Exhibit 1

STEPHEN FEINER

IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION

----------X

MIRROR WORLDS, LLC,

Plaintiff,

vs. No. 6:08 CV 88

APPLE, INC.,

Defendant.

----------X

CONTAINS CONFIDENTIAL PORTION

DEPOSITION OF STEPHEN FEINER

New York, New York

Thursday, January 7, 2010

REPORTED BY: BARBARA R. ZELTMAN
Professional Stenographic Reporter

Job Number: 27001

STEPHEN FEINER

STEPHEN FEINER,
having been first duly sworn by
Barbara R. Zeltman, Notary Public, was
examined and testified as follows:

EXAMINATION BY MR. STEIN:

Q Good morning, Dr. Feiner.

My name is Ken Stein, and I'm going to be asking you some questions today.

Let me start by asking you if you have ever been deposed before?

A Yes, I have.

Q How many times?

A Three times.

Q Three times?

And for each time, can you tell me what it was in connection with?

A The first time was in connection with a case that involved Adobe and Macromedia.

The second time was in a case that involved St. Clair Intellectual Property and Canon.

And the third time was in a

STEPHEN FEINER

case that involved Skyline -- I don't remember their full name -- and Google.

Q And in each of those cases, did you submit an expert report?

A Yes, I did.

Q Did you give trial testimony in any of those cases?

A I gave trial testimony in two of those cases.

Q Which two?

A The first two.

Q And when was the first one?

A When was it? I don't remember. It would have been on the list of prior engagements that I gave to counsel.

Q Was it within the last five years?

A I do not think so.

Q And the second case, when was that?

A The Canon case may have been 2005, '6, '7, something like that.

Again, I do not remember the exact date. It could have been 2004. I don't remember.

Q In the Skyline case --

STEPHEN FEINER

A I believe that was 2007.

Q 2007.

Are all those cases over at this point?

A Those are all over, yes.

Q Were they all patent cases?

A They were all patent cases.

Q What was the subject matter of the Adobe -- was that Adobe v. Macromedia?

A I don't remember if it was Adobe v. Macromedia or Macromedia v. Adobe. This was a situation where they were suing and countersuing each other.

I know I was involved in two cases, one of which they settled before the second one reached the point at which I was to be deposed. And I don't remember whether the two of them were A versus B and the second was B versus A, or if they were both A versus B.

That case involved computer software that embodied a Macromedia patent.

Q And broadly, what was the subject

STEPHEN FEINER

matter of that patent?

A The subject matter of the patent involved aspects of a drawing system.

Q So would it be fair to say that the subject matter was broadly directed to computer graphics?

A Computer graphics and graphical user interfaces.

Q And what was the subject matter of the St. Clair Intellectual Property v. Canon case?

A A set of patents owned by St. Clair that were being asserted against Canon.

Q And just broadly, what was the subject matter of those patents?

A The patents involved certain ways of storing images in digital cameras.

Q Would you broadly characterize the subject matter of that case as related to computer graphics?

A I think it could be characterized as relating to computer image processing and user interfaces for devices such as digital cameras.

STEPHEN FEINER

Q So coming back to the first case, which party did you represent in the Adobe/Macromedia case?
A I was hired by Macromedia. I'm not sure if I actually represented them.
Q And in the St. Clair v. Canon case, which party --
A Canon.
Q Canon.
And the third case, Skyline, was that Skyline v. Google?
A That was Skyline v. Google, yes.
Q And what was the subject matter of that --
A I'm sorry. I --
MR. BROWN: Just let him finish the question. That makes it easier for the court reporter.
Q What party did you represent in the Skyline v. Google case?
A Google.
Q And what was the subject matter of that case?
A That case involved certain ways of

STEPHEN FEINER

representing and processing data in systems such as -- certain ways of processing and accessing data in systems that would represent large geographical areas.
Q Would you characterize the subject matter of that case as relating to computer graphics?
A Yes, I would.
Q I'm sorry if I asked you this before, but did you say you testified at trial in the first two?
A I testified at trial in the first two, yes.
Q I actually might have been better to say this before. I just want to go over a couple of the ground rules before we get started with the questions.
You're probably familiar with this from your prior deposition testimony and probably from talking to your counsel.
You are represented by counsel today?
A I'm not sure whether I am or not.

STEPHEN FEINER

MR. BROWN: We're representing Apple.
BY MR. STEIN:
Q You understand that today you will be answering questions under oath, just as if you were in court before a judge and a jury, correct?
A Yes, I understand.
Q And you know, we need to get a clear transcript. So any time I ask you a question that you feel is unclear or needs clarification, please let me know, and I'll try to make it clear.
Okay?
A Yes, I will do that.
Q And if you answer a question, I will assume you understood the question.
Do you understand that?
A I understand.
Q If at any time you need to take a break, let me know, and I'll try to accommodate your request as soon as I can. If there's a question pending, you may want to finish that first. But otherwise, I'll

STEPHEN FEINER

try to accommodate your request for a break.
Do you understand?
A Yes, I understand. Thank you.
Q Is there any reason that you can think of that you would not be able to give your best and most accurate testimony today?
A No reason that I can think of.
Q Can you please state your full name and address for the record.
A Do you want my home address or my work address?
Q Home address.
A My name is Stephen Keith Feiner, and my home address is 90 Morningside Drive, New York, New York.
Q You have been retained by Apple as an expert in this case, correct?
A That is correct.
Q Are you being compensated for your testimony today?
A Yes, I am.
Q How much is your compensation?
A $600 per hour.
Q Are you being compensated at that

STEPHEN FEINER

Q Can you tell me what classes you have taught at Columbia?

A Certainly.

In my first semester at Columbia, I taught a course on digital logic, a graduate digital logic course, which is basic circuitry and principles that underlie, at a very low level, digital computers.

The second semester, I taught the graduate software engineering course.

The third semester that I was there -- I'm not sure I have this right -- I think I'm actually wrong about the second semester.

The second semester was probably the first semester that I taught my computer graphics course.

And I know that I taught the digital logic course only once, and that was then replaced with the software engineering course.

And I went for a while

STEPHEN FEINER

alternating semesters, teaching one course per semester, the graduate computer graphics course and the software engineering course.

And then there came a time, and I don't remember what that time was, at which I replaced the software engineering course with a user interface design course.

I later taught, at least twice, a graduate course on mobile and wearable computing. And now I pretty much alternate between a user interface design course and a course on 3D user interfaces and augmented reality.

In addition, I pretty much every semester teach project courses, which are ones in which individual students register, everywhere from undergraduates all the way up through graduate students, to work with me and my students on specific projects, sometimes on visual projects, sometimes group projects.

STEPHEN FEINER

They get course credit for that, but it's not a course that actually meets in a room on a regular basis with lectures.

Q Since starting working as an assistant professor in 1985, what areas of research focus did you have?

A The main foci of my research have been on various aspects of computer graphics and user interfaces, including the development of user interfaces that are intended to supplement or replace the interface that one might experience in a graphical user interface to an operating system.

In many cases, my research has addressed ones that take advantage of 3D interaction, although there have been a number of projects that we've worked on that have been more in the 2D realm.

My dissertation work was on the use of AI techniques to automatically design pictures, as I had mentioned before. One of the main themes of my

STEPHEN FEINER

group has been in fact that research.

One of the first projects I worked on in fact at Columbia, and in fact one of the longest research collaborations I've had, was with a colleague of mine who does natural language generation. And we've done a number of projects together that automatically generate combinations of graphics and text to explain stuff in a variety of different domains.

I've also done work, both separately and as well together with some of the things I just mentioned, on mobile user interfaces to computer systems, including wearable ones.

A lot of that work has also concentrated on systems in which the displays are built into things that are more or less like a pair of glasses and that overlay graphics in conjunction, in some cases, with audio on top of what you see normally in the world around you.

STEPHEN FEINER

testifying, as I am right now, with regard to this case.

So here I'm referring to the notion of being able to see spread out, in the kind of spread-out browsing described in the Mander paper, the contents of a stack of items.

Q Do you have a Mac OS X with Leopard?

A I have personally, although I do not use it very much, although it is used by my wife, a Mac. But I'm very sure that it does not have Leopard.

Q Have you ever used this particular feature shown in the figure at the bottom of page Feiner 205?

A I have used -- I do not personally own a Mac that has Leopard on it. I'm trying to think if that machine is in my lab.

I have neither purchased nor received in any way, personally or for my lab, a machine that has Leopard on it.

I have, however, used at times machines owned by my students which I'm

STEPHEN FEINER

pretty sure have included machines that are running Leopard. And I may well have used this feature on one of those machines.

Q Do you recall using it?

A I don't know.

I don't really use their machines all that much. Sometimes I use them in the process of working on a paper, and I may well have played with this, and I may well have not. I don't remember.

Q Do you know the relationship between the figure shown in the left-hand portion of that slide labeled 20 on the bottom of page Feiner 205 and the figure shown in the right-hand portion?

A What do you mean by, "the relationship"?

Q Do you understand that when I say "slide," this looks like a slide from a presentation? And the one at the bottom of the page is labeled 20, correct?

A That is correct.

STEPHEN FEINER

Q And there's two figures at the bottom, you know, within slide 20, in that figure, correct?

A That is correct.

Q To your knowledge, is there a relationship between the figure on the left and the figure on the right?

A I believe that there is a relationship between them, yes.

Q And what is the relationship?

A I believe that both of them show different ways of displaying the items that are in a stack in a doc.

Q Do you know if the user selects which one of these ways it is displayed?

A I do not know.

Q Do you know how to display the one on the right?

A No, I do not.

Q Do you know, when the stack on the left gets too large, whether the system automatically displays the one on the right?

A No, I do not.

Q Other than this particular feature,

STEPHEN FEINER

do you know of any -- strike that.

Other than this particular feature shown in slide 20 on Feiner 205, do you know if Apple has ever implemented piles in any form, as described in the Mander et al. paper, in a commercial product?

MR. BROWN: Objection.

A I am not an expert on Apple commercial products. So I haven't studied Apple commercial products carefully to determine whether or not they have actually implemented piles in them. I don't know.

It would not be fair for me to give you an answer on that without carefully studying all of Apple's commercial products.

Q Can you think of any -- strike that.

Can you think of any reason why a company would not want to implement piles as described in the Mander et al. paper?

A There's many reasons that companies don't implement things that they have

developed. They include issues of trying to incorporate them in a clear and consistent way within the user interface, if we're talking about a user interface facility or a facility that has some kind of user interface.

One reason would be that there might be issues that would need to be very carefully resolved to create an extension that would not break things that people were already familiar with.

So recalling the Mander paper, for example, there's a number of issues that are mentioned. I think I actually discuss some of those in this document here, which are ways about how one manipulates a pile.

Looking at Page 16, Bates number Feiner 203, I have some issues here, such as whether piles are distinctly manipulatable entities, whether you can move a pile by clicking and dragging, or whether you move the document by clicking and dragging.

There's other issues, again based on discussions in the paper, on slide 17 on Feiner 204. And these are all things that if one wanted to make a clean, consistent, understandable user interface, constrained by the fact that there already are products that your company makes whose users will be upset if you break things that they're used to.

One reason for not implementing something would be that you may feel the need to think more carefully about how to make sure it gets incorporated properly.

Another reason things don't get implemented could be that they may take so much code that it would make your operating system too big.

They may take code sufficiently complicated that you don't want to assign people to actually work on those aspects of the operating system when there are other things you may consider

to be more important.

There are many, many reasons that companies, not just Apple, but other companies I can think of whose researchers I've spoken with in some cases, who have gotten the company -- I'm not talking about Apple at this point, but another company -- to incorporate an early version of an operating system release functionality to do something that they really wanted to do, and then have that company, folks who actually worked on the released operating system, turn around and yank it out before the release because it just wasn't at the top of their stack of important things, and they wanted to spend their time on other things and didn't want to include something that would, for example, maybe not be as carefully tested as it would need to be to be put into an operating system, where things just couldn't afford to not work really well.

Q Apart from general considerations that might apply to a variety of different types of software, can you think of any specific reasons why a company would not want to implement piles in their operating system?

A When you say "piles," do you mean piles as described in the Mander et al. paper?

Q Yes.

A And you said outside of which reasons?

Q Outside of general reasons that may apply to any product or feature that may be implemented in a commercial product, can you think of any specific reasons why a company would not want to implement piles in particular in a commercial operating system?

A I thought I gave a specific reason before. And in fact, on the two slides that I just mentioned, I mention a variety of issues that were not just, how do we make something generic seamlessly part of an existing product, but here I was discussing some particular issues having to do with

STEPHEN FEINER

art that would make sense to anyone.

The only use of the term "stream" that I'm familiar with in computer science would be, for example, "stream I/O" in UNIX, which I really don't think this has anything to do with.

Q Does the term "diary" have a special meeting in computer science?

A "Diary" does not have a special meaning in computer science.

Q How about "electronic life"? Does that have a special meeting in computer science?

A That does not have a special meaning in computer science, at least not one that I know of.

Q Well, what does the phrase "diary of a person or entity's electronic life" mean?

A If you'll permit me to elaborate a little bit, thinking in fact electronic life, there are folks doing artificial life in computer science -- it's not work that I do, but I know colleagues of mine who do -- which

STEPHEN FEINER

sounds a little bit like it. But I don't think that has anything to do with that.

I'm trying to think. If someone said, "electronic life," to a computer scientist, I suspect that without any context, some folks would say, "Are you referring to artificial life?" Because again, it's a similar-sounding term.

And as I said, it has nothing to do, I think, with what's in the patents.

Q What does the phrase "diary of a person or entity's electronic life" mean?

A Looking at the specification of the patent, I think that this is referring to the things that happen to a person or an entity. An entity could be a corporation or it even could be a machine, for that matter. Entity is a thing.

"Electronic life," I think, has to do with data. And I think this is basically talking about the notion of -- because it is in the context of this

STEPHEN FEINER

quote, I'm referring to things that are time ordered.

I think it's talking about a set of things ordered in time that correspond to the history of a person who might be the user of a computer or an entity that could be the computer itself or it could be the corporation the computer was involved in.

And we're talking about documents that, in this time-ordered sequence, document the history of that person or entity, as represented on the computer.

As one of ordinary skill in the art who is also familiar with work like that done by William Newman and Mik Lamming and colleagues on systems like PEPYS, which is a rank EuroPARC system that uses what we call Active Badges, which could track a person's location as they went around their organization to the granularity of a room, for example, that was a system that created

STEPHEN FEINER

automatically a diary of what that person did physically as they walked around.

It wasn't really their electronic life. At least that part of it wasn't their electronic life. It was their physically moving-around life.

And I'm trying to remember whether the PEPYS System, which again is something that people would think of when they heard the word "diary," also included documentation of things that people did when they were sitting at their computers. I don't remember offhand right now whether it did.

But you asked me about the meaning of "diary of a person or an entity's electronic life." And I'm trying to think of things that, to a person of ordinary skill in the art, would resonate through that phrase.

Q Did Mirror Worlds' patents relate to the PEPYS System?

A I'm not sure what you mean by,

"relate to."

The PEPYS System certainly created and, I believe, sent to the user by e-mail on a daily basis a diary of what that person did, time ordered chronologically.

So in some sense, I guess insofar as there's a lot of discussion of things that are ordered by time, you could argue that there was some relationship.

But "relationship" is a kind of loose, general word. I don't think it has a very specific meaning. So I'm not sure what you mean by, "relate to."

I'm just keying on that because you asked me about the word "diary." And one thing that pops into my mind when I hear "diary" is that particular system, which is designed to create kind of an electronic automatically-constructed diary of a person's life.

Q How do you know whether something is or is not functioning as a diary of a person

or entity's electronic life?

A If you told me something was a diary of a person's electronic life and it contained no information at all about that person and things that they did when they were doing stuff electronically, then I would claim it was a diary of their life, I guess, is an easy answer.

So this is not a diary of my life. This is not a dairy of my life.

MR. STEIN: Excuse me. He pointed to his can of Coke and a document sitting next to him.

A A can of Coke is not a diary of my life.

The term "diary," in fact, comes from a group having to do with time. "Day," I believe, is part of the root.

And so a diary would usually be something that was time ordered. And so I would expect something that was a diary of my life, electronic or otherwise, would be time ordered.

That's the conventional facts of a noncomputational diary where I allowed an entry, you know, possibly every day, maybe more often, maybe less often.

So I think a diary of an electronic life -- again, this is not a term of the art -- would probably be capitally ordered.

This is not something I've thought about very carefully. So you'll have to forgive me if I say things that I might, when retrospectively and more carefully, I would have to say were spoken too hastily.

But I think a diary of an electronic life would be something time ordered. It would probably be something complete.

But on the other hand, does a diary have to be complete? I think most people's diaries certainly aren't complete, unless the person or entity creating them is incredibly obsessive.

And even then, it probably isn't totally complete.

I don't know if there are standards for something that would be complete.

I know that certainly within the context of the patents and the disagreements about the patents, that there is some notion of, you know, which items are actually part of a stream.

Q Are there any other characteristics that you would look to to determine whether something is or is not functioning as an electronic diary? Strike that.

Are there any other characteristics that you would look to to determine whether something is or is not functioning as a diary of a person or entity's electronic life?

A So I'd want to know about the things that were in that time-ordered sequence or things that were in the diary. Were they things about the entity, or were they things that were completely unrelated to the entity

or person?

So I would expect the things in there to be related to the thing that it is a diary of. If there were things that were not related, then I think they would not be the right things to have in a diary, per se.

I'm trying to think of whether this would mean that the diary would have to be complete. I'm not sure about that.

Because I'm also looking at the context in which the phrase appears, I was not asked to construe what "diary of a person or an entity's electronic life" by itself, meant.

And as I said, I chose that phrase not because I invented it myself, but because it was actually found in the patent specification.

And in fact, looking now at that portion of the specification or at least the portion I quoted of it, that is followed by additional writing in

which it says, "A stream, according to the present invention, is a time-ordered sequence of documents that functions as a diary of a person or an entity's electronic life. Every document created and every document send," with a 'D," to a person or entity is stored in the main stream."

Q And where are you reading from?

A I'm reading from the '227 Patent Specification, Column 4, starting on Line 6, although this is probably a little bit later than that, since the part that begins, "A stream, according to the present invention," is at Line 6 Column 4."

And you should turn to the copy that is appended to my --

Q Just so the record is clear, you were reading from your declaration at Page 5?

A I was reading from my declaration page 5. And now I'm about to actually turn to the specification of the '227 patent, Column 4, to make sure that I get the line numbers right.

And looking at Column 4, beginning with Line 6, there is a chunk of that specification that I quoted in my report between Line 6 and Line 30 that gives more information about that first sentence, part of which I used in the construction.

So it is, in fact, that material following the part that I used in my construction that helps me understand what's meant by the terms that are used in the construction.

And there it says, "Every document created and every document send to a person or entity is stored in the main stream," so it's indicating some notion of completeness.

It's all documents created by the person or entity. All the documents that were sent to the person or entity are all in a main stream.

And this, in part, also talking about the past, talking about the present, talking about the future,

compounded and composed together with material from Columns 5 through 6 of the patent, which also talks about past, present and future, led me to conclude the portion of my definition of 'stream' that refers to the past, present and future.

Q You used the term "patents-in-suit" in Mirror Worlds' patents in the past testimony.

Is your understanding of those terms the patents that are attached as Exhibits B, C, D and E of your declaration?

A When I use that term, I'm referring to the ones attached as B, C, D and E. And those are the ones, at the time of preparing the report, that I understood to be involved in the work that I am doing.

Q And are you familiar with the patents-in-suit?

A I'm not sure what you mean by "familiar."

Q Have you read them?

STEPHEN FEINER

the spec that I quote on my Page 5, it's talked about in the patent in Columns 5 and 6 of the '227 patent specification.

And if I look at those columns to refresh my memory, I'm now looking at Line 55 in Column 5, where it talks about "the timepoint in the stream where new documents are created and where incoming documents are placed." And so this is in the present portion.

And looking a little bit later over here, in Column 6, starting at Line 2, it says, "The system allows you to dial to the future by selecting a future timepoint for a document. The present invention keeps the document until that future time occurs. When the time of a document's timepoint arrives, the reminder document is brought into view and the document enters the present portion of the stream."

So there's this notion of a timepoint that the user can set -- they refer to it as dialing into the

STEPHEN FEINER

future -- at which a document can be placed that isn't the current time, but rather, a time into the future. And that is clearly a property of a stream.

Q Is there a notion in the term "stream" that the time ordering of the documents in the stream is maintained?

A I'm not sure what you mean by that.

Q Is the time ordering of the documents in the stream maintained?

MR. BROWN: Objection.

A Did you mean, does the system maintain that time ordering? Does the system keep that time ordering or remember that time ordering; is that what you meant?

Q Yes.

A Looking at the specification on Column 6, Line 3, "The present invention keeps" -- I'm sorry, I think I have it wrong. One second.

Looking at the specification on Column 4, Line 26, it says, "The stream preserves the order and method of document creation."

STEPHEN FEINER

So I think saying that the stream preserves the order of document creation would indicate that it maintains that order. They say pretty much the same thing.

Q And when a document is added to the stream at a particular timepoint, it is included in the time order at that timepoint?

A When a document is added at a particular timepoint, it is my understanding that it is included in the order at that timepoint because a stream is a time-ordered sequence.

I think when it says "time," it's referring to the time as either the actual time, as far as the computer knows it, or possibly the manipulated time, manipulated using the timepoint that the stream supports that allows a user to create a document with a future time.

So if you created a document that was dated ten days from now, it would go into the stream with that

STEPHEN FEINER

ten-days-from-now timestamp at a future point in the stream, as opposed to at the point in the regular computer's clock time that's right now.

Q Is it your understanding that the software that implements a stream has or maintains three separate and distinct entities? One being the past portion of the stream, one being the current portion of the stream and one being the future portion of the stream?

A It depends on what you mean by "separate and distinct."

I think there is a distinction between past, present and future. And I don't think that the system needs to use some mechanism which, as time changes, you need to actually move a document from one place to another. You move it from a present portion to a past portion, for example.

Q Are you saying that that's not required, or it is required?

A No, I don't think that -- I'm trying

to describe a very heavy-handed, inefficient way of doing something in which one could have three different data structures where, as time moved, something that was in the future now is to be literally copied from one place to another.

I think that one could perhaps implement a stream that way. I don't think it's required that you do that. It probably would be advisable if you were trying to do it efficiently.

So while I think that the stream needs to have past, present and future portions, I think that there are ways that someone of ordinary skill in the art could, by the use of pointers, for example, be able to indicate where the different portions begin and end in a way that would be efficient, as opposed to inefficient.

So I'm being a little careful. When you say, "needs to be separate and distinct," I'm wondering if you're asking whether they need to do it in a

very inefficient kind of way.

Q Can you describe for me how a user would locate a particular document in the stream?

A I would need to know something about the document and what the user's task was.

Q Say a user would like to find a document that it created two weeks ago. How would a user go about finding that document in the stream?

MR. BROWN: Objection.

A I'm trying to see if I can find a picture that might help and a description of it.

I'm looking at Figure 1 of the patent, which includes a widget at the lower left-hand corner of the screen.

One, looking at it right now, which I believe to have a typo in it, because I suspect that someone has mixed up the years on the left and right, since it purports to go from 12-4-96 to 12-6-95, and yet at the bottom, it goes from 2-8-95 to 12-14-96, I believe,

looking at this picture over here, I think someone of ordinary skill would see this as being a widget that might let me control the number of things that I'm actually seeing on the screen and that I could either, depending upon my knowledge of exactly when the document was created -- I think you said two weeks ago?

Q Yes.

A Which is kind of fuzzy. I mean two weeks ago to the second, to the hour, to the day, kind of a couple of days, give or take, depending what I might know about that time, it seems to me -- and again, I'm just speculating right now, although I believe there may be some discussion which I'll certainly look for, if you'd like me to, here in this figure, could use that widget to be able to kind of narrow in.

And depending upon if, for example, the user used this system only once around two weeks ago and maybe made a very small number of documents, he may

be able to find it very quickly that way.

However, if they made only one document, let's say, because they were on vacation and came back for a day and quickly made a document and this appeared again, then if they set the dates to being right before and right after the one document was created, then you may see perhaps only one document over here.

But if there were more documents, then you can move your mouse over the collection of icons, and the picture is showing you dates off to the side, dates at the top of the document representation. And I could then, by moving my cursor over the documents, find one from that particular date.

It's not unlike, for example, scrolling in a chronologically-ordered window in a more classical operating system, looking for something that was done between one time or another,

STEPHEN FEINER

create new levels of hierarchy, creates a folder within one of those folders and place documents inside of it, can then move the documents individually within and between folders and also can move the folders themselves both within and between folders.

And this is a venerable aspect of operating systems both in terms of the user interfaces that they present, and as well in terms of the way in which, at a low level, a program of the operating system calls, the files are represented and manipulated.

Q Folders and directories are the same thing, correct?

A I'm using "folder" and "directory" to mean the same thing.

Q What does it mean to store a file in a directory?

A There are several different ways of interpreting that term. If you said to someone, for example, "Store that file in that directory," pointing to a file, let's say, that you see on the screen, and pointing

STEPHEN FEINER

to a directory whose iconic representation you see on the screen, that would mean that that command could be satisfied in many operating systems by selecting and dragging that representation of that file into the folder.

And that would make it so that you would access it, if you were working from the top of the directory hierarchy down, by descending that qualified named hierarchy to go from the very top down to the place in which you had just put it.

When you say, "What does it mean to store it there," that doesn't necessarily mean, in an efficient implementation, that the actual bits within the file were moved.

They may not be moved on the disk. They may not be moved in a variety other places in the software, but some portion of the data structures representing that directory hierarchy would be modified to indicate the new

STEPHEN FEINER

location, so to speak, of a new way to access that file.

Q And you just mentioned, "a data structure representing a directory."

What did you mean by that?

A That in the operating system, there are data structures at different parts of the operating system, including ones that are used to manipulate the graphical representations on the screen, all the way down to very low-level ones that are used to represent what's actually on the disk and ultimately where things are on the disks whenever the place in which the file was being stored. And indeed, there may be places that aren't yet written to disk.

And this very potentially complex set of data structures determines where things are said to be within the operating system.

Q And what is a data structure?

A "Data structure" is a fundamental term of the art in computer science.

And it refers to a

STEPHEN FEINER

representation of data, typically represented as bits, sometimes organized into bytes and words and strings, which stores information and which typically is created and manipulated by means of software. And that software embodies algorithms.

EVENING SESSION

(6:00 p.m.)

STEPHEN FEINER,

resumed, having been previously duly sworn, was examined and testified further as follows:

CONTINUED EXAMINATION BY MR. STEIN:

Q Can there be two data structures that point to the same document on disk?

A It depends on what you mean by, "two data structures point to go the same document."

I can certainly write a piece of code for you right now in which two data structures point to the same document.

Q And on the bottom of Page 4, what

STEPHEN FEINER

A Yes, I could.
Q So why can't it treat the documents stored in the stream the same way that a conventional operating system creates documents stored in a directory or a subdirectory?
A I'm confused, because a conventional operating system needs to store the documents and not just the higher-level things that kind of give you access to them.
I think I answered your question correctly, but I'm not sure what you're getting at.
MR. STEIN: Let's take a break.
(A brief recess was taken.)
BY MR. STEIN:
Q Can you could turn to Page 6, under subheading B?
A Are you talking about my report?
Q Of your report, yes.
You say in the last sentence of the first paragraph that the term "main stream" had a specific meaning in the

STEPHEN FEINER

context of the patents, as opposed to an ordinary meaning in the context of computer science, correct?
A That's correct.
Q What is the ordinary meaning of "main stream" in the context of computer science?
A Good point. I don't think there is an ordinary meaning. So it says here that it has a specific meaning, as opposed to an ordinary meaning. It doesn't say, "the ordinary meaning." I don't think there is an ordinary meaning in the context of computer science.
The only one I could possibly think of -- and this is off the top of my head -- is only that he was doing main stream research.
I would think that he was not doing anything particularly different from other people who were doing it. So it's not any different from the way in which you would use the term "main stream" in any other art.

STEPHEN FEINER

Q You were just handed Exhibit 6, which is a copy of US Patent Number 6,006,227. It's the same as Exhibit D to your declaration or report.
A I'm looking for it right now and making sure that all the pages are there.
It is a copy of the '227 patent.
Q So looking back at Page 6 of your report and that sentence we just referred to, you state that, "A person of ordinary skill would understand that the phrase 'main stream' in the Mirror Worlds patents refers to 'a stream that stores every document received by or generated by a computer system.'"
Do you see that?
A Yes, I do.
Q If you look at Exhibit 6, Claim 1, in Line 11, it says --
A Hold on one second, please. Claim 1, Line 11, Column 15?
Q Correct.
It says, "A main stream of data

STEPHEN FEINER

units" -- sorry, strike that.
Exhibit 6, Column 15, Line 12 states, "The main stream for receiving each data unit received by or generated by the computer system" --
A Hold on a second. Which line again, please?
Q Twelve. It starts, "The main stream for receiving each data unit received by or generated by the computer system," correct?
A Yes.
Q Why did you change the word "each" to "every" in your report regarding the term "main stream"?
A Because I looked not only at the claims, but I also looked at the specification, where there are a number of places that talk about the main stream.
And looking, for example, at Column 4, Line 8, I see, "Every document created and every document send to a person or entity is stored in a main stream."
Q Is there a difference between using

STEPHEN FEINER

the word "each" and the word "every"?

A It depends on the context in which the words are used.

I think people often use the words for connotations, as well as denotations. And I think that the statement made in Column 4 is a very strong one. It repeats the word "every."

And I believe that the reason that it did that, trying to look into the mind of the author or authors, is that they were trying to make very clear the inclusiveness that every document created and every document sent to a person was to be stored in the main stream.

Q So in your view, would that language in Claim 1 have a different connotation if the word "every" was used instead of "each"?

A I would need to look at it carefully to figure out how I feel about that.

So looking at it right now -- and again, this is the first time I'm

STEPHEN FEINER

actually trying to make that determination -- it seems to me that if I simply replace "each" in both places that it appears or in the first element or in only the first place --

Q Where it says, "The main stream for receiving each data unit received by or generated by the computer system," the "each" in that phrase.

A Right.

Again, just thinking about it right now and not having thought about it carefully before, I think it's less emphatic than "every." And clearly, it's the term used in the claim.

And so if it is for "receiving each data unit received by or generated by," I'm just trying to be careful, thinking about phrases like "each" and "every," which is really used to hammer home "all."

To be really careful, I would need to think a bit more about it.

Q So without thinking more about it,

STEPHEN FEINER

you don't know right now if using "each" versus "every" in that phrase would have different connotations?

A I don't know.

Q If you look at the next term on Page 7, Section C, of your report, you state that, "A timestamp is a date and time value that uniquely identifies each document," correct?

A Yes.

Q What is the basis for your opinion that the timestamp must be a unique date and time value?

A I said that the timestamp must uniquely identify each document.

And my basis for saying that it must uniquely identify each document is based on a variety of things, including parts of the file history, and as well also the specification, in which it tells us that file names are strictly optional.

And given that file names are strictly optional, there's none of the

STEPHEN FEINER

normal ability to refer to a file specifically by a fully-qualified name starting at the group or by means of once it established a particular directory that its in by its unique name in that directory.

And given that, since it's the nature of the operating systems and the way in which one manipulates files that there needs to be some way of being able to refer unambiguously to a file, this seems to be the only thing left to do that.

And as I said, there are many places where the specification makes it very clear that that must be the case.

Let me find some of those places.

Q Is it your opinion that the timestamp must be a unique date and time value?

MR. BROWN: Objection.

A What do you mean by -- I don't think I actually said that. I said, "It's a date

STEPHEN FEINER

and time that uniquely identifies each document."

And in my description of possible ways later on that one might implement a timestamp that uniquely identifies a document, I mentioned the use of additional data in the form, for example, of a counter, let's say, that could be used to be sure that something received a unique timestamp even if it was being created at the exact same time and the exact same date as something else.

Q So is it your opinion that a timestamp does not need to be a unique date and time value?

MR. BROWN: Objection.

A In the context of the patent?

Q Yes.

A I think that a timestamp, in the context of the patent, needs to, in some way, be unique for each document, so that one could not have two different documents that received the same timestamp.

STEPHEN FEINER

Q Does it have to be a unique date and time value?

A I think that the date and time, if you're referring to the actual clock date and time in a functioning of limitation in which two things could be done at the same time, that clearly would not uniquely identify a document.

And therefore, something would need to get done to create a timestamp that identifies documents uniquely to be able to make sure that that timestamp wasn't going to ambiguously be given to more than one document.

Q Then is the answer to my previous question "no"?

MR. BROWN: Objection.

A Could you ask the question again?

Q Does a stamp time need to be a unique date and time value?

MR. BROWN: Objection.

A It depends on what you mean by, "a unique date and time value." That's not what I said over here.

STEPHEN FEINER

One way to create a timestamp would be that either you make sure you could not somehow make two things or receive two things at the same time, in which case you should also make sure that a person having set the time into the future couldn't set it to the exact same time again.

And you could do that -- you know, I opined about this a little bit later on in my report, on various ways you could do things of that sort.

But I think that you could create a timestamp which, taking the date and time and adding additional information, would make that timestamp unique.

Q So when you stated in your report that, "A timestamp is a date and time" -- strike that.

When you stated in your report that, "A timestamp is a date and time value that uniquely identifies each document," you were including the

STEPHEN FEINER

possibility of additional information beyond the date and time?

A I was including the possibility of additional information beyond the date and time, and so I'm qualifying value with date and time. Because clearly, the timestamp needs to actually indicate time and date somehow.

But to make it unique, you would either have to ensure that you're never allowed to use that date and time again once you've used it, which it seems to me ...

For example, if I time tripped into the future and I set the date and time to a particular date and time, and then I went back in the past, or if I time tripped into the future and set the date and time to a particular date and time and then set it again to the same date and time, and I created one thing after setting it the first time and created a second document after setting it the second time, and the system then

STEPHEN FEINER

proceeded to give both of those things the exact same timestamp, that would be problematic.

And so the date and time value would need to have something else. For example, maybe an artificial extension of the time or maybe, as I discussed, the use of some kind of counter in there to be able to distinguish between the two date and time values that, as a user, I might have only specified the exact same name in each case.

I might have said, "Tomorrow at exactly 4:10 p.m. Eastern time," in both cases, in one case before creating a document, and in the second case before creating another document. And the system would need to make sure that those two documents did not have the same timestamp.

And you'll appreciate that I spent some time later on in the document talking about several different possible approaches that someone might do

STEPHEN FEINER

something like this from the standpoint of how it would be implemented, and that there are a number of subtleties, which I think I discussed in my report.

And as well, I of course wasn't being exhaustive in describing some ways in which this could be done.

Q In the example that we discussed earlier in which you described how a user would locate a particular document created two weeks ago, does the user have to know the timestamp assigned to that document in order to locate the document?

A The user in that specific example?

Q Yes.

A The user needs to know what specific example -- we're talking about a time-based search. And I don't think the user needs to know exact bits in the timestamp. I don't see anything indicating that they would need to know that.

I think in that case, they need to know the time at which the document was created.

STEPHEN FEINER

And as I said, you could imagine an example in which, because there weren't a lot of documents created over that time, if there were like, say, only one document, for whatever reason, created during that period of time, they'd simply need to know both the upper and lower bound on the time period, and then maybe they would find that single document.

If they found multiple ones, then maybe they would position their cursor over the objects that they saw on the screen to be able to gain more information about it and figure out which one it was.

Q Moving to Section D, labeled Archiving, or titled Archiving, do you consider yourself an expert in archiving?

A I'm not sure what you mean by "expert."

Q You've been retained as an expert in connection with this case by Apple, correct?

A That is correct.

STEPHEN FEINER

Q Do you have an understanding of what it means to be an expert in this case?

A My understanding is that I need to have at least ordinary skill in the art, as I defined it. And I think I have ordinary skill in the art with regard to the contents of the patents.

Q Do you have ordinary skill in the art with respect to archiving?

A As the term is used in the context of the patents, yes, I believe I do.

Q Do you think you have more than ordinary skill in the art with respect to archiving?

MR. BROWN: Objection.

A I'm not quite sure what that means.

Q Have you done any work in archiving or related to archiving?

A I have certainly archived stuff that I owned, or I had stuff I owned archived for me.

Q Have you done any research into archiving techniques?

A It was not my understanding that a

person needs to have done research to be considered of ordinary skill in the art in something.

But outside of research that I may have done to try to figure out what technology I might want to use for archiving something, no, I have not written papers, per se, on archiving.

Q Have you ever designed any software that implements archiving?

A No, I have not.

Q Can you name any commercially available archiving utilities or software that were available in the 1996 time frame?

A I can think of facilities that were used in the process of archiving, which can also be used for other purposes. And there's a range of ways in which the term is used.

For example, the tar program under UNIX is a program that was available well before the 1996 time frame that can be used to create an archive. It can be used for what people normally would think as being archived.

Q And you have used the tar command in UNIX?

A I have used the tar command in UNIX, yes.

Q When you use the tar command in UNIX to create an archive, does tar require that the original file that you're archiving be deleted?

A Tar was created, I think very wisely, not to require that the file that is being put into the tar file itself be deleted. Because if it did that, all sorts of really nasty stuff can happen if something went wrong.

Q Like what?

A Like if the file that you were creating using tar, the tar file, as I'm referring to it, overflowed the amount of surface that was available and it deleted the file that you were putting into it, you could perhaps even irrevocably lose the file.

Q So did tar copy the file that you were archiving into the archive?

A Tar archiving -- I'm trying to

remember. Tar archives are compressed. I'm trying to remember what options I have for doing the compression. And there are different implementations of tar, understand.

But I have usually used it in a way in which the file that was placed into the tar file created by tar can be, in its representation in tar, smaller than the file was represented on the disk originally.

And so there's this subtlety as to whether that's a copy or -- I mean certainly it's a loss list representation of the file, but it may not be a bit-for-bit copy of the file.

Q And the original file in that particular example remained on the disk?

A The original file remained on the disk.

I would say that usually, in those case, I wasn't thinking of myself as archiving, per se, even though the command is called tar, which technically stands for "tape archive," which is

UNIX's abbreviation.

UNIX names a number of commands in rather confusing ways. One of the most well-known examples is the cat command, which is an abbreviation for catenate.

Typically, if you give multiple files to the cat commands, it catenates. It appends one file after another. And for those people who use cat -- and very few of them use it now -- it was very often used to print on the screen or into another file, just a single file. So it really didn't get used to catenate.

So tar I often used and I do use now to create a file that contains a set of files. I really wouldn't say I was archiving them.

I would often use this to create a file that was smaller than the files were individually that was just a single file and that then I could send to somebody else, for example, through

STEPHEN FEINER

data unit.

Q Can you think of any way of going from the main stream data structure to the data unit itself?

A Again, I could imagine -- and we're talking about in implementation -- that you would have in the main stream structure -- that one approach might be a link list, let's say.

And then the link list would be a link-listed structure, and the structure might contain, say, a timestamp and some kind of pointer which would let me get actually at the data unit as it is stored in permanent memory, or perhaps even if it's stored in nonpermanent memory, if you can have that notion. I mean it's a very general notion. And that's how I would do it.

I would do something in which I had some kind of structure that had the timestamp and that actually had some way of accessing, for example through a pointer or an address, the data unit.

STEPHEN FEINER

And there are certainly other ways in which I might have a big array of data units, but that might not be very general. Maybe, in particular cases, I might have a big array of certain kind of data units.

Q If you look at the bottom of Page 13, the next element, heading G, it relates to a claim element "means for associating each data unit with at least one chronological indicator having a respective timestamp which identifies the data unit."

Do you see that?

A Yes, I do.

Q This element also doesn't recite a means for putting a timestamp into the chronological indicator, does it?

A This claim does not cite a means for putting the timestamp into the chronological indicator, but it does more specifically refer to the timestamp as identifying the data unit.

Q If you look at Page 14 at the bottom, heading H relates to claim element

STEPHEN FEINER

that recites "means for generating a main stream of data units."

Do you understand that a main stream is a data structure?

A I understand that a main stream stores every document on the system that's relevant to the user or entity.

"Data structure" is a word that it's easy to use it to refer to a very small entity, a very small structure.

But sometimes you may have a data structure that points to another data structure that points to yet another data structure that together comprises some larger entity.

So it's not clear to me that I would necessarily say the main stream was a data structure. I may say the main stream is comprised of a bunch of data structures.

For example, since a main stream contains documents, the data structures for the documents may differ from one kind of document to another

STEPHEN FEINER

kind of document.

There may be some kind of other information, perhaps a link list of some sort, which is used in conjunction with the things that elements of that link list point to, which could ultimately be the documents that are part of that main stream.

Q Do you know how to generate an instance of a data structure in software?

A There is more than one way to generate an instance of a data structure in software.

Q Can you name one?

A Sure.

One way to generate an instance of a data structure in software is to have some prototype of that data structure that, you know, would tell the computer how much space needs to be allocated and then make a call to some facility that generates that amount of space and returns the address of that space so I can then set a pointer to

that generated data structure.
Q And how many lines of software code would that take in typical high-level programming?
A If the code includes the definition of the data structure itself, the data structure might be extraordinarily large. It may have many, many pieces each, which, for clarity, might correspond to a single line.
So I would need both a definition of the data structure, sometimes even literally stored in a completely different file, included as a header file, for example.
And then I would need a line that would actually do the allocation if we're talking about something that involves a set of different data structures, a link list, let's say, that points to a variety of data structures, each of which represents a document and perhaps has a timestamp in it.
I recall that a main stream, of course, each stream needs to have a

past, present and a future.
I mentioned one way to do that would be to have pointers that pointed into where the past turned into the present and the present turned into the future. Those points would need to be set.
And while you might have one single line of code in a nicely-written program that, at the top level, was called to create that set of data structures to create the main stream, the code that actually did the creating might have quite a large number of lines of coding, including ones for setting the pointers, let's say, if there were pointers being used to differentiate between past, present and future.
Sometimes in situations like that you might have initially when you make a main stream, let's say, from scratch, there might be nothing in it at all.
And then maybe you need to be

sure that things were properly nulled out. Maybe there's a token item that's kind of like the one thing that things point to, so that you don't have to have a special case of having nothing being pointed to at you all, so you'd have to create that.
It really depends on the particular implementation.
At the lowest level, I don't think we're talking about a single line of code, for example. I think we may be talking about many lines of code.
Q You said, "the lowest level." What do you mean by that?
A What I mean is that, for example, how many lines of code does it take to create an integer, an int?
Well, I can declare something to be an int and say, "I've got it." Or I could malik, let's say, allocate an integer, and that's one line of code. But an integer is a very, very simple kind of data structure.

I could allocate a struct, let's say in C, that had a very small number of entries. Like probably to make things look very nice, I'd need a line for each member of the structure.
And then I might make literally a single line of code that called the allocation procedure malik and assigned the address that it returned to a pointer to that data structure. So that would be a very simple example.
I don't think in this case, given how much more complex the main stream is, that a very simple structure, a character or a floating-point number, that we're talking about just a single line of code. I think we're talking about potentially many, many lines of code.
Like how many lines does it take to create a directory in an operating system?
I would suspect that if one were to look at the source code for an

Apple operating system, there would be many, many lines of code involved in creating a directory.

And I would suspect that if one were to look at half of the code that embodied one of these implementations of streams, then we'd also find many lines of code involved in creating a main stream.

Q If you could turn to Page 18, and referring to the paragraph in the middle, the heading N, "Document Organizing Facility" --

A Hold on. I also want to take a look at the '427 patent.

Okay.

Q You state in the middle the phrase, "'Document organizing facility' encompasses any possible means of performing the function of 'document organizing.' It includes a public library and the desk of an IRS worker or just about any office worker in America."

Do you see that?

A Yes, I see it.

Q Do you believe that the term

"facility" in the claims of the '427 patent are referring to a public library?

A This is being a little bit facetious over here. But the point is that I'm not sure -- it says, "A document organizing facility receiving documents automatically associating respective selective indicators" --

I'm trying to see other occurrences of "document organizing facility" in this claim.

So I think the point that's being made here is that I don't think that the term "document organizing facility" by itself is really a sufficient structure. And that, you know, one needs to actually look into the specification to see what is being referred to.

Q But you don't believe the claim is literally referring to the desk of an IRS worker, correct?

A I agree, yes.

Q And you don't literally believe that

the claim is referring to a document organizing facility as a public library, correct?

A That is correct.

Q And you go on to say that, "A document organizing facility encompasses any number of computer software processes and programs running on computers, including, for example, Windows Explorer and Mac OS Finder," correct?

A Correct.

Q How is Windows Explorer a document organizing facility?

A In Windows Explorer, I can see a list of documents. I actually can see a list of documents. I can see a graphical representation that isn't actually literally a list.

I can change the way in which the documents are organized. So I can sort them by date from most recent to least recent, from least recent to most recent. I can sort them by type, I can sort them by name, again in either

direction.

And I can also use Windows Explorer to change the name of a document, to move it up and down inside the hierarchy. I might cut a document and then move up in the hierarchy in Explorer and paste it somewhere else or down in the hierarchy and place it somewhere else or even into another copy of Windows Explorer and place it somewhere else.

I can do the same kind of thing in Mac OS Finder.

Q How do you know if something is a document organizing facility?

A Well, that's a good question.

Given that very general term, "document organizing facility," it seems to me that it's not a term of the art. It's clearly being used in the patent, and I would give it probably a kind of ordinary interpretation that it referred to something to organize documents.

And you kind of know what a