I think are

important, especially from the aspect of how best to implement the Constitutional directive. One is that software is pervasive in our technology. It pervades our lives, it pervades our jobs, it pervades all other types of technology that hithertofore we've considered different. We've heard somebody from the petroleum industry talk. We've heard about a case involving software and the medical industry. There is software in the banks, software in stock exchanges, there's software in your automobiles. Not only does software now pervade our lives, it will do so more in the future.

A second observation is, you cannot extricate the software and treat it separately, in other words, I question whether we can talk intelligently about software-related inventions, software's such an integral part of our lives. And lastly, the last observation generally on software is that although software has some unique aspects, so does every other type of technology; certainly biotechnology has unique aspects. Certainly chemistry and pharmaceuticals have unique aspects, but I think there are some things about software that it has in common with other innovations and with other technologies that are particularly important to how patents will affect that.

One of those characteristics is software is extremely useful; it is, as I indicated before, pervasive, but it has the potential, and has already actualized much of that potential for dramatic impacts on our life.

Second, the more we know about software, and I think there is some testimony on this point also, the more that is known about software, the greater will be the development in order to avoid ploughing ground that's been ploughed, and the faster will be the rate of that improvement.

Given those observations then, what role do patents play and can they play? And I think patents have already played, and will continue to play, a role in three major aspects. The one generally starting from what I just talked about is in publication. We heard a speaker this morning testify that indeed one large company allowed publication of ideas because they were protected by way of patents. In addition to that, however, and I think a much stronger point, is the fact that the patents themselves are publications of the ideas, and publications in a very important way that really hasn't existed. They are publications of information in a structured format by way of the Patent Office's own classification system. As the Commissioner spoke this morning, the fact of the matter is that because we have relied so long on trade secrets there is perhaps a lack of this structured database. The way to solve that is not to avoid the patent system, but rather to embrace it, and to look at and perhaps adopt many of the recommendations that have been made here on how to improve the accessibility to prior art.

The second issue, and this has been a key issue here, is that of investment, and rather than repeat what's been said, it has been my experience that not only do investors, and by investors I not only include venture capitalists, but also large concerns that are interested in some sort of partnering agreement, but these type of investors care more about patents than they do about trade secrets, if in all honesty copyrights are kind of a wash. They're there anyhow, it doesn't make much difference.

But given that, there's often a choice between whether to

keep processes secret or obtain a patent on it, I find investors like patents much better, for two reasons. One is, they don't like dealing with trade secrets because they have to sign a confidentiality agreement and a lot of investors won't do that. The second reason is, and I think even more compelling, is that the investors are afraid that the trade secrets will have a short lifetime. They can easily be lost. They can be lost in an instant by an inadvertent publication. They much prefer patents.

The third area that I think that patents play in software is that of innovation. There's been I guess some dispute here on whether software is fast-developing or slow-developing, but I think there is one observation we can make, and that is, it's generally easy to change software. It's more flexible to change software than hardware, indeed that's why so many of our developments have software in it. Well, of course one of the options that the patent system offers, and one of the opportunities it offers, is that if there's a patent out there, and you don't feel like paying the license fee for it, you are encouraged to design around the patent, and indeed the Federal Circuit has indicated that a key aspect of the patent laws is the designing-around.

Software, by its nature, by the ease and quickness by which you can modify your procedures and modify your algorithms, is particularly adapted to designing around other patents, and particularly adapted to then promoting new developments. It has been my experience, in summary, that the patents have served the software type of developments very well, and I believe in general that the Patent Office, especially in Group 2300 with which I've had the most experience, has also done a good job of serving the system well, but I notice, I think, two problems currently, with the Patent and Trademark Office in the area of patent protection of software-related inventions. The one is, I believe in the Section 101 area as I think other witnesses have indicated, that there is a reluctance, and almost stubborness by the Patent Office to taking the most contrary position that they can on whether subject matter is patentable, and indeed in the form paragraphs which the patent and trademark office uses as a bases for its rejections, it had been able to pick and choose among different cases, especially cases from the 1970s, to support their positions.

I think that's contrary to the trend of the law. I think it's contrary to two major Supreme Court cases, the most recent cases in this area. Because in the Jacobardi case, as the Notice indicates, the patent laws are supposed to extend to anything under the sun made by man, and in the Diehr case, there was a direction that we're supposed to look at the claim as a whole and not dissect it into its old elements, meaning its mathematical algorithms, and its new elements. And I do not believe currently that the Patent and Trademark Office is following that, and I think that the result has been, at least in my experience, two things. One is frustration by some applicants because they have abandoned their application rather than pursue this to the Board, and for those people that have pursued to the Board, at least in our firm, they've been very successful, and all it's resulted in is an additional expense to those applicants.

The other issue, and I think the Patent Office I understand the last week made I think a major change, is the Patent Office I understand now allows or will allow Group 2300 to

hire computer scientists as examiners. I think that's a very good step. However, it's my understanding though that if you're a computer scientist out practicing in the world you may not currently sit for the Patent Bar. The belief is that you don't have sufficient technical training. I think that should change, and certainly if you're qualified enough to examine patents, you ought to be qualified enough to prosecute those patents in front of the Office.

Thank you very much.

COMMISSIONER LEHMAN: Thank you very much, Mr. Yoches, for coming all this way to share those thoughts with us.

Next I'd like to call our final witness of the afternoon, Jim Shay of the firm of Morrison and Foerster.

--oOo--

JIM SHAY

MORRISON & FOERSTER

MR. SHAY: I find myself in the very difficult position of playing clean-up and of trying to say something new, because many good things have already been said and many of my remarks I think only serve to reinforce those things, but perhaps that's useful as well. My name is Jim Shay, I am with the law firm of Morrison & Foerster in San Francisco. We represent the Multimedia Development Group, a trade association based on San Francisco, as well as other clients in the software and multimedia industry. My comments today are my own, however. They do not necessarily represent the views of the firm or its clients.

I am a patent attorney. I spent three years as a patent examiner before entering private practice. I've also served as inhouse counsel for a medical technology company, and I've worked in a variety of technologies in a variety of ways, prosecution, litigation, licensing, representing individual inventors, large companies and investors.

I believe in the value of patent protection as a tool for spurring innovation and for helping inventors, whether corporate or individual, obtain the benefit of their contributions. In my opinion this principle applies as much to software-related inventions as to any tangible mechanical, chemical or electrical invention.

Specifically, the software industry as a whole and software companies and developers individually benefit from the patent system. The software industry I'm referring to is not just the companies whose primary products reside on floppy disks or CD-ROMs. In my view the term software industry includes any suppliers of products incorporating programmable microprocessors, products such as medical monitors, animated toys, automobile electronic ignitions, audio products, just to name a few. Advances in microprocessor technology have made software ubiquitous and protection of patentable inventions embodying that software is therefore of concern not only to companies writing and selling software per se, but also to all manner of high, medium and low-tech companies serving a variety of markets.

As the PTO has acknowledge in conducting these hearings, there appear to be a particularly high amount of concern over the validity of software patents. A good example is the public reaction to the Compton's new media patent, a patent

I came to know very well in my position as counsel to the Multimedia Development Group. I participated in question and answer sessions about the Compton's patents with members of the MDG's Executive Committee and with individual members of the MDG. Many expressed many strong, negative opinions about the conduct of the Compton's patent applicants before the PTO and about the ability of the PTO to examine and issue valid patents in the subject area.

My review of the file history of that patent, however, showed no evidence of any particular lapse or failure on either part. Nonetheless, the consensus of nearly all to whom I spoke was that the broadest claims of the Compton's patent could not possibly be valid, and that anyone associated with the multimedia industry would agree. The eventual disposition of the Compton's patent remains to be seen. The discussion surrounding that patent, however, has pointed to some possible deficiencies in the current patent system, especially as applied to software-related inventions.

First, as other people have noted, patent examiners do not have easy access to the best prior art for software-related inventions. The best prior art consists of actual software, operators manuals, research papers and the like. These references are not generally accessible to patent examiners.

Second, the PTO's relative lack of experience in software-related inventions because of the relative newness of the patentability of software makes it difficult for an examiner to determine how one of ordinary skill in the art would have approached the problem that patent claims address. Often it is the feeling that an invention would have been obvious that leads an examiner to find the most pertinent prior art references, to make the most compelling argument regarding the unpatentability of the claims. This disconnect between the gut feelings of the patent examiners and the gut feelings of skilled artisans in the software industry undermines the industry's faith in the PTO.

I would now like to make some recommendations based on these observations. These are not new, these will merely reinforce other recommendations made earlier today.

First, operating within the current statutory framework, I believe that the PTO and the software industry could benefit greatly from a more formal interaction. Specifically, the software industry operating through industry groups such as the Multimedia Development Group, could provide the PTO with kinds of prior art references that the PTO currently lacks. I have spoken to many members of these groups who at least now are expressing a willingness to work with the Patent and Trademark Office if the PTO will work with them in compiling these prior art references. Such a program would require the industry groups to dig up and send, and the Patent Office to accept and classify, prior art references related to the past and present software inventions.

In addition, the PTO and industry groups should cooperate to train examiners working with software inventions. I'm aware, for example, of training programs offered by the Software Patent Institute. I also believe that the PTO should undertake the task of teaching the software industry about the patent process so that the industry can use the existing process more effectively. One of the most surprising things I learned in the Compton's process was how little people

actually knew about the patent system.

To the extent that the PTO is willing to support statutory changes and as I learned this morning, you are, I believe that a system of pre-grant publication and opposition proceedings would help improve the quality of software patents. This one aspect of change is the one thing mentioned more often by more people in discussing the current patent situation. A less radical statutory change would seem to be opening the reexamination process to provide for full participation by interested parties in addition to the patent owner. I advocate the use of oral testimony. Experts in the field can be the best source of prior art, and this would be useful in the reexamination process. This change could encourage the submission of all relevant prior art instead of the current practice of withholding the best prior art for use in license negotiations and in District Court infringement proceedings.

In conclusion, while the emphasis of our remarks has been on the deficiencies I perceive in the patent system, I should state that I believe that there is much right with the current system. Our proposals will only be minor changes to a system that has served us well in promoting the useful arts.

Thank you.

COMMISSIONER LEHMAN: Thank you very much, Mr. Shay, I appreciate those very specific recommendations.

I'd like to thank everybody in the audience for having the interest in what others had to say, to stay all day and be with us, and we'll reconvene tomorrow morning at 9:00 o'clock, and our first witness at that time will be Jerry Fiddler, CEO and Chairman of Wind River Systems. Thank you very much.

--o0o--

UNITED STATES PATENT AND TRADEMARK OFFICE
PUBLIC HEARING
ON USE OF THE PATENT SYSTEM
TO PROTECT SOFTWARE-RELATED INVENTIONS
January 27, 1994

COMMISSIONER LEHMAN: Good morning. Welcome to our second day of hearings on the use of the patent system to protect software-related inventions.

Yesterday we had an excellent series of speakers. I think we all learned a lot here, those of us who came from the Commerce Department in Washington. You all gave us a wide variety of opinions and I know that today is going to be just as good and we're going to be armed with all the information we need to improve our patent system when we get back to Washington.

I'd like to just, for those of you who might not have been here today, briefly introduce who we are here on this panel.

I am Bruce Lehman. My official title is Assistant Secretary of Commerce and Commissioner of Patents and Trademarks.

And to my immediate right is Ginger Lew, Assistant Secretary of Commerce and General Counsel of the Department of Commerce Designate, and Ginger Lew was a practicing lawyer until a few months ago up here in the Bay Area and knows this area very well and knows a lot of the industries and businesses that are involved very well.

And then to my far right is Michael Kirk. Mike Kirk is the current Assistant Commissioner of Patents and Trademarks for External Affairs and the President has nominated him to be the Deputy Commissioner of Patents and Trademarks.

And to my left, immediate left, is Lawrence Goffney, and Larry Goffney the President has nominated him to be our new Assistant Commissioner for Patents. He will be running the entire patent operation, with over 3,OOO employees at our Patent Office in Washington, and will play a very critical role in the development of these policies.

And then finally Jeff Kushan is a Staff Member of our Office of Legislation and International Affairs and he is the person who did a lot of the leg work in setting this up and his name is listed, as you know, on the Federal Register Notice.

I'd also like to -- I don't know if Gerry Goldberg is here yet this morning -- but I want to observe the presence of Gerry Goldberg, who is the Director of Group 23O, or 23OO, which is the Software Examining Group.

Is Gerry --? I don't see him around here yet. Okay, well, he'll be here later. I think many of you know him and I'm certain he will be available to you if you have private comments to make to him.

Finally, I'd like to introduce the young lady in the blue suit who just came in is Ruth Ford, who's our Director of Media Relations, and I know we have a number of media and press people who've been here and if you have any needs that need to be dealt with, Ruth will be happy to assist you with that.

So I'd just like to basically review again the ground rules that we're going to be operating with this morning.

The people who will be testifying today should have received a schedule indicating their approximate time that they've been assigned to give their remarks and I think we even have that on a table up front. The list is available there.

And I'd encourage all of the people who are going to be talking with us today to be here at least 2O minutes before your time, your assigned time slot. And sometimes we get going a little bit -- we get speeded up, maybe somebody didn't show up and so then we end up having our schedule advanced beyond what we thought it would be.

Each person will have 11 minutes for their presentation and the computer monitor in front of us here will display a green screen for 9 minutes and then it will turn yellow and when the screen turns red that means that the 11 minutes is up. And I encourage everybody to be cooperative with us if at all possible and try to stick to those limits. Otherwise we'll sort of have to politely ask you to wrap up.

To the extent that you can finish before 11 minutes, it's not such a bad idea because it gives us a little more freedom to have a dialogue and ask questions and get really a better sense of where you're coming from and understand your testimony better.

In addition to, of course, these oral comments, we're open to additional written comments from everybody, and additional written comments from all those who are going to be testifying today, maybe something is said by one of the other people that you feel you have to follow up on, and those can be submitted to us in our office and I think the address for all of that has been indicated in the Federal Register Notice that has been circulated through the Internet and was in the Federal Register Notice itself.

That Notice can be retrieved from our FTP site, which is: Comments, period, USPT O, period, GOB.

The transcripts of the hearings will be available after February 7th and paper copies will be available for a charge of $3O. The transcripts will also be available through our FTP site.

Once again, I want to welcome everybody here, and it's wonderful to see that we have this kind of interest for a second day in how we can improve the legal basis for high technology in the United States.

I'd like to call on our first speaker, who will be Jerry Fiddler, CEO and Chairman of Wind River Systems. Welcome, Mr. Fiddler.

--oOo--

JERRY FIDDLER

CEO/CHAIRMAN

WIND RIVER SYSTEMS

MR. FIDDLER: Thank you.

I just dashed in the door. Traffic was terrible.

I stand before you not as an expert in intellectual property law. I'm not a lawyer. Most people in this room know far more about intellectual property law than I ever will. Rather, I wish to speak to you as an expert software engineer and the founder and CEO of a successful software company, Wind River Systems.

Wind River Systems is a 3O million dollar public company, with 4O percent of our revenue from overseas. We create software for embedded systems, the microprocessors found inside our cars, fax machines, telephones, robots, factories,

consumer electronics. According to Software Magazine, last year we were the 92nd largest software company in the US.

My perspective on software patents is simple: stop issuing software patents. Software patents should not exist. I say this for a number of reasons.

First and foremost, I look at the reasons patents exist, which is for the benefit of society. Certainly there are fields where patents are essential because of the large investment involved for creation of technology and the ease of copying that technology. In such situations, patents incent the major investments necessary for those inventions which benefit society.

This doesn't apply to software. Availability of patent protection is not necessary to incent creation of software. Copyright and trade secret protection are entirely adequate and more appropriate. Yes, major investments are necessary to create software, but that investment is primarily involved in quality implementation and support of the software, not development of the algorithms and ideas that might be patentable. Therefore, unlike a drug, for instance, it's not substantially cheaper or quicker to copy a program's functionality than it is to develop the original.

The deal society makes with the inventor, "Tell us about your invention and you can have a monopoly for 17 years," is not a fair deal today when it comes to software. In a field changing as fast as software is today, 17 years might as well be a millennium. The deal might as well be phrased, "Tell us about your invention and you can monopolize it forever," so the fact that we, society, know about it is meaningless.

In fact, patenting of software is actively harmful to society. People don't need software monopolies. They need software that's open, compatible, and that adheres to their expectations and standards. They need the software equivalent of expectations like "accelerator on the right, brake on the left". Patenting of software could only impede these goals.

Furthermore, patenting of software will not accelerate its creation or advancement. Rather, it will impede that advancement, which is far better driven by the free market than by monopoly.

Imagine where we would be today if patents had been granted on technology or concepts critical to word processors or spreadsheets. Rather than the sophisticated and elegant tools we now have available thanks to competition, we would still be using something very much like the primitive first versions of those tools. Worse still, we must remember that word processors and spreadsheets have been largely responsible for spawning an industry and making the personal computer a part of most of our lives. The quality and advancement in those tools have created opportunities for computer manufacturers and for other software vendors who can sell to users who have computers primarily to run those primary tools.

It's not too strong to say that if there had been strong patent protection for the first word processors and spreadsheets, the personal computer industry today might be five to ten years behind where it is. As another example, if aspects of TCP/IP, the network protocol, had received patent protection, today the Internet might very well not exist.

Creation of software will also be impeded by the difficulty of writing software that doesn't inadvertently trip across a patent somewhere. This is true in other fields where patenting is less controversial, but it's far worse in software. It's not unusual for a program to be a million lines long and consist of many thousands of subroutines and functions. Algorithms and ideas are embodied in each of those components and in combinations of them. Some of these algorithms may be studied in school or found in books, but many are developed "on the fly" as the program is created. Many of these subroutines and functions might be far afield from the purpose of the program as a whole.

An operating system, for instance, might contain routines for sorting and searching, handling queues, parsing text, controlling hardware, testing memory, et cetera. It will be impossible to know which of these routines, algorithms and ideas violate a patent, because every programmer would need to understand every software patent -- every software patent that's active. Software is simply too complex, composed of too many pieces which are too easy to create, to lend itself to being broken down into patent-sized chunks.

I can easily envision a world in which progress in software is totally blocked by a web of patents owned by a very few very large companies; not the best or the most creative companies but rather those with the most lawyers. In a world like that it would be completely impossible to start and build a company like mine -- and this nightmare could come to pass very quickly.

To date there has been little litigation regarding software-related patents. God help us all when that litigation does begin. Judges and juries will be asked to rule on whether a large complicated program, potentially millions of lines long, written in an obscure computer language, violates an arcane patent. The claimed violation will be built into the very fiber of the program, hidden within the program's structure and data in complex and subtle ways. One expert will say one thing, another expert will say the opposite, neither judge nor jury will be competent to understand the nature or veracity of the patent, much less which expert is closer to the truth. The patent will have been issued by an examiner who is not expert in the specific software field and might not understand the concepts essential to operating systems, fuzzy logic, or whatever the specific field is, much less the prior art. The chances of a fair and informed decision will be vanishingly small.

Software is, perhaps, more analogous to literature and music than it is to mechanical invention. It would be silly to think about patenting the first-person novel or the sonata form, yet there are software patents already that to a software engineer are just as absurd.

As a software company CEO, I am perfectly content to compete based on the quality of the software we create and the support we provide for it. I am fully satisfied with copyright, contract and trade-secret protection for the software we write. We have begun to work on some patent applications because I think we may need them for defensive purposes, but I would far rather we didn't need to do so.

If software patents become prevalent, it will seriously interfere with our ability to continue improving our products and our ability to continue developing new ones. It will also interfere with our ability to provide openness and compatibility to our customers -- a key part of the value we

provide to them.

The best possible result of these hearings for us, for our customers and for society would be for software patents to simply go away.

COMMISSIONER LEHMAN Thank you very much, Mr. Fiddler. I would just make an observation and ask a question.

You indicate where would we be if we had had patents on the spreadsheets, and so on and so forth, and I think that suggests that I think it's one of the reasons why we may not have patents on spreadsheets and the idea of a word processing program, and so on, is because those particular items were not patentable, they didn't meet the test of patentability.

And I think herein lies a lot of the problem when you say software shouldn't be patentable. Well, it well may be that there's a lot of confusion as to where that threshold is drawn, and that indeed some software-related inventions could and, you know, are very appropriately patentable, but there seems to be a lot of confusion about where the test, where the threshold, what kind of innovation meets the test of novelty and unobviousness, where that's drawn.

And how would you feel about a more vigorous examination of where that line of nonobviousness is drawn?

MR. FIDDLER: You know, obviously, to the extent patents exist, I'd like them to be as narrow and as well-defined as possible. Clearly, that's in everybody's benefit.

But I think that, yes, it's true that probably the concept of a word processor is not a patentable concept, but there certainly are key components of those that very well may have passed patent law, particularly as patents seem to be being issued, you know, very recently.

There are, I think, that even if the Patent Office is perfect, even if it issues only patents that are entirely appropriate, are entirely correct, are novel and nonobviousness, and so forth, which is I think a very unlikely place to get to, but even if we can assume that the PTO is perfect in those respects, I still think that it will have -- it makes it far more difficult to create software.

If I sit here, I mean you can set for me a problem and say please write a program that does something, and, depending on the problem, in somewhere between five minutes and a couple hours I may be able to do that. Is what I have done patentable? Maybe. Maybe there's an idea in there that is, maybe there isn't.

To find out if there is, it will take me far longer to find that out, and there's no way in the world I can be familiar with it and it will be very difficult for me to find it. It may be in a field far away from the one in which I'm working. It will multiply my work not by 1O or 2O or 5O percent, but potentially by thousands of percent.

COMMISSIONER LEHMAN How is that any different, really, from an engineer that's working in electronic components of aircraft in --

MR. FIDDLER: I think it's different --

COMMISSIONER LEHMAN -- in Seattle where there's obviously a lot of innovation and they're constantly asked to design all kinds of gizmos and do things and yet that's an area clearly where there's been patentability for a long period of time and they aren't, you know, suggesting that somehow or other engineers can't make a move and put pen to paper or turn on their workstation without consulting the legal department?

MR. FIDDLER: I think it's different in a couple ways. For one thing, copyright doesn't work for them and it works fine for us. For another thing, it's far easier to create software ideas and to make them work.

When I start and write a program, I may write however many lines of code it is, I may start with a design and do that, and I may actually have it debugged within a very few minutes. I can make changes to it by saying, "Change this line of code." I can make it work in a very few minutes. That's very different than a hardware concept or building something in hardware, where the turnaround time is much longer, the number of concepts probably embodied -- certainly the number of novel concepts embodied in any specific project are probably much smaller.

As I said, a very small number of programmers, two or three or five programmers, can certainly write a million-line program with many thousands of ideas that may potentially be patentable. Have they been patented? Have they not been? Is there prior art? Isn't there? It's almost a question of luck and almost impossible to find out and it will make it extremely difficult to work to create these kinds of programs.

COMMISSIONER LEHMAN Thank you very much.

MR. FIDDLER: Thank you.

COMMISSIONER LEHMAN Gerry? I wanted to have Gerry Goldberg stand up, the Director of Group 23O. He's an important person for all of you to know. He'll probably be back here, I would guess, following up on some of the aspects that will come out of these hearings to try to improve our procedures.

So Gerry is our point man on software. I hope you all get to know him, if you don't already.

Next I'd like to ask Jim Warren of Autodesk to step forward.

Oh, I think we think you're the person who got the Internet legislation for the California legislation passed --

MR. WARREN: That's correct.

COMMISSIONER LEHMAN -- passed in three weeks?

MR. WARREN: It's been online.

COMMISSIONER LEHMAN We've got a couple of job openings at the Patent and Trademark Office, so I mentioned that yesterday, maybe we ought to --

(Laughter)

MR. WARREN: We're going after the campaign disclosure information now.

--oOo--

JIM WARREN

AUTODESK, INC.

MR. WARREN: Mr. Chairman and other distinguished representatives of the Department of Commerce:

My name is Jim Warren.

First, I am a Member of the Board of Directors of Autodesk, a multi-national software company specializing in computer-aided design. As a 4OO million dollar company,

we have been recently identified as the sixth largest PC software publisher in the world. I am presenting its recommendations.

Secondly, I have been a computer professional since 1968, have founded multi-million dollar companies in Silicon Valley, and have held numerous leadership roles in personal computing essentially since its inception in the 197Os, in the mid 197Os.

I was founding President of the Microcomputer Industry Trade Association, received the Electronic Frontier Foundation's first Pioneer Award, hold graduate Degrees in Computer Engineering, Medical Information Science, Mathematics and Statistics.

I was founding Editor of microcomputing's first software periodical, was founder of the first, first free newspaper and the first subscription newspaper, InfoWorld, and founding host of television's oldest Computer Weekly, as well as founding the world's largest public microcomputer conventions and chairing them in the first decade of the industry.

My remarks are excerpted from three parts of my prepared statement; namely, principles, pragmatics and some specific recommendations. I am not speaking as an intellectual-property attorney. I am speaking as a technological innovator with proven experience and as a long-time observer of this industry. I've written approximately 6O to 7O articles about the future of this industry that have received wide circulation, in excess of 22O,OOO copies per issue.

We all know that software is somehow different from all traditional inventions. The difference -- but how does it differ from the devices that are surely what the framers of the Constitution envisioned when they mandated patent protection? The difference is that all traditional inventions enhance our physical capabilities, whereas software mimics the mind and enhances our intellectual capabilities. This is what makes software different from all patentable devices and this is what justifies sui generis.

Let me define what software is for the purpose of our discussion, based on its functionality, its utility, the useful character of its art: software is what occurs between stimulus and response, with no physical incarnation other than as representations of binary logic.

The fundamental question is: Do we want to permit the monopoly possession of everything that works like logical intellectual processes? I hope not.

The mind has always been sacrosanct. The claim that intellectual processes of logical procedures that do not primarily manipulate devices, as in Diamond vs. Diehr, can be possessed and monopolized, simply extends greed and avarice much too far.

What frightens and infuriates so many of us about software patents is that they seek to monopolize our intellectual processes when their representation and performance is aided by machine.

I respectfully object to the title of these hearings, "Software-Related Inventions". The title illustrates an inappropriate and seriously-misleading bias. In fact, in more than a quarter century as a computer professional and observer and writer in this industry, I don't recall ever

hearing or reading such a phrase -- except in the context of legalistic claims for monopoly where the claimants were trying to twist the tradition of patented devices in order to monopolize the execution of intellectual processes.

To pragmatics.

There is absolutely no evidence whatsoever, not a single iota, that software patents have promoted or will promote progress. And I provide examples in my paper.

Of the thousands of programmers I have known in the last quarter century, I have never heard a single one say they didn't develop a program because they could not monopolize its functionality.

Of the thousands of programs I have known about as a multi-decade industry observer, I don't know of a single one that was innovative enough to promote progress, much less perhaps qualify for a patent as a useful art, that couldn't find funding.

The system was not broken when there were no software patents.

Now, however, there is growing evidence that software patents have begun to harm and deter progress. And I provide a number of examples, including the company for which I am speaking, Autodesk, holds some number of software patents and has applied for others, which, of course, remain secret under current US law. However, all are defensive and an infuriating waste of our technical talent and financial resources made necessary only by the lawyers' invention of software patents.

Autodesk has faced at least 17 baseless patent claims made against it in recent years and has spent over a million dollars defending itself, with millions more certain to pour down the bottomless patent pit. Fortunately, we have the financial and technical resources to rebuff such claims. We rebutted all but one of the claims even before the patent holders could file frivolous lawsuits and will litigate the remaining claim to conclusion.

Your Office has issued at least 16 patents that we have successfully rebutted and we never paid a penny in these attempted extortions that your Office assisted, but it is an enormous waste of resources that could better be invested in useful innovation.

COMMISSIONER LEHMAN Could I ask a question about that?

MR. WARREN: Out of your time or my time?

COMMISSIONER LEHMAN It can be out of your time -- out of my time.

MR. WARREN: That's what I was -- oh, okay, thank you.

COMMISSIONER LEHMAN We have a procedure for re-examination of patents. It sounds to me like what happened here --

MR. WARREN: I was about to recommend that.

COMMISSIONER LEHMAN Well, we have that now, you know. In other words, were those 16 --

It sounds to me like what happened here is that people basically threatened you with lawsuits and, you know, you got your lawyers all geared up and basically scared them away before you went to court, but left it there. Whereas, one of the things that you could have done, under our

existing procedures, is that you could have come into the Patent and Trademark Office and petitioned for re-examination of those patents and have them held invalid.

Did you consider doing that, and, if you didn't, why?

MR. WARREN: I am certain that we did the least expensive thing that we could do.

And I have no specifics. You'll have to talk to our legal eagles on that, or have to ask our legal folks on that. But this is an enormous --

Incidentally we not only invested our financial resources, we invested our technical talent. Instead of them creating something, they had to go research prior art to fight off these frivolous claims. That ain't right.

Back to my prepared remarks.

That does not reward innovation nor promote progress. Furthermore, software patents can probably deter progress, and I provide a number of examples.

Finally, there is an intense danger that software patents pose to our industry's global competitiveness, and I detail how.

To specific recommendations. Okay, this is the goodies.

Let us agree that those who hold software patents probably prefer patent protection -- IBM, I think, is the largest holder and MicroSoft is the second largest -- and those who spend their time and resources creating technical innovation and national progress rather than creating patent applications and litigation probably prefer unfettered freedom to innovate.

Let us also agree that the Constitutional intent -- very important -- is to "promote progress". So let us disregard who wants what for self-benefit and act on principle. We propose as a principle that those processes that are exclusively intellectual and exclusively algorithmic, even when mimicked by machine, must not be monopolized.

We offer two recommendations, the second having 12 parts, so to speak, the 12 Apostles of redress of the current problems.

The first recommendation: Issue a finding that software, as I have defined it, implements intellectual processes that have no substantive physical incarnation, processes that are exclusively analytical, intellectual, logical and algorithmic in nature; plus the clearly stated Constitutional intent to declare that -- and use those findings to declare that the Patent Office acted in error when it granted software patents; declare that software patents monopolize intellectual and algorithmic processes and also fail to fulfill the Constitutional mandate to promote progress; declare that software as a mimic of the mind cannot be patented.

Second, until and only until software patents are definitively prohibited, reject or freeze all such applications pending conclusive action on the following 12 points:

(1) Redress serious errors of previous administrations.

Issue a finding that there have been extensive and serious errors of judgment in a large percentage of software patents granted in the past and immediately recall all software patents for re-review and possible revocation.

Encourage industry assistance. And I offer some comments about how and some legislation that's needed.

Make the information available via the Internet and solicit maximum public input.

(2) Mandate disclosure upon filing.

Issue a finding that it unconstitutionally suppresses progress to hide software threats in secret filings for one to five years. Note that most of the other high-tech nations with which we compete require disclosure upon filing or very soon thereafter.

Require disclosure upon filing or at least within, say, 9O days of filing. This will give software developers essential early warning of possible danger. It will also allow them to provide badly needed prior art, perhaps years before the patent might be granted and become a threat.

Let it be the responsibility of those seeking lengthy monopolies to defend the truly novel and truly non-obvious character of their innovations in a public patent-application review process. Do not continue to force that responsibility onto all other practitioners after the fact.

(3) Recommendation 3. Require disclosure of complete source code and documentation upon filing.

That will slow this stuff down.

Reiterate that the -- (Laughter).

That was not in my prepared remarks.

Reiterate that the major function of the patent system is to assure complete public disclosure of innovation in order that all may benefit and progress be promoted.

Issue a finding that software patents require full disclosure of complete original source code and complete internal documentation. Then require its disclosure, preferably upon filing or perhaps 9O days later, but at least upon the granting of the software patent. Note that this implements the "best mode" requirement.

Software patent disclosures in the past have often failed to fulfill this minimum requirement; therefore, require such disclosures from all present software patent holders. Those who decline to so disclose in a timely manner must have their patents invalidated as being improperly granted.

(4) Prohibit filings after any public exposure.

Issue a finding that most of the nation's high-tech competitors prohibit patent filings after any public exposure of their proposed innovation.

Further, find that patentable innovation in software is unclear, vaporware is rampant, early disclosure is common, sharing of disclosed innovation is almost universal, and possibly infringing development using such disclosures is almost inescapable. Use that finding to prohibit any filing after the date of any public exposure.

Recommendation (5) Reduce requirements for challenging software patents.

Find that the evaluation of what constitutes new, novel and un-obvious innovation in software is highly subjective and essentially impossible for the Patent Office to judge, since the Office does not have the 5O years of prior art that exists.

Change the standard for invalidating software patents from a requirement for "clear and convincing evidence" to no presumption of validity at all -- which is usually the case if the experience of well-funded defendants who can do the adequate research, such as Autodesk, is any measure.

(6) Reduce the protection period.

Issue a finding that 17-year software protection patents are clearly unreasonable where, in an industry where significant innovation can often be created in months, most innovation has minimal costs relative to traditional inventions, manufacturing and distribution is trivial, products can be shipped within weeks of being finalized, great profits can be attained in less than a year, the life of a product typically is only a few years, and all of the growth of the industry, from inception to Diamond vs. Diehr in 1981, was barely three times the 17-year monopoly period.

Shorten the one-time protection period to no more than, say, two years. Sui generis is justified.

(7) Replace -- wow, I'm still in the green -- or no I'm not -- or have I run out of time?

COMMISSIONER LEHMAN   I was giving you your maximum.

MR. WARREN: Oh, sorry about that. May I finish the other remarks very quickly?

(7) Replace first-to-invent with first-to-file.

Issue a finding that this nation is almost alone in granting monopolies on the basis of first-to-invent. If the patent system is justified and public disclosure has merit, then encourage it by awarding monopolies only on the basis of first-to-file-and-disclose, but, of course, retain the principle that prior art always invalidates a patent.

(8) Declare that useful intellectual communications cannot be monopolized.

This is the look and feel issue. We don't want to protect it under patent any more than we want to protect it under copyright, when they are not primarily aesthetic and not primarily artistic and not primarily for controlling equipment. And I address that more properly.

(9) To promote continuing progress, mandate cross licensing.

If you are going to grant monopolies over our algorithmic processes, then at least mandate that we can use them under license from the monopolists. And I suggest how.

In particular, we suggest mandatory licensing rates not exceeding, say, 5 percent of a licensee's profits prorated across all cross licensers for a given product.

(1O) Provide a nationally accessible prior-art collection.

I'm sure you heard that from 5O other people. If you don't have the resources to do it -- and make it available across the Internet -- if you don't have the resources to do it, then inform Congress that you are unable to perform your assigned functions without endangering national progress.

(11) Exercise much greater due diligence with regard to software patents.

You must stop leaving it up to endless threats, defenses, court battles among those who can afford them to ascertain which few patents might be valid, which is too often determined only by the relative wealth of the combatants.

(12) And finally, create a large public advisory body, a commission of volunteers who are technologists, those who produce the nation's progress in this area, not just intellectual-property attorneys.

Seek them from a broad spectrum of software publishers, great and small producers, including individuals.

These recommendations require Congressional action, and this industry has been politically asleep, but continuing software patent debacles are beginning to awaken it, most especially its innovators, and we certainly have the financial resources, the communication tools and the tenacity to seek effective redress as we finally organize and choose to act.

However, the needed Congressional action can be greatly facilitated by supportive recommendations from your Office. Please draft them soon. But not cloistered inside the Washington Beltway, rather with extensive Internet circulation of all drafts and discussion.

Let us stand on each others' shoulders rather than on each others' toes.

Thank you.

COMMISSIONER LEHMAN   Thanks very much, Mr. Warren. We gave you a few extra minutes there --

MR. WARREN: I appreciate that.

COMMISSIONER LEHMAN -- because of my intervention.

I want to thank you for coming out here. I think we'll look at your recommendations very carefully and I think with regard to this idea of, first of all, I hope you will appreciate the fact that we're not inside the Beltway right now --

MR. WARREN: Every two days outside we appreciate.

COMMISSIONER LEHMAN   -- we have a little capacity to innovate, even Washington lawyers can come up with a few good ideas every once in awhile.

And secondly, I think that we do need to have closer, a better means for communicating directly with the innovative community and not just for patent lawyers, and so we need to do a little innovative work to figure out the mechanisms for doing that ourselves and I really appreciate your comments. Thanks.

MR. WARREN: Ask us for help -- I mean all of the industry -- and we will help.

Thank you.

COMMISSIONER LEHMAN Next I'd like to ask Mr. Michael Glenn, from the Intellectual Property Section of the State Bar of California, to step forward. Maybe he can defend the lawyers.

--oOo--

MARY O'HARE

CHAIR, EXECUTIVE COMMITTEE

INTELLECTUAL PROPERTY SECTION, STATE BAR OF CALIFORNIA

MS. O'HARE: I am not Michael Glenn. He felt as though he needed some company up here.

Assistant Secretary and Commissioner Lehman, my name is Mary O'Hare. I am the Chair and am speaking on behalf of the Executive Committee of the Intellectual Property Section of the State Bar of California.

The Section is voluntary, comprised of more than 37OO attorneys practicing in the various intellectual property fields of copyrights, trademarks, trade secrets and patents. Our members represent, for the context of this hearing,

individuals, non-profit organizations, small and large businesses.

We are proud that our organization was one of the first to have Commissioner Lehman as its keynote speaker and we thank you for holding these hearings in California.

All too often in the past, as Commissioner Lehman has noted, California, and sometimes the needs of its attorneys and clients, more than 26OO miles away from the Patent Office, have been felt to be out of sight and out of mind. Nonetheless, California has been the center of the United States cultural and technical renaissance of the late 2Oth century. California's two largest industries, entertainment and technology, are also the United States' two largest export engines.

We in the Section hope that these hearings will signal the Office's willingness to have easy, open access to the Patent Office for Californians, a privilege until recently primarily enjoyed by the Washington, DC, patent bar.

While I am Chair of this Section, my intellectual property expertise has been gained in the context of a motion picture entertainment practice. Commissioner Lehman, we know you have a sense of humor, we know you know that we've been through an earthquake recently, but the tragic earthquake in Southern California may have rattled our homes, our offices and our psyche, but let me assure you that Californians are tough, we are not so rattled as to ignore the importance of your presence here and that's why we are here or to presume to have a motion picture attorney address you on matters at the Patent Office.

Therefore, I am privileged to present Michael Glenn, an officer of our Section, who is a patent attorney in the Silicon Valley who has represented both individual inventors and large corporations before the Patent Office for the past 14 years.

His qualifications are set forth in our written statement and he will present the statement of the Section.

Thank you.

COMMISSIONER LEHMAN Thank you.

--oOo--

MICHAEL GLENN, ESQ.

MR. GLENN: Commissioner Lehman, today's hearings have been convened to receive comments from the public on patent protection for software inventions. Rather than respond to the specific questions raised in the Notice of these hearings, we will address the important, broader issues that form the context in which the issue of patent protection for software inventions arises.

These issues include: (1) the expertise and ability of the Examining Corps, especially with regard to the difficult task of applying complex legal principles to emerging and sophisticated technologies; (2) the availability of task appropriate tools and resources to the Examining Corps; (3) the need to make Patent Office services and resources readily available to the public; and (4) the understanding that the US Constitution, in providing the Congress with "the power to promote the progress of science and useful arts by securing for a limited time to authors and inventors the exclusive right to their respective writings and discoveries", did not limit the types of discoveries for which a grant of

exclusive rights would be secured.

Preliminarily, it must be observed that patent myths abound and the Patent Office should use its best efforts to dispel these myths. These hearings are one excellent way to raise the general level of public understanding of the US patent system. However, the primary job of the Patent Office is to examine patent applications. A quality examination and precise application of the patent laws by the Patent Office are necessary to assure that the interests of both the public and the inventor are properly served.

First, while recent efforts to improve the quality of the Patent Office services, especially the quality of the Examining Corps and as a result the quality of patent examination and patents issued by the Patent Office have not gone unnoticed, more needs to be done.

Because the process of examining a patent application necessarily demands both a high level of technical expertise and a thorough understanding of the legal standards that are applied during the examination, the Patent Office must continue to attract and retain Examiners who not only have the technical knowledge necessary to understand the invention, but who also understand the legal framework within which the Patent Office functions. To this end, ability and merit should be the most important standards by which Examiners are hired, promoted and retained.

Secondly, we encourage the Patent Office to do more with regard to improving the quality of the patent examination process.

For example, in many technical areas a search of issued US patents alone cannot reveal the most relevant prior art. In rapidly developing technology, such as computer software and biotechnology, where the enforceability and availability of intellectual property rights in the past have been uncertain, the most relevant art may be found in industry journals and in proceedings of professional societies and institutes.

The Examining Corps should be encouraged to search all relevant information sources. Intensive training in using these information sources should be provided the Examining Corps such that the most relevant priority is applied by the Examiners to every patent application filed with the Patent Office.

Thirdly, since the Patent Office is also a tremendous depository of knowledge, we encourage the Patent Office to explore the possibility of giving the public throughout the United States free or inexpensive access to the Patent Office database through an online source such as the Internet.

At present the few public patent depositories scattered across the US are underfunded, understaffed and resource-constrained. For example, online searching is not available at the Sunnyvale patent depository here in Silicon Valley and those wishing to perform a computerized search of the now available CD-ROM database there are limited to only 2O minutes of use.

The Patent Office search room in Washington, DC, is not accessible for the public at large, attorneys and inventors who live, work and invent in California. Ready public access to such publicly-owned information would allow inventors to make informed decisions about whether or not they should pursue patent protection, would allow those seeking to

enter a new market to review the patent literature before entering upon a course of action that could lead to a wasteful, potentially disastrous patent infringement lawsuit, and would allow those seeking to license technology to have access to the marketplace of ideas contained in the Patent Office database and be better able to establish a fair value for such technology. As important, the public would become more familiar with and better educated concerning the patent system.

Fourth, from time to time an issue may arise when a recently-issued patent is publicized as part of a marketing campaign by a successful patent applicant or as part of an ideological debate concerning the applicability of patent laws to the technology protected or the breadth of coverage afforded the invention by the patent's claims. As a result a discussion ensues concerning the wisdom of extending patent protection to new and emerging technologies. We caution the Patent Office not to allow the mere existence of a public debate alone to provide a rationale for establishing separate rules for such technologies.

This discussion is not new. In the days of the Wright brothers there was the fear that the future development of aviation would be seriously impeded if Wilbur and Orville should be allowed a basic patent on their invention. As we all know, this was not the case. As Wilbur Wright put it: "When a couple of flying machine inventors fish, metaphorically speaking, in waters where hundreds had previously fished, and spending years of time and thousands of dollars finally succeed in making a catch, there are people who think it a pity that the courts should give orders that the rights of the inventors shall be respected and that those who wish to enjoy the feast shall contribute something to pay the fishers."

With regard to enforceability of patent rights for new and emerging technologies, the Patent Office must show leadership. The Statutory mandate of the Patent Office is clear: novel and unobvious inventions that comprise patentable subject matter must be granted a patent. As a general principle, patentable subject matter cannot be limited to known technologies, but, as stated by the Supreme Court in the Chakrabarty case, must also encompass "anything under the sun that is made by man." Otherwise, only old technologies will be found to comprise patentable subject matter, at which point the patent system will lose all meaning.

It is the ownership of invention that spurs innovation, not just the promise of exclusivity afforded by patent grant, but more significantly, in the incentive to avoid a patent by inventing around the patented invention.

Finally, while the patentability of software inventions has long been an interesting topic of discussion, first in the courts and the Patent Office and now in the press, much of the discussion may be caused by misunderstanding and confusion. We suggest that some of the misunderstanding stems from the confidential nature of the examination process. In many areas it is not possible to perform an infringement search to clear a new product because the most relevant patents are still pending in the Patent Office and not available to the public.

The Patent Office could explore opportunities for involving the public in the examination process to avoid any surprise attendant with the grant of broad-reaching patents. For example, the Patent Office may want to consider the pre-grant publication of patent applications and/or pre-grant public opposition hearings.

We applaud the Patent Office decision to re-examine a recently issued patent on its own initiative in light of new art discovered after issuance of the patent. As an organization we have no opinion regarding the outcome of the re-examination, we only applaud this bold and welcome policy on the part of the Patent Office to pursue excellence. The ultimate outcome of such actions will be to improve the stature and regard with which a United States patent is held. This in turn will provide more certainty concerning the validity of an issued patent. Reducing the likelihood of a successful attack on the validity of a patent should encourage early settlements of patent disputes and strengthen American industry by strengthening the incentive to innovate rather than to litigate.

In closing, the Patent Office must continue to serve the needs of a broad range of applicants, from independent inventors to multi-national corporations, while taking into account the effects of a fast-changing global economy.

Patents not only protect inventions, they also protect employment and national wealth. The United States is a technology leader because of the incentives it provides to those persons who take the effort and risk involved in bringing new inventions to the marketplace. Of all the nations in the world, the United States has the only significant software industry, the only significant biotech industry, and the only significant microprocessor industry, to name a few. These industries form a mighty technology river that has human creative energy as its source. The American experience shows us that such creative energy requires incentive. The role of the Patent Office is paramount because the Patent Office is charged by law with providing incentives for this creative energy by protecting patentable inventions.

We pledge that if you involve California's inventors and practitioners in the ongoing discussion of Patent Office procedure and policy, your job will be easier and we can together ensure that the patent system and the Patent Office fulfills the Constitutional proviso of promoting the progress of science and the useful arts, all to the economic benefit of the citizens of California and the rest of the United States.

Thank you.

COMMISSIONER LEHMAN Thank you very much.

Next I'd like to call Mr. Lippe of Synopsys.

--o0o--

PAUL LIPPE

GENERAL COUNSEL

SYNOPSIS

MR. LIPPE: Thank you, Mr. Chairman.

By way of introducing myself to the Panel, let me say that I've sat where you're sitting. I used to be Chairman of a thing called the Colorado Air Quality Control Commission and having sat through two days of stupifyingly dull testimony about aromatic emissions of oxygenated fuels, I respect your stamina and your willingness to sit through this stuff.

COMMISSIONER LEHMAN   Well, there's a big difference. That may have been stupifyingly dull, but this isn't. It's intensely interesting. It really is.

MR. LIPPE: So I'm going to try not to echo the comments that you've heard before, but I do want to stand in strong ratification of some of the critiques that Mr. Fiddler from Wind River made about the software patent system.

The problem is, from my perspective, the legal system --

COMMISSIONER LEHMAN   Can you tell us just a little bit about Synopsys?

MR. LIPPE: Yeah, I will.

The broad problem is that the legal culture and the legal domain is so different from the technical innovation world that when you try to bring them together, at least from the technical people's side, it doesn't work very well.

I'm General Counsel of Synopsys. Synopsys is an electronic design automation software company. I'm also the head of a little thing called "The Public Affairs Committee of EDAC". EDAC is our industry trade association. It stands for EDA Companies. There are about 4O companies in EDAC, ranging from very raw startups to some half a billion dollar companies. EDA is probably one of the two or three principal domains within what your Notice refers to as "computer integrated design".

It is a strategically critical-technology area for the United States, and Synopsys is, in the new parlance, clearly a national technology champion. We make software which is used in the design of complex electronics parts and our customers are in the semiconductor computer systems and telecommunications industries. People such as Sun, Hitachi, IBM, Intel, Siemens.

Synopsys itself was founded about seven years ago, and, in the term of art, industry analysts expect that we'll do around 2OO million dollars of revenue this year. We are probably the second fastest growing company in the computer-aided integrated design sector, the fastest growing company in EDA, and we are considered to be one of the hot companies in our field.

The reason I'm speaking today is I want to challenge what I think has been the animating idea behind the move towards enhanced intellectual property protection and patent protection and that is that enhanced intellectual property protection is per se beneficial for US companies.

And my challenge comes not as an intellectual property lawyer, although I am a lawyer, and not as a technologist, because I'm not a technologist, but as somebody with some deep experience in the political sector who's given some thought to what mix of policies makes the most sense to advance America's industrial interests.

And as somebody taking a political approach, I think it's important, when you examine these policies, to think of -- to focus on the outcomes and who wins and who loses and not so much on the product, as well as what the ideas are that are advanced by the various speakers.

The concern that I've got, and I think the gentleman from Wind River and other people have, is that the startup process and the innovation process is inherently fragile, and, as the domain becomes increasingly littered with patents, to have the ability to kill companies at each stage of the process. There are various what you might call choke points, at the financing stage, at the various financing stages, and at the stage of trying to begin to sell to customers, and it's all too easy for innovative companies to be blocked from bringing their products to market. And I want to talk about that a little more.

The fundamental assumption that enhanced protection for patents is favorable to US industry is an idea that I think gained currency in the late '7Os and early '8Os and it was based on the notion, the basic idea -- and I hope I don't offend anyone by saying this -- that Americans invent and Japanese copy, and the way to make America stronger is to help to enhance intellectual property protection. My fear is that we've gone too far, that we've moved towards more aggressive patent enforcement, at the same time we've moved towards less aggressive anti-trust enforcement, and that the remedy, the inherent remedy for patent of monopoly protection and the nature of patents being issued is not -- we've gone too far.

And the other thing to focus on in terms of the software industry is that software, as the gentleman from Microsoft used to say but won't say today, is a natural monopoly. Being first to market confers an enormous advantage in terms of the ability to set the standard, there are high barriers to entry, high fixed costs and low variable costs, so you've already got a huge head start if you're first to market.

It's not clear to me that there's, as some of the earlier speakers have said and I agree with, that you're really furthering the goal to encourage people to innovate by conferring additional monopoly.

And there tends not to be a lot of success in the software industry for copiers, clones, and followers. I think you'd be -- there are very few examples of people who followed, who have executed a following strategy copying other people's technology, that have been successful in software.

Some of the ideas that underlie increased protection for patents, I think, are misconceptions, at least in the domain where we live.

First, the key idea that I think is wrong is the notion that invention per se is what's important. If you go to a venture capitalist in Silicon Valley and you say "I've invented something", they've got zero interest in that because they recognize that the whole Silicon Valley paradigm is based on the notion that what matters is customer-delivered innovation, which is very different from the level of invention that you need to get a patent, and that's why today the perception of the people in this room who are on the anti-patent side is that most patents are going to big companies who don't sell the products, they get the patents out of their industrial labs and then this group of people that you might call the lone inventors.

But what really creates value for the United States and for the customers is when you deliver the technology to customers in a way they can be used and that, that has not been the focus of the patent law, for good and sufficient reasons, in the past.

The second thing that I think, at least in our domain, that is a misconception is that people actually read patents and use them to advance the wrong technology. No engineer I've ever known has been willing to read other people's patents,

and most people feel, at least in our field, that patents don't describe things with enough particularity to know how to copy them anyway.

The third problem is the patents, as you've heard over and over again, I won't belabor the point, have been very incremental, they haven't been significant, and so there's so much overlap space between the existing patents.

And the fourth misconception and I think the most important one is that the patent system protects small companies. As I said earlier, the patent process is fundamental in the legal process, a lot of lawyer bashing goes on, some of it justified, much of it not, but in any case recognize that the process of delivering innovation to the customer is a totally different culture, it's a totally different process, than that required to obtain and enforce a patent.

That doesn't mean that obtaining and enforcing patents is a bad thing, but it's always going to be a diversion of energy and resources, as Mr. Warren said, from that process of delivering innovation to customers, and the litigation process is almost always going to favor the bigger guy because he's got the resources and he's acculturated to going through that kind of drill. Small companies hate it. Engineers, most engineers I've ever known hate it, and they're very uncomfortable and they're very vulnerable to this kind of process.

The other point is that the legal system doesn't really comprehend the technology. We happen to be the leaders in our field, we're glad of that, but the consequence of that is that, of the ten people in the world who understand what we do, eight of them work for us, none of them work for the Patent Office, and it's very unlikely that anybody who's got that kind of leading edge expertise would want to work for the Patent Office, no disrespect to the folks in the Patent Office, but they would like to be building the products and, you know, doing the things that people around here do.

So there's an enormous amount of randomness in the system because the legal system cannot adequately -- and cannot be expected to -- adequately comprehend the technology at the level that our folks do. That randomness, then, introduces enormous transfer costs and friction costs because it doesn't really afford, the current system, doesn't really afford us a lot of inexpensive ways to resolve the issue.

There's an article in "Electronic Engineering Times" which talks about patents in the EDA industry. The EDA industry is probably the most, maybe along with desktop software publishing, American-dominated industry, 99 percent of worldwide revenues from American companies, and it is absolutely a strategic technology industry, central to everything happening in electronics today, but the people who hold the patents by and large are Japanese companies, with the exception of IBM which is the largest patentholder. Well, these Japanese companies happen to be our customers, they're not our competitors because they don't sell any products, but it is a little worrisome that Hitachi's got 49 patents in this area and they don't sell anything and we've got zero patents in this area. So if we were an earlier-stage company, it would be even more worrisome because the ability of the large company to block the small company creates a lot of uncertainty.

And I was always taught and always believed that in the law

predictability has got to be one of the principal goals of any well-conceived legal system and right now people feel like there's very little predictability in the system, instead there's a lot of randomness.

The other thing that's happening in terms of where the world is going in our domain, and I don't know how to deal with this one, to tell you the truth, there's what I call "hardware/software convergence". We're able to represent in software things that were formerly only represented in hardware and so we now have sort of a confluence of the most patent-oriented domain, which is electronic parts, and the least patent-oriented domain, which is software, and it's very confusing.

We've also got the reality that the traditional US patent holders, in particular IBM and AT&T, are no longer as constrained as they have been historically about their anti-trust worries and have been aggressively going after people.

So, you know, one of the anomalies is that there's a very significant technology called RISC, Reduce Instructions Set Computing, it was invented by IBM in 1975 in Fishkill, and that's great but they didn't do anything with it, they left it in the closet for eight years, until Sun brought RISC to market and made a very significant technology shift and delivered a lot of value to customers through RISC. Well, IBM went after Sun and they were able to get Sun to pay them royalties on the technology, but the really important event that occurred was, not the conceptualization and creation of RISC in the lab, it was Sun creating the market and delivering the value to customers around RISC.

In terms of suggestions, you know we've all got sort of overlapping suggestions so I won't belabor the point, I think some of the suggestions that were made were really good, but I think we ought to be thinking, and I would ask you to consider, I don't know how you get there, that some kind of sales is a requirement, that there be some kind of -- I think first-to-file is not going to solve any of the problems, but some notion of first-to-deliver-value as opposed to just having an invention in the lab. Especially where we've got this three-year black hole, where somebody can file a patent and everybody else is shipping products and then three years later people find out that they've got a problem with the products, and obviously Compton's Multi-Media Patent is an example of that.

So my focus point is that intellectual property protection per se is not necessarily a good thing for America. It's good for some companies, it's not good for others.

I think on the whole the thing that we are best at, which is the smaller-company innovation, it is a very worrisome trend and a lot of companies are very concerned about it. I think it poses a significant threat to hurt the job creation and innovation and company creation machines that we've got going, and I'd like you to look for ways to rein back where we are.

COMMISSIONER LEHMAN Thank you very much.

Next I'd like to call Tim Boyle, Executive Director of Multimedia Development Group.

--oOo--

TIM BOYLE

EXECUTIVE DIRECTOR

MULTIMEDIA DEVELOPMENT GROUP

MR. BOYLE: Good morning.

My name is Tim Boyle and I represent the Multimedia Development Group. I'm the acting Executive Director of that Group.

The Multimedia Development Group is a market-development oriented trade association. It's located in San Francisco. Our members are primarily interested in the software side of this industry.

We represent about 4OO companies that build the software for multimedia titles. These include about 2OO multimedia developers and publishers, 5O technology companies, about 15O service providers, including accountants, public relations firms, marketing research firms, and over 25 law firms. We also represent 2O to 25 educational, nonprofit and governmental organizations.

Our mission is to help the emerging multimedia software companies become commercially viable by facilitating the communication between the parties who develop, fund, service, sell and, in your case, regulate these titles.

I would like to thank you for the forum and we appreciate the fact that you are soliciting our opinions.

I'd like to address three points today.

The first is the need to stimulate the creative processes in this industry and the commercial structures that support them through an equitable code of intellectual property.

Secondly, the need for this code to meet the digital challenge by distinguishing between what is a patentable invention and a copyrightable creation.

And finally, some suggestions for your consideration, such as the possibility of incorporating some of the precepts of academic science into the work of the Patent Office, in particular the concept of peer review.

I'd like to start by saying that we support patent protection for inventions that integrate software with other elements. I would also like to let you know that the furor over the Compton's New Media patent claim comes in part from our community. While we note with pride that Compton's New Media is a member of our organization and we wish to see their creations appropriately protected, the majority of our members believe that the ideas at issue in that claim are better protected by copyright rather than patent.

Compton's Multimedia Encyclopedia is a very clever and extremely innovative use of the new vocabulary of digital communications. As such, it represents a unique and creative arrangement of fundamental elements that constitute this new vocabulary of the artist and the author in the digital age.

Graphical screen elements, windows, buttons and such, search and navigation methodologies, multiple views of databases, are part and parcel of this new vocabulary. It is their use in the expression and representation of ideas that creates value, and this value, our membership believes, must be protected. We look to the Patent Office to identify, understand and protect the fundamental concepts of this new media. These concepts properly belong in the public domain because they are the alphabet, the building blocks of our new media.

The multimedia developer community has a vested interest in protecting their intellectual property from unfair copying or infringement and to ensure that the concepts on which they are based can be freely exchanged. We believe that one can only properly assess the patentability of a work after reviewing that work in the context of all the work which has gone before it.

There was a much simpler task in the industrial age and this is now much more difficult in the information age that we're moving into.

How would theatre have developed if the concept of plot were owned by someone? I mean William Shakespeare never could have afforded a license. It is the innovator and society who will suffer if we fail to protect the novel ideas or fail to recognize the obvious in this new media.

I'd like to close with a few suggestions that we have for the Office. The first is opening up the patent application process. The current process has been characterized as "secretive". We would like to see that characterization changed.

We would also suggest peer review. In academic science a discovery is accorded recognition only when it has passed the test of peer review. If you'll remember "cold fission". That was taken care of by the scientists.

We recognize that there is a problem that open review presents for inventions with great commercial potential, but there are other members of our industry who have put forward a number of proposals in that regard. Our organization endorses the general concept of peer review as an important element in the evaluation of software patent applications.

And the third and final suggestion is building a definitive library of prior art. The Multimedia Development Group has published a call for prior art and we would like to offer your Office access to any an all materials that we received. We would like to establish an ongoing relationship to ensure that your Office has access to any prior art it requires.

The Multimedia Development Group's purpose is to represent the interests of the multimedia community and to grow this industry. We would like to extend our hand to assist you in gaining access to and understanding the needs of that community. We would like to have as strong a relationship with our Patent Office as the one enjoyed by the metal fabricating inventors of the industrial age. That Patent Office created the basis for the most explosive economic growth that the world had ever seen, and this Patent Office has the opportunity to dwarf that achievement by creating the basis for a global information economy.

Thank you for your time.

COMMISSIONER LEHMAN Thank you very much.

By "peer review" do you mean that we would have like a panel of multimedia -- say we had a multimedia patent application -- we would have a panel of multimedia developers who were actual developers out there who we would convene somehow or other or we would send around the patent application to them the way in an academic setting they might send around a paper?

If you were a biochemist, you know, and you were about to

publish a paper in science, the editor sends it around to three other scientists to look at it and to make comments and corrections. Usually the author of the original paper doesn't even know it, who the people are. Is that the kind of thing that you're talking about?

MR. BOYLE: That is one way to go about it and I would want that reality factor in there.

The other method is a scholastic review, but not in the traditional, old school sense. We have a school, San Francisco State University, that is providing 5O courses and training over 9OO students a semester in multimedia. They have a group of people, many of them our members, who understand this. That school or an institute of that type, and there are many around the country, would be very happy to act as the agent to tell you what had happened in the past.

COMMISSIONER LEHMAN Well, I think part of what you're talking about there is better communication, better education of our Examining Corps, and more fluid and constant communication between the Examining Corps and the people in it with people who understand these industries.

MR. BOYLE: Right.

COMMISSIONER LEHMAN Now, for example, we could send some people from the Examining Corps out to attend this course that you just described.

I think one of the things -- and I don't want to open up, you know, an unmanageable floodgate here -- but I think these hearings are a very formalized kind of procedure, but I like to think of them as sort of the beginning of the process since it's so clear that we need to have better communication with our customers. That will solve some of our problems right there, if we just kind of have better communication with them, and that's why I think that we're going to have to.

I mentioned Mr. Goldberg was here and I think on an informal basis, we're not like a court, you know, you can only hear us in a hearing room, we can have informal contacts with industries and people and I encourage those of you who are in the room to get to know some of the Patent Office people who are here and sort of start to develop where you bring us into your peer group a little and we'll try to cooperate with that.

Obviously, we have to be doing our job, we can't be running around the country on junkets all the time, but we -- but it's important for us to develop better means of communication. Of course, electronic communication, Internet style of communication, too, is an important part of that. Anyway, I appreciate your comments and I have a little better understanding what you mean by "peer review".

MR. BOYLE: And I also would suggest that opening up the Patent Office electronically as widely as possible is going to give you access to that community. And you have many trade associations and professional development societies that would be happy, that are looking forward to working with you.

And I think everybody understands that we're on -- that this is the beginning of something new. My favorite is that the theme song of multimedia is "Something's happening here, what it is ain't exactly clear", and that is the state we're in.

Thank you.

COMMISSIONER LEHMAN Thank you.

Next I'd like to ask Mr. Ronald Laurie, Attorney at Weil, Gotshal & Manges, a prominent member of the intellectual property bar, to step forward.

--oOo--

RONALD S. LAURIE, ESQ.

WEIL, GOTSHAL & MANGES

MR. LAURIE: Thank you, Mr. Secretary.

I'd like to address the issue of patent protection for software-related inventions generally rather than the specific questions raised in the Notice.

The views expressed are personal, they're my own, and they don't necessarily represent the position of either my law firm or any particular client.

By way of introduction, I worked in Silicon Valley for 33 years, initially as a programmer and software designer and later as a patent lawyer focusing on computer technology. I currently teach a course at Stanford Law School entitled "Intellectual Property Protection for Information Technologies".

I have previously served on advisory panels to the National Research Council and the National Academy of Science in connection with software protection, and I was the only patent lawyer on the Advisory Panel to the Office of Technology Assessment in its recent study "Intellectual Property Protection for Software".

When one listens carefully to the impassioned arguments against quote "patenting software" unquote, it becomes apparent that the arguments and the basic intellectual property policy positions which underlie them can classified into three categories:

First position: patents are bad. Second position: software patents are bad. Third position: bad software patents are bad.

The first position is most often heard in the halls of academia and raises fundamental social and economic issues which go far beyond the scope of the present inquiry.

The second position is often advanced by some, though not by any measure all, of the software companies that emerged and flourished during the early and mid '8Os as a direct result of the commercial introduction of the microprocessor at a time when the industry generally and incorrectly believed that software-based inventions were unpatentable as a class.

The third position is the one most widely held today and for obvious reasons the easiest to defend.

In the heat of the debate over software patents, the boundaries between the second and third positions tend to blur, but I submit it is critically important to address them separately.

I respectfully submit that the second position is legally unsound and that the third while correct does not represent an insoluble problem. In support of my thesis, I offer the following:

Premise one: U.S. patent law does not protect software. Rather, it protects processes and machines that are quote "within the technological arts" unquote. In one of the first

cases to consider the question of whether inventions involving computer programs constituted patentable subject matter, Judge Giles Rich, who was one of the principal architects of the current patent statute, introduced the phrase, "technological arts" as the modern equivalent of the constitutional term "useful arts," and is therefore defining the outer boundaries of patentable subject matter, both under the Constitution and under Section 101 of the Patent Act. That case was, In re Musgrave, decided in 1970.

Over the more that two decades since Musgrave, Judge Rich's formulation has remained unchallenged by any subsequent decision, although unfortunately it has been ignored by many. Thus any process that is not sufficiently applied the physical environment in which it operates to qualify as being quote "within the technological arts" unquote, constitutes unpatentable subject matter.

The critical distinction then is between applied technology and abstract ideas. Examples of the latter include; laws of nature, scientific principles, methods of doing business, printed matter and unapplied mathematical relationships.

Premise two: Computer implemented solutions to technological problems in the form of processes and/or machines typically exist along a design spectrum, ranging from pure hardware, that is random logic, to pure software, that is an externally-loaded computer program running on a general purpose digital computer.

Intermediate points along the spectrum involve designs which may be described as special purpose computers and which combine elements of hardware and software in varying proportions, using random logic, array logic, such as PLAs and PALs, microcode and firmware, firmware being fixed programs stored in internal read-only memory.

The particular point along the design spectrum that represents the optimum solution to a given problem is determined by a variety of factors, such as cost, speed, size, flexibility and so on. Moreover, the optimum design point moves over time as competing implementation technologies evolve at different rates. For example, in the mid '70s, complex video game functionality was implemented entirely in random logic. After the arrival of the microprocessor, the very same functionality was realized using firmware.

Finally, technologies such as logic synthesis are becoming available, by which a software solution can be quote "translated," unquote into an equivalent hardware solution, and vice versa. It should be self evident that as a matter of legal policy, the law should not promote artificial distinctions that the technology does not recognize.

And I should point out that Mr. Lippe's company is in the business of making a product which in effect translates software into hardware. Another example which I think illustrates the point is the technology of neural nets, which was originally created as a pure hardware solution and has evolved now into a software technology.

Premise three: The fact that a particular solution can be expressed mathematically or is a series of logical operations should be irrelevant to the patentability of the solution.

In 1972, based on what many commentators believe to be an erroneous interpretation of its prior decisions involving laws of nature and scientific principles, the U.S. Supreme Court announced, in Benson v. Gottschalk that a patent claim describing a process which, quote, wholly preempts a mathematical algorithm is nonstatutory; that is, does not define patentable subject matter under Section 101 of the Patent Act.

The result of this formulation has been over two decades of confusion and inconsistency in the case law involving the patentability of software-implemented processes. The fact is that mathematics is a language, albeit a very precise one, and like other languages can be used to describe concepts and relationships that are technologically applied as well as those of a more abstract nature that are not so applied.

As noted by Professor Chisholm in an article called The Patentability of Algorithms, the real issue is probably not one of subject matter under Section 101, but rather one of indefinite claiming of the invention under Section 112. Under the constitutional standard within the technological arts, it is the subject matter of the invention and not the language chosen to describe it that should determine the presence or absence of patentable subject matter.

Premise Four: Even if a particular software equipment and solution represents patentable subject matter, in order to justify the exclusionary benefits conferred by a patent, it must also pass the test of novelty and nonobviousness over the prior arts.

And Commissioner, you have pointed this out to several of the speakers, that there is significant difference between the patentability of software as a class and the patentability of any particular software invention.

This is the key factor that interrelates the second and third positions, i.e., software patents are bad versus bad software patents are bad; that is, even if a software implemented solution is sufficiently technologically applied to pass muster under the statutory subject matter test, in order to quality for patent protection, the solution must also be novel and nonobvious to a person of ordinary skill in the art.

It is submitted that given the objective to be accomplished and accepted principles of software design, the great majority of the software written today would not pass the nonobviousness test. Thus, the effectiveness of a patent system in a particular area of technology is directly related to the degree to which the examining authority -- in this case the Patent & Trademark Office -- has access to the most relevant prior art. To the extent that there are or can be created mechanisms through which the Patent & Trademark Office can access the widest body of software-related prior art, the system will work.

A number of such mechanisms have been discussed during these hearings, and they include PTO access to the growing number of commercial and public databases of software technology, private sector assistance in supplementing the PTO internal database, early publication of patent applications coupled with third party submission of prior art.

The important point is that the problem of bad software patents is mechanical and not inherent. That is, over time it can be engineered away or at least reduced to a commercially tolerable error rate.

Finally, Premise Five: A very heavy burden of persuasion should be placed on anyone who advocates that a particular kind of technology should be exempted from the normal operation of the patent system.

In 1980, in the Chakrabarty case, the U.S. Supreme Court interpreted the patent copyright clause of the U.S. Constitution to require that the scope of patentable subject matter should be as broad as possible -- anything under the sun that is made by man.

Those who maintain that software based invention should be excluded as a class from patent protection argue that software is different. It's different, they argue, in terms of its essential character -- it's logical. It's different in terms of the creative process by which it comes into being -- it's authored rather than engineered. Or it's different in terms of the underlying economic model governing its production, distribution and life cycle.

These differences have been discussed and debated at gatherings of distinguished software developers, computer scientists, economists and legal scholars and practitioners under the sponsorship of a number of governmental agencies, including the National Research Council, the National Academy of Science, the Office of Technology Assessment, the U.S. Congress and the U.S. Patent & Trademark Office.

Despite the fact that the positions on both sides have been eloquently expressed, the results are inconclusive. The primary reason is that there is no hard data available to support the anti-software patent position, and the evidence is anecdotal at best. Clearly, software is different, but is it different enough from all other technologies to justify a special exemption from the normal operation of the patent laws.

Given the unavailability for reliable data on the societal costs and benefits of patenting software-implemented technology, we are presented with a situation where important policy decisions must be based on fundamental legal principles. In such a setting, we must conclude that those who would withhold patent protection from technologically-applied processes and machines, that happen to be implemented partially or wholly in software, have failed to satisfy the burden that the Constitution, the Supreme Court, and sound legal policy have placed upon it.

Thank you.

COMMISSIONER LEHMAN Thank you very much for an excellent statement, Mr. Laurie.

There was something I was going to ask, and I may have to follow up now on it because it slipped my mind. But I think that was a good description of -- you parsed out the problem very well.

MR. LAURIE: If I could address a point that came up yesterday, relating to the role of competition in intellectual property law, and where the competition is more appropriately addressed under the anti-trust laws or under the intellectual property laws, I'd like to say that I think that there is, there are many places in intellectual property law where competition plays a role, and the patent law of misuse is an example, and as shown by the Seiko v. (Accolade) case in the Ninth Circuit, in the copyright law under fair use, competition plays a very important role.

Thank you.

COMMISSIONER LEHMAN Thank you.

Next I'd like to ask Lee Patch, the Deputy General Counsel of Sun Microsystems to come forward.

--oOo--

LEE PATCH

SUN MICROSYSTEMS

MR. PATCH: Mr. Commissioner and colleagues, my name is Lee Patch. I speak today on behalf of Sun Microsystems, a 12-year-old $4 billion Silicon Valley-based manufacturer of computer workstations and related software products.

Sun invests approximately one-half of its substantial R&D budget in software development, particularly UNIX-based operating systems, development tools and application programs.

I serve as Deputy General Counsel and Chief Intellectual Property Counsel at Sun. I have the responsibility there for the patent activities.

You've heard and will no doubt continue to hear today widely diverging opinions concerning the virtues, or on the other hand, great evils of software patents, and you will note no lack of emotion and commitment to the speakers on either side of the issue. It's quite remarkable, I believe, that the normally quiet, calm environment of the patent practice and of the software development community has been so disrupted in recent years by loud and impassioned philosophical debate on subject matter that historically and traditionally had been only of interest to esoteric patent practitioners.

We've seen luminaries such as Mr. Warren take time out of his busy schedule to speak to you very passionately about evils of software. What I'd like you to take with you from these hearings, if nothing else, is that from the perspective of a company like Sun Microsystems, the system is indeed broken and needs addressing. The current operation of the system is creating an unacceptable amount of uncertainty within the software and computer industries, and as I'm sure you will appreciate, business executives who routinely or daily make million dollar, multimillion dollar gambles on issues of technology or on issues of the marketplace, hate the need to take gambles and to make bets upon the outcome of a legal system.

In the face of this problem with the system, you have heard radically different proposals for solution, ranging from abolition of software patents outright to rather modest suggestions of improvement in the searching capabilities that exist within the Patent & Trademark Office.

I would like to summarize if I could what I believe to be the three fundamental problems that have been the subject of much testimony before you this week. The first problem that I believe has been identified and addressed significantly relates to the quality problem within the Patent & Trademark Office. The second problem that I would like to address, which has been discussed previously, relates to the surprise problem, that is of considerable concern to the software industry. And finally, I'd like to address briefly the subject of the tied-hands problem, which is another aspect of the frustration you may be hearing this week.

To the subject of the quality problem, I would say in my experience that it is indeed the case that low quality software patents are routinely being issued -- I should also mention that none of those are being issued to my

company -- but they are indeed being issued. Largely it appears, due to the fact that prior art is not available or being overlooked or being misunderstood, and secondly, because it's a widely-held view that the nonobviousness standard that is being applied is simply too low a threshold.

COMMISSIONER LEHMAN I hate to interrupt your train of thought, but what is the -- I should probably ask some other people this, but -- what is the relationship of the Court of Appeals or the Federal Circuit to that? Do you think they're not giving us the right kind of guidance that they should give us on the obviousness standard here? Or is this our problem more?

MR. PATCH: I believe it's a merged problem. I believe that the Court of Appeals for the Federal Circuit has lowered that standard, and unduly so, and I also believe that the standard is by no means uniformly being applied within the Patent & Trademark Office.

In my role as corporate patent counsel, in-house counsel, I spend considerable time defending against infringement charges, and I have had personal opportunity to confront most of the famous bad software patents that you will and have here heard about, including such as the exclusive or patent, and the Soderblom patents and AT&T Pike patents, and other like the Mark Williams byte order patent. There's a list of them which have been widely disseminated as problem patents. I have personal experience in having rebutted and defended against charges of infringement with relation to many of those. From that experience, I would like to offer a few observations for your benefit:

First, there is indeed a quality problem that exists, and it is costing the industry a great deal in terms of lost cycle time, a lot of expensive effort being undertaken that would ideally not be necessary.

That having been said, I'd also like comment that no, it is not the case that in the software industry the sky is falling. We have not reached a stage where these types of problems are bringing to a screeching halt progress in the industry. Our confrontations with this group of questionable patents were fortunately all resolved prior to litigation, and where there was some payment, they were relatively inconsequential amounts.

A third observation I'd like to mention is that to address these kinds of infringement charges, which are perceived to have little merit, it's often quite necessary in my position to do some mining in the memories of a series of old and experienced practitioners of this art, who I happen to have at my disposal inside the corporate environment.

There is not a great deal of public documentation, patent or otherwise, that serves as a ready vehicle or mechanism for solving these problems when they arise. A lot of dusty basements have to be explored and old computers that haven't been powered up for many years have to be discovered and reactivated in order to deal with these problems as they exist today.

The last observation I wanted to make is that the quality problem that I believe to exist is not unique to the software industry. In my business questionable patents are being enforced not just with respect to software inventions but in the hardware arena and the semiconductor arena, there is a considerable amount of that going on, driven not so much by

the nature of the technology but by the nature of the economic interests and an opportunity being presented to people to realize some very significant money to the bottom line.

The second issue I wanted to mention briefly was, as I said, the surprise problem. And that's a very real source of frustration as you've heard at length this week. The industry feels as though it's being kept in the dark for long periods of time, and then surprised by some unanticipated patent jeopardies. This is clearly the result of delayed publication on the one hand, coupled with the ability in the Patent Office for an applicant to prolong the prosecution for an inordinate period of time, with no penalty, without loss of legal rights. This creates an environment where surprises become the rule rather than the exception to the industry.

Finally, the third issue of the tied-hands problem has also been mentioned, and I'd like to simply summarize my experience as an in-house counsel as it relates to that. When a low-quality patent is perceived to have issued in the software industry, most feel powerless to do anything about it. There exists today no quick, cost-effective mechanism to remedy this situation. The court system is typically unavailable unless you've already been accused and would be able to initiate a declaratory judgment action. And even if it was available, it's viewed as slow, expensive and lacking expertise.

The current Patent Office re-examination procedure, which has been discussed, is frankly considered in the industry as a trap to the unwary, and it is consciously avoided in most cases of the type that I mentioned. It's viewed as biased in favor of the patent applicant, and it's also viewed as dangerous, as a spoiler of otherwise powerful prior art. Frankly, the best prior art that you know about you would never offer up into the current patent re-examination procedure. You hold that back quite consciously.

COMMISSIONER LEHMAN Why is that? Because you don't want to disclose it because it's trade --

MR. PATCH: No. If you have a good reference, your very best reference, you don't wish to throw it over the fence to the Patent & Trademark Office and --

COMMISSIONER LEHMAN You want to save that for litigation.

MR. PATCH: You wish to have an opportunity to advocate aggressively the significance of that reference, and the Patent & Trademark Office re-examination proceeding simply does not provide that, and as a result of that lacking, good references are spoiled and no longer of significant use to you later in litigation.

COMMISSIONER LEHMAN And is that why a lot of people just choose to forego the re-examination process and go into litigation then directly?

MR. PATCH: Absolutely, absolutely.

A couple suggestions as to how you might address some of these problems: The quality problem unfortunately is the most difficult of the three. It's not one that can be solved overnight, it's not one that can be solved with the wave of the legislative wand. It requires, like in operating a business, daily, consistent execution. And that is your challenge, Mr. Commissioner, in implementing a solution to the quality problem that exists today.

The notice problem is one which I think creates a great deal of the emotion that you're seeing displayed today, and one which can be directly addressed and resolved. Since it's fundamentally due to the delay in the publication, plus the process in the Patent & Trademark Office which allows one to prolong extensively the prosecution history, the solution should be quite clear. Publish patent applications early and secondly, discourage prolonged prosecution by measuring the life of the patent from its filing date. If somebody wishes to prolong their prosecution for eight or ten years after that, it's at their own jeopardy.

This may sound in some regards like a call for harmonization, but I don't wish it to be misunderstood as such, because I'm a strong advocate of harmonization only when there's a specific identifiable problem being resolved by such harmonization.

I'd also suggest that the software industry would probably be a supporter of the idea of implementing a very short time period prior for publications after filing, perhaps shorter than even the current European and Japanese model of 18 months.

Concerning the tied-hands problem, the proposal or suggestion that I have is really in two parts -- a minimal approach and then a more comprehensive approach. The minimal approach would involved converting the existing re-examination procedure into an inter partes procedure, where opportunity for equal advocacy and appeal would be available. I would also strongly recommend that you permit a much broader range of prior art to be introduced into the proceeding. Even when a challenged company would like to take advantage of the re-examination procedure in our industry, it's very commonly the case that the form of the prior art that we have available to us is not acceptable in the re-examination procedure, and I would recommend you open it up much more widely, allow oral testimony, allow physical demonstration and make it a true inter partes procedure. Address all issues relating to validity and reference to prior art.

Finally, I would remove any appearance of a bias in favor of the patent applicant. I would do so by changing the trier of fact in the re-examination procedure, and I would strongly recommend you upgrade it and so the credibility of the Patent & Trademark Office is enhanced and prolonged.

The more aggressive approach that I would recommend would be a full-fledged opposition approach. Time for challenge to a patent should be set for a reasonable period after its issuance. I would recommend a speak-now-or-forever-hold-your-peace approach to that opposition procedure. Companies can come forward during the time allotted, challenge with their best shot, and thereafter the issue has been resolved before the Patent & Trademark Office, no longer subject to repetitive review by every court that might have the patent presented before it. This will have the benefit of unburdening the courts from dealing with frankly issues they are not well-positioned to deal with, and it would streamline subsequent court proceedings to deal with issues relating principally to infringement.

COMMISSIONER LEHMAN I think we're running a little over, so maybe we can wrap up.

MR. PATCH: I only have one more comment, and I will excuse myself. Sorry.

The term of the patent in an opposition such as I've recommended should be extended for the period in which the opposition goes forward to avoid abuses of well-funded challengers keeping an opposition going for a long period of time.

Thank you.

COMMISSIONER LEHMAN Thank you very much. Those were interesting ideas. If you were here yesterday, you know, we announced actually that we have already taken one of your suggestions, and that is that we are going to be submitting legislation to Congress to deal with the so-called submarine patent problem in part by moving to a system 20 years from filing will be our new patent term, and that will at least in part address that difficulty. And then it turned out to be kind of a win-win situation because we were able to get the Japanese to make some concessions to us in return for our agreeing to do that.

And I'd like to also just use this as a forum to make a point about harmonization process. We announced earlier this week that we were going to suspend the patent harmonization exercise that we had been engaged in with the World Intellectual Property Organization which would have required us to change to a first-to-file system in the United States. That's not because this administration or I am opposed to harmonization. In fact, quite the opposite. Clearly the best patent system for Americans would be one in which you could file a patent in our Patent Office and then with great certainty and trustworthiness get very, very rapid protection everywhere else in the world that would be sound and that you could trust and so on.

That was not the deal that we had cut in the last administration, that wasn't the harmonization exercise that we were engaged in. And so we're basically going back and we're not going to abandon the principles of the system that many Americans feel favor them until we really have a system in which we receive very, very tangible results in other patent systems. And indeed this illustration of where we move to -- or we agree to do something that is also in our benefit in this case, I think the 20-year term from filing, we have received the tangible benefit from the Japanese.

I am very optimistic that we will, in fact, achieve true harmonization at some point in the not too distant future, but it won't happen if we just simply make all the changes unilaterally. So since you raised that point, I just wanted to explain where the Patent Office was on that. Thanks.

Yeah, I guess Christopher Byrne, Senior Intellectual Property Counsel for StorageTek, who is representing the American Committee for Interoperable Systems, ACIS.

--oOo--

CHRISTOPHER BYRNE

STORAGETEK -- ACIS

MR. BYRNE: Good morning. I'm Chris Byrne, Senior Intellectual Property Counsel for Storage Technology Corporation, or StorageTek. I am testifying today on behalf of the American Committee for Interoperable Systems, or ACIS, to which StorageTek belongs. ACIS sincerely appreciates this opportunity to provide testimony.

By way of introduction, I am an electrical engineer and a lawyer, and a registered patent attorney before joining StorageTek as patent counsel in 1991. I spent six years on the Intellectual Property Staff of the Hewlett-Packard Company.

I will address Topic A, Questions 4(a) and 5: Does the present framework of patent, copyright and trade secret law effectively promote innovation in the field of software? Do you believe a new form of protection for computer programs is needed?

Because these questions are two sides of the same coin, I will respond to them together. ACIS members include numerous innovative high technology companies such as Sun Microsystems, NCR and Broderbund Software. My own company, StorageTek, is headquartered in Louisville, Colorado, which is about five miles east of Boulder. We employ thousands of people worldwide, and we had 1993 revenues of approximately $1.4 billion. StorageTek designs and manufactures high performance data storage and retrieval systems for mainframe, mid-range and networked desktop computer systems.

Our customers include many Fortune 200 communication, transportation and financial companies. In fact, if you recently made a phone call, bought an airline ticket or bought or sold securities, chances are that records of your activity is stored on one of our products, awaiting ready access and retrieval when necessary. Our competitors include IBM, Hitachi and Fujitsu.

Like other ACIS members, we rely heavily on our nation's intellectual property system to protect our most valuable assets: the innovations of our engineers, particularly our software engineers. Without adequate intellectual property protection, we could not protect and recover our substantial investment in research and development. For instance, at StorageTek last year, we invested approximately 10% of our revenues in R&D -- that's over 140 million dollars. Without that R&D investment, we simply cannot stay competitive and in business. Indeed, last November Vice President Gore himself toured our substantial R&D facilities and personally previewed key technology which we believe will facilitate his grand vision of the information superhighway.

While all ACIS companies believe in strong intellectual property protection, we also believe in balance. We believe that overprotection is as threatening to innovation as underprotection. The need for this sophisticated balance is particularly important with respect to software, which is so pervasive in our economy and critical to its growth in our national leadership and high technology.

ACIS believes that it would be a dangerous act of underprotection to deny patent protection to software subject matter per se. But it is an equally dangerous example of overprotection to fail to expeditiously implement needed corrections in the way we currently do software patents, if not all our patents. Those needed corrections are well known, and ACIS has gone on record in support of them. They include:

Improving the software prior art database so that it is accurate, timely and includes both patent and nonpatent prior art. The quality of the software patent database will be directly related to the quality of the software patentability examination by the PTO.

Working to raise the skill level of PTO examiners who are charged with the vital and difficult task of examining software patent applications. One way to accomplish this is with site visits by examiners. For instance, last year two groups of examiners, one from Art Unit 2308 led by Michael Fleming and another group from Art Unit 2507 led by Bruce Arnold, visited StorageTek. They spent valuable time with our engineers and our patent committee learning how we do R&D and how we make our decisions about which inventions to seek to patent. We were very favorably impressed with the legal and technical expertise of the examiners, who too often are merely names at the end of an office action.

To our mutual benefit, the examiners learned about the challenges we face in innovative R&D and we received a much better understanding of the difficult nature of the examiners' work. Educational visits by examiners is one way to raise those skills.

Implementing key procedural reforms to prevent applicants from secretly and indefinitely submerging their applications in the PTO until they are ready to ambush the public -- and I think your 20-year limit is going to go a long way toward solving that problem. Otherwise we need to speed the examination process and include accompanying public notice of possible patents. Many such procedural reforms are considered as a function of possible harmonization of patent law, but such reform in this country should proceed with or without harmonization.

And just a footnote here: I think one of the very positive fallouts of this meeting has been the offline interaction among participants. And just as an example, yesterday I spent some time brainstorming with my counterpart at Silicon Graphics, Tim Casey, and Rob Stern, an attorney in private practice from DC, and just over lunch we were talking about the problems that we have with the Patent Office and we were brainstorming possible solutions, and a number of those were things like regionalizing the Patent Office, industry-sponsored technical colleges for examiners, expedited application procedures, possibly limiting patents to one independent claim, and the automation requirements of the modern patent system. And one of the conclusions that we came to was that we definitely believe that the water glass at the PTO is half full as opposed to half empty, and we think that hearings like this are going to be an important first step towards filling the glass.

But all the good work that we undertake to improve software patents will be simply undermined if we do not address another balancing issue, and that is the proper balance between patent versus copyright protection of software. This is because copyright, if misapplied, can achieve patent-like protection for software functionality. This misapplication is particularly dangerous when we consider that there is no examination for copyright as there is for patents; a copyright registration does not specify the boundary line of protected expression in a work, whereas a patent is explicitly bounded by the terms of its claims; and copyright protection outlasts patent protection by at least a factor of four.

This de facto patent protection under copyright is

particularly pernicious with respect to interface specifications. Unlike novels and plays, which stand alone and do not need to interact with other works, computer programs never stand by themselves; they function only by interacting with a computer environment. If the developer of an environment can use copyright to prevent other developers from conforming to the system of rules governing interaction within the environment -- to its interface specifications -- the first developer can gain a patentlike monopoly without ever subjecting his system of rules to a patent examination. In the absence of competition, the first developer would have little incentive to develop more innovative and less costly products. Moreover, this result is particularly dangerous to a company such as mine.

StorageTek designs and manufactures data storage peripherals which interface with the computers made by the dominant American, European and Japanese computer vendors. With de facto patentlike copyright control of their operating systems, these vendors have the potential to therefore control functional access to that interface and therefore exert market control over subject matter, i.e., the peripheral device, which is completely beyond the scope of the copyright itself. This is dangerous overprotection of software via copyright.

StorageTek joined ACIS because of our concern that the courts and the U.S. government were losing sight of the importance of maintaining a balance between incentives and competition in the area of intellectual property protection of software, particularly copyright protection.

From the outset, it was our believe that the proper application of traditional copyright principles such as the idea/expression dichotomy, merger, scenes a faire, and the fair use doctrine would yield the appropriate scope of protection for software. Recent court decisions have validated this.

The Second, Ninth and Federal Circuits have all found that copyright does not protect functional interface specifications. Further, the Ninth and Federal Circuits have found the reverse engineering technique known as disassembly to be a fair use and proper means to achieve functional interoperability. In our view, the Altai, Sega, and Atari decisions are not radical departures from traditional principles; rather, they return copyright to its proper course. We expect that the First Circuit will soon be consistent and overturn Judge Keeton's decision in Lotus.

Despite this positive trend in the case law, however, we fear that the U.S. government has allowed its laudable goal to improve the balance of trade to inadvertently divert its attention from the ultimate goal of our patent and copyright system: promoting the progress of science and the useful arts, as explicitly provided for in Article I, Section 8, Clause 8, of the U.S. Constitution.

We applaud the manner and spirit of these hearings, therefore, as solid indication that the U.S. government clearly appreciates that more protection is not necessarily better. We are also encouraged that Assistant Attorney General Bingaman has established a task force to review and reformulate the Antitrust Division's policies on intellectual property and antitrust. We applaud her observation that the scope of copyright protection for computer software has important competitive implications.

In summary, we see no need for a sui generis software protection law. Until recently, courts applied copyright in a manner that overprotected software, but the Altai, Atari, and Sega decisions corrected that aberration. Bad software patents also dangerously risk overprotecting software, but let's not throw out the baby with the bath water; let's move quickly to implement needed improvements in the way we do our software patents.

Thank you for this opportunity to present this testimony for your kind attention. I would be glad to answer any questions.

COMMISSIONER LEHMAN   Thank you very much. Appreciate your sharing that with us today.

Next I'd like to ask Gideon Gimlan from Fliesler, Dubb, Meyer & Lovejoy.

--oOo--

GIDEON GIMLAN

FLIESLER, DUBB, MEYER & LOVEJOY

MR. GIMLAN: Honorable Commissioner, distinguished members of the panel, may name is Gideon Gimlan, and I do not come here to represent any particular organization. It's true that one of the labels I wear, if you want to define where I am coming from, is that I am a patent attorney with the law firm of Fliesler, Dubb, Meyer & Lovejoy of San Francisco and Sunnyvale. This particular firm represents numerous high technology companies located in Silicon Valley and elsewhere. The work of the firm and my own work includes the preparation and prosecution of software-related patent applications in a variety of areas, including networked computer systems, graphic imaging systems and mainframe computers.

I have to add the immediate legal proviso that these comments are my own personal views based on general experience, and not those of any member of the law firm or of any clients represented by the firm.

I come before you wearing an additional label -- this is part of my general experiences -- that prior to becoming an attorney, prior to so-called defecting into law school, I was also an engineer who worked in the field for over seven years. I would characterize the nature of the work that I did as being a hardware/software engineer. And the reason I use that characterization is that a lot of the work assignments that I followed through with included the step of choosing whether to implement particular functionalities in software or hardware.

Insofar as the experience I've had from that background, I'll have to repeat what Ron Laurie so eloquently phrased, is that there is a spectrum, continuous spectrum, in terms of what we define as hardware and software, and it's almost impossible to cut that spectrum in half and define some line that separates something from being hardware or software.

Also, while I'm on that topic, it brings back to mind while I was working as a hardware/software engineer, Mr. Fiddler, who was here before, mentioned something about word processing being a old and obvious technique that shouldn't be patentable. I unfortunately go back to the days when people were doing affordable word processing with hard-wired machines back in the early '70s. The original

versions of affordable word processing came in the form of the IBM magnetic card, and there were a lot of companies who came out during that time and started to produce hard-wired word processing that eventually led to software types of devices. Generalized computer has taken it over, but the origins of it really lie in hardware in terms of having affordable word processing capabilities.

The question that I really wanted to focus on here today was Question No. 3 in your requests for comments: What are the implications of maintaining or altering the current standards for patent eligibility for software-related inventions?

And I'd like to retitle that as "What is the current PTO practice? And where is it leading us to in the software arts?"

My own personal experience is that, insofar as anticipation and obviousness are concerned, the examining corps treats software-related inventions no differently than other kinds of inventions. The legal tests for 102/103 determination are fairly well-established and most examiners treat software-based cases with the same uniform fairness as hardware-based cases.

The issues of finding good prior art in software area is no different than that in any other art. As an aside, in terms of quality, I find that the European patent office tends to find closer prior art for particular inventions than does the United States Patent Office, but again, that applies to general subject matter and is not specific to software-related cases.

Insofar as Patent Office inquiries into 35 USC 101, what constitutes statutory subject matter, I fail to see any across-the-office uniform consensus on what is or is not statutory, the OG guidelines notwithstanding.

The treatment of statutory subject matter question appears to vary greatly from examiner to examiner. Some examiners are lenient in what they consider to be statutory, while others seem to be on a witch-hunt for a 101 basis of rejection. This injects a considerable degree of uncertainty into the application process. You cannot predict the outcome of a 101 issue with any degree of confidence. It very much depends on which examiner you draw for your case.

Perhaps "software-related" isn't the proper term for what I am trying to address here. The problem more properly fits under the broader rubric of algorithm-related inventions and should the PTO be expending so much time and energy trying to weed out claims that arguably extend or encroach into nonstatutory areas.

I suggest that the answer is no. The Patent Bar and Examining Corps are wasting client money and taxpayer money arguing over metaphysical abstractions. That to technologists in the field sounds like we are debating over how many angels dance on the head of a pin. The case of In re: Iwahashi serves as a good example. It was not strictly speaking a software-related case because the claim preamble started off with, "An autocorrelation unit, dot dot dot, comprising."

But if one wished to take some license and rewrite the preamble to start with, "A computer comprising," and I note that that was done in Example B of the PTO request for comments, then in my mind this should not materially alter the gist of the invention.

Any digital signal processor, including the one in Iwahashi, can be viewed as a computing machine, or quote "computer" if you choose, one could then go out on a limb to call each invention that uses a digital signal processor as being software-related because its operations can be described in algorithmic terms.

Notice that I didn't say controlled by a computer program or controlled by quote "software". There are those skilled in the art who will argue even today that a computer program can be used as a description of the operations to be carried out by the machine, and the description does not necessarily have to form part of the machine that actually performs the described operations. The machine's control lines could just as easily be driven by combinatorial logic as from a memory source.

In the end, it should make little difference that an invention is implemented in hardware, software, or in-between-ware. In the eyes of the electronic circuits that carry out a given invention, there really isn't any functional difference. A set of electrical signals are first supplied to the DSP machine. Perhaps the input signals originate from a memory device like a ROM or a floppy disk, perhaps they come from an x-ray machine. Irrespective of origin, the signals are somehow transformed by the machine. Then they are output, perhaps for return to memory, perhaps for routing to some other immediate use, such as creating a real-time high-definition video image.

One inventor recently looked at me with bewildered eyes when I tried to explain some of the 101 concerns related to his particular case, and he said, "I don't understand, data is data, what does it matter whether it comes from an x-ray machine or from memory? What is government up to?"

And in that quote I've taken some literary license to replace what the actual source of the question was, but okay.

I think the problem and the answer lie in how we as human beings come to appreciate the subtle implications of a given invention. We need to step back and ask, Has the inventor come up with a faster or cheaper way of doing things even if the improvement is found in software? Has the inventor compressed the physical size of an apparatus so that something smaller can now do the job of something that previously had to be much larger? Has the inventor obtained a higher level of resolution than was previously feasible?

We see in hindsight that these kinds of improvements -- faster, smaller, cheaper, better resolution -- have brought us the miracle of affordable palm-top computers, ones that have pen-based graphical user interfaces, and ones that, arguably, give even the technical neophyte access to the powers of the digital revolution because of their intuitive nature.

I think we can all agree in hindsight that these are the kinds of innovations that our patent system is supposed to protect and foster. But when we turn away from past glories and look to the next invention, we are somehow daunted by the enigma of this thing we call software. We are all, in a sense, blind men beating at a pachydermial beast, each finding something different based on the angle from which we approach it. Some say this software stuff is more like the punched paper in a player piano, or like the music recorded

on a vinyl record. Others say it's more like the mathematical proofs of their college calculus classes. Yet others say it's something that is still in its infancy, that will grow and evolve into something we still do not fully understand.

Of course, in the meantime, software applications keep pouring into the Patent Office. So what should we do? Should we tell those who craft new software to go away? You are not welcome at the Patent Office? Should we direct every algorithm-smith over to the line at the Board of Appeals? Every examiner has his or her own personal angle on how to deal with this problem.

This leads to a haphazard system which gives inventors -- particularly those that have had the misfortune of being assigned to an "anti-algorithm" or "anti-software" examiner -- the impression that they are not receiving uniform, fair treatment. It is absurd in the mind of many technology gurus that an invention is okay if implemented in hardware but suddenly becomes unaccepted because it is implemented in software.

The pat answer, of course, for inventors who face such examiners, is that they can always go to the Board of Appeals, and if not satisfied with the results there, they can go higher to the Federal Circuit. But that doesn't happen with regularity. What really happens is that many patent-worthy cases fall by the wayside, not because the applicants agree with the examiner's 101 position -- and as a side comment, I sometimes wonder if even the examiners themselves agree with their official position -- but because of monetary considerations, it's just too expensive to go forward any further and appeal.

One could argue that this problem could be taken care of by well-to-do corporations, that they should lead the charge into the courthouse and help us create better law, but that doesn't always work. Some corporations are afraid to get on the bad side of a key examiner. Even those that are brazen think twice about pouring more time and money into an application that is already twice rejected by an examiner. Most inventors, and corporate executives for that matter, do not have the experience or patience to grapple with the kind of metaphysical questions that are posed when a Section 101 rejection is raised. For example --

COMMISSIONER LEHMAN Mr. Gimlan, we're running out of time.

MR. GIMLAN: Oh, I am, okay. Then let me skip to my proposal then. I think that the ongoing witch-hunt at the patent office for nonstatutory subject matter is in essence driving technology gurus away from the system. They simply don't understand it and will bypass the system.

My proposal is that unless particular claim in an application is clearly limited to the practice of a mathematical algorithm, the Patent Office should allow the applicant to disclaim within the body of the claim that portion of the claimed system or process that falls outside the scope of

35 USC 101, and then allow the case to go to issue as is, assuming there are no other bases for rejection.

After the patent issues, we should let experience and the advice of technical gurus help us to decide whether an accused device falls within the scope of a claim as interpreted under 101, or whether that accused device is

protected because in order to enforce the claim, you would have to transmute its meaning such that it becomes a claim to a mathematical algorithm.

COMMISSIONER LEHMAN We can certainly take everything and read it over very carefully with the specific suggestions.

MR. GIMLAN: Okay, thank you, Commissioner.

COMMISSIONER LEHMAN We really thank you for sharing this with us today.

This idea that we have created a sort of artificial determination for patent -- an artificial subject matter, in a sense, for patent lawyers to avoid the, in order to deal with these 101 determination problems is clearly something that we've heard from other witnesses here. This is worthy of looking into.

Next I'd like to call Tom Cronan, Secretary and General Counsel of Taligent, Incorporated.

--oOo--

TOM CRONAN

TALIGENT, INC.

MR. CRONAN: Good morning, Mr. Commissioner. My name is Tom Cronan, and I'm the General Counsel and Secretary of Taligent, and I'm testifying today on behalf of Taligent.

I must say after listening to yesterday's session during the morning, it is with some trepidation that I approach the stand this morning. As a lowly high-tech copyright lawyer, I guess it's malpractice per se to be even testifying here today.

What we're going to talk about is the patent system and how the patent system is critical to stimulating investment in the software products and technologies areas.

We will recommend some refinement to that system. We want to have a system that is designed to support high technology, this industry which gives us high-tech, high-wage jobs. We want to continue to be the world technology leader in software. I will talk about both start-ups and large companies and how the patent system benefits both.

We want to stimulate investment, that's what we're all about. There's lots of various interpretations of whether the patent system will help or hurt venture capitalists and others make decisions on whether to invest in software. We certainly have an opinion on that that we'd like to share with you today.

So I'm going to discuss three things: Why software is important to new ventures, why software patent protection and related inventions is important to new ventures, why the patent system attracts investment, and the refinements that we discuss to the current system.

Taligent, as an example, is a new, founded in 1992, small, starting with 170 employees -- we now have 350 employees and by the end of this year will have 450 employees -- innovative, high-tech, high-risk venture. We're going into an extremely competitive marketplace, we're going to have an object-oriented operating environment that will compete with Microsoft's offerings, clearly the dominant player in a lot of competition in the area.

We are going to compete on the basis of innovation. We are going to try and establish a foundation technology for

the industry, a new type of technology that will be an operating environment that's open and extensible at all levels, based on a brand-new technology.

As I was sitting here this morning listening to our first speaker talk about the fact that two of his programmers could program a million lines of code, I was thinking about our own efforts. We have been developing this technology before the company began -- it's been now six years in development. And by the time we're done, we will have a very elegant program with 750,000 lines of code. I would think that some of our developments may be a little obvious than those this morning, which were the million lines of code.

This architecture is being developed from the ground up -- a clean sheet of paper. That's why it's going to be innovative, that's why it's foundation technology. We're not unique. There's going to be lots of other foundation technologies that will have to be established for the information superhighway, and they will not be done by garage shops. They will be done by larger ventures, by people in the industry who understand the technology, who understand the risks. We could have never attracted venture capitalists to our venture, we are too risky. We're funded by the industry, like now three separate large companies in the industry who will be funding us.

So as we look at our architecture, as we look at our developments, we see ourselves very much like the beginning processor developers. We're looking at an architecture that's extensible, we're looking at functions and features that are new, and we think that software should be protected in the same way as hardware.

We don't think there's any material difference and we think that the policies of the patent system are in favor of protecting software the same way as hardware.

Next I'd like to talk about why that patent system attracts investment.

The software industry is different than some other industries. It has a very front-end loaded investment. You have to invest all of your risk capital before you know whether or not your product is going to be competitive, before you know whether or not anyone will buy your product. So you have a lot of risk and you have a lot of uncertainty.

If you add on that additional risk and uncertainty associated with not knowing whether or not you'll be able to protect your new and innovative product from a major competitor, you may not have any investment at all. The market risks are very high, and if competitors can take a utilitarian function which you spent a lot of time and effort designing, as we mentioned, and use those against you without having the same R&D expenses, you would not be able to support these investments.

The other reason, policy reason, is as we look at this technology, copyright law will not be adequate to protect the types of object-oriented programming that we're developing. We have designed our system so that in object code developers using our development environment can go in and modify almost every portion of our system. The entire architecture is open. We cannot protect it by trade secret. There will be those, I'm sure, who will argue that the

entire system is an interface. That raises some copyright issues -- as us lowly copyright lawyers know.

We think that the patent law encourages disclosure. In the absence of patent protection for this technology, I would certainly advise my client to make sure that we keep as many of the interfaces closed and not open as possible, make sure that most of the important functions aren't disclosed in documentation.

With the patent system, we're able to disclose those not only in the patent applications but we also will want to make sure that for those important utilitarian aspects that we can't afford to patent, that we'll publish and establish the prior art in those areas.

We want to make sure that the United States continues to be the world leader. As you probably have read, there is now more R&D dollars going into software ventures and software technology in Silicon Valley than hardware. That wasn't true two years ago; it's true now and it increases every year, and this investment is dependent on having a system that protects that technology.

One other point I would make on this is, an earlier speaker said that there was presumption that the first to market would win. I would challenge that presumption. If you look at Excel, if you look at Word, I do not believe that they were the first spreadsheet or the first word processor on the market, but they have substantial market share.

Next I'd like to go back and talk about some refinements that can be made to the current system. There's been lots of suggestions that have been offered today, and I would like to just offer a few additional suggestions, and also some endorsements of some of the suggestions that have been made.

We applaud the Patent Office's initiatives on education. We have been very active with Gerry Goldberg in trying to educate the Patent Office. We spent time, money and effort sending Mike Patel, our VP of Advanced Technology, back to the Patent Office. He educated a large number of people for half a day. His talk was not about Taligent; it was about object-oriented programming from its inception. And he was a professor for eight years before he came to Taligent, so he had a very good background.

It was so well-received that the Patent Office invested in sending 17 senior supervisors out to Taligent and to other companies in the Valley. They spent a day and a half at Taligent understanding our technology in great detail. And most of the responses we got from them were extremely positive.

And I was there for the wrap-up of that session, and I heard one of the senior supervisors saying, "You know, this is great, we learned a lot, we now have new innovative ways to deny your claims."

One of the other things that I think would really help the Office -- and I know that you have begun this and we think you should continue it and with even greater speed -- hire more computer science graduates who don't have engineering degrees. These are the type of people who understand this technology, the type of people that should be evaluating these types of inventions.

I think that some of the other problems of obviousness, there are lots of issues with obviousness, but one of the

things, consistency and having people that understand the technology better would certainly help the nonobvious issue.

In addition, I have one other proposal that's a little different than some of the proposals. To deal with this problem of having all of the prior art not found anywhere or searchable by advanced technology -- and I know that you're trying to put together databases in advanced technology -- we would propose a human database.

There are people, there are consultants, there are people who can be hired who have lots of expertise and lots of industry experience in the relevant areas that a lot of the software patents will be filed on. These people, because of Internet and because of advanced technology, don't even have to be located with the Patent Office; they can be reached through advanced electronic communication, so that people in California who have a general reluctance to move back to Washington, D.C., especially after the snow storms of this winter -- except of course the people in Los Angeles -- can be reached by Internet.

Finally, on publication, we would be in favor of publication prior to issuance. I think that one of the issues that hasn't really been addressed is that there needs to be some certainty in this publication scheme, a publication scheme that would be tied to when the patent issued and publication prior to that would have a great deal of uncertainty because of the great deal of uncertainty in when the patent is processed through the Patent Office and when it gets issued.

It would be important to understand whether or not your product is out in the marketplace before the patent issues, and before the publication. So I think having a set time period, like 18 months, or some other time period that's appropriate for the industry, is the best way to proceed.

So let me just conclude. We think that protecting software-related inventions is critical to investment and for ensuring certainty on return in those investments. We propose some refinements to the current system; we think with those refinements we should be able to enhance competitiveness of the software industry in the United States, particularly the high risk, high level of investment foundation technologies, such as our company, which will be needed for the information highway.

Thank you.

COMMISSIONER LEHMAN Thank you very much, Mr. Cronan, for sharing those ideas with us.

Next I'd like to ask William Neukom, Vice President of Law and Corporate Affairs for Microsoft Corporation to come forward.

--oOo--

WILLIAM NEUKOM

MICROSOFT CORPORATION

MR. NEUKOM: Good morning. My name is Bill Neukom, I'm Vice President of Law and Corporate Affairs at Microsoft Corporation.

I appreciate having the opportunity to be here to present the current best thinking of our company on the subject of this hearing. If I'm making points that are not clear or deserve some comment or questions, please don't hesitate to interrupt. I think I'm scheduled to go most of my allotted time with my prepared remarks, but I want to communicate while I'm here this morning as best I can.

Microsoft is a developer and marketer and supporter of a very wide range of systems and applications software products for personal computers. By having helped to make it easier for users to work with their personal computers for an increasing number of purposes, the company's products have been able to contribute to what's sometimes referred to as the "PC revolution," which has occurred in the past 12 to 15 years. The growth of the company has paralleled an even more important statistic, which is the increase in the number of people who use personal computers in this country. About a million people were using personal computers in 1980; by today we estimate that probably 90 million or more people are using personal computers.

The software industry is a major contributor to the economy of this country. In the last five years, virtually every study of the key technologies of America's present and future have identified the vital role of computer software industry. Software is characterized by both its very rapid technological innovation and by the widespread use of that technology in downstream markets. Computer software improves the competitiveness of other industries in this country and around the world because it helps to make -- our products help to make those enterprises more efficient and more innovative, and it's the continuous evolution and enhancement and improvement of software products that permeates much of the economy of this country.

The US software industry has experienced quite remarkable growth. Measured over the past ten years, it is the fastest growing industry in this country by any rational measurement; it is now larger than all but four or five industries in this country's economy. The growth has been fueled by strong export performance by US companies; 75% of the world's sales of pre-packaged software come from US software companies; and the 100 largest American software companies earn more than 50% of their revenues from offshore sales.

The key to much of this is strong intellectual property protection, which we and our colleagues and competitors in the industry view as essential for US software industry to continue to compete globally and continue to play a leadership role in this nation's economy.

On this morning's subject of patent protection for computer software, we believe that the existing laws in the form of the statute and the regulations and the case law provide both an adequate and an appropriate framework in which to assess the patentability of software-related inventions. This is not to say however that the existing system cannot be improved, and we commend the patent office for its willingness to take a constructive view of that challenge.

We appreciate the Patent Office's commitment to the improvement of the examination process by increasing the number of examiners and the expertise of the examiners in software technology and providing better technical training for the examining corps.

We also agree that or support the Patent Office's decision to pursue some reform of the re-examination process. I read in the Commissioner's opening remarks from yesterday that there is some legislation forthcoming and we look

forward to reviewing that and supporting it in a constructive manner, assuming it does things which we think are beneficial to the process. The advantage of reforms to the re-examination process are measured both in terms of a more efficient determination of patentability, but have the very handsome byproduct of reducing the threat of expensive and protracted litigation.

We believe the software industry would benefit from greater availability of prior art; this is not a novel subject to you experts or to the audience, but patent applicants need to know more about prior art, the office needs to know more about it, and parties to infringement actions or threatened infringement actions could benefit from better, earlier information about prior art. We are a participant in the Software Patent Institute's efforts to gather that prior art, and we are trying to exhort our colleagues and competitors to step up and make more technical information available, so that it can become part of a richer and more relevant database of prior art.

And finally, we think that the industry would benefit from a reduction in the average pendency of applications before the Patent Office. We don't presume to think that that's an easy matter to accomplish, but we think it's important; the more prompt issuance of patents will provide industry participants with a better return on their substantial investments in technology and in the patent process itself. That is particularly material for an industry like ours, which is so fast-moving and where today's invention is next year's afterthought.

With a commitment from both the industry and from the Patent Office to implement these kinds of changes and perhaps others that have been suggested or will be thought of, we believe that the existing system can mature in a fashion that effectively achieves the constitutional goals of stimulating and protecting innovation in a competitive context.

Let me try to respond to each of the questions that have been published for these hearings. Question one asks, What aspects or specific examples of software-related inventions should be protectable via the patent system?

Without addressing each example individually, Microsoft notes that this inquiry appears to subsume two basic issues. First of all, should patent protection be available in some form for inventions embodied in software; and secondly, if so, how should protection be characterized? As to the first issue, we do not believe that patent protection should be withheld from an invention that otherwise meets the statutory requirements for patentability, simply on the basis that the invention is or may be embodied in software. I think that point is reasonably well resolved by the courts and by the Patent Office at this stage.

With regard to the second question: The characterization of the protection, we favor claims structures that clearly recite those aspects of computer software-related inventions that are novel and unobvious, and allow an accused infringer to readily identify the activity or activities that may be proscribed under the claim. The success of a particular claim in meeting these objectives may depend, however, less on the form and more on the substance of the claim and the supporting specification.

As to question number two, the impact of software-related patents on the industry, Microsoft has never initiated an action for patent infringement. We have, however, unfortunately been the defendant in several lawsuits involving software-related patents. The defense of those suits has consumed considerable of our resources, resources we'd prefer to use in positive and constructive research and development efforts. Even so, we are committed to the existing patent system as a reasonable and responsible vehicle for protecting software innovation, particularly when that process is viewed in light of the ongoing effort being made by the Patent Office and the courts and more and more, I'm pleased to say, by the industry to improve the systems application to our technology.

The dichotomy illustrated by our position reflects the equity that we think can be achieved by the existing system in balancing the competing interests of protecting innovation on the one hand and preserving competitive freedom on the other hand.

One potential way of lessening the negative impact of software-related patents on the industry would be to consider again this subject of reform of the re-examination process. The threat of litigation involving a patent of questionable validity can be particularly damaging to a smaller company, which may not have the financial or the human resources to effectively challenge the patent's validity in the federal court process. Although the existing re-examination process affords a potential defendant an alternative venue in which to contest a patent's validity, the utility of the current re-examination process is limited by its ex parte nature and the limited scope of prior art that can be considered.

The Patent Office, the Patent Bar and industry participants should carefully consider whether these and other limitations on the existing re-examination process should be overcome.

Question number three addresses the implications of maintaining or altering the standards for patentability of software-related inventions. Microsoft believes there are several advantages to the maintenance of the existing standards. We're not suggesting they should be frozen, but we believe that they are fundamentally sound and there are reasons to continue to rely on them in the main.

Workability. Although the expression and application of the existing standards may not yet have fully matured, the standards have evolved slowly over a number of years and do provide a stable framework in which to assess the patentability of computer software-related inventions. Improvements have already been made. The Patent Office has already taken steps to improve the quality of examinations, as we've noted, and the software industry is working to enhance the effectiveness of the Patent Office's application of existing standards through, among other means, the work of the Software Patent Institute.

Thirdly, this is a way to avoid greater near-term uncertainty. Both the industry and government have made considerable strides in understanding and applying the existing system, particularly in the last few years. The introduction of some new statutory or regulatory standards would almost certainly present a new set of uncertainties or ambiguities,

making a major revision perhaps more unsettling to the industry, at least in the short and perhaps the midterm.

And finally, there are investments that have been made under the current standards by industry members, and significant changes to the patent standards might compromise the value of those substantial investments.

Question four asks whether the existing framework of patent copyright, trade secret protection effectively protects and promotes innovation in the software field. Microsoft response to that would be, yes, it does. The importance in the growth of the software industry described earlier in my remarks has not occurred in a legal vacuum, as I'm sure you are all aware. As noted in the Patent Office's discussion of Topic A, the Supreme Court held in 1981 that the mere presence of a software-implemented mathematical algorithm in an invention does not automatically preclude the invention from being eligible to receive patent protection. Similarly, the copyright statute has expressly addressed the subject of computer programs since 1980. The maturation of the industry under the existing legal framework suggests that the framework is appropriate and that it is reasonably effective.

While copyright has been and is an important and effective tool for the software industry, that does not mean that there is no role for patent protection. Indeed, there is a large and growingly important role for patent protection.

Microsoft believes that the software patent law will continue to mature and we would trust rapidly enough to effectively support growing industry awareness and use of software patents.

The final question asks whether a new form of protection is required for computer programs. Microsoft does not believe that a new form of protection is required; the existing patent system has a long history which reflects an appropriate balance in protecting inventive technology. The system has served American industry well. We are aware of no compelling reasons at this time why it should not be continued to be applied and approved as it is applied to the field of computer software.

Thank you for this opportunity to share Microsoft's current thinking on this very important subject.

COMMISSIONER LEHMAN I've a question to ask if you'd just hang on for a moment.

Yesterday we had a witness from the League for Programming Freedom, who displayed a chart indicating, showing who had applied for patents and who hadn't, and not surprisingly, the chart showed, for example, that IBM had the most number of patents, AT&T had the next number, and then it went down to some of the companies that we associate more with the mass market software industry, like Microsoft and Lotus and Novell, Borland, Next, Oracle, etc. And he noted that Microsoft only had thirteen patents, Lotus only has seven, Novell has one, WordPerfect has none, for example.

And the implication of that was that basically the patent system has played virtually no role in the stimulation of this fabulous industry that you've talked about, and of course that's part of the Article 1, Section A, mandate is to stimulate progress in the arts, and that, in fact, this particular witness, a computer programmer, felt that it was having a counter stimulative influence because programmers didn't

want to even touch their keyboard before they consulted the legal department.

Microsoft obviously did develop to where it is now without significant patent protection. Do you see something, is there something that's changing in the industry that is causing you now to take a look at patents? What is different about now, today, than the early 1980s when you first came out with your first products?

MR. NEUKOM: I think what's different, in terms of patents is that our industry and certainly my company has become much more aware of the value of patenting software-related inventions. I think as a whole the industry relied extensively on trade secret and copyright protections. It's important to remember, particularly in terms of the mass market kind of software that you describe, that this is a very, very young industry. We tend to think of this as an industry which has always been about the size and had about the reach that it currently has in the mid 1990s, but when you realize that graphical computing, for example, personal computing, has really only come of age in the past four or five years, and portable computing has really only come of age in about the same timespan, you realize that this is an industry which has grown so fast and diversified so quickly that to think about the early '80s is to think about generations-old forms of the current industry.

And there were questions, as I know the Commissioner knows full well about the copyrightability of software, there were some questions among lawyers about the patentability if software. As those questions have been resolved by the courts, the companies have had to pay attention to that. I think that the companies at the top of that list, the IBMs and the AT&Ts tend to be companies which are hardware companies as well, who have a culture of patents and the companies toward the bottom of the list are more purely software companies who didn't come into the industry with that sort of culture and awareness of the values of the trade-offs of patents, and so we're essentially as an industry, I think, catching up with the patent process, and I think that there has been a very material increase in attention paid by the legal staffs to the prospect of patenting software.

We will soon have six patent lawyers in my department, and that's grown from a group which had none in it two and half years ago. We've always relied on outside counsel and will continue for purposes of applications and prosecutions because of the nature of the work involved in writing those claims, but it's certainly much more center-of-screen for law departments and I think that companies are making informed decisions about where they want to spend their scarce legal resources in terms of protecting their intellectual property rights.

I think that there will be -- the hardware companies went through this, I think, in a somewhat transferable piece of history where they were filing an increasing number of patent applications and as they were issued, there came a time when some of those companies had to reach cross-licensing kinds of accommodations with each other. But we, at Microsoft, take the view that to the extent that there is important technology, that seems to us to be patentable, we do want to raise the level of awareness among our technical people to be in touch with the law department, for us to decide whether to pursue an

application. I think that's generally happening around the industry. The number of patents issued, of course, is only a sense of how many are in the process, and I think the industry itself is putting more resources in the effort.

COMMISSIONER LEHMAN Microsoft obviously as a big company takes it a little bit on the chin because, the big guy, everybody is concerned about Microsoft's market power and size. That's come up in the discussion of issues, such as decompilation, for example, in the copyright side, where one of the things that we've heard very strongly is that we've had a lot of the testimony that oftentimes -- by the way, testimony from people who feel that there should be no patent coverage for software, but sometimes from people who think there should be patent coverage, too. We've had a lot of suggestions that the copyright law should be construed very narrowly also, so that only literal code is covered.

And then we've also heard some testimony about the decompilation issue too, that there ought to be access to products through that process, and I wonder if you have a comment on that.

MR. NEUKOM: Generally speaking, our view is that software ought to be treated by the copyright law and process the way any other original creative expression is treated, and not distinguished by the nature of its technology. And I think that the courts have been sensible about developing and reinforcing that notion, in terms of broadening an exception for reverse engineering or decompiling.

We are very concerned about that, not just in the law of this country but in the law of other countries, and as the panel knows, that matter is currently the subject of some serious consideration in Japan, and we are very much concerned, for example, in that context that an already none-too-strong copyright law may be further weakened by a too-broad exception for decompiling, which would essentially expose US -- this is not a Microsoft issue, this is a US software publishers' industrywide issue -- would lay open our technology to a shortcut by Japanese and other software companies who could bring to market products which would compete with a very unfair advantage, an advantage of not having had to spend the research and development resources to create and invent the expression and the ideas that go into that product.

COMMISSIONER LEHMAN Thank you very much.

MR. NEUKOM: I hope that's responsive.

COMMISSIONER LEHMAN Finally, next I'd like to ask Charley Morgan to step forward, Vice President of OPEB Funding for The Prudential Insurance Company of America.

I think, Mr. Morgan, you're one of a kind. You're our only insurance person.

--oOo--

CHARLES MORGAN

THE PRUDENTIAL INSURANCE COMPANY OF AMERICA

MR. MORGAN: Good morning, I'm Charles Morgan, I am Vice President, OPEB Funding. OPEB Funding is a business unit of the Prudential Asset Management Company. We fondly know of it as PAMCO. PAMCO is a wholly-owned subsidiary of The Prudential Insurance Company of America, the largest insurance company in the United States. PAMCO offers investment management and related administrative services to US employee benefit plans, foundations, endowments and other domestic and foreign institutional clients.

I have to say I've done a lot of flip-flops this morning and yesterday, listening to the other people talk as to whether I'm really relevant here. I think I'm going to be asking you to make a fairly significant shift in context as I give you my remarks. I'm really here to give you an anecdote -- sort of a horror story.

We are an old industry. The insurance industry in this country has been around for hundreds plus years, Prudential has been around since 1875. We have old products that have been expressed on paper, copyrightable, that are now finding, in manual systems to implement those old products, they are now finding new expression in electronic form, implemented through new systems that also use software and computers. And here I am, a former tax lawyer, who now finds himself running a small business with The Prudential. Prudential has 100,000 employees; my little business has eleven employees. So I listened to the conversations about big organizations against the little guy, the innovator, and wonder where I fit because I am both. I am not a patent lawyer, and I am not a software expert. All I know about software is that I use it every day.

As I mentioned, I work in OPEB Funding, O-P-E-B stands for Other Post Employment Benefits. OPEB Funding offers institutional investors financing solutions for their retiree healthcare liabilities. If you read the paper today, Clinton and his agenda includes healthcare reform, it's a big part of my life.

My work is different from traditional pension benefits, this is post retirement healthcare benefits that I work with. Our primary product is a flexible premium group life insurance contract. The contract offers the employer participating life insurance, but also a broad array of investment accounts similar to pension accounts. Typically it is purchased by a trust, and the employer uses that trust to finance the cost of those benefits.

Now, our product development work for this product that we're selling began in 1987. We're highly regulated. We had to file with state insurance departments for approval of our forms, their content are dictated in large part by the states. Our first state approval occurred in 1989, in April. Our first product installation occurred in August of 1989. Our system development work paralleled that timeframe, and built on existing Prudential systems already used for very similar products.

Now, I should note that our product took a very old idea; that is, a life insurance contract and a trust to fund employee benefits, and updated it by employing a group insurance wrapper rather than an individual policy wrapper. And that was a significant innovation only because legal impediments in our industry were perceived to say it was not possible to do it. We did it.

I then turned to our lawyers and I said, "How do I protect this innovation?" They said, "You cannot, you cannot patent a life insurance contract. You can copyright it, but you can't

patent it".

Much to our surprise, you issued a patent, we call it the (Premit Patent) in 1992, covering a system employing a VEBA trust, V-E-B-A, which in turn purchases variable life insurance contracts to fund retiree healthcare benefits. An employer contributes money to the VEBA and obtains a tax deduction. The money is invested in the insurance contracts, eventually it's distributed in the form of healthcare benefits through a health claim system.

The patent owner has approached our clients and prospective clients, advising them that we would owe him royalties as a percentage of the investments in our group variable life insurance product, and that we would try to pass the cost back to them, which we would.

The inventor -- or the invention, that is, covered by the patent is comprised of numerous elements, including a VEBA trust, a variable life insurance contract, a couple of healthcare liability calculation systems for financial accounting and tax accounting purposes, a death claim collection system and a health claim payment system among other elements. Significantly, the patent owner has focused on collecting royalties solely from Prudential and exclusively with respect to our group variable life insurance contract; that is, the life insurance contract segment of this invention.

Now, why would we, Prudential, owe the patent owner royalties on this? All we're doing is selling a variable life insurance contract. We do not perform most of the functions comprised of the segments in the invention. While it is true that we do have death claim systems, those systems are inherently connected with the business of selling and administering life insurance contracts. We also have healthcare systems connected with our group health insurance businesses. But it would be merely coincidental if we happened to administer a health claim system for a client who purchases our variable life insurance product.

The focus of his royalty claims on the insurance contract segment of this, quote "invention" unquote, suggests that he merely wanted to patent a variable life insurance contract, something that we thought was not possible. To accomplish his objective, he merely surrounded the contract with sufficient quote "system trappings" unquote, to justify issuance of the patent on his invention and persuaded you to issue it. Now, he has not implemented any of those systems, he has merely patented the concept.

The patent fails on three conditions of patentability. The invention is not new. It is perfectly obvious to a person in the field, and it is merely a method of doing business. The Prudential has been in the business of selling life insurance since 1875. We've been in the employee benefit business since the 1920s. We have worked with pension liabilities for more than half a century. We have worked with healthcare liabilities for decades. Individual life insurance products have been used to fund pension liabilities since at least the 1940s. The Internal Revenue Service even issued a ruling, what we call PS No. 58, in the 1940s, to cope with pension funding with pension trusts investing in life insurance contracts.

Now VEBAs, an important part of this supposed invention, came into unique prominence for funding retiree healthcare liabilities in the late 1980s, only because Congress enacted ERISA in 1974 and DEFRA in 1984, thereby curtailing the

other tax-motivated devices employed previously. Originally called "Retired Lives Reserves", several IRS rulings in the 1960s and early '70s established the foundations of the tax deductions that ultimately were codified by DEFRA in 1984. Life insurance policies and VEBAs were favored funding instruments for Retired Lives Reserves even before Congress codified the rules. The essential role to be played by VEBAs prospectively was obvious to anyone in the insurance industry, the benefits consulting community, anyone who had worked with pension plans, trusts, life insurance and Retired Lives Reserves. VEBAs are not new, they're not unusual. A VEBA is just name that Congress put on a 501C9 trust in the Internal Revenue Code. You go read it, and that's the label in the section.

The primary purpose of the trust is to secure the asset from the employer's creditors so employees like you and me will get the promised benefits. Section 501 of the Internal Revenue Code has been there ever since 1954 and has antecedents going back to the '39 code.

The Prudential has been a leader in the development and use of record keeping and other systems required to support life, health and annuity and pension products. We employed the earliest computers doing the 1940s for statistical purposes, during the '50s we installed the earliest machines from IBM. Our computer systems became a substantial part of our business in the '60s with the automation of our policy and group pension administration.

Prudential developed the first medical and dental claims systems in the United States in the '70s. One of our major life insurance systems contains tens of millions of lines of code, that require 750 people simply to maintain in.

In our group insurance and PAMCO operations alone, we have more than a hundred major applications in development or under maintenance. Our annual budget for systems applications runs into the many hundreds of millions of dollars. If I'd had time to research it, I'd daresay we're upwards of a billion annual.

Notwithstanding that big investment, we have not pursued patents within Prudential, we haven't stockpiled them. And I asked our patent lawyer how many he was aware of, he knew of none.

I was interested in the picture painted yesterday, the big guy against the little guy, as I said, we have not been using these as a tactic, and we may well have to turn to that.

In my little business, when confronted with this patent I have only a few very unpleasant options. I can shut myself down, I can spend a lot of money on lawyers, pursue royalty litigation or litigation to get rid of the patent or I can pay a royalty which is what I see as nothing more than a ransom, extortion.

THE PTO re-examination process was not available to me for the reasons that Lee Patch mentioned this morning.

Now, I only have two more brief paragraphs with my suggestions, but they're echoes of what you have heard already today. You need to introduce rigor into your research of prior art. People like us are available to you, you ought to seek us out and learn a little bit from the community that is affected by the patent application. I agree with those comments wholeheartedly. I am not a patent lawyer. It was obvious to me that that's something you need

to do, and I applaud these hearings as a first step in hearing about that sort of thing.

I don't see the software patenting issue as the real issue, I see the real issue is research into obviousness and newness.

That really about sums up my comments except to say that in the financial services community, we have a very broad spectrum of products which at one extreme or the other have significant differences. At any point in between those two extremes, the line drawing exercise for you and me is incredibly difficult. And applying new expressions to age-old products like this needs to proceed with care.

COMMISSIONER LEHMAN Thank you very much.

MR. MORGAN: Okay. Thank you.

COMMISSIONER LEHMAN Next up and finally, our final witness for the morning is Les Earnest.

--o0o--

LES EARNEST

MR. EARNEST: Les Earnest, speaking for myself.

Based on my 40 years of experience in the computer system development, much of it before software patents were introduced, I believe that the alleged connection between such patents and the stimulation of innovation is tenuous at best and probably negative. Let me confess that even though I oppose the continuation of software patents, as a defensive measure I've applied for some that have been granted.

When I entered the field as a programmer in 1954 there were only about a hundred of us in the whole world, and each of us was turning out thousands of inventions each year, or maybe it was hundreds depending on your standards, but a lot. Software was given the same kinds of protection as other documentation, namely copyright and trade secret.

It was certainly a good thing that there were no software patents because my colleagues and I could have papered over the field and retired for 17 years or so to collect royalties. Since patents didn't exist, we kept working and had quite a good time doing it, sharing ideas and standing on each other's shoulders to see how high we could reach.

In 1956 I went to MIT to help design the Sage Air Defense System, it was a technological marvel full of inventions, both hardware and software. It was the first real-time computer system and depended on the large software system that was cooperatively written by many people. That was the first such system.

This project helped transfer a lot of technology from MIT to IBM, but almost nothing was patented. Dozens of Sage systems were eventually deployed around the country, each with a vacuum tube computer that covered a floor area about the size of a football field and an air conditioning system to match.

It is fortunate that this power, that the Soviet Union, never attacked the U.S. in that era, because the marvelous technology in Sage had several Achilles' heels that would have caused it to fail catastrophically under attack. However, those short comings were kept well hidden from Congress and the public, and as a result the so-called command control communications technology became a major growth industry for the military industrial complex.

The most recent example of that line of development being the grossly defective Star Wars system, but that's another story.

Beginning in 1959 I developed the first pen-based computer system that reliably recognized cursive writing. I believe that it was more reliable than the 1993 version of Apple's Newton. But the idea of getting a patent on such a thing never occurred to me or my colleagues. It wouldn't have done much good anyway because the computer on which it ran filled a rather large room, and the 17-year life of the patent would have expired before small portable computers became available.

In order to cope with a personal shortcoming, I developed the first spelling checker in 1966.

(laughter)

I didn't think that was much of an invention and was rather surprised when many other organizations took copies. And, of course, nobody patented things like that.

When John McCarthy and I organized the Stanford Artificial Intelligence laboratory, and I served as its executive officer for 15 years, there was a great deal of innovation that came out of there, including the first interactive computer-aided design system for computers and other electronic devices, early robotics and speech recognition systems, the software invention that became the heart of the Yamaha music synthesizer, document compilation and printing technologies that later came to be called desktop publishing. The Sun workstation was invented there. And the guy who invented public key cryptography was in our lab.

Few of these inventions were patented in the early period, but we later began to file for such coverage. The pace of innovation I note has necessarily slowed over time as the technology matures, but concurrently, of course, the amount of patent protection has increased. I suspect that these changes are connected.

Yesterday in this forum, my friend Paul Heckle said that software patents stimulate new businesses. I'm afraid that Paul has that backwards. In fact, new businesses stimulate software patents. Venture capitalists want the comfort of patents on products that are being brought into the market even though know-how is far more important in most cases.

In 1980 I co-founded Imagen Corporation, which developed and manufactured the first commercial desktop publishing systems based on laser printers. We filed for software patents to try to appease the venture capitalists, even though it was not actually important to our business, I believe. Of course, they didn't understand and the lawyers were happy to take our money.

Based on my experiences, I also joined the League for Programming Freedom to help resist the patent conspiracy and I later served for a time on its board of directors.

In summary, for many years there has been a great deal of innovation, there was a great deal of innovation in the computer software field with no patents, under the quote, stimulation of software patents the pace now seems to have slowed. I believe that there may be a connection, not only because of the time that must be devoted to covering and deciding what to cover and filing a patent application, but also because patents are owned by other organizations, many of them in fact based on prior art, and constitute a

mine field that must be carefully navigated. I recommend a return to the good old days when success depended on moving faster than the other guys rather than trying to catch them in a trap.

Thank you.

COMMISSIONER LEHMAN Thank you very much, Mr. Earnest.

That will conclude our morning session, and we'll reconvene at 2 o'clock, at which time our first witness will be Richard Stallman.

(Luncheon recess taken)

COMMISSIONER LEHMAN We can get underway. It seems to be a usual human tendency of somehow or other always running a couple of minutes late. So perhaps we can start right out again by calling Richard Stallman forward, please.

I assume you're -- you're just listed as Richard Stallman, but I assume you still have the affiliation with the Free Software Foundation?

--o0o--

RICHARD STALLMAN

FREE SOFTWARE FOUNDATION

MR. STALLMAN: I guess I'm just speaking for myself because, yes, I am involved in software development with the foundation, and I guess this is probably the opinion of the foundation too since I'm its president.

(laughter)

COMMISSIONER LEHMAN Great. Well, welcome, and why don't you proceed.

MR. STALLMAN: Okay. Each year the government creates new bureaucratic programs. Each is created for a purpose. That doesn't mean it serves that purpose or that it is worth the cost.

It's hard the close down unnecessary bureaucracies because people presume they must do some good. It's easy to admit a government program has drawbacks, but many won't seriously consider whether it does its job at all.

Thus we see, in the announcement of these hearings, the supposition that software patents are helpful. We are asked whether they protect software developers enough. Patent lawyers chose the word "protection" to imply that patents are beneficial.

I'm not a lawyer. I'm a programmer, considered a good one. I am here to explain why software patents impede software development and retard software progress. Software is like other fields of engineering in many ways, but there is a fundamental difference: Computer programs are built out of ideal mathematical objects. A program always does exactly what it says. You can build a castle in the air supported by a line of zero mathematical thickness, and it will stay up.

Physical machinery isn't so predictable, because physical objects are quirky. If a program says to count the numbers from one to a thousand, you can be sure it will do that. If you build a counter out of machinery, a belt might slip and count the number 58 twice, or a truck might go by outside and you'll skip 572. These problems make designing reliable physical machinery very hard.

The result is that software is far easier to design per component than hardware. This is why designers today use software rather than hardware whenever they can. This is also why teams of a few people often develop computer programs of tremendous complexity.

People naively say to me, "If your program is innovative, then won't you get the patent?" This question assumes that one product goes with one patent.

In some fields, such as pharmaceuticals, patents often work that way. Software is at the opposite extreme: A typical patent covers many dissimilar programs and even an innovative program is likely to infringe many patents. That's because a substantial program must combine a large number of different techniques and implement many features. Even if a few are new inventions, that still leaves plenty that are not. Each technique or feature less than two decades old is likely to be patented already by somebody else. Whether it is actually patented is a matter of luck.

The only way a programmer can avoid this mess is by sticking to things that are obsolete. You may not recall the state of the computer field 17 years ago since most people didn't pay attention back then, there were no personal computers. If you were a hobbyist you might get a computer with a few thousand bytes of memory. If you were lucky it might run basic.

This shows another way that software is different, it progresses very quickly. A program three years old is becoming obsolete, and one that's six years old looks Stone Age. A 20-year monopoly for anything in computers is absurd.

In other fields a new technique may require development, building one device after another until you understand how to make the technique work. If a steel part functions badly and you think copper might be better, you can't type "replace steel with copper" and try the new device a minute later. The need to recoup the cost of this development is part of the usual argument for patents.

In software, an individual technique usually doesn't need much development. What we do develop are products that combine a new technique with dozens of other techniques. When a second programmer decides to use the same technique, he will have to do just as much development as the first programmer. Firstly, because he's probably using a different combination of techniques with that new one, and secondly, because the first programmer probably kept the results of development a trade secret.

The patent system is supposed to help by discouraging trade secrecy. In software, patents don't do this. Today, just as in 1980, most developers publish general ideas and keep the source code secret. Here is a copy of a compiler that I wrote with a few friends. It's printed four pages per sheet to make it manageable. This program is mainly the work of four people, another dozen helped substantially, and others occasionally. The two principal developers were not working on this full-time.

This compiler and its output are probably being used on more than a million computers today. Major companies such as Intel and Motorola have adopted it and now add to it. The U.S. Air Force is funding extensions to it. Many widely used systems are compiled with it. Just a few lines of

code can be enough to infringe a patent. This compiler has 10,000 pages -- how many patents does it infringe? I don't know, nobody does.

Perhaps you can read the code and tell me?

(laughter)

I know of one patent it infringes, I found it along with some near misses in a list I saw by luck. I believe I have prior art for that patent, but I can't be sure what a court would say. I don't dare tell you the patent number, because if I were sued, I couldn't pay for the defense. I would lose by default.

An invalid patent is a dangerous weapon. Defending a patent suit typically costs a million dollars and the outcome depends mostly on legal technicalities.

I've had at least one patentable idea in my career, I know this because someone else patented it years later. It's a feature for using abbreviations in a word processor. A couple of years ago, the users of the word processor XyWrite received a downgrade in the mail. XyWrite had an abbreviation feature, the developer removed it when threatened by the patent holder.

They knew about my earlier published work; why didn't they fight the patent? Sometimes a business can't afford to have a lawsuit that will drag on for years. At those times, even if the patent is invalid, you lose.

These patents are invalid because of luck. It was pure luck that these ideas were published by one person before they were patented by another. And it was luck that the ones who published didn't patent instead.

This is an important point: What is patented and what is not is mainly a matter of luck. When you develop a large system and you need to combine a large number of techniques and features, whether or not you can use each given one is a matter of luck.

The carelessness of the Patent Office in dealing with software is well known. So some people assume that if the PTO only did a better job, everything would be okay. They say we that we should wait while the invalid patents spew out, and eventually the PTO will understand software and do the job right.

There are two flaws in that suggestion: The PTO will not do a better job, and that would not solve the problem if they did.

Some years ago a professor I know patented Kirchoff's current law, which says that the electric currents flowing into a junction equal the currents flowing out. He did this to confirm, privately, his suspicion that the PTO could not handle the field of electronics. He never tried to enforce the patent which has since expired. I will disclose his name if you give assurances that he and his lawyer will not get in trouble for this.

Kirchoff's laws were formulated in 1845. If the PTO couldn't understand electricity after a century, how can we expect it to understand software in another decade or two.

(applause)

Computer scientists look at many software patents and say, "This is absurdly obvious". Defenders of a patent system reject our opinion. "You're using hindsight," they say. "You're more skilled than a typical practitioner." What we

consider obvious patents are not errors, they reflect a different definition of "obvious". It's not going to change.

What if the PTO stopped making mistakes and issued no more invalid patents? That would not solve the problem because all new techniques and features, those not known today, would be patented just the same.

Suppose the PTO were perfect, suppose that it is the year 2010 and you're a software developer. You want to write a program combining 200 patentable techniques. Suppose 15 of them are new, you might patent those. Suppose 120 of them were known before 1990, those would not be patented any longer. That leaves 65 techniques probably patented by others, more than enough to make the project infeasible. This is the gridlock we are headed for.

Today's PTO mistakes are bringing us to gridlock sooner, but the ultimate result is gridlock even with a perfect PTO.

I have explained how patents impede progress; do they also encourage it? Patents may encourage a few people to look for new ideas to patent. This isn't a big help, because we had plenty of innovation without patents. Look at the journals and the advertisements of 1980 and you'll see. New ideas are not the limiting factor for progress in our field. The hard job in software is developing large systems.

People developing systems have new ideas from time to time, naturally they use these ideas. Before patents they published the ideas too for kudos. As long as we have a lot of software development, we will have a steady flow of new published ideas.

The patent system impedes development. It makes us ask for each design decision, "Will we get sued?" And the answer is a matter of luck. This leads to more expensive development and less of it. With less development, programmers will have fewer ideas along the way. Thus, patents can actually reduce the number of patentable ideas that are published.

A decade ago the field of software functioned without patents, it produced innovations such as windows, virtual reality, spreadsheets and networks. And because of the absence of patents, programmers could develop software using these innovations.

We did not ask for the change that was imposed on us. There is no doubt that software patents tie us in knots. If there's no clear and vital public need to tie us up in bureaucracy, untie us and let us get back to work.

I'm finished. I hope I didn't exceed my time.

COMMISSIONER LEHMAN   No, you didn't. Thank you very much.

Does anybody else have any questions of Mr. Stallman?

Thank you very much.

(applause)

COMMISSIONER LEHMAN   Did you write this entire 10,000 pages worth of code?

MR. STALLMAN: No. As I said, about four people did most of the work. In what's here, I and one other person did most of the bulk of the work. And then there were like two or three others who did substantial pieces, and a dozen who did significant pieces.

By the way, with this goes another stack -- this tall -- of

machine descriptions for particular target machines, but I didn't include them because they have a lot more different contributors.

COMMISSIONER LEHMAN And what do you do with this? How do you make this available to the public?

MR. STALLMAN: It's on the Internet. You can FTP it and run it. Many organizations distribute it. We also supply it on compact disks.

COMMISSIONER LEHMAN So that's how you make most of your work which is dedicated to public domain as I understand it, pretty much.

MR. STALLMAN: It's not public domain, but that's getting into a digression, it's free software.

UNMIKED VOICE: Can you go back to the microphone?

COMMISSIONER LEHMAN I apologize, I should have asked that.

MR. STALLMAN: I don't want to get into free software because it's a digression I think for the most part, and it leads into an area where I have views that are definitely a small minority's views. What I've said I think most programmers would agree with. I've stayed away from my controversial beliefs.

But, yes, I distribute free software, and because of that I generally can't licence even one patent. Other software developers can muddle through if they have to license a few patents, gets to be enough and they get crushed.

But I can't -- I get stopped dead by even one.

COMMISSIONER LEHMAN Have you been sued yet?

MR. STALLMAN: No, but I've had to stop distributing software because I've seen other people being threatened for the same patents. I didn't wait till I got sued. People often wouldn't actually sue a charity. After all, I am the president of a charity, and they might feel it would look bad to be suing us, so instead they would just sue our users. Right? And how would I feel if I were trying to help the public, giving them something that's a trap, a trap for getting sued.

COMMISSIONER LEHMAN   Have any of your users been sued, of your work?

MR. STALLMAN: They haven't been sued for the programs that we write, but for other free software we use they have been threatened. They received letters from AT&T, everyone using XWindows got threatened by AT&T and by Cadtrack. So even though we didn't write that software, it's an essential part of the system we're trying to build.

COMMISSIONER LEHMAN Thank you very much.

Our next witness is Timothy Casey, Senior Patent Counsel with Silicon Graphics.

MR. STALLMAN: Would you like to keep this? It's an exhibit, we'll offer it to you, but we'll throw it away if you don't want it.

COMMISSIONER LEHMAN It seems like a shame to throw it away, but I think that it's going to be hard for us to take it back to Washington.

MR. CASEY: I'd be happy to recycle it for you.

(laughter)

COMMISSIONER LEHMAN But we can put it in a museum here in San Jose. Well, since it's available actually on the Internet, we can just call it up there, if we need it.

Please go ahead.

--o0o--

TIMOTHY CASEY

Senior Patent Counsel

SILICON GRAPHICS

MR. CASEY: Hello, I am Timothy Casey, Senior Patent Counsel with Silicon Graphics, and representing their views.

Silicon Graphics is the world's leading supplier of visual computing systems targeted for technical, scientific and corporate marketplace. The company pioneered the development of color three-dimensional computing, transforming it into practical and affordable mainstream solutions that improve the productivity and increase operational efficiencies across a broad array of industries, even though as of late, we seem to have gained our greatest notoriety from our involvements in films like Jurassic Park and Terminator 2.

I would also like to point out that the company was originally founded on an exclusive grant of a patent from Stanford University that's since run out, but we did have our basis around the patent.

Although Silicon Graphics designs and manufactures personal computers, workstations, servers and supercomputers based on our own designs for RISC processing technology, a large part of our overall development effort is now focussed on software, including, display, communication, development tools, operating systems, applications and user interface technologies.

Naturally, Silicon Graphics files a large number of patent applications related to both our hardware- and software-based inventions, and has a vested interest in maintaining such protections. But rather than use my time to further expand on the horrors or virtues of software patents, I would rather state that we are spending too much of our time in these hearings I believe, discussing the ill patient and not enough time discussing the disease. And shooting the patient I don't think is an adequate solution.

Software as incorporated into the patent system is not the great villain that many people would like us to see, but rather a misunderstood giant. As I mentioned earlier, most computer system manufacturers today invest a majority of their research and development efforts on software technologies in order to further distinguish their hardware products from their competition. I truly hate to think what would happen to this industry and this nation's economy, if less than half of this development effort was subject to patent protection.

It is also important to consider that in practical terms, software is not really different from hardware. It's just that the Patent Office understands hardware and is better prepared to adequately examine hardware-related cases in most situations. Patents have been granted on transistors, resistors, capacitors, clock circuits, filters, and the like, all necessary building blocks of many electronic designs. But has that seriously impeded the electronics industry? Why is software so different?

Are not most software products composed of basic

elemental blocks of code arranged in new ways to perform new tasks, much like the hardware elements of any modern electronic product? The reason software patents have developed into such a controversial topic is primarily because a number of overly-broad software patents have been allowed to issue. And why is that?

Well, we can argue that the prior art is inadequate, and that is true. And we can argue that the statutory subject matter tests are inadequate, and that is also true. But I would argue that one of the biggest causes is the fact that the Patent Office has not had the most important tool it needs to adequately examine software patents; and that is examiners who have the same fundamental understanding of basic software elements as they have of basic hardware elements.

I find it absolutely amazing that the Patent Office has issued so few overly-broad software patents given the level of training of many of the examiners and the complexity of the application subject matter. While I understand that the Patent Office has already undertaken steps to hire computer science majors in the future to help solve this problem -- which I loudly applaud -- there are many additional measures that can be taken to improve the services of the Patent Office, both with respect to already issued software patents, and any applications that might be examined in the future.

Some of these measures include revamping the re-examination process, so that re-examinations can be used to achieve the goals for which they originally intended. One step would be to make re-examinations an inter partes proceeding. Another would be to make them less expensive, both in terms of fees and the formal requirements of a re-examination request that presently force potential applicants to seek exceedingly expensive professional assistance in order to comply with the regulations.

It may also be in the country's best interest if an amnesty period is implemented over the next year or so during which applicants could institute a re-examination of any software patent, based on new art, meeting the requirements -- which may indeed need some revamping -- by simply filling out a one-page application form and filing a minimal fee, say of $500 instead of the present fees which is well over 2,000, I believe.

Second, instituting some form of prepublication of applications for the purpose of eliciting industry comment, such as publish issue patents on a tentative basis pending the discovery of new art during the comment period.

Third, establish routine industry-supported education programs to provide continuing education for examiners, even up to and including advance degree study for examiners who commit to an extended tenure with the Patent Office.

Silicon Graphics has participated in both bringing people to the Patent Office to do presentations on graphics technologies for the examiners. And we have also hosted a number of examiner groups at our office in Mountain View to attempt to give them additional education on our industry and our technologies. And I applaud the Patent Office's efforts in that area.

Four, employing technical specialists with broad industry knowledge and allowing them to roam between examining groups so they can provide expert assistance when needed

to less highly trained examiners.

Five, allowing examiners more exposure to a variety of technologies instead of pigeon-holing some of them in narrowly constructed examination areas.

Six, providing the examiners with better technical tools, such as network computer systems that allow examiners to do key element searching of both text and graphics on a single screen at the same desktop system that they use for word processing, video teleconferencing, and Internet communications.

Seven, introducing legislation to turn the Patent Office into a government corporation, so that the Patent Office can attract and maintain examiners at competitive pay scales to the industry without being constrained by the Civil Service pay guidelines, and allowing the Patent Office to actually keep and utilize all the money it raises from user fees.

Eight, instituting new limitations on the maximum number of claims permitted in an issued patent, such as three independent claims and no more that 30 total claims, in order to simplify the examination process and reduce the burden of accused infringers who are often forced to prepare opinions on patents with hundreds of primarily duplicative claims.

Nine, instituting per-page surcharges for patent applications with more than 35 pages of text and 10 drawing figures to force applicants to be more succinctly-descriptive of their inventions.

As long as I am on the subject of steps that the Patent Office can undertake to improve its services to its clients, I will introduce two additional measures that should be considered: namely, instituting a new type of expedited patent application, and regionalizing the Patent Office.

Silicon Graphics recently obtained a patent on some technology that was critical to the protection of one part of our systems, and that we knew has already been duplicated by after-market suppliers of such parts. Despite filing a Petition To Make Special with our application and being extremely diligent in our efforts to shepherd the application through the Patent Office as soon as possible, it still took eight months from the filing of the application for it to finally issue as an enforceable patent.

Now, you may think, "Eight months, that's pretty fast". But it would have been done in two or three months had it not been for the fact that it took three months to get from the mail room to the examiner and another four months to get from the examiner to the final print. The examiner completed the entire examination of the patent in less than one month.

In a similar case, we have a patent application on a fairly simplistic mechanical assembly sitting in the Patent Office since July of '91, despite the filing of a Petition To Make Special. During the pendency of this application, that mechanical assembly was copied by a number of other companies. Although this only resulted in a small amount of competition for us in a relatively narrow market for this part, it was a greater concern for us because the knock-off products are not always of adequate quality, it could cause damage to our customers' systems when used, for which we would ultimately be responsible for correcting in order to maintain our customer loyalty. This would not have been

the case had this application been allowed to issue.

The reason the one application took eight months and this other application has taken over two and a half years is because the Petition To Make Special Process only applies to the examination and not the remainder of the Patent Office process. What is needed therefore, is a new type of application that not only gets expedited when in front of the examiner, but throughout the entire process. I would be more than happy to pay a higher fee, such as a 1,000 to $1,500 in filing fees for such an application in these types of situations, because I would surely make that up in outside counsel fees when the attorneys have to relearn the technology two or three years later after we get a final office action.

My last suggestion is that the Patent Office seriously consider working to regionalize the Patent Office. No one in Washington can truly appreciate the difficulties in communicating with the Patent Office from the West Coast. Because of the time differences between the two parts of the country, and the new flexible hour programs instituted by the Patent Office, which I do think is a good idea, there's only a one-hour period each day, typically between 1:30 to 2:30 Pacific Standard Time, during which a practitioner on the West Coast or an applicant can expect to get ahold of an examiner on the telephone. Because of this, it has sometimes taken over a month to arrange a telephone interview with an examiner.

My colleagues on the East Coast however, can call at much more convenient hours, or even walk over to the Patent Office for an in-person visit, something which would cost me well over a thousand dollars to attempt. Given these restrictions, I hate to imagine what the independent inventor or startup organization on the West Coast thinks of our patent system.

Some other benefits of a regionalized Patent Office would include: An ability to draw from a larger pool of potential examiners; new economic growth in the parts of the country selected for the new Patent Office sites; and greater public accessibility to the Patent Office records and examiners, as well as greater exposure for the examiners to the relevant industries. It would certainly cut down on your costs of sending examiners across the country.

I hope these suggestions prove to be useful guides to the Patent Office and as an invitation to the Patent Office to consult with West Coast companies and practitioners and applicants for addition solutions that could be implemented by the Patent Office to resolve our present difficulties, but also to raise the Patent Office to new heights of service.

Thank you for this opportunity.

COMMISSIONER LEHMAN Thank you.

Do you think it would be a good idea, for example, to have Group 2300 located here in this area?

MR. CASEY: I would imagine that a majority of their clients are in this area so it would probably be a good idea, or at least some portion of them.

COMMISSIONER LEHMAN There were some reactions I had to certain things that you said, but I don't want to take away from some of the other -- we are already I think making changes along the lines of some of the things you've recommended, and certainly part of it has to do with our

automation system and other reforms that we are making. But thanks very much for your help.

MR. CASEY: Thank you.

COMMISSIONER LEHMAN Next I'd like to call Robert Sterne of Sterne, Kessler, Goldstein and Fox.

--oOo--

ROBERT GREEN STERNE, ESQ.

STERNE, KESSLER, GOLDSTEIN & FOX

MR. STERN: Good afternoon. I am Robert Greene Sterne.

And my testimony represents my own views as an attorney in private practice with over 15 years of experience representing almost exclusively U.S. companies in high technology electronic and computer technology. Based on my experience in the trenches, I believe that U.S. companies of all sizes, particularly startups and those creating leading edge technologies are served best by an intellectual property system that provides a broad scope of protection and strong enforcement remedies.

Having heard much of the testimony in these hearings, I would like to state that I agree with those speakers who have stated that technological innovation is fostered by a broad scope of intellectual property protection and by strong enforcement of such exclusive rights. I don't want to replow this ground today. Rather I would like to focus on a few points which I believe need further discussion so as to round out the record of these hearings.

First, from a purely selfish professional viewpoint, and I say this purely -- patent attorneys benefit financially from the uncertainty that exists concerning patent eligibility for emerging electronic technology, and from the complexity that the present patent rules create. So I ask you all to please make the system more precise and simpler for everyone.

Second, the panel seems to find war stories helpful. Let me give you one that I am involved in concerning a startup that is in the process of raising 100 million dollars. Patent applications are being written and product clearance studies are being done which are absolutely critical to this financing. The venture capital people would not feel comfortable in investing their money if they did not believe that they were free of infringement and had an excellent chance of obtaining broad patent protection for their innovation.

Now, this is not an isolated event. More and more, the financial community is requiring that the intellectual property portfolio of a startup electronic company be sufficient to provide a proprietary position and be free of infringement problems in order for necessary capital to be raised. As you know, startups and companies in emerging areas of electronic technology, Mr. Commissioner, create a disproportionate number of new jobs in this economy. We need to protect this process through a strong intellectual property system which will encourage investment and the creation of real wealth.

My experience is that without strong intellectual property, the business community will invest in less risky ventures outside of high technology electronics.

Third, the case law and the Patent Office position concerning so called "mathematical algorithms" which, if found, create nonstatutory subject matter status for otherwise inventive

electronic technology, frankly, is illogical and pernicious. No one, and I repeat no one, in my opinion, either inside or outside the Patent Office can draw the bright lines that the rest of the public believes should be capable of being drawn concerning patent eligibility as it relates to mathematical algorithms. This uncertainty is unnecessary and totally unsatisfactory.

For example, when I debated your solicitor on this subject before the Maryland Patent Law Association last October, he stated that the law of mathematical algorithms was so complex that even he had to review the cases each time he had to deal with the issue.

This uncertainty hurts innovation in this country.

On a substantive level, the alleged distinction between patentable algorithm inventions and unpatentable mathematical algorithm inventions makes no sense in reality, Every physical system, I repeat, every physical system and process can be represented mathematically. In fact, that is how physical systems and processes are modeled today in all industries using computer-aided design and manufacturing tools. To say that a product or process made by man is not statutory merely because the label "mathematical algorithm" is stuck to it, begs the question completely. Furthermore, all one has to do is pick up any two technical dictionaries and one will find that there is no agreement as to what the term "algorithm" and what the term "mathematical algorithm" each means.

So in addition to not reflecting what happens in the technical world, there is the added problem that there is no definitional certainty concerning algorithm and mathematical algorithm.

The Patent Office has a duty to discharge its constitutional mandate of promoting progress in the useful arts by making sure that the definition for statutory subject matter for electronic technology encompass anything and everything under the sun made by man in this technical area. Otherwise, critical emerging areas of technology, such as digital signal processing, voice recognition, computer graphics, compilers, multimedia, virtual reality, handwriting analysis, encrypted communications, and information retrieval, just to name a few, will be denied patent protection despite their innovation, because a patent examiner or an infringer will be able to stick the label "mathematical algorithm" on the otherwise patentable invention.

You, Mr. Commissioner, must be make sure that the people reporting to you seek broad interpretation of patentable subject matter in this electronics area like that being given in the biotechnology and other emerging areas of technology.

Fourth, I often say that electronics is applied functionality in the electronic domain. Implementation of this functionality is merely a design choice in most cases in terms of hardware or software. The choice of how to implement this inventive functionality depends on many factors, but it is the functionality that is the invention, and one should never forget that.

We should stop the endless debate of saying that hardware inventions should be treated differently by the Patent Office than software inventions. The alleged inventive distinction between hardware and software is ludicrous to those in the electronics industry.

Furthermore, we have the same definitional problem that we had with algorithm and mathematical algorithm. No one can agree upon the definition of what is hardware and what is software. The reason for this is simple. As the technology evolves, the blurred boundary between hardware and software implementation changes and shifts.

Fifth, opponents of so called "software patents" argue that the patent of software operates to remove well-known software inventions from the public domain because the Patent Office does not have an adequate database to properly examine them.

The PTO needs to continue to enhance its database in this area of technology as well as in all emerging areas of technology such as biotechnology, but denying patent protection for software inventions due solely to the database inadequacies is akin to throwing the baby out with the bath water. The examination process needs to be improved rather than restricting statutory subject matter for computer-related inventions.

Now, I would like to offer you a ready-made solution which is Rule 56. Under Rule 56, applicants must provide the Patent Office with the best art of which they are aware. Often, this art submitted under Rule 56 includes nonpatent literature. This nonpatent literature in most cases is better than the patents contained in the Patent database, because patents by definition are several years behind the technical literature. Thus, the Patent Office can significantly enhance its search database by feeding in the technical literature obtained under Rule 56.

Six, you inquired in Part A about claim formats, and I will address this in detail in my written submissions.

Let me make just two high level comments: First, claiming should be flexible. The goal should be to define the invention so as to fully protect the inventor while providing the precision that the public so critically needs in determining whether or not they are infringing. The present rigidity often exhibited by the Patent Office in terms of specialized claim formats in emerging electronic areas prevents one or both of these goals from being met. Let's stop putting form over substance, and let's start putting effective claim drafting back into the picture.

Let me give you just one example: The so-called computer program product claim is critically needed so as to allow the patentee to charge the infringer with direct infringement where only software is being sold on media or being transmitted electronically over networks such as the electronic superhighway.

The Patent Office should be seeking ways to have these types of claims made permissible, rather than engaging in a bureaucratic exercise that is tantamount to a war of delay and attrition in denying these types of claims.

Seventh, your twenty-year term proposal is excellent. But it must be accompanied with a rock-solid commitment from the Patent Office to significantly lower the pendency period of patent applications. Leading edge electronic companies in this valley and elsewhere now operate on product development cycles at 6 to 9 months. That is the time it takes from coming up with an idea to releasing a product into the marketplace.

Our firm routinely prepares over 200 U.S. patent applications on complex electronics each year. And we know from experience that it takes two to three years to typically get these applications through the Patent Office.

Mr. Commissioner, that's too long. I predict that if we cannot get pendency period down, the electronics industry will pull away from the patent system because it will take too long for them to get protection.

Finally, the patent system is of enormous benefit to the electronics industry in encouraging and protecting innovation. I think that the system needs to be worked on and improved, but I think overall everything is working out well. It's just that we need to make the system in terms of eligibility more precise.

Thank you very much.

COMMISSIONER LEHMAN Thank you very much.

Next I'd like to call forward Victor Siber, Counsel to the IBM Corporation.

--o0o--

VICTOR SIBER

SENIOR CORPORATE COUNSEL

IBM CORPORATION

MR. SIBER: Good afternoon, and thank you for the opportunity of allowing me to deliver my comments this afternoon.

I am Victor Siber, Senior Corporate Counsel for the International Business Machines Corporation. My comments do represent the views of IBM.

The IBM Corporation spent 6.5 billion dollars on research and development in 1962. In that same year, over 22,000 software programmers were employed by the company, and we sold approximately 18.5 billion dollars of software products and services. So obviously, we are intensely interested in the subject matter of these hearings.

We protect the detailed expression in every one of our software products by copyright. And approximately 3 to 5 percent of these programs contain new and unobvious functions that are protected by patent. Patent coverage on these inventive functions protects our investment, gives us important business leverage, as well as access into foreign markets.

I understand that this hearing was scheduled, in part, to address questions raised by some broad software patents that have recently issued. These patents are alleged to cover old processes. The response by some is to call for changes in the law to prevent the issuance of patents in this technology.

In the short term that would hurt the U.S. computer industry. And in the long term all industries that use computers to gain competitive advantage will suffer.

The argument over these controversial patents is not that they cover unpatentable abstractions. These patents cover quite useful functions, which others in the industry want to use. The issue, and the key to the controversy is something quite different: whether these patents cover truly new and nonobvious functions and thereby add something to the useful arts. In the final analysis that is the job expected from the U.S. Patent and Trademark Office. And it is a job that is doable.

The United States was the first and continues to be the global leader in computer hardware and software technology. This technology is important not only in itself, but as a driver of innovation in other fields. And today, it is facing stronger and broader global competition. As the industry matures and competition from overseas increases, patents will be the key to protecting the most valuable US-originated innovations.

You have posed a number of questions at the today. Beginning with Question 3 relating to altering the standards for patent eligibility for software-related inventions, I want to make it clear that we completely oppose such a proposal.

Going into the next century, the key inventions will be in information processing. Altering the standards for patent eligibility for software-related inventions will shift investment away from this area.

The purpose of research and development in any technology is to gain an advantage over your competitor. But if your competitor can legitimately copy the fruits from your R&D and can create a product that can compete head-on with your product while you are still trying to build a market for the product, then you've lost.

The long term value of R&D in the marketplace is in the new functions implemented by software. If such new functions are protected, investment flows to the industry. If not, investment will dry up.

There are several other points I want to make on this issue. We can't divorce computer program-related inventions from computer hardware and other microprocessor inventions. The overlap between the two is so great that cutting back on one automatically cuts back on the other.

An alteration in the standards for patent eligibility will also put the courts and your office in a quandary. That is because computer program-related inventions can be implemented in either hardware or software. Applicants will simply cast their patent claims in terms of electrical circuitry. And if you limit claim coverage over computer program implementations of circuit inventions, you will turn electrical patents into nullities, because the circuit functions can be implemented by a programed computer.

Furthermore, if process patents cannot reach computer implementations of the process steps, there will be a negative impact on every industry that uses computer-controlled industrial processing or uses microprocessors in their products.

From an international standpoint, a cut-back on patent eligibility for computer program-related inventions sets a truly unfortunate precedent for the developing world. It would violate GATT and NAFTA provisions that prohibit discrimination of patent eligibility based on technology. The biotechnology area provides a graphic example of what happens when countries limit protection for a technology. Leading European companies working in this area of technology simply moved their biotech R&D operations to the United States.

Concerning the issue of sui generis protection, replacing patent protection for new and unobvious program and process functions, and replacing copyright protection for the original expression contained in computer programs with

some form of sui generis right would be devastating to the industry.

In a single act, U.S. industry would unilaterally be disarmed relative to our competitors in Japan and Europe. We would lose patent priority rights in the 114 countries of the Paris Convention, and we would lose the copyright protection automatically afforded American program works in 102 countries as required by the Berne Copyright Convention.

As you know, the Trilateral Work Studies conducted jointly by the U.S. PTO and the Japanese Patent Office and the European Patent Office, concluded that the standards for patentability in the area of software-related inventions are generally the same except at the far margins.

The Japanese Patent Office has published for opposition in the last six years approximately (45) patent applications on software-related inventions, while the European patent office has issued approximately 2700 patents since 1980 which have resulted in multiple European member country patents.

The majority of these software-related patents in the European Patent Office and the great majority in Japan are issuing to non-U.S. companies. Without U.S. patents of a reasonable scope to bargain with, U.S. companies will potentially lose access to those markets.

With the introduction of a sui generis system, there would be --

COMMISSIONER LEHMAN Can I ask a question? Why do you think that is, that the majority of those patents -- we issued 8,600 patent this last year in the United States, so clearly we've got 8,600 people, and I think I cited those statistics before, two years ago it 6,500 or something like that. Clearly, we have the inventions, but people aren't seeking, American's aren't seeking patent protection in those foreign markets, why that is?

MR. SIBER: I think there's a questions of maturity of the industry, many smaller companies are not familiar with the process of filing overseas, possibly may not be able to afford it or feel they can't afford it.

Secondly, it is natural, particularly in Japan to have a large number of Japanese patent holders where the filings of large Japanese companies is very high.

COMMISSIONER LEHMAN So, but IBM, I assume does file in those.

MR. SIBER: Yes, we do.

COMMISSIONER LEHMAN And pretty extensive. I would assume when you file a patent application here, you file there too, since you're a worldwide company.

MR. SIBER: Correct.

COMMISSIONER LEHMAN Is that generally true? Are there situations where you don't file there and you would file here?

MR. SIBER: We do not file a counterpart patent in Europe and Japan for every patent that we file in the United States. It is selective, it is a selective process. We too have budget limitations, even though we just turned a substantial profit last quarter.

(laughter)

COMMISSIONER LEHMAN Sorry, to interrupt.

MR. SIBER: With the introduction of a sui generis system

there would be an extended period of uncertainty, the bane of businessmen and investors, as the body of case law were developed interpreting the new sui generis law and determining its scope.

Additionally, there would be a complete absence of effective international protection for the new right. The internationalization of a sui generis law would require a sui generis treaty. Such a treaty would be negotiated without any of the international consensus on what sort of protection regime would be appropriate. As you know, the primary multilateral treaty to date for a sui generis right, The Treaty on International Property in Respect of Integrated Circuits, known as the Washington Treaty, was strongly influenced by developing countries that were hostile to IP protection generally. The result was a treaty so flawed that not one single major chip-producing nation supported it.

The only remaining way to internationalize that sui generis chip-protection right was through reciprocity. But reciprocity has it rewards and its vices. The European community now points to the reciprocity provisions in the U.S. Semiconductor Chip Law to justify its discrimination against America's authors through the reciprocity provisions in the E.C. Directive on Copyright Term Extension, and for denying U.S. authors the benefit of unfair extraction right in the draft E.C. Directive on Database Protection, and in refusing to grant movie and music producers and authors their fair share of the blank tape and movie levies.

Finally, regarding the series of claim format examples in Question 1, we believe that the format of the claim should be viewed from the perspective of patentable subject matter. If the claim as a whole is directed to a machine, article of manufacture, or a machine-operated process, then subject matter eligibility should not be an issue. The focus of the examiner should be on the normal statutory tests of novelty, nonobviousness, and utility of the claim as a whole. And to insure compliance with the formality requirements of 35 USC 112.

Specifically, a claim directed to an article of manufacture comprising computer-usable medium and a plurality of computer-readable program code means for causing a computer to effect a plurality of specific and interrelated functions should be eligible for patenting under the statute. Such an article of manufacture constitutes a machine part that is commercialized separately. And it should be protected separate, like other machines parts, if it meets the statutory test of novelty and nonobviousness.

In summary, the patent system is designed to instigate the invention of new nonobvious and useful functions that add to the arts. The key to seeing whether the system is working is to see if there is strong competition in the marketplace; to see if new products are introduced in the market on a regular basis; to see if employers and investors vote for the system with their pocketbooks by funding new development work.

As for my company, we rely heavily on the patent system to protect our investment in new products. And we are negatively impacted probably more than most by poorly-examined patent applications. Thus, we want the issuance of poorly-examined patents curtailed. And that is clearly doable if the PTO would invest in hiring and training the best possible examiners, as well as in the creation of an

adequate database for software prior art.

Thank you for the opportunity. I would be delighted to answer any questions.

COMMISSIONER LEHMAN Thank you very much, Mr. Siber. Is there anybody else that wants to ask something?

Thank you very much for joining us today.

Next, I would like to call on Mr. Ewald Detjens, CEO of Exemplar Logic, Incorporated.

--o0o--

EWALD DETJENS

CHIEF EXECUTIVE OFFICER

EXEMPLAR LOGIC, INCORPORATED

MR. DETJENS: Thank you.

My name is Ewald Detjens. I am with Exemplar Logic. Opinions being expressed are both mine and those of the company.

Thank you very much for holding these meetings in the first place. It's a good opportunity to express the views of a large group of people that I don't think have been really addressed at this point.

I'd like to go through a little bit of the background of myself and the company, talk a little bit about the history of software and how the patents have crept into it, then talk about the effect on my industry and talk about the sort of conflict I see, and finally conclude.

And to not leave you in any suspense, I'll give you the conclusion right up front, I'd like to make life easier for you, I'd like there to be no software patents at all, and to back off from the ones that have been issued. I think that the copyright/trade secret law as is is totally effective for my industry and has been working quite well for us.

Exemplar Logic is a startup, it's totally privately-held, self-funded, we've been boot-strapped up. We're profitable, it's one of the things you can still do in software with very minimal money, is to get a software business going and create a new product in an area that didn't exist before.

We do logic synthesis for field programmable gate arrays, a very dynamic and emerging segment of semiconductor marketplace right now. I have been active in a number of trade organizations, IEEE, ACM, American Association for the Advancement of Science. I have written a number op-ed pieces on the area of software patents, and those have been in electronic magazines like EE Times, ASIC and EDA. I've gotten a lot of feedback from people in the industry about this already.

In terms of the history, I think it's pretty clear that the legislative intent had been to have software be copyrighted and not patented. And certainly, the Constitution does not provide any mandate for patents for software any more than it provides a mandate for patents for literature.

The courts have sort of slowly changed this over the years as the whole issue of software being integrated into physical machinery created problems for some people. I think those problems can be addressed without getting to the whole general issue of software patents. I think you can still protect that machinery with the special-purpose software that's in it without affecting the rest of the software industry that's been extremely successful given our current

structures.

Now, as I've mentioned, it seems like some of the players here, the programmers in the world have not been very informed of this process as it's been happening. It's been decisions that are out of the realm of your typical software engineer, and it's been in the realm of the lawyers. And I was almost a bit concerned to see the list of people speaking here today being very dominant on the side of lawyers versus the software people.

I think that certainly the fury you're seeing over the Compton patent is just the tip of the iceberg, it's a much greater percentage beneath the surface. The issues revolving around software patents will be affecting more and more people as we go on.

In particular, it has some definite effects on my industry. It's sort of like changing horses in midstream, so it's very difficult for us to have things covered both by patent and software, and know exactly what we should be doing, and to be left with the situation of prior art not in place and known by the wider public.

I think that if you do have the necessity for doing software patents in the future, it will certainly slow down the pace of innovation. It's going to make programming far harder than it is today. It's going to drive the cost of software up. It's going to lock us into old systems for far longer than we are. A sort of analogy is, we would be locked into DOS for many years, into the next century before Windows or something could come out so we could get rid of some patents so the next generation software could appear.

So some of the people looking at the short term interests in patent -- I mean certainly, a patent is a good thing because it provides you an unfair advantage, but you look at the overall industry, I think it's going to provide very detrimental effects compared to the short term effects some individuals might see for their company looking in the narrow focus.

Some of this is kind of hard to explain. I mean, I think there have been other people that have gotten up in front of you and explained how difficult software development is, and certainly the state of the art hasn't advanced a lot in the last few years as far as just pure development is concerned. There aren't any magic bullets here going forward. The systems are getting more complex and we don't have any tools to deal with the larger complexity that faces us. So it's sort of a bleak view for the programmer out there.

COMMISSIONER LEHMAN What do you think was the single most important magic bullet in this industry in the last 20 years? Was there ever such a thing? Was there ever an innovation that was just so fundamental that --

MR. DETJENS: Well, it's kind of hard to say. There's been a number of things promoted as being a magic bullet. Certainly, object-oriented programming, that paradigm has been one recently. I believe it's a very effective method of doing programming. I don't think it's something that gives you an order of magnitude improvement. Maybe it makes it twice as effective or something. And yet, we're seeing demands on us for providing systems that are 10 times more complex, detailed than they have been in the past.

And the average size of a program is going way up in the number of bytes and the deliverables. If you look at it, it's incredible. What used to be delivered would fit on one little

floppy. And now, for your average word processor, they include billions of files with it that have all kinds of added functionality for a huge realm of people.

CD ROMs are sort of an expression of this. You see many people switching to CD ROMs. You can't deliver software effectively on floppies any more, nobody is going to take 40 floppies and be inserting them just to get going with their program.

So there doesn't seem to be any magic bullets really on the horizon, and we're still programming almost as ineffectively as we did in the early '60s in some sense.

The best thing that has happened has not been a software thing, but it's been the fundamental hardware has gotten better, and that's helped us out. We can do programs, if the hardware speeds up by 10 times, we can deal with writing a program that's 10 times slower in getting it to market faster.

COMMISSIONER LEHMAN Actually, that's what I thought your answer would be that the hardware, that the magic bullets have tended to be more in hardware. And hasn't hardware classically been protected by the patent system, and most people aren't proposing that that be changed?

MR. DETJENS: No, no. I don't think any software engineer would venture into the realm to say that patents should be backed off for any other type of thing, certainly not computer hardware at all.

COMMISSIONER LEHMAN I didn't want to get you off the track too much.

MR. DETJENS: No, that's okay.

So one of the interesting things in terms of the op-ed pieces I've been writing, too, has been the feedback. And so far it's come down a hundred percent, a hundred percent. The people that were pro software patenting were the programmers out in the profession and the people that were -- excuse me, did I say that correctly? The people that were against patents were the software people, and the people that were for patents were the lawyers.

And this to me is amazing. I thought it was sort of benign at first, but the question is who benefits from this? And the lawyers are driving it certainly. You can see that the programmers are getting more and more annoyed by this kind of thing.

I don't think that Exemplar would really exist the way it does today, if we didn't -- if we did have software patents in general use, I don't think that you can boot-strap a company in our industry any more that way. You would need far more funding to just prevent against somebody suing you.

Certainly, one of the biggest problems for us is, even if it's an invalid patent, the onus is on the infringer, which means that somebody has an invalid patent and they tell me that I can't proceed with my program. It's up to me to dig out the prior art, go through a very expensive court procedure to do something about that. So that's going to provide just an incredible negative impact for the sort of garage-shop industry where a lot of new types of companies have been spawned.

So just in summary, I would like to say that the preference of the majority of the programmers out there in the world that I have been talking to is that we should not have software patents. The copyright/trade secret are working and are excellent methods for software protection.

Thank you very much.

COMMISSIONER LEHMAN Thank you very much.

I would like to next call on Michael B. Lachuck, of Poms, Smith, Lande and Rose.

COMMISSIONER LEHMAN We know when you have four last names you must be a lawyer. Michael Lachuck of Poms, Smith, Lande and Rose.

--o0o--

MICHAEL LACHUCK

POMS, SMITH, LANDE & ROSE

MR. LACHUCK: Pick four, right.

Good afternoon, Mr. Commissioner and distinguished panel.

I am a partner in the intellectual property law firm of Poms, Smith, Lande and Rose. And I will be speaking today solely on my own behalf based on my own experience and those of some of my partners.

I would like to depart at least a little bit from some of the comments that you've heard from so many other patent lawyers that have had an opportunity to speak at these hearings. Virtually, every one of those attorneys has said to you that they've been able to obtain patents for software-related inventions, and frankly, my experience has been no different.

I would like to point out that what this means is that irrespective of what the Patent Office decides to do, clever patent drafters will always a way to disguise a mathematical algorithm or some other form of software-related invention either as a piece of hardware, or as some form of methodology having significant post solution activity necessary to get a patent.

The problem is one that simply isn't going to go away, it can to a certain extent be curtailed. What I would like to do is make some suggestions on how the Patent Office might be able to change its practices to alleviate this problem.

There are two issues I would like to address: One, is specifically how the Patent Office might change its practice, and the second is how the Patent Office might propose legislation to effect remedies available in patent litigation.

Speaking to the first issue of patent practices: at the office, several people have come up here and suggested that applications might be laid open during the pendency of the application and before a patent issues.

I would like to argue against this practice for three specific reasons: the first reason is that this practice would shift the fundamental contract between the patentee and the U.S. government concerning the grant of a patent in the first place. Currently, the application is obliged, or rather the patentee is obliged to provide a specification having an enabling disclosure, one that teaches the public or those reasonably skilled in the relevant arts how to make and use the invention. In exchange for this, he's awarded certain rights and remedies under the patent laws. Now, if the application is simply laid open, the same standard or obligation of teaching merely avails the application of a possibility of obtaining a patent at some later date based on the presence of prior art that they, themselves, may or may not be aware of.

Secondly, I would like to most strenuously point out that if the practice of laying open an application for public comment were to follow current practices of re-examination procedure, it simply wouldn't work. The problem is that right now the public has only one bite at the apple. They can submit prior art of a certain form, prior patents and printed publications, and they are allowed to comment exactly once on the relevancy of this art to the application.

Thereafter, the process again becomes an ex parte process where the applicant's advocate, some skilled patent attorney is able to argue with the examiner over whether or not that art is in fact relevant or does make the invention obvious. The ant-like persistence of patent attorneys is legendary, as even the well known jurist Learned Hand once noted. This process of allowing the ex parte communication between the patent applicant's advocate and the examiner, is a decidedly risky proposition.

What I would like to propose instead is that the re-examination process be modified to allow representatives of the public or interested members of the public to participate in that further communication that takes place between the applicant's advocate and the Patent Office examiner, more or less in the manner that litigation motions currently proceed; that is, the public would be afforded an opportunity to submit prior art and comment on its appropriateness or applicability to the application. Thereafter, the applicant's advocate would be afforded an opportunity to comment and then the public or their representatives would be afforded an opportunity to respond to the applicant's comments.

I am concerned that if this isn't done in some fashion or other, then the suggestions that have been made concerning the submission of prior art to the Patent Office, particularly in this area where there appears to be a dearth of prior art, isn't going to change very much.

Under current circumstances, the district court judges typically afford great weight to the decisions of the Patent Office examiner. Unfortunately, the public representatives are never provided an opportunity to debate or address the subsequent arguments that are made by an applicant's representative. The result is that very few litigators suggest filing a re-examination application. The best prior art is lost, and the patent is simply strengthened.

Turning to the subject of patent remedies. I would like to suggest a drastic reduction or elimination of injunctive relief in patent enforcement litigation. The threat or promise of injunctive relief drastically distorts the economic value of a patent and the attendant costs of litigation. It's been my personal experience, and the experience of every patent attorney I have ever talked to that the cost of litigation is directly related to the potential liability exposure of the defendant.

Patent infringement litigation, because of the threat of injunctive relief typically becomes a venture company-type lawsuit. The client does insist on a most vigorous defense proportional to the risk that they face. If potential liability exposure for patent infringement litigation were reduced to a few thousand dollars, the cost of litigation would drop to some small fraction of that value.

The basis for award of injunctive relief is based solely on

tradition, dating back to old English law, which now, in the hands of a patent owner who frequently does not market any product, is ludicrous. The effect of patent infringement litigation with the threat of injunctive relief is to waste a tremendous amount of American wealth and resources. It is time to bring this practice to heel.

(applause)

Thank you for the opportunity to speak to you today, and I thank you also for your taking the time to personally come out here and hear from so many representatives of the industry.

COMMISSIONER LEHMAN Thank you very much. I should add that I've been waiting for somebody to make the point that you did. And I am somewhat surprised that it has not been made before, and that is that there is a deal in the patent system, and that is that one does not really give up their trade secrecy rights until they are certain that they've got the patent. And certainly to go to a prepublication system would create a hiatus in which that bargain in effect would be jeopardized.

But I do find it interesting that thus far, you're the only person who's raised that. That by and large the wave of testimony has been, I gather, that it's worth taking that risk in order to improve the availability of the prior art to the patent examiner.

Anyway, thank you very much.

MR. LACHUCK: I would like to address that one comment. I suggest that if the statutes were amended to provide that any patent infringement litigation were automatically stayed pending re-examination, then the need to have post issuance examination or comment by the public, the need to have the application laid open would be alleviated.

The current reasoning I think that exists for people wanting to have an application laid open during the pendency of the application is to avoid the situation where the trial court judge decides to continue with the litigation while the re-examination takes place. And as things currently stand, there's also the threat of preliminary injunctive relief, so if you get rid of that threat, it would probably be a more practical alternative to allow the same public comment through re-examination.

Thank you for your time.

COMMISSIONER LEHMAN Thank you.

Next, I'd like to call on Gregory Aharonian, from Source Translation and Optimization.

--oOo--

GREGORY AHARONIAN

SOURCE TRANSLATION AND OPTIMIZATION

MR. AHARONIAN: I run a consulting service up in the Boston area, dealing with software re-use technology transfer, and more recently software prior art and software patenting.

And to summarize how I feel with what's been said today, I am for software patents, for better prior art, open re-examination, and I'm against prepublication.

No, I am not a patent lawyer, or have any software patents on file.

The chief asset of my consulting business is what I consider

to be the largest software prior art database in the country. I have information on over 15,000 government, corporate, and university research programs, 5,000 patents, and over 100,000 journal, article, technical reports and books and other such things in which software technology might be described. I monthly monitor the output of about 150 corporate academic and government research labs and about 250 journals.

The problem of software prior art is very, very nontrivial, and I think that's why so many other government agencies have had problems over the years dealing with the similar issue of how do you track all this country's software technology.

I have many more comments on the art of software prior art, and I'd like to speak about them in February hearings in which it's more appropriate. What I would like to speak about now is kind of observations on having examined so much software over the years on a few of the issues that have been brought up today.

One, is I think it's going to be very difficult to change the rules to deal with software patents. There's already a current set of statutory guidelines that are pretty well-reasoned, pretty consistent, certainly comprehensive, if you read all the court cases. But I don't feel that they're working well, for the following reasons: I've examined over 5,000 software patents as part of my technology transfer business, and I've seen a lot of very trivial software concepts actually getting a patent. I've seen many software patents with very broad claims, I've seen things that are as close to a pure math algorithm as possible with maybe one claim in there for a piece of hardware, and I see more and more business practices being awarded software patents. And in some cases, I see pure source code being patented. In 1992, the Air Force got the patent for the difference in source code between two versions of a public domain program at Ames. And I consider that to be fairly trivial.

If you try to come up with more rules to guide the process of awarding software patents, I think you're just going to come up with more ways for patent lawyers to get around the rules. It's tough to treat software any differently than any other technology, and I think you have a lot of problems.

So in general, based on just examining a ton of software in the past 10 years, I'm not sure there's much anyone can do in terms of trying to come up with more rules, it's very difficult.

And on the practical side of the patents I've examined to date, I could probably successfully challenge 25 percent of them on software prior art and a few related issues. There's a lot getting through that should not be awarded.

Of course, my phone is not off the hook, asking for my services to challenge software patents, so I still don't think to date it's a big problem. I think it's being exaggerated because of stunts like the Compton incident, which is great for PR, but not much more.

The second issue I would like to address is that of what one of the earlier speakers, Robert Sterne, testified that there's really no difference any more between hardware and software, and that if you try to change the software patent rules in isolation without treating the hardware patent rules, you're not going to do anything. You're going to leave a big

loophole for people to get around the hardware rules by just doing things over on the -- or getting around the software rules by doing things over in the hardware world. It's even happening today.

Existing technology now in the market where I can take a circuit schematic, which anyone would consider a piece of hardware for the most part, automatically convert it into a computer representation language and then convert it once more into an algorithm which I think most people consider to be software.

Similarly, there's other technology out there that let's me take an algorithm written to a traditional language like Pascal or C, covert it into another intermediate language and feed that into a hardware design tool, and get out an integrated circuit. So here I'm starting out in the software world and ending up in a hardware world. To me, it becomes impossible to find what software or hardware is.

And quite recently, in fact, last week a company in Germany announced a tool that integrates -- and that's the overhead I have up there -- computer-aided software engineering tools which is the domain of the software world with hardware design tools, so that within one tool set I can type algorithms, draw circuits and go back and forth and not really care at all, the computer will take it all and account for me.

And with such a tool I can design a new device, and with a cleverly-drafted set of claims, where I have a broad independent claim that talks about systems and methods and things of that nature, dependent claims somewhere that actually mention hardware and software, I can protect infringements in both worlds with one patent. So to make these distinctions between hardware and software, I think is a mistake.

And in fact, there are a few of these design tools in which it should be possible within a year or two to not only allow the user of the tool to generate either an integrated circuit or a computer program, but at the same time a patent application. And if you want to have a million software engineers and electronics designers having an automated patent generation tool on their hands, it's going to become a possibility quite soon. And I'd hate to think of the headaches you're going to have then.

One other thing, the equivalence of hardware and software also complicates the issue of software prior art, because if these mappings are true and you can go back and forth between the two domains of hardware and software, and conceivable someone's circuit schematic somewhere could serve as software prior art in another case.

And I track both software and hardware, so it's no big difference to me, but if the Patent Office intends to seriously treat the problem of prior art, it will not be able to do software prior art separate from hardware prior art. It is all one field of computing devices.

Now, one thing I would like to suggest is currently I publish over the Internet a news service dealing with patent information. Each week I mail out to about a thousand sites on the Internet the titles and numbers of the latest patents coming out of the Patent Office. In Boston there is an APS terminal that I can use for free, which is very nice. And just once a week I go down there, and dump off a couple files

worth of data, broadcast it out over the Internet. It's a very well-received service, I offer it for free since it doesn't take too much of my time. And the main demand is for people who are trying to find out more information on software patents.

At best, if someone actually cared to make an effort to find out what was being patented in the software world, you would go to a local patent repository and use one of the CD ROMs, (Casus Bib) or something, and those tend to be five or six months out of date at the typical repository. And people tend to want more recent information.

So the various calls to get the APS system on the Internet will be very well-received in the Internet world, and there are many out there glad to help out with the process.

So In general, I don't think there's much you can do to change the rules of dealing with software in isolation. The current rules are a good set of rules. I'm not a patent lawyer, but rules are rules. I don't think you're going to be able to do too much better. You can change some of the procedures, and many people are calling for that. But dealing solely with statutory type things, I don't think it's going to have much of an effect.

And certainly, if you had seen a lot of the software patents I've examined over the past four or five years, it's hard not to conclude that they just don't work. I think the open re-examination process will help, but I think you're grossly underestimating the amount of paperwork and headaches that that's going to entail.

If everyone in Internet, at the request of someone, decides to forward their pet document to the Patent Office, I mean, you'll get a million people sending in something that pertains to one particular patent. You could fill up this hall many times over for each case.

That's all that I have to say.

COMMISSIONER LEHMAN Thank you.

One thing actually I had meant to ask one of the other witnesses, but since you indicate that you see a lot of patents issued that you're quite convinced would not meet the nonobviousness test -- our decision to reexamine the Compton case was a very unusual one, but the Commissioner does have the power to order re-examination himself. What would you think of as one method of attempting to clean the files of allegedly nonpatentable incorrectly issued patents if we had some kind of a program where maybe we work with people like you and attempted to identify some of these patents that we issued, we ordered our own re-examination?

(applause)

MR. AHARONIAN: Well, I think it's a great idea, and certainly would love to bid on a contract to do that. There are two practical problems I see with that.

The first is, I'm actually an inventor in the world of electric power equipment. That's how I got into a lot of this patenting stuff. I don't think the problems of prior art in the software world are any worse off than in some of the other fields.

For example, many of the high temperature superconducting patents for devices being issued today would be invalidated by low temperature superconducting device patent

applications dating back to the '30s and '40s. Most of those patents no one knows about anymore. You really can't get at them through APS or any of the CD ROMS or anything else. They're literally lost unless you go look for them. But they couldn't validate many of the new high temperature superconductor device applications. So I could argue that if you're going to consider doing that in the world of software, you could do it in all the fields because they all have that same problem. And most information dating back before the early '60s or so in any field is literally off the abstracting services of anyone, and it's a big problem.

The second problem is, while I've been out here I've been kind of working the venture capital circuit to see if I could actually raise some funding to actually start a business for providing software prior art and services. I'm going around looking for $10 million, it's a very expensive process to keep track of everything. And I happen to be good at it, I mean if anyone else was going to do it, I would say it would cost 20 million or more. I've been doing it for 10 years, I have 10 years of leg-work out of the way. It's a very expensive thing.

I know the Defense Department spends $10 million a year just trying to track all of its software, and they haven't had much luck. The DOE, NASA, all the agencies have not had much luck. It's a very difficult process of tracking it all.

And I'll give you a case that will explain why it will be difficult when the re-examination process opens up: In the signal processing world there's a technique called a "Fast (40 A) Transform, it is used all over the place. In my life I've seen 200 different implementations of this one algorithm, and as an algorithm it's not very complex to begin with, three or four indented loops where some math goes on. When I do my service, I have to look at these 200 algorithms and figure out which four or five I'm going to include in my database. To make those type of decisions in all the different aspects of software is very complicated. You need the type of person who's not trained anywhere. I mean, I didn't go to school to learn this, I had to look at the stuff over the years. So the types of decisions that have to be made in these re-examination processes are very complex, require tons of data, and I'm not sure it even can be done, but I think you're going to try your best anyway.

COMMISSIONER LEHMAN Thank you very much.Next, I'd like to call George Cole.

--oOo--

GEORGE COLE, ESQ.

MR. COLE: Good afternoon. The first problem area I would like to address is the paucity of adequately-educated patent examiners. I realize, and I presume that we are familiar enough with the problem, I don't have to detail it. A hundred and sixty examiners are simply not enough to examine thousands of patents every year, to keep up with the literature and expanding number of technical fields to remain abreast of continually changing and adapting common law, and in addition, to eat, sleep, get lunch and a few things like that.

The problem is particularly acute in the area of software-related invention, which is why we're here today. It's so bad there's a serious possibility that the current approach will be abandoned perhaps in favor of alternatives such as privatizing the process. We could try giving it to all

the people in Ann Arbor where their database is to give them a trade secrets database of computer inventions, which would then serve as the basis for seeing whether or not a right to sue existed.

Frankly, I'm in favor of improving the review process not weakening or abandoning it. It costs society far less to have a determination made during an impartial review process than through adversarial patent litigation. What the PTO needs, what we need the PTO to have is more people with the knowledge needed in the patent review process. We need the process to continue to keep the current legal and technical standards, to improve on them, particularly the technical standard, particularly in fast-advancing fields such as computer software.

Yet, we face an inherent problem in the current system that will keep us from solving it by just adding more examiners. That's because to become a patent examiner you need a scientific degree, a Bachelor's degree, and the legal education. Virtually all who have such dual qualifications pick them up in that order -- science first, then the law. Very few do it in the opposite order which is what I did, law first and then a science. In my case a Master's in computer science at Stanford in '87, six years after I had a law degree from Michigan. The order most prospective examiners gain their dual backgrounds matter, and I'll get back to it in a moment.

The old approach would have been simply adding more examiners, this is necessary, and I would suggest that with some creativity on the part of government and business, we can find the money to do it. I mean, suppose every patent issues gave the government a four percent royalty. But this is not going to be an effective solution overall. It doesn't use the advances in communications that we have, it doesn't react to societal trends.

For one thing, there are often far more rewarding opportunities available to people who have dual training. For example, if you want to practice in Washington, D.C., you're giving up the weather that you've got outside today in favor of an East Coast Winter.

COMMISSIONER LEHMAN   I should note; you know, we don't require them to be a lawyer to be a patent examiner. Maybe we should, but we don't. All you have to have is the technical background, so you don't really have to have the dual training.

MR. COLE: Don't need the dual training. And then when the review process comes in and the examination goes through.

The other problem is -- I'm sorry, let me continue with this line for a moment. Another approach would have been to abandon, as people suggest, giving patents in computer software. If the U.S. was the only country where that was going to have an effect, it would be something that makes sense. But since it is not the only country where software patents can issue, and only larger companies can realistically and consistently press for patents abroad, and then by international treaty enforce them in the United States, this is going to put us at a severe disadvantage.

Our most valuable resource in software is in intellectual creativity, and it's going to end up possibly completely stifled by this. As the companies go outside, enforce the patents there, get them there and then come in and squelch it here.

One prior speaker announced how that happened in a slightly different field, where they came away from the area where they do not afford protection to the U.S. I would rather not see that happen the other way around.

Another approach would be, we get rid of the legal and technical evaluation that goes on. Now, patent examiners may not have to have a legal background, but you're going to have to pay some attention to what the courts are doing along the way, or in a challenge there is going to some review of what the current state of the law regarding software algorithms is. The courts aren't going to come up with a solution in the near future, I suspect.

A real problem in a way is that the background that the patent examiners have is a Bachelor's and this just isn't enough nowadays. It used to be enough, it used to be just like a basic college degree was enough to be on the leading edge of our society, but by the time the science makes its way from the lab to the educators into the curriculum and into the students, it's dated. It's going to be maybe five years behind the times, and the undergraduates aren't going to be able to keep up with the leading edge because they're still too busy picking up the fundamental basis so they can understand the leading edge.

So we have that delay built in, and it's just going to get worse if you sit there and ask the patent examiners to pick up a graduate degree. They're going to add more years to the time, but that's also going to add the pressure for them to go elsewhere. So how do we provide the benefits of this process that gives both the evaluation, but copes also with an inherent limitation on the number of people that you've got?

What I'd like to suggest; the Patent Office consider finding means to leverage its personnel, just as the federal judiciary can call on special masters, or judges pro tem to extend the effectiveness of the cadre of full-time judges. The Patent Office should look for ways to use the thousands of individuals who have technical advanced education in our society, not as full-time patent examiners working solely for the government, but additional personnel called in and paid for as needed. When you need more computer science people, you get more computer science people. When you need more biomechanical people, you get more biomechanical people. But you don't have to commit to a full-time government career service job. And perhaps you could get the specialist fees paid for by the people who are applying, or as one person suggested, if they're going for a special process, if they want the technical specialty, if they want speed, let them pay for it, but call it in from outside.

The patent examiners job then changes from doing all the work on every application, him or herself that he gets, to managing the process, to getting the patent applications, coordinating and consulting with the technicians, consulting with legal scholars if a complex legal issue seems to appear and then coordinating the results. Is there a concern over maintaining traditional secrecy of the patent process?

I point out that researchers have lab assistants or graduate students, law partners have associates. I suggest, since they've already faced and solved this type of problem in dealing with confidentiality and tracing of information, the Patent Office could find a solution to this.

There a lot of individuals throughout this country who could be called on to assist. Fees could be paid to them, maybe you arrange a tax waiver instead if they'd like that. You've got professors at universities, you've got legal scholars at law schools, researchers in governmental labs, graduate students in science or law who might work on an in internship basis.

With the modern technology available facsimile transmission could allow the PTO to coordinate such efforts throughout the country. The Patent Office then becomes the coordinator rather than the sole worker of the information process and allows them to master the information and the rapid changes that are coming through, not to be overwhelmed.

The second area that I would like to address is some of the problems in the current regulations which attempt to provide protection against inadequately trained individuals serving as patent attorneys. And frankly, this is an area that's probably worse seen in the software area than in many other area.

Section 10.7(a)2(ii) requires an individual who seeks to be registered to practice before the Patent and Trademark Office to establish to the director satisfaction that he or she possesses the legal, scientific and technical qualifications necessary to enable him or her to render applicants for patents valuable service. There is no requirement in the statute or regulations that this competency be maintained. The examination of a would-be patent attorney is only an initial hurdle; once passed, it acts as a lifetime assurance the attorney possesses the necessary qualifications. This may have been enough when science did not advance rapidly, it is not adequate anymore.

Furthermore, the PTO presumes that a patent attorney is competent in the field of science underlying a patent the attorney is prosecuting before the Patent and Trademark Office. It can be left to the client to discover his patent attorney lacked the scientific competency to adequately evaluate or prosecute that patent.

The PTO, though it initially requires the attorney to show the scientific field, does not maintain that information, does not retain it, and does not check it against the application. That protection for the public is abandoned after the first hurdle.

Nationally, the Patent Bar is a uniform bar, you've got to have it that way. But you run into a problem for attorneys who want to keep up a scientific background, who want to study in the field and have to keep up an education for current legal standings. Continuing legal education in most states just does not allow, the course work simply is not there that focuses on a technical side. It's a rare exception to come across it. I was lucky, there have been some in California.

This places an additional burden on attorneys who are trying to keep up the legal and technical background. I am submitting, and it's in the written comment, a proposed amendment to the regulation currently existing that addresses these problems. I urge your attention to it. It tries to balance -- I'm not saying it's a perfect solution -- but it tries to balance the needs that I've addressed.

Do you have any questions at this time?

COMMISSIONER LEHMAN No. Thank you very much. I'll just point out that we do have a substantial number of people with advanced degrees in our Examining Corps. We have at least 38 people with Master's degrees and at least 5 Ph.D.s, maybe more. And obviously they're there to help guide those who don't have that. And we're recruiting for people all the time with more education.

My sense is that our problem -- there's been a lot of discussion, a lot of suggestion that we need to have more attractive compensation structure et cetera, therefore we don't want to be a part of the government. I am not sure that's true. I think our capacity to hire people at up to about $90,000 a year as a patent examiner.

Now, in order to do that they have to have a variety of qualifications, they have to have an advanced degree, they have to be in an area that's been designated hard to get, which generally I think this area is, a variety of things. And then even that includes I think a bonus payment that we give them as well. But my impression is that -- and we're never going to compete, nor should we take from Silicon Valley the folks who can go out there and make -- start companies that will make millions of dollars and create thousands of jobs, we don't want them examining patents.

My sense is that to get a competent technologist who can understand the art here, that probably we ought to be able to get people in that range that I just described. But that's not our most critical problem. Though, it has occurred to me from time to time that maybe one of the, sort of a drastic solution about reinventing government would be that we should be moving the Patent Office to Silicon Valley, would be that we could contract out patent examiners. You know, we could hire law firms to do it, or consulting firms and in various areas.

I have a feeling that we wouldn't automatically improve our quality in doing that, but we certainly have an open mind to all kinds of solutions whatever they may be. But we are steadily working on this problem, and I appreciate the suggestions of people like you. So thanks very much for coming here today.

And let me say, when it comes to drastic solutions like that, you know part of my job is I also am a policy maker, but I'm also in effect the CEO and head of an institution with 5,000 people, and their lives and their families are involved in this, and we're not going to take steps that are going to disrupt these people's lives. You know, I think we have an obligation to -- we have problems with our workforce to do that in a way that's fair to the people who work there.

Next I'd like to call Barry Graham, who is an attorney with the International Federation of Industrial Property Attorneys, or an attorney who I assume is representing the International Federation of Industrial Property Attorneys.

MR. GRAHAM: That's correct.

--oOo--

BARRY GRAHAM, ESQ.

INTERNATIONAL FEDERATION OF

INDUSTRIAL PROPERTY ATTORNEYS

MR. GRAHAM: Good afternoon. As the 22nd witness today and the 47th witness overall, I thank you, Commissioner Lehman, and the members of the panel for your patience and continued interest in all the speakers that

went yesterday and have gone forward today.

Should the United States continue to provide patent protection for software-related inventions?

The answer should be clear; yes, the United States should continue to provide such patent protection. Patent's own software-related inventions are beneficial. The ongoing evolution of patent jurisprudence with regard to software-related inventions is sound and should be allowed to continue.

The United States section of the Federation International De Consul and Propriete Industrial, that is FICPI. And in English it is International Federation of Industrial Property Attorneys, appreciated the opportunity to express its views on the subject of patenting software-related inventions.

FICPI was created on September 1, 1906, as an association of industrial property attorneys in private practice. It's principal aims are: One, to enhance international cooperation within the profession of industrial property attorneys in private practice, promote the exchange of information, and harmonize and facilitate relations between members.

Two, to maintain the dignity of its members and the standards of the profession of industrial property attorneys in private practice on an international scale.

And three, to express opinion with regard to newly proposed international and national legislation insofar as it is of general concern to the profession. The members of the Federation deal generally with all matters in the field of industrial property in the countries in which they practice and in other countries through associates. And especially to the extent permitted by their national laws with A) filing and prosecution of applications for patents and utility models where applicable, trademarks and designs, and the maintenance of such industrial property rights. And B) advising in matters relating to industrial property rights, and those concerning unfair competition, licensing, no-how, and transfers of technology.

FICPI has as its members, the leading representatives of the private practice bar in all major countries of the world. The United States section of the Federation, known as FICPI/US, consists of over 100 U.S. attorneys in private practice who specialize in intellectual property law. The member of FICPI/US, such as myself, come from both small and large law firms.

With respect to software-related invention matters, the members of FICPI/US in their private practices represent small startup companies as well as small, medium and large established companies. Representation includes: obtaining patent protection on software-related inventions, asserting patents on software-related inventions, and defending against the assertion of such patents, as well as general counseling on software-related inventions with respect to U.S. intellectual property law.

For the record, I am Barry Graham, and am a partner in the law firm of Finnegan, Henderson, Farabow and Dunner, in the firm's Washington, D.C. office. My partner Bob Yoches spoke here yesterday.

The following represent the view of FICPI/US, and not necessarily the views of the World Federation, my law firm or myself. The World Federation however, has spoken actively in support of protection for software-related inventions by use of the patent laws.

The views address briefly question No. 4 of the public notice focussing on the patent system; FICPI/US believes, based upon actual experiences various members have encountered in representation of clients in the U.S., that the present framework of the patent system as it has and continues to evolve has and will continue to effectively provide innovation in the field of software by providing adequate protection to software-related inventions.

The patent system promotes innovation by assisting startup companies and establishing themselves by obtaining needed capital to operate and grow, based in part at least on company assets in the form of patents on software-related inventions developed by the companies. These startup companies can then continue their research and development on new software-related inventions, using the startup monies obtained.

For example, and inventor developed a neural network system for forecasting stock price movement, the system uses a math program. After learning that patent protection was possible on software-related inventions and was beneficial, the inventor filed for patent protection. Based on the filing, the patent applicant's business went forward successfully generating income.

The patent system also promotes innovation by helping startup companies and established companies protect their commercial products and thereby promote the development in bringing to the market new commercial products. For example, a small company developed a TV rating system, the company sought patent protection on its new system which included software. The company has now been able to go into the market and compete against established systems such as the well known A.C. Nielsen Company system.

Another example involves a well established company. The company developed a software-based BUS protocol for its line of computer systems, and obtained patent protection on the protocol. The company licenced the protected technology, and has used monies generated from its licensing to fund further activities.

The patent system, vis-a-vis software-related inventions, is not perfect. It has the same weaknesses and problems as does the system in other technologies. Quote/unquote, bad patents issue along with good patents. The PTO's efforts towards improving the Examining Corps and the prior art searching ability of the PTO with respect to software-related inventions are appreciated and encouraged. The PTO's efforts to clarify and refine what it understands as standards of patentability of software-related inventions to be, are also appreciated and encouraged.

Those undertakings coupled with the efforts of the federal courts in providing guidance in the evolution of patent jurisprudence have and will continue to provide a sound patent system for software-related inventions. These efforts will in turn help foster innovation in the ever burgeoning field of software, all to the benefit of the United States. The evolution should be allowed to continue without artificial or quick-fixes to an already adequate patent system.

Furthermore, it should be noted that the United States has with success encouraged countries within the international

community to adopt laws which allow patent protection for software-related inventions. If the United States were now to dismantle its own laws on patent protection for such inventions, our country would lose much credibility within the international community.

On behalf of FICPI/US, I thank you, Commissioner Lehman, the panel members and all of those at the PTO responsible for these hearings, for setting up the hearings and for the time allowed for my comments presented today for the Unites States section OF FICPI.

COMMISSIONER LEHMAN Thank you very much for your compliment, and thank you for coming.

Next I'd like to call Edward Y.W. Lemon, III. He's a software engineer with Network Computing Devices.

--oOo--

EDWARD Y.W. LEMON, III,

SOFTWARE ENGINEER

NETWORK COMPUTING DEVICES

MR. LEMON: Howdy. I would like to thank you for providing me an opportunity to give my comments. My name -- normally, people call me Ted Lemon.

COMMISSIONER LEHMAN I assume that's the way you wanted me to announce you, Edward Y.W. Lemon, III, no?

MR. LEMON: No. It just says that on my passport and voting record.

COMMISSIONER LEHMAN Just Ted, okay.

MR. LEMON: So anyway, I am software engineer at Network Computing Devices, speaking just on my own behalf, not on the behalf of Network Computing Devices.

And I'm basically here to give you a message that I think is very important. I know quite a few software engineers, I know very few that argue in favor of software patents. I believe that we do not need patents on software. Most of us don't want patents on software, and I believe that patents will actually hurt us very badly.

For example, the IBM guy just mentioned that in 1965 -- that was the year I was born, by the way -- they sold 18 billion dollars a year in software. 1965 was 15 years before Diamond vs. Diehr, the first software patent that's widely acknowledged.

COMMISSIONER LEHMAN I think he might have gotten that date wrong. I think he meant -- he said 1962, I think he meant 1992 because I don't think in 1962 -- they said they had a 6 billion dollar research budget, I don't think in 1962 IBM sold 15 billion dollars worth of stuff, or if they did it was just about that. So I think he was -- I think it was probably 1992. I should have probably clarified that point.

MR. LEMON: In any case, I actually have to say that I found that number quite astounding too, because I didn't think that the entire market was that big in 1962, but I was willing to take it.

However, the software market has grown dramatically over the course of the last two decades, and I don't think that that growth can be attributed to software patents in any way.

Let me see if I can come up with a couple of other examples here. As another person mentioned a few minutes ago, the

difficulty in developing software is not in coming up with interesting new innovations. The difficulty in developing software is taking innovations, putting them together and producing a complex system that does what you want. That's a very difficult thing to do, but I don't know of any way that you can really patent it.

And certainly, I would like to see great rewards being given to people for doing that thing, but again, the patent system is not the way to provide those rewards.

Now, on the subject of how patents will hurt us, I can give you a couple of examples from my personal experience. I've worked at four companies in my life, starting with a company back East called New Media Graphics Corporation, and of those four companies three of them have been sued by a company known Cadtrak.

Actually, I am not sure that this actually proceeded to a lawsuit, they may have settled before a lawsuit was made, but I know for a fact that all three of those companies have been approached by Cadtrak, have been told that they were violating or infringing on the Cadtrak patent and have paid substantial sums of money to Cadtrak for the privilege of not being sued essentially.

The Cadtrak patent is widely acknowledged by most people, I've never heard of anybody saying that the Cadtrak patent is an example of a patent that should have been issued. It's a very old patent.

COMMISSIONER LEHMAN Does that mean you think that that Cadtrak patent was an invalid patent?

MR. LEMON: Yeah, I think that it fails the test --

UNMIKED VOICE: We have prior art on it.

MR. LEMON: And in addition to the prior art which this person has mentioned, it clearly fails the test of obviousness. It's based on a simple mathematical principle and there is no other way to do the thing which the Cadtrak patent claims to do. And not only is there no other way to do that, but the thing that you want to do is very obvious, drawing a cursor on the screen.

COMMISSIONER LEHMAN So but the essence of the problem here then is that there's been a statement here that there was a patent issued that didn't meet the test of patentability.

MR. LEMON: Right.

COMMISSIONER LEHMAN And now, in effect it's being used to extort money out of people, and they just, you know, buy into the extortion scheme and then they pay up rather than solve it.

It reminds me a little bit of the old thrillers that you used to see on television when I was a kid about the Mafia holding up the candy store, and people would let that happen, you know, getting protection money out of them. And every once in a while the vigorous prosecutor would come along and the uncorruptible police officer and stop the business. Maybe that's my role to do that --

(laughter -- applause)

MR. LEMON: Well, that would certainly be appreciated.

COMMISSIONER LEHMAN -- but it doesn't necessarily mean that the answer in those days would have been to do away with the law.

MR. LEMON: That's absolutely true.

COMMISSIONER LEHMAN And so as I mentioned, we do have a capacity to take some look at these things ourselves, and maybe we should start doing that more. Anyway, please go ahead.

MR. LEMON: Well, a further example on that subject is another patent which I'm sure you've heard bandied about here before which is the Natural Order Recalc Patent. I don't know that the Natural Order Recalc Patent is obvious, and I think there may exist prior art, but nonetheless, the Patent Office wasn't aware of the prior art. And if there hadn't existed prior art, one might argue that the Natural Order Recalc Patent would be valid.

However, the Natural Order Recalc Patent, is a technique which I personally independently invented, and I'm not saying this to blow up my own ego, I'm just saying that I personally invented it and thought nothing of it, when I had been working in the industry for a year.

It's a very simple concept, and I can't think of any other way that you could solve the problem which is intended to be solved. However, it's complicated enough that I could easily see where it could be granted a patent. And if this is not an example of something like that, then certainly there are other patents which would be valid under the current scheme of things.

Now, the problem with that is that, as I was saying before, software is built of large complicated systems, and these systems require small building blocks like the Natural Order Recalc Patent. The Natural Order Recalc Patent is a trivial part of most of these software systems.

The difficult part of creating software systems is having a user interface that people can use, making sure that it doesn't break when you give it the wrong input, designing a whole system of processes. Many of which one could easily imagine patenting, designing this whole system of processes to produce a final end product.

Now, there are of course parallels in other industries, but the difference is that in the software industry, these processes are not only -- these simple processes that we use to build the systems are processes that one would come up very easily simply in the process of designing the whole system. They're not something where you would have to go out and learn about someone's new technology and incorporate that technology.

To be honest with you, I very rarely do any research at all personally. I never look up prior art in the field. I mean, I read journals to some extent, but in general the journals speak about these systems that I'm telling you about, they do not speak about the small simple techniques.

And so when I create a program, that program is made up of things which have been -- I mean, I have examples here -- which have been patented, and which one could easily argue are patentable. And unfortunately that means that in the process of building this valuable thing, I am subject to being sued by people who have created small tiny things which are of no value.

And that means that in theory at least when I write a program, I have to go research all the little nuts and bolts that I use to build the program. I have to go learn about all

these things. I don't have time to do that. And frankly, I don't think that most people that are working in the software industry in startup companies have either the time or the money to do that.

The result being that there is a -- if software patents become as widely used as patents on things like systems within automobiles or something like that, then essentially the entrepreneurial spirit of the industry will die.

I have this dream that someday I will be able to start my own company and sell software. I happen to be a believer in free software, so the mechanism for that may be a little bit difficult.

(laughter)

But I believe that the way the industry has been in the past, I should be able to form that company and I should be able to make a good living at it. However, if software patents become as prevalent as patents in cars and hardware for that matter, I won't be able to form that company, and I won't be able to make a living. And that is why I do not want to see software patents continue as they appear to be.

And one other thing, I wanted to address a point that somebody else brought up earlier. Just because something is done in software does not mean that the hardware patent is equivalent, or rather just because something can be done in software does not mean that the patent on the equivalent hardware is equivalent. Because the hardware implementation is generally much more useful, and if it isn't, the software will outsell it.

So if you have a patent on something which can be done better in hardware, then by all means do it in hardware. And if you don't have a patent on something that can be done better in hardware, then the fact that it can be done in software will mean that your hardware won't sell. So I don't see any reason why we should be concerned about the fact that hardware and software can do the same things. It's really not relevant.

I think that really concludes what I need to say here. So if you have any questions, I would be happy to answer them.

COMMISSIONER LEHMAN Thank you very much. I appreciate your taking the time to share your concerns with us.

You know, I would like to make a point, -- a number of witnesses ago, made a point that we have, you know, it seems like half the people are lawyers who are here testifying, I'd like to point out that in no sense was our witness list rigged. We put out an announcement, we put it out on this Internet, we tried to make it available to everybody.

So basically what you see in terms of the people who are here are people who have an interest and took it upon themselves to come and share their time with us, and their thoughts with us. And by the way, I really appreciate that. I'm getting paid for sitting here, some of you aren't -- some of you are, some of you aren't. And we appreciate the fact that some of you did take out of your own time and your own busy lives to come here and talk with us.

The world is imperfect, I wish we could sort of drag out all the people that probably had other things to do who might be able to enlighten us, but that's just the way it goes. But I

do think we're getting a pretty good picture, a pretty good cross section of views on this. Even though there are awful lot of lawyers here, we're hearing from a lot of non-lawyers too.

So our next witness is Roger Schlafly of Real Software. You have the real software that really should be patented then I guess.

MR. SCHLAFLY: Yeah, right.

COMMISSIONER LEHMAN The original software.

--oOo--

ROGER SCHLAFLY

REAL SOFTWARE

MR. SCHLAFLY: My name is Roger Schlafly, I am self-employed and a software developer and nobody is paying me to be here.

I have a Ph.D. in mathematics and I have worked for both small and large companies. I have one issued patent, and I have been sued for patent infringement. I have a couple of patents pending, and I have some pending patent litigation.

Most of the discussion so far has come from software companies who've argued about whether software patents are good for the industry, and from patent lawyers who favor patenting everything. There have also been a number of good suggestions for improving the system, but most of these apply equally to nonsoftware patents. Instead, I want to focus on some legal and technical issues related to the scope of software patents.

First of all, I want to say that software patents are not as new as everyone seems to act that they are. The first Patent Act explicitly made processes eligible for patents, and the government's been getting process patents for 200 years. Processes are indistinguishable from algorithms, many process patents are enforced.

For example, the Polymerase Chain Reaction, PCR invention which won the Nobel prize in chemistry last year was a process patent. It was upheld in court and was sold for $300 million. It is a recipe for cooking DNA, but legally it is indistinguishable from an algorithm path.

For an older example, Samuel Morse's patent on the telegraph was upheld by the U.S. Supreme Court in 1853. Morse specifically claimed the system of dots and dashes we now know as Morse code. This was a software patent.

My next point is that many software patents are not algorithm patents, and much of the discussion of the legality of software patents focuses on the patentability of algorithms, but actually many if not most, claim a system or an apparatus or a machine. Even if we could reinstate the anti-algorithm bias of the Benson Decision, it would not eliminate software patents.

Many of these software patents, especially the nonalgorithm patents are legally indistinguishable from traditional nonsoftware patents. Many special purpose electronic circuits and chips are designed using software techniques, and they often have microcoded programs etched onto chips. Nobody is seriously suggesting that patent protection should not be available for electronic gadgets.

I don't see how you could justify protection for a special-purpose circuit and deny protection for software on general-purpose computer that performs the same function.

My next point is that the law does not have to change with new technology. Many people here have argued that software is different from other technology and therefore requires a sui generis protection scheme. I think this is a big mistake.

For 200 years copyrights and patents have served to protect intellectual property without any fundamental change in the law. I only know of two cases where sui generis protection scheme was created as a result of industry pleading that some new technology required it.

One, is plant patents. As you know, it was eventually decided that ordinary patents suffice for animal inventions, so the notion that special patents were needed for plants turned out to be unnecessary.

The other case is The Semiconductor Chip Protection Act, eventually though, it turned out that ordinary copyright law was sufficient to protect complicated chip designs, and this special law turned out to be unnecessary also.

Thus, I think you should reject the notion that the law must change to keep up with changing technology unless there is clear and compelling evidence to the contrary.

One advocate here of the sui generis system was Oracle, and to tell you where they're coming from, I want to quote from their policy on software patents. They say, "New developments influential to the software industry frequently emanate from individuals and small companies that lack substantial resources".

So from this I suggest that here they are one billion dollar company, what they want to do is get their innovation for free and not pay for it.

Okay. My next point is that mathematical algorithms are not distinguishable from other algorithms. The Benson decision tried to make this distinction, most everyone including the Court of Appeals for the Federal Circuit and the recent Arrhythmia decision, agreed that this was nonsense. All computer algorithms are essentially mathematical in character.

I see we have up in the podium the Knuth bible on computer algorithms, and it's impossible to say that some of these are mathematical and some of them aren't. There's just no distinction there. Fortunately, all these are in the public domain.

My next point is that many software patents should have failed the novelty or nonobviousness test. For example, the Benson patent, which was rejected by the Supreme Court in 1972 for being unstatutory subject matter, was for an algorithm for multiplying by 10. It consists of noticing that the decimal number 10 is equal to one-zero one-zero in binary notations. So that on a binary computer multiplication by 10 can be accomplished by two shifts and an add. This trick is obvious to any skilled programer. If anyone had tried to publish the trick as a research result, he would be laughed at.

A lot of patents seem to merely consist of taking some well known method and putting it on a computer. These are not novel. Any method can be put on a computer, that is inherent in the nature of computers, and people shouldn't get patents just for the idea of computerizing something.

Now, I want to talk about some of the specific claims that were in the Notice of Announcement for this hearing. The last one, Invention G seems to be in this category of just computerizing something. It claims an accounting system implemented on a computer. It is obvious that any accounting system can be implemented on a computer. Accounting systems are normally not patentable and it shouldn't make any difference that it is implemented on a computer. So I don't think that claim should be patentable.

Inventions A and B are for mathematical algorithms. Patents for these were thrown out by the Benson decision and reinstated by later decisions. Current practice is to look unfavorably towards these, but to allow them as long as the claim mentioned some hardware. The hardware mentioned might be merely some memory chips. Many older patents have drawn elaborate diagrams of flip-flops and other electronics hardware, but the Patent Office seems to be getting more lenient about this all the time.

It seems to me that the hardware requirement is not serving any useful purpose anymore, it is not required by the law, and does not serve to limit the claims in any meaningful way. It might as well be abolished.

Invention claim C-1 and D-1, directly claim a computer listing. I think this is a mistake and should not be allowed because it confuses patent and copyright protection. Copyright law protects program listings. If someone else sells a similar program, such a claim will not provide any useful guidance as to what constitutes an infringement.

On the other hand, copyright law provides a framework for deciding what plagiarism is. Software patent claims should distinctly claim the invention as with other patent claims.

The alternate version of the claims for that invention, Claims C-2 and D-2 more reasonably describe the software invention. I don't see that it makes any difference whether such a claim directly refers to a computer program, or something more tangible such as a disk. The legal scope of the claim will be the same. The traditional Patent Office tangibility requirement is not accomplishing anything useful here, it should be dropped.

Invention E is a computer data structure, I think the arguments for and against this are the same as for mathematical algorithms. Much as people may not like it, I don't see any basis under current law for rejecting these claims.

Invention F is a computational diagnostic method performed on a computer. If it is a genuinely novel invention, it shouldn't be rejected just because it uses a computer. An MRI device might be in this category. Computer calculations perform a necessary part of MRI scans, but most of the novelty lies elsewhere.

A great many electronic devices use microprocessors with software. But the claim in Invention F is very similar to the claims in the Meyer and Weissman patent which was rejected by the Court of Customs and Patent Appeals in 1982 for unstatutory subject matter. That invention was not really a diagnostic device, but a computer program for inputting medical test results comparing them to a list of known (ontcomps) and applying a simple criterion for determining whether the patient has a problem.

It might be acceptable to grant a patent for some very clever novel way of implementing such a criterion, but the Patent Office should not give a broad patent for something like this. It is completely obvious that there are many criteria to chose from, and that any criterion can be computerized.

In sum, whether anyone likes it or not, it's my opinion that software-related patents will continue to be issued because that's the law. The best we can do, is improve the system by trying to give patents only for truly novel and nonobvious inventions.

COMMISSIONER LEHMAN Thank you very much.

Next I would like to ask Wallace Judd of Mentrix Corporation to come forward.

--o0o--

WALLACE JUDD

MENTRIX CORPORATION

MR. JUDD: Thank you. My name is Wallace Judd, I am president of Mentrix Corporation. I am the holder of a patent that's not for software. I am also a programmer. And even though our software company in Nevada City, California, is very small, the testing and training that we provide impacts over 800,000 people annually in the United States. So even small software companies have a fairly large impact.

I would like to ascertain, are all of you up there lawyers?

Yeah. It's been interesting to me the dichotomy --

COMMISSIONER LEHMAN Sorry, we run the world, it's just -- I think about 90 percent of the presidents are, you know.

MR. JUDD: Well, you know, but this is interesting. It's curious to me because all the programmers --

COMMISSIONER LEHMAN Julius Caesar was a lawyer, you know. The pharaoh was a lawyer, you can't get away from that.

MR. JUDD: Well, all the programmers seem to be opposed to patent for software, and all the lawyers have testified in favor of it. And I think that bodes somewhat inauspicious for my plea to request that you not protect --

COMMISSIONER LEHMAN I think we just heard from one that was in favor of patent protection.

MR. JUDD: Beg your pardon?

COMMISSIONER LEHMAN I think --

MR. JUDD: Mr. Schlafly was in favor it, he was the only one.

COMMISSIONER LEHMAN There were a few others.

MR. JUDD: He's in firmware.

Today, I am going to outline --

COMMISSIONER LEHMAN You make a point, though. There is no question about it that the lawyers seem to very much in favor of patent protection. Companies tend to be somewhat split, and programmers who've testified, though not all, a majority of them have testified against it.

We have noticed that fact.

MR. JUDD: Today I am going to continue to outline some procedural questions with regard to software patents that I believe are essentially unanswerable. Then I am going to show some instances over the past several years in which software patents simply did not work. And finally, I will try

to discuss the disastrous impact of allowing software patents on software development.

The first, the issuance of a patent revolves around several salient points: establishing the time of invention, documenting the invention, and establishing the nonobviousness of the invention to a practitioner in the field.

In the world of software we shall see that all of these I believe are essentially unanswerable. With the physical objects the time of invention is the achievement of the working model, or the creation of a drawing of a working model. In software the time of invention is unknowable. If it is the writing of code exhibiting the claims, what if the code has bugs? Has the invention been discovered? How many bugs are allowable? How major can the bugs be to disallow establishment of the claim? Does the code have to be free of bugs? If so, then DOS 6 wouldn't qualify, and certainly most of my Windows software wouldn't qualify.

In essence, you've got an unanswerable question here; when has the claim been established in software?

Suppose you develop a stark prototype, just the skeleton of a program that demonstrates the claims, how robust must this prototype be? Must it demonstrate all features, the salient features? I can see people rushing down to your offices with two-page executing code sketches making exorbitant claims for their little hack.

Documenting the invention is another issue. What language would programs have to be submitted in? Does the Patent and Trademark Office have to compile the code to execute it? How can an examiner test the claims of the patented code? What if the code doesn't do what it claims to do?

To mount an acceptable challenge do I have to execute the patented code and show it doesn't work?

What if I don't happen to have the compiler for that code? What if the language is unique? What if the code exists only on a virtual machine? How can I then demonstrate that the code does or doesn't work?

Should the Patent and Trademark Office define acceptable languages in which submissions should be made? If so, that will guarantee that most discoveries are years behind the times since many leading edge applications are programmed in languages designed for a special purpose that don't have a wide following.

What's obvious to a practitioner in the field, if I obtain a patent for a software training program which monitors the user's every action, is it obvious that simply by adding a scoring algorithm I have a test? It's obvious if you think about it and yet, patents have awarded independently for testing and training programs.

What level of expertise is exhibited by a practitioner in the field? The degrees of modification to make the one I just suggested are obvious to a computer 101 programmer, and yet the questions is would they be obvious to an examiner?

The Commissioner this morning pointed out to Jerry Fiddler that there are no patents on spreadsheets or word processing programs, and I think that was an appropriate thing to point out. However, in my field there are patents being awarded for training programs, for testing programs, and for help systems all of which are obvious and have been in practice for a number of years.

In fact, apparently this field is getting patent protection where word processors and spreadsheets did not get it. So there's a real problem, there's a real issue here in my particular area.

COMMISSIONER LEHMAN I'd like to make a point about this, and I don't mean this is critical of you, but we're obviously supplying this forum and we're obviously getting the message from various people about frustrations they have and unhappiness they have with the system. But one thing I hope that people will leave here with, those who are still in the room, is the notion that we're not on Mars or someplace. If people are unhappy with the system, you know, we're never more than a letter away, and now even with Internet we're an Internet message, an electronic mail message away.

So I think that if people in the business start to see things that they don't like, like the issuance of patents in this area, even without having to use the re-examination procedure, I'd like to think that people can write a letter to the Commissioner of Patents. And you can be certain that if start getting lots of letters and we get complaints we'll start to look into these things. That as it is, you know, it seems like we sort of have to read about things in the newspaper, or they have to really get disastrous before we know about the problem.

So I just hope that one of the sort of teaching points that can come out of this is that dialogue is a two-way street. We're going to try to have a more open ear, but we encourage people to communicate with us too, when they perceive that things are not going the way they'd like them to go.

I'm sorry to interrupt you, you can have the rest of your time back.

MR. JUDD: That's all right.

Robert Greene Sterne earlier testified that it is the functionality that is the invention when testifying with regard to the distinction between software and hardware, and the fact that there was no essential distinction between the two. And yet, I would like to make an argument for the opposite case.

In fact, if you invented a unique way of trapping mice, you get revenues for 17 years, but you can't protect the notion of trapping mice. If somebody comes along with a better mousetrap, you can't prevent them from trapping mice and licensing their trap if it doesn't violate your method. Copyright protection for software and for circuit designs is adequate. You just don't want somebody to steal your code or your circuit design. Stealing the goal, the objective, the function of the patent is as old as patent itself.

You trap a mouse with a string, I'm going to drown those little suckers, voila, I've got a patent. Okay? You cannot patent the notion of trapping a mouse. And I think the same thing is true here. We don't have a problem with the distinction between hardware and software. That in fact, if you copyright a circuit design, that is effective protection for that particular idea.

Since I have only read reviews of the Compton's patent, please understand that my next remarks are based on hearsay not close analysis of the patent itself. Nonetheless, I believe the issues I raise are germane, whether or not the details of the patent as I present them.

The problem with the Compton's patent is that there claims are so broad as to virtually disallow any other method of index access to a CD ROM. Essentially, they want to patent the idea of trapping mice, not their particular mousetrap.

But as any basic database programer knows, there are dozens of ways to create indexed access to data, whether the data is on a CD ROM, a hard disk, in random access memory or stored in magnetic donuts, the principles are identical.

The fact that the Patent Office would grant a patent on access to a CD Rom simply shows that the examiner doesn't understand the generality of random access storage devices.

Another famous or infamous patent software case is the Apple Microsoft litigation with regard to Windows and the Macintosh Look and Feel. The case I felt was truly ridiculous since the Look and Feel of the Macintosh were established at Xerox Park years before the Macintosh was invented. Yet the litigation sucked up millions of dollars, tens of millions of dollars worth of legal fees per year for a number of years. All you had to do was drive up Pagemill Road, run the old Park examples and you would see a Macintosh system. And yet, apparently nobody did that.

Now, how do I know this? The answer is, I worked at Xerox Park in 1979 and '80, then was hired by Apple to work on the Lisa system, which was the precursor to the Macintosh. At Park, I worked on teaching the Star interface to users, so I am particularly qualified to comment on that particular issue.

At Apple our implicit charter was to emulate the Xerox system. Any programmer with two eyes and an index finger, would have looked at the Star System in the Macintosh and thrown the case out of court, yet it consumed millions of dollars in litigation.

Any person looking at the issue of software patent who is familiar with both Xerox's Star System and Apple's Macintosh would have to conclude that the millions spent on litigation between Apple and Microsoft would have done more good spent almost anywhere else in our society.

UNMIKED VOICE: Here, here.

MR. JUDD: Although, over a hundred million dollars were spent or will be spent on these cases, our society is no better off as a result of this litigation.

Increased cost, if patent law is the standard form of protection awarded software there will be clearly a dramatic increase in defending a software product from infringement claims. Today at lunch, as a matter of fact, I was privy to find out that we are infringing a patent issued in 1992 for a help system. Now, this is a computer-assisted learning support system filed in Tokyo, Japan -- of course, the system that we're infringing it with was invented in 1985, and incidentally had the examiners been familiar with the prior art back to 1981, the help system for Lotus 1-2-3, version 1.0, was a perfect example of this very patent and the series of claims contained therein. So in essence, what's going to happen is a small software company such as I is going to have to hopefully be able to notify the Commissioner, or at the worst case, spend $2,000 having this claim re-examined.

The sheer number of potentially patentable aspects of a computer program would make it prohibitively expensive to research them all. While a computer chip or a tire or a drug may have several different arenas of patent law to research, the number of arenas impacting software are exponentially more. Assuring that your software is free of infringements would require research into database maintenance, disk access, user interface, memory managers, interrupt handling, queueing theory, and literally dozens of other programming issues. Writing a few lines of code would require days of research to see whose code you might have infringed.

Another impact will be in marketing delays caused by a year of uncertainty until everyone comes out of the woodwork who might have invented something remotely related to your program. Suppose software is patentable. What's the optimal low finance strategy for a person such as myself? Obviously it's to sandbag it. File patents 360 days after I've documented a program; then if my claims are allowed, I can sue all the folks who have big marketing bucks in similar programs. It's not an enhancement for society. If you manage to establish priority, you've got their marketing investment already engaged behind your license program.

To summarize, I've shown elements critical for establishing a patent are indeterminate. I've illustrated the problem with patent enforcement, using the Compton's and Apple vs. Microsoft cases, and I've shown several disastrous impacts that can be predicted from widespread use of software patents. As a software developer, I beg you, keep patent law out of software. Don't let legal entanglements destroy the software industry as they did the private airplane industry. Clarify that the protection available to software developers is copyright, not patent.

COMMISSIONER LEHMAN   Our next witness -- we're getting near the end of the day here -- is Russell Brand, Senior Computer Scientist & Product with Reasonings Systems, Incorporated.

Mr. Brand, if you'll bear with me a little bit, I'm going to leave the room for about three minutes and I'll be right back, and if there is any chairing work that needs to be done, hopefully my colleague, Commissioner Goffney, will take that over. So I'll be right back and I hope you'll forgive me for missing your opening part of your remarks. Thanks.

Why don't you proceed.

--oOo--

RUSSELL BRAND

REASONINGS SYSTEMS, INC.

MR. BRAND: Mr. Commissioner, members of the panel, my name is Russell Brand, I'm a Senior Computer Scientist and Product Manager at Reasonings Systems in Palo Alto. I've been a programmer for more than 15 years. I speak only for myself and not for my company.

I'm here today to speak against software patents, as most programmers seem to, and am here primarily to talk a little bit about part of the history that seem to have been lost. Before I start with my prepared remarks, there are enough issues that have been raised in the few hours that I've been in the room that I think are worthy of attention, that I had not considered carefully before coming here, did not realize they were issues.

The system that my company sells is a hundred times larger than the stack of paper you see there on the side. The

system that I write reads source code so the people who own the source code can figure out what it does. Often the source code that we read is a factor of twenty larger than that, written by a team of a hundred people over ten years. If we could in any manner figure out what it did easily, if someone could do that, we'd be out of business. It's hard, even with all the help of the people there, to know everything it does. It would be impractical to find out what patents it infringes. If you could give me a good machine description of every software patent, I don't think, even with my tools, and my tools are the best in the world by perhaps twenty years, I could go through the software in an automated fashion, and find out what patents are being violated.

There have been questions as to whether there are distinctions between numerical and non-numerical algorithms. I think the late Admiral Grace Hopper would turn over in her grave to think that there is no difference and that we couldn't understand the difference. She brought the industry forward perhaps thirty years by the realization that computers could work with characters and could do things that were not fundamentally thought of as mathematical. I learned while sitting in the office that a tool that I wrote in an afternoon about two years ago to help me deal with dyslexia, to fix some of the spelling as I type it, probably violates two patents. There is nothing nonobvious in it. There was a problem; I spent an hour to solve it. Should I stop using it now? Should I rely on the patent having been invalid?

There's been talk about changing the rules to narrow the edges a little bit. You're dealing with programmers as one of the groups best suited to find ways around rules. Working around social rules, working around machine restrictions, that's part of what makes us programmers. You're also dealing with lawyers, who are probably the second best group at working their way around rules, and I imagine that microtuning and managing the procedures is not going to help much. It will buy you six months or nine months, and someone will find the new bugs, and there are more of them who will be looking for ways around it than there can possibly be of you trying to fix up the rules.

In addition to being a full-time programmer, six to ten hours a day, five to seven days a week, I'm also a law student three nights a week. I bring a laptop with me to class; when class gets dull, I work on programming. I have an open lawbook next to my terminal while waiting for compiles. I have determined that it's going to take me only probably five years part-time to learn enough about law to speak intelligently on this, and I imagine to understand what's going on with algorithms would take someone without scientific training ten or fifteen years. All of my free time now goes into understanding legal issues, primarily issues of information privacy, constitutional issues, but also in the patent issues.

There's been questions about how do we find the prior art. If in my field I could get via Internet all the new patents nominally in my field, and could send back by Internet mail, here are the things that we have done ten years ago, here are the articles, check it against my databases online that I use so I know who to cite when I write articles. It would take me a little effort; I could do that, I wouldn't need to

charge anyone to do it because it's a small increment over what I'm normally doing, and it's keeping me aware of the current research, and I imagine specialists in a hundred other fields could do the same thing.

Part of what the users of my system do when they are studying software to find out what's good and what's bad, is they introduce defects and see how many of them are found. We send our system for testing, we put in ten defects; if the testers only find eight of those ten defects and they find a hundred other defects, we can bet there are twenty other defects that weren't found.

At this point there is at least a wide belief that many of the software patents should not have been granted. My statistical study, grabbing patents at random and reading them, is more pessimistic that anyone else's prediction in the room. Nineteen out of every twenty I've read are voidable on at least three grounds.

Perhaps it's time that we start introducing ridiculous patents, like the (Letvin Kirchoff) current law patent into the system and see how many get through. And if more than one percent of them get through, then we should address it as a quality control problem, as we would address a quality control problem in any other industry.

To move on to the history, which is the basis of prepared remarks that I'd like to make, I'm on a number of committees that run conferences, annual conferences large and small for professional organizations. I had a very hard time getting speakers for one of my conferences this year, a state of the art technical conference. More than half of the speakers that I approached said they couldn't speak this year because of patent-related restrictions placed upon them by their company's corporate counsel. In previous years, ten years ago when this series started, no one had any problems. People talked about what they did. This year I lost half of my best speakers. It's going to be another two or three years to find out what they are doing, and so everyone working in that same field isn't going to be able to build on that research as quickly.

One of the speakers, in order to give a talk, managed to hack his internal legal system and get a publication out, such that they started their one-year clock from that date running, and he was able to talk publicly. He had to hack his internal legal system in order to make the information publicly available and allow us to build on his knowledge in the field.

Five groups of my colleagues doing work in cryptographic technique have moved or are in the process of moving their work outside the United States. They say the patent restrictions and the export restrictions in combination here prevent them from doing development, prevent them from doing marketing, prevent them from starting a company. They'll move it to Europe. By the time they finish building it, most of the patents they care about will have expired; they'll bring it back and start selling it.

Two months from now I'll be giving a half-day tutorial at the Computer Freedom and Privacy Conference sponsored by the ACM. The tutorial will be on election fraud. In it we'll talk about some of the techniques that could be used to prevent election fraud. Many of them are cryptographic, and I believe at this point all of them have restrictions based on

patents that would prevent them from being used, and all of them could be constructed from information that was in the literature before any thought of patenting those things came out.

Last year I gave a tutorial at the Computer Freedom and Privacy Conference on privacy of data about individuals, and we talked about what could be done cryptographically to better protect the data. And again, the best techniques are protected by patents and you can't license these patents to use in good and strong ways. The licensing restrictions are not just, we want so much money, but we want to control the way you use the patents. So that level of privacy, a level of a fair voting system, a level that will allow people to speak in an anonymous, safe way, and to prove who they are is held hostage to a patent system that will keep us from entering the next level of participatory democracy, hold it up at least another three years, perhaps another ten.

Lastly, I'd like to talk about what I see as the coming age of defensive patent portfolios. At this point companies get defensive patent portfolios so that they can force other people to cross-license to them. Individual programmers like myself, I don't have such a portfolio. I'll need to join someone who has it so that I can cross-license everything I need so that I can publish. If I have an individual patent, it won't do me any good because I can't build anything without that cross-licensing. So the patent will afford me as an individual developer no protection, but afford the large companies, the companies IBM, AT&T, HP, with giant defensive portfolios the ability to control the new technologies that come out, whether they've invented it or not.

In perhaps a related issue, people in my area tend to think of patents and the software look/feel copyrights at the same time. We look at the history of the look/feel copyright. It was validated by the court to protect video games, a video game named "Scramble," that I enjoyed playing when it first came out. And it's been extended and extended. If we look in the same manner, the first software patent that was granted, Diamond vs. Diehr, it was a computer system, part of process control. The idea was that a statutory bar on numerical algorithms would not prevent it from being part of a combination patent. We have gone from computers stopping you from getting patents, to the computer part being okay, to now anything once you put a computer in it, it's a form. Well, you couldn't copyright a form, it's on a screen; now I can get a look/feel copyright. It's an equation; you can't patent an equation, it's part of a computer program with no physical relation to the world. The rules say you can't patent it, but 1400 such patents were granted. The rules, as they were written, would provide a valuable service; the rules, as they are executed, especially with the giant defensive patent portfolios, do a disservice to developers and to the American public as a whole.

Thank you.

COMMISSIONER LEHMAN   Thank you very much for sharing that with us, Mr. Brand.

Next, Willis Higgins, with the law firm of Cooley, Godward, Castro, Huddleson & Tatum.

--oOo--

WILLIS HIGGINS, ESQ.

COOLEY GODWARD LAW FIRM, PALO ALTO, CALIFORNIA

MR. HIGGINS: Thank you very much, Commissioner Lehman, ladies and gentlemen. I should state first that I'm a partner at the Cooley Godward firm and I've headed the firm's electronics and software patent prosecution practice, but will add the views that I express in my written statement and here at this oral testimony are personal and do not represent the views of the firm or any of its clients.

In general I'm going to follow the order of my prepared statement, but I'm not going to read it into the record. At this point I think you've heard enough discussion on this issue that if I were to read this into the record, that would probably put you all to sleep.

I think that a fundamental issue that merits some discussion is the distinction between the way the copyright system works administratively and the way the patent system works administratively, because I think that difference has very significant policy implications. Of course, the copyright system, being a registration system, there is essentially no administrative record that a court can rely on to decide the cases of nonliteral copyright infringement, the so-called look and feel cases. What that means is that a Federal District Court judge or, even worse, a Federal District Court jury is put in the position, to a certain extent, of the patent examiner and has to take a look at the copyrighted work, other similar earlier copyrighted or noncopyrighted works, the prior art, if you will, and then attempt to distill what's the contribution of the copyrighted work in question, and all of this with nothing in the record as an aid to determining what is the proper scope of this copyrighted work and has that been infringed in the nonliteral similarities between the copyrighted work and the alleged infringing work.

On the other hand, in the patent system at least the courts start with a headstart. There is an administrative record, there is a testing of the alleged inventive subject matter against some prior art and an attempt to define in the patent claims what the scope of that work is. And for that reason, I think any attempt to eliminate patents for software is going to cause a lot of problems by throwing these difficult questions back into the copyright arena, and the courts have struggled for years to devise tests that will make the determination of the scope of these copyrights easy, and they simply haven't been able to do it, and the nature of the intense in fact inquiries required to make those judgments I think means that they're just not going to be able to come up with easy tests in that area, so that the inherent nature of the patent system and the copyright system say that it's absolutely essential for the patent system to play a significant role in the protection of American intellectual property.

I think, and I've developed that theme further in two publications that are attached to my prepared statement. But the second area I'd like to talk about, I haven't developed in those publications, it's developed to some degree in the statement, and that concerns the implications to be derived form the institutional practices in the Patent Office and what conclusions should be drawn from those institutional practices with respect to how the examination of software-related patents are handled. Of course, we all know that the time that examiners can devote to each patent application is limited. The examiners, of course, are

on a form of a quota system in which their performance is measured by disposals. Now what the means is that, if examiners are handicapped with respect to other examiners by extra procedures that don't apply in other art areas, it gets very difficult to provide a good objective measurement standard for their performance.

Fundamentally what I'm talking about here is the whole issue of statutory subject matter and the very complex set of rules that have been developed with respect to that, and, of course, anyone who reads the guidelines, reads the case law, sees that extremely fine distinctions are drawn in the case law and it's very difficult to conclude is this claim on the right side of those guidelines or on the wrong side of the guidelines.

Based on my practical experience, I find that too often in the examination process, most of the effort is directed to testing on the 101 issues and on claim language, and very little attention is devoted in many cases to measuring the contribution against the prior art to determine whether that contribution is new and unobvious. And, of course, given the limited time that examiners have available for each application, if they're in effect forced to deal with these extra issues of statutory subject matter, that means that they're going to have less time available to deal with the truly significant issues of is it new and is it useful and has the new and useful contribution properly been defined in the claims that are before the office.

So the suggestion that I would make is in terms of statutory subject matter is follow the lead suggested by the Supreme Court that essentially the patent system should cover anything under the sun developed by man, and move on to the really significant questions of judging against the prior art. And I think most of the comments, most of the comments that I've heard, and no doubt most of the comments that you've heard earlier are not that the Patent Office is calling it wrong in the statutory subject matter area, but calling it wrong in some cases with respect to prior art that either wasn't available or that wasn't properly evaluated. So that I think is the sum and substance of the second area I wanted to discuss.

The third area I want to talk a little bit about is the form and content of patent applications, although as set forth in the notice of these hearings, that's really going to be more primarily the subject of your next hearing, but I think there are some policy issues here as well.

Again, in determine what kind of form, what kind of format should be used in patent applications and deposits of source code if they're going to be used, what's important, I think, is to make a judgment call on whether a particular form, format or procedure adds value to the process. Does it do something to give us stronger patents. So, for example, requiring applicants to deposit source code in a very rigidly defined microfiche format probably doesn't add a whole lot of value to the process. Why not allow the applicant to deposit the source code, if it's going to be deposited, on disk in machinery to perform so that it can be more easily accommodated in online databases and other machine searchable tools. So again, as an example here, look to the question of what value is being added by the procedures, and I would say that requiring one form over another as opposed to what's the content of what you're submitting doesn't add

value any more than the debate over the fine lines in statutory subject matter adding value.

So in summary, what I would say is that in order to strengthen our patent system, we should cut to the chase, get to the questions of novelty and unobviousness over the prior art, and I think as an interim measure, what we as patent attorneys have to do is assume a proactive stance. We've got our Rule 56 obligations that say that we should call the attention to the Patent Office of prior art that we know about. If we want to obtain good, strong, valid patents for our clients, we've got to do more than that. We've got to go out and look for the prior art and get it in the record so that it's considered and overcome. And if we do that, coupled with increased efforts on your part to develop your own prior art databases, then I think the end result will be a much stronger patent system.

Thank you very much.

COMMISSIONER LEHMAN   Thank you very much, Mr. Higgins, for taking the time to think through all of this and give us your comments.

Finally, our last witness of the day and of the entire two days of hearings is Mr. Joseph Grace of Tetrasoft.

--o0o--

JOSEPH GRACE

TETRASOFT

MR. GRACE: Thank you, Commissioner Lehman, for this opportunity to get some thoughts out on software patenting. I feel like after hearing these comments that -- well, first I should say, I don't have a vested interest in software patents; I have a future vested interest in the sense that I'm starting a company.

I find the system daunting and counterproductive from my perspective, and it keeps me up late at night because it's hard enough starting a company with natural disasters, but when somebody can hold a man-made disaster over your head, i.e., you write some software and somebody says 16 years down the road on their patent that you owe them money -- they're not just flushing your dream down the drain, they're flushing your employees down the drain, they're flushing your customers down the drain and they're flushing your suppliers down the drain. To me, that just doesn't make sense. And even if you do succumb to that, and you don't go the way of spending a fortune on lawyers -- I'll try to mention this only once -- I don't believe that the software industry should subsidize the legal services industry. So, if you don't go down that path, which you won't because it will cost you more money to go down that path even if you're in the right, than it would just to knuckle under to some patent challenge, you still end up spending money you shouldn't have to spend. And that money may be going to a competitor who hasn't even earned it, okay? And this is for a system that basically doesn't work.

If I wrote the software to represent this system, it would be riddled with race conditions, okay? That's a bozo no-no in software design, and it would have deadlocks; that means you're going to court. If you want to know an example of a race condition, you could have two companies --

By the way, I don't think these comments are limited strictly to software. I think you're going to see this problem crop

up in other industries as they start using computers as intellectual development tools, because the computers are going to accelerate their development. The reason we're seeing it in software is because software is already accelerated by using computers.

For example, one of the worst race conditions that you have two companies developing something and spend $10 million on it, seven years of people's lives, whatever. Somebody gets the patent first. The patent system says they deserve the reward for that innovation. I disagree. I think the market deserves the reward of both innovations, and both companies deserve the chance to try to build on their opportunity.

Anyhow that's my background, that's why I'm here today because this stuff keeps me up late at night. I don't think it's such an incredibly complicated issue. From here on out I'm going to try not to give you my perspective, though this will definitely be a slanted presentation. I'm going to try and give you some information that I have, that I think could help you understand why there are so many conflicting attitudes and opinions, and how all these people can be basically telling the honest truth. And this is a little slanted, but with the sole exception possibly of the lawyers, since of all people they have the most incredible vested interest in maintaining a system which subsidizes their industry, even at the expense of the software industry.

The title is "Software Patents and Why They Should Be Abolished." The reason is they are unconstitutional. The Constitution says in Article 1, Section 8, Clause 8, The Congress shall have power to promote -- to promote -- the progress of science and useful arts by securing for limited times to authors and inventors the exclusive right -- and I would put that in quotes because that's a sort of nebulous idea -- to their respective writings and discoveries.

The key word in that is "promote." If the system stops promoting growth, the system is unconstitutional. Now, the lawyers may say, "Well, it's grandfathered in because the Congress already established the patent law." Well, that may be true, but that's a legal perspective and not a practical perspective.

Let me define patents further. The basic idea of patent, as I see it, which I think is pretty accurate, is promoting innovation, which is a factor in business and an important one, over free competition, which is a principle of business, and you don't sacrifice that principle lightly, okay? Whoever came up with the patent idea decided that it was worthwhile to promote innovation, even at the expense of free competition, my basic claim is that was a wise idea and it has served its purpose. But it's now outgrown and outlived its purpose and it's time to move on to productivity without patents, because now they're becoming counterproductive and slowing people down. And the reason basically is because it used to be that patents were few and far between -- I call that sparce, okay? Now our industries have matured enough that the patents, the ideas and the technologies and the innovations are coming very rapidly -- that's dense. You're having people trip all over themselves.

As soon as you get into that situation, there are interdependencies between these patents, so when you add a patent, you're not adding N goes to N+1 complexity,

you're going from N to N times N+1 complexity. That's a fact in computer science; that's called factorial growth. And that's heinous, okay? We're not talking linear growth add one, we're talking factorial growth. If you've got thousands of patents and you multiply those by another thousand in complexity, which is what all the inter-relationships between the patents are, you've got a problem. And that's why you can't keep up with the patents, and that's also why you're not going to be able to, ever.

Well, I'm going to drift through this, because I bet I burn more time than I expect.

I call this the Medusa effect. You've heard plenty of views on those conflicting accounts. Each of these views is like an eye of the Medusa. And Medusa is the Medusa of unfair competition, okay. We're sacrificing free competition for the Medusa of unfair competition. The nature of the system is that its outgrown and outlived its usefulness. It's degenerated into a win-lose industrial paralyzing influence. Basically that's the nature of our patent system. I call this degenerate influence the Medusa effect. It's all the same system but the view is different and conflicting from each eye. Each of our presenters is a different eye. The head is overcrowded with eye stalks, and worst of all, every seeing person in the vicinity of Medusa has genuinely high blood pressure -- that's the entrepreneurs who don't have portfolios of patents, okay?

Living near Medusa makes for a scary, unhealthy and unproductive situation, i.e., unconstitutional. The software industry already suffers from the Medusa effect today, due to the interference of the software patent system. That's like interference between patents except the system is doing it to an entire industry.

I would like to see this situation rectified and simplified, i.e., the elimination of patents from software.

Two challenges: To rectify this situation entails two challenges. The first challenge, of course, is to see the forest for the trees. Or in the case of Medusa and software patent, to see the whole Medusa instead of just some eye stalks. I shall try to solve the problem for you in the first part of my presentation shortly.

The second challenge is to fix the system. To fix the system is left to you; if I had the authority, I'd take care of it myself. This talk, unfortunately, is the limit of my contribution so far, but I have a feeling this will be around for ten years and I'll be back.

Fortunately, by applying -- the principle I'd like you to apply is the "kiss" principle, which is "Keep it simple, silly," and that means get rid of the system. The first challenge is eyeing the Medusa of unfair competition; that's being able to see it. I'd like to read a fragment of a New York Times article by John Markoff. He says:

Critics now say that the system is creating a public policy contradiction. On one hand the Clinton administration is eager to foster competition in telecommunications. On the other, the agency continued to grant 17-year monopolies just as it did when technologies involved at century-long intervals.

That's the crux of the problem. Mr. Markoff identifies the basic dilemma for the patent system: Times change. The tradeoff in sacrifice of free and fair competition to promote

innovation may no longer be a prudent tradeoff. Times change and the patent system has done its job to build an innovative, industrious, technological base. Now the patent system needs to step aside and let commerce generate the rest of its momentum instead of gumming up the works with unnecessary and counterproductive litigation.

I'd like to mention four books -- I think this stuff was covered very well in the Markoff article for an overview. In habit four of the Seven Habits of Highly Effective People, which is "think win-win." Right now we've degenerated into lose-lose-lose. The patent holder loses because the society doesn't grow; its competitors lose because they can't even get into the market; and the customers lose because nobody is delivering service as rapidly as they could have.

The next book is the One-Minute Manager Builds High-Performing Teams. This is by Ken Blanchard. I would like to see some situational management understanding applied to this. He goes by stages as well. In the early stages you need directive, coercive management, and in the most latter stages, you need hands-off management.

And the third book that I think applies is Crossing the Chasm, by Jeffrey A. Moore. He uses a multistage system as well. I think we've left the early adoptive cycle of his technology life adoption cycle, which is where patents are beneficial, and we have entered the early majority part of the cycle, where they start to gum up the works because the industry can maintain its own momentum. That's the basic gist.

I think you should take a look at this: Where we started, where we are today, and where we need to go. And I think you need to look at it in terms of stages, and I think if you do, you'll begin to understand why there are so many conflicting remarks. People are coming at this from different stages, and the software engineers are coming at it from the most current stage. And I think that holds value.

How do you kill the Medusa? You use this understanding as a mirror to look at her, and you slay her. And the weapon I would use to slay her is Akim's razor. That goes by the name of the kiss principle as well, and it's also known in legal circles as necessary and sufficient, and only necessary. And what we have now in the patent system is no longer necessary. Besides which, it's also insufficient.

Thank you very much for this opportunity to talk.

COMMISSIONER LEHMAN Thank you very much, Mr. Grace.

I'd like to thank everybody who has testified over the last two days, even those who aren't here, all those who have come to watch the process. I'd like to think it's a process of open government; I'd like to think it's a process of customer service. I hope you'll help us at the Patent & Trademark Office to improve our customer service by taking the advantage of keeping in touch with us over the weeks and months to come.

I can tell you this: I'm not sure that I'm going to propose next week to abolish the Patent Office, but I can assure you there are going to be some real and substantive changes that are going to come out of this process, and you will see those in the coming months. Some of them will be administrative changes that we can make; some them we can make right away, just simple policy changes. Others require work.

For example, the full potential of the Internet and electronic communications, even if we change our policies with regard to what we can hear from, who we can hear from and what we can get from them require technological improvements at the Patent & Trademark Office that will require an expenditure of capital, it will require money. So some things will be phased in, some things will happen very quickly.

The next category of changes that you're going to see is that we are certainly going to be coming up with some legislative proposals to change the statutory system. For example, were we to decide not to have software patents or to eliminate the Patent Office, I think those would have to be legislative changes and we'll have to get Congress' approval. I'm not suggesting we're going to necessarily propose either of those two alternatives, but we are going to be proposing to Congress some changes which will make the system work better.

So those are two examples of the things we're going to do. The final one is we're going to definitely be more aggressive in the Patent & Trademark Office in not only developing our own legal policies, but in working with the various courts, and the Court of Appeals for the Federal Circuit in particular, in trying to help them develop clearer legal standards and do their part in resolving some of these problems.

So in the next months and certainly during the remainder of this Clinton presidency, you can look forward to a series of changes in the areas that I've just outlined, and you'll be able to have many opportunities as these changes unfold, as the decision-making process unfolds, to give us feedback, to let us know what you think. I am a little scared to say that in this group, because I have a feeling the probably the Internet system and the computer system is going to break down at the Patent & Trademark Office with all of the feedback that we get, but let's try and see how it works.

Thank you very much for coming, and I look forward to continuing to work with you all for the next three years. Thanks.

(Public hearing concluded)

Index to Participant Testimony